Federated Bayesian Optimization

Dinesh Krishnamoorthy

Consider the optimization problem

$$\min_{x \in \mathcal{X}} \sum_{i=1}^{N} f_i(x) \tag{1}$$

where $f_i(\cdot)$ is unknown, but we can observe $f_i(\cdot)$.

Federated Bayesian Optimization: *Solve (1) using Bayesian optimization across N agents in a distributed manner.*

Problem (1) can equivalently be written as a consensus problem

$$\min_{x_i \in \mathcal{X}} \sum_{i=1}^{N} f_i(x_i), \quad \text{s.t. } x_i = x_0, \quad \forall i = 1, \dots, N \tag{2}$$

and this is solved by N subsystems in a distributed manner. Each subsystem models $f_i(x_i)$ as a Gaussian process

$$f_i(x_i) \sim \mathcal{GP}\left(\mu(x_i), k(x_i, x_i')\right)$$

and updates the posterior locally every time subsystem i queries x_i.

Typically, the next query point x_i in each subsystem is given by minimizing an acquisition function

$$x_i^+ = \arg\min_{x_i} \alpha(x_i) \tag{3}$$

where $\alpha(x_i)$ is formulated using the updated posterior function. However in a distributed setting, the acquisition function (3) must be modified to take into account the consensus constraint from (2).

Using the ADMM framework, each subproblem can be formulated as:

$$x_i^{k+1} = \arg\min_{x_i \in \mathcal{X}} f_i(x_i) + \lambda_i^{k\mathsf{T}}(x_i - \bar{x}^k) + \frac{\rho}{2}\|x_i - \bar{x}^k\|_2^2, \qquad \forall i = 1, \dots, N \tag{4}$$

$$\lambda_i^{k+1} = \lambda_i^k + \rho(x_i^{k+1} - \bar{x}^{k+1}), \qquad \forall i = 1, \dots, N \tag{5}$$

where $\bar{x}^{k+1} = 1/N \sum_{i=1}^{N} x_i^{k+1}$. Looking at (4), the next query point can be computed using

$$x_i^+ = \arg\min \alpha(x_i) + \underbrace{\lambda_i^{k\mathsf{T}}(x_i - \bar{x}^k)^{k+1}\frac{\rho}{2}\|x_i - \bar{x}^k\|_2^2}_{\delta(x, \lambda_i^k, \bar{x}^k)}$$

Hence the ***federated acquisition function*** can be in general formulated as

$$x_i^+ = \arg\min_{x_i} \alpha(x_i) + \delta(x_i, \lambda_i^k, \bar{x}^k) \tag{6}$$

This can be used with typical acquisition functions such as ϵ-greedy, PI, EI, GP-UCB, Thompson sampling etc.

Similarly, in resource allocation/optimal exchange problems of the form,

$$\min_{x_1, \dots, x_N} \quad \sum_{i=1}^{N} f_i(x_i) \quad s.t. \quad \sum_{i=1}^{N} x_i = 0 \tag{7}$$

the acquisition function can be modified by adding $\delta(x, \lambda^k, \bar{x}^k) := \lambda^k x_i + 0.5\rho\|x_i - \bar{x}_{-i}^k\|^2$ to the acquisition function, and central coordinator updates the shadow price λ^k which ensures coordination among the different subsystems.

The proposed decomposition-coordination based Bayesian optimization framework is schematically represented in Fig. 3.

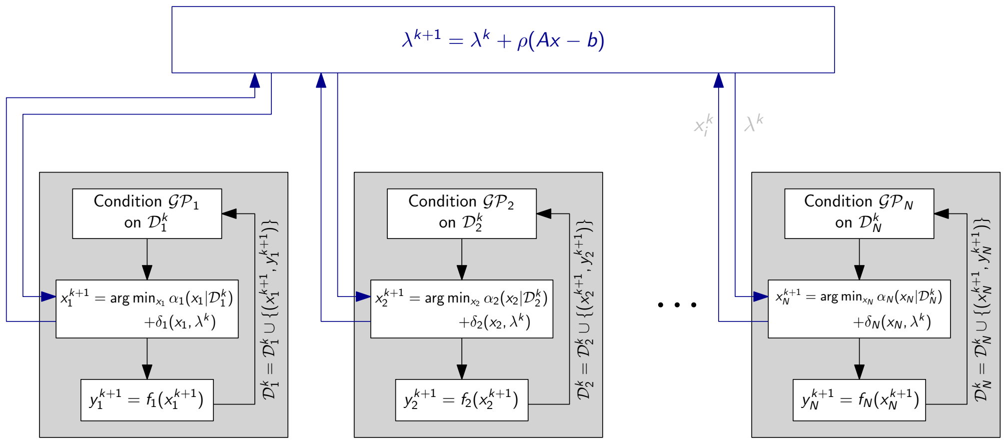

Figure 1: Schematic representation of the decomposition-coordination based federated Bayesian optimization framework.

Preliminary results

The proposed approach is demonstrated on a simple toy example. The code is developed in python and can be found in the Github repo: `https://github.com/dinesh-krishnamoorthy/Federated-BayesOpt`

The two local objectives are as follows

$$f_1(x_1) = x_1^4 + x_1^3 - 2x_1^2 - 2x_1 \tag{8}$$

$$f_2(x_2) = 0.5x_2^2 \tag{9}$$

subject to the consensus constraint $x_1 = x_2$. The simulation results are shown in Fig. 2. The problem with an allocation constraint $x_1 + x_2 = 0$ is also shown in Fig. 3 for the sake of illustration.

Future work

Since decisions are taken across multiple decentralized agents with limited data sharing, such a *decomposition-coordination* based collaborative framework can be useful in many applications with data-privacy issues and/or high communication cost. In particular, it would be interesting to explore such ideas for the following two applications:

1. **Machine Learning**: Bayesian optimization for hyperparameter tuning in federated learning - multiple agents must reach consensus on the hyperparameter values.
2. **Network Flow problems**: Routing commodities from arrival point to destination over a network such as in packet routing in wireless networks, traffic routing in transport etc.
3. **Resource allocation**: Resource allocation and optimal exchange in large-scale processing and manufacturing plants such as in eco-industrial parks.

It would be interesting to study the convergence properties of this approach, and other variations to improve its convergence/robustness properties.

Illustrative example - Allocation

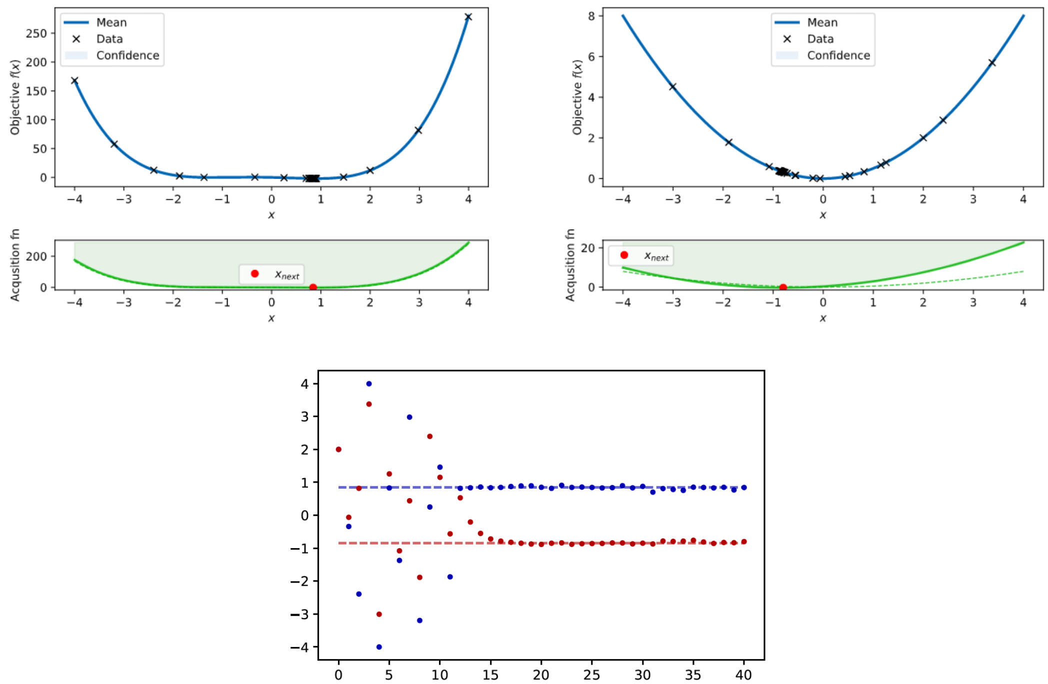

Figure 2: Allocation problem solved using federated Bayesian optimization.

Illustrative example - Consensus

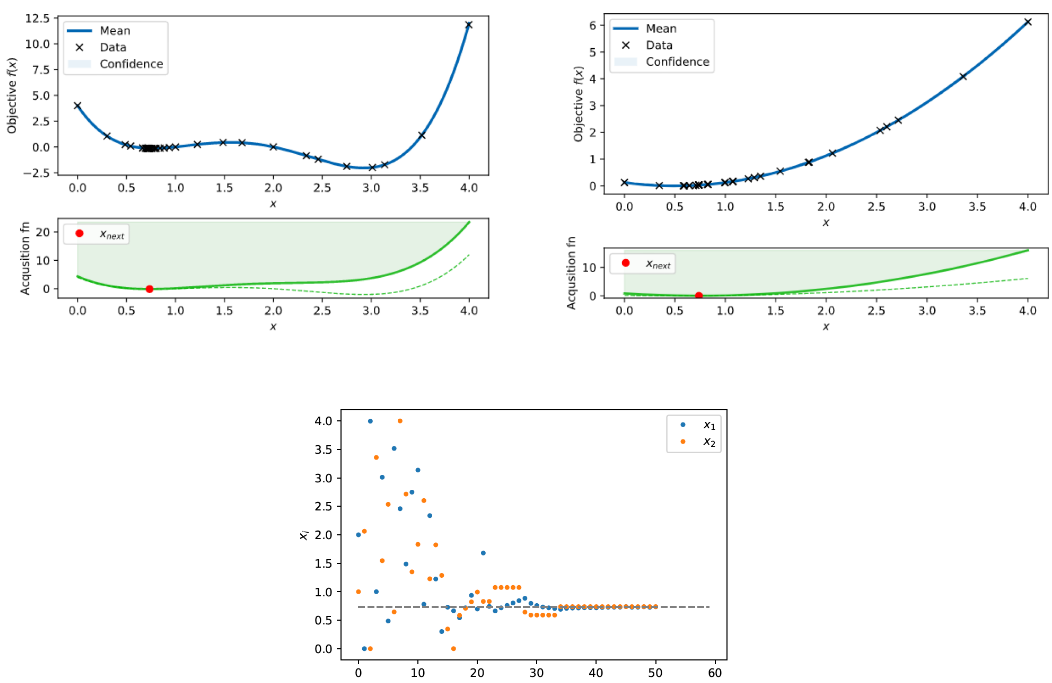

Figure 3: Consensus problem solved using federated Bayesian Optimization.