OPTIMAL SWITCHING INSTANTS FOR THE CONTROL OF HYBRID SYSTEMS

J. Alexandre dit Sandretto, A. Chapoutot, and O. Mullier
Optimal control

Generic case

\[
\begin{align*}
\max_{u(.)} \quad & J(u(.)) = \int_0^T h(x(t), u(t)) dt + g(x(T)) \quad \text{(cost function)} \\
\text{s. t.} \quad & \dot{x} = f(x(t), u(t)), \ 0 < t \leq T \quad \text{(dynamical constraint)} \\
& x(0) = x_0, \ h(x(t)) \in \mathcal{H}, \ 0 < t \leq T \quad \text{(boundary conditions)} \\
& u(t) \in \mathcal{U}, \ \forall t \quad \text{(bounded control)}
\end{align*}
\]

- The dynamical constraint coupled with the boundary conditions is an IVP-ODE depending on a control function \( u(t) \);
- the cost function is a pay-off function where \( x(t) \) is the solution for the dynamical constraint, \( h \) is the running cost and \( g \) is the terminal cost.
An IVP-ODE is defined by

\[
\begin{cases}
\dot{x} = f(x) \\
x(0) \in \mathcal{X}_0 \subseteq \mathbb{R}^n, \ t \in [0, t_{\text{end}}].
\end{cases}
\]

The goal is to compute \( x(t; \mathcal{X}_0) = \{x(t; x_0) \mid x_0 \in \mathcal{X}_0\} \).

**Phase 1** a priori enclosure \( [\tilde{x}_i] \) of

\[ \{x(t_k; x_i) \mid t_k \in [t_i, t_{i+1}], x_i \in [x_i]\} \]

**Phase 2** tight enclosure of \( [x_{i+1}] \) at time \( t_{i+1} \).
Dynibex

- C++ library using ibex (constraint processing over real numbers);
- proof of existence and uniqueness of solution for ODEs and DAEs;
- combined with contractors (HC4), easy to use in branching algorithms;
- verification of temporal constraints.

Example of temporal constraints

- Stayed in $\mathcal{A}$ until $\tilde{t} < t_{\text{end}}$:

  $$\forall t \in [0, \tilde{t}] , \{ y(t; y_0) | y_0 \in [y_0] \} \subseteq \text{int}(\mathcal{A})$$

- Included in $\mathcal{A}$ inside $[0, t_{\text{end}}]$:

  $$\exists t \in [0, t_{\text{end}}] , \{ y(t; y_0) | y_0 \in [y_0] \} \subseteq \text{int}(\mathcal{A}).$$
Example

System of Rossler: Initial states: (0; −10.3; 0.03), some parameters: 
a = 0.2, b = 0.2, c = 5.7

\[
\begin{align*}
\dot{x} &= -y - z \\
\dot{y} &= x + ay \\
\dot{z} &= b + z(x - c)
\end{align*}
\]
For $\mathcal{U} \subseteq [u] \in \dot{x} \in f(x(t), [u])$ and

$$J(u(.)) = \int_0^T h(x(t), u(t))dt + g(x(T))$$

$$= \sum_{i=0}^{n} \int_{t_i}^{t_{i+1}} h(x(t), u(t))dt + g(x(T))$$

$$\in \sum_{i=0}^{n} (t_{i+1} - t_i) h([\tilde{x}_i], [u]) + g([x_T])$$
\[
\begin{align*}
\max_{u(\cdot)} & \quad J(u(\cdot)) = \int_0^T h(x(t), u(t))dt + g(x(T)) \quad \text{(cost function)} \\
\text{s. t.} & \quad \dot{x} = f(x(t), u(t)), \ 0 < t \leq T \quad \text{(dynamical constraint)} \\
& \quad x(0) = x_0, \ h(x(t)) \in \mathcal{H}, \ 0 < t \leq T \quad \text{(boundary conditions)} \\
& \quad u(t) \in \mathcal{U}, \ \forall t \quad \text{(bounded control)}
\end{align*}
\]

**Restriction**

- Particular kind of dynamics:
  - the integral is provided by the **dynamical constraints**, 
  - the set of possible control \(u(t)\) is known and is discrete;

- the **cost function** is monotonic;

- the **boundary conditions** only occurs at a specific time \(\tau\).
$n$-mode hybrid system

\begin{equation}
(S_i) \left\{ \begin{align*}
\dot{x} &= f_i(x) \\
x(t_i) &= x_i
\end{align*} \right. \text{ in the time interval } [t_i, t_{i+1}]
\end{equation}

- $f_i : \mathbb{R}^m \to \mathbb{R}^m$;
- $x_i \in \mathbb{R}^m$ is the initial condition for all modes $0 \leq i \leq n - 1$.

A sequence $\{(S_1), \ldots, (S_k)\}$ corresponds to the switching of control law.

- $x_0$ is fixed;
- $x_i$ is taken as the solution at time $t_i$ of $(S_{i-1})$.

⇒ not necessarily continuously differentiable.
Our problem can be modeled using the following optimization problem

\[
\begin{align*}
\max_{t_1, \ldots, t_{n-1}} & \quad g(x(\tau)) \\
\text{s. t.} & \quad \forall 0 \leq i \leq n - 1, (S_i) \\
& \quad h(x(\tau)) > 0 \\
& \quad \tau \in [t_{n-1}, t_n]
\end{align*}
\]

with

- the decision variables \( t_1, \ldots, t_{n-1} \in \mathbb{R}^n_+ \) the search space for the different times;
- the cost function \( g : \mathbb{R}^m \to \mathbb{R} \) on the state variable at given time \( \tau \in [t_{n-1}, t_n] \);
- some constraints defined by the dynamical systems \((S_i)\) and the times \( t_i \);
- a reachability constraint using \( h : \mathbb{R}^m \to \mathbb{R} \).
EXAMPLE: GODDARD’S ROCKET

Model of the ascent of a rocket in the atmosphere:

\[
\begin{align*}
\text{max} & \quad m(T) \\
\text{s.t.} & \quad \dot{r} = v \\
& \quad \dot{\nu} = \frac{u - Av \exp(k(1-r))}{m} - \frac{1}{v^2} \\
& \quad \dot{m} = -bu \\
& \quad u(.) \in [0, 1] \\
& \quad r(0) = 1, \nu(0) = 0, m(0) = 1 \\
& \quad r(T) \geq R_T
\end{align*}
\]

with the parameters

\[
\begin{align*}
& \quad \cdot b = 2, \\
& \quad \cdot T_{\text{max}} = 0.2, \\
& \quad \cdot A = 310, \\
& \quad \cdot k = 500, \\
& \quad \cdot r_0 = 1, \nu_0 = 0, m_0 = 1, \\
& \quad \cdot R_T = 1.01.
\end{align*}
\]

According to the time \( t \):

\[
u(t) = \begin{cases} 
3.5 & \text{for } t \in [0, t_1] \ (S_0) \\
3.5 \tanh(1 + t) & \text{for } t \in [t_1, t_2] \ (S_1) \\
0 & \text{for } t \in [t_2, T] \ (S_2)
\end{cases}
\]
Algorithm 1: simu\( (t_1, t_2, \text{max}) \) – simulates the system from 0 to \( T \).

**Input:** time \( t_1, t_2 \) to switch dynamics; current maximum mass \( \text{max} \)

**Output:** the mass \( m \) or 0 if simulation will not produce a better solution

\[
([r_{t_1}], [v_{t_1}], [m_{t_1}]) \leftarrow \text{simulation of } x_0 \text{ using } (S_0) \text{ from 0 to } t_1;
\]

if \( m_{t_1} \leq \text{max} \) then

\[ \text{return } 0; \]

\[
([r_{t_2}], [v_{t_2}], [m_{t_2}]) \leftarrow \text{simulation using } (S_1) \text{ from } t_1 \text{ to } t_2;
\]

if \( m_{t_2} \leq \text{max} \) then

\[ \text{return } 0; \]

\[
([r_T], [v_T], [m_T]) \leftarrow \text{simulation using } (S_2) \text{ from } t_2 \text{ to } T;
\]

if \( r_T \geq R_T \) then

\[ \text{return } [m_T]; \]

else

\[ \text{return } 0; \]
Algorithm 2: finds the optimal switching times

Input: set of dynamics \{({\mathcal{S}_0}), ({\mathcal{S}_1}), ({\mathcal{S}_2})\}

Output: switching times \( t_{1,\text{max}} \) and \( t_{2,\text{max}} \)

\[
\text{max} \leftarrow 0; \\
\text{for } t_1 \leftarrow 0 \text{ to } T - \epsilon \text{ do} \\
\hspace{1em} \text{for } t_2 \leftarrow t_1 + \epsilon \text{ to } T \text{ do} \\
\hspace{2em} [m, \overline{m}] \leftarrow \text{simu}(t_1, t_2); \\
\hspace{2em} \text{if } m > \text{max} \text{ then} \\
\hspace{3em} m \leftarrow \text{max}; \\
\hspace{3em} t_{1,\text{max}} \leftarrow t_1; \\
\hspace{3em} t_{2,\text{max}} \leftarrow t_2; \\
\hspace{2em} \text{else} \\
\hspace{3em} \text{break;} \hspace{1em} \text{// Due to monotonicity of the cost function} \\
\hspace{1em} \text{return } t_{1,\text{max}} \text{ and } t_{2,\text{max}}; \\
\]

Optimal switching instants for the Control of Hybrid Systems – O. Mullier et al.
Figure 1: Mesh for $t_2$ w.r.t. $t_1$
Figure 2: Optimal controller
Conclusion

- Promising results on the computation of optimal switching mode;
- easy-to-use tool development;
- benefits shown on an example.

Perspectives

To release restrictions to handle the problem in its generality
Thank you
Robert H Goddard.  
A method of reaching extreme altitudes.  
*Nature*, 105:809–811, 1920.

Knut Graichen and Nicolas Petit.  
Solving the goddard problem with thrust and dynamic pressure constraints using saturation functions.  
*IFAC Proceedings Volumes*, 41(2):14301–14306, 2008.