Fast and Guaranteed Safe Controller Synthesis (FACTEST)

Kristina Miller, Chuchu Fan, Sayan Mitra
Department of Electrical and Computer Engineering, College of Engineering, University of Illinois at Urbana-Champaign

Introduction

Autonomy is a highly active area with many open problems. **Safe motion planning** is a related problem that becomes more complicated when the environment is dynamic and unknown. This problem has both perception and control components.

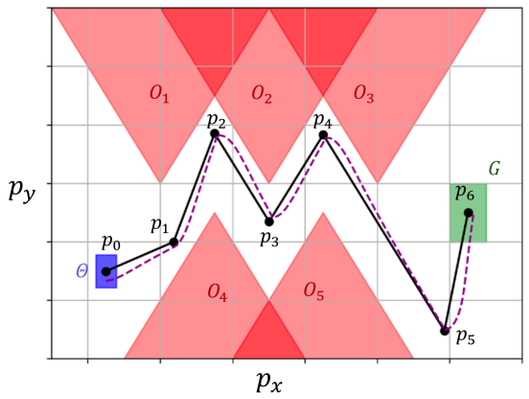
Control can be addressed using **controller synthesis**. The goal is to generate a controller that guarantees a system will satisfy some high-level requirement.

Satisfiability solvers can be used to generate controllers. However, **nonlinear constraints** increase the complexity of the SAT equations.

Here, we address the problem by decomposing it into two parts: **reference path synthesis** and **bounding tracking error**. This approach can synthesize controllers for models with **6-dimensions** in **less than a second**.

Reach-avoid specifications

A system must drive from some initial set Θ to some goal G while avoiding the obstacles O_i. The obstacles can be either static or dynamic.



Nonlinear vehicle models

We consider nonlinear vehicles models with tracking controllers.

1. $\mathcal{X} \subset \mathbb{R}^n$ is the state space
2. $\Theta \subseteq \mathcal{X}$ is the initial set
3. $\mathcal{Z}$ is the set of reference segments where $z \in \mathcal{Z}$ is of the form $z: \mathbb{R}_{\geq 0} \to \mathcal{X}$
4. The dynamic function $f: \mathcal{X} \times \mathcal{Z} \to \mathcal{X}$ is Lipschitz continuous w.r.t. the first argument

The trajectory of the system satisfies $\xi(0) = x_0$ and $\frac{d\xi}{dt} f(x_0, z)$.

Computing error bounds

Tracking error: $e(t) = \xi(t) - \xi_{ref}(t)$.

Tracking error increases when the system switches modes.

Use reachability analysis to bound the tracking error.

Lyapunov stability: for any $\epsilon > 0$ there exists $\delta > 0$ such that if $\|\xi(0) - \xi_{ref}(0)\| < \delta$ then $\|\xi(t) - \xi_{ref}(t)\| < \epsilon$ for all t.

Symmetrical reachsets: A line segment can be transformed to create any other segment. The vehicle state can be transformed to follow any line segment. Therefore, we transform precomputed reachsets to compute error bounds for any reference segment.

We can inductively find the error bounds over each segment.

The error bound for the i^{th} reference trajectory segment is ℓ_i.

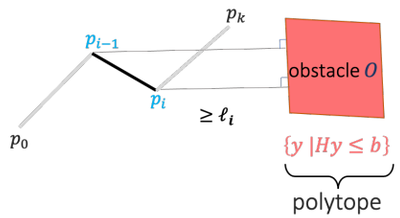
Lemma 1. Any system system starting in Θ and following $\xi_{ref}(t)$ remains within ℓ_i of the i^{th} reference trajectory segment.

Reference trajectory synthesis

Nonlinear vehicles can follow piecewise linear (PWL) reference trajectories.

The i^{th} segment is defined by the waypoints p_{i-1} and p_i.

We must ensure that p_{i-1} and p_i lie at least ℓ_i outside at least one face of every obstacle.



The constraint for one such segment is as follows:

$$\bigvee_{j=1}^{\#\text{surfaces}} \frac{H_j p_{i-1} - b_j}{\|H_j\|_2} > \ell_i \wedge \frac{H_j p_i - b_j}{\|H_j\|_2} > \ell_i$$

For a PWL reference trajectory with k line segments, every $p_0, \ldots, p_k$, must satisfy the above constraint for every obstacle O_i.

The number of constraints scales with the number of faces on the obstacles.

Putting it all together

The constraints are encoded in Synth_WPS.

$$\exists p_0, \cdots, p_k, \text{ such that}$$

$$p_0 = x_{ref}(0),$$

$$\bigwedge_i \bigwedge_{o_i} \bigvee_j \frac{H_j p_{i-1} - b_j}{\|H_j\|_2} > \ell_i \wedge \frac{H_j p_i - b_j}{\|H_j\|_2} > \ell_i,$$

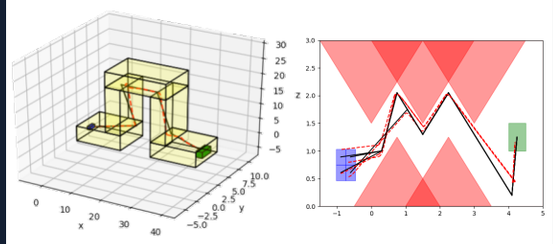
$$\bigvee_G \bigwedge_j \frac{H_j p_k - b_j}{\|H_j\|_2} < \ell_k$$

Given a reach-avoid scenario and the maximum number of line segments N:

1. Compute error bounds $\{\ell_i\}_{i=1}^N$ for arbitrary $\xi_{ref}(t)$.
2. Start from $k = 1$. Attempt to synthesize waypoints using Synth_WPS.
3. If waypoints are synthesized, return $\xi_{ref}(t)$.
4. If waypoints are not synthesized, $k = k + 1$ and repeat step 2.
5. If $k = N$, partition Θ. Repeat steps 2-5 until the maximum number of partitions is reached.

Theorem 1. Given the reference trajectory $\xi_{ref}(t)$, the actual trajectory $\xi(t)$ will satisfy $\xi(t) \cap O_i$ for all obstacles O_i, for all time.

This is implanted in a tool called **FACTEST**: https://kmmille.github.io/FACTEST/



Navigating dynamic and partially unknown environments

A perception oracle returns a sensed subset of obstacles $\{O_i\}_{i=m}^n$.

Using a perception oracle and solving the problem in a receding horizon fashion, the vehicle remains safe for all time.

Theorem 2. Given $\xi_{ref}(t)$ that avoids $\{O_i\}_{i=m}^n$ and synthesis period T_S, the actual trajectory $\xi(t)$ will satisfy $\xi(t) \cap O_i$ for all obstacles O_i, for all time.

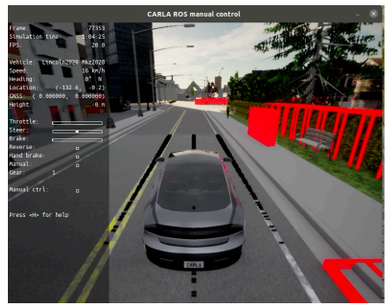
Experimental evaluation

Model	Dim	Initial set	# O_i	# Parts	Time (s)
car	3	0.2	9	1	0.028
car	3	0.4	9	4	0.144
car	3	0.6	9	16	0.605
car	3	0.2	22	1	0.078
car	3	0.07	19	1	0.415
robot	4	0.2	9	1	0.025
robot	4	0.4	9	4	0.196
robot	4	0.6	9	16	0.612
robot	4	0.2	22	1	0.498
robot	4	0.07	19	1	0.635
auv	6	0.866	8	1	0.317
hover	4	0.866	8	1	0.140

Benchmarking and evaluation - GRAIC

Unfortunately, there are no standard benchmarks for controller synthesis. To address the lack of benchmarks, we present **GRAIC** (**G**eneralized **RA**cing **I**ntelligence **C**ompetition).

We are accepting submissions soon! More information can be found here: https://popgri.github.io/Race/



References

Fan, Chuchu, Kristina Miller, and Sayan Mitra. "Fast and guaranteed safe controller synthesis for nonlinear vehicle models." *International Conference on Computer Aided Verification*. Springer, Cham, 2020.

Miller, Kristina, Chuchu Fan, and Sayan Mitra. "Planning in Dynamic and Partially Unknown Environments", Under Review, 2021