Bayesian optimization under mixed constraints with a slack-variable augmented Lagrangian

Authored by:

Robert B. Gramacy
Victor Picheny
Stefan Wild
Sebastien Le Digabel

Abstract

An augmented Lagrangian (AL) can convert a constrained optimization problem into a sequence of simpler (e.g., unconstrained) problems which are then usually solved with local solvers. Recently, surrogate-based Bayesian optimization (BO) sub-solvers have been successfully deployed in the AL framework for a more global search in the presence of inequality constraints; however a drawback was that expected improvement (EI) evaluations relied on Monte Carlo. Here we introduce an alternative slack variable AL, and show that in this formulation the EI may be evaluated with library routines. The slack variables furthermore facilitate equality as well as inequality constraints, and mixtures thereof. We show our new slack "ALBO" compares favorably to the original. Its superiority over conventional alternatives is reinforced on several new mixed constraint examples.

1 Paper Body

Bayesian optimization (BO), as applied to so-called blackbox objectives, is a modernization of 197080s statistical response surface methodology for sequential design [3, 14]. In BO, nonparametric (Gaussian) processes (GPs) provide flexible response surface fits. Sequential design decisions, socalled acquisitions, judiciously balance exploration and exploitation in search for global optima. For reviews, see [5, 4]; until recently this literature has focused on unconstrained optimization. Many interesting problems contain constraints, typically specified as equalities or inequalities: min {f (x) : g(x) ? 0, h(x) = 0, x ? B} , x

(1)

where B ? Rd is usually a bounded hyperrectangle, f : Rd ? R is a scalar-valued objective function, and g : Rd ? Rm and h : Rd ? Rp are vector-valued constraint functions taken componentwise (i.e., gj (x) ? 0, j = 1, . .

. , m; hk (x) = 0, and k = 1, ... , p). The typical setup treats f , g, and h as a ?joint? blackbox, meaning that providing x to a single computer code reveals f (x), g(x), and h(x) simultaneously, often at great computational expense. A common special case treats f (x) as known (e.g., linear); however the problem is still hard when g(x) ? 0 defines a nonconvex valid region. Not many algorithms target global solutions to this general, constrained blackbox optimization problem. Statistical methods are acutely few. We know of no methods from the BO literature natively accommodating equality constraints, let alone mixed (equality and inequality) ones. Schonlau et al. 30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

[21] describe how their expected improvement (EI) heuristic can be extended to multiple inequality constraints by multiplying by an estimated probability of constraint satisfaction. Here, we call this expected feasible improvement (EFI). EFI has recently been revisited by several authors [23, 7, 6]. However, the technique has pathological behavior in otherwise idealized setups [9], which is related to a so-called ?decoupled? pathology [7]. Some recent information-theoretic alternatives have shown promise in the inequality constrained setting [10, 17]. We remark that any problem with equality constraints can be ?transformed? to inequality constraints only, by applying h(x) ? 0 and h(x) ? 0 simultaneously. However, the effect of such a reformulation is rather uncertain. It puts double-weight on equalities and violates certain regularity (i.e., constraint qualification [15]) conditions. Numerical issues have been reported in empirical work [1, 20]. In this paper we show how a recent BO method for inequality constraints [9] is naturally enhanced to handle equality constraints, and therefore mixed ones too. The method involves converting inequality constrained problems into a sequence of simpler subproblems via the augmented Lagrangian (AL, [2]). AL-based solvers can, under certain regularity conditions, be shown to converge to locally optimal solutions that satisfy the constraints, so long as the sub-solver converges to local solutions. By deploying modern BO on the subproblems, as opposed to the usual local solvers, the resulting meta-optimizer is able to find better, less local solutions with fewer evaluations of the expensive blackbox, compared to several classical and statistical alternatives. Here we dub that method ALBO. To extend ALBO to equality constraints, we suggest the opposite transformation to the one described above: we convert inequality constraints into equalities by introducing slack variables. In the context of earlier work with the AL, via conventional solvers, this is rather textbook [15, Ch. 17]. Handling the inequalities in this way leads naturally to solutions for mixed constraints and, more importantly, dramatically improves the original inequality-only version. In the original (non-slack) ALBO setup, the density and distribution of an important composite random predictive quantity is not known in closed form. Except in a few particular cases [18], calculating EI and related quantities under the AL required Monte Carlo integration, which means that acquisition function evaluations are computationally expensive, noisy, or both. A reformulated slack-AL version emits a composite that has a known distribution, a so-called weighted non-central Chi-square (WNCS) distribution. We show that, in that setting, EI calculations involve a simple 1-d integral via

ordinary quadrature. Adding slack variables increases the input dimension of the optimization subproblems, but only artificially so. The effects of expansion can be mitigated through optimal default settings, which we provide. The remainder of the paper is organized as follows. Section 2 outlines the components germane to the ALBO approach: AL, Bayesian surrogate modeling, and acquisition via EI. Section 3 contains the bulk of our methodological contribution: a slack variable AL, a closed form EI, optimal default slack settings, and open-source software. Implementation details are provided by our online supplementary material. Section 4 provides empirical comparisons, and Section 5 concludes.

2

A review of relevant concepts: EI and AL

EI: The canonical acquisition function in BO is expected improvement (EI) [12]. Consider a surrogate f n (x), trained on n pairs (xi , yi = f (xi)) emitting Gaussian predictive equations with mean ?n (x) n and standard deviation ? n (x). Define fmin = mini=1,...,n yi , the smallest y-value seen so far, and n let I(x) = max{0, fmin ? Y (x)} be the improvement at x. I(x) is largest when Y (x) ? f n (x) has n substantial distribution below fmin . The expectation of I(x) over Y (x) has a convenient closed form, n revealing balance between exploitation (?n (x) under fmin) and exploration (large ? n (x)): n

n

fmin ? ?n (x) fmin ? ?n (x) n n E{I(x)} = (fmin ? ?n (x))? + ? (x)? , (2) ? n (x) ? n (x) where ? (?) is the standard normal cdf (pdf). Accurate, approximately Gaussian predictive equations are provided by many statistical models (e.g., GPs). In non-Gaussian contexts, Monte Carlo schemes? sampling Y (x)?s and averaging I(x)?s?offer a computationally intensive alternative. AL: Although several authors have suggested extensions to EI for constraints, the BO literature has primarily focused on unconstrained problems. The range of constrained BO options was recently extended by borrowing an apparatus from the mathematical optimization literature, the augmented Lagrangian, allowing unconstrained methods to be adapted to constrained problems. The AL, as a 2

device for solving problems with inequality constraints (no h(x) in Eq. (1)), may be defined as m 1 X 2 ¿ max {0, gj (x)} , (3) LA (x; ?, ?) = f (x) + ? g(x) + 2? j=1 where ? ¿ 0 is a penalty parameter on constraint violation and ? ? Rm + serves as a Lagrange multiplier. AL methods are iterative, involving a particular sequence of (x; ?, ?). Given the current values ?k?1 and ?k?1 , one approximately solves the subproblem

min LA (x; ?k?1 , ?k?1) : x ? B , (4) x

via a conventional (bound-constrained) solver. The parameters (?, ?) are updated depending on the nature of the solution found, and the process repeats. The particulars in our setup are provided in Alg. 1; for more details see [15, Ch. 17]. Local convergence is guaranteed under relatively mild conditions involving the choice of subroutine solving (4). Loosely, all that is required is that the solver ?makes progress? on the subproblem. In contexts where termination depends more upon computational budget than on a measure of convergence, as in many BO problems, that added flexibility is welcome. However, the AL

does not typically 0 0 enjoy global scope. The local min- Require: ? ? 0, ? ¿ 0 ima found by the method are sen- 1: for k = k1, 2, . . . do solve (4) sitive to initialization?of starting 2: Let x (approximately) k?1 1 k 0 0 0 g (xk)}, j = 1, . . . , m 3: Set ? = max{0, ? + choices for (? , ?) or x ; local j j ?k?1 j searches in iteration k are usually 4: If g(xk) ? 0, set ?k = ?k?1 ; else, set ?k = 12 ?k?1 started from xk?1 . However, this 5: end for dependence is broken when statistiAlgorithm 1: Basic augmented Lagrangian method cal surrogates drive search for solutions to the subproblems. Independently fit GP surrogates, f n (x) for the objective and n g n (x) = (g1n (x), . . . , gm (x)) for the constraints, yield predictive distributions for Yfn (x) and Ygn (x) = (Ygn1 (x), . . . , Ygnm (x)). Dropping the n superscripts, the AL composite random variable Pm 1 2 Y (x) = Yf (x) + ?¿ Yg (x) + 2? j=1 max{0, Ygj (x)} can serve as a surrogate for (3); however, it is difficult to deduce its distribution from the components of Yf and Yg , even when those are independently Gaussian. While its mean is available in closed form, EI requires Monte Carlo.

3

A novel formulation involving slack variables

An equivalent formulation of (1) involves introducing slack variables, sj , for j = 1, . . . , m (i.e., one for each inequality constraint gj (x)), and converting the mixed constraint problem (1) to one with only equality constraints (plus bound constraints for sj): gj (x) ? sj = 0, sj ? R+ , for j = 1, . . . , m. Observe that introducing the slack "inputs" increases dimension of the problem from d to d + m. Reducing a mixed constraint problem to one involving only equality and bound constraints is valuable insofar as one has good solvers for those problems. Suppose, for the moment, that the original problem (1) has no equality constraints (i.e., p = 0). In this case, a slack variable-based AL method is readily available?as an alternative to the description in Section 2. Although we frame it as an ?alternative?, some would describe this as the standard version [see, e.g., 15, Ch. 17]. The AL is m 1 X 2 LA (x, s; ?g , ?) = f (x) + ?¿ (g(x)+s) + (gj (x)+sj) . (5) 2? j=1 This formulation is more convenient than (3) because the ?max? is missing, but the extra slack variables mean solving a higher (d + m) dimensional subproblem compared to (4). That AL can be expanded to handle equality (and thereby mixed constraints) as follows: ? ? p m X X 1 2 ¿ ? LA (x, s; ?g , ?h , ?) = f (x)+?¿ (gj (x)+sj) + hk (x)2 ?. (6) g (g(x)+s)+?h h(x)+ 2? j=1 k=1

¿ ¿ ¿ ¿ ¿ ¿ Defining c(x) := g(x) , h(x) , ? := ?g , ?h , and enlarging the dimension of s with the understanding that sm+1 = ? ? ? = sm+p = 0, leads to a streamlined AL for mixed constraints LA (x, s; ?, ?) = f (x) + ?¿ (c(x) + s) +

3

m+p 1 X 2 (cj (x) + sj) , 2? j=1

(7)

with ? ? Rm+p . A non-slack AL formulation (3) can analogously be written as ? ? p m X X 1 2 ¿ ? LA (x; ?g , ?h , ?) = f (x) + ?¿ max {0, gj (x)} + hk (x)2 ? , g g(x) + ?h h(x) + 2? j=1 k=1

p with ?g ? Rm + and ?h ? R . Eq. (7), by contrast, is easier to work with

because it is a smooth quadratic in the objective (f) and constraints (c). In what follows, we show that (7) facilitates calculation of important quantities like EI, in the GP-based BO framework, via a library routine. So slack variables not only facilitate mixed constraints in a unified framework, but they also lead to a more efficient handling of the original inequality (only) constrained problem.

3.1

Distribution of the slack-AL composite

If Yf and Yc1 , . . . , Ycm+p represent random predictive variables from m + p + 1 surrogates fitted to n realized objective and constraint evaluations, then the analogous slack-AL random variable is Y (x, s) = Yf (x) +

m+p X j=1

m+p 1 X ?j (Ycj (x) + sj) + (Ycj (x) + sj)2 . 2? j=1

(8)

As for the original AL, the mean of this RV has a simple closed form in terms of the means and variances of surrogates. In the Gaussian case, we show that we can obtain a closed form for the full distribution of the slack-AL variate (8). Toward that aim, first rewrite Y as: Y (x, s) = Yf (x) +

m+p X j=1

= Yf (x) +

m+p X

m+p m+p

1 X 2 1 X ?j sj + s + 2?j ?Ycj (x) + 2sj Ycj (x) + Ycj (x)2 2? j=1 j 2? j=1

?j sj +

j=1

m+p m+p i 2 1 Xh 1 X 2 sj + ?j + Ycj (x) ? ?j2 , 2? j=1 2? j=1

with ?j = ?j ? + sj . Now decompose the Y (x, s) into a sum of three quantities: 1 W (x, s), with 2? m+p m+p m+p X 1 X 2 1 X 2 s ? ? r(s) = ?j sj + 2? j=1 j 2? j=1 j j=1

Y (x, s) = Yf (x) + r(s) +

(9) and W (x, s) =

m+p X

?j + Ycj (x)

2

.

j=1

Using Ycj ? N ?cj (x), ?c2j (x) , i.e., leveraging Gaussianity, W can be written as W (x, s) =

m+p X

?c2j (x)Xj (x, s),

with Xj (x, s) ? ?

j=1

2

?cj (x) + ?j dof = 1, ? = ?cj (x)

2 ! . (10)

The line above is the expression of a weighted sum of non-central chi-square (WSNC) variates. Each of the m + p variates involves a unit degrees-of-freedom

(dof) parameter, and a non-centrality parameter ?. A number of efficient methods exist for evaluating the density, distribution, and quantile functions of WSNC random variables. Details and code are provided in our supplementary materials. Some constrained optimization problems involve a known objective f (x). In that case, referring back to (9), we are done: Y (x, s) is WSNC (as in (10)) shifted by a known quantity f (x) + r(s). When ? (x, s) = Yf (x) + 1 W (x, s) is the weighted sum of a Gaussian Yf (x) is conditionally Gaussian, W 2? and WNCS variates, a problem that is again well-studied?see the supplementary material. 3.2

Slack-AL expected improvement

Evaluating EI at candidate (x, s) locations under the AL-composite involves working with EI(x, s) = n

n n } , given the current minimum y E (ymin ? Y (x, s)) I{Y (x,s)?ymin min of the AL over all n runs. 4

n n When f (x) is known, let wmin (x, s) = 2? (ymin ? f (x) ? r(s)) absorb all of the non-random quantities involved in the EI calculation. Then, with DW (?; x, s) denoting the distribution of W (x, s),

1 n E (wmin (x, s) ? W (x, s)) IW (x,s)?wmin (x,s) 2? n n Z wmin Z wmin (x,s) (x,s) 1 1 = DW (t; x, s)dt = DW (t; x, s)dt 2? ?? 2? 0

EI(x, s) =

(11)

n if wmin (x, s) ? 0 and zero otherwise. That is, the EI boils down to integrating the distribution function n of W (x, s) between 0 (since W is positive) and wmin (x, s). This is a one-dimensional definite integral that is easy to approximate via quadrature; details are in the supplementary material. Since W (x, s) is quadratic in the Yc (x) values, it is often the case, especially for smaller ?-values in later AL iterations, n that DW (t; x, s) is zero over most of [0, wmin (x, s)], simplifying numerical integration. However, this has deleterious impacts on search over (x, s), as we discuss in our supplement. When f (x) is n n unknown and Yf (x) is conditionally normal, let w ?min (s) = 2? (ymin ? r(s)). Then, n Z w?min (s) i

1 h n 1 ? EI(x, s) = E w ?min (s) ? W (x, s) IW = DW n (s) ? (t; x, s)dt. ? (x,s)?w ?min 2? 2? ??

Here the lower bound of the definite integral cannot be zero since Yf (x) may be negative, and thus ? (x, s) may have non-zero distribution for negative t-values. This can challenge the numerical W quadrature , although many library functions allow indefinite bounds. We obtain better performance by supplying a conservative finite lower bound, for example three standard deviations in Yf (x), in units of the penalty (2?), below zero: 6??f (x). Implementation details are in our supplement. 3.3

AL updates, optimal slack settings, and other implementation notes

The new slack-AL method is completed by describing when the subproblem (7) is deemed to be ?solved? (step 2 in Alg. 1), how ? and ? updated (steps 3?4). We terminate the BO search sub-solver after a single iteration as this matches with the spirit of EI-based search, whose choice of next location can be shown to be optimal, in a certain sense, if it is the final point being selected.

It also meshes well with an updating scheme analogous to that in steps 3?4: updating only when no actual improvement (in terms of constraint violation) is realized by that choice. That is, n o step 2: Let (xk , sk) approx. solve minx,s LA (x, s; ?k?1 , ?k?1) : (x, s1:m) ? B? 1 (cj (xk) + skj), for j = 1, . . . , m + p step 3: ?kj = ?k?1 + ?k?1 j k step 4: If c1:m (x) ? 0 and —cm+1:m+p (xk)— ? , set ?k=?k?1 ; else ?k = 21 ?k?1

Above, step 3 is the same as in Alg. 1 except without the ?max?, and with slacks augmenting the constraint values. The ?if? statement in step 4 checks for validity at xk , deploying a threshold ¿ 0 on equality constraints; further discussion of the threshold is deferred to Section 4, where we discuss progress metrics under mixed constraints. If validity holds at (xk , sk), the current AL iteration is deemed to have ?made progress? and the penalty remains unchanged; otherwise it is doubled. An alternate formulation may check —c1:m (xk) + sk1:m — ? . We find that the version in step 4, above, is cleaner because it limits sensitivity to the choice of threshold . In our supplementary material we recommend initial (?0 , ?0) values which are analogous to the original, non-slack AL settings. Optimal choice of slacks: The biggest difference between the original AL (3) and slack-AL (7) is that the latter requires searching over both x and s, whereas the former involves only x-values. In what follows we show that there are automatic choices for the s-values as a function of the corresponding x?s, keeping the search space d-dimensional, rather than d + m. For an observed cj (x) value, associated slack variables minimizing the AL (7) can be P obtained analytm m ically. Using the form of (9), observe that mins?Rm y(x, s) is equivalent to min s?R j=1 2?j ?sj + + + s2j + 2sj cj (x). For fixed x, this is strictly convex in s. Therefore, its unconstrained minimum can only be its stationary point, which satisfies 0 = 2?j ? + 2s?j (x) + 2cj (x), for j = 1, . . . , m. Accounting for the nonnegativity constraint, we obtain the following optimal slack as a function of x: s?j (x) = max {0, ??j ? ? cj (x)} , 5

j = 1, . . . , m.

(12)

Above we write s? as a function of x to convey that x remains a ?free? quantity in y(x, s? (x)). Recall that slacks on equality constraints are zero, sk (x) = 0, k = m + 1, . . . , m + p, for all x. In the blackbox c(x) setting, y(x, s? (x)) is only directly accessible at the data locations xi . At other x-values, however, the surrogates provide a useful approximation. When Yc (x) is (approximately) Gaussian it is straightforward to show that the optimal setting of the slack variables, solving mins?Rm E[Y (x, s)], are s?j (x) = max{0, ??j ? ? ?cj (x)}, i.e., the same as (12) with a prediction + ?cj (x) for Ycj (x), the unknown cj (x) value. Again, slacks on the equality constraints are set to zero. Other criteria can be used to choose slack variables. Instead of minimizing the mean of the composite, one could maximize the EI. In our supplementary material we explain how this is of dubious practical value, being more computationally intensive and providing near identical results in practice. Implementation notes: Code supporting all methods in this manuscript is provided in two opensource R packages: laGP [8] and DiceOptim [19], both on CRAN [22]. Implementation details vary somewhat across those packages, due primarily to particulars of

their surrogate modeling capability and how they search the EI surface. For example, laGP can accommodate a smaller initial design size because it learns fewer parameters (i.e., has fewer degrees of freedom). DiceOptim uses a multi-start search procedure for EI, whereas laGP deploys a random candidate grid, which may optionally be ?finished? with an L-BFGS-B search. Nevertheless, their qualitative behavior exhibits strong similarity. Both packages also implement the original AL scheme (i.e., without slack variables) updated (6) for mixed constraints. Further details are provided in our supplementary material.

4

Empirical comparison

Here we describe three test problems, each mixing challenging elements from traditional unconstrained blackbox optimization benchmarks, but in a constrained optimization format. We run our optimizers on these problems 100 times under random initializations. In the case of our GP surrogate comparators, this initialization involves choosing random space-filling designs. Our primary means of comparison is an averaged (over the 100 runs) measure of progress defined by the best valid value of the objective for increasing budgets (number of evaluations of the blackbox), n. In the presence of equality constraints it is necessary to relax this definition somewhat, as the valid set may be of measure zero. In such cases we choose a tolerance ? 0 and declare a solution to be ?valid? when inequality constraints are all valid, and when —hk (x)— ¡ for all k = 1, . . . , p. In our figures we choose = 10?2 ; however, the results are similar under stronger thresholds, with a higher variability over initializations. As finding a valid solution is, in itself, sometimes a difficult task, we additionally report the proportion of runs that find valid and optimal solutions as a function of budget, n, for problems with equality (and mixed) constraints. 4.1

An inequality constrained problem

We first revisit the ?toy? problem from [9], having a 2d input space limited to the unit cube, a (known) linear objective, with sinusoidal and quadratic inequality constraints (henceforth the LSQ problem; see the supplementary material for details). Figure 1 shows progress over repeated solves with a maximum budget of 40 blackbox evaluations. The left-hand plot in Figure 1 tracks the average best valid value of the objective found over the iterations, using the progress metric described above. Random initial designs of size n = 5 were used, as indicated by the vertical-dashed gray line. The solid gray lines are extracted from a similar plot from [9], containing both AL-based comparators, and several from the derivative-free optimization and BO literatures. The details are omitted here. Our new ALBO comparators are shown in thicker colored lines; the solid black line is the original AL(BO)-EI comparator, under a revised (compared to [9]) initialization and updating scheme. The two red lines are variations on the slack-AL algorithm under EI: with (dashed) and without (solid) L-BFGS-B optimizing EI acquisition at each iteration. Finally, the blue line is PESC [10], using the Python library available at https://github.com/HIPS/Spearmint/tree/PESC. The take-home message from the plot is that all four new methods outperform those considered by the original ALBO paper [9]. Focusing on the new comparators only, observe that their progress is nearly statistically equivalent during the

first 20 iterations. However, in the latter iterations stark distinctions emerge, with Slack-AL+optim and PESC, both leveraging L-BFGS-B subroutines, outperforming. This 6

?4 log utility gap

?5

1.1 1.0 0.9

?7

0.8 0.6

0.7

best valid objective (f)

Original AL Slack AL Slack AL + optim PESC

?6

1.2

Initial Design Gramacy, et al. (2016)

0

10

20

30

40

20

25

blackbox evaluations (n)

30

35

40

blackbox evaluations (n)

Figure 1: Results on the LSQ problem with initial designs of size n = 10. The left panel shows the best valid value of the objective over the first 40 evaluations, whereas the right shows the log utility-gap for the second 20 evaluations. The solid gray lines show comparators from [9]. discrepancy is more easily visualized in the right panel with a so-called log ?utility-gap? plot [10], tracking the log difference between the theoretical best valid value and those found by search.

4.2

Mixed inequality and equality constrained problems

1.0 0.8 0.6 0.4

2

0.0

1

0.2

proportion of valid and solved runs

nlopt/140 NOMAD?P1/15 NOMAD?AL?P1/15 NOMAD?AL?PBP1/15

3

4

Original AL Slack AL Slack AL + optim EFI

0

best valid (1e?2 for equality) objective (f)

Next consider a problem in four input dimensions with a (known) linear objective and two constraints. The first inequality constraint is the so-called ?Ackley? function in d = 4 input dimensions. The second is an equality constraint following the so-called ?Hartman 4-dimensional function?. Our supplementary material provides a full mathematical specification. Figure 2 shows two views into

0
10
20
30
40
50
0
blackbox evaluations (n)
10
20
30
40
50
blackbox evaluations (n)

Figure 2: Results on the Linear-Ackley-Hartman mixed constraint problem. The left panel shows a progress comparison based on laGP code with initial designs of size n = 10. The x-scale has been divided by 140 for the nlopt comparator. A value of four indicates that no valid solution has been found. The right panel shows the proportion of valid (thin lines) and optimal (thick lines) solutions for the EFI and ?Slack AL + optim? comparators. progress on this problem. Since it involves mixed constraints, comparators from the BO literature are scarce. Our EFI implementation deploys the (?h, h) heuristic mentioned in the introduction. As representatives from the nonlinear optimization literature we include nlopt [11] and three adapted NOMAD [13] comparators, which are detailed in our supplementary material. In the left-hand plot we can see that our new ALBO comparators are the clear winner, with an L-BFGS-B optimized EI search under the slack-variable AL implementation performing exceptionally well. The nlopt and NOMAD comparators are particularly poor. We allowed those to run up to 7000 and 1000 iterations, respectively, and in the plot we scaled the x-axis (i.e., n) to put them on the same scale as the others. 7

The right-hand plot provides a view into the distribution of two key aspects of performance over the MC repetitions. Observe that ?Slack AL + optim? finds valid values quickly, and optimal values not much later. Our adapted EFI is particularly slow at converging to optimal (valid) solutions.

1.0 0.8 0.6 0.4
1.0 0.5 0.0
0.0
0.2
proportion of valid and solved runs
nlopt/46 NOMAD?P1/8 NOMAD?AL?P1/8 NOMAD?AL?PBP1/8

1.5
2.0
Original AL Slack AL Slack AL + optim EFI
?0.5
best valid (1e?2 for equality) objective (f)

Our final problem involves two input dimensions, an unknown objective function (i.e., one that must be modeled with a GP), one inequality constraint and two equality constraints. The objective is a centered and re-scaled version of the ?Goldstein?Price? function. The inequality constraint is the sinusoidal constraint from the LSQ problem [Section 4.1]. The first equality constraint is a centered ?Branin? function, the second equality constraint is taken from [16] (henceforth the GBSP problem). Our supplement contains a full mathematical specification. Figure 3 shows our results on

0
50
100
150
0
blackbox evaluations (n)
50
100
150
blackbox evaluations (n)

Figure 3: Results on the GBSP problem. See Figure 2 caption. this problem. Observe (left panel) that the original ALBO comparator makes rapid progress at first, but dramatically slows for later iterations. The other ALBO comparators, including EFI, converge much more reliably, with the ?Slack AL + optim? comparator leading in both stages (early progress and ultimate convergence). Again, nlopt and NOMAD are poor, however note that their relative comparison is reversed; again, we scaled the x-axis to view these on a similar scale as the others. The right panel shows the proportion of valid and optimal solutions for ?Slack AL + optim? and EFI. Notice that the AL method finds an optimal solution almost as quickly as it finds a valid one?both substantially faster than EFI.

5 Conclusion

The augmented Lagrangian (AL) is an established apparatus from the mathematical optimization literature, enabling objective-only or bound-constrained optimizers to be deployed in settings with constraints. Recent work involving Bayesian optimization (BO) within the AL framework (ALBO) has shown great promise, especially toward obtaining global solutions under constraints. However, those methods were deficient in at least two respects. One is that only inequality constraints could be supported. Another was that evaluating the acquisition function, combining predictive mean and variance information via expected improvement (EI), required Monte Carlo approximation. In this paper we showed that both drawbacks could be addressed via a slack-variable

reformulation of the AL. Our method supports inequality, equality, and mixed constraints, and to our knowledge this updated ALBO procedure is unique in the BO literature in its applicability to the most general mixed constraints problem (1). We showed that the slack ALBO method outperforms modern alternatives in several challenging constrained optimization problems. Acknowledgments We are grateful to Mickael Binois for comments on early drafts. RBG is grateful for partial support from National Science Foundation grant DMS-1521702. The work of SMW is supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research under Contract No. DE-AC02-06CH11357. The work of SLD is supported by the Natural Sciences and Engineering Research Council of Canada grant 418250. 8

2 References

[1] C. Audet, J. Dennis, Jr., D.W. Moore, A. Booker, and P.D. Frank. Surrogate-model-based method for constrained optimization. In AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, 2000. [2] D. Bertsekas. Constrained Optimization and Lagrange Multiplier Methods. Academic Press, New York, NY, 1982. [3] G. E. P. Box and N. R. Draper. Empirical Model Building and Response Surfaces. Wiley, Oxford, 1987. [4] P. Boyle. Gaussian Processes for Regression and Optimization. PhD thesis, Victoria University of Wellington, 2007. [5] E. Brochu, V. M. Cora, and N. de Freitas. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. Technical report, University of British Columbia, 2010. arXiv:1012.2599v1. [6] J. R. Gardner, M. J. Kusner, Z. Xu, K. W. Weinberger, and J. P. Cunningham. Bayesian optimization with inequality constraints. In Proceedings of the 31st International Conference on Machine Learning, volume 32. JMLR, W&CP, 2014. [7] M. A. Gelbart, J. Snoek, and R. P. Adams. Bayesian optimization with unknown constraints. In Uncertainty in Artificial Intelligence (UAI), 2014. [8] R. B. Gramacy. laGP: Large-scale spatial modeling via local approximate Gaussian processes in R. Journal of Statistical Software, 72(1):1?46, 2016. [9] R.B. Gramacy, G.A. Gray, S. Le Digabel, H.K.H. Lee, P. Ranjan, G. Wells, and S.M. Wild. Modeling an augmented Lagrangian for blackbox constrained optimization. Technometrics, 58:1?11, 2016. [10] J.M. Hern?ndez-Lobato, M. A. Gelbart, M. W. Hoffman, R. P. Adams, and Z. Ghahramani. Predictive entropy search for Bayesian optimization with unknown constraints. In Proceedings of the 32nd International Conference on Machine Learning, volume 37. JMLR, W&CP, 2015. [11] S. G. Johnson. The NLopt nonlinear-optimization package, 2014. via the R package nloptr. [12] D. R. Jones, M. Schonlau, and W. J. Welch. Efficient global optimization of expensive black box functions. J. of Global Optimization, 13:455?492, 1998. [13] S. Le Digabel. Algorithm 909: NOMAD: Nonlinear Optimization with the MADS algorithm. ACM Transactions on Mathematical Software, 37(4):44:1?44:15. doi: 10.1145/1916461.1916468. [14] J. Mockus. Bayesian Approach to Global Optimization: Theory and Ap-

plications. Springer, 1989. [15] J. Nocedal and S. J. Wright. Numerical Optimization. Springer, second edition, 2006. [16] J. Parr, A. Keane, A Forrester, and C. Holden. Infill sampling criteria for surrogate-based optimization with constraint handling. Engineering Optimization, 44:1147?1166, 2012. [17] V. Picheny. A stepwise uncertainty reduction approach to constrained global optimization. In Proceedings of the 7th International Conference on Artificial Intelligence and Statistics, volume 33, pages 787?795. JMLR W&CP, 2014. [18] V. Picheny, D. Ginsbourger, and T. Krityakierne. Comment: Some enhancements over the augmented lagrangian approach. Technometrics, 58(1):17?21, 2016. [19] V. Picheny, D. Ginsbourger, O. Roustant, with contributions by M. Binois, C. Chevalier, S. Marmin, and T. Wagner. DiceOptim: Kriging-Based Optimization for Computer Experiments, 2016. R package version 2.0. [20] M. J. Sasena. Flexibility and Efficiency Enhancement for Constrained Global Design Optimization with Kriging Approximations. PhD thesis, University of Michigan, 2002. [21] M. Schonlau, W. J. Welch, and D. R. Jones. Global versus local search in constrained optimization of computer models. Lecture Notes-Monograph Series, pages 11?25, 1998. [22] R Development Core Team. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Aus., 2004. URL http://www.R-project.org. ISBN 3-900051-00-3. [23] J. Snoek, H. Larochelle, and R. P. Adams. Bayesian optimization of machine learning algorithms. In Neural Information Processing Systems (NIPS), 2012.

9