

Cost-aware Bayesian Optimization via the Pandora's Box Gittins Index

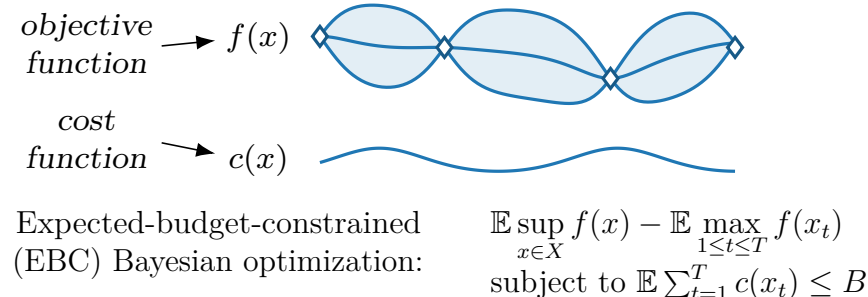
Qian Xie,¹ Raul Astudillo,² Peter Frazier,¹ Ziv Scully,¹ and Alexander Terenin¹
¹Cornell University ²Caltech



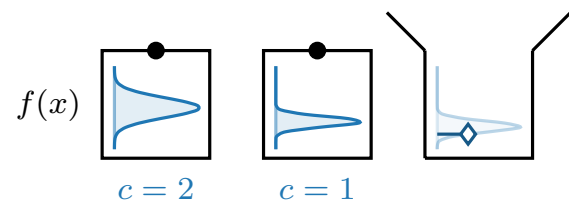
Abstract

Bayesian optimization is a technique for efficient global optimization of black-box unknown functions. In many practical settings, it is desirable to explicitly incorporate function evaluation costs into acquisition functions used for Bayesian optimization. To do so, we develop a connection between cost-aware Bayesian optimization and *Pandora's Box*, a decision problem from economics. The Pandora's Box problem admits a Bayesian-optimal solution based on an expression called the *Gittins index*, which can be reinterpreted as an acquisition function. We demonstrate empirically that this acquisition function performs well on cost-aware Bayesian optimization, particularly in medium-high dimensions. We further show that this performance carries over to classical Bayesian optimization without explicit evaluation costs. Our work constitutes a first step towards integrating techniques from Gittins index theory into Bayesian optimization.

Cost-aware Bayesian Optimization



Pandora's Box



Cost-per-sample (CPS) objective: $\mathbb{E} \max_{1 \leq t \leq T} f(x_t) - \mathbb{E} \sum_{t=1}^T c(x_t)$

Optimal policy (notation: $\text{EI}_\psi(x; y) = \mathbb{E} \max(0, \psi(x) - y)$):
 $\alpha^*(x) = g$ where g solves $\text{EI}_f(x; g) = c(x)$

Our work: EBC and CPS problems are equivalent
 (extends prior work on generalized Pandora's boxes to continuous rewards)

Key difference from Bayesian optimization: no correlations

Pandora's Box Gittins Index: a new acquisition function

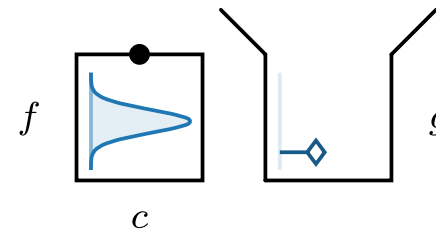
$$\alpha_t^{\text{PBGI}}(x) = g \quad \text{where } g \text{ solves} \quad \text{EI}_{f|x_{1:t}, y_{1:t}}(x; g) = \lambda c(x)$$

Idea: extend α^* by plugging posterior in for f
 λ : cost scaling factor from budget-constraint Lagrangian duality
 Computation: one-dimensional convex optimization

Where does α_t^{PBGI} come from?

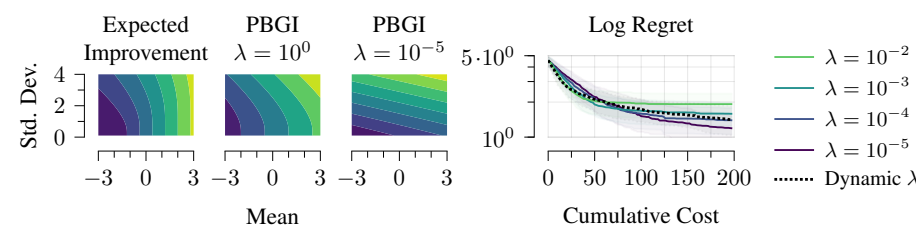
Simplified problem: one closed and one open box

Decision Value
 Open box $\mathbb{E} \max(f, g) - c$
 Don't open g



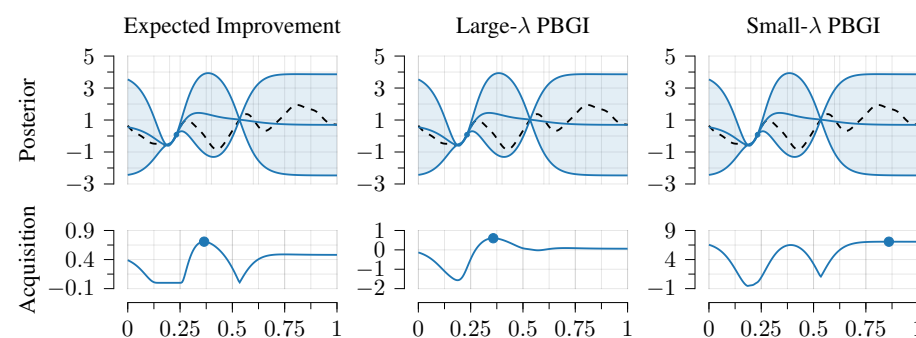
Should one open the closed box? Depends on the observed value g !
 If *both* opening and not opening are optimal: g is a *fair value*
 α_t^{PBGI} : pick points according to their fair values

Behavior and Comparisons

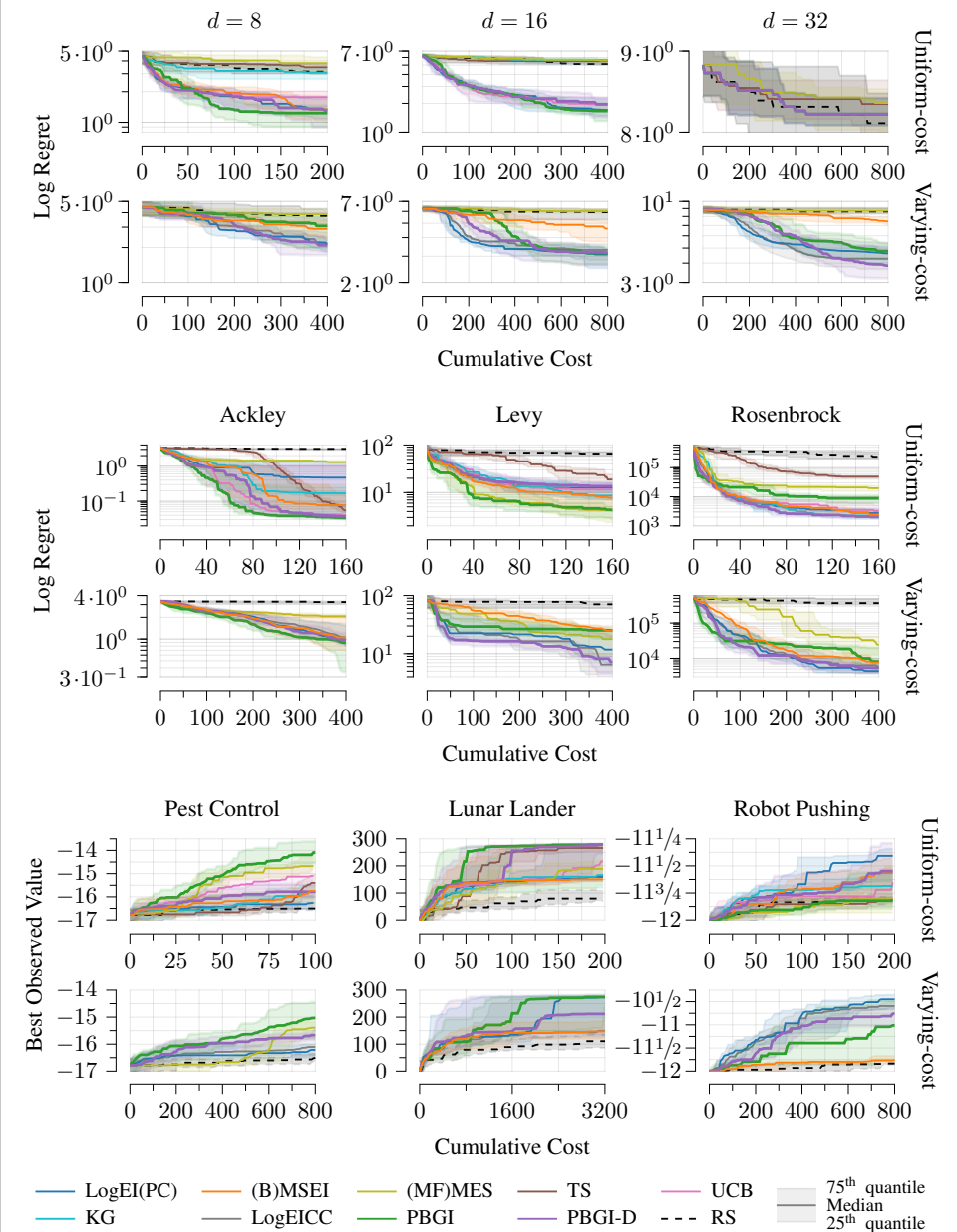


Large λ : similar to α_t^{EI}

Small λ : similar to α_t^{UCB}



Performance



Computation Time

