Supplement to Constrained Bayesian Optimization with Noisy Experiments

This supplemental material contains details about the experiments and additional simulation results.

1 Synthetic functions

We used four synthetic problems for our study. The first is the following problem of Gramacy et al. (2016), with two parameters and two constraints:

$$\begin{aligned} &\min_{\mathbf{x}\in[0,1]^2} x_1 + x_2 \\ &\text{subject to } 1.5 - x_1 - 2x_2 - 0.5\sin(2\pi(x_1^2 - 2x_2)) \leq 0, \\ &\qquad x_1^2 + x_2^2 - 1.5 \leq 0. \end{aligned}$$

This problem is visualized in Fig. S1.

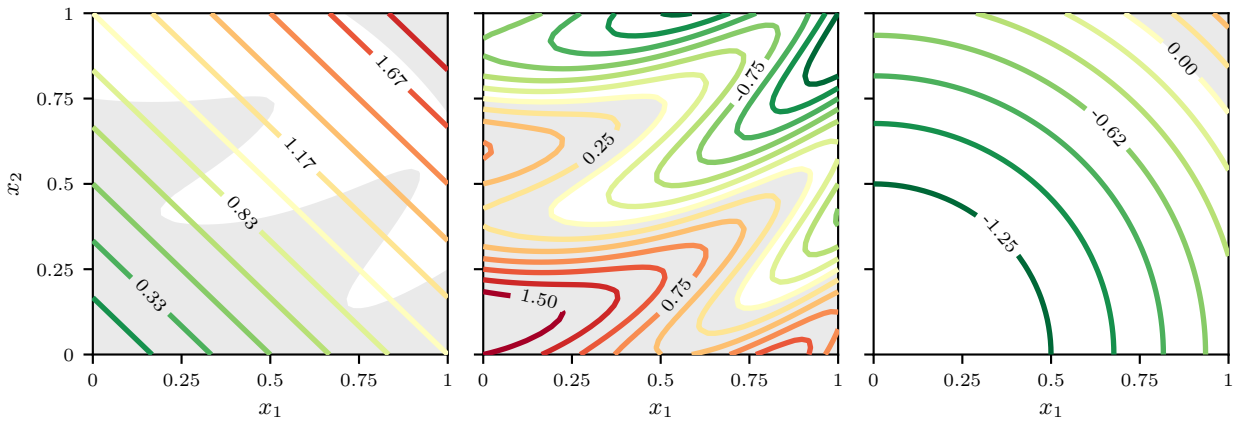

Figure S1: (Left) The objective for the Gramacy problem, with infeasible regions shaded in gray. (Center and right) The constraints for the problem.

The second problem is a constrained version of the Hartmann 6 problem, with six parameters and one constraint:

$$\begin{aligned} &\min_{\mathbf{x}\in[0,1]^6} -\sum_{i=1}^{4} \alpha_i \exp\left(-\sum_{j=1}^{6} A_{ij}(x_j - P_{ij})^2\right) \\ &\text{subject to } \|x\| \leq 1, \end{aligned}$$

with $\alpha = [1.0, 1.2, 3.0, 3.2]$,

$$A = \begin{pmatrix} 10 & 3 & 17 & 3.5 & 1.7 & 8 \\ 0.05 & 10 & 17 & 0.1 & 8 & 14 \\ 3 & 3.5 & 1.7 & 10 & 17 & 8 \\ 17 & 8 & 0.05 & 10 & 0.1 & 14 \end{pmatrix}, \text{ and } P = 10^{-4} \begin{pmatrix} 1312 & 1696 & 5569 & 124 & 8283 & 5886 \\ 2329 & 4135 & 8307 & 3736 & 1004 & 9991 \\ 2348 & 1451 & 3522 & 2883 & 3047 & 6650 \\ 4047 & 8828 & 8732 & 5743 & 1091 & 381 \end{pmatrix}.$$

A slice of this problem is shown in Fig. S2.

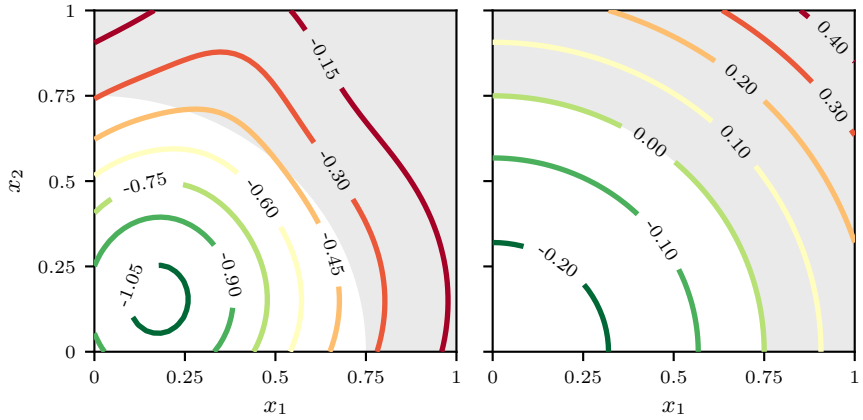

Figure S2: A slice of the Hartmann6 objective function (left) and constraint (right). Shaded region is infeasible. Parameters $x_3, \ldots, x_6$ were set to 0.5 for this figure.

The third problem is the classic Branin function, with a constraint added by Gelbart et al. (2014):

$$\begin{aligned} &\text{minimize } a(x_2 - bx_1^2 + cx_1 - r)^2 + s(1-t)\cos(x_1) + s \\ &\text{subject to } (x_1 - 2.5)^2 + (x_2 - 7.5)^2 - 50 \leq 0, \\ &\qquad x_1 \in [-5, 10], x_2 \in [0, 15], \end{aligned}$$

with $a = 1$, $b = 5.1/(4\pi^2)$, $c = 5/\pi$, $r = 6$, $s = 10$, and $t = 1/(8\pi)$. This function and its constraint are shown in Fig. S3.

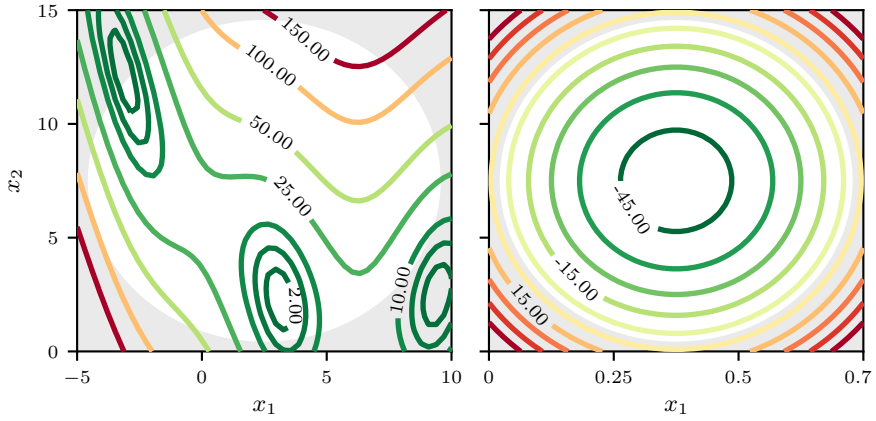

Figure S3: The Branin objective function (left) and constraint (right). Shaded region is infeasible.

The fourth synthetic problem is given by Gardner et al. (2014):

$$\begin{aligned} &\text{minimize } \cos(2x_1)\cos(x_2) + \sin(x_1) \\ &\text{subject to } \cos(x_1)\cos(x_2) - \sin(x_1)\sin(x_2) - 0.5 \leq 0 \\ &\qquad x_1 \in [0, 6], x_2 \in [0, 6], \end{aligned}$$

This function and its constraint are shown in Fig. S4.

Noisy observations were simulated by adding normally distributed noise $\mathcal{N}(0, \tau^2)$ to objective and constraint evaluations. We used $\tau = 0.2$ for the Gramacy, Hartmann6, and Gardner problems, and $\tau = 5$ for the Branin problem.

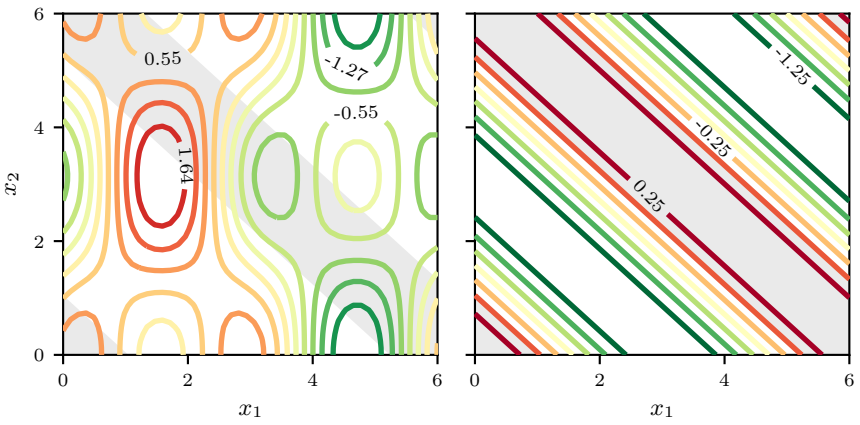

Figure S4: The Gardner problem objective function (left) and constraint (right). Shaded region is infeasible.

2 QMC performance simulations

Each problem was initialized with 5 noisy observations, and 5 unobserved, pending observations, all taken from a scrambled Sobol sequence. The NEI including these observations is shown for each of the 2-d problems in Fig. S5. This is the ground-truth NEI, measured using 10^4 MC samples.

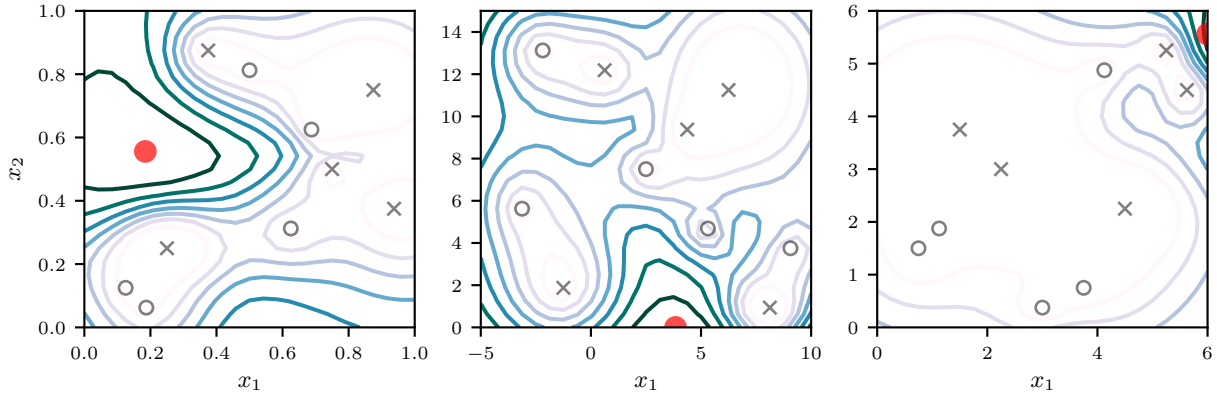

Figure S5: The NEI surface for the initialization of each 2-d problem: Gramacy (left), Branin (center), and Gardner (right). The locations of the 10 completed/pending observations are indicated with gray markers: "x" for completed observations, "o" for pending. The location of the global optimum NEI is marked in red.

Integration error in Fig. 2 in the main text was measured at the global optimum in Fig. S5, and distance to the true optimizer was the distance to that point. Fig. S6 shows the integration error and optimizer distance for the Hartmann6, Branin, and Gardner problems. These figures show the same improvement in optimization performance shown with the Gramacy problem in the main text.

3 Optimization performance simulations

Empirically, we found that the candidates from EI+heuristics tended to clump, especially within a batch. It was not unusual for a batch to contain duplicates of a single point. This behavior is illustrated in Fig. S7 which shows the NEI and EI+heuristics candidates for one round of optimization on the Gramacy problem. In this particular optimization, many of the EI+heuristics iterations were spent re-sampling near a single point, which will happen until its probability of feasibility is driven above the threshold or it becomes clear that it is not feasible. In this particular optimization that point was not feasible, and by iteration 50 the

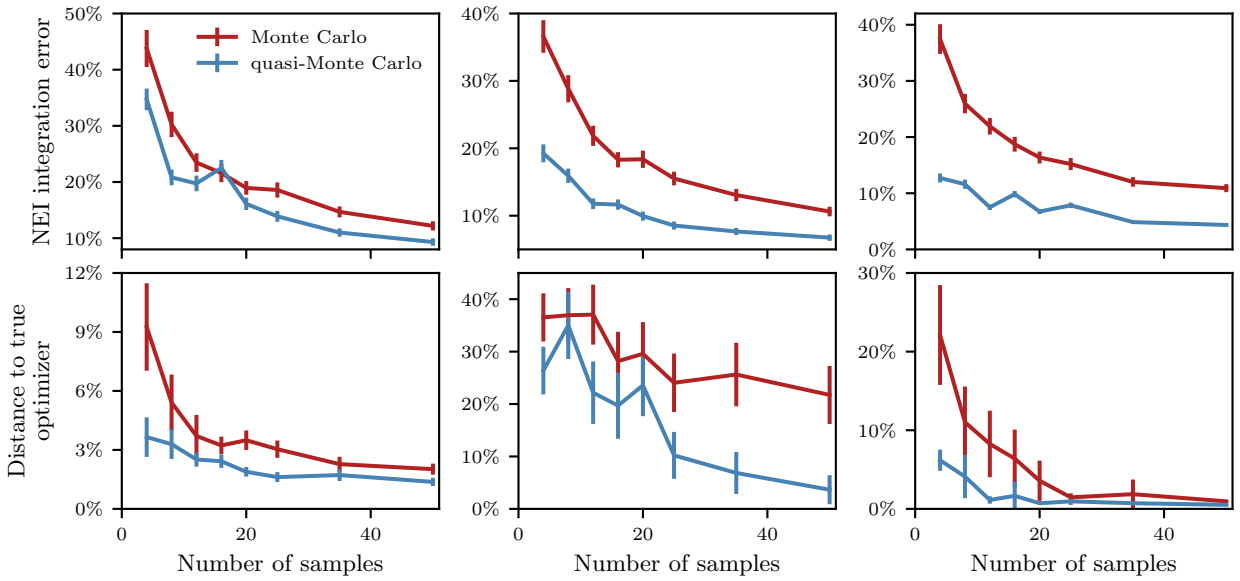

Figure S6: (Top) NEI integration error as a function of the number of MC or QMC samples used for the approximation, for the (left) Hartmann6, (center) Branin, and (right) Gardner problems. For each number of samples, the line indicates the average error over 500 replicates, and error bars are two standard errors of the mean. (Bottom) Average distance of replicates from the optimizer using the approximated NEI to the true NEI global optimum, as a percent of the maximum distance in the search space.

best feasible point that EI+heuristics had found was 0.83. NEI candidates were more widely spread near the optimum, which still allowed the GP to estimate feasibility in that region while also acquiring much better feasible points.

This clumping can be seen across all of the optimization runs in Fig. S8. This figure shows a smoothed density of the pairwise distances between all 50 of the candidates within each optimization run, for all 100 of the runs. With the exception of the Hartmann6 problem, EI+heuristics showed a peak of samples very close to each other where NEI did not.

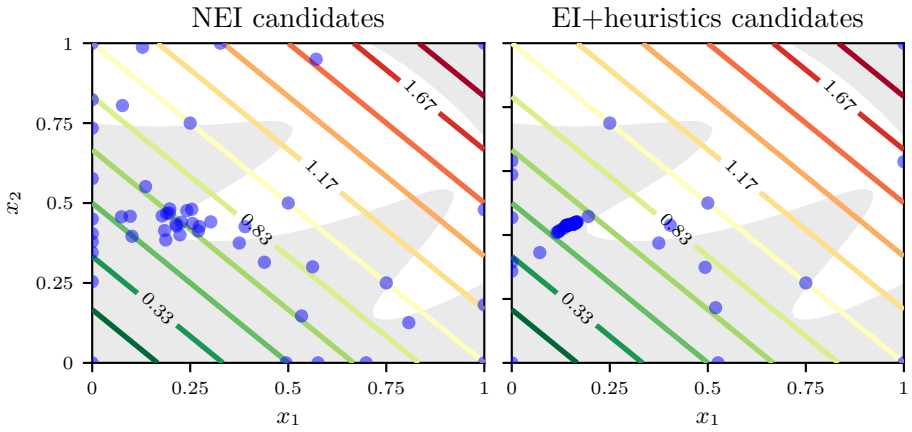

Figure S7: Candidates for one round of optimization on the Gramacy problem for NEI (left) and EI+heuristics (right). Candidates from EI+heuristics often cluster.

NEI is able to sample better points (Fig. 3 of the main text), but the exploitation seen in EI+heuristics could make it easier to identify the best feasible point. Fig. 4 of the main text showed that for the Hartmann6 problem, when we use the GP to identify the best feasible point after each batch, NEI is able to identify

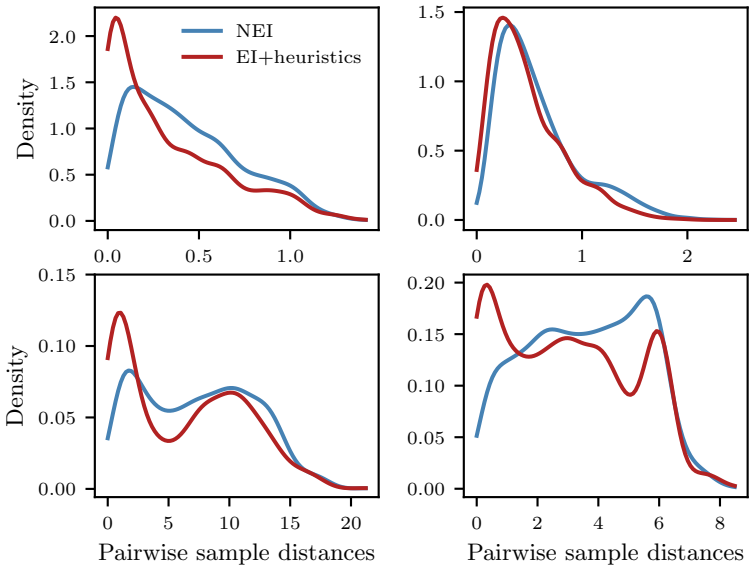

Figure S8: Smoothed density of the pairwise distance between EI-proposed samples for the Gramacy (top left), Hartmann6 (top right), Branin (bottom left), and Gardner (bottom right) problems. For three of the problems, EI+heuristics had a much higher concentration of points very close to each other.

better points. This result is shown for the other three problems in Fig. S9. For the Gramacy problem, the optimum sits on the boundary of feasibility. This criterion is relatively conservative in the probability of feasibility that it tolerates, and selects arms that are feasible with probability greater than 0.9. At this level of feasibility, both NEI and EI+heuristics had similar best objectives. For the other problems where the optimum w as not on a boundary of feasibility, NEI outperformed EI+heuristics.

These figures used the strategy of (4) from the main text to choose the best arm. An alternative strategy is to choose the arm with the best posterior objective mean, which is feasible with probability at least $1-\delta$. The objective values and feasibilities of the best arm chosen using this criterion are shown in Fig. S10, for a range of requested feasibility probabilities $1-\delta$. Shown here is the best point after all batches of optimization. For the Gramacy problem, the actual proportion of feasible points was close to the requested probability. When we are less strict in requiring feasibility, NEI has points with lower objective value due to having better sampled around the optimum.

NEI requires setting a penalty M for ending the optimization without a feasible point. Equation (5) in the main text shows that the EI with no feasible points is $(M-\mu(\mathbf{x}))\mathbb{P}(\mathbf{x}$ is feasible$)$. If $M < \min_{\mathbf{x}} \mu(\mathbf{x})$, then EI will be maximized by minimizing the probability of feasibility. To avoid this degenerate behavior, M must be set large enough that we prefer to find a feasible point, however the actual value had little importance in the optimizations done here. Equation (5) in the main text shows that M only plays a role in the acquisition function in draws for which there are no feasible observations. For the Gramacy, Branin, and Gardner problems, the model fit to the first batch of 5 points (the initialization) provided a feasible observation point in 100% of posterior samples. M thus played no role in those optimizations. For Hartmann6, 27% of model posterior samples had no feasible observations after the initialization. M thus played a role in constructing the first optimized batch, but after the first optimized batch and for the subsequent 8 batches there was a feasible point in 100% of samples and so M played no role. Generally, M will be important in problems where it is difficult to find a feasible solution.

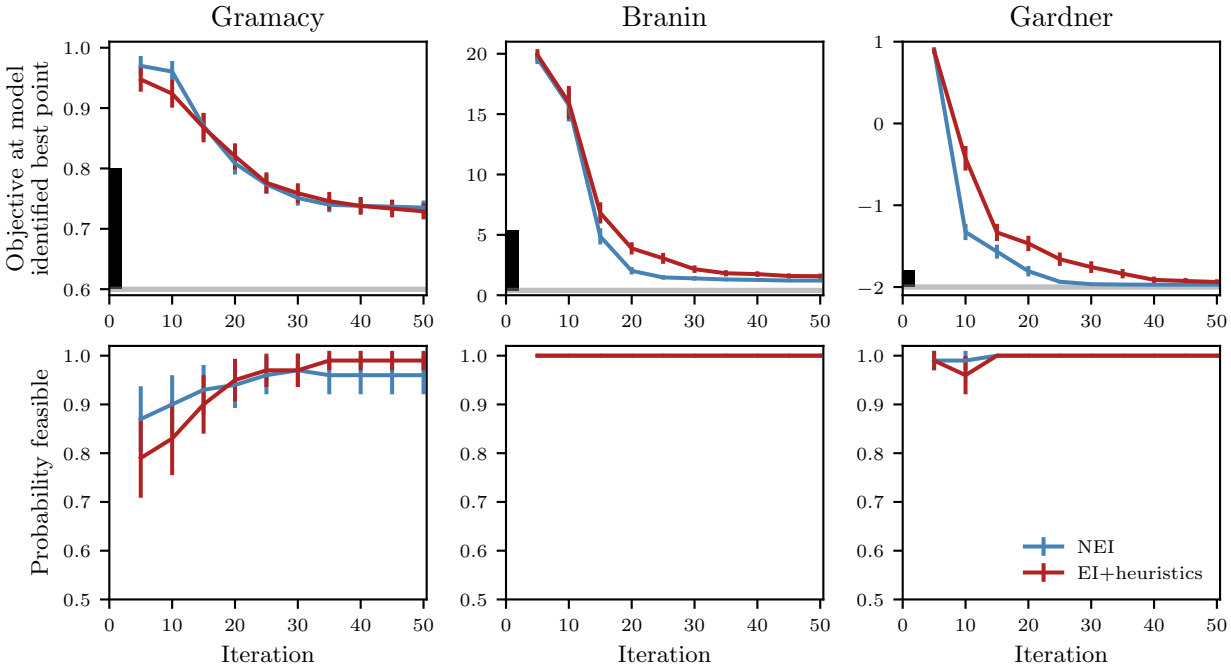

Figure S9: (Top) For each problem, the objective value (mean and two standard errors) of the arm identified from the model as being best after each batch of the simulation in Fig. 3 of the main text. These results use the identification strategy of (4) from the main text. (Bottom) The proportion of replicates in which the model identified best point was actually feasible.

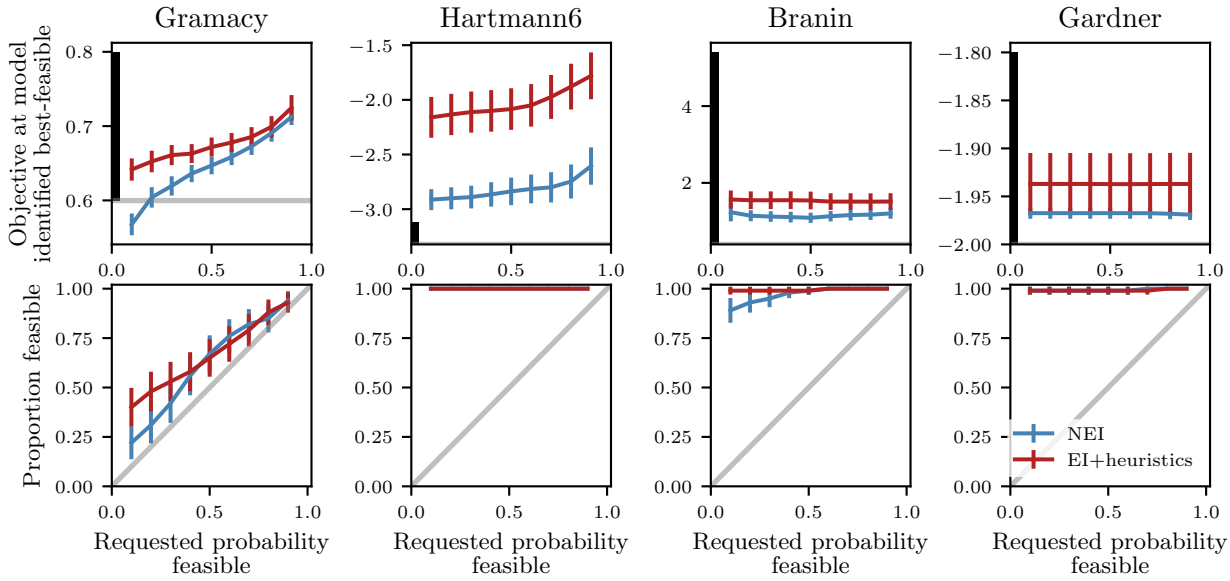

Figure S10: (Top) For each problem, the objective value (mean and two standard errors) of the model identified best-feasible point at the end of optimization, where feasibility was requested at the level indicated. (Bottom) The proportion of the model estimated best-feasible points that were actually feasible.

4 HHVM optimization parameters

The HHVM optimization tuned the the following 7 parameters, with ranges given in parentheses:

- HHIRInliningMaxReturnDecRefs (4 - 13)
- HHIRInliningMaxVasmCost (200 - 1200)
- HHIRMixedArrayProfileThreshold (0.7 - 0.9)
- JitMaxRegionInstrs (500 - 1500)
- JitMaxTranslations (6 - 18)
- JitPGOReleaseVVMinPercent (5 - 15)
- JitUnlikelyDecRefPercent (5 - 20)

The parameter HHIRMixedArrayProfileThreshold was a float, and the other parameters were integers.

References

Gardner, J. R., Kusner, M. J., Xu, Z., Weinberger, K. Q. and Cunningham, J. P.: 2014, Bayesian optimization with inequality constraints, *Proceedings of the 31st International Conference on Machine Learning*, ICML.

Gelbart, M. A., Snoek, J. and Adams, R. P.: 2014, Bayesian optimization with unknown constraints, *Proceedings of the 30th Conference on Uncertainty in Artificial Intelligence*, UAI.

Gramacy, R. B., Gray, G. A., Digabel, S. L., Lee, H. K. H., Ranjan, P., Wells, G. and Wild, S. M.: 2016, Modeling an augmented Lagrangian for blackbox constrained optimization, *Technometrics* **58**(1), 1–11.