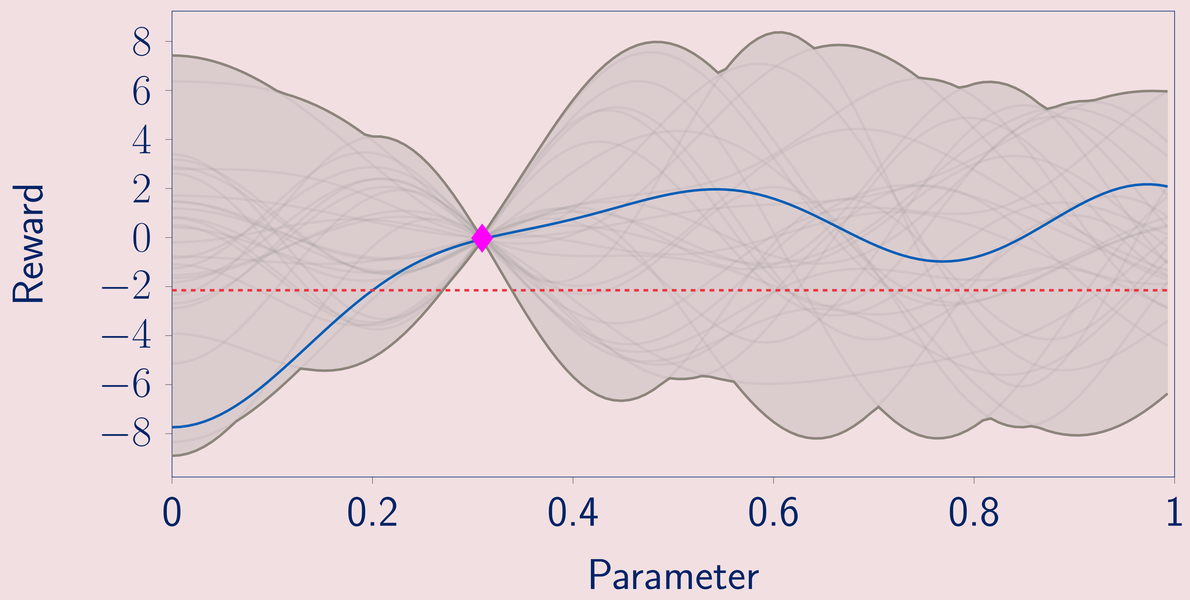
Safe learning-based control via function-based uncertainty quantification

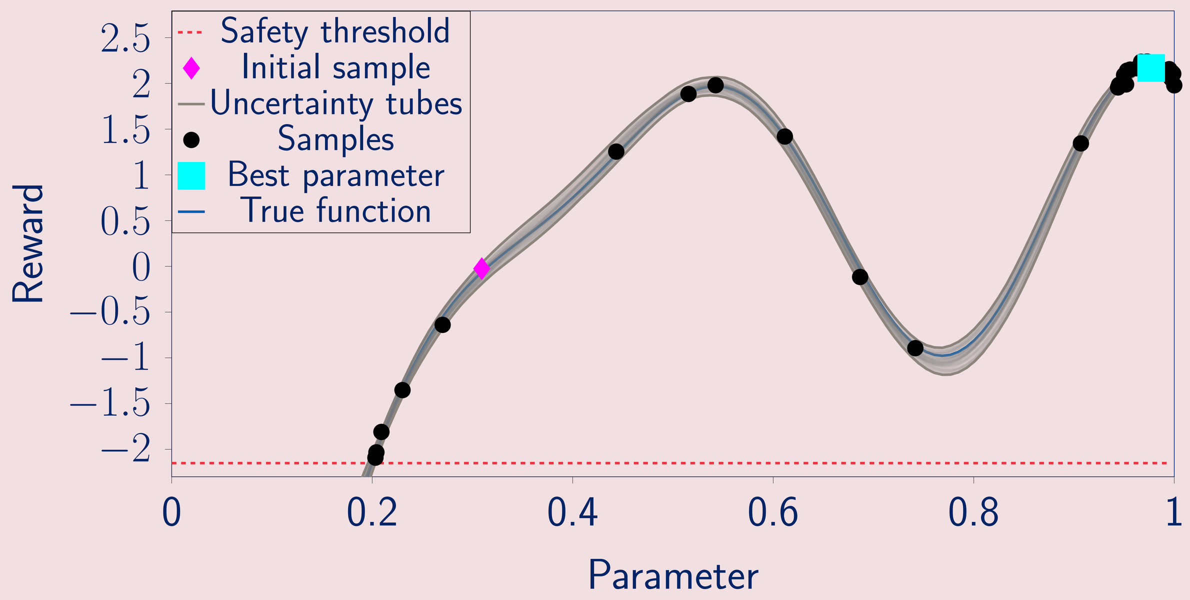
A. Tokmak [1,2] T. Karvonen [3] T. B. Schön [4] D. Baumann [1]

[1]Aalto University [2]Nokia Bell Labs [3]LUT University [4]Uppsala University

Contribution

Uncertainty quantification is essential in **safety-critical** applications such as safe learning-based control. It is typically realized by constructing **uncertainty tubes** around an unknown function, e.g., a reward of given control parameters. We assume that this function is **random** and that we can generate functions that are **i.i.d. copies**, thereby obtaining **high probability uncertainty tubes** [1].





For example, we apply our uncertainty tubes in **safe Bayesian optimization** for **control parameter tuning**, as one instance of safe learning-based control. Starting from the initial safe set, we sequentially gather samples to maximize the unknown ground truth while guaranteeing safety. The blue curve is the ground truth; the light-gray functions are **i.i.d. copies** conditioned on the samples. The bounds **tighten** as more constraints are added (right), and no unsafe experiment is conducted while identifying the best parameter.

Problem definition

- Many uncertainty tubes require a **functional norm**, e.g., a known RKHS norm or assume a **specific stochastic process**, e.g., that the unknown function is a sample from a GP
- **Restrictive:** an RKHS-norm bound is rarely known, and GPs struggle with discontinuities
- **Our assumption:** Conditioned on the samples, we can generate **i.i.d. scenarios** of the unknown function, which yield **uncertainty tubes**
- **Implementation examples:**
  - Sample a prior function and a noise realization, then map them onto the data by a **minimum-norm projection** that **interpolates** the observations. This induces a data-consistent sampling measure.
  - Any source of i.i.d. realizations is admissible, e.g., several episode rollouts in which pedestrians move, producing observed trajectories under different noise.

Sampling-based uncertainty tubes via the scenario approach [2]

- **Generalization:** If the solution of a (convex) problem holds for the sampled scenarios, it generalizes with high probability to another sample from the same probability space (Theorem 1).
- **Uncertainty tubes:** At iteration t, generate m_t i.i.d. scenarios $\rho_t^{(j)}: \mathcal{A} \to \mathbb{R}$, $j \in [m_t]$, of the unknown function h. The tubes ℓ_t, u_t are the tightest bounds that enclose all scenarios uniformly over $\mathcal{A}$, obtained from the convex program

$$\min_{\ell_t, u_t} \mathbf{1}^\top (u_t - \ell_t) \quad \text{s.t.} \quad \rho_t^{(j)}(a) \in [\ell_t(a), u_t(a)] \quad \forall a \in \mathcal{A},\ j \in [m_t]. \tag{1}$$

- **Closed-form solution:** The unique solution is the point-wise envelope of the scenarios:

$$\ell_t(a) = \min_{j \in [m_t]} \rho_t^{(j)}(a), \qquad u_t(a) = \max_{j \in [m_t]} \rho_t^{(j)}(a). \tag{2}$$

- The required number of scenarios m_t depends on the probability and confidence levels

Theorem 1: Uncertainty tubes via the scenario approach

Suppose:

- *The unknown function h is a random function and, conditioned on the samples, the scenarios $\rho_t^{(j)}$, $j \in [m_t]$, are i.i.d. copies of h under $\mathbb{P}$*
- *The uncertainty tubes ℓ_t, u_t are the solution of the scenario program*

Choose any $\nu \in (0,1)$ and any $\kappa \in (0,1)$. If the number of scenarios m_t satisfies

$$\sum_{r=0}^{2|\mathcal{A}|-1} \binom{m_t}{r} \nu^r (1-\nu)^{m_t - r} \leq \kappa, \tag{3}$$

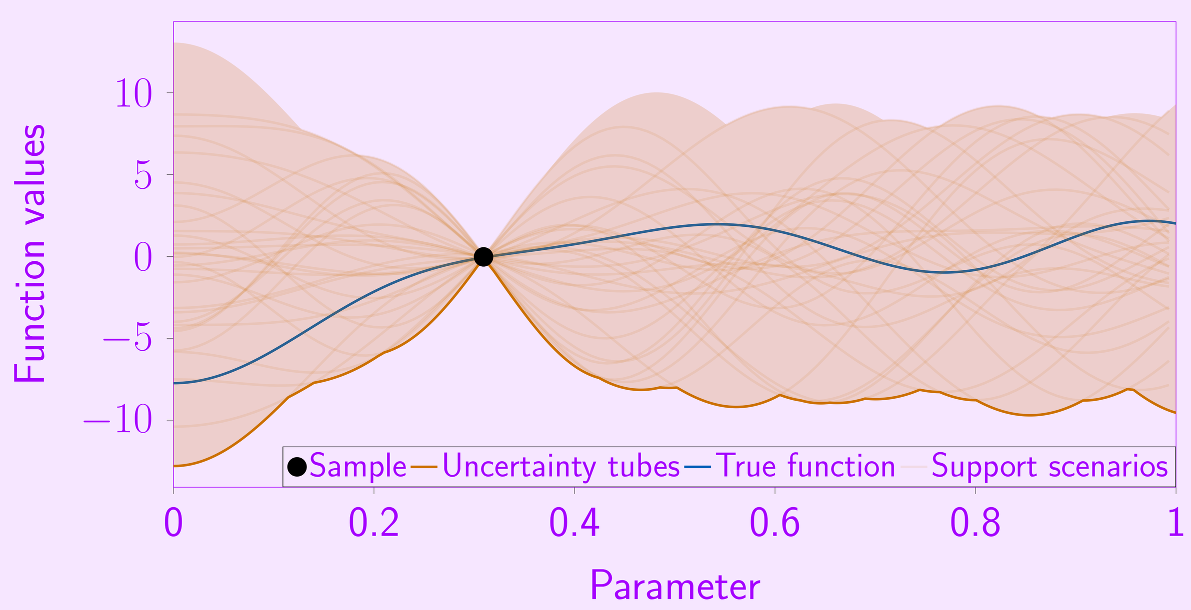
then h is contained within the uncertainty tubes with high probability, i.e.,

$$\mathbb{P}^{m_t}[\mathbb{P}[\exists a \in \mathcal{A}:\ h(a) \notin [\ell_t(a), u_t(a)]] \leq \nu] \geq 1 - \kappa. \tag{4}$$

Proof (idea):

- *The scenario program enforces $\rho_t^{(j)}(a) \in [\ell_t(a), u_t(a)]$ for all sampled scenarios*
- *Since h and $\rho_t^{(jj)}$ are i.i.d., generalization follows from scenario theory*

Synthetic example: Scenario approach



- The unknown function is contained in the uncertainty tubes, but only a few **support scenarios** (whose removal changes the tube) actually determine the tubes.
- Theorem 1 required $m_t = 21403$ scenarios here ($\nu = 10^{-1}$, $\kappa = 10^{-3}$), yet only $s_t = 40$ of them are support scenarios.

Uncertainty tubes via the wait-and-judge scenario approach [3]

- **Idea:** The a priori sample size in Theorem 1 is set by the Helly dimension and can be huge. In wait-and-judge, we first **wait**: solve the scenario program. Then we **judge**: count the number of **support scenarios** s_t and certify the tubes a posteriori.
- **Procedure:** Start from some m_t, sample the scenarios, and compute ℓ_t, u_t. Count s_t (the scenarios whose removal changes the tubes) and solve for $\tau \in (0,1)$ from

$$\frac{\kappa}{m_t + 1} \sum_{r = s_t}^{m_t} \binom{r}{s_t} \tau^{r - s_t} - \binom{m_t}{s_t} \tau^{m_t - s_t} = 0. \tag{5}$$

- **Stopping:** If $\tau \geq 1 - \nu$, the tubes are accurate enough and we stop. Otherwise we increase m_t and repeat. We never exceed the sample size of Theorem 1.

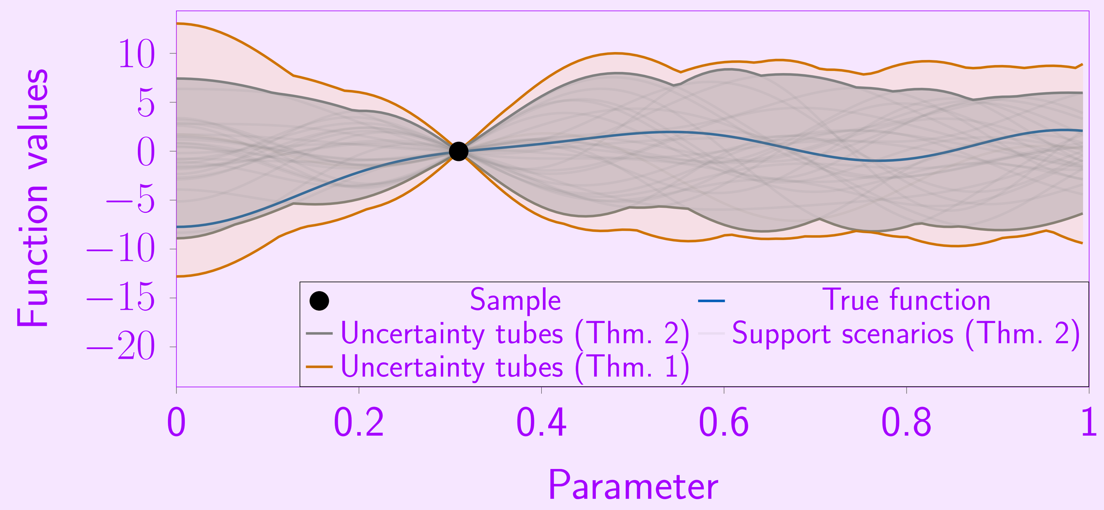
Theorem 2: Uncertainty tubes via wait-and-judge scenario approach

Consider the setting of Theorem 1. If ℓ_t, u_t are computed by wait-and-judge from the observed number of support scenarios s_t with $\tau \geq 1 - \nu$, then

$$\mathbb{P}^{m_t}[\mathbb{P}[\exists a \in \mathcal{A}:\ h(a) \notin [\ell_t(a), u_t(a)]] \leq \nu] \geq 1 - \kappa, \tag{6}$$

i.e., the uncertainty tubes spanned by the support scenarios contain h with high probability.

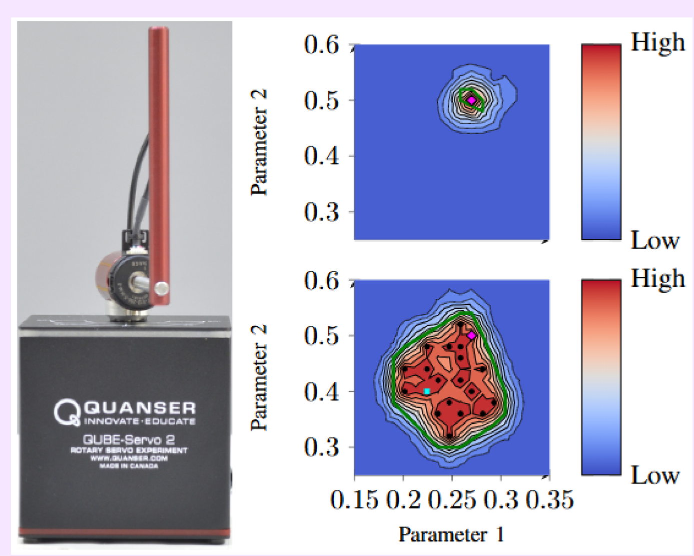
Synthetic example: Wait-and-judge scenario approach



- **Classic scenario (Thm. 1):** $m_t = 21\,403$, $s_t = 40$ scenarios; tubes (orange) are conservative.
- **Wait-and-judge (Thm. 2):** $m_t = 596$, $s_t = 31$ scenarios; tubes (gray) are tighter.

Safe Bayesian optimization for control parameter tuning on hardware

- We integrate the uncertainty tubes into a safe Bayesian optimization algorithm and tune two feedback gains on a **real Furuta pendulum**, starting from an initial safe parameter.
- At every iteration we observe reward, generate scenarios **conditioned on the samples**, compute the tubes via wait-and-judge, and feed them back into safe BO.
- After $T = 20$ episodes we expand the **safe set**, refine the tubes, and return a better gain (cyan) **without a safety violation**.



Conclusions

- We model the unknown function as **random** and assume that **i.i.d. copies** can be sampled (function-based UQ), and construct high-probability **uncertainty tubes wait-and-judge scenario** framework.
- We integrate the tubes into **safe BO** and tune control parameters on **real hardware** without unsafe experiments.

References

[1] A. Tokmak, T. Karvonen, T. B. Schön, and D. Baumann. "Safe learning-based control via function-based uncertainty quantification". In: *IEEE Conference on Decision and Control*. 2026.

[2] M. C. Campi and S. Garatti. *Introduction to the scenario approach*. SIAM, 2018.

[3] M. C. Campi and S. Garatti. "Wait-and-judge scenario optimization". In: *Math. Prog.* (2018).