Parallel Bayesian Global Optimization of Expensive Functions, for Metrics Optimization at Yelp

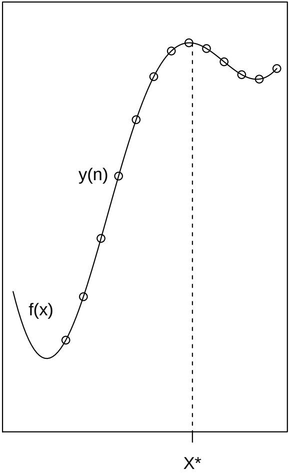
Peter Frazier [1] Jialei Wang [1] Scott Clark [2] Eric Liu [2]
Deniz Oktay [2] Norases Vesdapunt [2]

[1]School of Operations Research & Information Engineering, Cornell University

[2]Yelp Inc.

Thursday October 16, 2014
MIT OR Center
Supported by AFOSR and NSF

Derivative-Free Black-box Global Optimization

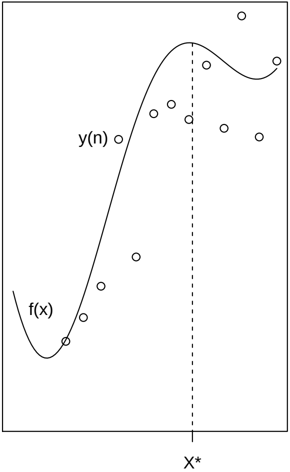


- Objective function $f : \mathbb{R}^d \mapsto \mathbb{R}$, continuous but not concave.
- Feasible set $A \subseteq \mathbb{R}^d$.
- Our goal is to solve

$$\max_{x \in A} f(x)$$

- Assumptions: f is time-consuming to evaluate (hours or days), and derivative information is unavailable.

Derivative-Free Black-box Global Optimization with Noise



- We cannot evaluate $f(x)$ directly.
- Instead, we can evaluate $f(x)$ with noise, e.g., via a stochastic simulator, or a physical experiment.
- We can observe $g(x, \omega) = f(x) + \varepsilon(x, \omega)$, where $E[g(x, \omega)] = f(x)$.
- Our goal is still to find a global maximum,

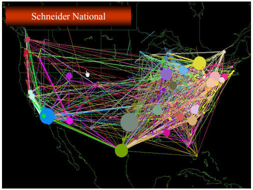
$$\max_{x \in A} f(x)$$

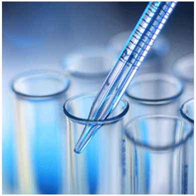
- The term **simulation optimization** is also used.

Bayesian Global Optimization is a class of methods for Derivative-Free Black-Box Global Optimization

- One class of methods for derivative-free black-box global optimization is the class of **Bayesian Global Optimization (BGO)** methods.
- In these methods, we place a **Bayesian prior distribution** on the objective function f. (This is typically a Gaussian process prior).
- Ideally, we would find an algorithm with optimal average-case performance under this prior.
- We will settle for an algorithm with good average-case performance.
- (There are many other types of DFO methods. We do not discuss these in this talk.)

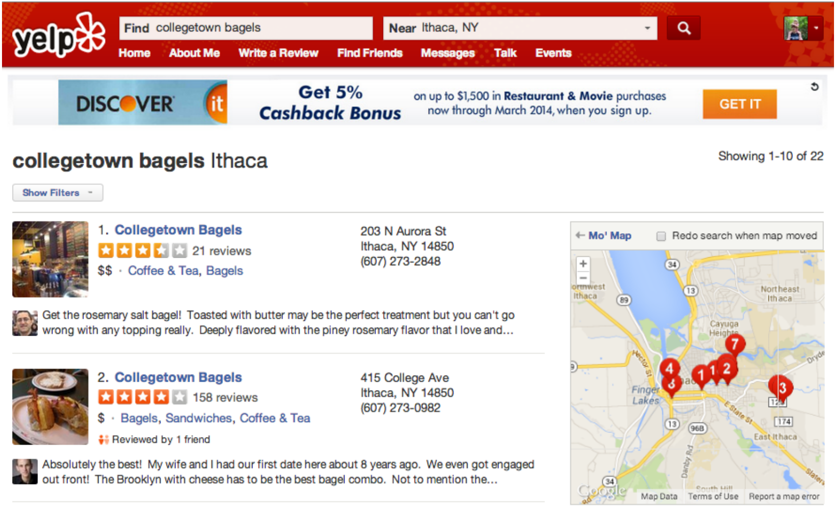
BGO is useful when optimizing with computational or physical experiments





- BGO is often used for optimizing large-scale computational models.
  - Example: Design of grafts to be used in heart surgery. [Xie et al., 2012]
  - Example: Calibration of a simulation-based logistics model. [Frazier et al., 2009c].
- BGO can also be used for optimization problems where "evaluating the objective function" means running a physical experiment
  - Example: Optimizing growth of carbon nanotubes [ongoing research with AFRL].
  - Example: Drug discovery [Negoescu et al., 2011]

BGO is useful when optimizing with computational or physical experiments, at Yelp



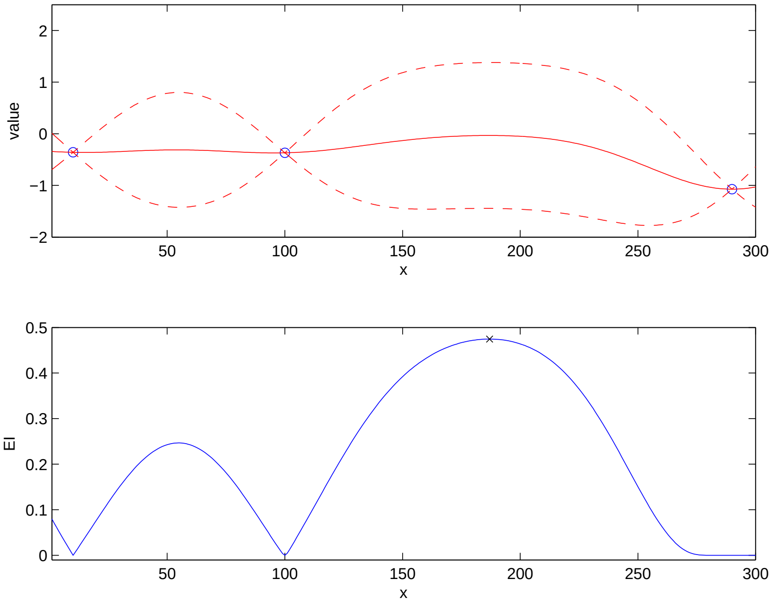
BGO is useful when optimizing with computational or physical experiments, at Yelp

- Yelp uses BGO to optimize tunable parameters for prediction algorithms on previously collected data.
  - Yelp uses machine learning algorithms to predict what users will do, e.g., whether a user will click on an ad.
  - These algorithms have tunable parameters not already optimized by the ML algorithm itself, e.g., hyperparameters in feature definitions.
  - We can evaluate the quality of a parameter set by seeing how well it predicts on previously collected testing data.
- Yelp uses BGO to optimize tunable parameters on live user traffic.
  - Yelp also tunes parameters by trying them out on live users.
  - Example: parameters govern whether Yelp shows (distance + 2 lines of reviews), vs. (3 lines of reviews, without distance), as a function of business category, time of day, location, and desktop/mobile.
  - If I search for coffee on my phone, I want to know how close it is.
  - If I search for a plumber on my laptop, I care about reviews.

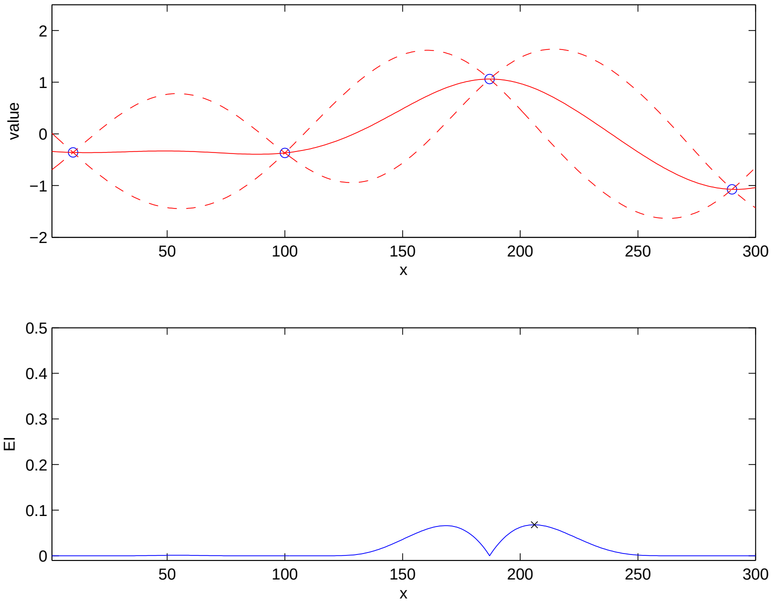
Background: Expected Improvement

- In Bayesian global optimization, we use value of information computations to decide where to sample next.
- Expected Improvement (EI) is a classic method for valuing information in derivative-free global optimization without noise.
  - Perhaps the most well-known expected improvement method is called "Efficient Global Optimization" (EGO) [Jones et al., 1998].
  - The idea of expected improvement goes back further, to at least [Mockus, 1972].
  - Expected improvement is closely related to the knowledge-gradient method [Frazier et al., 2009a] (KG lifts EI's restriction of the implementation decision to previously evaluated points)
- Recently [Ginsbourger et al., 2007] generalized the idea of expected improvement to parallel function evaluations.

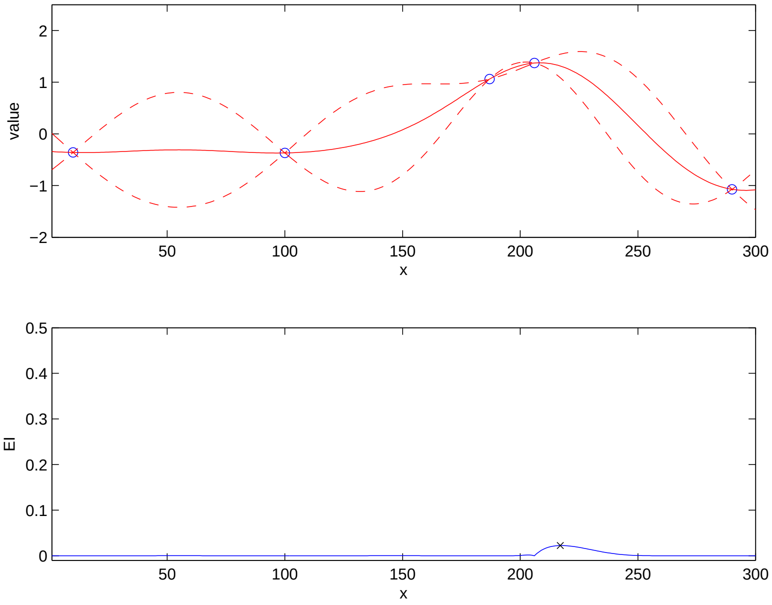
Background: Expected Improvement



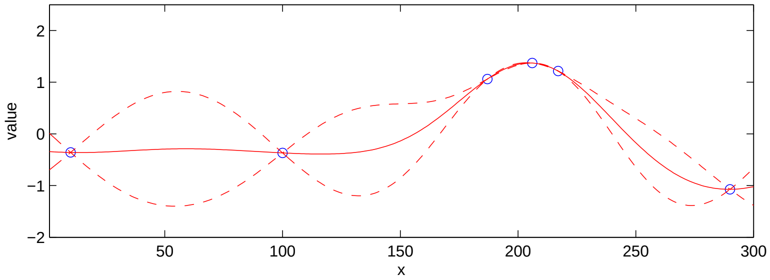
Background: Expected Improvement

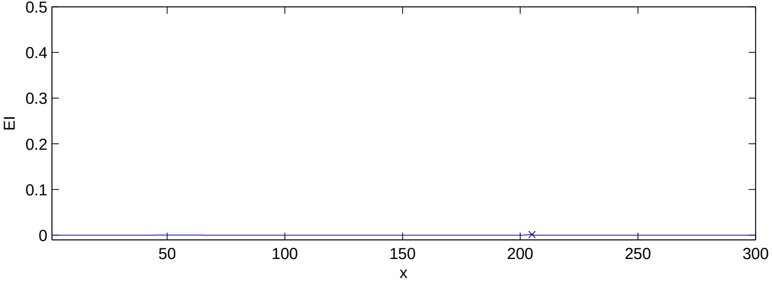


Background: Expected Improvement



Background: Expected Improvement





Almost all existing BGO methods are sequential

- Early work: [Kushner, 1964, Mockus et al., 1978, Mockus, 1989]
- Convergence analysis:
  [Calvin, 1997, Calvin and Zilinskas, 2002, Vazquez and Bect, 2010].
- Perhaps the most well-known method is Efficient Global Optimization (EGO) [Schonlau, 1997, Jones et al., 1998], which uses the notion of expected improvement.
- Recently many metheds have been developed that allow noise:
  [Calvin and Zilinskas, 2005, Villemonteix et al., 2009,
  Frazier et al., 2009b, Huang et al., 2006]

These methods are all fully sequential (one function evaluation at a time).

How can we extend BGO to multiple simultaneous function evaluations?

parallel computer

parallel A/B tests

- What if we can perform multiple function evaluations simultaneously?
- This is the case with parallel computing, in Yelp's live A/B tests, and in many other experimental settings (e.g., in biology).
- We explore an idea that follows naturally from a decision-theoretic analysis.
- This idea was previously suggested by [Ginsbourger et al., 2007].

We generalize to multiple function evaluations using a decision-theoretic approach

- We've evaluated $\vec{x}^{(1)}, \ldots, \vec{x}^{(n)}$, and observed $f(\vec{x}^{(1)}), \ldots, f(\vec{x}^{(n)})$.
- Once sampling stops, we will select the best point evaluated so far.
- **What would be the Bayes-optimal way to choose the set of points $\vec{x}_1, \ldots, \vec{x}_q$ to evaluate next?**
- In general, the optimal points are given by the solution to a dynamic program. (Difficult to solve)
- When this is the last stage of measurements, the dynamic program becomes a more straightforward optimization problem.

Generalizing to multiple function evaluations

- We've evaluated $\vec{x}^{(1)}, \dots, \vec{x}^{(n)}$, and observed $f(\vec{x}^{(1)}), \dots, f(\vec{x}^{(n)})$.
- Let $f_n^* = \max_{m=1,\dots,n} f(\vec{x}^{(m)})$ be the best value observed so far.
- If we measure at new points $\vec{x}_1, \dots, \vec{x}_q$, and if that is our last stage of measurements, then the expected value of our solution is

$$\mathbb{E}_n \left[\max \left(f_n^*, \max_{i=1,\dots,q} f(\vec{x}_i) \right) \right]$$

- This can be rewritten as $\mathrm{EI}_n(\vec{x}_1, \dots, \vec{x}_q) + f_n^*$ where

$$\mathrm{EI}_n(\vec{x}_1, \dots, \vec{x}_q) = \mathbb{E}_n \left[\left(\max_{i=1,\dots,q} f(\vec{x}_i) - f_n^* \right)^+ \right]$$

  is given the name **q-EI** (and also "multipoints expected improvement") by [Ginsbourger et al., 2007].

q-EI gives the single-stage Bayes-optimal set of evaluations

- If we have one stage of function evaulations left to take, and must take our final solution from the set of points that have been evaluated, then evaluating

$$\arg\max_{\vec{x}_1,\ldots,\vec{x}_q} \mathrm{EI}_n(\vec{x}_1,\ldots,\vec{x}_q)$$

  is **Bayes optimal**, i.e., optimal with respect to average case performance under posterior.
- If we have more than one stage left to go, it is a heuristic.

q-EI lacks an easy-to-compute expression

$\mathrm{EI}_n(\vec{x}_1,\ldots,\vec{x}_q) = \mathbb{E}_n\left[(\max_{i=1,\ldots,q} f(\vec{x}_i) - f_n^*)^+\right]$

- When $q = 1$ (no parallelism), this reduces to the expected improvement of [Jones et al., 1998], which has a closed form.
- When $q = 2$, [Ginsbourger et al., 2007] provides an expression in terms of bivariate normal cdfs.
- When $q > 2$, [Ginsbourger et al., 2007] proposes estimation through Monte Carlo, and [Chevalier and Ginsbourger, 2013] proposes exact evaluation using repeated calls to the multivariate normal cdf. Both are difficult to optimize.

q-EI is hard to optimize

- From [Ginsbourger, 2009], "*directly optimizing the q-EI becomes extremely expensive as q and d (the dimension of inputs) grow.*"
- Rather than actually solving $\arg\max_{\vec{x}_1,\ldots,\vec{x}_q} \mathrm{EI}(\vec{x}_1, \ldots, \vec{x}_q)$ when $q > 2$, [Ginsbourger et al., 2007] and [Chevalier and Ginsbourger, 2013] propose other heuristic schemes.

Our Contribution

- Our first contribution is an efficient method for solving

$$\operatorname*{arg\,max}_{\vec{x}_1,\ldots,\vec{x}_q} \mathrm{EI}(\vec{x}_1,\ldots,\vec{x}_q)$$

- This transforms the single-batch Bayes-optimal function evaluation plan, previously considered to be a purely conceptual algorithm, into something implementable.
- Our second contribution is to provide a high-quality open source implementation, which is currently in use at Yelp.

Our approach to solving $\arg\max_{\vec{x}_1,\ldots,\vec{x}_q} \mathrm{EI}(\vec{x}_1,\ldots,\vec{x}_q)$

1. Construct an unbiased estimater of

$$\nabla \mathrm{EI}(\vec{x}_1,\ldots,\vec{x}_q)$$

   using infinitessimal perturbation analysis (IPA).

2. Use multistart stochastic gradient ascent to find an approximate solution to $\max_{\vec{x}_1,\ldots,\vec{x}_q} \mathrm{EI}(\vec{x}_1,\ldots,\vec{x}_q)$.

We construct an estimator of the gradient

- $\mathrm{EI}_n(\vec{x}_1,\ldots,\vec{x}_q) = \mathbb{E}_n\left[(\max_{i=1,\ldots,q} f(\vec{x}_i) - f_n^*)^+\right] = \mathbb{E}_n\left[h\left(\vec{x}_1,\ldots,\vec{x}_q,\vec{Z}\right)\right]$ where $\vec{Z}$ is a q-dimensional standard normal.
- Using sufficient conditions described on the next slide, we switch ∇ and expectation to obtain our unbiased estimator of the gradient,

$$\nabla\mathrm{EI}\left(\vec{x}_1,\ldots,\vec{x}_q,\vec{Z}\right) = \mathbb{E}\vec{g}\left(\vec{x}_1,\ldots,\vec{x}_q,\vec{Z}\right),$$

  where

$$\vec{g}\left(\vec{x}_1,\ldots,\vec{x}_q,\vec{Z}\right) = \begin{cases} \nabla\left[h\left(\vec{x}_1,\ldots,\vec{x}_q,\vec{Z}\right)\right] & \text{if } \nabla\left[h\left(\vec{x}_1,\ldots,\vec{x}_q,\vec{Z}\right)\right] \text{exists,} \\ 0 & \text{if not,} \end{cases}$$

- $g(\vec{x}_1,\ldots,\vec{x}_q,\vec{Z})$ can be computed using results on differentiation of the Cholesky decomposition.

Our gradient estimator is unbiased, given sufficient conditions

Theorem

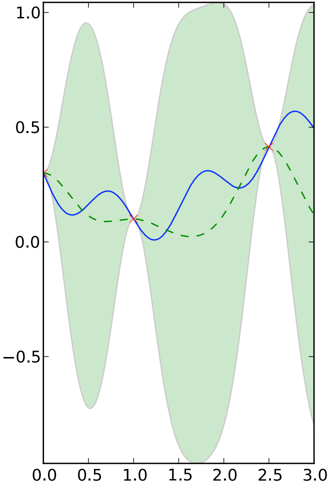
Let $\vec{m}(\vec{x}_1,\ldots,\vec{x}_q)$ and $C(\vec{x}_1,\ldots,\vec{x}_q)$ be the mean vector and Cholesky factor of the covariance matrix of $(f(\vec{x}_1),\ldots,f(\vec{x}_q))$ under the posterior distribution at time n. If the following conditions hold

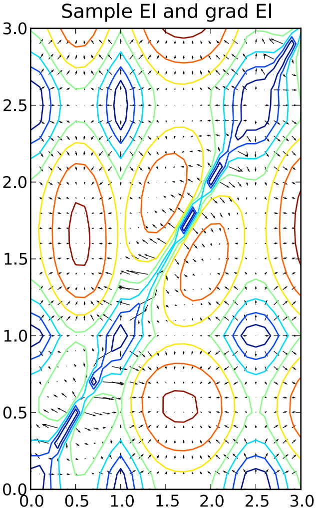
- *$\vec{m}(\cdot)$ and $C(\cdot)$ are three times continuously differentiable in a neighborhood of $\vec{x}_1,\ldots,\vec{x}_q$.*
- *$C(\vec{x}_1,\ldots,\vec{x}_q)$ has no duplicated rows.*

then

$$\nabla \mathrm{EI}(\vec{x}_1,\ldots,\vec{x}_q) = \mathbb{E}_n\left[g(\vec{x}_1,\ldots,\vec{x}_q,\vec{Z})\right].$$

Example of Estimated Gradient



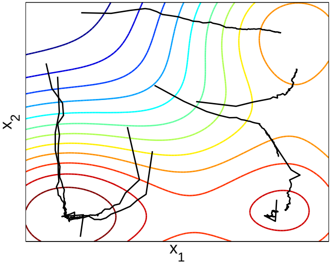


Multistart Stochastic Gradient Ascent

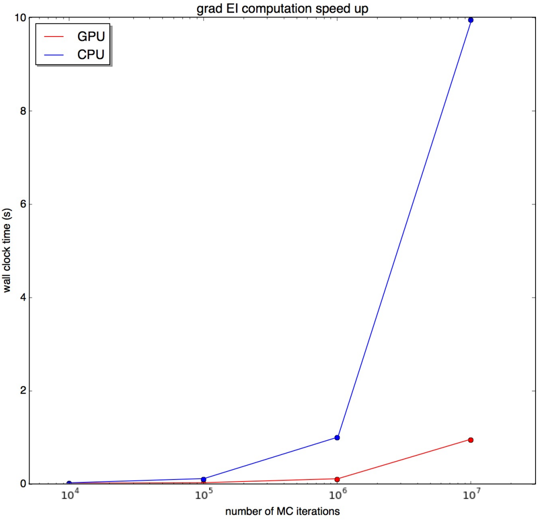
1. Select several starting points, uniformly at random.
2. From each starting point, iterate using the stochastic gradient method until convergence.

$$(\vec{x}_1, \ldots, \vec{x}_q) \leftarrow (\vec{x}_1, \ldots, \vec{x}_q) + \alpha_n g(\vec{x}_1, \ldots, \vec{x}_q, \omega),$$

   where (α_n) is a stepsize sequence.
3. For each starting point, average the iterates to get an estimated stationary point. (Polyak-Ruppert averaging)
4. Select the estimated stationary point with the best estimated value as the solution.

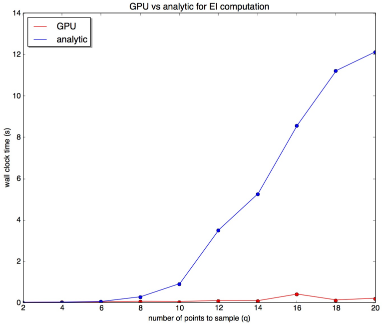


Estimating ∇q-EI can be parallelized on a GPU



Estimating ∇q-EI is faster than computing q-EI exactly

- Evaluating q-EI exactly requires q^2 evaluations of the $q-1$ dimensional multivariate normal cdf.
- This slows as q increases.



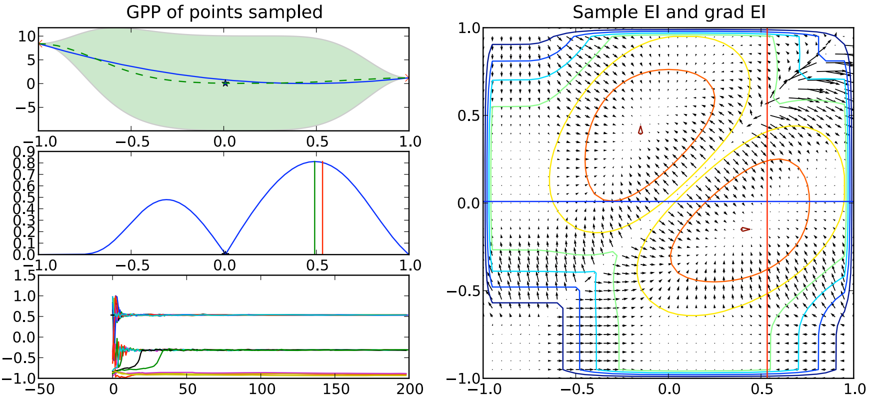
- In progress: speed comparison between our stochastic gradient method, and optimizing the exact q-EI with a derivative-free solver.

We can handle asynchronous function evaluations

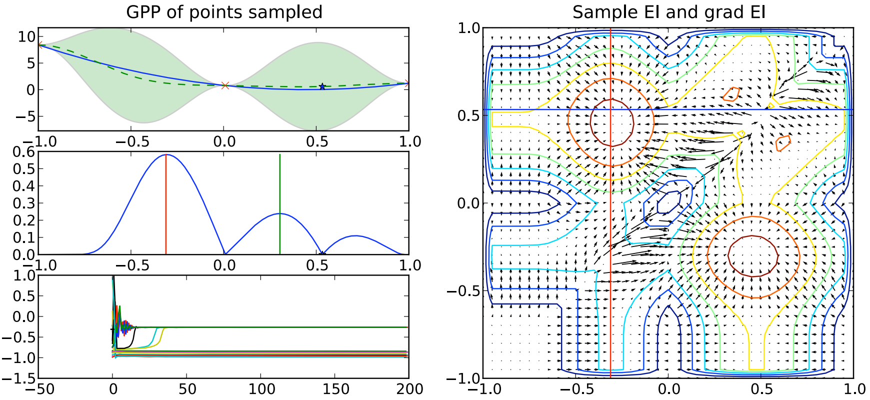
- As previously described, if there are no function evaluations currently in progress, we solve
$$\max_{\vec{x}_1,\ldots,\vec{x}_q} \mathrm{EI}(\vec{x}_1,\ldots,\vec{x}_q)$$
to get the set to run next.
- If there are function evaluations already in progress, say $\vec{x}_1,\ldots,\vec{x}_k$, we take these as given and optimize the rest $\vec{x}_{k+1},\ldots,\vec{x}_q$.

$$\max_{\vec{x}_{k+1},\ldots,\vec{x}_q} \mathrm{EI}(\vec{x}_1,\ldots,\vec{x}_q)$$

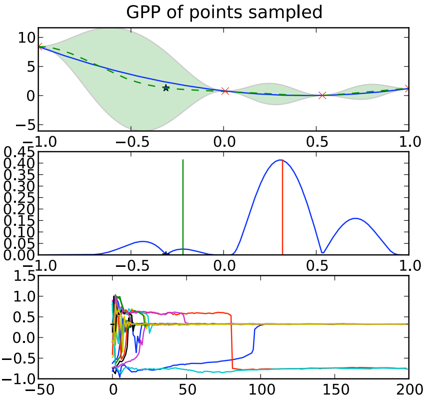
Animation



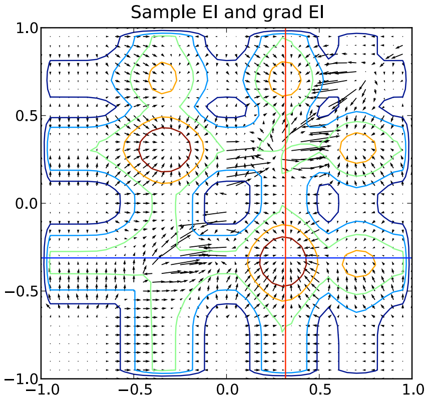
Animation



Animation

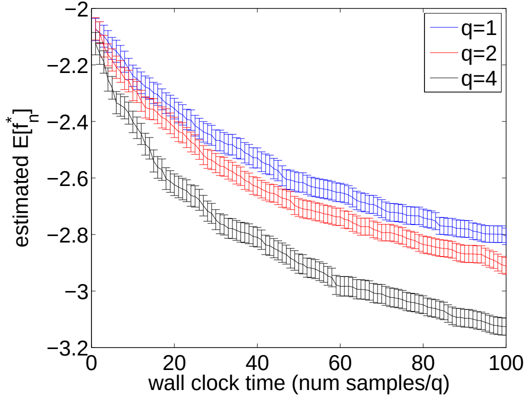


GPP of points sampled
10
5
0
−5
−1.0
−0.5
0.0
0.5
1.0
0.45
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
1.5
1.0
0.5
0.0
−0.5
−1.0
−50
0
50
100
150
200



Sample EI and grad EI
1.0
0.5
0.0
−0.5
−1.0
−1.0
−0.5
0.0
0.5
1.0

The method works: adding parallelism improves average-case performance



- Average performance is evaluated over 156 problems chosen at random from a GP prior.
- The $q = 1$ (one thread) line corresponds to the EGO method of [Jones et al., 1998].

Our procedure is only Bayes-optimal for a single batch

- If we do just one batch, then our procedure is Bayes-optimal.
- We might not stop after one batch of size q.
- We might run many batches, starting each new batch after the previous one completes.
- Our procedure is no longer optimal in this situation.

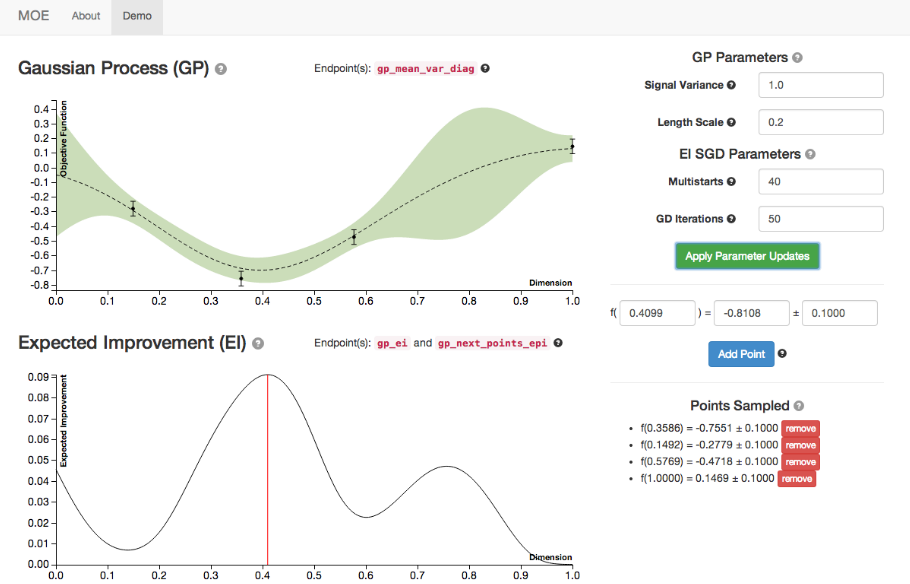
Finding the Bayes-optimal **multi-batch** procedure is hard

- The optimal procedure for $N > 1$ batches requires solving a partially observable Markov decision process (POMDP).
- This is well-understood theoretically, but impossible computationally because:
    - the amount of memory required is exponential in d (the problem dimension), q (the batch size), and N (the number of batches).
    - This exponential scaling is called the "curse of dimensionality."

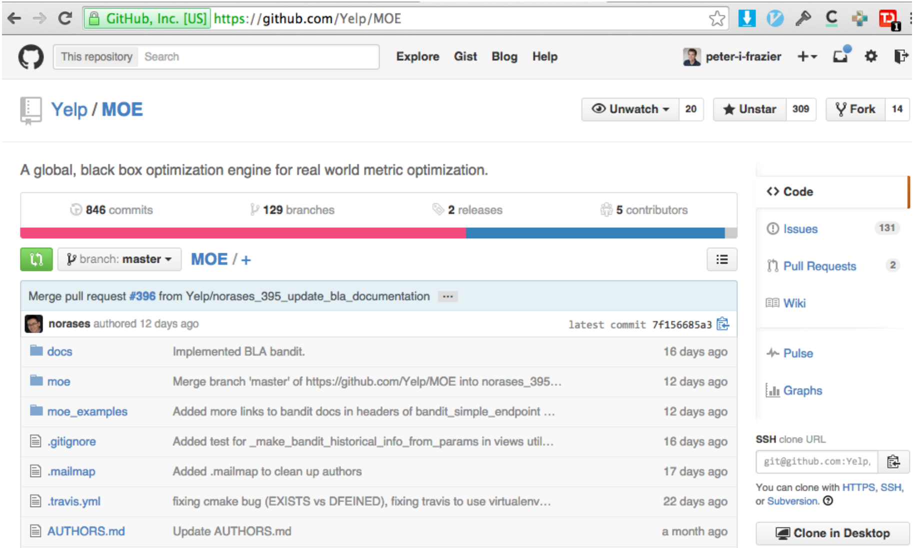
We have found Bayes-optimal multi-batch procedures for other related learning problems

- We have found Bayes-optimal multi-batch procedures, or upper bounds on their value, for these related problems:
    - multiple comparisons [Xie and Frazier, 2013a, Hu et al., 2014],
    - stochastic root-finding [Waeber et al., 2013],
    - ranking and selection [Xie and Frazier, 2013b],
    - information filtering [Zhao and Frazier, 2014].

With Yelp, we made a high-quality implementation, called MOE (Metrics Optimization Engine)

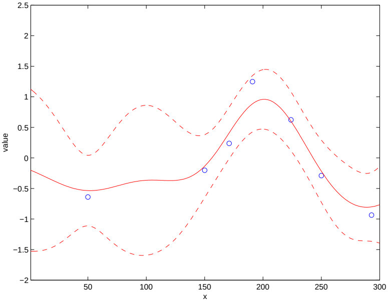


Yelp is using MOE in production.
MOE is open source (http://yelp.github.io/MOE/), so you can use it too.



This q-EI method can be used in the noisy case, but it loses its decision-theoretic motivation

- We use Gaussian process regression with normally distributed noise.
- The red line is the posterior mean, $\mu_n(x) = \mathbb{E}_n[f(x)]$
- The largest posterior mean is $\mu_n^* = \max_{i=1,\ldots,n} \mu_n(\vec{x}^{(m)})$.



- We use $\mathrm{EI}_n(\vec{x}_1, \ldots, \vec{x}_q) = \mathbb{E}_n\left[(\max_{m=1,\ldots,q} \mu_{n+1}(\vec{x}_i) - \mu_n^*)^+\right]$
- This ignores that $\mu_{n+1}(x) \neq \mu_n(x)$ for previously evaluated x.
- A more principled approach is possible (e.g., generalize knowledge gradient method to multiple evaluations), but we haven't done it yet.

Conclusion

- We considered a previously proposed conceptual method for parallel Bayesian global optimization (BGO). This conceptual algorithm was difficult to implement in systems with a large degree of parallelism.
- We used methods from simulation optimization to provide a better implementation.
- The new method provides a significant speedup over EGO when parallelism is available.
- Yelp is using the new method to help you find good restaurants.

References I

Calvin, J. M. (1997).
Average performance of a class of adaptive algorithms for global optimization.
The Annals of Applied Probability, 7(3):711–730.

Calvin, J. M. and Zilinskas, A. (2002).
One-dimensional Global Optimization Based on Statistical Models.
Nonconvex Optimization and its Applications, 59:49–64.

Calvin, J. M. and Zilinskas, A. (2005).
One-Dimensional global optimization for observations with noise.
Computers & Mathematics with Applications, 50(1-2):157–169.

Chevalier, C. and Ginsbourger, D. (2013).
Fast Computation of the Multi-points Expected Improvement with Applications in Batch Selection.
In *Lecture Notes in Computer Science*, pages 59–69.

Frazier, P. I., Powell, W. B., and Dayanik, S. (2009a).
The knowledge gradient policy for correlated normal beliefs.
INFORMS Journal on Computing, 21(4):599–613.

Frazier, P. I., Powell, W. B., and Dayanik, S. (2009b).
The Knowledge Gradient Policy for Correlated Normal Beliefs.
INFORMS Journal on Computing, 21(4):599–613.

Frazier, P. I., Powell, W. B., and Simão, H. P. (2009c).
Simulation Model Calibration with Correlated Knowledge-Gradients.
In *Winter Simulation Conference Proceedings, 2009*. Winter Simulation Conference.

References II

Ginsbourger, D. (2009).
Two advances in Gaussian Process-based prediction and optimization for computer experiments.
In *MASCOT09 Meeting*, pages 1–2.

Ginsbourger, D., Le Riche, R., and Carraro, L. (2007).
A Multi-points Criterion for Deterministic Parallel Global Optimization based on Kriging.
In *Intl. Conf. on Nonconvex Programming, NCP07*, page ..., Rouen, France.

Hu, W., Frazier, P. I., and Xie, J. (2014).
Parallel bayesian policies for finite-horizon multiple comparisons with a known standard.
In Tolk, A., Diallo, S. D., Ryzhov, I. O., Yilmaz, L., Buckley, S., and Miller, J. A., editors, *Proceedings of the 2014 Winter Simulation Conference*, Piscataway, New Jersey. Institute of Electrical and Electronics Engineers, Inc.

Huang, D., Allen, T. T., Notz, W. I., and Zeng, N. (2006).
Global Optimization of Stochastic Black-Box Systems via Sequential Kriging Meta-Models.
Journal of Global Optimization, 34(3):441–466.

Jones, D. R., Schonlau, M., and Welch, W. J. (1998).
Efficient Global Optimization of Expensive Black-Box Functions.
Journal of Global Optimization, 13(4):455–492.

Kushner, H. J. (1964).
A new method of locating the maximum of an arbitrary multi- peak curve in the presence of noise.
Journal of Basic Engineering, 86:97–106.

Mockus, J. (1972).
On Bayesian methods for seeking the extremum.
Automatics and Computers (Avtomatika i Vychislitelnayya Tekchnika), 4(1):53–62.

References III

Mockus, J. (1989).
Bayesian approach to global optimization: theory and applications.
Kluwer Academic, Dordrecht.

Mockus, J., Tiesis, V., and Zilinskas, A. (1978).
The application of Bayesian methods for seeking the extremum.
In Dixon, L. C. W. and Szego, G. P., editors, *Towards Global Optimisation*, volume 2, pages 117–129. Elsevier Science Ltd., North Holland, Amsterdam.

Negoescu, D. M., Frazier, P., and Powell, W. B. (2011).
The knowledge gradient algorithm for sequencing experiments in drug discovery.
INFORMS Journal on Computing, 23(1).

Schonlau, M. (1997).
Computer experiments and global optimization.
PhD thesis, University of Waterloo.

Vazquez, E. and Bect, J. (2010).
Convergence properties of the expected improvement algorithm with fixed mean and covariance functions.
Journal of Statistical Planning and Inference, 140(11):3088–3095.

Villemonteix, J., Vazquez, E., and Walter, E. (2009).
An informational approach to the global optimization of expensive-to-evaluate functions.
Journal of Global Optimization, 44(4):509–534.

Waeber, R., Frazier, P. I., and Henderson, S. G. (2013).
Bisection search with noisy responses.
SIAM Journal on Control and Optimization, 51(3):2261–2279.

References IV

Xie, J. and Frazier, P. I. (2013a).
Sequential bayes-optimal policies for multiple comparisons with a known standard.
Operations Research, 61(5):1174–1189.

Xie, J. and Frazier, P. I. (2013b).
Upper bounds on the bayes-optimal procedure for ranking & selection with independent normal priors.
In Pasupathy, R., Kim, S.-H., Tolk, A., Hill, R., and Kuhl, M. E., editors, *Proceedings of the 2013 Winter Simulation Conference*, pages 877–887, Piscataway, New Jersey. Institute of Electrical and Electronics Engineers, Inc.

Xie, J., Frazier, P. I., Sankaran, S., Marsden, A., and Elmohamed, S. (2012).
Gaussian-process-based black-box optimization for cardiovascular bypass graft design under parameter uncertainty.
In *The 50th Annual Allerton Conference on Communication, Control and Computing*, pages 406–413, Piscataway, New Jersey. Institute of Electrical and Electronics Engineers, Inc.

Zhao, X. and Frazier, P. I. (2014).
Exploration vs. exploitation in the information filtering problem.
in preparation for submission, arXiv preprint arXiv:1407.8186.