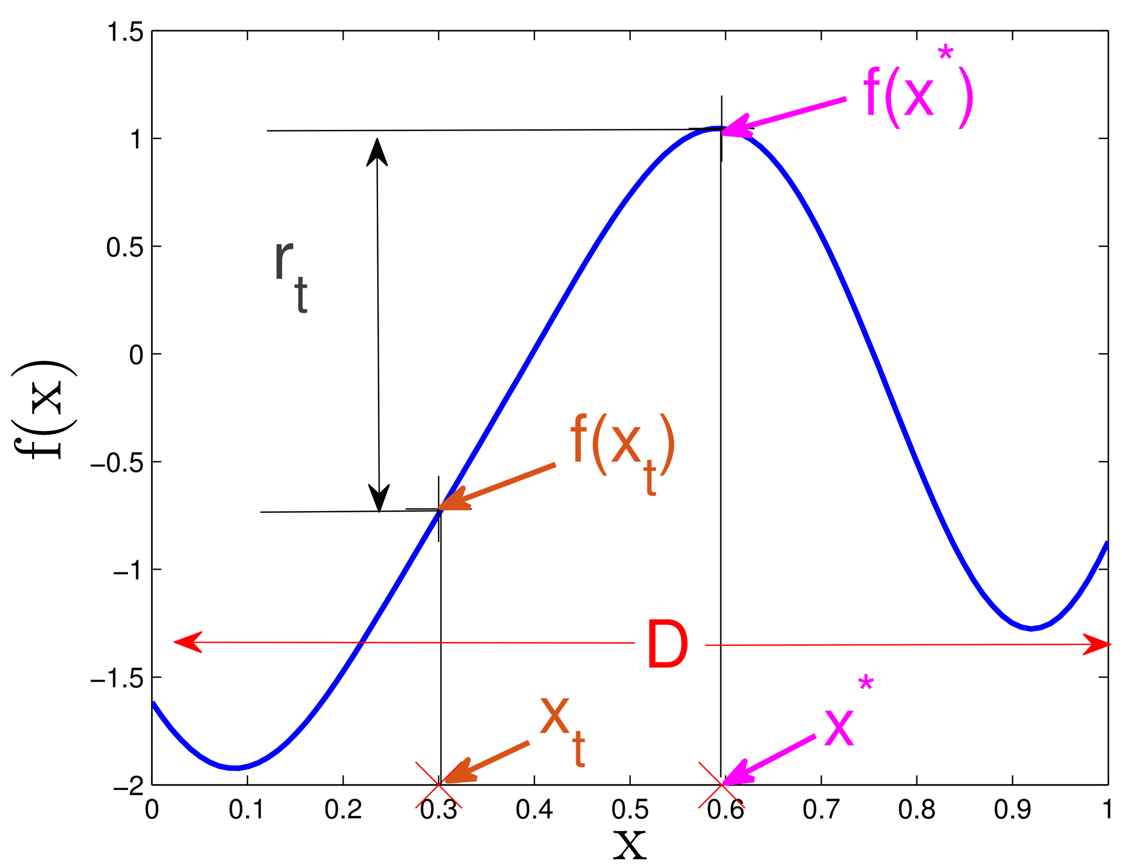
On Batch Bayesian Optimization

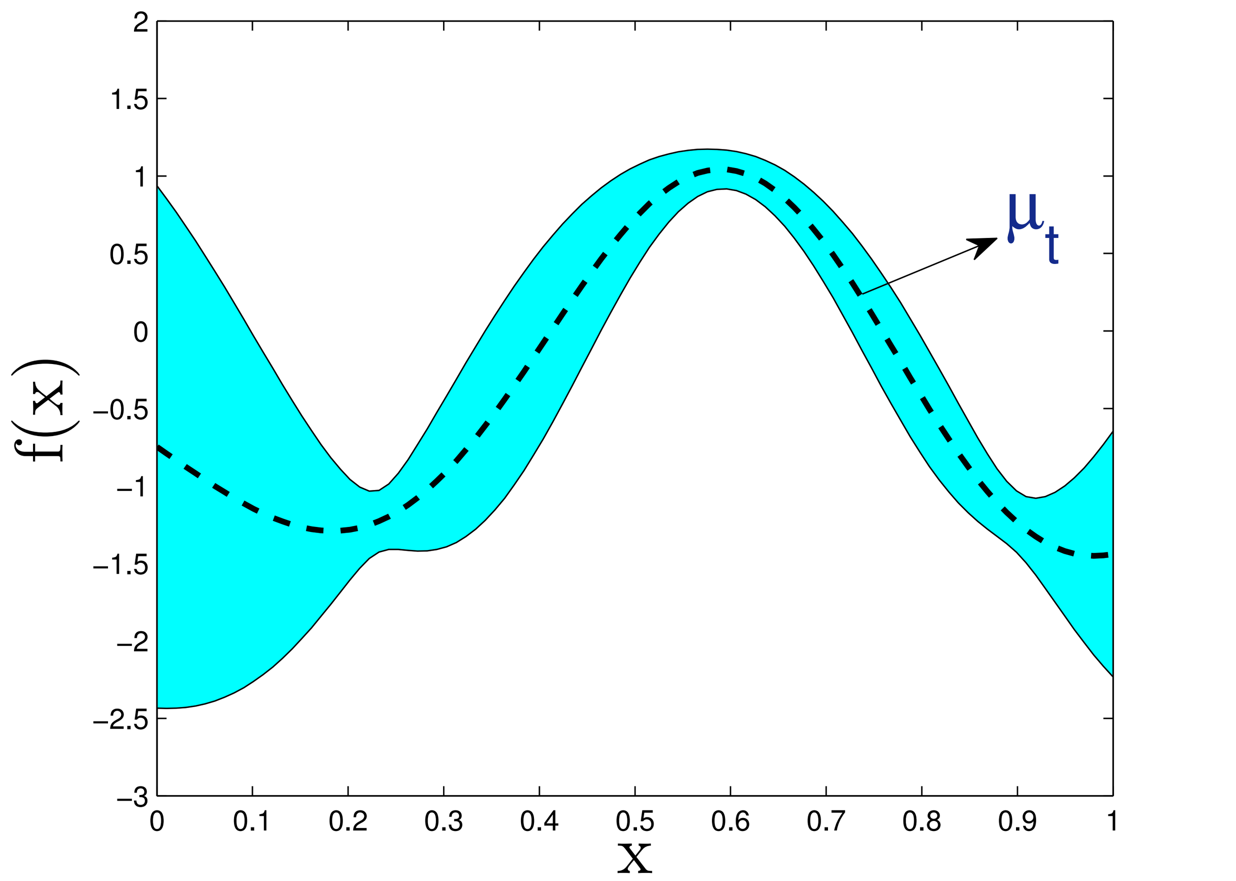
Sayak Ray Chowdhury   Aditya Gopalan

Department of Electrical Communication Engineering, Indian Institute of Science

Bayesian Optimization

Sequentially maximize an *unknown* function $f : D \to \mathbb{R}$, $D \subset \mathbb{R}^d$





At every round t, an agent:

1. Chooses $x_t \in D$ based on past observations
2. Observes noisy samples of $f(x_t)$
3. Suffers regret $r_t = f(x^\star) - f(x_t)$

Goal: Minimize cumulative regret

- Represent uncertainty over f using **G**aussian **p**rocess prior $GP(0, k(x,y))$
- Squared Exponential (SE) kernel: $k(x,y) = \exp\left(\frac{-\|x-y\|_2^2}{2l^2}\right)$
- Posterior of f: $GP(\mu_{t-1}(x), k_{t-1}(x,y))$

Application and key idea: Hyperparameter tuning in DeepNN – huge set of parameters to tune – *number of layers, weight regularization, layer size, nonlinearity type, batch size, learning rate schedule, stopping conditions etc* – grid search is expensive – *optimize a cheap proxy function instead !*

Batch Bayesian Optimization

Feedback Structure

- Rewards are delayed or available in batches
- $\mathcal{S}(t) \le t-1$ is the most recent round upto which rewards are available
- x_t is chosen using rewards obtained till round $\mathcal{S}(t)$
- Assume $t - \mathcal{S}(t) \le M$, $M \ge 1$ known

Regularity Assumptions

- f lies in **RKHS** of functions: $D \to \mathbb{R}$
- Reproducing property: $f(x) = \langle f, k(x,\cdot)\rangle_k$
- Induces smoothness: $|f(x) - f(y)| \le \|f\|_k \|k(x,\cdot) - k(y,\cdot)\|_k$
- $\|f\|_k \le B$ (known)
- Zero mean additive sub-Gaussian noise

Application

- Parallelizing an expensive computer simulation over multiple cores
- *Simple batch* setting: $\mathcal{S}(t) = M\lfloor (t-1)/M \rfloor$
- *Simple delay* setting: $\mathcal{S}(t) = \max\{t-M, 0\}$
- *Strictly sequential* setting: $M = 1$ or $\mathcal{S}(t) = t-1$

Posterior GP with Batch Feedback

- Available rewards at round t: $y_1, \ldots, y_{\mathcal{S}(t)}$
- *Hallucinate* missing rewards $y_{\mathcal{S}(t)+1}, \ldots, y_{t-1}$ using the most recently updated posterior mean $\mu_{\mathcal{S}(t)}$
- Set $y_s = \mu_{\mathcal{S}(t)}(x_s)$, $s = \mathcal{S}(t)+1, \mathcal{S}(t)+2, \ldots, t-1$
- Posterior mean remains $\mu_{\mathcal{S}(t)}$, but the posterior covariance decreases to k_{t-1}

Algorithms: IGP-BUCB and GP-BTS

IGP-BUCB: At each round t, choose $x_t = \text{argmax}_{x \in D}\, \mu_{\mathcal{S}(t)}(x) + \beta_t \sigma_{t-1}(x)$

GP-BTS: Sample $f_t \sim GP(\mu_{\mathcal{S}(t)}, v_t^2 k_{t-1})$ and choose $x_t = \text{argmax}_{x \in D_t} f_t(x)$

- β_t and v_t: (a) Governs exploration and exploitation trade-off, (b) Compensates for the bias created by the hallucinated data in the attempt to aggressively shrink the confidence interval and reduce exploration
- **IGP-BUCB:** Reduced width (β_t) of confidence interval compared to **GP-BUCB** (*Desautels et al., JMLR 2014*) – Improved theoretical and numerical performance
- **GP-BTS:** D_t is a suitable discretization of D
- *Strictly sequential* setting: Recover **IGP-UCB** and **GP-TS** algorithms (*Chowdhury et al., ICML 2017*)

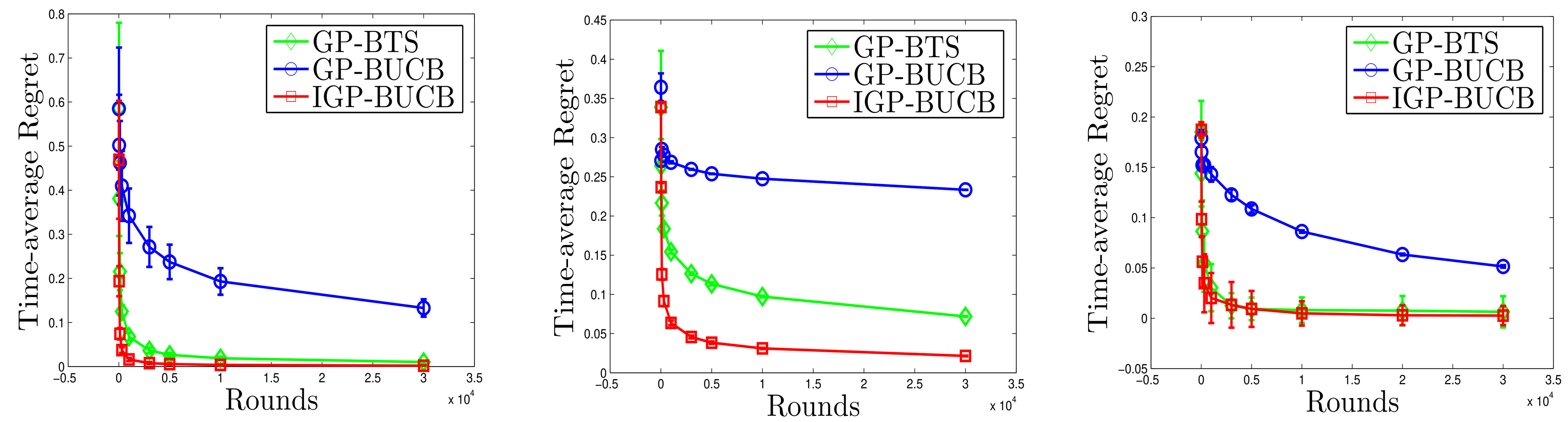
Regret Bounds

IGP-BUCB: $O\left(\sqrt{\xi_M T}\left(B\sqrt{\gamma_T} + \gamma_T\right)\right)$ with high probability

GP-BTS: $O\left(\sqrt{\xi_M T d \ln(BdT)}\left(B\sqrt{\gamma_T} + \gamma_T\right)\right)$ with high probability

- γ_T: *Maximum Information Gain* about f after T rounds – quantifies reduction in uncertainty
- Squared-exponential kernel: $\gamma_T \asymp O(\ln T)$ – Cumulative regret grows **sublinearly** with T
- ξ_M: Bounds information gain about f from at most M hallucinations given actual rewards
- Squared-exponential kernel: $\xi_M \asymp O(M)$ – Cumulative regret grow linearly with the batch size M
- *Uncertainty sampling* based initialization scheme makes ξ_M constant – *average per-round regret vanishes*

Numerical Results



(a)   (b)   (c)

Figure 1: Time-average regret for (a) RKHS functions of SE kernel, (b) Rosenbrock function, (c) Temperature sensor data.

References

[1] *Parallelizing exploration-exploitation tradeoffs in gaussian process bandit optimization, Thomas Desautels, Andreas Krause, and Joel W Burdick, JMLR 2014.*

[2] *On Kernelized Multi-armed Bandits, S. R. Chowdhury and A. Gopalan, ICML 2017.*

Acknowledgment. Google India PhD fellowship grant in Machine Learning, 2017