Capture Basin Approximation using Interval Analysis

Mehdi Lhommeau, Luc Jaulin and Laurent Hardouin

LISA - Université d’Angers, France
LDTN - ENSIETA, France
# Table of contents

1. **Introduction**
   - The Roller Coaster
   - Nonlinear Dynamical System

2. **Capture Basin**
   - Definition
   - Approximation

3. **Capture Basin Algorithm**
   - Introduction
   - Inclusion function

4. **Algorithm**
   - Results
   - Example

5. **Conclusion**
Curvilinear coordinates:
- $s$ is the position on the track, measured by the path length
- $\dot{s}$ is the velocity of the ball
Nonlinear dynamical system

\[ \dot{x}(t) = f(x(t), u(t)) \] (1)

Where \( t > 0 \):
Nonlinear dynamical system

\[ \dot{x}(t) = f(x(t), u(t)) \] (1)

Where \( t > 0 \):

- \( x(t) \in \mathbb{R}^n \) is the state vector
Nonlinear dynamical system

\[ \dot{x}(t) = f(x(t), u(t)) \] (1)

Where \( t > 0 \):
- \( x(t) \in \mathbb{R}^n \) is the state vector
- \( u(t) \in U \subset \mathbb{R}^m \) is the control

Roller Coaster dynamic

If \( x = (s, \dot{s}) \) then

\[
\begin{align*}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= -g \sin(\Theta(x_1)) - \alpha x_2 + u
\end{align*}
\]

Where
- \( \alpha \) : friction force
- \( \Theta : x_1 \mapsto \theta \) is a given function
- \( u \) is the control
Flow

\[ t \mapsto x(t) = \varphi(t; x_0, u), \quad (2) \]

is the unique solution to (1). Where
Capture Basin Approximation using Interval Analysis

Mehdi Lhommeau, Luc Jaulin and Laurent Hardouin

Introduction

The Roller Coaster Nonlinear Dynamical System

Capture Basin Definition Approximation

Capture Basin Algorithm

Algorithm Introduction Inclusion function

Algorithm Results Example

Conclusion

Flow

\[ t \mapsto x(t) = \varphi(t; x_0, u), \quad (2) \]

is the unique solution to (1). Where

- \( x(0) = x_0 \) is the initial condition
Flow

\[ t \mapsto x(t) = \varphi(t; x_0, u), \quad (2) \]

is the unique solution to (1). Where

- \( x(0) = x_0 \) is the initial condition
- \( u \in L^\infty([0, \infty), U) \) is the control function
Flow

\[ t \mapsto x(t) = \varphi(t; x_0, u), \quad (2) \]

is the unique solution to (1). Where

- \( x(0) = x_0 \) is the initial condition
- \( u \in L^\infty([0, \infty), U) \) is the control function

The whole trajectory is given by

\[ \varphi([0, t]; x_0, u) = \bigcup_{\tau \in [0, t]} \varphi(\tau; x_0, u). \quad (3) \]
Define two compact sets $T$ and $K$ such that $T \subset K \subset \mathbb{R}^n$. $T$ is the target and $K$ is the viable set. The capture basin $C$ is the subset of states of $K$ from which there exists at least one solution inside $K$ reaching the target $T$ in finite time $t$:
Capture Basin

Define two compact sets $T$ and $K$ such that $T \subset K \subset \mathbb{R}^n$. $T$ is the target and $K$ is the viable set. The capture basin $C$ is the subset of states of $K$ from which there exists at least one solution inside $K$ reaching the target $T$ in finite time $t$:

$$C = \{ x_0 \in K \mid \exists t > 0, \exists u \in L^\infty([0, t], U), \varphi(t; x_0, u) \in T \text{ and } \varphi([0, t]; x_0, u) \subset K \}.$$
The aim

Find two sets $C^-$ and $C^+$ such that
The aim

Find two sets $\mathbf{C}^-$ and $\mathbf{C}^+$ such that

$$\mathbf{C}^- \subset \mathbf{C} \subset \mathbf{C}^+$$
Proposition 1

We have

\[(i) \ x_0 \in T \Rightarrow x_0 \in C\]
Proposition 1

We have

(i) \( x_0 \in T \Rightarrow x_0 \in C \)

(ii) \( x_0 \notin K \Rightarrow x_0 \notin C \)
**Proposition 1**

We have

(i) \( x_0 \in T \Rightarrow x_0 \in C \)

(ii) \( x_0 \notin K \Rightarrow x_0 \notin C \)

(iii) \( \exists u, \varphi(t; x_0, u) \in C \land \varphi([0, t]; x_0, u) \subset K \) \( \Rightarrow x_0 \in C \)
Proposition 1

We have

(i) \( x_0 \in T \Rightarrow x_0 \in C \)

(ii) \( x_0 \notin K \Rightarrow x_0 \notin C \)

(iii) \( (\exists u, \varphi(t; x_0, u) \in C \land \varphi([0, t]; x_0, u) \subset K) \Rightarrow x_0 \in C \)

(iv) \( (\forall u, \varphi(t; x_0, u) \cap C = \emptyset \land \varphi([0, t]; x_0, u) \cap T = \emptyset) \Rightarrow x_0 \notin C \)
Guaranteed Numerical Integration

**Notation**

If \([t] \in \mathbb{IR}, [x_0] \in \mathbb{IR}^n\) and \([u] \in \mathbb{IR}^m\)

\[
[\varphi]( [t]; [x_0], [u]) \overset{\text{def}}{=} \{ \varphi(t; x_0, u) \mid t \in [t], x_0 \in [x_0], u \in L^\infty([0, t], [u]) \}.
\]
Proposition 2

\((i)\) \([x_0] \subset T \Rightarrow [x_0] \subset C\)
**Proposition 2**

(i) \([x_0] \subset T \Rightarrow [x_0] \subset C\)

(ii) \([x_0] \cap K = \emptyset \Rightarrow [x_0] \cap C = \emptyset\)
Proposition 2

(i) $[x_0] \subset T \Rightarrow [x_0] \subset C$

(ii) $[x_0] \cap K = \emptyset \Rightarrow [x_0] \cap C = \emptyset$

(iii) $(\exists u \in [u], [\varphi](t; [x_0], u) \subset C \land ([\varphi]([0, t]; [x_0], u)) \subset K) \Rightarrow [x_0] \subset C$

(iv) $([\varphi]([0, t]; [x_0], u) \cap C = \emptyset \land ([\varphi]([0, t]; [x_0], u) \cap T = \emptyset) \Rightarrow [x_0] \cap C = \emptyset$
Proposition 2

(i) \([x_0] \subset T \Rightarrow [x_0] \subset C\)

(ii) \([x_0] \cap K = \emptyset \Rightarrow [x_0] \cap C = \emptyset\)

(iii) \((\exists u \in [u], \ [\varphi](t; [x_0], u) \subset C \land ([\varphi](0, t; [x_0], u)) \subset K) \Rightarrow [x_0] \subset C\)

(iv) \(([\varphi](t; [x_0], [u]) \cap C = \emptyset \land [\varphi](0, t; [x_0], [u]) \cap T = \emptyset) \Rightarrow [x_0] \cap C = \emptyset\)
**Algorithm**

- Initialize the sets $C^- = \emptyset$ and $C^+$ is a union of boxes covering $K$
Algorithm

- Initialize the sets \( C^- = \emptyset \) and \( C^+ \) is a union of boxes covering \( K \)
- Iterate :

1. Take a box \([x_0]\) in \( C^+ \)
2. If \([x_0] \subset T\) then \( C^- := C^- \cup [x_0] \); goto 1;
3. If \([x_0] \cap K = \emptyset\) then \( C^+ := C^+ \setminus [x_0] \); goto 1;
4. Take \( t \in \mathbb{R}^+ \) and \( u \in [u] \);
5. If \([\varphi](t; [x_0], u) \subset C^- \) and \([\varphi]([0, t]; [x_0], u) \subset K\) then \( C^- := C^- \cup [x_0] \); goto 1;
6. If \([\varphi](t; [x_0], [u]) \cap C^+ = \emptyset\) and \([\varphi]([0, t]; [x_0], [u]) \cap T = \emptyset\) then \( C^+ := C^+ \setminus [x_0] \); goto 1;

- Until no more change can be observed
Output of the algorithm

The Roller Coaster Nonlinear Dynamical System

Capture Basin Definition Approximation

Algorithm Introduction Inclusion function

Example

Conclusion
Conclusion

- We have presented a new algorithm that provides guaranteed inner and outer approximations of the capture basin
  - Hybrid systems
    - Roller coaster where the ball can jump off the track
Conclusion

- We have presented a new algorithm that provides guaranteed inner and outer approximations of the capture basin
  - Hybrid systems
    - Roller coaster where the ball can jump off the track
- Inner and outer approximations of the kernel viability

\[ V = \{ x_0 \in K \mid \exists u \in L^\infty([0, \infty), U), \forall t > 0, \varphi(t; x_0, u) \in K \} \]