# Batch Multi-Fidelity Bayesian Optimization with Deep Auto-regressive Networks

Shibo Li, Robert Mike Kirby, Shandian Zhe
School of Computing, University of Utah
{shibo, kirby, zhe}@cs.utah.edu

## Abstract

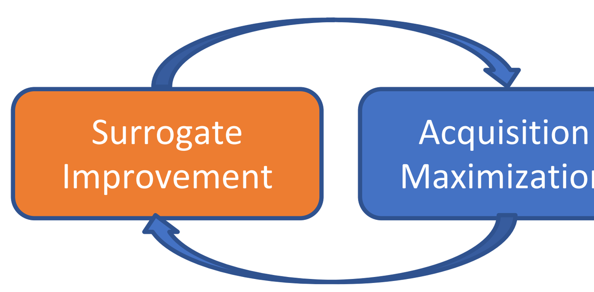
Bayesian optimization (BO) is a powerful approach for optimizing black-box, expensive-to-evaluate functions. To enable a flexible trade-off between the cost and accuracy, many applications allow the function to be evaluated at different fidelities. In order to reduce the optimization cost while maximizing the benefit-cost ratio, in this paper we propose Batch Multi-fidelity Bayesian Optimization with Deep Auto-Regressive Networks (BMBO-DARN). We use a set of Bayesian neural networks to construct a fully auto-regressive model, which is expressive enough to capture strong yet complex relationships across all the fidelities, so as to improve the surrogate learning and optimization performance. Furthermore, to enhance the quality and diversity of queries, we develop a simple yet efficient batch querying method, without any combinatorial search over the fidelities. We propose a batch acquisition function based on Max-value Entropy Search (MES) principle, which penalizes highly correlated queries and encourages diversity. We use posterior samples and moment matching to fulfill efficient computation of the acquisition function and conduct alternating optimization over every fidelity-input pair, which guarantees an improvement at each step. We demonstrate the advantage of our approach on four real-world hyperparameter optimization applications.

## Introduction

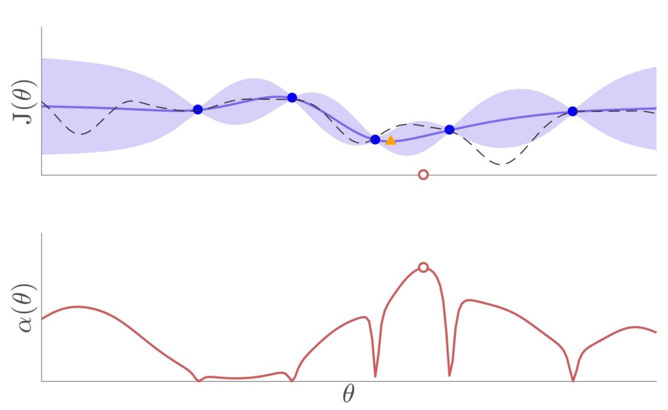
**Bayesian Optimization(BO):** Black-box optimization without access of gradient information.

- Probabilistic surrogate: distribution of objective functions
- Acquisition function: exploration-exploitation trade-off
- Real applications: AutoML, Adaptive Control, Molecule Designs

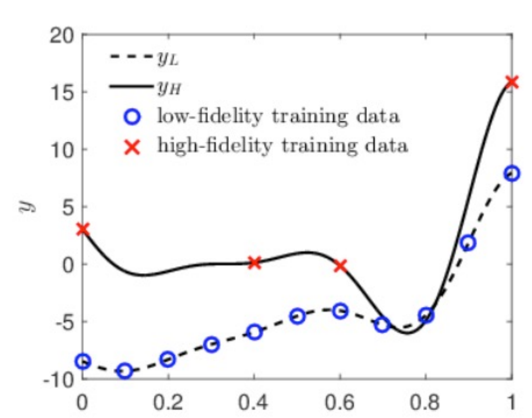
$p(\boldsymbol{\theta}_{t+1}|\mathcal{D}_t \cup \{\mathbf{x}_{t+1}, y_{t+1}\})$




$a(\mathbf{x}) = \mathbb{I}(f(\mathbf{x}), f^*|\mathcal{D})$
$x_{t+1} = \mathrm{argmax}\ \ a_t(x)$




(Neumann-Brosig et al., 2018)




(Xuhui Meng et al. 2019)

**Multi-Fidelity Bayesian Optimization (MFBO):** Objective function can be evaluated at different fidelities:

- Low-fidelity query: cheap but inaccurate
- High-fidelity query: accurate but expensive
- Goal: how to best balance benefit and cost?

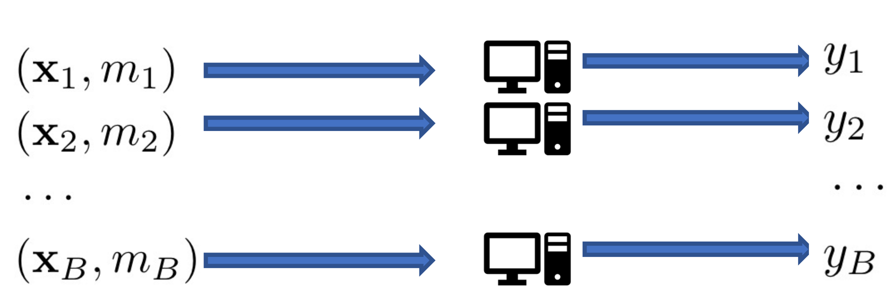
$$m_{t+1}, x_{t+1} = \mathrm{argmax}\ \ a_t(m, x)$$

**MFBO Challenges:**

- Model the correlations within fidelity levels
- Tractable computation of multi-fidelity version of acquisition

**Why Batch?**

- Parallel oracles




- Avoid Collapsed Samples

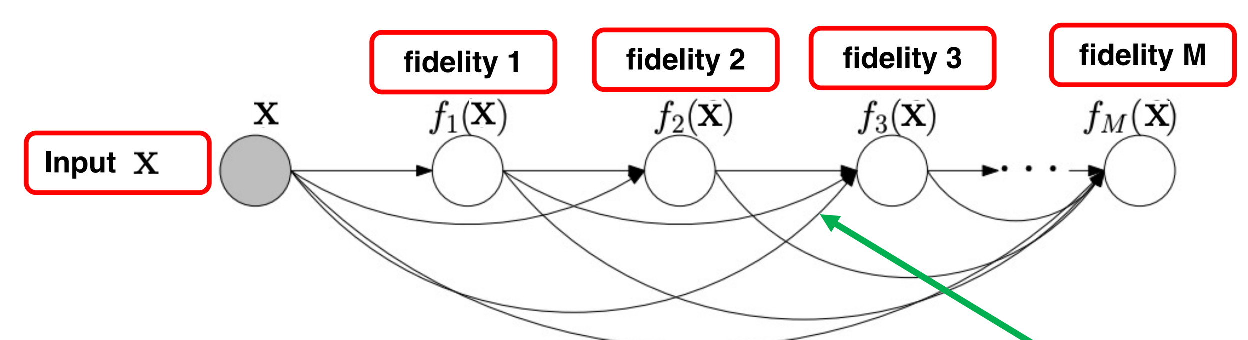

Sequential strategy | Batch strategy

**Our Contribution:**

- **A deep auto-regressive model**: Integrating training examples at *various fidelities*. Flexibly capture the complex correlations
- **Batch acquisition function**: Based on state-of-the-art MES
- **Efficient computation of batch acquisition** : Avoid combinatorial search Guarantees the improvement at each step

## Methods

**Surrogate Modeling: Full Auto Regressive Network**



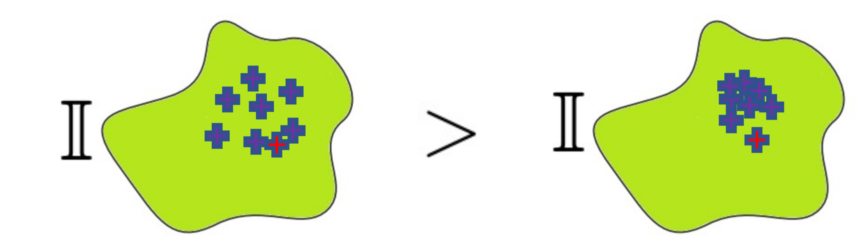

$$\begin{cases} \mathbf{x}_m = [\mathbf{x}; f_1(\mathbf{x}); \ldots; f_{m-1}(\mathbf{x})] \\ f_m(\mathbf{x}) = \psi_{\mathcal{W}_m}(\mathbf{x}_m, f_1(\mathbf{x}), \ldots, f_{m-1}(\mathbf{x})) \\ y_m(\mathbf{x}) = f_m(\mathbf{x}) + \epsilon_m \\ \mathcal{D} = \{\{(\mathbf{x}_{nm}, y_{nm})\}_{n=1}^{N_m}\}_{m=1}^{M} \end{cases}$$

$$p(\mathcal{W}, \tau, \mathcal{Y}, \mathcal{S}|\mathcal{X}) = \underbrace{\mathcal{N}(\mathrm{vec}(\mathcal{W})|\mathbf{0}, \mathbf{I})}_{\text{NN weights prior}} \underbrace{\prod_{m=1}^{M} \mathrm{Gam}(\tau_m|a_0, b_0)}_{\text{Noise prior}} \underbrace{\prod_{m=1}^{M}\prod_{n=1}^{N_m} \mathcal{N}\left(y_{nm}|f_m(\mathbf{x}_{nm}), \tau_m^{-1}\right)}_{\text{Likelihood of observations}}$$

**Batch Acquisition: Jointly identify** $(\mathbf{x}_1, \mathbf{m}_1), \ldots, (\mathbf{x}_B, \mathbf{m}_B)$

$$a_{\text{batch}}(\mathbf{X}, \mathbf{m}) = \frac{\mathbb{I}(\{f_{m_1}(\mathbf{x}_1), \ldots, f_{m_B}(\mathbf{x}_B)\}, f^*|\mathcal{D})}{\sum_{k=1}^{B} \lambda_{m_k}}$$

***Explicit*** penalization of fidelities costs



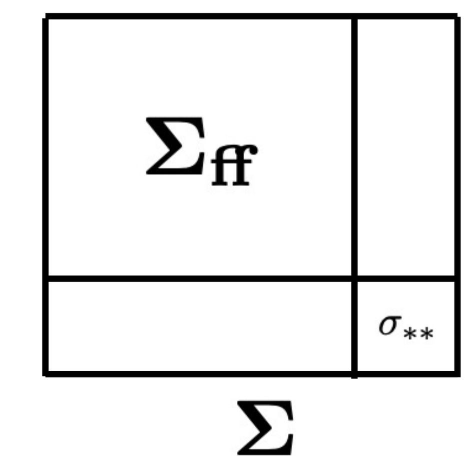

***Implicit*** penalization of highly correlated queries

**Efficient Computation**

Posterior samples $\{(\widehat{\mathbf{f}}_1, \widehat{f}_1^*), \ldots, (\widehat{\mathbf{f}}_L, \widehat{f}_L^*)\}$

First and Second moments

$$\boldsymbol{\mu} = \frac{1}{L}\sum_{j=1}^{L} \widehat{\mathbf{h}}_j, \quad \boldsymbol{\Sigma} = \frac{1}{L-1}\sum_{j=1}^{L} (\widehat{\mathbf{h}}_j - \boldsymbol{\mu})(\widehat{\mathbf{h}}_j - \boldsymbol{\mu})^\top$$




Approximate batch acquisition

$$a_{\text{batch}}(\mathbf{X}, \mathbf{m}) \approx \frac{1}{2\sum_{k=1}^{B} \lambda_{m_k}} \left(\log|\boldsymbol{\Sigma}_{\mathbf{ff}}| + \log \sigma_{**} - \log|\boldsymbol{\Sigma}|\right)$$

**Alternating Optimization**

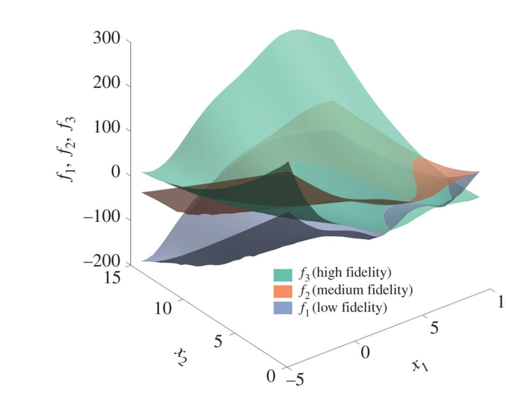
$$\mathcal{Q} = \{(\mathbf{x}_1, m_1), (\mathbf{x}_2, m_2), \cdots, (\mathbf{x}_j, m_j), \cdots (\mathbf{x}_B, m_B)\}$$

$$a_{\text{batch},k}(\mathbf{x}, m) = \frac{\mathbb{I}(\mathcal{F}_{\neg k} \cup \{f_m(\mathbf{x})\}, f^*|\mathcal{D})}{\lambda_m + \sum_{j \neq k} \lambda_{m_j}}$$

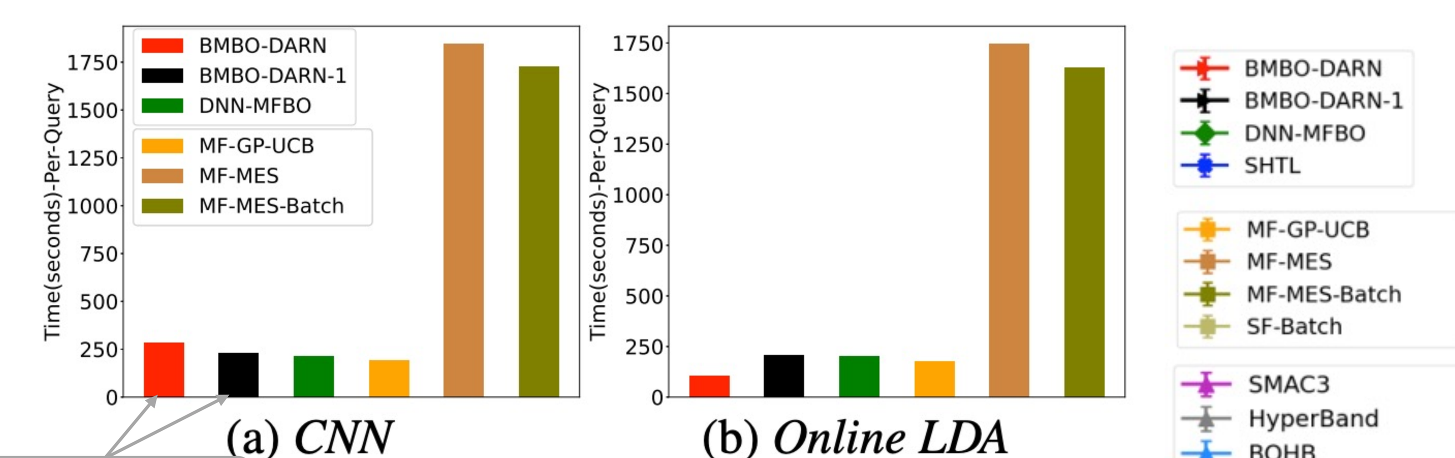
**Improvement Guaranteed!**

## Experiment

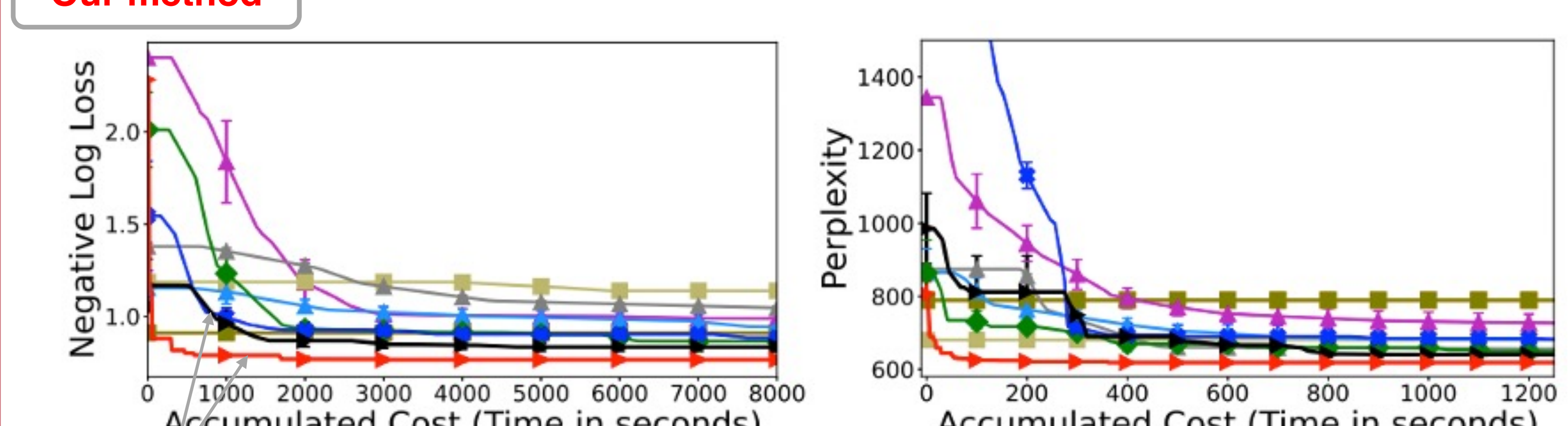
| | *Levy* | nRMSE | MNLL |
|---|---|---|---|
| Independent GPs | MF-GP-UCB | $0.831 \pm 0.195$ | $1.824 \pm 0.276$ |
| Linear Correlated GPs | MF-MES | $0.581 \pm 0.032$ | $1.401 \pm 0.031$ |
| Shared base ABLR | SHTL | $0.443 \pm 0.009$ | $1.208 \pm 0.026$ |
| Seq auto-regressive NN | DNN-MFBO | $0.365 \pm 0.035$ | $1.081 \pm 0.011$ |
| Full auto-regressive NN(our model) | BMBO-DARN | $\mathbf{0.348 \pm 0.021}$ | $\mathbf{1.072 \pm 0.016}$ |
| | *Branin* | | |
| | MF-GP-UCB | $0.846 \pm 0.147$ | $1.976 \pm 0.208$ |
| | MF-MES | $0.719 \pm 0.099$ | $1.796 \pm 0.128$ |
| | SHTL | $0.835 \pm 0.218$ | $1.958 \pm 0.646$ |
| | DNN-MFBO | $0.182 \pm 0.022$ | $0.973 \pm 0.013$ |
| | BMBO-DARN | $\mathbf{0.158 \pm 0.016}$ | $\mathbf{0.965 \pm 0.005}$ |



MF Branin Function by P. Perdikaris et al 2019




(a) *CNN* (b) *Online LDA*




(a) *CNN* (b) *Online LDA*