Formal Verification of State and Temporal Properties of Neural Network-Controlled Systems

Antoine Besset, Joris Tillet, and Julien Alexandre dit Sandretto

Projet STARTS
Sécurité et Probabilité de Succès
(SafeTy And pRobAbiliTy Success)
starts.ensta-paris.fr

Summary : Ensuring the safety of Neural Network Controlled Systems (NNCS) remains a major challenge due to the opacity of neural networks, especially when temporal properties are involved. By combining **interval analysis** with **reachability techniques**, we ensure compliance with **spatial and temporal specifications** formally expressed using **Signal Temporal Logic (STL)**. The proposed framework provides a rigorous method for guaranteeing the correctness of uncertain dynamical systems controlled by a neural network.

Cyber-physical systems

Consider a continuous dynamical system modeled by the following differential equation:

$$\dot{y}(t) = f(y(t), w(t)), \quad y(t) \in \mathbb{R}^n, \ w(t) \in \mathcal{W}, \tag{1}$$

where $y(t)$ is the state of the system, $w(t)$ is a bounded external input, and $\mathcal{W} \subseteq \mathbb{R}^p$ is a compact set. For any initial state $y_0 \in \mathbb{R}^n$ and any measurable input $w : \mathbb{R}^+ \to \mathcal{W}$, the system admits a unique trajectory denoted by $\xi(\cdot, y_0, w)$.
In the presence of **bounded uncertainty**, the objective is to determine **the set of possible trajectories** over the interval $[t_0, T]$. The set of reachable states at time $t \in \mathbb{R}^+$ from an initial set $\mathcal{Y}_0 \subseteq \mathbb{R}^n$ is defined as:

$$\text{Reach}_t(\mathcal{Y}_0, \mathcal{W}) = \{\xi(t, y_0, w) \mid y_0 \in \mathcal{Y}_0, \ w(s) \in \mathcal{W}, \ \forall s \in [0, t]\}. \tag{2}$$

A **continuous-time representation** on $[t_j, t_{j+1}]$, $\bigcup_{j=0}^{N-1}[t_j, t_{j+1}] = [t_0, T]$, called a ***tube*** and denoted $[\tilde{y}](t)$ for $t \in [t_0, T]$, is essential for preserving the set of all possible system behaviors [6]. This *tube* enables the analysis of the satisfaction of a temporal logic formula.

Combining STL and reachability analysis

To formally analyze the system behavior, we use *interval analysis* [5] and introduce a **set - valued extension of predicates**:

$$([\tilde{y}], t) \vDash \mu_i := \begin{cases} 1, & \text{if } [\tilde{y}](t) \subset \mathcal{X}^\mu, \\ 0, & \text{if } [\tilde{y}](t) \cap \mathcal{X}^\mu = \emptyset, \\ [0,1], & \text{otherwise.} \end{cases} \tag{3}$$

Propagation in temporal logic is handled using **Boolean intervals** [2, 7], e.g [3].:

$0 \wedge [0,1] = 0, \quad 0 \vee [0,1] = [0,1], \quad 1 \wedge [0,1] = [0,1], \quad 1 \vee [0,1] = 1.$

Signal temporal logic

We use the formalism of **Signal Temporal Logic (STL)** [4]:

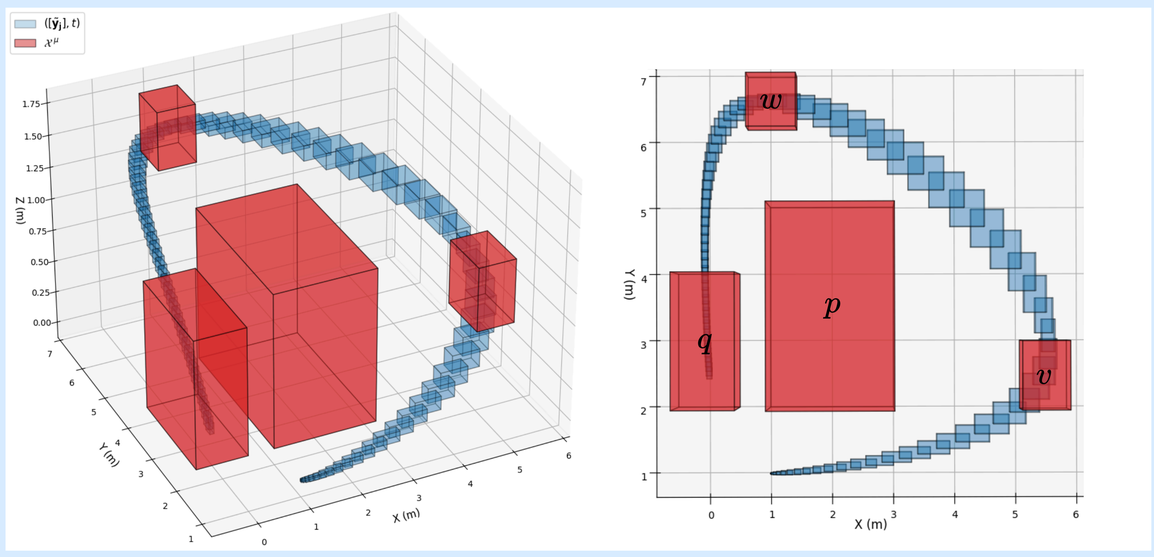
$$\varphi := \top \mid \mu \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \, \mathbf{U}_{[t_1,t_2]} \varphi_2$$

where the temporal operator **U** (*Until*) specifies that a property must hold until another becomes true within a given time interval. The operator **F** (*Finally*) expresses that a goal must be reached within a time window, while **G** (*Globally*) states that a property must hold throughout a time interval.

Example: an automaton executing a periodic task.

$$\varphi = \mathbf{G}_{[0,5]}(v \Rightarrow \mathbf{F}_{[3.5,4.5]} w) \ \wedge \ \mathbf{G}_{[0,8]}(\neg p) \ \wedge \ \mathbf{F}_{[8,9]} q$$

- $\mathbf{G}_{[0,5]}(v \Rightarrow \mathbf{F}_{[3.5,4.5]} w)$: Always on $[0, 5]$s, if v is reached then w must follow within 3.5–4.5s.
- $\mathbf{G}_{[0,8]}(\neg p)$: The obstacle p must never be encountered during the first 8s.
- $\mathbf{F}_{[8,9]} q$: The stand-by zone q must be reached between 8s and 9s.



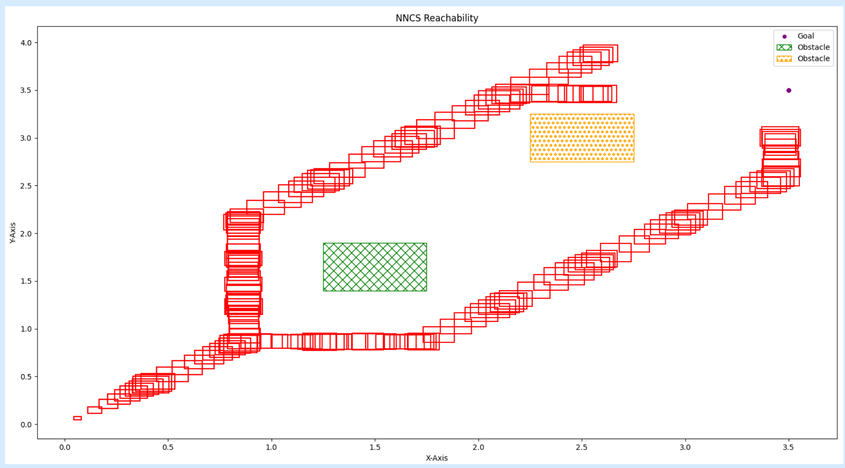
Tube $([\tilde{y}], t)$ *in blue and zones* ($\mathcal{X}^\mu$) *in red.*

Application : A robot controlled by a neural network

A robot is controlled by a **neural network**, whose internal behavior is **difficult to interpret**. The presented methods provide **formal guarantees** that it **reaches its goal**, avoids obstacles, and does so within a **given time bound**, even in the presence of uncertainties [1]. The system employs a neural network to choose, from a set of motion primitives, the action that drives it toward the goal while avoiding obstacles. An example of specification could be:

$$\varphi = \neg C \ \mathbf{U}_{[t_1,t_2]} \ T.$$

This means that no collision ($\neg C$) must occur until the target (T) is reached, within the given time horizon $[t_1, t_2]$.



The reachable tube by the robot is shown in red, obstacles are in green and yellow, the target point is in purple.

Set propagation in neural network

To conduct reachability analysis of NNCS, activation functions such as the sigmoid can be expressed as ODEs, e.g.

$$\frac{d\sigma}{dx}(x) = \sigma(x)(1 - \sigma(x)).$$

This allows ODE solvers to be combined with **affine arithmetic**, where uncertain quantities are represented as

$$x = x_0 + x_1\varepsilon_1 + \cdots + x_n\varepsilon_n, \quad \varepsilon_i \in [-1, 1].$$

Using shared noise symbols preserves dependencies between neurons, enabling **accurate error tracking** and avoiding the overestimation of interval arithmetic.

References

[1] Antoine Besset, Julien Alexandre dit Sandretto, and Joris Tillet. Real-time guaranteed monitoring for a drone using interval analysis and signal temporal logic. In *Proceedings of the 2025 IEEE/RSJ IROS*, Hangzhou, China, 2025. IEEE.
[2] Antoine Besset, Joris Tillet, and Julien Alexandre dit Sandretto. Uncertainty removal in verification of nonlinear systems against signal temporal logic via incremental reachability analysis. In *Proceedings of the 64th IEEE CDC*. IEEE, 2025.
[3] Luc Jaulin, Michel Kieffer, Olivier Didrit, and Eric Walter. Applied interval analysis. In Luc Jaulin, Michel Kieffer, Olivier Didrit, and Eric Walter, editors, *Applied Interval Analysis*, pages 11–43. Springer, 2001.
[4] Oded Maler and Dejan Nickovic. Monitoring temporal properties of continuous signals. In *Formal Techniques, Modelling and Analysis of Timed and Fault-Tolerant Systems*, volume 3253, pages 152–166. Springer, 2004.
[5] Ramon E. Moore. *Interval Analysis*. Series in Automatic Computation. Prentice Hall, 1966.
[6] Julien Alexandre Dit Sandretto and Alexandre Chapoutot. Validated explicit and implicit runge-kutta methods. *Reliable Computing*, 22, 2016. Special issue devoted to material presented at SWIM 2015.
[7] Joris Tillet, Antoine Besset, and Julien Alexandre Dit Sandretto. Guaranteed satisfaction of a signal temporal logic formula on tubes. *Acta Cybernetica*, 2025. Accepted, to appear.

U2IS, ENSTA – Institut Polytechnique de Paris, 828 boulevard des Maréchaux, 91120 Palaiseau, France