Formal Verification of Braking while Swerving in Automobiles

Aakash Abhishek
aakashme@umich.edu

Harry Sood
hsood@umich.edu

Jean-Baptiste Jeannin
jeannin@umich.edu

University of Michigan

ABSTRACT

Many vehicle accidents result from collision with foreign objects. Automatic and provably safe collision avoidance systems are thus of prime importance to the automobile industry. Previous work on formally verifying car collision avoidance maneuvers typically only focuses on braking-only or swerving-only maneuvers. In this work, we study combined braking and swerving maneuvers and establish formally verified conditions under which safety from collision is ensured. One major constrain in performing such joint maneuvers is that a vehicle's tires have limited traction which can be used either for braking or swerving. So in essence, a combined maneuver can trade off braking ability for turning when it is advantageous to do so and vice-versa. In this work, we study the full continuous range of combined maneuvers, from maximal turning with little braking to maximal braking with little turning.

We use a unicycle model with Ackermann's steering for the car's motion, and the circle of traction forces to model the trade-off between braking and swerving. Resulting vehicle kinematics are formulated as a hybrid program in *differential dynamic logic* d$\mathcal{L}$. We use the automated theorem prover KeYmaera X to formally verify the correctness of the collision avoidance property. This verification provides a mathematical guarantee that a given maneuver can prevent the car from collision with obstacles under certain conditions. The employed method is generic with a purely symbolic model and, thus, can be applied to verify other types of collision avoidance systems exhibiting richer behaviour.

CCS CONCEPTS

• **Software and its engineering** → **Formal software verification**; • **Computing methodologies** → *Model verification and validation*; • **Computer systems organization** → *Robotic autonomy*.

KEYWORDS

Formal Verification, Non-linear Hybrid Systems, Collision Avoidance, Automotive Control

ACM Reference Format:
Aakash Abhishek, Harry Sood, and Jean-Baptiste Jeannin. 2020. Formal Verification of Braking while Swerving in Automobiles. In *23rd ACM International Conference on Hybrid Systems: Computation and Control (HSCC '20), April 22–24, 2020, Sydney, NSW, Australia.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3365365.3382217

HSCC '20, April 22–24, 2020, Sydney, NSW, Australia

1 INTRODUCTION

One of the leading causes of vehicle accidents is on-road collision with other objects. Hence the automotive industry is interested in developing technologies that assist the driver in such emergency situations, e.g., automated collision avoidance systems, advanced driver assisting systems, etc. These systems either guide the vehicle to safety, or issue safe advisories for the driver to follow. Internally, they utilize vehicle response models and path planning controllers.

Although vehicle response models and path planning controllers for collision avoidance have been extensively studied, e.g. [8, 10, 11, 15, 26, 29], the implementation of these collision avoidance systems involves interaction between cyber systems (high level path planning controllers) and physical systems (vehicle's motion). This interaction, combined with increasing complexity of such systems, calls for extreme caution during their design. Furthermore, due to the safety critical nature of such systems, there is a need for mathematical validation of collision avoidance before their deployment.

In this paper, we extend our previous work [1] on formally verifying a swerving maneuver for obstacle avoidance. We formally verify a generic collision avoidance system which issues safe advisories to an automobile moving in a planar scenario. The system is generic in the sense that it can issue a whole class of advisories ranging from swerving-only to braking-only, as well as *any possible combination of swerving and braking*. The formally verified collision avoidance system provides a mathematical guarantee of safety under a set of well-defined conditions. We have modeled the vehicle's behaviour using a unicycle model conforming to pure Ackermann steering [13]. For simplicity, the obstacle has been modeled as a static point object in the vehicle's plane of motion. Our collision avoidance system is a discrete controller issuing discrete safety maneuvers to the vehicle. This results in piece-wise, continuous vehicle kinematics. Furthermore, the dimensions used in this work have been kept symbolic to increase its applicability.

The piece-wise, continuous nature of the overall kinematics of the vehicle and the required task of formally verifying a cyber-physical system calls for a hybrid program-based modeling of the problem. Additionally, the specification of collision avoidance can be easily formulated as an equivalent safety property of the developed hybrid program for verification. In this work we develop such hybrid programs and formally verify the safety property of no-collision under a set of well-defined preconditions on system variables, thereby verifying the collision avoidance system. Since our overall kinematics involve piece-wise, continuous differential equations, we have used *differential dynamic logic* d$\mathcal{L}$ [19] which is intuitive in handling such continuous dynamics, to develop the hybrid program-based model of our collision avoidance system. We then utilize the d$\mathcal{L}$ theorem prover KeYmaera X [7] to perform the formal verification of our model.

Challenges. The central challenge of this work arises from the relatively generic nature of our vehicle kinematics model. This model

conforms to a non-linear hybrid program based modeling for our collision avoidance system. Furthermore, the general solution of the developed vehicle kinematics model involves transcendental functions, namely trigonometric and exponential functions. Due to the undecidable nature of such functions, the amount of machinery applicable for the task of formal verification becomes greatly reduced. To circumvent this, we used a differential-invariant based approach to verify the safety properties of our hybrid program.

Organization. In Section 2, we describe our vehicle kinematics model and discuss its behaviour during different maneuvers. This Section also contains a brief introduction to *differential dynamic logic* d$\mathcal{L}$. Section 3 presents the collision avoidance system, its corresponding hybrid program and the formal verification of collision avoidance during a swerve-only maneuver. Section 4 extends the content of section 3 to include braking while swerving maneuvers in our collision avoidance system model. These sections 3 and 4 also describe our differential invariant based method for formally verifying the collision avoidance system. In section 5 we compare our differential dynamic logic based method of formal verification with other available techniques such as reachability analysis, and discuss their respective applicability and disadvantages for our problem. Finally, we discuss related work in section 6 and conclude this paper in section 7.

2 KINEMATIC MODELING

2.1 Equations of Motion

To develop a kinematic model of the planar vehicle with a rectangular body, we first derive a similar model for a point vehicle. Subsequently we assume that the extended vehicle conforms to pure Ackermann's steering [13] (no sideslip at any wheel). Using this assumption along with the kinematics of the point vehicle (strategically chosen to be at the midpoint of the rear axle), we generate the kinematic model of the extended vehicle. The strategic choice for the location of point vehicle to coincide with the midpoint of rear axle of extended vehicle (point C in Fig. 1), provides a simplification that the heading of the extended vehicle lies parallel to the instantaneous velocity $\vec{v}$ of our point vehicle at all times.

Fig. 1 shows the kinematic diagram of the vehicle. We use variables x and y for the Cartesian position coordinates of the point C, and v for its speed. The instantaneous radius of curvature for the point vehicle is depicted as R, and its heading angle as θ.

(1) - (5) are the resulting equations of motion for the point vehicle.

$$\dot{x}(t) = v(t)\sin(\theta(t)) \tag{1}$$

$$\dot{y}(t) = v(t)\cos(\theta(t)) \tag{2}$$

$$\dot{v}(t) = -\mu g\cos\phi \tag{3}$$

$$R(t) = \begin{cases} \frac{v(t)^2}{\mu g\sin(\phi)} & \text{if } R_{min} \le \frac{v(t)^2}{\mu g\sin(\phi)} \text{ and } \mu \le \mu_o \\ R_{min} & \text{if } R_{min} > \frac{v(t)^2}{\mu g\sin(\phi)} \end{cases} \tag{4}$$

$$\dot{\theta}(t) = \frac{v(t)}{R(t)} = \begin{cases} \frac{\mu g\sin(\phi)}{v(t)} & \text{if } R_{min} \le \frac{v(t)^2}{\mu g\sin(\phi)} \text{ and } \mu \le \mu_o \\ \frac{v(t)}{R_{min}} & \text{if } R_{min} > \frac{v(t)^2}{\mu g\sin(\phi)} \end{cases} \tag{5}$$

The following parameters have been used in the above equations:

- g: the magnitude of the acceleration due to gravity;
- μ: the effective coefficient of friction between the driving surface and the vehicle's tires;
- μ_0: the coefficient of the corresponding static friction; note that we always have $\mu \le \mu_0$ by definition of static friction;
- R_{min}: the minimum turning radius, constrained by the steering geometry of the vehicle;
- ϕ: the angle of braking (Fig. 2).

Fig. 2 shows the circle of traction force for the vehicle [5]. The radius of the circle equals the total traction force, F_μ, generated by the contact between the tires and driving surface. The angle of braking, ϕ, is a parameter that describes the allocation of total traction force, F_μ, between braking force F_B, and turning force F_T. For instance, when $\phi = 0$, all the available force is allocated to the braking force, F_B, and there is no turning. Similarly, when $\phi = \frac{\pi}{2}$, all the available force is allocated to turning force, F_T, and there is no braking. Note that c_1 and c_2 (described below) are constant throughout the swerve and represent $\mu g\cos(\phi)$ and $\mu g\sin(\phi)$, respectively. This means the amount of force allocated for both swerving and braking does not change during the maneuver.

$$F_B = \mu mg\cos(\phi) = c_1 * m \qquad F_T = \mu mg\sin(\phi) = c_2 * m$$

$$F_\mu = \mu mg = \sqrt{F_B^2 + F_T^2}$$

2.2 Assumptions and Behavior

In the rest of this paper (unless stated otherwise), we make the following assumptions:

- The vehicle is approximated to be a rectangle of length, L, and width, W.
- Skidding effect at the tires during any of the maneuvers has been ignored.
- We also make the simplification that the vehicle has no body behind its rear-axle, to ignore the effect of notch formation during swerving. The phenomenon of notch formation has been studied in detail in our previous work [1], and it has been shown to be of the order of a few centimeters for a

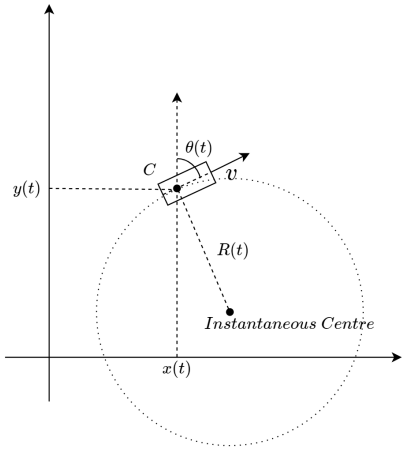

Figure 1: Kinematic Diagram

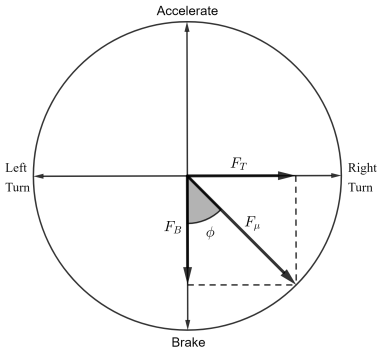

Figure 2: Circle of Traction Forces [5]

standard car. For vehicles with a longer length behind the rear-axle such as busses, the notch becomes more important. However, the formulation of our d$\mathcal{L}$ models and theorems will remain mostly unchanged.

2.3 Differential Dynamic Logic d$\mathcal{L}$

Differential dynamic logic d$\mathcal{L}$ [19] is an extension of dynamic logic with support for differential equations. It supports discrete assignments, implementation of choice and control loops, and execution of differential equations [20–22], making it an appropriate modeling choice for our work. We have used hybrid programs (HP) [20], the language of d$\mathcal{L}$, in our modeling. A brief description of some of the operators of d$\mathcal{L}$ used in our models is given here.

- α^* non-deterministic repetition operator which repeats the program α for zero or more times.
- $x' = \theta \,\&\, Q$: evolves x along the differential equation ($x' = \theta$) for any arbitrary amount of time in the evolution domain Q.
- $p \rightarrow [\alpha]\ q$: says that all executions of hybrid program α starting in a state satisfying logical formula p, end up in a state satisfying q. It is simillar to the Hoare triple $\{p\}\alpha\{q\}$ with precondition p and post-condition q.

For a more comprehensive description of differential dynamic logic d$\mathcal{L}$ and its operators, readers are referred to [19–22].

3 FORMAL VERIFICATION OF SWERVING-ONLY MANEUVERS

3.1 Collision Avoidance System[1]

The collision avoidance system for the swerving-only maneuver is modeled as a discrete controller. By using steering as an input to the vehicle, the controller is able to swerve the vehicle into a circular trajectory. This is followed by a subsequent straight-line motion once the obstacle is passed.

This collision avoidance system is depicted in Fig. 3. Since the swerving-only maneuver applies no force towards braking, effectively the angle of braking, ϕ, is set to 90 degrees. Also the origin of the coordinate system is shifted to the location of the center of turn for simplicity (Fig. 3).

The system's advisories are assumed to be of the following form:

[1]The formal models and proofs described in Sections 3 and 4 of this paper are available at https://jeannin.github.io/papers/hscc20.zip

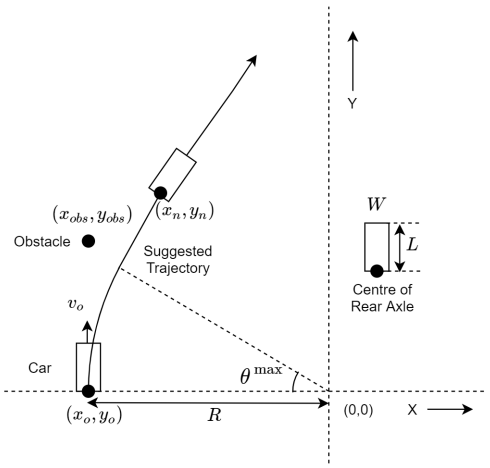

Figure 3: Collision Avoidance System

- (R, θ^{max}). Here R is the advised radius of the turn (measured at point C (2.1) which is represented by (x, y) and (x_n, y_n) in Fig. 3) and θ^{max} is the angle of the turn with respect to the initial direction of travel.
- R is assumed to be satisfying the two constraints of eq. (4): $R \geq R_{min}$ and $v_o^2/R \leq \mu_0 g$
- We assume that the values of $\cos(\theta^{max}) = c_{min}$ and $\sin(\theta^{max}) = s_{max}$ are available to us.
- After turning through angle θ^{max} with a constant speed of $v = v_o$, the car proceeds straight with the same speed. The coordinate system is assumed to be centered at the center of the advised turn.
- $t_o = \frac{R\theta^{max}}{v_o}$, refers to the time instant when car leaves the circular trajectory and starts moving tangentially.
- The advisory is assumed to involve right turn only. This does not result in any loss of generality, since the case of left turn is symmetric to that of right turn.

Using these assumptions, the solution trajectories for Equations (1)-(5), representing the location and heading of the point vehicle are given by Equations (6) - (7).

$$t \leq \frac{R\theta^{max}}{v_o} : \begin{cases} x(t) = -R\cos(\theta(t)) \\ y(t) = R\sin(\theta(t)) \\ \theta(t) = (v_o t)/R \end{cases} \tag{6}$$

$$t > \frac{R\theta^{max}}{v_o} : \begin{cases} x(t) = -R\cos(\theta^{max}) + v_o(t - t_o)\sin(\theta^{max}) \\ y(t) = R\sin(\theta^{max}) + v_o(t - t_o)\cos(\theta^{max}) \\ \theta(t) = \theta^{max} \end{cases} \tag{7}$$

These solution trajectories result in a path called a Dubins Path [4] and our vehicle behaves similar to a Dubins vehicle where its behaviour is restricted to traversing a combination of circular curves and straight lines in a fixed plane, at a constant speed.

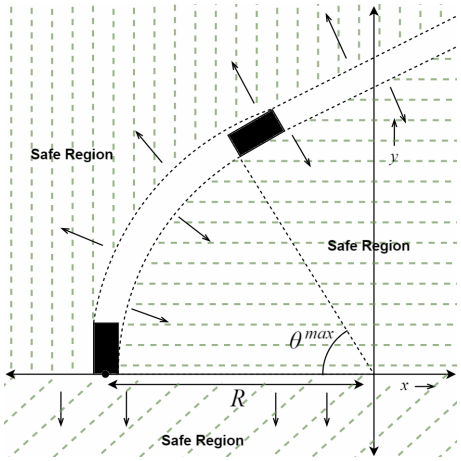

Figure 4: Safety Region

3.2 d$\mathcal{L}$ Model

In developing the d$\mathcal{L}$ model for the collision avoidance system, we have utilized the notion of "safety regions" [12] for a given advisory (R, θ^{max}). The safety region for a given advisory is defined as the set of all possible locations of the obstacle where the advisory results in a no-collision case. Fig. 4 depicts the safety region for one specific case. In general this depiction will look different depending on the dimensions of the car as well as the specific advisory issued. For more details, readers are referred to our previous work [1].

To verify the collision avoidance system, we follow the approach of [12] and use two different (but equivalent) formulations of the safety region: the *implicit* formulation, which is better amenable to formal proofs but cannot directly be checked at run-time because it contains quantifiers; in contrast with the *explicit* formulation, which contains explicit expressions for the safety region boundary but is comparatively less amenable to formal proofs. The approach consists of proving our model with respect to the implicit formulation (it is easier), then proving that our implicit and explicit formulations are equivalent. Therefore, we formulate the following safety theorems, that we make more precise in the next section:

Theorem 1 (sketch):
(obstacle initially in the implicit safety region)
→ [(Turning Dynamics)*] (No collisions in the future)
Theorem 2 (sketch):
(implicit safety region L_{impl_1}) ↔ (explicit safety region L_{expl_1})

The first theorem is a safety property of the hybrid program modeling our collision avoidance system. It encodes the requirement for the collision avoidance system to never let the car collide with the obstacle. The second theorem states the equivalence between two definitions of safety regions, thereby providing sufficient conditions for collision avoidance which can be checked in run-time.

In the following d$\mathcal{L}$ theorems, the following notation is used:

- Point C's location (centre of the rear axle) (Fig.5) is represented by the coordinate (x, y).
- The sine and cosine of the vehicle's heading angle when it is at (x, y) is represented by s and c.
- The time t is the time spent in straight-line motion after the circular turn. t during the turn is set ideally to 0.
- A general point located on the nominal trajectory (solution trajectory) is represented by the coordinate (x_n, y_n).
- The sine and cosine of the vehicle's heading angle at (x_n, y_n) is represented by s_n and c_n.
- The time t_n is the associated time in straight-line motion for the nominal trajectory.
- (x_{obs}, y_{obs}) represent the location of the stationary obstacle.

3.3 d$\mathcal{L}$ Theorem

The following theorems are a mathematical representation of the theorem sketches presented before. The intuition behind the *implicit* formulation of the safety region is: "for any position of the car along its trajectory $(\forall x_n, y_n, t_n, s_n, c_n \ldots)$, no obstacle should intersect the car (a rectangle) at that position." "$(|c(x_{obs} - x) - s(y_{obs} - y)| > W/2 \vee |s(x_{obs} - x) + c(y_{obs} - y) - L/2| > L/2)$". In contrast, the *explicit* safety region explicitly models the safety region drawn on Fig. 4. $case_1 \rightarrow bound_1$ encodes the safety of the bottom half-plane $(y_{obs} < 0)$; $case_2 \rightarrow bound_2$ and $case_3 \rightarrow bound_3$ encode the safety regions surrounding the turning part of the maneuver, while $case_4 \rightarrow bound_4$ encodes the safety region surrounding the straight part of the maneuver. Fig. 5 is provided to help in understanding the geometric meaning of some of the expressions appearing in the *explicit* formulation of safety region.

$$
\begin{aligned}
init_1 \equiv \Big(& v_o > 0 \ \wedge \ l \geq 0 \ \wedge \ w \geq 0 \ \wedge \ R > \frac{W}{2} \\
& \wedge \ c_{min} > 0 \ \wedge \ c_{min} \leq 1 \ \wedge \ s = 0 \wedge \ c = 1 \wedge \\
& x = -R \wedge \ y = 0 \wedge \ t = 0 \Big)
\end{aligned}
$$

Implicit Formulation: Safety Region

$$
\begin{aligned}
L_{impl_1} \equiv \ & \forall x_n, \forall y_n, \forall t_n, \forall s_n, \forall c_n \Big(\\
& \Big((t_n = 0 \ \wedge \ c_n \geq c_{min} \ \wedge \ s_n \geq 0 \ \wedge \ s_n^2 = 1 - c_n^2 \\
& \wedge \ x_n = -Rc_n \ \wedge \ y_n = Rs_n \Big) \\
& \vee \ \big(t_n \geq 0 \ \wedge \ c_n = c_{min} \wedge \ s_n \geq 0 \\
& \wedge \ s_n^2 = 1 - c_n^2 \ \wedge \ x_n = -Rc_n + v_0 t_n s_n \\
& \wedge \ y_n = Rs_n + v_0 t_n c_n \big) \Big) \\
& \rightarrow \Big(\Big(|c(x_{obs} - x) - s(y_{obs} - y)| > \frac{W}{2} \Big) \\
& \quad \vee \ \Big(|\ s(x_{obs} - x) + c(y_{obs} - y) - \frac{L}{2}\ | > \frac{L}{2} \Big) \Big)
\end{aligned}
$$

Explicit Formulation: Safety Region

$$
\begin{aligned}
case_1 &\equiv y_{obs} < 0 \\
bound_1 &\equiv -\infty \leq x_{obs} \leq \infty \\
case_2 &\equiv 0 \leq y_{obs} < L \\
bound_2 &\equiv x_{obs} < -\left(R + \frac{W}{2}\right) \\
&\quad \vee \left[y_{obs} < \left(R - \frac{W}{2}\right) s_{max} \right. \\
&\quad \left. \wedge \left(x_{obs}^2 < \left(R - \frac{W}{2}\right)^2 - y_{obs}^2 \vee x_0 > 0 \right) \right] \\
&\quad \vee \left[y_{obs} \geq \left(R - \frac{W}{2}\right) s_{max} \right. \\
&\quad \wedge y_{obs} < \left(R + \frac{W}{2}\right) s_{max} + L c_{min} \\
&\quad \left. \wedge -x_{obs} c_{min} + y_{obs} s_{max} < \left(R - \frac{W}{2}\right) \right] \\
&\quad \vee \left[y_{obs} \geq \left(R + \frac{W}{2}\right) s_{max} + L c_{min} \right. \\
&\quad \wedge \left(\left(x_{obs}^2 + y_{obs}^2 > L^2 + \left(R + \frac{W}{2}\right)^2 \right. \right. \\
&\quad \left. \left. \wedge -x_{obs} c_{min} + y_{obs} s_{max} > \left(R + \frac{W}{2}\right) \right) \right. \\
&\quad \left. \left. \vee -x_{obs} c_{min} + y_{obs} s_{max} < \left(R - \frac{W}{2}\right) \right) \right] \\
case_3 &\equiv L \leq y_{obs} < \sqrt{L^2 + \left(R + \frac{W}{2}\right)^2} \\
bound_3 &\equiv x_{obs} < \left(-R - \frac{W}{2} \right) \\
&\quad \vee \left[y_{obs} < \left(R - \frac{W}{2}\right) s_{max} \right. \\
&\quad \left. \wedge \left(x_{obs}^2 < \left(R - \frac{W}{2}\right)^2 - y_{obs}^2 \vee x_{obs} > 0 \right) \right] \\
&\quad \vee \left[\left(R - \frac{W}{2}\right) s_{max} \leq y_{obs} \right. \\
&\quad \left. \wedge -x_{obs} c_{min} + y_{obs} s_{max} < \left(R - \frac{W}{2}\right) \right] \\
&\quad \vee \left[x_{obs}^2 > \left(R + \frac{W}{2}\right)^2 + L^2 - y_{obs}^2 \right. \\
&\quad \left. \wedge -x_{obs} c_{min} + y_{obs} s_{max} > \left(R + \frac{W}{2}\right) \right]
\end{aligned}
$$

$$
\begin{aligned}
case_4 &\equiv \sqrt{L^2 + \left(R + \frac{W}{2}\right)^2} \leq y_{obs} \\
bound_4 &\equiv -x c_{min} + y s_{max} < \left(R - \frac{W}{2}\right) \\
&\quad \vee -x c_{min} + y s_{max} > \left(R + \frac{W}{2}\right) \Big)
\end{aligned}
$$

$$
L_{expl_1} \equiv (\wedge_{i=1}^{4} (case_i \rightarrow bound_i))
$$

The safety property representing collision avoidance has been formulated in and proved in KeYmaera X. Proving this theorem, in turn proves the correctness of the collision avoidance system.

$$
\begin{aligned}
dyn_1 \equiv \; & \Big(\big(\dot{s} = c\frac{v_0}{R},\ \dot{c} = -s\frac{v_0}{R},\ \dot{x} = v_0 s,\ \dot{y} = v_0 c \\
& \quad \& \ t = 0 \wedge c \geq \ c_{min} \big) \\
& \quad \cup \ \big(\dot{t} = 1,\ \dot{x} = v_0 s,\ \dot{y} = v_0 c \ \& \ c = c_{min} \big) \Big) \\
no\_collision_1 \equiv \; & \Big(\mid c(x_{obs} - x) - s(y_{obs} - y) \mid > \frac{W}{2} \\
& \quad \vee \mid s(x_{obs} - x) + c(y_{obs} - y) - \frac{L}{2} \mid > \frac{L}{2} \Big)
\end{aligned}
$$

Theorem 1: Verification for Collision Avoidance

$$
init_1 \wedge \ L_{impl_1} \rightarrow \left[(dyn_1)^* \right] (no\_collision_1) \tag{8}
$$

We can now formally prove the equivalence between the implicit and explicit formulations of the safety region, in KeYmaera X:

Theorem 2: Implicit-Explicit Safety Region Equivalence

$$
init_1 \rightarrow \left(L_{impl_1} \ \leftrightarrow \ L_{expl_1} \right) \tag{9}
$$

3.4 Proof Strategy

In order to prove (9), the quantifier $\forall$ must be eliminated by strategically substituting certain nominal coordinates in L_{impl_1} expression. This theorem gives an easier method to verify safety of the collision avoidance system, rather than using the explicit region directly,

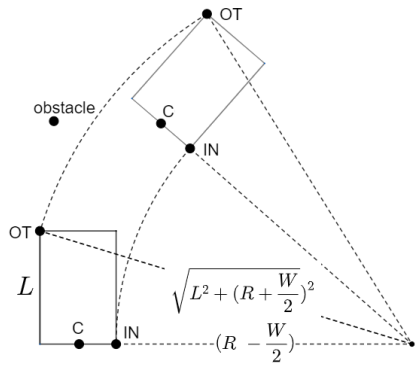

Figure 5: Turning Circle for Swerving-only Maneuver

which would require an evaluation of the safety conditions at every point in the given region. Due to the existence of infinite points in the given region and undecidable algebra in the solutions, this direct, this explicit approach is less plausible.

Classical techniques of proving properties of hybrid program involving differential equations [9], such as (8), require solving the dynamics and subsequently proving the required properties using these obtained solutions. Such techniques can not be used here as the solutions involve trigonometric functions which are arithmetically undecidable. Instead we have utilized a differential invariant based method. This method uses the differential invariants of the turning dynamics of the vehicle to verify its properties. This proof is done in 3 parallel steps. In order to prove (8), we equivalently prove the following:

- The nominal trajectory T_{nom_1} (10) is a differential invariant of the vehicle dynamics. Technically, this means that if the vehicle start from any point on T_{nom_1} and evolves through the dynamic equations for any non-negative time duration, then the vehicle can only end up at a point on T_{nom_1}. This is equivalent to saying that once the car gets on the nominal trajectory, it forever stays on it.

$$
\begin{aligned}
&T_{nom_1} \equiv \\
&\Big(\big(t_n = 0 \ \wedge \ c_n \geq c_{min} \ \wedge \ s_n \geq 0 \ \wedge \ s_n^2 = 1 - c_n^2 \\
&\wedge \ x_n = -Rc_n \ \wedge \ y_n = Rs_n\big) \\
&\vee \ \big(t_n \geq 0 \ \wedge \ c_n = c_{min} \wedge \ s_n \geq 0 \\
&\wedge \ s_n^2 = 1 - c_n^2 \ \wedge \ x_n = -Rc_n + v_0 t_n s_n \\
&\wedge \ y_n = Rs_n + v_0 t_n c_n\big)\Big)
\end{aligned}
\tag{10}
$$

  To prove that T_{nom_1} is a differential invariant of the vehicle dynamics, we observe that differentials of the different relations that define T_{nom_1} (10), are all 0 along the direction of system of differential equations that govern the dynamics.
- The vehicle is on the nominal trajectory T_{nom_1} at $t = 0$. This fact is easily proved by evaluating the relations of T_{nom_1} at $t = 0$ and observing that the initial state of the vehicle satisfies those relations.
- If the vehicle is on the nominal trajectory T_{nom_1}, then it is not colliding with the obstacle. This fact easily follows from the definition of implicit safety region L_{impl_1}.

This completes the proof of our model.

4 FORMAL VERIFICATION OF BRAKING-WHILE-SWERVING MANEUVER

4.1 Collision Avoidance System

Due to the complexity of general solutions for the braking-while-swerving maneuvers (shown in section 4.2), the collision avoidance system is modeled to be much simpler than for the swerving-only case. Here, we restrict the controller to provide only the braking angle ϕ as input. We also ignore the switching behaviour of the controller from the swerving-only case, where it steered the vehicle back to follow a straight line once the obstacle was passed. Rather the vehicle is assumed to follow the same advisory throughout its motion. For further simplification, the vehicle is assumed to be a point object.

4.2 Braking-while-Swerving Maneuver

The braking-while-swerving maneuver is based on the same equations of motion (1) - (5) as the swerving-only maneuver. The key difference is that the braking angle $\phi \neq \frac{\pi}{2}$ and can be varied between 0 and $\frac{\pi}{2}$. Varying ϕ values result in different combinations of braking and swerving, further resulting in different trajectories.

Solving (1) - (5) for a generic fixed value of ϕ we get:

$$c_1 = \mu g \cos(\phi) \qquad\qquad c_2 = \mu g \sin(\phi)$$

$$v(t) = v_o - c_1 t \tag{11}$$

$$\theta(t) = \frac{c_2}{c_1} \ln\left(\frac{v_0}{v_0 - c_1 t}\right) \tag{12}$$

$$
\begin{aligned}
x(t) = -\Bigg\{ &\frac{(v_0 - c_1 t)^2 \left(2c_1 \sin\left(\frac{c_2(\ln(v_0) - \ln(v_0 - c_1 t))}{c_1}\right)\right)}{c_2^2 + 4c_1^2} \\
&+ \frac{(v_0 - c_1 t)^2 \left(c_2 \cos\left(\frac{c_2(\ln(v_0) - \ln(v_0 - c_1 t))}{c_1}\right)\right)}{c_2^2 + 4c_1^2} \Bigg\} + \frac{v_0^2 c_2}{c_2^2 + 4c_1^2}
\end{aligned}
\tag{13}
$$

$$
\begin{aligned}
y(t) = -\Bigg\{ &\frac{(v_0 - c_1 t)^2 \left(2c_1 \cos\left(\frac{c_2(\ln(v_0) - \ln(v_0 - c_1 t))}{c_1}\right)\right)}{c_2^2 + 4c_1^2} \\
&- \frac{(v_0 - c_1 t)^2 \left(c_2 \sin\left(\frac{c_2(\ln(v_0) - \ln(v_0 - c_1 t))}{c_1}\right)\right)}{c_2^2 + 4c_1^2} \Bigg\} + \frac{2v_0^2 c_1}{c_2^2 + 4c_1^2}
\end{aligned}
\tag{14}
$$

Substituting $v(t)$ and $\theta(t)$ from (11) - (12) in (13) - (14), we get the following solutions.

$$x(t) = -\left(\frac{v^2 \left(2c_1 \sin\theta + c_2 \cos\theta\right)}{c_2^2 + 4c_1^2}\right) + \frac{v_0^2 c_2}{c_2^2 + 4c_1^2} \tag{15}$$

$$y(t) = -\left(\frac{v^2 \left(2c_1 \cos\theta - c_2 \sin\theta\right)}{c_2^2 + 4c_1^2}\right) + \frac{2v_0^2 c_1}{c_2^2 + 4c_1^2} \tag{16}$$

Numerically simulating these equations in MATLAB provides the car's trajectory under a braking while turning maneuver (Fig 6), which turns out to be a *logarithmic spiral* (shown later). In this numerical simulation, the values used for different constants are $g = 9.8$ m/s, $\mu = 0.7$, $\phi = 70^o$ and $v_0 = 15$ m/s the initial speed of the car. The features shown in Fig. 6 are explained below:

- The dashed path represents the circular trajectory followed by the swerving-only ($\phi = \frac{\pi}{2}$) system for a full rotation.
- The solid path represents the trajectory followed by the braking-while-swerving ($\phi = 70^o$) until v decreases to 0.
- Each 'x' represents the instantaneous center of turn for the braking-while-swerving at several points along its course.
- 'O' marks the initial center of turn.
- (x_0, y_0) is the initial location of car for both maneuvers.
- 'F' is the final location for the braking-while-swerving maneuver. This is the point where the vehicle comes to a stop.

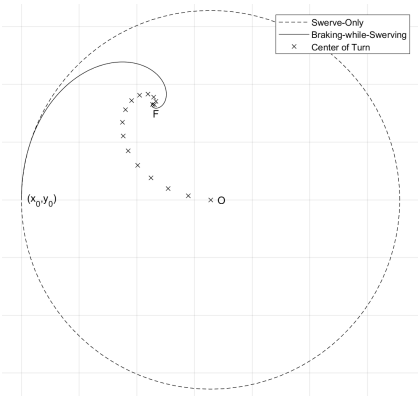

Figure 6: Comparison of Swerving-only and Braking-while-Swerving Maneuvers

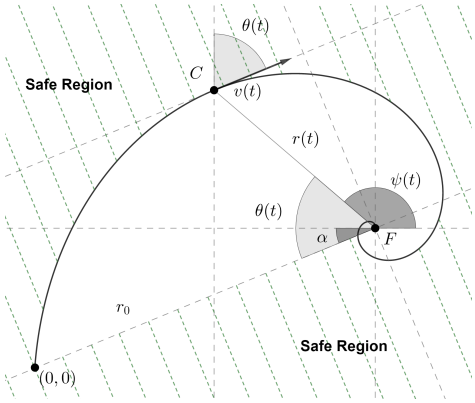

Figure 7: Logarithmic Spiral Trajectory and Parameters

Upon analysing the car's trajectory, we found that it is identical to a logarithmic spiral (Fig. 7). Below are the solutions for $x_C(t)$ and $y_C(t)$ which represent the location of the point vehicle C with respect to the final point F (the location of C when vehicle comes to a full stop (Fig. 7)).

$$x_C(t) = \left[-v(t)^2(2c_1\sin(\theta) + c_2\cos(\theta))\right] \Big/ \left[c_2^2 + 4c_1^2\right] \quad (17)$$

$$y_C(t) = \left[-v(t)^2(2c_1\cos(\theta) - c_2\sin(\theta))\right] \Big/ \left[c_2^2 + 4c_1^2\right] \quad (18)$$

Manipulating (17) - (18) results in the following equation.

$$x_C(t)^2 + y_C(t)^2 = \frac{v(t)^4}{c_2^2 + 4c_1^2} = \frac{v_0^4 e^{-(4c_1/c_2)\theta(t)}}{c_2^2 + 4c_1^2} \quad (19)$$

In order to show that the braking-while-swerving trajectory follows a logarithmic spiral, we must convert $x_C(t)$ and $y_C(t)$ into polar coordinates defined by $(r(t), \psi(t))$. (Fig. 7)

$$x_C(t) = r(t)\cos(\psi(t)) \qquad y_C(t) = r(t)\sin(\psi(t))$$

Furthermore, $r(t)$ can be determined using (19) and $\psi(t)$ can be determined using the resulting equations.

$$r(t) = \sqrt{x_C(t)^2 + y_C(t)^2} = v_0^2 e^{(-2c_1/c_2)\theta} \Big/ \sqrt{c_2^2 + 4c_1^2} \quad (20)$$

$$\cos(\psi(t)) = \left[-2c_1\sin(\theta(t)) + c_2\cos(\theta(t))\right] \Big/ \sqrt{c_2^2 + 4c_1^2} \quad (21)$$

$$\sin(\psi(t)) = \left[-2c_1\cos(\theta(t)) - c_2\sin(\theta(t))\right] \Big/ \sqrt{c_2^2 + 4c_1^2} \quad (22)$$

Here, we define a new parameter, α, which is a constant (Fig. 7). Using α and (21) - (22), we obtain the following equations.

$$\cos(\alpha) = c_2 \Big/ \sqrt{c_2^2 + 4c_1^2} \qquad \sin(\alpha) = 2c_1 \Big/ \sqrt{c_2^2 + 4c_1^2}$$

$$\cos(\psi(t)) = -\cos(\alpha - \theta(t)) \qquad \sin(\psi(t)) = -\sin(\alpha - \theta(t))$$

Finally, $\theta(t)$ can be represented in terms α and $\psi(t)$ (in degrees).

$$\theta(t) = 180 + \alpha - \psi(t)$$

Now that we have obtained $\theta(t)$ in this form, we can show that (20) fits into the form of an logarithmic spiral. This then allows us to rotate our axis (so that it matches the line joining (0,0) and F) and convert the problem to polar coordinates $(r(t), \theta(t))$ and define safety region to be the region outside of the spiral (Fig. 7).

General Logarithmic Spiral: $r = k_1 e^{k_2\theta}$

Our Polar Equation: $r(t) = \dfrac{v_0^2}{\sqrt{c_2^2 + 4c_1^2}}\, e^{-(2c_1/c_2)\theta(t)}$

4.3 d$\mathcal{L}$ Model: Cartesian Coordinates

Similar to the case of swerving-only maneuvers, in developing the d$\mathcal{L}$ model for the collision avoidance system, we have utilized the notion of "safety regions" for a given advisory (Fig. 7). To verify the collision avoidance system, we again use two different (but equivalent) formulations of the safe region: the *implicit* formulation and the *explicit* formulation. Therefore, we formulate the following safety theorems, that we make more precise in the next section:

Theorem 3 (sketch):
(obstacle initially in the implicit safety region)
→ [(Braking while Swerving Dynamics)*]
(No collisions in the future)

Theorem 4 (sketch):
(implicit safety region L_{impl_2}) ↔ (explicit safety region L_{expl_2})

The first theorem is a safety property of the hybrid program modeling in our collision avoidance system, which encodes the requirement for the collision avoidance system to never let the car collide with the obstacle. The second theorem states the equivalence of two definitions of safety regions, thus giving sufficient conditions for collision avoidance which can be checked in run-time.

Due to complexity in the general solutions of our vehicle kinematics model (11) - (14), we make the following assumptions for the collision avoidance system for braking-while-swerving maneuvers:

- Both the vehicle and obstacle are assumed to be point objects.
- The advisory issued by the system is in the form of values of $c_1 = \mu g \cos\phi$ and $c_2 = \mu g \sin\phi$ where both are assumed to be greater than 0. These values in turn dictate the trade-off between braking and turning ability.
- The vehicle continues on the spiral path till it comes to a complete stop. We do realise that such a behaviour is not realistic, and that the equation (12) blows up when v approaches 0, but for the sake of simplicity we ignore this.
- In our d$\mathcal{L}$ model, we encode the sin and cosine of the vehicle's heading angle by variables s and c.
- The origin of the Cartesian coordinate system $(0, 0)$ is fixed at the starting point of the maneuver (Fig. 7).
- In order to encode the exponential behaviour of θ we introduce an auxiliary variable $I = e^{\frac{-c_1\theta}{c_2}}$ and follow its evolution with time by adding its dynamics to our equations of motions. One fact to be noted here is that this auxiliary variable I is a coordinate variable and only the values of $\theta \in [0, \pi]$ are allowed, corresponding to the polar coordinated of the point. Here θ is measured as shown in Fig. 7. Note that the tilted base of the spiral shown in the figure joins the final and the starting point of the vehicle's trajectory.
- We assume that we have the knowledge of the position coordinates of the obstacle $(x_{obs}, y_{obs}, I_{obs})$. We point out the fact that I_{obs} is a function of x_{obs}, y_{obs} and this relation is assumed to hold true before hand from the verification process (explained later). For reference, the relation between x_{obs}, y_{obs} & I_{obs} is given below:

$$\cos(\theta_{obs}) = \frac{-x_F(x_{obs} - x_F) - y_F(y_{obs} - y_F)}{\sqrt{x_F^2 + y_F^2}\sqrt{(x_{obs} - x_F)^2 + (y_{obs} - y_F)^2}} \tag{23}$$

$$\sin(\theta_{obs}) = \frac{x_F y_{obs} - y_F x_{obs}}{\sqrt{x_F^2 + y_F^2}\sqrt{(x_{obs} - x_F)^2 + (y_{obs} - y_F)^2}} \tag{24}$$

$$I_{obs} = e^{\frac{-c_1\theta_{obs}}{c_2}} \tag{25}$$

- x_F and y_F represent the coordinates of the final point of the vehicle's trajectory (stopping point). The coordinates for this point are given by:

$$x_F = \frac{v_0^2 c_2}{c_2^2 + 4c_1^2} \qquad y_F = \frac{2v_0^2 c_1}{c_2^2 + 4c_1^2}$$

- (x, y, I) denote the coordinates of any general point in the plane and (x_n, y_n, I_n) denote the coordinates of a point on the nominal trajectory (solution trajectory). These coordinates also exhibit the relation described in (23)-(25).

4.4 d$\mathcal{L}$ Theorem: Cartesian Coordinates

The following theorems are a mathematical representation of the theorem sketches presented before.

$$init_2 \equiv \Big(c_1 > 0 \ \wedge \ c_2 > 0 \ \wedge \ x = 0 \ \wedge \ y = 0 \ \wedge \ s = 0 \ \wedge \ c = 1 \ \wedge \ v = v_o \wedge \ I = 1 \wedge x_F = \frac{v_o^2 c_2}{(c_2^2 + 4c_1^2)} \wedge \ y_F = \frac{2v_o^2 c_1}{(c_2^2 + 4c_1^2)} \Big)$$

Implicit Formulation: Safety Region

$$L_{impl_2} \equiv \ \forall x_n, \forall y_n, \forall I_n \Big(I_n \leq 1 \ \wedge \ I_n > 0 \wedge (x_n - x_F)^2 + (y_n - y_F)^2 = \frac{v_o^4 I_n^4}{c_2^2 + 4c_1^2} \Big) \rightarrow \Big(I_{obs} \neq I_n \vee (x_{obs} - x_F)^2 + (y_{obs} - y_F)^2 > (x_n - x_F)^2 + (y_n - y_F)^2 \Big)$$

Explicit Formulation: Safety Region

$$L_{expl_2} \equiv \ \left((x_{obs} - x_F)^2 + (y_{obs} - y_F)^2 \geq \frac{v_o^4 I_{obs}^4}{c_2^2 + 4c_1^2} \right)$$

The safety property representing collision avoidance has been formulated below in d$\mathcal{L}$. Only Theorem 3 was formally proved in KeYmaera X. Although Theorem 4 is proved by hand, we were not able to complete its formal proof in KeYmaera X, due to a lack of support for trigonometric and exponential functions (see Section 5).

$$dyn_2 \equiv \ \Big(x' = vs \ \wedge \ y' = vc \ \wedge \ s' = \frac{cc_2}{v} \ \wedge c' = \frac{-sc_2}{v} \ \wedge \ v' = -c_1 \ \wedge \ I' = \frac{-c_1 I}{v} \ \& \ I > 0 \wedge v > 0 \Big)$$

$$no\_collision_2 \equiv \ \Big(I_{obs} \neq I \vee (x_{obs} - x_F)^2 + (y_{obs} - y_F)^2 > (x - x_F)^2 + (y - y_F)^2 \Big)$$

Theorem 3: Verification for Combined Maneuver

$$init_2 \wedge \ L_{impl_2} \rightarrow \big[(dyn_2)\big](no\_collision_2) \tag{26}$$

Theorem 4: Implicit-Explicit Safety Region Equivalence

$$init_2 \rightarrow \big(L_{impl_2} \ \leftrightarrow \ L_{expl_2}\big) \tag{27}$$

4.5 Proof Strategy: Cartesian Coordinates

Considering the involvement of transcendental functions like trigonometric as well as exponentials in the general solution of our braking while swerving dynamics, the classical methods (by invoking direct solutions) of proving properties for hybrid programs involving differential equations is out of question. Hence again we utilize the differential invariants of our dynamics to prove the required safety property. This proof is once again done in 3 parallel steps. In order to prove (26), we equivalently prove the following -

- The nominal trajectory $T_{nom_2}(28)$ is a differential invariant of the vehicle dynamics.

$$T_{nom_2} \equiv \Big(I_n \leq 1 \ \wedge \ I_n > 0 \ \wedge (x_n - x_F)^2 + (y_n - y_F)^2 = \frac{v_o^4 I_n^4}{c_2^2 + 4c_1^2}\Big) \tag{28}$$

  However, unlike the case of swerve-only model, proving that the nominal trajectory $T_{nom_2}(28)$ is a differential invariant of the vehicle dynamics, is not straight forward. A direct differentiation of relations expressed in T_{nom_2}, does not equals to 0 along the direction of system dynamics. We require some auxiliary conditions to hold true, in order for the differentiation of relations expressed in T_{nom_2} to be equal to 0 along the direction of system dynamics. In particular, we utilize the following direct differential invariants of our dynamics:

$$s^2 + c^2 = 1 \qquad \frac{v}{I} = v_o$$

$$(x - x_F) = \frac{-v^2(2c_1 s + c_2 c)}{c_2^2 + 4c_1^2} \qquad (y - y_F) = \frac{-v^2(2c_1 c - c_2 s)}{c_2^2 + 4c_1^2}$$

  To make these auxiliary invariants available to the theorem prover when proving invariance of T_{nom_2}, we use subsequent differential cut (dC) arguments to add them to the evolution domain of our dynamics. Subsequently, differentiating the relations expressed in T_{nom_2}, equates to 0 along the direction of system evolution.
- The vehicle is on the nominal trajectory T_{nom_2} initially. This fact is easily proved by evaluating the relations of T_{nom_2} and observing that the initial state of the vehicle satisfies those relations.
- If the vehicle is on the nominal trajectory T_{nom_2}, then it is not colliding with the obstacle. This fact easily follows from the definition of implicit safety region L_{impl_2}.

4.6 d$\mathcal{L}$ Model: Polar Coordinates

In section 4.3, we had mentioned the fact that between the variables $(x_{obs}, y_{obs}, I_{obs})$ or for that matter between any general (x, y, I), there is an inherent relation (23) - (25), which is assumed to hold true beforehand from the verification process. The reason we resorted to make this assumption was that the equations (23) - (25) involve trigonometric and exponential functions, and at present there is no direct support within KeYmaera X for representing such functions. Due to this same reason, the proof of (27) was skipped in this work.

However, the fact that the general solution trajectories for the braking while swerving maneuver turn out to be a *logarithmic spiral*, point us to the possibility that the polar representation of the problem, albeit being unintuitive to directly obtain, might prove to be simpler as compared to the Cartesian one. For developing the polar representation, we make the following additional assumptions:

- The origin of the polar coordinate system $(0, 0)$ is fixed at the focal point (F) of the resultant spiral trajectory.
- We assume that we have the knowledge of the position coordinates of the obstacle (r_{obs}, I_{obs}).
- (r, I) denote the coordinates of any general point in the plane and (r_n, I_n) denote the coordinates of a point on the nominal trajectory (solution trajectory).

4.7 d$\mathcal{L}$ Theorem: Polar Coordinates

The below d$\mathcal{L}$ theorems formulate the same safety property/theorem mentioned in section (4.4), to verify the collision avoidance of the braking-while-swerving maneuvers in polar coordinates. In contrast to section (4.4), we are able to prove both Theorem 5 and Theorem 6 in KeYmaera X.

$$\begin{aligned}
init_3 \equiv \ & \Big(c_1 > 0 \ \wedge \ c_2 > 0 \ \wedge \ k > 0 \ \wedge \ k^2 = c_2^2 + 4c_1^2 \\
& \wedge \ v_o > 0 \ \wedge \ v = v_o \wedge \ r_o = \frac{v_o^2}{k} \wedge \\
& r = r_o \wedge \ I = 1 \ \wedge \ I_{obs} \leq 1 \ \wedge \ I_{obs} > 0 \Big) \\
L_{impl_3} \equiv \ & \forall r_n, \forall I_n \Big(I_n \leq 1 \ \wedge \ I_n > 0 \ \wedge r_n = r_o I_n^2 \Big) \\
& \rightarrow \Big(I_{obs} \neq I_n \vee r_{obs} > r_n \Big) \\
L_{expl_3} \equiv \ & r_{obs} > r_o I_{obs}^2 \\
dyn_3 \equiv \ & \Big(r' = \frac{-2c_1 r_o I^2}{v} \wedge \ v' = -c_1 \\
& \wedge \ I' = \frac{-c_1 I}{v} \quad I > 0 \wedge v > 0 \Big) \\
no_collision_3 \equiv \ & \Big(I_{obs} \neq I \vee r_{obs} > r \Big)
\end{aligned}$$

Theorem 5: Verification for Combined Maneuver

$$init_3 \wedge \ L_{impl_3} \rightarrow \big[(dyn_3)\big](no\_collision_3) \tag{29}$$

Theorem 6: Implicit-Explicit Safety Region Equivalence

$$init_3 \rightarrow \big(L_{impl_3} \ \leftrightarrow \ L_{expl_3}\big) \tag{30}$$

The proof strategy for **Theorem 5** (29) is similar to the Cartesian case and proving **Theorem 6** (30) involves simple quantifier elimination.

5 DISCUSSION

The hybrid models and their formal proofs discussed in this work (**Theorem 1** (8), **Theorem 2** (9), **Theorem 3** (26),**Theorem 5** (29) and **Theorem 6** (30)), have all been developed using *differential dynamic logic* d$\mathcal{L}$ [19] and have been formally verified in the d$\mathcal{L}$ theorem prover KeYmaera X [7]. *Differential dynamic logic* d$\mathcal{L}$ provides an efficient way of modeling piece-wise continuous differential equations as hybrid programs.

However, throughout the development of d$\mathcal{L}$ models (3.2) & (4.3) for our collision avoidance systems, we faced challenges due to the fact that in its current implementation, *differential dynamic logic* d$\mathcal{L}$ and the theorem prover KeYmaera X, provide no direct ways of expressing exponentials and trigonometric functions. KeYmaera X does not allow directly expressing those functions on purpose, so as to preserve the decidability of its logic. We circumvented this problem up to a certain extent, by encoding trigonometric and exponential expressions as independent variables (e.g. s, c & I in **Theorem 3** (26) etc.), but this method proves to be challenging when the involved expressions become complicated. For example, in developing **Theorem 3** (26) and **Theorem 4** (27) , we assumed that the inherent relations (23)-(25), between the position coordinates

of the obstacle $(x_{obs}, y_{obs}, I_{obs})$ or between any generic point's coordinates (x, y, I) holds true before the verification process. This assumption was made because we could not find a relatively easy method of encoding the inherent relationship as a precondition check in *differential dynamic logic* d$\mathcal{L}$ and the theorem prover KeYmaera X. We faced similar difficulties in expanding our current **Theorem 3** (26) to include a finite rectangular shape for the vehicle (as done in **Theorem 1** (8) and **Theorem 2** (9)). Also, formally proving equivalence between implicit and explicit safety region in **Theorem 4** (27) was not done in KeYmaera X due to the lack of support for trigonometric and exponential function.

Recent results [24, 25] show a complete axiomatization for differential equation invariants described by Noetherian functions, including trigonometric and exponential functions. These results were developed concurrently to this work, and we were not able to use them. However, they are currently being implemented in the newest version of KeYmaera X, and should simplify the kind of analysis done in this work in the future.

6 RELATED WORK

Formal verification of collision avoidance has been of great interest to the formal methods community. In the context of automobile maneuvers, past formal verification work focuses on braking-only or turning-only maneuvers, and to the best of our knowledge this is the first work exploring formal verification of braking while swerving. But much of the past work on collision avoidance has focused on airplanes, as well as on robots in more recent years.

Regarding robot collision avoidance, Mitsch *et al.* [18] formally verify the collision avoidance for planar robots using non-linear dynamic program based modeling. But their collision model is based purely upon the center to center distance, without considering any realistic geometry for the robot's body. Martin *et al.* [17] formally verify station keeping maneuvers for a planar robot. They have used a non-linear hybrid program to model the overall dynamics and a differential invariant based approach for proving related safety properties. However, they consider the robot's environment free of any obstacles and do not analyze collision avoidance conditions. Both of these works pertain to swerving-only motion of the vehicle.

Regarding aircraft collision avoidance, Tomlin *et al.* [28] formally verify conflict resolution maneuvers for aircraft, however their hybrid program is based upon automata-theoretic modal logic and the analysis and proofs are non-intuitive and considerably more involved. Platzer and Clarke [23] formally verify collision avoidance maneuvers for aircraft. They analyze planar turning maneuvers but they do not consider any extended geometry for the aircraft, instead modeling collision purely on the basis of center to center distance. Jeannin *et al.* [12] formally verify the ACAS X system (Next-Generation Airborne Collision Avoidance System), by formally verifying the geometric configurations of aircraft, under which the advice of ACAS X is safe. They use a hybrid program and a safety-region-based approach for the task of formal verification, however the dynamic model considered is linear and does not consider rotations of the aircraft. Dowek *et al.* [3] provide a provably safe distributed conflict resolution strategy for aircrafts, considering both horizontal and vertical maneuvers. Their dynamics model is similar to [12], and does not consider rotation of the safety buffer around the aircraft. Again all of these works consider either turning-only or braking-only maneuvers for the vehicle.

Regarding car collision avoidance, Loos *et al.* [16] and Sturm and Tiwari [27] formally verify the correctness of adaptive cruise control algorithms. In their models, the dynamic motion of the car is constrained to a straight line without turns.

Reachability analysis methods, such as CORA [2], dReach [14] or SpaceEx [6], offer an alternative method to formally verify dymamical systems. Although those techniques are typically more automated, they rely on approximations and cannot give guarantees in terms of parametric safe regions, expressed symbolically in terms of the different parameters. In contrast, the parametricity of the safe regions is crucial in our work, so that a maneuver can be deemed safe or unsafe at runtime in a small amount of calculation time. All the verification is performed offline and simple, formally verified safety checks can be implemented online.

Overall our approach differs from previous related works in that:

- unlike [16–18], we explore simultaneous braking and swerving during the same maneuver;
- unlike [28], we base the model of our non-linear hybrid program on d$\mathcal{L}$ to handle differential equations;
- unlike [16–18, 23, 27], we utilize a safety-region-based modeling technique for formally verifying the safety property of the hybrid program. This technique provides a faster on-line implementation for vehicle guidance;
- unlike [12], we consider a non-linear hybrid program for the dynamics model, including the effects of vehicle body rotation;
- unlike [12, 16, 18, 23, 27], our vehicle model is more realistic having an extended geometric body.
- unlike reachability analysis methods [2, 6, 14], we verify exact and parametric safe regions.

7 CONCLUSIONS AND FUTURE WORK

This paper focuses on a Unicycle Model for swerving-only and swerving-while-braking systems. A useful extension of this work would be to relieve some crucial parameters, e.g., the coefficient of friction, to be non-deterministic. Such an extension would make the model more realistic and the results more widely applicable.

Future work can include expansion to more sophisticated models, such as a Bicycle Model. The Bicycle Model improves upon the Unicycle Model by using more realistic control parameters, such as steering angle and braking force, in the differential equations.

Furthermore, the models of this paper feature a single, stationary obstacle. While this represents a common scenario found on the road, many other scenarios exist. Future work would handle multi-agent systems consisting of a variety of obstacles such as multiple stationary obstacles and moving intruders.

ACKNOWLEDGMENTS

The authors would like to thank Nikos Aréchiga, Stefan Mitsch and André Platzer for valuable discussions, as well as the anonymous reviewers for their feedback on this paper. Toyota Research Institute ("TRI") provided funds to assist the authors with their research but this article solely reflects the opinions and conclusions of its authors and not TRI or any other Toyota entity.

REFERENCES

[1] Aakash Abhishek, Harry Sood, and Jean-Baptiste Jeannin. 2020, to appear. Formal Verification of Swerving Maneuvers for Car Collision Avoidance. In *2020 American Control Conference (ACC)*. IEEE.

[2] Matthias Althoff. 2015. An introduction to CORA 2015. In *Proc. of the Workshop on Applied Verification for Continuous and Hybrid Systems*.

[3] Gilles Dowek, César Munoz, and Victor Carreño. 2005. Provably safe coordinated strategy for distributed conflict resolution. In *AIAA guidance, navigation, and control conference and exhibit*. 6047.

[4] Lester E Dubins. 1957. On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *American Journal of mathematics* 79, 3 (1957), 497–516.

[5] Tony Foale. 2006. *Motorcycle handling and chassis design: the art and science*. Tony Foale.

[6] Goran Frehse, Colas Le Guernic, Alexandre Donzé, Scott Cotton, Rajarshi Ray, Olivier Lebeltel, Rodolfo Ripado, Antoine Girard, Thao Dang, and Oded Maler. 2011. SpaceEx: Scalable verification of hybrid systems. In *International Conference on Computer Aided Verification*. Springer, 379–395.

[7] Nathan Fulton, Stefan Mitsch, Jan-David Quesel, Marcus Völp, and André Platzer. 2015. KeYmaera X: An axiomatic tactical theorem prover for hybrid systems. In *International Conference on Automated Deduction*. Springer, 527–538.

[8] Thomas D Gillespie. 1992. *Fundamentals of vehicle dynamics*. Technical Report. SAE Technical Paper.

[9] Thomas A Henzinger. 2000. The theory of hybrid automata. In *Verification of Digital and Hybrid Systems*. Springer, 265–292.

[10] Michael Hoy, Alexey S Matveev, and Andrey V Savkin. 2015. Algorithms for collision-free navigation of mobile robots in complex cluttered environments: a survey. *Robotica* 33, 3 (2015), 463–497.

[11] Yong K Hwang and Narendra Ahuja. 1992. Gross motion planning–a survey. *ACM Computing Surveys (CSUR)* 24, 3 (1992), 219–291.

[12] Jean-Baptiste Jeannin, Khalil Ghorbal, Yanni Kouskoulas, Aurora Schmidt, Ryan Gardner, Stefan Mitsch, and André Platzer. 2017. A formally verified hybrid system for safe advisories in the next-generation airborne collision avoidance system. *International Journal on Software Tools for Technology Transfer* 19, 6 (2017), 717–741.

[13] Desmond King-Hele. 2002. Erasmus Darwin's improved design for steering carriages—-and cars. *Notes and records of the Royal Society of London* 56, 1 (2002), 41–62.

[14] Soonho Kong, Sicun Gao, Wei Chen, and Edmund Clarke. 2015. dReach: δ-reachability analysis for hybrid systems. In *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*. Springer, 200–205.

[15] Jean-Paul Laumond et al. 1998. *Robot motion planning and control*. Vol. 229. Springer.

[16] Sarah M Loos, André Platzer, and Ligia Nistor. 2011. Adaptive cruise control: Hybrid, distributed, and now formally verified. In *International Symposium on Formal Methods*. Springer, 42–56.

[17] Benjamin Martin, Khalil Ghorbal, Eric Goubault, and Sylvie Putot. 2017. Formal Verification of Station Keeping Maneuvers for a Planar Autonomous Hybrid System. *arXiv preprint arXiv:1709.02561* (2017).

[18] Stefan Mitsch, Khalil Ghorbal, David Vogelbacher, and André Platzer. 2017. Formal verification of obstacle avoidance and navigation of ground robots. *The International Journal of Robotics Research* 36, 12 (2017), 1312–1340.

[19] André Platzer. 2008. Differential dynamic logic for hybrid systems. *Journal of Automated Reasoning* 41, 2 (2008), 143–189.

[20] André Platzer. 2010. *Logical analysis of hybrid systems: proving theorems for complex dynamics*. Springer Science & Business Media.

[21] André Platzer. 2012. Logics of dynamical systems. In *Proceedings of the 2012 27th Annual IEEE/ACM Symposium on Logic in Computer Science*. IEEE Computer Society, 13–24.

[22] André Platzer. 2018. *Logical Foundations of Cyber-Physical Systems*. Springer.

[23] André Platzer and Edmund M Clarke. 2009. Formal verification of curved flight collision avoidance maneuvers: A case study. In *International Symposium on Formal Methods*. Springer, 547–562.

[24] André Platzer and Yong Kiam Tan. 2018. Differential equation axiomatization: The impressive power of differential ghosts. In *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science*. 819–828.

[25] André Platzer and Yong Kiam Tan. 2019. Differential Equation Invariance Axiomatization. *arXiv preprint arXiv:1905.13429* (2019).

[26] Rajesh Rajamani. 2011. *Vehicle dynamics and control*. Springer Science & Business Media.

[27] Thomas Sturm and Ashish Tiwari. 2011. Verification and synthesis using real quantifier elimination. In *Proceedings of the 36th international symposium on Symbolic and algebraic computation*. ACM, 329–336.

[28] Claire Tomlin, George J Pappas, and Shankar Sastry. 1998. Conflict resolution for air traffic management: A study in multiagent hybrid systems. *IEEE Transactions on automatic control* 43, 4 (1998), 509–521.

[29] Shaopu Yang, Yongjie Lu, and Shaohua Li. 2013. An overview on vehicle dynamics. *International Journal of Dynamics and Control* 1, 4 (2013), 385–395.