Itô–Tanaka Certificates

Filip Morawiec · 2026 University of Birmingham Summer Research Programme in AI Safety
Supervised by **Grigory Neustroev** in the lab of **Prof. Mirco Giacobbe**

Motivation (the big picture). For a complex stochastic system such as a humanoid robot, we want to formally prove safety—for example, that it will not fall while walking.

Motivation (the realistic picture). Previous work [1] assumes that V is twice continuously differentiable. Weakening this assumption could lead to faster verification.

Introduction

We study stochastic differential equations

$$dX_t = f(X_t)\,dt + \sigma(X_t)\,dW_t, \qquad X_t \in K \subset \mathbb{R}^d,$$

on compact K. Given initial, target, and unsafe sets X_0, X_T, X_u, choose $0 \le \alpha < \beta$ and a *nonnegative* $V : K \to \mathbb{R}$ satisfying

$$\sup_{X_0} V \le \alpha, \quad \inf_{X_u} V \ge \beta, \quad \inf_{\partial K \setminus X_T} V \ge \beta.$$

Let τ be the stopping time of reaching X_T, ∂K, or $\{V \ge \beta\}$. This key theorem, as a consequence of Ville's inequality, connects the construction of a supermartingale to safety bounds for the SDE.

Theorem (reach–avoid guarantee). If $V(X_{t\wedge\tau})$ is a supermartingale, then

$$\mathbb{P}(\tau_{X_u} \wedge \tau_{\partial K} < \tau_{X_T}) \le \alpha/\beta.$$

So the probability of reaching the unsafe set or leaving the domain before reaching the target set is bounded by α/β.

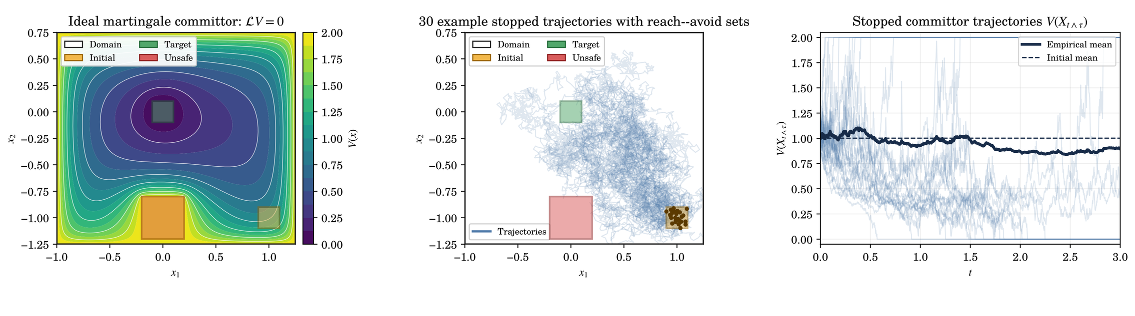


Figure 1. An ideal reach–avoid certificate and realizations of the stopped martingale $V(X_{t\wedge\tau})$. It's numerically estimated to be a martingale.

Following the motivations, these questions arise:

Research Question 1: Can non-C^2 certificates be valid?

Finding: Yes. By applying the Itô–Tanaka–Meyer formula we notice that we have to additionally account for local time.

Research Question 2: What is a useful class of certificates?

Finding: We identify three important properties: piecewise-quadratic structure, representational power, and local-time safety by construction through our **Deep Residual ICNN**.

Research Question 3: Do these certificates work in practice?

Finding: Yes, but so far only in very simplified settings. We identify useful techniques and several failure modes.

RQ1: Validity of non-C^2 certificates

Let V be continuous and C^2 on each polyhedral cell K_i. Across a shared facet F, we have a normal vector $n_{i\to j,F} \in \mathbb{R}^d$ from K_i to K_j. Define the normal-gradient jump to be

$$\gamma_F(x) = \big(\nabla V_j(x) - \nabla V_i(x)\big)^\top n_{i\to j,F}.$$

The multidimensional Itô–Tanaka formula gives

$$V(X_{t\wedge\tau}) = V(X_0) + \int_0^{t\wedge\tau} \mathcal{L}V(X_s)\,ds + M_t + \frac{1}{2}\sum_F \int_0^{t\wedge\tau} \gamma_F(X_s)\,dL_s^F,$$

where $\mathcal{L}V = \nabla V^\top f + \frac{1}{2}\operatorname{tr}(\sigma\sigma^\top \nabla^2 V)$ and L^F is nondecreasing surface local time.

Theorem (Itô–Tanaka certificate). Assume the set constraints above, the stochastic integral M_t is a true martingale, and below level β outside X_T:

$$\mathcal{L}V_i(x) \le 0, \qquad \gamma_F(x) \le 0$$

on every cell and every shared facet. Then $V(X_{t\wedge\tau})$ is a supermartingale and the reach–avoid bound holds.

Proof sketch. For $s \le t$, apply the formula on $(s,t]$ and condition on $\mathcal{F}_s$. The generator integral is nonpositive; $dL^F \ge 0$ and $\gamma_F \le 0$ make every interface integral nonpositive; the martingale increment has conditional mean zero. Hence

$$\mathbb{E}[V(X_{t\wedge\tau}) \mid \mathcal{F}_s] \le V(X_{s\wedge\tau}). \qquad \square$$

RQ2: A useful class of non-C^2 certificates

2.1 Piecewise-quadratic structure

We propose that V has this form:

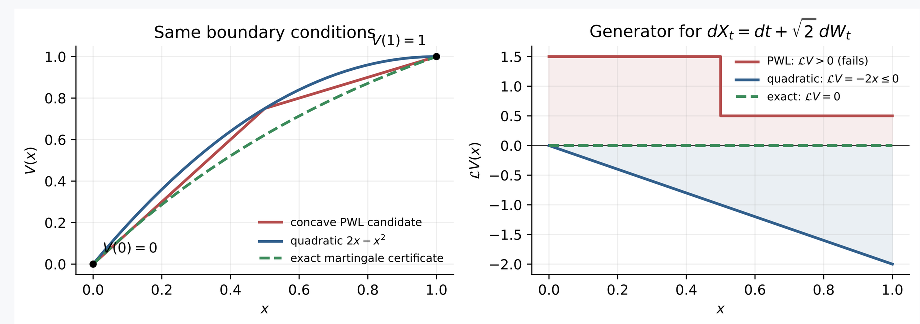
$$V(x) = \sum_{k=1}^{m} \lambda_k \phi(z_k(x)) + c, \quad z(x) = \mathtt{ReLUNN}(x),$$

where ϕ is a fixed continuous piecewise-quadratic activation. The ReLU network partitions K into polyhedral cells and, after refinement by ϕ, $V_i(x) = c_i + p_i^\top x + \frac{1}{2}x^\top Q_i x$. Thus $\nabla V_i = p_i + Q_i x$ and $\nabla^2 V_i = Q_i$, which we use when checking the generator and local-time conditions.

Why is plain ReLU insufficient? For $dX_t = dt + \sqrt{2}\,dW_t$ on $[0,1]$, suppose $V(0) < V(1)$. A continuous piecewise-linear V satisfying $\gamma_F \le 0$ has nonincreasing slopes, but must begin with some positive slope a_i. Inside that piece,

$$\mathcal{L}V = V' + V'' = a_i > 0,$$

contradicting the generator condition. On the other hand, the quadratic $V(x) = 2x - x^2$ works: $\mathcal{L}V - 2x \le 0$.



2.2 Representational power

Complex SDEs can require many quadratic regions. Deep neural networks can represent these regions compactly and discover their partition during training. Piecewise quadratics also have strong approximation power: under strict margins, a smooth 2D certificate can be approximated by valid C^1 piecewise-quadratic splines. Following existing committor-function literature, which uses deep neural networks to train committors [3,4], we use deep networks to learn certificates.

2.3 Local-time safety by construction

Checking the local-time condition creates a verification bottleneck. If the network induces N full-dimensional regions, naively testing all pairs is of order $O(N^2)$.
To avoid manually checking the local-time condition, we propose the **Deep Residual ICNN** architecture

$$\boxed{V_\theta(x) = \rho\big(S_\theta(x) - C_\psi(x)\big), \qquad \rho > 0.}$$

The C^1 piecewise-quadratic branch may be

$$S_\theta(x) = c + p^\top x + \tfrac{1}{2}x^\top H x + \tfrac{1}{2}\sum_k v_k\,\mathrm{ReLU}(a_k^\top x + b_k)^2.$$

The convex piecewise-affine branch is a ReLU Input Convex Neural Network (ICNN) [2].

Theorem (local-time condition by construction). For every pair of adjacent cells and every parameter choice,

$$\big(\nabla V_{\theta,j} - \nabla V_{\theta,i}\big)^\top n_{i\to j,F} \le 0.$$

Therefore the local-time condition is satisfied.

Proof sketch. Restrict convex C_ψ to a line normal to F. The derivative of a one-dimensional convex function is nondecreasing, so

$$(\nabla C_{\psi,j} - \nabla C_{\psi,i})^\top n_{i\to j,F} \ge 0.$$

Because $S_\theta \in C^1$, its gradient jump is zero. Consequently

$$(\nabla V_{\theta,j} - \nabla V_{\theta,i})^\top n = -\rho(\nabla C_{\psi,j} - \nabla C_{\psi,i})^\top n \le 0. \qquad \square$$

RQ3: Experiments and failure modes

At first, we tried to train a plain ReLU MLP with a piecewise-quadratic output layer and found it was difficult to train. After proposing the Deep Residual ICNN architecture, training became easier and we found valid certificates for a simple SDE with loose bounds. We trained it by first approximating a martingale with $\mathcal{L}V = -\epsilon$, *disabling the ICNN branch* and then fitting S_θ with linear programming.

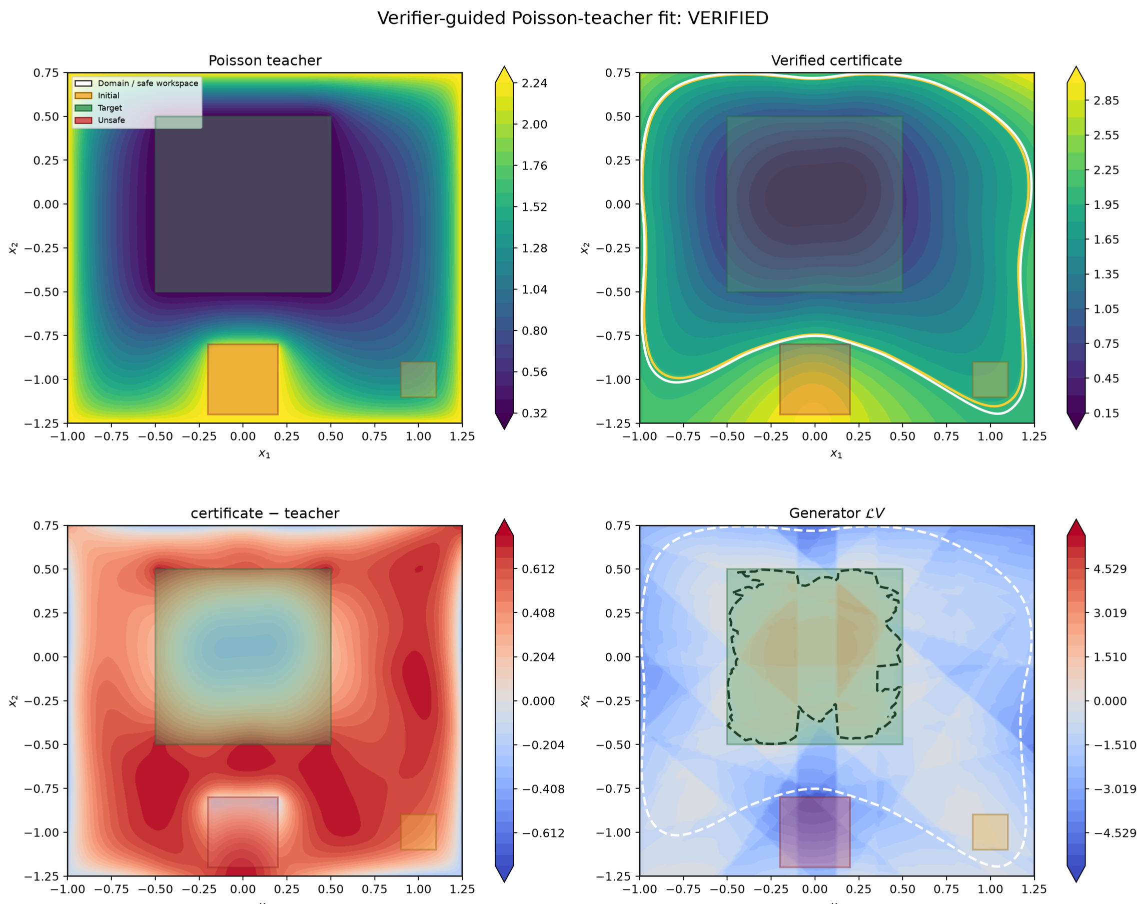


Figure 3. Verified 2D OU benchmark. The verifier checked 1,177 cells with no unresolved cells for $\alpha = 1.97$, $\beta = 2$, and $\epsilon = 0.1$, giving $\mathbb{P}(\text{unsafe or domain exit before target}) \le 0.985$. The realistic optimum bound from the approximation would be roughly $1.2/2 = 0.6$.

Limitations and future work

1. We could not train a valid certificate using Adam alone; success required linear programming on the S_{θ_1} branch.
2. The ICNN branch C_{θ_2} was set to zero in the verified certificate shown above. Nonetheless, it conveyed useful geometry during empirical experiments.
3. **We found a connection to a large body of literature on committor functions, which offers efficient methods for training certificate-like functions** [3,4]. Its sampling methods improved our training with Adam, although they have not yet produced a verified certificate.
4. Although not explored here, a valuable research direction would explore **compositionality**, which could potentially allow for richer specification descriptions (e.g. LTL) [5] and efficient training via a divide-and-conquer-like manner by training multiple certificates and gluing them together.

These results remain limited to simplified settings. The main open challenge is to make both branches train effectively while retaining exact verification.

Mathematical background

Martingales. On a filtered probability space $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\ge 0}, \mathbb{P})$, an adapted integrable process Y_t is a supermartingale when

$$\mathbb{E}[Y_t \mid \mathcal{F}_s] \le Y_s, \qquad s \le t.$$

It is a martingale when equality holds.
Local time. Let $F = \{x : h_F(x) = 0\}$ locally, where h_F is the signed distance to the facet, positive in the direction $n_{i\to j,F}$. The surface local time of X on F is the symmetric semimartingale local time

$$L_t^F := L_t^0\big(h_F(X)\big) = \lim_{\varepsilon\downarrow 0} \frac{1}{2\varepsilon}\int_0^t \mathbf{1}_{\{|h_F(X_s)|<\varepsilon\}}\,d\langle h_F(X)\rangle_s,$$

whenever the limit exists. It is nondecreasing and increases only when $X_t \in F$.

Python package

We propose a Python package, `tanaka_certificates`, with utilities for verifying Tanaka certificates and a thorough test suite. Example usage:

```
sde, problem = make_enlarged_target_ou_problem()
model, _ = train_fixed_pwq_lp() # only for that problem for now
model.eval()
verifier = VerifierLocalTimeByConstruction(sde, problem, model)

# use when local time is not guaranteed by construction
# verifier = VerifierPiecewiseQuadratic(sde, problem, model)

verification = verifier.verify()
assert verification.value == "verified"
```

Listing 1: Verify a piecewise-quadratic certificate.

Blogpost
and Github repository
filipmorawiec.com/tanaka

[1] G. Neustroev et al., "Neural Continuous-Time Supermartingale Certificates," *AAAI*, 2025, doi:10.1609/aaai.v39i26.34966. [2] B. Amos, L. Xu, and J. Z. Kolter, "Input Convex Neural Networks," arXiv:1609.07152, 2017. [3] Y. Khoo, J. Lu, and L. Ying, "Solving for high dimensional committor functions using artificial neural networks," arXiv:1802.10275. [4] Y. Wang et al., "Estimating Committor Functions via Deep Adaptive Sampling on Rare Transition Paths," arXiv:2501.15522. [5] A. Abate, M. Giacobbe, and D. Roy, "Quantitative Supermartingale Certificates," arXiv:2504.05065.

Contact: filip.morawiec@gmail.com