Automated Synthesis of Safe Autonomous Vehicle Control Under Perception Uncertainty

Susmit Jha and Vasumathi Raman

United Technology Research Center, Berkeley
{jhask,ramanv}@utrc.utc.com

Abstract. Autonomous vehicles have found wide-ranging adoption in aerospace, terrestrial as well as marine use. These systems often operate in uncertain environments and in the presence of noisy sensors, and use machine learning and statistical sensor fusion algorithms to form an internal model of the world that is inherently probabilistic. Autonomous vehicles need to operate using this uncertain world-model, and hence, their correctness cannot be deterministically specified. Even once probabilistic correctness is specified, proving that an autonomous vehicle will operate correctly is a challenging problem. In this paper, we address these challenges by proposing a *correct-by-synthesis* approach to autonomous vehicle control. We propose a probabilistic extension of temporal logic, named Chance Constrained Temporal Logic (C2TL), that can be used to specify correctness requirements in presence of uncertainty. We present a novel automated synthesis technique that compiles C2TL specification into mixed integer constraints, and uses second-order (quadratic) cone programming to synthesize optimal control of autonomous vehicles subject to the C2TL specification. We demonstrate the effectiveness of the proposed approach on a diverse set of illustrative examples.

1 Introduction

Intelligent systems with varying degrees of autonomy, from recommendation systems [34] to fully autonomous aerial vehicles [23], have been widely adopted for controlling ground, air and under-water vehicles. These systems are increasingly deployed in safety-critical applications, both in military domains such as aerospace missions, search and rescue, and surveillance, as well as in civilian infrastructure like factories and farms. Their increasing prevalence makes it vital to be able to ensure the correctness of their operation in an efficient and reliable manner. Currently, these systems are often designed manually, and their certification relies on tests and extensive requirements on the design process. These are complex systems with tightly-coupled components that implement control, perception and logical decision making, and proving the correctness of manual designs is challenging [33, 26]. The difficulty of this task is further amplified by the uncertain environment in which these systems operate, and the inherent probabilistic nature of the statistical techniques used to observe the environment. In this paper, we address this challenge by defining a new specification language, Chance Constrained Temporal Logic (C2TL), that extends linear temporal logic to capture uncertainty in environment and perception. We present a novel approach to designing autonomous control algorithms that are guaranteed to satisfy C2TL properties.

An autonomous control system can be conceptually divided into two key subsystems: a perception pipeline to observe the world, and a control pipeline comprising high-level reasoning and low-level motion planning. Both these subsystems are well-studied in the control and robotics literatures, but the quantification of uncertainty in perception [14] and control under uncertainty [4] remain challenging. The traditional approach to the design of autonomous systems decouples perception uncertainty and control by using probabilistic thresholds in perception, and building a conservative world model: the control is designed with respect to this conservative model. This decoupling leads to overly conservative control in practice, and also makes it difficult to establish formal guarantees and prove safety of these systems. For example, it is clear that any qualitative Boolean property would be violated with non-zero probability in a setting with perception uncertainty modeled using Gaussian noise. Chance constraints [31] provide a natural way to specify probabilistic correctness properties, but have so far only be shown useful for specifying invariant-like properties. On the other hand, temporal logics such as signal temporal logic (STL) [15] and linear temporal logic (LTL) [27] have emerged as effective specification languages for verifying and synthesizing automated control subject to complex specifications, including history-dependent and timing requirements.

C2TL extends temporal logic with chance constraints, thus providing an effective specification language for the autonomous control of systems operating under uncertainty. We show that C2TL formulae can be compiled into mixed integer constraints; thus, C2TL strikes the right balance between expressiveness and ease of reasoning. Quadratic cone programming can be used to automatically synthesize optimal control satisfying the C2TL specifications.
We make the following contributions:

1. We define *Chance Constrained Temporal Logic* (C2TL) and demonstrate its use to specify correctness of autonomous vehicle system control.
2. We formulate the problem of synthesizing autonomous vehicle control subject to C2TL specifications while optimizing a quadratic cost function; we reduce this problem to a second order (quadratic) cone program that can be solved using scalable tools such as CVXOPT [3].
3. We demonstrate the effectiveness of our approach on a diverse set of examples.

2 Background and Related Work

Projects such as the Defense Advanced Research Projects Agency (DARPA) Urban Challenge [32] and the VisLab Intercontinental Autonomous Challenge [10] have been instrumental in spurring the development and maturation of autonomous vehicle technology. One key area where autonomous systems still struggle is in dealing with uncertainty, arising from stochastic environments or noisy perception. Most autonomous systems learn about their environment using sensors such as cameras and LIDAR units to infer the environment state, which is maintained in the form of probabilistic beliefs. Uncertainty in these probabilistic beliefs arise from two sources [21, 25, 13, 20]. First, the environment states are often dynamic and change over time. Second, the information gathered from sensors is often not sufficient to exactly infer the environment state. As an example, consider a popular perception technique like *simultaneous localization and mapping* [5](SLAM), which is used for determining the current position of an autonomous vehicle. The estimated position of the vehicle and the coordinates of other entities in the map are often assumed to have Gaussian noise. Aside from localization and mapping, another critical perception challenge for autonomous vehicles is obstacle detection and tracking [22, 9]. Camera and laser range finders are used to locally detect and avoid obstacles during navigation for a previously constructed map. This is particularly useful in the presence of dynamic objects whose locations are not fixed in the environment map. The uncertainty in the parametric models representing the obstacles is usually also modeled using Gaussian random variables. The proposed C2TL specifications incorporate these Gaussian models of uncertainty in perception by allowing the predicates in the formulae to be chance constraints [31] over Gaussian random variables.

The control of stochastic systems has been extensively investigated, beginning with the work of Pontryagin [28] and Bellman [7], and extending to more recent literature [17, 30, 29, 11]. Its applications include optimal guidance for spacecrafts [2] and flight-controllers [6]. The focus has been on the safety problem, where the goal is to determine a control policy that maximizes the probability of remaining within a safe set during a finite time horizon [1]. This safe control problem is usually reformulated as a stochastic optimal control problem with multiplicative costs over a controlled Markov chain. In contrast, our goal is to satisfy a probabilistic temporal logic specification while optimizing over a given cost metric. This can be naturally modeled using chance constrained programs [12, 24], used for uncertainty modeling in various engineering fields [19, 37]. For a detailed recent survey of the literature on chance constrained programming approaches, the interested reader is directed to [31]. Here we extend these approaches to temporal logic specifications. Another dimension along which we extend existing stochastic control techniques [36] is in our consideration of non-convex feasible spaces, which is critical for autonomous vehicles operating in environments with obstacles.

Recent work has developed scalable, optimization-based methods for the automatic synthesis of controllers from temporal logic specifications with deterministic constraints [16]. Signal temporal logic (STL) [15] has been proposed for controller synthesis, because it combines dense time modalities with numerical predicates over continuous state variables. C2TL extends STL to specify probabilistic temporal properties, by allowing predicates to be *chance constraints* over continuous state variables rather than just real-valued functions. The uncertainty is restricted to probabilistic predicates, and temporal operators are not probabilistic; this is in contrast to other probabilistic extensions of temporal logics [18]. We show that C2TL can be used to specify correctness requirements for an autonomous vehicle under perception uncertainty. We also present a reduction from C2TL constraints to mixed integer constraints which are linear in the state variables. Thus, C2TL provides a balance between expressiveness of the specification language and efficiency of automated synthesis.

3 Automated Synthesis of Autonomous Vehicle Control

We first define *Chance Constrained Temporal Logic* (C2TL), and then illustrate how the correctness of autonomous vehicle control can be specified using C2TL. We then describe how C2TL specifications can be compiled into deterministic mixed integer conic constraints. We then formulate the problem of synthesizing the correct control of autonomous systems as a second order cone programming problem. The cost being optimized is quadratic and optimization is done with respect to conic constraints that are bilinear in the state variables and perception coefficients.

Notation: The correctness property is specified over the system state variables $X = \{x_1, x_2, \ldots, x_n\}$, which can represent the position of the vehicle, its velocity, acceleration, orientation, angular velocities and other relevant parameters. The domain of X is denoted $Dom(X)$, and is usually a subset of $\mathbb{R}^n$. The state of the system at time t is denoted by $\mathbf{x}_t \in Dom(X)$.

In this work, half-planes form the basic unit of representation of knowledge acquired through perception. This is motivated by the observation that perception algorithms often employ half-plane learning techniques such as Bayesian linear regression and classifiers. For example, an obstacle can be perceived as an intersection of half-planes which represent the convex hull of the obstacle. Half-planes are represented as $\phi_{lin} : \mathbf{a}_i\mathbf{x}_t + b_i \leq 0$ or $\mathbf{a}_i\mathbf{x}_t + b_i < 0$, where the coefficients $\mathbf{a}_i, b_i$ are inferred by perception algorithms. Due to uncertainty in perception, the coefficients are not deterministically known: rather, we only know the probability distribution over the coefficients. Let $Dom(\mathbf{a}_i), Dom(b_i)$ denote the domain of the coefficients, and $p(\mathbf{a}_i), p(b_i)$ denote the respective probability density functions. So, the constraints from perception are not tautological, but instead hold with an associated probability, that is, $Pr(\mathbf{a}_i\mathbf{x}_t + b_i \leq 0) \geq 1 - \delta$ or $Pr(\mathbf{a}_i\mathbf{x}_t + b_i < 0) \geq 1 - \delta$.

We denote the control inputs of the autonomous system, which are the values to be synthesized, by U; the value at each time instant t is $\mathbf{u}_t$. A trace of system states and control values is denoted by $\tau : \mathbb{R}_{\geq 0} \rightarrow X \times U$ where $\tau(t) = (\mathbf{x}_t, \mathbf{u}_t)$.

3.1 Chance Constrained Temporal Logic

We now define chance constrained temporal logic as a probabilistic extension of signal temporal logic, motivated by two key observations:

- For specifications applied to autonomous systems, temporal aspects of correctness arise from mission requirements such as reaching specific positions in sequence while staying away from particular regions. These temporal aspects of mission requirements do not usually have any associated uncertainty.
- Perception gathers information about a particular instant of time, and uncertainty in perception is hence reflected only in the predicates computed on the system states at a given time, and not on the temporal operators.

We therefore introduce chance constraints at the atomic predicate level of our logic. The syntax definition of C2TL is as follows:

$$\begin{aligned}
\phi_{det} &:= \phi_{lin} \mid \phi_{lin} \wedge \phi_{lin} \mid \neg\phi_{lin} \\
\phi_{cc} &:= [Pr(\phi_{det}) \geq 1-\delta] \mid \neg\phi_{cc} \mid {\sim}\phi_{cc} \mid \phi_{cc} \wedge \phi_{cc} \mid \phi_{cc} \vee \phi_{cc} \mid \phi_{cc} U_{[a,b]} \phi_{cc},
\end{aligned}$$

where:

- *linear predicate* ϕ_{lin} over the variables $v \subseteq X \cup U$ is of the form
$$\phi_{lin}(v) : \mathbf{a}_i v + b_i \leq 0 \ \text{ or } \ \mathbf{a}_i v + b_i < 0$$
- *deterministic predicate* ϕ_{det} is a Boolean combination of linear predicates.
- *chance-constraint* [12] is a probabilistic extension of deterministic predicates and is of the form $Pr(\phi_{det}) \geq 1-\delta$. where $0 \leq \delta \leq 1$ represents uncertainty about whether the inequality holds.
- The coefficients $\mathbf{a}_i, b_i$ of the chance constraints are random variables with Gaussian probability distributions, rather than constants.

The set of coefficients that satisfy a deterministic predicate ϕ_{det} over variables v is denoted by $R(\phi_{det}, v)$. So, the probability of satisfying ϕ_{det} when the coefficients are probabilistic is given by $p_c(\phi_{det}, v) = \int_{c \in R(\phi_{det}, v)} p(c) dc$ where $c = (\mathbf{a}, b)$. C2TL admits the standard *globally* (G), *eventually* (F) and *until* (U) operators of temporal logic; here we restrict discussion to the *until* (U) operator, which can be used to represent all of the others. The subscripts of the operators denote the time interval associated with the property, as in STL.

The satisfaction of a C2TL formula over a trace τ at time t is defined recursively as follows:

$$\begin{aligned}
&\tau(t) \models \phi_{lin} && \Leftrightarrow && \phi_{lin}(\tau(t)) \\
&\tau(t) \models \neg\phi^1_{lin} \wedge \phi^2_{lin} && \Leftrightarrow && \phi^1_{lin}(\tau(t)) \wedge \phi^2_{lin}(\tau(t)) \\
&\tau(t) \models \neg\phi_{lin} && \Leftrightarrow && \neg\phi_{lin}(\tau(t)) \\
&\tau(t) \models [Pr(\phi_{det}) \geq 1-\delta] && \Leftrightarrow && p_c(\phi_{det}, \tau(t)) \geq 1-\delta \\
&\tau(t) \models \neg[Pr(\phi_{det}) \geq 1-\delta] && \Leftrightarrow && p_c(\phi_{det}, \tau(t)) < 1-\delta \\
&\tau(t) \models {\sim}[Pr(\phi_{det}) \geq 1-\delta] && \Leftrightarrow && \tau(t) \models [Pr(\neg\phi_{det}) \geq 1-\delta] \\
&\tau(t) \models \phi^1_{cc} \wedge \phi^2_{cc} && \Leftrightarrow && \tau(t) \models \phi^1_{cc} \wedge \tau(t) \models \phi^2_{cc} \\
&\tau(t) \models \phi^1_{cc} \vee \phi^2_{cc} && \Leftrightarrow && \tau(t) \models \phi^1_{cc} \vee \tau(t) \models \phi^2_{cc} \\
&\tau(t) \models \phi^1_{cc} U_{[a,b]} \phi^2_{cc} && \Leftrightarrow && \exists t_1 \ t+a \leq t_1 \leq t+b \wedge \tau(t_1) \models \phi^2_{cc} \\
& && && \wedge (\forall t_2 \ t \leq t_2 \leq t_1 \Rightarrow \tau(t_2) \models \phi^1_{cc})
\end{aligned}$$

As a special case, when $\delta = 0$, chance constraints become deterministic. Chance constraints have two kinds of negations: *logical* negation denoted by $\neg$ and *prob-*

abilistic negation denoted by $\sim$. Consider a deterministic formula ϕ_{det} and its logical negation $\neg\phi_{det}$, and corresponding chance constraints $\phi_{cc} \equiv Pr(\phi_{det}) \geq 1-\delta$ and the probabilistic negation $\sim\phi_{cc} \equiv Pr(\neg\phi_{det}) \geq 1-\delta$. If $\delta = 0.8$, then $\phi_{cc} \equiv Pr(\phi_{det}) \geq 0.2$, that is, $Pr(\neg\phi_{det}) < 0.8$. This is consistent with $\sim\phi_{cc} \equiv Pr(\neg\phi_{det}) \geq 0.2$. Thus, it is possible for both ϕ_{cc} and its probabilistic negation $\sim\phi_{cc}$ to simultaneously be true.

The following theorem relates probabilistic negation and logical negation when $\delta < 0.5$. This case is relevant because it corresponds to "likely" chance constraints, where the probability of violation is less than 0.5. In practice, most useful constraints obtained from perception have significantly high confidence and δ is very small.

Theorem 1. *If $\delta < 0.5$, probabilistic negation is equivalent to logical negation, that is, $\neg\phi_{cc} \equiv \sim\phi_{cc}$.*

Proof. $\neg\phi_{cc} \equiv \neg[Pr(\phi_{det}) \geq 1-\delta] \equiv \neg[Pr(\neg\phi_{det}) < \delta]$. Now, $\delta < 0.5 \equiv \delta < 1-\delta$. Thus, $\neg\phi_{cc} \equiv \neg[Pr(\neg\phi_{det}) < \delta < 1-\delta]$, that is, $\neg\phi_{cc} \equiv \neg[Pr(\neg\phi_{det}) < 1-\delta]$ when $\delta < 0.5$. Further, $\neg[Pr(\neg\phi_{det}) < 1-\delta] \equiv [Pr(\neg\phi_{det}) \geq 1-\delta] \equiv \sim\phi_{cc}$. Hence, $\neg\phi_{cc} \equiv \sim\phi_{cc}$ if $\delta < 0.5$. □

3.2 C2TL Specification for Autonomous Vehicle Control

We now describe how the correctness properties of an autonomous system can be specified using C2TL.

Obstacles: Any obstacle can be approximated by a union of a finite number of convex polytopes. The planes forming the convex polytopes are only probabilistically known, due to perception uncertainty. A convex polytope is a conjunction of half-planes (linear constraints), and can be represented as $\bigwedge_i(\mathbf{a}_i\mathbf{x}_t + b_i > 0)$, where the coefficients $\mathbf{a}_i \sim \mathcal{N}(\mathbf{a}_i^\mu, \mathbf{a}_i^\Sigma)$ are assumed to be Gaussian variables whose mean and variance are estimated by the perception pipeline. Since the coefficients are Gaussian, collision with obstacles cannot be ruled out deterministically. Let δ_{obs} be the user-specified threshold for the maximum allowable probability of collision with obstacles. This collision avoidance property is specified in C2TL as: $Pr(\bigvee_i \mathbf{a}_i\mathbf{x}_t + b_i \leq 0) \geq 1-\delta_{obs}$. The property of avoiding multiple obstacles j is specified as: $Pr(\bigwedge\limits_j \bigvee\limits_i \mathbf{a}_{ij}\mathbf{x}_t + b_{ij} \leq 0) \geq 1-\delta_{obs}$.

We assume that the map consists of static and dynamic obstacles as well as real or virtual walls that restrict the vehicle to be within a bounded region, but outside of obstacle areas. Let $\mathbf{a}_{ij}$ be the coefficients of the obstacles and $\mathbf{w}_{ij}$ be the coefficients of the perceived walls. The unobstructed map with uncertainty can thus be represented using a formula $\phi_{map} :=$
$[Pr(\bigwedge\limits_j \bigvee\limits_i \mathbf{a}_{ij}\mathbf{x}_t + b_{ij} \leq 0) \geq 1-\delta_{obs}] \wedge [Pr(\bigwedge\limits_j \bigvee\limits_i \mathbf{w}_{ij}\mathbf{x}_t + b_{ij} \leq 0) \geq 1-\delta_{wall}]$
where $\mathbf{a}_{ij} \sim \mathcal{N}(\mathbf{a}_{ij}^\mu, \mathbf{a}_{ij}^\Sigma)$ represents the uncertain perception of obstacles, and $\mathbf{w}_{ij} \sim \mathcal{N}(\mathbf{w}_{ij}^\mu, \mathbf{w}_{ij}^\Sigma)$ represents the uncertain perception of walls (which in practice includes uncertainty in self-localization). Similar constraints can be added

for other parameters of an autonomous system such as constraints on speed or acceleration based on the system's current region in the map.

Mission: Apart from the safe navigation requirement represented by the global property $G(\phi_{map})$, a second set of useful specifications on autonomous vehicles corresponds to mission requirements. For example, the vehicle must reach its final destination within some time-bound t_{max}. Because of uncertainty in perception, we can not guarantee this property deterministically. Given a user-specified probability threshold $\delta_{mission}$ of failing to achieve the mission goals, the goal of reaching the destination is specified as $F_{[0,t_{max}]}(Pr(\mathbf{x} = \mathbf{x}_{dest}) \geq 1 - \delta_{mission})$. Other examples include the requirement that an autonomous car wait at a stop sign until all cross-traffic arriving at the intersection before it has passed, and that an aircraft flies straight without turning till it reaches the safe velocity range for turning. These properties can be specified using *until* properties, $\phi_1 U_{[0,t]} \phi_2$. We denote the set of mission constraints by $\phi_{mission}$.

The overall specification for the safe control of autonomous system is thus $\phi_{map} \wedge \phi_{mission}$: that is, the system achieves the temporal specification of mission goals while remaining safe with respect to the map. We note that the focus of this paper is on autonomous vehicles, but C2TL can also be used to specify behavior of other autonomous systems such as robotic manipulators, and the techniques presented in this paper extend beyond this application domain.

3.3 C2TL to Conservative Linear Constraints

In this section, we present a translation of C2TL constraints over Gaussian random variables to deterministic linear constraints. The constraints are linear with respect to system (state) variables and conic overall due to uncertain coefficients. The first part of the translation deals with temporal logic formulae and Boolean combinations of elementary chance constraints. The second part of translation focuses on elementary chance constraints, and reduces those to deterministic constraints linear in the state variables.

We focus on chance constraints with violation probability threshold less than 0.5 [1]. Similar to the STL encoding provided in [16], we introduce Boolean, that is, $\{0,1\}$ integer variables $m_t^{\phi_{cc}}$ for each chance constraint ϕ_{cc} and time t. These Boolean variables are related in the same way as for the STL encoding.

- Negation: $m_t^{\neg\phi_{cc}} = 1 - m_t^{\phi_{cc}}$
- Conjunction: $m_t^{\phi_{cc}^1 \wedge \phi_{cc}^2} = \min(m_t^{\phi_{cc}^1}, m_t^{\phi_{cc}^2})$
- Disjunction: $m_t^{\phi_{cc}^1 \vee \phi_{cc}^2} = \max(m_t^{\phi_{cc}^1}, m_t^{\phi_{cc}^2})$
- Until: $m_t^{\phi_{cc}^1 U_{[a,b]} \phi_{cc}^2} = \max_{t' \in [t+a, t+b]}(\min(m_{t'}^{\phi_{cc}^2}, \min_{t'' \in [t,t']}(m_{t''}^{\phi_{cc}^1})))$

[1] As discussed in Section 3.1, probabilistic negation is not the same as logical negation when violation probability (δ) can be 0.5 or more, and hence, we will need two $\{0,1\}$ integer variables to represent the truth value of each chance constraint, to account for four cases depending on the truth value of the chance constraint and its probabilistic negation. For likely (violation probability $\delta < 0.5$) chance constraints, one $\{0,1\}$ integer variable is sufficient by Theorem 1.

The next challenge is in translating the probabilistic chance constraints over Gaussian variables to deterministic mixed integer constraints that are linear in the state variables. We consider chance constraints of the form:

$$\phi_{cc}^{elem} \equiv Pr(\bigwedge_j \bigvee_i^{N_j} \mathbf{a}_{ij}\mathbf{x}_t + b_{ij} \leq 0) \geq 1 - \delta_{tm}.$$

In the rest of the section, we show how we can conservatively over-approximate ϕ_{cc}^{elem} using mixed integer constraints which are satisfiable only if ϕ_{cc}^{elem} is satisfiable. We first note that $\phi_{cc}^{elem} \equiv$:

$$Pr(\bigwedge_{i,j} \mathbf{a}_{ij}\mathbf{x}_t + b_{ij} - Mz_{ij} \leq 0) \geq 1 - \delta_{tm} \wedge \bigwedge_j \left(\sum_i z_{ij} < N_j \wedge z_{ij} \in \{0,1\} \right),$$

where M is a sufficiently large positive number. This transformation uses the big-M reduction common in non-convex optimization, see [8] for examples. The above equivalence holds because at least one z_{ij} is 0 for each j since $\sum_i z_{ij} < N_j$ and $z_{ij} \in \{0,1\}$, and thus, at least one of the constraints in $\bigvee_i^{N_j} \mathbf{a}_{ij}\mathbf{x}_t + b_{ij} \leq 0$ must be true for each j.

Next, we use Boole's inequality to decompose the conjunction in the probabilistic chance constraint as follows.

$$Pr(\bigwedge_{i,j} \mathbf{a}_{ij}\mathbf{x}_t + b_{ij} - Mz_{ij} \leq 0) \geq 1 - \delta_{tm} \Leftrightarrow Pr(\bigvee_{i,j} \mathbf{a}_{ij}\mathbf{x}_t + b_{ij} - Mz_{ij} > 0) < \delta_{tm}.$$

Further, $Pr(\bigvee_{i,j} \mathbf{a}_{ij}\mathbf{x}_t + b_{ij} - Mz_{ij} > 0) < \sum_{i,j} Pr(\mathbf{a}_{ij}\mathbf{x}_t + b_{ij} - Mz_{ij} > 0)$ since the probability of union of events is less than the sum of the individual probabilities of the occurrence of each event.

Next, we introduce new variables $0 \leq \epsilon_{ij} \leq 1$ with $\sum_{i,j} \epsilon_{ij} < \delta_{tm}$, and conservatively approximate the chance constraint as:

$$Pr(\bigwedge_j \bigvee_i^{N_j} \mathbf{a}_{ij}\mathbf{x}_t + b_{ij} \leq 0) \geq 1 - \delta_{tm} \quad \Leftarrow \quad \bigwedge_{i,j} Pr(\mathbf{a}_{ij}\mathbf{x}_t + b_{ij} - Mz_j \leq 0) \geq 1 - \epsilon_{ij}$$
$$\wedge \bigwedge_{ij} 0 \leq \epsilon_{ij} \leq 1 \wedge \sum_{ij} \epsilon_{ij} < \delta_{tm} \ \wedge \ \sum_j z_j < N_j \ \wedge \ \bigwedge_j z_j \in \{0,1\}$$

With $N = \sum_j N_j$, we choose $\epsilon_{ij} = \delta_{tm}/N$, which corresponds to uniform risk allocation among the probabilistic constraints above. However, more efficient risk allocation techniques [38] can also be used. Since $\mathbf{a}_{ij}$ is a Gaussian random variable, the linear combination of Gaussian variables $\mathbf{a}_{ij}\mathbf{x}_t + b_{ij} - Mz_j$ is also Gaussian. Further, the uniform risk allocation ensures that the violation probability bounds are constant. So, $Pr(\mathbf{a}_{ij}\mathbf{x}_t + b_{ij} - Mz_j \leq 0) \geq 1 - \epsilon_{ij}$ can be translated to a deterministic constraint $\mathbf{a}_{ij}\mathbf{x}_t + b_{ij} - Mz_j \leq \texttt{ErfInv}(\epsilon_{ij})$ where `ErfInv` is the Gaussian inverse error function computed using the table for Gaussian distributions, as discussed in [36]. Consequently, the probabilistic chance constraints are reduced to a set of deterministic constraints. This completes the translation of C2TL constraints to a set of deterministic mixed integer linear constraints over the system variables.

The following theorem summarizes the conservative nature of the above translation. Given the control specification for an autonomous vehicle ψ^{C2TL}, the above translation generates ψ^{MILP} which conservatively approximates ψ^{C2TL}.

Theorem 2. *Given C2TL constraints ψ^{C2TL}, the translation presented above will generate a set of mixed integer constraints ψ^{MILP} such that $\psi^{C2TL} \Rightarrow \psi^{MILP}$.*

There are two sources of conservativeness of ψ^{MILP}:

- We use the sum of the probabilities of chance constraints to upper-bound the probability of their disjunction. If the constraints are completely independent of each other, the sum of their individual probabilities is exactly the probability of their disjunction. The approximation is small if the constraints are mostly independent, which is often the case for specifying autonomous vehicle systems, since obstacles usually do not overlap.
- We use a uniform risk allocation of the violation probability bounds for each individual constraint. This can be further improved using more effective risk allocation techniques [38].

Thus, the translation of C2TL constraints to mixed integer constraints is conservative, but the approximation introduced is expected to be tight for C2TL specifications used for automated vehicle control.

3.4 Optimal Autonomous Vehicle Control

The goal of synthesizing optimal control for autonomous vehicles is to automatically generate the control inputs $\mathbf{u}$. The control inputs applied at time k are denoted by $\mathbf{u}_k$. Often, the dynamical system can be approximated by *linearizing the system* around the current point of operation and using *model predictive* or *receding horizon control.* A detailed discussion on model predictive control for signal temporal logic can be found in [16]. We employ a similar approach here.

A finite parametrization of a linear system assuming piecewise constant control inputs yields the following difference equation:

$$\mathbf{x}_{k+1} = A_k\mathbf{x}_k + B_k\mathbf{u}_k,$$

where $\mathbf{x}_k \in \mathcal{R}^{n_x}$ is the system state in n_x dimensions, $\mathbf{u}_k \in \mathcal{R}^{n_u}$ denotes the n_u control inputs, and A_k, B_k are coefficients representing linear system dynamics around the state $\mathbf{x}_k$. We consider the control problem over a bounded time horizon T, that is, $0 \leq k \leq T$.

Further, the control inputs $\mathbf{u}_k$ at all time steps k are required to be in a convex feasible region $\mathcal{F}_u$, that is,

$$\mathcal{F}_u \equiv \bigwedge_{i=1}^{N_g} (g_i^T\mathbf{u} \leq c_i); \quad \bigwedge_k \mathbf{u}_k \in \mathcal{F}_u$$

where the convex region $\mathcal{F}_u$ is represented as intersection of N_g half-planes.

The state variables are required to satisfy the autonomous vehicle correctness specification ψ_{ap}^{C2TL}, that is, $\mathbf{x}_k \models \psi_{ap}^{C2TL}$ for all k. We can conservatively approximate the autonomous vehicle correctness specification by ψ_{ap}^{MILP} as discussed earlier, that is, $\mathbf{x}_k \models \psi_{ap}^{MILP} \Rightarrow \mathbf{x}_k \models \psi_{ap}^{C2TL}$

In addition to correctness specification, the synthesized vehicle control is also expected to minimize a user-specified cost function $J(\mathbf{x}, \mathbf{u})$. We restrict the cost function J to be quadratic in order to ensure that solving the control synthesis problem is computationally efficient. Quadratic functions can capture cost metrics of the form $\sum_i \mathbf{u}_k^\dagger U^\dagger U \mathbf{u}_k + \mathbf{x}_k^\dagger S^\dagger S \mathbf{x}_k$ with appropriate scaling vectors U and S, where † denotes the transpose of a matrix. These can represent metrics such as fuel consumption as well as metrics on the vehicle path.

Problem 1 (Autonomous Vehicle Control).

$$\begin{aligned} &\arg\min_{\mathbf{u}} J(\mathbf{x}, \mathbf{u}) \\ &\text{s.t.} \qquad \mathbf{x}_{k+1} = \mathbb{A}_k \mathbf{x}_k + \mathbb{B}_k \mathbf{u}_k, k = 1 \ldots T, \mathbf{u}_k \in \mathcal{F}_u, \mathbf{x}_k \models \psi_{ap}^{C2TL} \end{aligned}$$

Problem 2 (Conservative Autonomous Control).

$$\begin{aligned} &\arg\min_{\mathbf{u}} J(\mathbf{x}, \mathbf{u}) \\ &\text{s.t.} \qquad \mathbf{x}_{k+1} = \mathbb{A}_k \mathbf{x}_k + \mathbb{B}_k \mathbf{u}_k, k = 1 \ldots T, \mathbf{u}_k \in \mathcal{F}_u, \mathbf{x}_k \models \psi_{ap}^{MILP} \end{aligned}$$

Recall that every solution to Problem 2 also solves Problem 1. Moreover, for a bounded time horizon T and a quadratic cost function, since all the constraints are linear in system variables and conic due to the presence of uncertain coefficients, the conservative autonomous control problem can be solved using scalable second order (quadratic) cone programming tools such as CVXOPT [3]. The following theorem summarizes the correctness guarantee:

Theorem 3. *The solution to Problem 2 is sound with respect to Problem 1: if control inputs are synthesized for the conservative problem, they are guaranteed to satisfy the specified correctness property* ψ_{ap}^{C2TL}.

This theorem follows from Theorem 2 because $\mathbf{x}_k \models \psi_{ap}^{C2TL} \Leftarrow \mathbf{x}_k \models \psi_{ap}^{MILP}$. Note, however, that the proposed synthesis method (i.e. solving the more efficiently solvable conservative problem using second order cone programming) is incomplete for the autonomous control problem due to the conservative approximation of C2TL constraints ($\psi_{ap}^{C2TL} \Leftarrow \psi_{ap}^{MILP}$).

The incompleteness relates to degree of conservative approximation introduced in the translation of C2TL constraints to MILP constraints.

4 Case Studies

We now experimentally demonstrate the effectiveness of our approach. All experiments were done on a Intel Core-i7 2.9 GHz x 8 machine with 16 GB memory. Where applicable, we use a baseline comprised of a modified LQG-based motion planning algorithm [35] and a Monte Carlo sampling-based search algorithm to find an optimal trajectory over the uncertain world model. Our technique is more general than sampling-based approaches because we can enforce temporal logic specifications beyond reachability goals common in classical motion planning. Additionally, the uncertainty in our problem lies within the perceived world model rather than the system evolution.

Navigation in an uncertain map: The first case-study considers the problem of navigation in an uncertain map from [39]. Parameter values and other details of the map can be found in [39]. A point mass with two modes – moving forward and turning – is expected to navigate safely in the map shown in Figure 1. The walls in the map and the obstacle in the center are modeled using probabilistic constraints that incorporate the uncertainty in perception. The uncertain walls are illustrated in the map by sampling values of the coefficients and drawing the corresponding walls. The probabilistic safety requirement in this case is a global property requiring that the vehicle avoid the walls and obstacles with a very high probability. The objective function being optimized is quadratic in the final state as well as the control inputs:

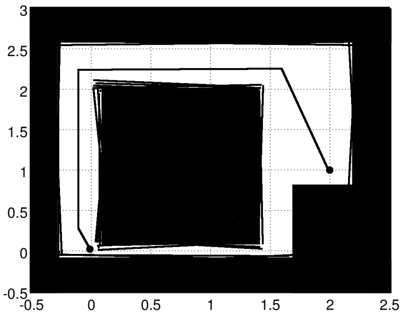


Fig. 1: Navigation in an uncertain map

$$f(\mathbf{x}, \mathbf{u}) = 50(\mathbf{x}_N - \mathbf{x}_{dest})^T(\mathbf{x}_N - \mathbf{x}_{dest}) + 0.001 \sum_i \mathbf{u}_i^T \mathbf{u}_i,$$

where $\mathbf{x}_{dest}$ is the destination state $(2, 1)$. Observe that although the cost function drives the optimization to minimize the path length, the generated path goes around the obstacle, taking the longer path. This is because the shorter path would violate the C2TL safety constraints due to the uncertainty in the location of the obstacles and walls. This is illustrated in Figure 1.

When compared to the approach in [39], the method proposed in this paper takes 4.1 seconds instead of 25.2 seconds to compute a sequence of control inputs. Monte Carlo simulation was used to estimate the probability of constraint violation. For each simulation, the location of the walls and the obstacles was determinized by sampling from the corresponding Gaussian distribution. We then checked whether the automatically generated path intersected with the walls or obstacles, violating the safety requirement. When the violation probability in the C2TL specification was set to 0.001, Monte Carlo trials did not find a single instance out of 10000 simulations in which the property was violated. We increased the violation probability to 0.01, and found 8 out of 10000 simulations that violated the probability; i.e., the estimated violation probability was 0.0008. This demonstrates how the proposed approach conservatively approximates the specified probabilistic constraint, generating a motion plan that satisfies the probabilistic safety property.

Lane Change: The second case-study is on the synthesis of control for an autonomous vehicle such as a car, trying to pass a tractor-trailer in an adjacent lane, as described in [40]. The trailer can probabilistically switch into the passing car's lane. If the car is ahead of the trailer when the trailer initiates a lane change, then the car should accelerate, and if the car is behind the trailer when the trailer initiates the lane change, the car should decelerate. If the trailer switches lanes when it is just adjacent to the car, the car has no action to prevent an accident. Thus, a completely safe course of action is not possible for the autonomous car and it can only try to keep the risk below a user-specified threshold by passing the trailer quickly and not staying in the unsafe region for long. The uncertainty arises due to a probabilistic model of when the trailer will switch lanes, based on the car's observations of its behavior. This case-study assumes a static jump Markov model of this uncertainty, as shown in Figure 3 of [40]. The safety specification requires that the passing car is either decelerating and behind the trailer until the trailer make the lane switch, or the trailer remains in its lane until the passing the car is accelerating and ahead of the trailer. We also require the separation between the car and trailer to be above a safe limit with a high probability. The threshold of violing the specification was set to 0.015. The cost function was the time spent behind the trailer but not in the same lane. Autopilot generation took 5.8 seconds, and Monte Carlo simulations of the generated autopilot showed that the actual threshold of violation is 0.0004.

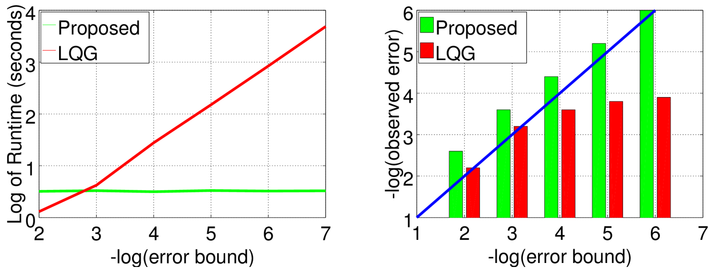
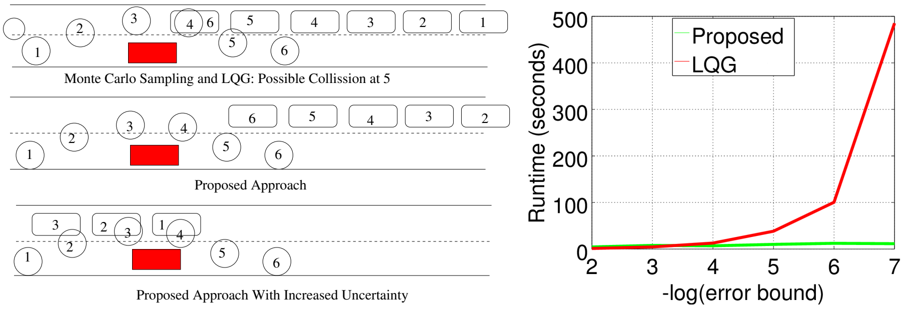


Fig. 2: (a) Runtime Comparison (b) Accuracy Comparison

In order to compare with LQG-based sampling techniques, we change the cost function to incorporate temporal logic requirements by penalizing the car for coming close to trailer, and rewarding it for either passing the trailer or traveling behind it in the same lane if the trailer changed lanes. In Figure 2(a), we compare runtime of the synthesis technique for each specified violation probability. While our proposed technique's runtime is not very sensitive to the violation probability, the runtime of the sampling-based approach increases sharply due to the increase in the number of required simulation runs. In Figure 2(b), we present the violation probability observed in Monte Carlo simulations when both approaches are given the same runtime, by restricting the number of simulation runs. All bars above the diagonal line satisfy the probabilistic constraint, while bars below it do not (note the negative log scale on y-axis as well as x-axis). No violations were found for our proposed technique for error bounds 10^{-6} and lower. Thus, the proposed method always satisfies the specification, whereas sampling fails to do so for smaller error bounds.

Passing a Vehicle Using Oncoming Traffic Lane: The third case-study is from recent work by Xu et al [41]. In this case-study, a vehicle's lane is blocked and it needs to move into the lane of oncoming traffic to go around the obstacle.

The perception pipeline on the vehicle estimates the position and the speed of oncoming traffic before deciding to get into the oncoming traffic lane. The dynamics and parameters are described in [41], and we discuss only the results here. Due to uncertainty in perception, we can not deterministically guarantee safe maneuvering of the vehicle, but we require that the probability of collision with oncoming traffic or with the obstacle in the vehicle's lane is below a threshold of ϵ. The uncertainty in perception of the speed of the oncoming traffic is represented by the standard deviation sd of the random variable representing the speed. We modify the cost function from the original case-study, because we use C2TL constraints to specify the safety conditions. The cost function measures the time taken to re-enter the lane after crossing the obstacle.



(a) Illustration of Synthesized Control (b) Runtime vs $-\log(\epsilon)$

Fig. 3: Left: Positions of the autonomous vehicle (circle) and oncoming traffic (rectangle) at different (1-6) time steps are shown. The red rectangle is the obstacle. Right: Runtime comparison for different violation probability bounds.

We illustrate the qualitative nature of the synthesized control in Figure 3(a). For violation probability $\epsilon = 0.0001$, the control synthesized by the sampling-based technique in time comparable to our approach (4 seconds) is not probabilistically safe. The control synthesized using the proposed technique relies on speeding up and getting around the obstacle before the oncoming traffic. When we increase the standard deviation in the perception of the speed of the oncoming traffic by 10X, the control synthesized by our approach picks a less optimum, higher-cost solution in order to meet the safety violation probability requirement, which slows the vehicle and waits for the oncoming traffic to pass before going around the obstacle. Figure 3(b) shows that the runtime of the sampling-based approach increases rapidly with a decrease in ϵ, while it does not change significantly for our technique.

5 Conclusion

In this paper, we present a formal approach to synthesizing autonomous vehicle control in presence of perception uncertainty. Chance constrained temporal logic (C2TL) is proposed to capture correctness specifications in the presence of uncertainty. The autonomous vehicle control synthesized by our technique is guaranteed to satisfy the probabilistic specifications, as demonstrated in several case studies.

References

1. Alessandro Abate, Maria Prandini, John Lygeros, and Shankar Sastry. Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems. *Automatica*, 44(11):2724–2734, 2008.
2. Behcet Acikmese and Scott R Ploen. Convex programming approach to powered descent guidance for mars landing. *Journal of Guidance, Control, and Dynamics*, 30(5):1353–1366, 2007.
3. Martin S Andersen, Joachim Dahl, and Lieven Vandenberghe. Cvxopt: A python package for convex optimization, version 1.1. 6. *Available at cvxopt. org*, 2013.
4. Karl J Åström. *Introduction to stochastic control theory.* Courier Corporation, 2012.
5. Tim Bailey and Hugh Durrant-Whyte. Simultaneous localization and mapping (slam): Part ii. *IEEE Robotics & Automation Magazine*, 13(3):108–117, 2006.
6. Neal M Barr, Dagfinn Gangsaas, and Dwight R Schaeffer. Wind models for flight simulator certification of landing and approach guidance and control systems. Technical report, DTIC Document, 1974.
7. Richard Bellman, Richard Ernest Bellman, and Richard Ernest Bellman. *Introduction to the mathematical theory of control processes*, volume 2. IMA, 1971.
8. Pietro Belotti, Jon Lee, Leo Liberti, Francois Margot, and Andreas Wachter. Branching and bounds tightening techniques for non-convex minlp. *Optimization Methods Software*, August 2009.
9. Nicola Bernini, Massimo Bertozzi, Luca Castangia, Marco Patander, and Mario Sabbatelli. Real-time obstacle detection using stereo vision for autonomous ground vehicles: A survey. In *ITSC*, pages 873–878. IEEE, 2014.
10. Alberto et. al. Broggi. Autonomous vehicles control in the vislab intercontinental autonomous challenge. *Annual Reviews in Control*, 36(1):161–171, 2012.
11. Christos G Cassandras and John Lygeros. *Stochastic hybrid systems*, volume 24. CRC Press, 2006.
12. A. Charnes, W. W. Cooper, and G. H. Symonds. Cost horizons and certainty equivalents: An approach to stochastic programming of heating oil. *Management Science*, 4(3):pp. 235–263, 1958.
13. Roderick De Nijs, Sebastian Ramos, Gemma Roig, Xavier Boix, LV Gool, and Kolja Kuhnlenz. On-line semantic perception using uncertainty. In *IROS*, pages 4185–4191. IEEE, 2012.
14. Luc Devroye, László Györfi, and Gábor Lugosi. *A probabilistic theory of pattern recognition*, volume 31. Springer Science & Business Media, 2013.
15. Alexandre Donzé and Oded Maler. Robust satisfaction of temporal logic over real-valued signals. In *FORMATS*, pages 92–106, 2010.
16. Vasumathi Raman et. al. Model predictive control with signal temporal logic specifications. In *CDC*, pages 81–87, Dec 2014.
17. Xenofon Koutsoukos and Derek Riley. Computational methods for reachability analysis of stochastic hybrid systems. In *HSCC*, pages 377–391. Springer, 2006.
18. Marta Kwiatkowska, Gethin Norman, and David Parker. Prism: Probabilistic symbolic model checker. In *Computer performance evaluation: modelling techniques and tools*, pages 200–204. Springer, 2002.
19. Pu Li, Harvey Arellano-Garcia, and Gnter Wozny. Chance constrained programming approach to process optimization under uncertainty. *Computers and Chemical Engineering*, 32(1-2):25–45, 2008.

20. P Martinet, C Laugier, and U Nunes. Special issue on perception and navigation for autonomous vehicles, 2014.
21. Christoph Daniel et. al. Mathys. Uncertainty in perception and the hierarchical gaussian filter. *Frontiers in Human Neuroscience*, 8(825), 2014.
22. Timothy G McGee, Raja Sengupta, and Karl Hedrick. Obstacle detection for small autonomous aircraft using sky segmentation. In *ICRA 2005*, pages 4679–4684. IEEE, 2005.
23. Lorenz Meier, Petri Tanskanen, Friedrich Fraundorfer, and Marc Pollefeys. Pixhawk: A system for autonomous flight using onboard computer vision. In *ICRA*, pages 2992–2997. IEEE, 2011.
24. Bruce L. Miller and Harvey M. Wagner. Chance constrained programming with joint constraints. *Operations Research*, 13(6):930–945, 1965.
25. Matthew R et al. Nassar. An approximately bayesian delta-rule model explains the dynamics of belief updating in a changing environment. *The Journal of Neuroscience*, 30(37):12366–12378, 2010.
26. Charles Patchett, Mike Jump, and Michael Fisher. Safety and certification of unmanned air systems. *Engineering & Technology Reference*, 1(1), 2015.
27. Amir Pnueli. The temporal logic of programs. In *Providence*, pages 46–57, 1977.
28. LS Pontryagin. Optimal control processes. *Usp. Mat. Nauk*, 14(3), 1959.
29. Stephen Prajna, Ali Jadbabaie, and George J Pappas. A framework for worst-case and stochastic safety verification using barrier certificates. *Automatic Control, IEEE Transactions on*, 52(8):1415–1428, 2007.
30. Maria Prandini and Jianghai Hu. Stochastic reachability: Theory and numerical approximation. *Stochastic hybrid systems, Automation and Control Engineering Series*, 24:107–138, 2006.
31. András Prékopa. *Stochastic programming*, volume 324. Springer Science, 2013.
32. Christopher Rouff and Mike Hinchey. *Experience from the DARPA urban challenge.* Springer Science & Business Media, 2011.
33. John Rushby. New challenges in certification for aircraft software. In *EMSOFT*, pages 211–218. ACM, 2011.
34. Brent A et. al. Terwilliger. Advancement and application of unmanned aerial system human-machine-interface (hmi) technology. In *Human Interface and the Management of Information*, pages 273–283. Springer, 2014.
35. Jur Van Den Berg, Pieter Abbeel, and Ken Goldberg. LQG-MP: Optimized path planning for robots with motion uncertainty and imperfect state information. *Int. J. Rob. Res.*, 30(7):895–913, June 2011.
36. Michael Vitus. *Stochastic Control Via Chance Constrained Optimization and its Application to Unmanned Aerial Vehicles.* PhD thesis, Stanford University, 2012.
37. Michael P Vitus and Claire J Tomlin. Closed-loop belief space planning for linear, gaussian systems. In *ICRA*, pages 2152–2159. IEEE, 2011.
38. Michael P. Vitus and Claire J. Tomlin. On feedback design and risk allocation in chance constrained control. In *CDC 2011*, pages 734–739, Dec 2011.
39. Michael P. Vitus and Claire J. Tomlin. A hybrid method for chance constrained control in uncertain environments. In *CDC*, pages 2177–2182, Dec 2012.
40. Michael P. Vitus and Claire J. Tomlin. A probabilistic approach to planning and control in autonomous urban driving. In *CDC*, pages 2459–2464, 2013.
41. Wenda Xu, Jia Pan, Junqing Wei, and John M Dolan. Motion planning under uncertainty for on-road autonomous driving. In *ICRA*, pages 2507–2512. IEEE, 2014.