Trajectory Optimization with Continuous-Time Constraint Satisfaction via Penalized Trust Region Sequential Convex Programming

`trajopt-ctcs`

Autonomous Controls Laboratory
University of Washington, Seattle

Contents

1 Notation

$\mathbb{R}^n = \mathbb{R}^{n\times 1}$	Vectors are matrices with one column
$1_n, 0_n$	Vector of ones and zeros, respectively, in $\mathbb{R}^n$ (subscript inferred whenever omitted)
$0_{n\times m}$	Matrix of zeros in $\mathbb{R}^{n\times m}$
I_n	Identity matrix in $\mathbb{R}^{n\times n}$
$\lvert v\rvert_+$	$\square \mapsto \max\{0, \square\}$ applied element-wise for vector v
v^2	$\square \mapsto \square^2$ applied element-wise for vector v
(u, v)	Concatenation of vectors $u \in \mathbb{R}^n$ and $v \in \mathbb{R}^m$ to form a vector in $\mathbb{R}^{n+m}$
$[A\, B]$	Concatenation of matrices A and B with same number of rows
$\begin{bmatrix} A \\ B \end{bmatrix}$	Concatenation of matrices A and B with same number of columns

2 Problem Formulation

Dynamical system:

$$\frac{\mathrm{d}\xi(t)}{\mathrm{d}t} = \dot{\xi}(t) = F(\xi(t), \nu(t)) \tag{1}$$

for $t \in [t_\mathrm{i}, t_\mathrm{f}]$, with state ξ and control input ν.

Path constraints:

$$\begin{aligned} g(\xi(t), \nu(t)) &\leq 0 \\ h(\xi(t), \nu(t)) &= 0 \end{aligned}$$

where g and h are vector-valued functions. Operators "$\leq$" and "$=$" are interpreted element-wise.

2.1 Time-Dilation and Constraint Reformulation

Time-dilation:

$$\frac{\mathrm{d}\xi(t(\tau))}{\mathrm{d}\tau} = \mathring{\xi}(\tau) = \frac{\mathrm{d}\xi(\tau)}{\mathrm{d}t} s(\tau) = s(\tau) F(\xi(\tau), \nu(\tau)) \tag{2}$$

for $\tau \in [0, 1]$, where $t(\tau)$ is a strictly increasing function with domain $[0, 1]$. Note that ξ and ν are re-defined to be functions of $\tau \in [0, 1]$ instead of t.

Free-final-time problems are converted to fixed-final-time problems via time-dilation. The dilation factor:

$$s(\tau) = \frac{\mathrm{d}t(\tau)}{\mathrm{d}\tau}$$

is treated as an additional control input and is required to be positive.

The control input is re-defined to be $u = (\nu, s)$, and the final time is given by:

$$t_{\mathrm{f}} = t_{\mathrm{i}} + \int_0^1 s(\tau)\mathrm{d}\tau$$

Constraint reformulation augments path constraints to the time-dilated system dynamics (2) as follows:

$$\overset{\circ}{x}(\tau) = \begin{bmatrix} \overset{\circ}{\xi}(\tau) \\ \overset{\circ}{y}(\tau) \end{bmatrix} = s(\tau) \begin{bmatrix} F(\xi(\tau), \nu(\tau)) \\ 1^\top |g(\xi(\tau), \nu(\tau))|_+^2 + 1^\top h(\xi(\tau), \nu(\tau))^2 \end{bmatrix} = f(x(\tau), u(\tau)) \tag{3}$$

for $\tau \in [0, 1]$, where $x = (\xi, y)$ is the re-defined state.

Boundary condition on the constraint violation integrator:

$$y(0) = y(1) \tag{4}$$

ensures that the path constraints are satisfied for all $\tau \in [0, 1]$. Note that we can also have an exclusive state for each constraint.

In the subsequent development, we treat

$$\overset{\circ}{x}(\tau) = f(x(\tau), u(\tau)) \tag{5}$$

with $x(\tau) \in \mathbb{R}^{n_x}$ and $u(\tau) \in \mathbb{R}^{n_u}$, for $\tau \in [\tau_{\mathrm{i}}, \tau_{\mathrm{f}}]$, as the template for describing a dynamical system irrespective of whether the time-dilation and constraint reformulation operations have occurred. The domain of the independent variable τ is defined accordingly to handle problems that originally have fixed-final-time. State x and control input u are appropriately defined for problems that are subjected time-dilation and constraint reformulation.

2.2 Discretization

Discretization of interval $[\tau_{\mathrm{i}}, \tau_{\mathrm{f}}]$ into a grid with K nodes:

$$\tau_{\mathrm{i}} = \tau_1 < \ldots < \tau_K = \tau_{\mathrm{f}} \tag{6}$$

If time-dilation is performed, then time duration over sub-interval $[\tau_k, \tau_{k+1}]$ is given by:

$$\Delta t_k = \int_{\tau_k}^{\tau_{k+1}} s(\tau)\mathrm{d}\tau$$

for $k = 1, \ldots, K-1$.

Constraints on dilation factor and time duration:

$$\begin{gathered} s_{\min} \le s(\tau) \le s_{\max} \\ t_{\mathrm{f}} = \sum_{k=1}^{K-1} \Delta t_k \le t_{\mathrm{f},\max} \\ \Delta t_{\min} \le \Delta t_k \le \Delta t_{\max} \end{gathered}$$

The constraints above are defined in `misc.time_cnstr`

Parameterization of control input:

Zero-order-hold (ZOH):

$$u(\tau) = u_k$$

for $\tau \in [\tau_k, \tau_{k+1})$ and $k = 1, \ldots, K-1$. Define $u_K = u_{K-1}$ for convenience.

First-order-hold (FOH):

$$u(\tau) = \left(\frac{\tau_{k+1} - \tau}{\tau_{k+1} - \tau_k}\right) u_k + \left(\frac{\tau - \tau_{k+1}}{\tau_{k+1} - \tau_k}\right) u_{k+1}$$

for $\tau \in [\tau_k, \tau_{k+1}]$ and $k = 1, \ldots, K-1$.

Note that $u(\tau_k) = u_k$ holds for both ZOH and FOH. The node points values of the state and control input: x_k and u_k, respectively, are treated as decision variables.

Convex constraints on the control input need not be subjected to constraint reformulation when ZOH or FOH parameterization is chosen. It is sufficient to impose the convex constraints only at the node points.

We will use the term "discretization" to refer to the combined discretization and control input parameterization operations. This is because the form of ZOH and FOH parameterizations that we adopt are dependent on the choice of discretization grid (6).

Discretized dynamics:

$$x_{k+1} = x_k + \int_{\tau_k}^{\tau_{k+1}} f(x(\tau), u(\tau)) \mathrm{d}\tau \tag{7}$$

for $k = 1, \ldots, K-1$, where $x(\tau)$ is the solution to (5) over $[\tau_k, \tau_{k+1}]$ with initial condition x_k and $u(\tau)$ is ZOH- parameterized using u_k or FOH-parameterized using u_k and u_{k+1}.

If constraint reformulation is performed, then the boundary condition (4) is relaxed to

$$y_{k+1} - y_k \leq \epsilon$$

for $k = 1, \ldots, K-1$, so that all feasible solutions do not violate LICQ. Note that y_k is value of the state y in (3) at node τ_k.

Parameter	Code variable(s)	Meaning
g	`cnstr_fun`	Inequality constraint function
h	`cnstr_eq_fun`	Equality constraint function
$s_{\min}$, $s_{\max}$	`smin, smax`	Upper- and lower-bounds on dilation factor
$\Delta t_{\min}$, $\Delta t_{\max}$	`dtmin, dtmax`	Upper- and lower-bound on sub-interval time duration
$t_{\mathrm{f},\max}$	`ToFmax`	Upper bound on final time
x_k	`x(:,k)`	State at node τ_k
u_k	`u(:,k)`	Control input at node τ_k
y_k	`y(:,k)`	Constraint violation integrator at node τ_k
ϵ	`eps_cnstr`	Constraint relaxation tolerance

2.2.1 Linearization

At each iteration of SCP, the discretized dynamics (7) is linearized with respect to the previous iterate, denoted by $\bar{x}_k$, $\bar{u}_k$, for $k = 1, \ldots, K$.

For each $k = 1, \ldots, K-1$, let $\bar{x}^k(\tau)$ denote the solution to (5) over $[\tau_k, \tau_{k+1}]$ generated with initial condition $\bar{x}_k$ and control input $\bar{u}(\tau)$, which is parameterized using $\bar{u}_k$, for $k = 1, \ldots, K$. Note that, in general, $\bar{x}^k(\tau_{k+1}) \neq \bar{x}_{k+1}$ before SCP converges.

The Jacobian of f in (5) evaluated on $\bar{x}^k(\tau)$, $\bar{u}(\tau)$ are denoted by:

$$A^k(\tau) = \frac{\partial f(\bar{x}^k(\tau), \bar{u}(\tau))}{\partial x}$$
$$B^k(\tau) = \frac{\partial f(\bar{x}^k(\tau), \bar{u}(\tau))}{\partial u}$$

for $\tau \in [\tau_k, \tau_{k+1}]$.

The discretized dynamics (7) is linearized as follows.

ZOH:

$$x_{k+1} = A_k x_k + B_k u_k + w_k \tag{8}$$

for $k = 1, \ldots, K-1$, where A_k, B_k, w_k result from the solution to the following initial value problem over $\tau \in [\tau_k, \tau_{k+1}]$:

$$\begin{aligned}
\mathring{\Phi}_x(\tau, \tau_k) &= A^k(\tau)\Phi_x(\tau, \tau_k) \\
\mathring{\Phi}_u(\tau, \tau_k) &= A^k(\tau)\Phi_u(\tau, \tau_k) + B^k(\tau)\left(\frac{\tau_{k+1} - \tau}{\tau_{k+1} - \tau_k}\right) \\
\Phi_x(\tau_k, \tau_k) &= I_{n_x} \\
\Phi_u(\tau_k, \tau_k) &= 0_{n_x \times n_u}
\end{aligned}$$

$$\begin{aligned}
A_k &= \Phi_x(\tau_{k+1}, \tau_k) \\
B_k &= \Phi_u(\tau_{k+1}, \tau_k) \\
w_k &= \bar{x}^k(\tau_{k+1}) - A_k\bar{x}_k - B_k\bar{u}_k
\end{aligned}$$

FOH:

$$x_{k+1} = A_k x_k + B_k^- u_k + B_k^+ u_{k+1} + w_k \tag{9}$$

for $k = 1, \ldots, K-1$, where A_k, B_k^-, B_k^+, w_k result from the solution to the following initial value problem over $\tau \in [\tau_k, \tau_{k+1}]$:

$$\begin{aligned}
\mathring{\Phi}_x(\tau, \tau_k) &= A^k(\tau)\Phi_x(\tau, \tau_k) \\
\mathring{\Phi}_u^-(\tau, \tau_k) &= A^k(\tau)\Phi_u^-(\tau, \tau_k) + B^k(\tau)\left(\frac{\tau_{k+1} - \tau}{\tau_{k+1} - \tau_k}\right) \\
\mathring{\Phi}_u^+(\tau, \tau_k) &= A^k(\tau)\Phi_u^+(\tau, \tau_k) + B^k(\tau)\left(\frac{\tau - \tau_k}{\tau_{k+1} - \tau_k}\right) \\
\Phi_x(\tau_k, \tau_k) &= I_{n_x} \\
\Phi_u^-(\tau_k, \tau_k) &= 0_{n_x \times n_u} \\
\Phi_u^+(\tau_k, \tau_k) &= 0_{n_x \times n_u}
\end{aligned}$$

$$\begin{aligned}
A_k &= \Phi_x(\tau_{k+1}, \tau_k) \\
B_k^- &= \Phi_u^-(\tau_{k+1}, \tau_k) \\
B_k^+ &= \Phi_u^+(\tau_{k+1}, \tau_k) \\
w_k &= \bar{x}^k(\tau_{k+1}) - A_k\bar{x}_k - B_k^-\bar{u}_k - B_k^+\bar{u}_{k+1}
\end{aligned}$$

3 Miscellaneous

3.1 Parsing to Canonical QP

Canonical QP:

$$\begin{aligned}
&\text{minimize} && \frac{1}{2}z^\top \hat{P} z + \hat{p}^\top z && \text{(10a)}\\
&\text{subject to} && \hat{G} z = \hat{g} && \text{(10b)}\\
& && \hat{H} z \leq \hat{h} && \text{(10c)}
\end{aligned}$$

3.1.1 Convex Subproblem with Scaled Variables

$$\begin{aligned}
&\text{minimize} && w_{\text{cost}} e_x^\top \hat{x}_K + \frac{1}{2} w_{\text{tr}} \sum_{k=1}^{K} \|\hat{x}_k - \hat{\bar{x}}_k\|_2^2 + \|\hat{u}_k - \hat{\bar{u}}_k\|_2^2 + w_{\text{vc}} \sum_{k=1}^{K-1} 1_{n_x}^\top (\mu_k^+ + \mu_k^-) && && \text{(11a)}\\
&\text{subject to} && -\hat{x}_{k+1} + \hat{A}_k \hat{x}_k + \hat{B}_k^- \hat{u}_k + \hat{B}_k^+ \hat{u}_{k+1} + \mu_k^+ - \mu_k^- + \hat{w}_k = 0 && 1 \leq k \leq K-1 && \text{(11b)}\\
& && E_y(\hat{x}_{k+1} - \hat{x}_k) \leq \hat{\varepsilon}_k && 1 \leq k \leq K-1 && \text{(11c)}\\
& && \mu_k^+ \geq 0,\ \ \mu_k^- \geq 0 && 1 \leq k \leq K-1 && \text{(11d)}\\
& && \hat{u}_{k,\min} \leq \hat{u}_k \leq \hat{u}_{k,\max} && 1 \leq k \leq K && \text{(11e)}\\
& && E_{\text{i}} \hat{x}_1 = \hat{z}_{\text{i}},\ \ E_{\text{f}} \hat{x}_K = \hat{z}_{\text{f}} && && \text{(11f)}
\end{aligned}$$

Affinely-scaled absolute variables:

$$\begin{aligned}
x_k &= S_x \hat{x}_k + c_x\\
\hat{x}_k &= S_x^{-1}(x_k - c_x)\\
\hat{\bar{x}}_k &= S_x^{-1}(\bar{x}_k - c_x)\\
\hat{A}_k &= S_x^{-1} A_k S_x\\
\hat{B}_k^- &= S_x^{-1} B_k^- S_u\\
\hat{B}_k^+ &= S_x^{-1} B_k^+ S_u\\
\hat{w}_k &= S_x^{-1}(w_k + A_k c_x + B_k^- c_u + B_k^+ c_u - c_x)\\
\hat{\varepsilon}_k &= \epsilon E_y S_x^{-1} E_y^\top 1_{n_y}\\
\hat{u}_{k,\min/\max} &= S_u^{-1}(u_{\min/\max} - c_u)\\
\hat{z}_{\text{i/f}} &= E_{\text{i/f}} S_x^{-1} E_{\text{i/f}}^\top (z_{\text{i/f}} - E_{\text{i/f}} c_x)
\end{aligned}$$

Linearly-scaled deviation variables:

$$\begin{aligned}
x_k &= S_x \hat{x}_k + \bar{x}_k\\
\hat{x}_k &= S_x^{-1}(x_k - \bar{x}_k)\\
\hat{\bar{x}}_k &= S_x^{-1}(\bar{x}_k - \bar{x}_k) = 0_{n_x}\\
\hat{A}_k &= S_x^{-1} A_k S_x\\
\hat{B}_k^- &= S_x^{-1} B_k^- S_u\\
\hat{B}_k^+ &= S_x^{-1} B_k^+ S_u\\
\hat{w}_k &= S_x^{-1}(w_k + A_k \bar{x}_k + B_k^- \bar{u}_k + B_k^+ \bar{u}_{k+1} - \bar{x}_{k+1})\\
&= S_x^{-1}(\bar{x}^k(\tau_{k+1}) - \bar{x}_{k+1})\\
\hat{\varepsilon}_k &= \epsilon E_y S_x^{-1} E_y^\top 1_{n_y} - E_y S_x^{-1}(\bar{x}_{k+1} - \bar{x}_k)\\
\hat{u}_{k,\min/\max} &= S_u^{-1}(u_{\min/\max} - \bar{u}_k)\\
\hat{z}_{\text{i}} &= E_{\text{i}} S_x^{-1} E_{\text{i}}^\top (z_{\text{i}} - E_{\text{i}} \bar{x}_1)\\
\hat{z}_{\text{f}} &= E_{\text{f}} S_x^{-1} E_{\text{f}}^\top (z_{\text{f}} - E_{\text{f}} \bar{x}_K)
\end{aligned}$$

3.1.2 Parsed Quantites

$$\begin{aligned}
z &= \big(\hat{x}_1, \ldots, \hat{x}_K,\, \hat{u}_1, \ldots, \hat{u}_K,\, \mu_1^-, \ldots, \mu_{K-1}^-,\, \mu_1^+, \ldots, \mu_{K-1}^+\big) \in \mathbb{R}^{(n_x+n_u)K + 2n_x(K-1)}\\
\hat{P} &= \text{blkdiag}\big(w_{\text{tr}} I_{(n_x+n_u)K},\, 0_{2n_x(K-1)\times 2n_x(K-1)}\big)\\
\hat{p} &= w_{\text{cost}}\big(0_{n_x(K-1)},\, e_x,\, 0_{n_u K + 2n_x(K-1)}\big)\\
&\quad - w_{\text{tr}}\big(\hat{\bar{x}}_1, \ldots, \hat{\bar{x}}_K,\, \hat{\bar{u}}_1, \ldots, \hat{\bar{u}}_K,\, 0_{2n_x(K-1)}\big)\\
&\quad + w_{\text{vc}}\big(0_{(n_x+n_u)K},\, 1_{2n_x(K-1)}\big)
\end{aligned}$$

$$
\begin{aligned}
\hat{G} &= \begin{bmatrix} \hat{G}_{\mathrm{i,f}} & 0_{(n_{\mathrm{i}}+n_{\mathrm{f}})\times n_u K} & 0_{(n_{\mathrm{i}}+n_{\mathrm{f}})\times 2n_x(K-1)} \\ \hat{G}_x & \hat{G}_u & \hat{G}_\mu \end{bmatrix} \\
\hat{G}_{\mathrm{i,f}} &= \begin{bmatrix} E_{\mathrm{i}} \ 0_{n_{\mathrm{i}}\times n_x(K-1)} \\ 0_{n_{\mathrm{f}}\times n_x(K-1)} \ E_{\mathrm{f}} \end{bmatrix} \\
\hat{G}_x &= \begin{bmatrix} \hat{A} \ 0_{n_x(K-1)\times n_x} \end{bmatrix} - \begin{bmatrix} 0_{n_x(K-1)\times n_x} \ I_{n_x(K-1)} \end{bmatrix} \\
\hat{G}_u &= \begin{bmatrix} \hat{B}^- \ 0_{n_x(K-1)\times n_u} \end{bmatrix} + \begin{bmatrix} 0_{n_x(K-1)\times n_u} \ \hat{B}^+ \end{bmatrix} \\
\hat{G}_\mu &= \begin{bmatrix} I_{n_x(K-1)} & -I_{n_x(K-1)} \end{bmatrix} \\
\hat{A} &= \mathrm{blkdiag}(\hat{A}_1, \ldots, \hat{A}_{K-1}) \\
\hat{B}^- &= \mathrm{blkdiag}(\hat{B}_1^-, \ldots, \hat{B}_{K-1}^-) \\
\hat{B}^+ &= \mathrm{blkdiag}(\hat{B}_1^+, \ldots, \hat{B}_{K-1}^+) \\
\hat{g} &= (\hat{z}_{\mathrm{i}}, \hat{z}_{\mathrm{f}}, -\hat{w}_1, \ldots, -\hat{w}_{K-1}) \\
\hat{H}_{u,\mu} &= \begin{bmatrix} 0_{2n_u K\times n_x K} & \begin{matrix} I_{n_u K} \\ -I_{n_u K} \end{matrix} & 0_{2n_u K\times 2n_x(K-1)} \\ 0_{2n_x(K-1)\times n_x K} & 0_{2n_x(K-1)\times n_u K} & -I_{2n_x(K-1)} \end{bmatrix} \\
\hat{H}_y &= -\begin{bmatrix} I_{K-1}\otimes E_y \ 0_{n_y(K-1)\times n_x} \end{bmatrix} + \begin{bmatrix} 0_{n_y(K-1)\times n_x} \ I_{K-1}\otimes E_y \end{bmatrix} \\
\hat{H} &= \begin{bmatrix} \hat{H}_{u,\mu} \\ \hat{H}_y \ 0_{n_y(K-1)\times n_u K + 2n_x(K-1)} \end{bmatrix} \\
\hat{h} &= (\hat{u}_{1,\max}, \ldots, \hat{u}_{K,\max}, -\hat{u}_{1,\min}, \ldots, -\hat{u}_{K,\min}, 0_{2n_x(K-1)}, \hat{\varepsilon}_1, \ldots, \hat{\varepsilon}_{K-1})
\end{aligned}
$$

3.2 Extrapolated PIPG

Algorithm 1 xPIPG implementation with FOH

Require: $j_{\max}$, α, β, ρ

Initialize: ζ^1, η^1, χ^1

for $j = 1, \ldots, j_{\max}$ **do**

- $z^{j+1} \leftarrow \zeta^j - \alpha(\hat{P}\zeta^j + \hat{p} + \tilde{G}^\top \eta^j + \tilde{H}^\top \chi^j)$
- $E_{\mathrm{i}}\hat{x}_1^{j+1} \leftarrow z_{\mathrm{i}}$
- $E_{\mathrm{f}}\hat{x}_K^{j+1} \leftarrow z_{\mathrm{f}}$
- $\hat{u}_k^{j+1} \leftarrow \max\{\hat{u}_{k,\min}, \min\{\hat{u}_{k,\max}, \hat{u}_k^{j+1}\}\}$ $\quad k = 1, \ldots, K$
- $(\mu_k^+)^{j+1} \leftarrow \max\{0, (\mu_k^+)^{j+1}\}$ $\quad k = 1, \ldots, K-1$
- $(\mu_k^-)^{j+1} \leftarrow \max\{0, (\mu_k^-)^{j+1}\}$ $\quad k = 1, \ldots, K-1$
- $w^{j+1} \leftarrow \eta^j + \beta(\tilde{G}(2z^{j+1} - \zeta^j) - \tilde{g})$
- $v^{j+1} \leftarrow \max\{0, \chi^j + \beta(\tilde{H}(2z^{j+1} - \zeta^j) - \tilde{h})\}$
- $\zeta^{j+1} \leftarrow (1-\rho)\zeta^j + \rho z^{j+1}$
- $\eta^{j+1} \leftarrow (1-\rho)\eta^j + \rho w^{j+1}$
- $\chi^{j+1} \leftarrow (1-\rho)\chi^j + \rho v^{j+1}$

end for

$$
z^j = (\hat{x}_1^j, \ldots, \hat{x}_K^j, \hat{u}_1^j, \ldots, \hat{u}_K^j, (\mu_1^+)^j, \ldots, (\mu_{K-1}^+)^j, (\mu_1^-)^j, \ldots, (\mu_{K-1}^-)^j) \tag{12}
$$

$$
\tilde{G} = \begin{bmatrix} \hat{G}_x \ \hat{G}_u \ \hat{G}_\mu \end{bmatrix} \tag{13}
$$

$$
\tilde{g} = -(\hat{w}_1, \ldots, \hat{w}_{K-1}) \tag{14}
$$

$$\tilde{H} = \begin{bmatrix} \hat{H}_y & 0_{n_y(K-1)\times n_u K + 2n_x(K-1)} \end{bmatrix} \tag{15}$$

$$\tilde{h} = (\varepsilon_1, \dots, \varepsilon_{K-1}) \tag{16}$$

3.2.1 Customization

WIP

3.3 Signed Distance

The signed-distance of z to set $\mathcal{Y}$ is defined as

$$\mathrm{sd}(z, \mathcal{Y}) = \min_{y\in\mathcal{Y}} \|z - y\|_2 - \min_{y\in\mathcal{Y}^c} \|z - y\|_2 \tag{17}$$

and its gradient with respect to z is defined as

$$\partial \mathrm{sd}(z, \mathcal{Y}) = \begin{cases} \dfrac{z - \operatorname*{argmin}_{y\in\partial\mathcal{Y}} \|z - y\|_2}{\mathrm{sd}(z, \mathcal{Y})} & \text{if } z \notin \partial\mathcal{Y} \\ \hat{n}(z, \mathcal{Y}) & \text{otherwise} \end{cases} \tag{18}$$

where $\partial\mathcal{Y}$ is the boundary of set $\mathcal{Y}$, and $\hat{n}(z, \mathcal{Y})$ is an outward unit-normal at a point $z \in \partial\mathcal{Y}$.

3.4 Passive Safety

Given a trajectory $x(\tau) \in \mathbb{R}^{n_x}$, for $\tau \in [\tau_\mathrm{i}, \tau_\mathrm{f}]$, passive safety with respect to set $\mathcal{A} \subset \mathbb{R}^{n_x}$ requires that the free-drift starting from the trajectory does not intersect with $\mathcal{A}$ for a finite duration, say τ_s. In other words:

$$x^\tau(\gamma) \notin \mathcal{A} \tag{19}$$

for $\gamma \in [0, \tau_\mathrm{s}]$, where $x^\tau(\gamma)$ solve the following initial value problem over $[0, \gamma]$:

$$\frac{\mathrm{d}x(\gamma)}{\mathrm{d}\gamma} = \overset{\diamond}{x}(\gamma) = f(x(\gamma), 0_{n_u}) \tag{20}$$

with initial condition $x^\tau(0) = x(\tau)$. For simplicity, we assume that $\mathcal{A}$ is a polytope defined by:

$$\mathcal{A} = \{\, z \in \mathbb{R}^{n_x} \mid H_\mathcal{A} z \le h_\mathcal{A} \,\}$$

Since (19) is a nonconvex constraint, we will linearize (19) with respect to an arbitrary trajectory $\bar{x}(\tau) \in \mathbb{R}^{n_x}$, for $\tau \in [\tau_\mathrm{i}, \tau_\mathrm{f}]$.

For each $\tau \in [\tau_\mathrm{i}, \tau_\mathrm{f}]$, let $\bar{x}^\tau(\gamma)$ denote the solution to (20) over $[0, \tau_\mathrm{s}]$ with initial condition $\bar{x}(\tau)$. The Jacobian of f in (20) evaluated on $\bar{x}^\tau(\gamma)$ is denoted by:

$$A^\tau(\gamma) = \frac{\partial f(\bar{x}^\tau(\gamma), 0_{n_u})}{\partial x}$$

Then, the free-drift $x^\tau(\gamma)$, for $\gamma \in [0, \tau_\mathrm{s}]$, starting from $x(\tau)$ can be approximated as:

$$\begin{aligned} x^\tau(\gamma) &\approx \bar{x}^\tau(\gamma) + \Phi^\tau(\gamma, 0)(x(\tau) - \bar{x}(\tau)) \\ &= \Phi^\tau(\gamma, 0)x(\tau) + \phi^\tau(\gamma, 0) \end{aligned}$$

where $\Phi^\tau(\gamma, 0)$ solves the following initial value problem over $[0, \gamma]$:

$$\overset{\diamond}{\Phi}{}^\tau(\gamma, 0) = A^\tau(\gamma)\Phi^\tau(\gamma, 0) \tag{21a}$$

$$\Phi^{\tau}(0,0) = I_{n_x} \tag{21b}$$

and $\phi^{\tau}(\gamma, 0) = \bar{x}^{\tau}(\gamma) - \Phi^{\tau}(\gamma, 0)\bar{x}(\tau)$.

The passive safety constraint on $x(\tau)$ can be approximated as

$$\begin{aligned}
& \Phi^{\tau}(\gamma,0)x(\tau) + \phi^{\tau}(\gamma,0) \notin \mathcal{A} \\
\iff & x(\tau) \notin \mathcal{B}^{\tau}_{\gamma} = \{\, z \,|\, H_{\mathcal{A}}\Phi^{\tau}(\gamma,0) \le h_{\mathcal{A}} - H_{\mathcal{A}}\phi^{\tau}(\gamma,0) \,\} \\
\iff & \mathrm{sd}(x(\tau), \mathcal{B}^{\tau}_{\gamma}) > 0 \\
\impliedby & \mathrm{sd}(\bar{x}(\tau), \mathcal{B}^{\tau}_{\gamma}) + \partial\mathrm{sd}(\bar{x}(\tau), \mathcal{B}^{\tau}_{\gamma})(x(\tau) - \bar{x}(\tau)) > 0
\end{aligned}$$

for $\gamma \in [0, \tau_{\mathrm{s}}]$, where $\Phi^{\tau}(\gamma, 0)$ is invertible. The final implication holds because of the convexity of $\mathcal{B}^{\tau}_{\gamma}$.

The sets $\mathcal{B}^{\tau}_{\gamma}$ are the backward reachable sets (BRS) associated with the polytope $\mathcal{A}$ for the affine system:

$$\overset{\diamond}{x}(\gamma) = A^{\tau}(\gamma)x(\gamma) + f(\bar{x}^{\tau}(\gamma), 0_{n_u}) - A^{\tau}(\gamma)\bar{x}^{\tau}(\gamma) \tag{22}$$

Suppose that $\hat{x}^{\tau}(\gamma)$, for $\gamma \in [0, \tau_{\mathrm{s}}]$, is a solution to (22), and let $\hat{x}^{\tau}(0) \in \mathcal{B}^{\tau}_{\tilde{\gamma}}$, for some $\tilde{\gamma} \in [0, \tau_{\mathrm{s}}]$. Then, $\hat{x}^{\tau}(\tilde{\gamma}) \in \mathcal{A}$.

3.4.1 `node-only-cnstr`

Impose (19) at the nodes of a 2D grid defined in $[\tau_{\mathrm{i}}, \tau_{\mathrm{f}}] \times [0, \tau_{\mathrm{s}}]$.

Similar to (6), define a grid with K_{s} nodes in $[0, \tau_{\mathrm{s}}]$:

$$0 = \gamma_1 < \ldots < \gamma_{K_{\mathrm{s}}} = \tau_{\mathrm{s}}$$

Given a arbitrary sequence of state $\bar{x}_k$ at nodes τ_k, for $k = 1, \ldots, K$, compute $\Phi^{\tau_k}(\gamma_j, 0)$ by solving (21) with $A^{\tau_k}(\gamma)$ evaluated on the free drift starting from $\bar{x}_k$. Then, compute the backward reachable set $\mathcal{B}^{\tau_k}_{\gamma_j}$

$$\begin{aligned}
& \mathrm{sd}(\bar{x}_k, \mathcal{B}^{\tau_k}_{\gamma_j}) + \partial\mathrm{sd}(\bar{x}_k, \mathcal{B}^{\tau_k}_{\gamma_j})(x_k - \bar{x}_k) > 0 \\
\implies & \Phi^{\tau_k}(\gamma_j, 0)x_k + \phi^{\tau_k}(\gamma_j, 0) \notin \mathcal{A}
\end{aligned}$$

for $k = 1, \ldots, K$ and $j = 1, \ldots, K_{\mathrm{s}}$.

4 Examples

4.1 `doubleint-circle-obs`

4.1.1 `node-only-cnstr`

Solve free-final-time problem with time-dilation and constraints imposed at nodes τ_k.

Dynamical system:

$$\overset{\circ}{x}(\tau) = \begin{bmatrix} \overset{\circ}{r}(\tau) \\ \overset{\circ}{v}(\tau) \end{bmatrix} = s(\tau) \begin{bmatrix} v(\tau) \\ T(\tau) + a - c_{\mathrm{d}}\|v(\tau)\|_2 v(\tau) \end{bmatrix} = f(x(\tau), u(\tau)) \tag{23}$$

for $\tau \in [0, 1]$, where $x = (r, v)$ and $u = (T, s)$.

Quantity	Code variable	Meaning (non-dimensional)	Membership
r	`r`	Position	$\mathbb{R}^2$
v	`v`	Velocity	$\mathbb{R}^2$
T	`T`	Acceleration input	$\mathbb{R}^2$
a	`accl`	External acceleration	$\mathbb{R}^2$
c_d	`c_d`	Drag coefficient	$\mathbb{R}_+$

This system model is provided in `plant.doubleint.dyn_func`.

Constraints:

$$
\begin{aligned}
\|r(\tau) - r^i_\mathrm{obs}\|_2 &\geq q^i_\mathrm{obs} \qquad i = 1,2 && \text{(nonconvex)}\\
\|v(\tau)\|_2 &\leq v_\mathrm{max} \\
\|T(\tau)\|_2 &\leq T_\mathrm{max} \\
T_\mathrm{min} &\leq \|T(\tau)\|_2 && \text{(nonconvex)}
\end{aligned}
$$

Quantity	Code variable(s)	Meaning	Membership
r^i_obs, q^i_obs	`robs(:,i)`, `qobs(i)`	Center and radius of ith circular obstacle	$\mathbb{R}^2$, $\mathbb{R}_+$
v_max	`vmax`	Speed upper-bound	$\mathbb{R}_+$
T_min, T_max	`Tmin`, `Tmax`	Upper- and lower-bounds on input magnitude	$\mathbb{R}_+$

4.1.2 `ctcs`

Solve free-final-time problem with time-dilation and constraints augmented to the system dynamics (23).

Inequality constraints function:

$$
g(r,v,T) = \begin{bmatrix} -\|r - r^1_\mathrm{obs}\|_2 + q^1_\mathrm{obs} \\ -\|r - r^2_\mathrm{obs}\|_2 + q^2_\mathrm{obs} \\ \|v\|_2^2 - v^2_\mathrm{max} \\ -\|T\|_2 + T_\mathrm{min} \end{bmatrix}
$$

Dynamical system:

`exclusive-integrator-states`

$$
\mathring{x}(\tau) = \begin{bmatrix} \mathring{r}(\tau) \\ \mathring{v}(\tau) \\ \mathring{y}(\tau) \end{bmatrix} = s(\tau) \begin{bmatrix} v(\tau) \\ T(\tau) + a - c_\mathrm{d}\|v(\tau)\|_2 v(\tau) \\ |g(r,v,T)|_+^2 \end{bmatrix} \tag{24}
$$

for $\tau \in [0,1]$, where $y(\tau) \in \mathbb{R}^4$.

`single-integrator-state`

$$
\mathring{x}(\tau) = \begin{bmatrix} \mathring{r}(\tau) \\ \mathring{v}(\tau) \\ \mathring{y}(\tau) \end{bmatrix} = s(\tau) \begin{bmatrix} v(\tau) \\ T(\tau) + a - c_\mathrm{d}\|v(\tau)\|_2 v(\tau) \\ 1_4^\top |g(r,v,T)|_+^2 \end{bmatrix} \tag{25}
$$

for $\tau \in [0,1]$, where $y(\tau) \in \mathbb{R}$.

4.2 `doubleint-ellip-obs`

4.2.1 `node-only-cnstr`

4.2.2 `ctcs`

4.3 `doubleint-polytope-obs`

4.3.1 `ctcs`

Solve free-final-time problem with time-dilation and constraints augmented to the system dynamics (23).

Inequality constraints functions:

$$g(r, v, T) = \begin{bmatrix} -\mathrm{sd}(r, \mathcal{P}^1_{\mathrm{obs}}) \\ -\mathrm{sd}(r, \mathcal{P}^2_{\mathrm{obs}}) \\ \|v\|_2^2 - v_{\mathrm{max}}^2 \\ -\|T\|_2 + T_{\mathrm{min}} \end{bmatrix}$$

The obstacle $\mathcal{P}^i_{\mathrm{obs}}$, for $i = 1, 2$, is a polytope define by

$$\mathcal{P}^i_{\mathrm{obs}} = \{z \,|\, H^i_{\mathrm{obs}} z \leq h^i_{\mathrm{obs}}\}$$

The code variables for the ith polytope parameters are `Hobs{i}` and `hobs{i}`.

Dynamical system is the same as (24).

4.4 `doubleint-passive-safety`

4.4.1 `node-only-cnstr`

Notes:

- Obtain solution on a coarse grid, which inevitably exhibit inter-sample violation.
- Use coarse-grid solution to warm-start SCP for a finer grid. Such a successive warm-start process is very effective and reliable for obtaining high-quality (low-cost) solutions.

4.4.2 `ctcs-fixed-step-integrate`

4.5 `rocket-landing`

4.5.1 `node-only-cnstr`

4.5.2 `ctcs`

4.6 `nrho-passive-safety`