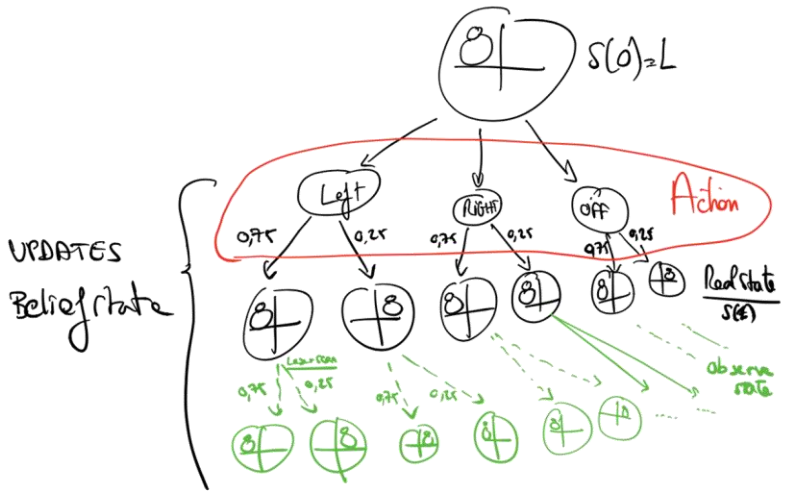
Funds: LOCPU (PID2020-116585GB-I00) & SIECMA (PID2023-146747NB-C21)
MICIU/AEI/10.13039/501100011033 and ERDF/EU

Nonmyopic Bayesian process optimization with a finite budget



José L. Pitarch[1*], Leo Armesto[2], Antonio Sala[1]

[1]*Instituto de Automática e Informática Industrial (ai2), UPV*

[2]*Instituto de Diseño y Fabricación (IDF), UPV*

**jlpitarch@isa.upv.es*

ESCAPE /35
9 JULY 2025

Motivating example

ALHAMBRA ANDALUCÍA LIMITED SERIES

'Azahar', 'Matalahúva', 'Salicornia'

- Go from lab to industrial scale.
  - Recipe (i.e. model) partially known.
- Just N batches are to be produced.
- Goal: Maximize the cummulative benefit

$$\max_{x} \sum_{i}^{N} \mathrm{P_{sell}} \cdot \mathrm{Product}_i(x) - \mathrm{Cost}_i \cdot \mathrm{Resource}_i(x)$$

$$\text{subject to: } \mathrm{Quality}_i\ (x) \geq \eta$$

- Just apply the model optimum?
- Extensive experiments to refine the model?
- Use robust or stochastic optimization?
- Apply reinforcement learning?
- Apply modifier adaptation methods?
- Use Bayesian optimization?

Motivating example

TYPICAL METHODS DO NOT APPROACH THE PROBLEM IN THE RIGHT WAY

- Model optimum won't be the true one: Room for improvement.
- Many identification/learning experiments "eat" the potential benefit.
- Robust/stochastic optimization focus on constraint satisfaction but does not "learn".
- Modifier adaptation drives experimentally to the true optimum, but:
  - Measurement noise hampers gradient estimation and only a few trials are possible (not for free).
  - No guarantees on constraint satisfaction during the trials.
- *Bayesian optimization* looks for probabilistic improvement to drive experiments.
  - Heuristics of particular exploitation-exploration tradeoffs must be chosen in advance.
  - It is myopic, does not consider experimentation costs nor a budget limit.

POMDP approach

Partially-Observable Markov Decision Process (POMDP)

$\gamma \equiv$discount factor

- *Decision process*:
  - Optimize expected cummulative cost: (immediate + terminal costs)

$$J := \mathbb{E}\left\{\sum_{k=1}^{N-1} \gamma^k \, \boldsymbol{r}(\boldsymbol{x_k}) + \gamma^N \boldsymbol{J_N}(\boldsymbol{x_N})\right\}$$

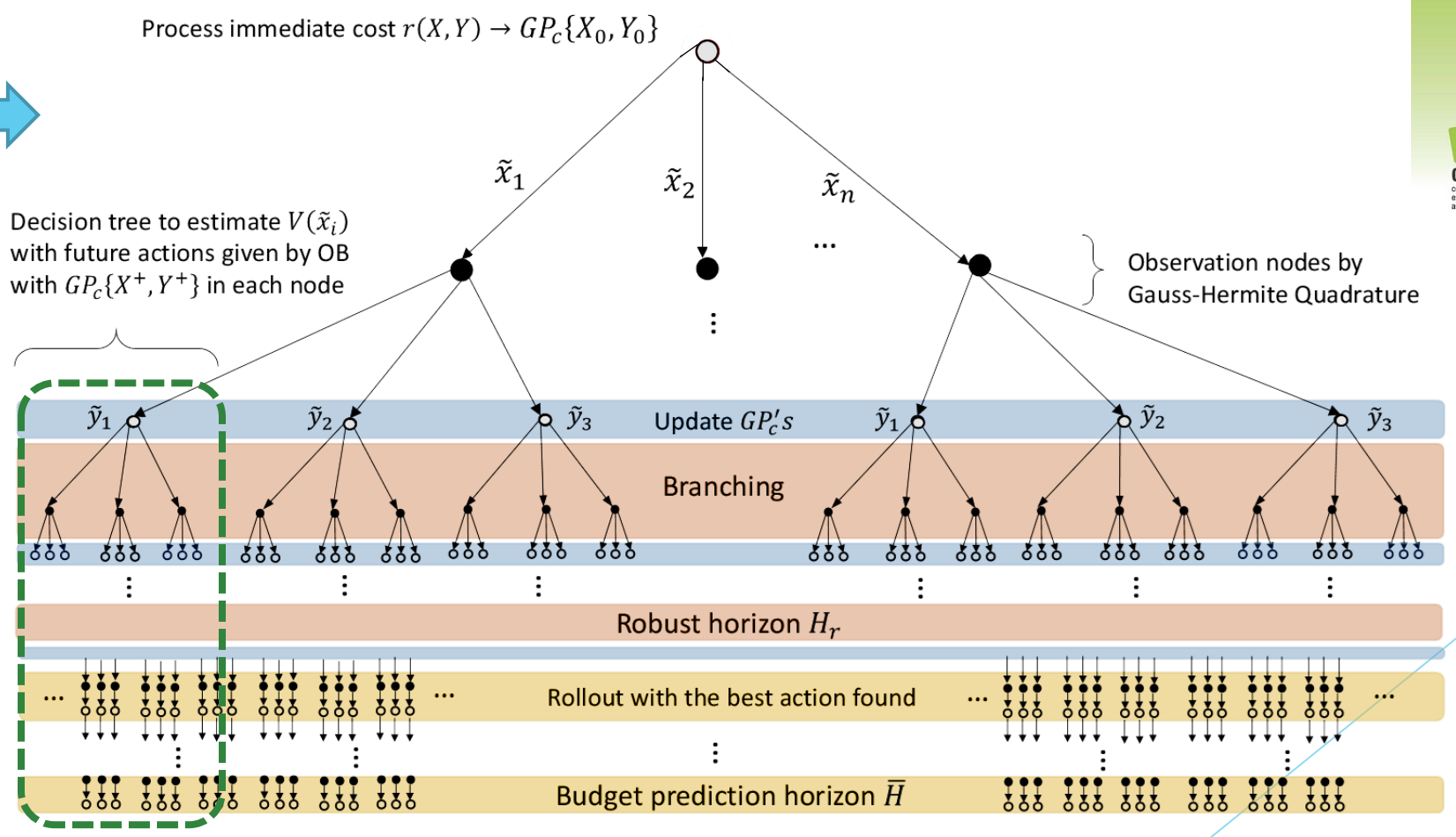
  - Each problem poses a particular exploration-exploitation tradeoff.
- *Partially observable*:
  - Uncertain process "*belief*" available a priori (e.g. first principles + Gaussian process).
  - Each (noisy) observation updates such *belief* on cost and constraints.
- KEY IDEA: Consider and cuantify how worth is "knowing that process belief could be better known along the remaining budget horizon".

RIAI. Volúmen 21. Número 4
Octubre/Diciembre 2024
Revista Iberoamericana de Automática e Informática Industrial

POMDP scenario decision tree

- Infinite action-observation scenarios in continuous systems → INTRACTABLE PROBLEM
- PROPOSAL:

IDEA: Virtually "learn" the first-decision value function



*Evaluate a scenario tree is CPU intensive.

Main reason to apply BO

Proposed POMDP-BO Algorithm

Algorithm 1: Nonmyopic BO acquisition function.

Global: $X_0, Y_0, N, H_r, V_0, \text{optol}$ %Initial data

1: **Create** $GP_c\{X_0, Y_0\}$ %Build root cost GP

2: **For** $k \coloneqq 1$ **to** N **do:**

3: **If** $k = 1$ **then** $\tilde{x} \leftarrow X_0$; **otherwise**

$\tilde{x} \leftarrow \arg\min_{x \in \mathbb{X}} F(GP_V, x)$ %Value-function BO

4: $b = \text{NodeV}(GP_c)$ %Create the root node

5: $\bar{H} \leftarrow N - k$ %Reduce prediction horizon

6: $b.\text{Expand}(\tilde{x})$ %Branch second-stage tree

7: $\hat{V} \leftarrow b.\text{ValueFunction}()$ %Estimate V($\tilde{x}$)

8: $GP_V \leftarrow GP_V \oplus \{\tilde{x}, \hat{V}\}$ %Update GP of V(x)

9: **If** $|V_0 - \hat{V}| > \text{optol}$ **then** $V_0 \leftarrow \min_{\hat{V}} GP_V$; **Goto 3**

10: **Realize** $x \leftarrow \arg\min_{\tilde{x}} \min_{\hat{V}} GP_V$ %Actual experiment

11: **Get** $y_f \leftarrow f(x) + w$ %New observation

12: $GP_c \leftarrow GP_c \oplus \{x, y_f\}$; %Update GP of r(X,Y)

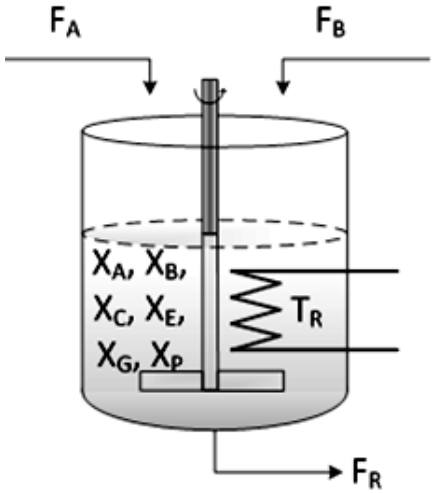
Benchmark results

TEST 1: Average performance in 100 sinthetic randomly generated processes.

Accumulated regret

Method:	Lam-17	Pitarch-25	Pitarch-24	BO-PI	BO-EV
Regret:	7.92	7.77	8.28	9.93	9.73
CPU time:	22.3s	13s	0.68s	-	-

TEST 2: Otto-Williams CSTR



$$\frac{dX_A}{dt} := F_A - (F_A + F_B)\, X_A - (r_1 + r_2)\, V_R$$

$$\frac{dX_B}{dt} := F_B - (F_A + F_B)\, X_B - (2\, r_1 + r_2)\, V_R$$

$$\frac{dX_E}{dt} := -(F_A + F_B)\, X_E + 2\, r_1 V_R$$

$$\frac{dX_G}{dt} := -(F_A + F_B)\, X_G + 3\, r_2 V_R$$

$$\frac{dX_P}{dt} := -(F_A + F_B)\, X_E + (r_1 - r_2)\, V_R$$

*Plant-model mismatch: agent is not aware of presence of product C

$$\underset{F_b, T_r}{\text{maximize}} \sum_{k=1}^{10} (X_{p,kPp} + X_{e,kPe})\, F_{r,k} - F_a C_a - F_{b,k} C_b$$

subject to:

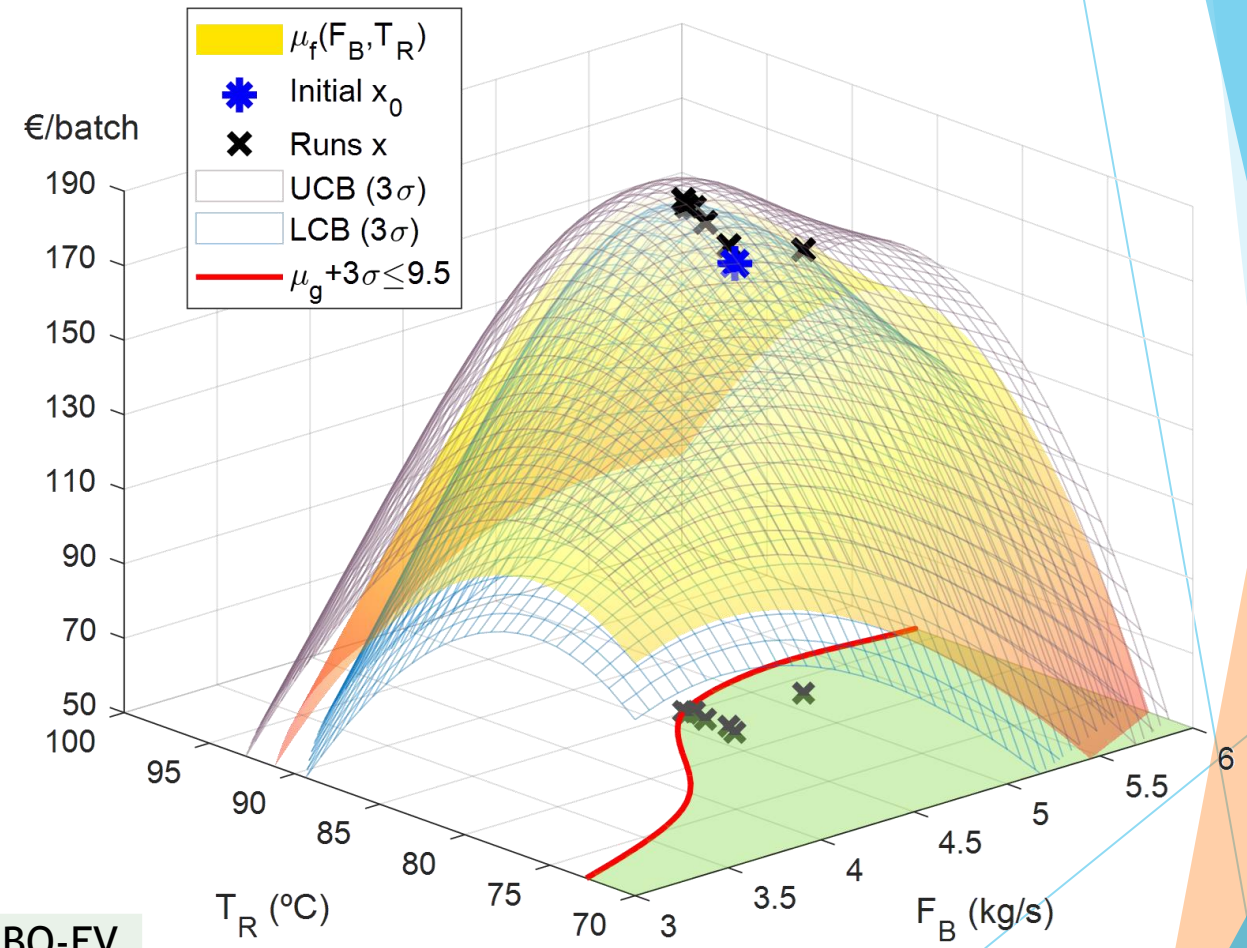
$$X_g \le 9.5\%$$

$$3 \le F_b \le 6 \text{ kg/s}$$

$$70 \le T_r \le 100\ ^\circ\text{C}$$

Benchmark results

CSTR performance belief, i.e. process knowledge, after producing 10 batches



Performance comparison against myopic BO:

Method:	Pitarch-25	BO-EI	BO-PI	BO-UCB	BO-EV
Regret(€):	**44.37**	56.64	74.7	48.38	76.45

Take-home remarks

Advantages

- Nonmyopic approaches unlock extra benefits in problems of POMDP nature.
- Efficient myopic BO is useful to evaluate rollout POMDP trees.
- In the test with the chemical reactor, our proposal performs better than standard BO.

Drawbacks

- Improvent are given in terms of expectation.
- The problem is still of exponential complexity.
- A single test with a particular plant is not statistically conclusive. However “one-chance run” is often the situation in reality…

- Performance is mainly driven by ability of the probabilistic belief to represent the actual plant-model mismatch.