

TRUST REGION BASED STOCHASTIC OPTIMIZATION WITH ADAPTIVE RESTART: A FAMILY OF GLOBAL OPTIMIZATION ALGORITHMS

Logan Mathesen
Giulia Pedrielli

Szu Hui Ng

School of Computing-IDSE
Arizona State University
699 S. Mill Ave.
Tempe, Az 85281, USA

Industrial & Systems Engineering Management
National University of Singapore
1 Engineering Drive, 4
SG 258127, SINGAPORE

ABSTRACT

The field of simulation optimization has seen algorithms proposed for local optimization, drawing upon different locally convergent search methods. Similarly, there are numerous global optimization algorithms with differing strategies to achieve convergence. In this paper, we look specifically into meta-model based algorithms that stochastically drive global search through an optimal sampling criteria evaluated over a constructed meta-model of the predicted response considering the uncertainty of the response. We propose Trust Region Based Optimization with Adaptive Restart (TBOAR), a family of algorithms that dynamically restarts a trust region based search method via an optimal sampling criteria derived upon a meta-model based global search approach. Additionally, we propose a new sampling criteria to reconcile undesirable adaptive restart trajectories. This paper presents preliminary results showing the advantage of the proposed approach over the benchmark Efficient Global Optimization algorithm, focusing on a deterministic black box simulator with a d -dimensional input and a one-dimensional response.

1 INTRODUCTION

In this paper, we consider single objective simulation optimization problems. We assume a d -dimensional compact solution space, $\mathbb{X} \subset \mathbb{R}^d$, and a deterministic objective function $y(\mathbf{x}) : \mathbb{X} \rightarrow \mathbb{R}$ that can only be observed as an outputted response of a deterministic black box simulator; where \mathbf{x} is a single point in \mathbb{X} . We desire to identify the global minimum of $y(\mathbf{x}) : \mathbb{X} \rightarrow \mathbb{R}$. Formally:

$$(P) : \min y(\mathbf{x}) \\ s.t. \quad \mathbf{x} \in \mathbb{X}$$

The proposed solution approach is a blend of global and local stochastic optimization methods. The Trust Region Based Optimization with Adaptive Restart (TBOAR) family of algorithms drives optimization with local searches, while relying upon globally convergent methods as a guide for adaptive restart. The formal integration proposed in this paper combines a derivative based local search and a meta-model based global sampling algorithm.

1.1 Background

Global stochastic optimization algorithms can generally be classified as either a direct or a meta-model based search method. Direct search such as stochastic approximation methods (Yin and Kushner 2003), or COMPASS (Xu et al. 2010, Hong and Nelson 2006), directly call the simulator as a means of objective function estimation. These methods are generally held back when observing a simulation run requires high

computational effort. In these scenarios, meta-model (or surrogate) based methods offer a computational compromise estimating a response model from a few simulator observations. The estimated model enables a predicted performance evaluation of the function over the entire space without the need to call the simulator. Efficient Global Optimization (EGO, Jones et al. 1998) and Two Stage Sequential Optimization (TSSO, Quan et al. (2013)) represent two meta-model based algorithms which both use the Kriging meta-model form. Kriging has been shown to be particularly successful in computer experiments as its closed form predictors allow closed form computation of sampling criteria, like the Expected Improvement (EI) function. In this regard, we explore the use of improvement functions as a foundation for adaptive restart.

For relative context algorithms combining local and global search have been proposed, such as Industrial Strength COMPASS (ISC) and BEESE (Xu et al. (2010), Andradóttir and Prudius (2009)). Exploratory and exploitative behavior are structured in two phases. First a global search samples the entire space identifying promising regions to sample further, then a local search exploits the identified regions. The algorithm is typically then terminated with a clean up or estimation phase.

1.2 Motivation

The two pertinent trade-offs between global and local searches that motivate TBOAR are: (1) the typical differences in data management, and (2) finite-time versus asymptotic algorithm performance. Classic local searches are often "memoryless" procedures that make decisions based on purely local information. Global searches typically accumulate all observed response data and compute sampling criteria based on sample observation paths, which is then used to make sampling decisions on which data should be accumulated for subsequent model fit. This complete information storage leads global search techniques such as Efficient Global Optimization (Jones et al. 1998) or the Two stage Sequential Optimization algorithm (Quan et al. 2013) to have appealing asymptotic performance guarantees. Nevertheless, the more myopic local search techniques often provide stronger finite-time performance in finding a good solution providing milestones of improvement, while guarantees for finite-time performance of global methods remains a hard problem (Zabinsky et al. 2010).

Several empirical studies have been conducted on the nature of local optimization and the powerful benefits that random restarts can have on the performance of such methods. This observation has promoted work in the dynamic generation of restarting points trying to address two main questions: (1) when to restart a local algorithm, (2) where to restart. The results of Dynamic Multistart Sequential Search presented by Zabinsky et al. (2010) illustrate the power that restarts can have on local search performance. In their final concluding remarks, the authors acknowledge that the DMSS framework ignores function values across separate runs of the local search. This acknowledgment motivates the formal integration of information over multiple local search runs through the use of a global meta-model, which we propose in this contribution.

1.3 Contributions

We propose the TBOAR family of algorithms to fill the current research gap and give a means of integration of local and global searches through sequential globally informed adaptive restarts of local algorithms. TBOAR integrates serial local searches through a dynamic global model capturing local minimums and adaptively directing restarts. We interpret adaptive restarts as a global optimization sampling problem.

It is of note that the TBOAR family of algorithms takes motivation from the GSTAR algorithm previously proposed in Pedrielli and Ng (2016). GSTAR serves as a novel work, integrates information generated from a global and numerous locally estimated models into a single ensemble model to guide starting centroids for local trust region models. The local models created in GSTAR are constructed through a Gaussian process fit to a space filling design across a trust region of arbitrary size around an arbitrary centroid \mathbf{x}_c . In contrast, TBOAR employs a pure local search that does not seek to describe the surface within the trust region, as GSTAR's local models do, but rather to identify a local minimum \mathbf{x}_l^* within a neighborhood of size $\varepsilon > 0$, such that $\mathbf{x}_l^* \in \mathbb{X} : y(\mathbf{x}_l^*) < y(\mathbf{x}), \forall \mathbf{x} \in \mathcal{N}_\varepsilon(\mathbf{x}_l^*)$. This difference in local search methods drives

TBOAR algorithms to converge to local minima quicker than GSTAR. While the local search employed in TBOAR is less sophisticated in comparison to algorithms such as STRONG (Chang, Hong, and Wan 2013), the trust region based optimization closely follows the testing procedures laid out in STRONG. All TBOAR algorithms presented employ a trust region based subproblem optimization routine, but for breadth, a locally convergent simplex hill-climber with adaptive restarts driven by global meta-modeling would fit well in the TBOAR framework.

The overall target of the proposed TBOAR family of algorithms is to monopolize the relative efficiency enjoyed by locally convergent methods, while simultaneously modeling salient response observations to estimate a global response surface to adaptively guide restarts of the local search.

2 TBOAR OVERVIEW

TBOAR refers to a family of algorithms comprised of three main components: (1) an estimated global meta-model, (2) a trust region based local optimization, and (3) a sampling criterion for the restart of the local search. In this section, we will discuss the role and implementation of each of the three components, along with two alternatives for both the local trust based search and optimal sampling criteria. In Section 3, we present the numerical results from experimentations over four instances of TBOAR algorithms. The main steps and flow of a TBOAR algorithm is summarized in Figure 1 and outlined below:

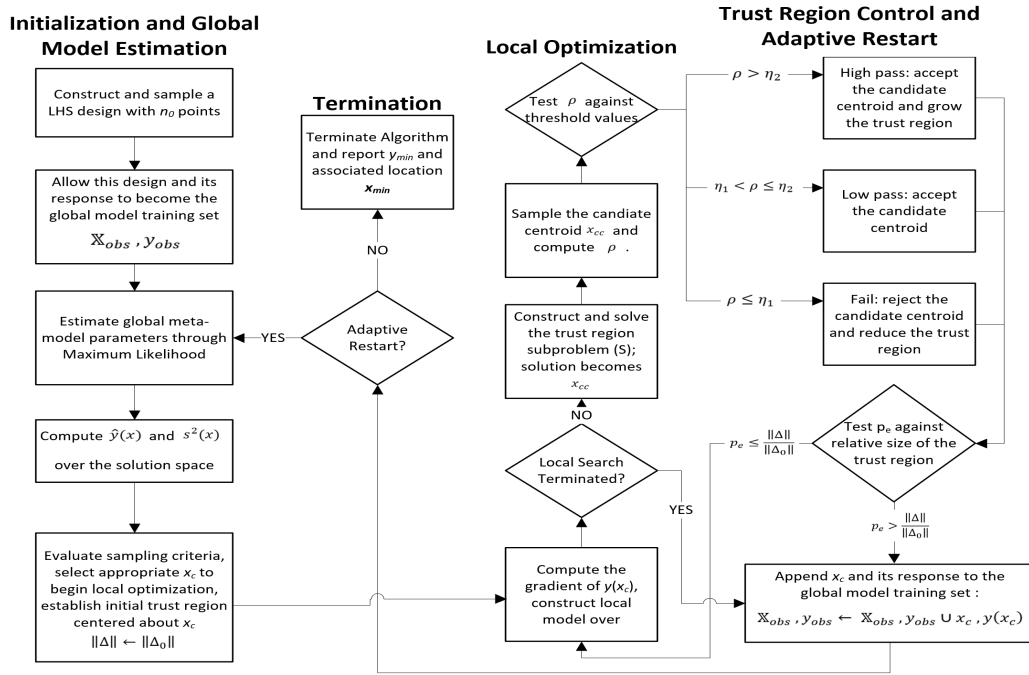


Figure 1: Flow of TBOAR algorithm framework.

1. Initialization

- Construct and observe an initial design \mathbb{X}_{obs} of n_0 points over the solution space, $\mathbb{X}_{obs} \subset \mathbb{X}$. Estimate the global meta-model to these n_0 points, and produce $\hat{\mathbf{y}}(\mathbf{x})$ and $\mathbf{s}^2(\mathbf{x})$. If Cross validation is passed, go to Step 2c. If Cross validation is failed, re-sample the initial design.

2. Adaptive Restart

- Test whether or not to exit the current trust region.
- If exit from the trust region is granted, go to Step 4. Otherwise, go to Step 3.
- Use $\hat{\mathbf{y}}(\mathbf{x})$ and $\mathbf{s}^2(\mathbf{x})$, and derive the sampling criteria over all \mathbb{X} .

- (d) Select the previously unobserved location that best optimizes the computed sampling criteria, this is the centroid \mathbf{x}_c where the local search will restart. Construct an initial trust region Δ of size $\|\Delta\| \leftarrow \|\Delta_0\|$ about the centroid \mathbf{x}_c . Proceed to local optimization, Step 3.
3. **Local Optimization**
 - (a) Formulate the local model constrained to the current trust region.
 - (b) Solve the trust region subproblem (S); the optimal solution becomes the candidate centroid \mathbf{x}_{cc} .
 - (c) Evaluate $y(\mathbf{x}_{cc})$, and adjust the size of the trust region appropriately; go to Step 2.
4. **Global Meta-Model Estimation**
 - (a) Update the global set of sampled points \mathbb{X}_{obs} and \mathbf{y}_{obs} with the final observed centroid location and its associated response $(\mathbf{x}_c, y(\mathbf{x}_c))$.
 - (b) Estimate the global response parameters by fitting the desired meta-model to $\{\mathbb{X}_{obs}, \mathbf{y}_{obs}\}$.
 - (c) Generate $\hat{\mathbf{y}}(\mathbf{x})$ and $s^2(\mathbf{x})$ throughout the entire solution space, $\mathbf{x} \in \mathbb{X}$.
 - (d) Go to Step 2c.

Section 2.1 presents the Gaussian process meta-model pertinent to Step 4, section 2.2 presents Local Optimization (Step 3), and finally section 2.3 presents alternative sampling criteria for adaptive restarts.

2.1 Global Meta-Model Estimation

A common approach in the literature for global optimization entails the estimation of meta-models. Within this class, Kriging models with Gaussian correlation have found favor in the research community (Jones et al. (1998), Huang et al. (2006), Kleijnen (2009)). This is largely due to the flexibility and concise closed-form expressions for both expectation and predictive mean squared error of a Gaussian process model. In the TBOAR algorithmic framework presented, we use Gaussian process models as our global meta-model structure. Latin hypercube sampling is used to generate initial global Gaussian process models.

2.1.1 Structure of the Gaussian Process Model

Using a Gaussian process model we attempt to fit n training pairs of simulation input-output observations, using a univariate output response $y_i \in \mathbb{R}$ and a d -dimensional input $\mathbf{x}_i \in \mathbb{R}^d$. We assume that the response is adequately modeled as $y(\mathbf{x}) = \mu + z(\mathbf{x})$ where μ is the overall process mean, and $z(\mathbf{x})$ is the Gaussian process, we formally assume that $z(\mathbf{x}) \sim N(0, \sigma_z^2)$. Comprehensive Gaussian process details can be found in Santner et al. (2013), and Cressie and Cassie (1993).

The specific model that is employed as the global emulator is the Stationary Ordinary Kriging (OK) model, yielding a constant mean with exactly interpolating behavior to the n training pairs in \mathbb{X}_{obs} and \mathbf{y}_{obs} . We use the Gaussian correlation function, a member of the power exponential correlation family: $R_{ij} = \prod_{k=1}^d e^{(-\theta_k |\mathbf{x}_{ik} - \mathbf{x}_{jk}|)^2}$ where θ is a d -dimensional vector of smoothing hyper parameters.

We follow maximum likelihood method for estimation of OK model parameters and hyper-parameters- μ , σ_z^2 , and θ . The closed form estimators of the model parameters μ and σ_z^2 are: $\hat{\mu} = \frac{\mathbf{1}_n^T \mathbf{R}^{-1} \mathbf{y}_{obs}}{\mathbf{1}_n^T \mathbf{R}^{-1} \mathbf{1}_n}$ and $\hat{\sigma}_z^2 = \frac{(\mathbf{y}_{obs} - \mathbf{1}_n \hat{\mu})^T \mathbf{R}^{-1} (\mathbf{y}_{obs} - \mathbf{1}_n \hat{\mu})}{n}$. The best linear unbiased predictor (BLUP) associated with the OK model at any location $\mathbf{x} \in \mathbb{X}$ is:

$$E[y(\mathbf{x})] = \hat{\mathbf{y}}(\mathbf{x}) = \hat{\mu} + \mathbf{r}^T \mathbf{R}^{-1} (\mathbf{y}_{obs} - \mathbf{1}_n \hat{\mu}) \quad (1)$$

with a corresponding predictive error of:

$$Var[\hat{\mathbf{y}}(\mathbf{x})] = s^2(\mathbf{x}) = \sigma_z^2 \left(1 - \mathbf{r}^T \mathbf{R}^{-1} \mathbf{r} + \frac{(\mathbf{1}_n - \mathbf{1}_n^T \mathbf{R}^{-1} \mathbf{r})^2}{\mathbf{1}_n^T \mathbf{R}^{-1} \mathbf{1}_n} \right) \quad (2)$$

To complete the formal definition of the OK model predictor, we define \mathbf{r} as the n -dimensional vector of correlations between the systematic error at the predictive location \mathbf{x} and the observed model error at all previously observed locations \mathbb{X}_{obs} , i.e. $r_i = Corr(\mathbf{x}, \mathbf{x}_i) : \mathbf{x}_i \in \mathbb{X}_{obs}$.

2.2 Local Optimization

In this work, the local search is based upon the construction of a trust region, a d -dimensional area Δ centered about an arbitrary centroid \mathbf{x}_c . The trust region carves out a subspace $\Delta \subset \mathbb{X}$, enabling the construction of an effective model over a reduced decision space Δ . Essentially the trust region subproblem becomes that of: (1) estimation of a local model within the trust region, (2) optimization over that model, and (3) control of the trust region in size and location to find a more promising region of \mathbb{X} and an appropriate trust region size, allowing for a better fit of the simple local model.

The use of trust regions was a logical choice for TBOAR as a candidate local search method to incorporate into a global-local search method. An analogous algorithmic framework to TBOAR is the basic branch and bound algorithm. Branch and bound works to iteratively solve an integer program through a series of easier to solve linear program subproblems. If we assume our trust region is small enough such that the response surface is convex over it, then iteratively solving local trust region searches to termination, coupled with an adaptive restart procedure will, in infinite time, converge to the true globally optimal solution. Where branch and bound deals with finite, discrete, and mutually exclusive subproblems; TBOAR iteratively solves continuous, and non-mutually exclusive trust region subproblems.

2.2.1 Estimation and Optimization of Local Model

In this work, the specific local models constructed over the trust region Δ , are first and second order Taylor expansions centered around a trust region centroid. While this is not the most sophisticated local model, if we assume a continuous surface and relatively small trust regions, then a Taylor expansion is not ill-suited. Formally, the linear ($\hat{y}_L(\mathbf{x})$, equation (3)) and quadratic ($\hat{y}_Q(\mathbf{x})$, equation (4)) models are:

$$\hat{y}_L(\mathbf{x}) = y(\mathbf{x}_c) + \nabla y(\mathbf{x}_c)(\mathbf{x}_c - \mathbf{x}) \quad (3)$$

$$\hat{y}_Q(\mathbf{x}) = y(\mathbf{x}_c) + \nabla y(\mathbf{x}_c)(\mathbf{x}_c - \mathbf{x}) + \frac{1}{2} \nabla^2 y(\mathbf{x}_c)(\mathbf{x}_c - \mathbf{x})^2 \quad (4)$$

Because the function response can only be observed through a deterministic black box simulator, the response gradient $\nabla y(\mathbf{x}_c)$, and Hessian $\nabla^2 y(\mathbf{x}_c)$ are calculated through a local finite differencing sampling strategy. Establishing an initial trust region size of $\|\Delta_0\|$, we assume that Δ is a compact set. Formulating the trust region subproblem (S); the linear or quadratic constrained optimization problem (S) becomes:

$$(S) \min \quad \hat{y}_i(\mathbf{x}), \quad i = \{L, Q\} \quad (5) \\ s.t. \quad \mathbf{x} \in \Delta$$

The optimal solution to (S) becomes the candidate centroid \mathbf{x}_{cc} for the next trust region subproblem. Testing of the identified optimal solution \mathbf{x}_{cc} , indicates whether the current local model and subproblem (S) are trustworthy, i.e. if the local model over the constrained trust region: $\hat{y}_i(\mathbf{x}) : \mathbf{x} \in \Delta$ is effective. Please note there is no mixing of models, model choice for the local search is an *a priori* decision of the user, although there may be future opportunities to dynamically select local model forms.

The quadratic model allows estimation of curvature over the trust region, and offers the ability to predict a unique local minimum as an interior point of Δ . Alternatively the linear model does not offer this capability, and trivially reduces to an exercise in trust region control. However, a leveling trade-off between the two models is the number of observations required to fit the local model. In d -dimensions, the linear model requires $d + 1$ observations to compute $\nabla y(\mathbf{x}_c)$; where the quadratic model requires $2d + 1$ observations to additionally estimate $\nabla^2 y(\mathbf{x}_c)$. The assumption that observing a simulator response is the most expensive operation (by at least two orders of magnitudes) makes the need to calculate finite differences burdensome on the finite time performance on all of our experimental TBOAR algorithms. In high dimensionality, polynomial models, optimally designed experiments and bridge designs offer derivative free alternatives that may be more computationally efficient to model trust regions, see (Conn et al. 2009) for examples.

2.2.2 Trust Region Control: Translation and Reshaping

Control of the trust region is arguably just as influential on the quality of trust region based optimization as the formulation and solving of the subproblem (S). There are four parameters necessary for basic trust region control: initial size of trust region, $\|\Delta_0\|$, rate of trust region growth, $\gamma > 1$, rate of trust region reduction, $\omega \in (0, 1)$, and decision criteria to determine how to control trust regions. Trust region control serves to move the centroid and trust region and exploit promising regions in \mathbb{X} . Given infinite time Δ reduces to a single point x_c the local minimum defined as:

$$\mathbf{x}_l^* \in \mathbb{X} : y(\mathbf{x}_l^*) \leq y(\mathbf{x}), \forall \mathbf{x} \in \mathcal{N}_\varepsilon(\mathbf{x}_l^*), \varepsilon > 0 \quad (6)$$

The trust region literature offers little consensus on initial trust region size, optimal rate of growth, or reduction of a trust region. Tests such as the sufficient reduction (SR), and ratio-comparison (RC) are common decision tools. Both tests require an observation of response at the candidate centroid \mathbf{x}_{cc} ; a comparison test statistic is then constructed from the true response $y(\mathbf{x}_{cc})$ and the locally predicted response $\hat{y}_i(\mathbf{x}_{cc})$. The Ratio-Comparison (RC) test statistic is used in TBOAR as the decision criteria for trust region control and is implemented similarly to Chang et al. (2013). The RC test statistic is compared with two user specified threshold values: η_1, η_2 . The RC test statistic ρ is calculated as:

$$\text{RC Test: } \rho = \frac{y(\mathbf{x}_c) - y(\mathbf{x}_{cc})}{\hat{y}_i(\mathbf{x}_c) - \hat{y}_i(\mathbf{x}_{cc})} \quad (7)$$

$\rho > 1$ implies that the current local model $\hat{y}_i(\mathbf{x}) : \mathbf{x} \in \Delta$ is effective in exploiting the trust region. $\rho < 0$ indicate that the chosen model form is inadequate at describing the current trust region. Comparing ρ with the threshold values, the trust region is controlled in one of the three following ways:

- Case 1 If $\rho \leq \eta_1$ then there is no movement of the centroid and no translation of the trust region, $\mathbf{x}_c \leftarrow \mathbf{x}_c$, the trust region is reduced by a factor to improve model fit, $\|\Delta\| \leftarrow \|\Delta\| \cdot \omega$.
- Case 2 If $\eta_1 < \rho \leq \eta_2$, we conditionally accept the candidate centroid and translate the trust region, $\mathbf{x}_c \leftarrow \mathbf{x}_{cc}$. There is no growth or reduction in trust region size, $\|\Delta\| \leftarrow \|\Delta\|$.
- Case 3 If $\eta_2 < \rho$ then we accept the candidate centroid and move the trust region, $\mathbf{x}_c \leftarrow \mathbf{x}_{cc}$. Due to our confidence in our model's ability to describe the space, we grow the trust region size, $\|\Delta\| \leftarrow \|\Delta\| \cdot \gamma$.

Under *Case 1*, we immediately return to formulating subproblem (S) with the same local model estimation but under more restrictive trust region constraints. Under *Case 2* or *Case 3* we must recompute $\nabla y(\mathbf{x}_c)$ and $\nabla^2 y(\mathbf{x}_c)$ in order to estimate $\hat{y}_i(\mathbf{x}) : \mathbf{x} \in \Delta$. With the local model re-estimated we reformulate the optimization subproblem (S) and continue to iterate until one of two local search termination criteria are met. The termination criteria are: $\|\nabla y(\mathbf{x}_c)\| < \varepsilon_1$, and $\|\Delta\| < \varepsilon_2$ ($\varepsilon_{1,2} > 0$). Upon exiting the trust region and terminating the local search, we append the last centroid and its associated response $\mathbf{x}_c, y(\mathbf{x}_c)$ to the training data set $\mathbb{X}_{obs}, \mathbf{y}_{obs}$ for the global OK meta-model.

We assume that the last centroid \mathbf{x}_c is relatively close to a true local minimum. Due to minimums being the points with furthest deviation from the mean and the nature of exact Kriging interpolation, we argue that local minimums are those points which best support Kriging models and best represent the behavior of a region, especially when optimization is goal. It is of note that with infinite time, density will be achieved, and every input location will be considered as a centroid; this guides our intuition that TBOAR is globally convergent. It should be apparent, in terms of simulation calls, how costly trust region search can become when termination is based solely on gradient and trust region size. In TBOAR, we compensate by monitoring local searches and forcing adaptive restarts prior to gradient conditions being met.

2.3 Adaptive Restart

As highlighted in Zabinsky et al. (2010), to perform random restarts we need to address two questions: (1) when to restart, and (2) where to restart.

Concerning the first point it is clear that when a local search terminates due to gradient conditions we will need to restart. Additionally, we also apply an adaptive restart criteria to monitor the progress of local searches and dynamically force local search restarts. The decision is made based upon an indicator chosen to be the proportional size of current trust region to the initial trust region size, $\frac{\|\Delta\|}{\|\Delta_0\|}$. We further the dynamic nature of the criteria by sampling a number $p_e \sim U(0,1)$, and then comparing p_e with the current trust region size ratio. If $p_e \leq \frac{\|\Delta\|}{\|\Delta_0\|}$ then we remain in the current trust region and continue the local search by formulating the next subproblem. Otherwise, the trust region is exited and we continue to the second question of where to restart. The question of when to restart is answered by an adaptive process where the trust region ratio acts as a surrogate for the current performance of the local search. A large $\frac{\|\Delta\|}{\|\Delta_0\|}$ ratio indicates that the local search is improving and that it should not be terminated prematurely. Alternatively, a ratio tending to zero means the improvement rate of the local search has slowed and an adaptive restart should be applied soon. Figure 2 shows the progression of a sample path of the stochastic adaptive restart process. Evolution through the sample path is a consequence of trust region control of growth and reduction, following the ratio comparison test. In Figure 2 restarts are indicated by the right facing triangles, note these restarts always re-initialize the trust region ratio, $\|\Delta\|/\|\Delta_0\| = 1$. The top section filled by slanted lines indicates where $\|\Delta\|/\|\Delta_0\| \geq 1$ where an adaptive restart will never be applied; the bottom section filled by circles indicates $\|\Delta\|/\|\Delta_0\| < 0.5$ where a restart becomes much more likely.

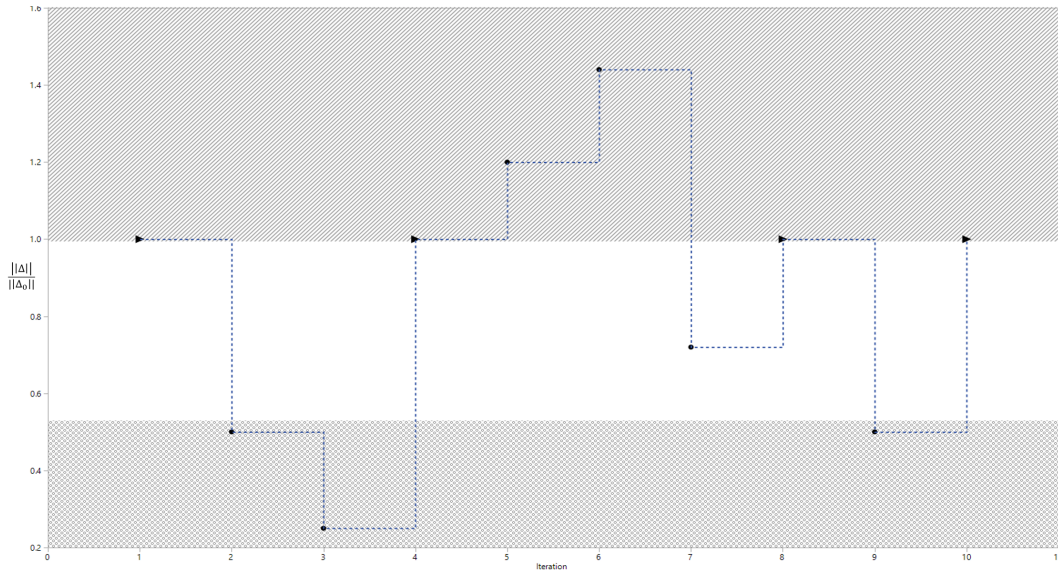


Figure 2: Sample adaptive restart path, 4 local search restarts are indicated by the 4 triangles at $\frac{\|\Delta\|}{\|\Delta_0\|} = 1$.

Concerning the second issue of where to restart, the next trust region centroid \mathbf{x}_c is selected as the location that maximizes some sampling criteria applied over the uncertain expected response surface predicted by the global meta-model. Allowing $y_{min} = \min\{\mathbf{y}_{obs}\}$, the expected improvement criteria is recognized as:

$$\mathbf{x}_c \in \arg \max_{\mathbf{x} \in \mathbb{X}} E[\max(y_{min} - \hat{y}(\mathbf{x}), 0)] \quad (8)$$

If $\max EI(\mathbf{x}_c) = 0$, next centroid is selected randomly such that $\mathbf{x}_c \notin \mathbb{X}_{obs}$. The expected improvement (EI) criteria has been shown to strike a balance between exploration and exploitation of a space, and was a logical choice for use as a sampling criteria in TBOAR. Nevertheless, the expected improvement criteria, may place too much emphasis on exploitation of promising regions. This "biased" behavior towards exploitation and clustering of observations is noted in Ranjan et al. (2011), and can be seen in Figure 11 (a) and (b) presented in Jones et al. (1998). In an attempt to mitigate this effect, we propose the probabilistic improvement (PI)

criteria. The PI criteria follows the same assumption as EI but takes on EI's pre-integrated form, namely:

$$PI(\mathbf{x}) = \frac{P(I(\mathbf{x}))}{\int_{\mathbb{X}} P(I(\mathbf{x}))} = \frac{\Phi\left(\frac{y_{min} - \hat{y}(\mathbf{x})}{s(\mathbf{x})}\right)}{\int_{\mathbb{X}} P(I(\mathbf{x}))} \quad (9)$$

Discovering the cumulative probability of improvement for $\mathbf{x} \in \mathbb{X}$, we optimize the sampling criteria in a modified manner to that of EI. We add an extra component of randomness to promote exploration by sampling from a uniform distribution $c^* \sim U(0, 1)$ and comparing c^* with $PI(\mathbf{x})$ for all $\mathbf{x} \in \mathbb{X}$. The next initial local search centroid \mathbf{x}_c is then chosen as $\mathbf{x}_c \notin \mathbb{X}_{obs}$ such that $\mathbf{x}_c \in \{\arg \min |PI(\mathbf{x}) - c^*| : \mathbf{x} \in \mathbb{X}\}$. In the following section we present a numerical study of TBOAR.

3 NUMERICAL RESULTS & IMPLEMENTATION DETAILS

In this section we present the challenges facing implementation and basic numerical results of experimentation on TBOAR over two one-dimensional test cases, and one two-dimensional case. We conducted numerical studies on the relative performances of four alternative TBOAR instances. In the local search procedure, two instances fit first order linear Taylor expansion models centered about \mathbf{x}_c , and two used second order quadratic models. Two of the TBOAR formulations use the expected improvement (EI) sampling criteria, and the other use the probabilistic improvement (PI) criteria to determine adaptive restart location. We refer to the four TBOAR algorithms as: Linear EI, Linear PI, Quadratic EI, and Quadratic PI; and to better test absolute algorithm performance we implemented a basic EGO algorithm with stationary OK model.

We make comparisons through two metrics (1) is the mean Euclidean distance between the true global minimum x^* , and the identified global minimum x_{min} - averaged over all macro-replications, and (2) is the percent of macro-replications that identify the true global minimum. Note we relax the true minimum to a target region of size ξ corresponding to a hypersphere centered around the true minimum with volume equal to ξ percent of the total solution space volume. In two dimensions the target region is a circle centered on the true minimum, with radius such that $\pi r^2 = \xi \cdot A$, where A is the total box constrained area.

3.1 Computational Issues in OK Model Parameter Estimation

The numerical instability of exact Gaussian process interpolation of deterministic simulation data is well investigated in Ranjan et al. (2011). The instability arises in the maximum likelihood estimation of Gaussian process model parameters which requires the computation of the determinant and inverse of the model correlation matrix, R . By definition R is positive definite, but when two points in \mathbb{X} with dimensional distance approaching zero are fit, R becomes sparse, near-singular, and ill-conditioned.

We overcome this matrix ill-conditioning through the introduction of a *nugget*, δ , to perturb the diagonal of R . As proposed by Ranjan et al. (2011) we employ a lower bound on this nugget, δ_{lb} , to minimize excessive over-smoothing and non-interpolation when estimating OK model parameters. The nugget lower bound, and new correlation matrix are $\delta_{lb} = \max\left\{\frac{\lambda_n(\kappa(R) - e^a)}{\kappa(R)(e^a - 1)}, 0\right\}$, where $\kappa(R)$ is the condition number of R , and λ_n is the maximum eigenvalue of R . The use of a lower bound on the nugget implies that perturbation is only employed when R is identified as being potentially ill-conditioned. This lower bound gives computational robustness to model fitting, balancing the necessity to numerically compute parameters while still achieving a near-exact interpolating OK emulator.

3.2 One and Two Dimensional Experimental Results

Both one-dimensional test cases have at least one unique local and a unique global minimum, in both cases we assume that \mathbb{X} is a set of box-constraints on the true function domain. The first test function is a sinusoidal wave with increasing amplitude:

$$f_1(x) = (2x + 9.96) \cos(13x - 0.26) \quad (10)$$

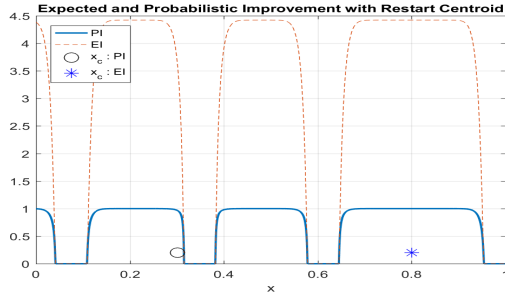
where $\mathbb{X}_1 \in [0, 1]$. The global minimum is located in $x^* = 0.746$ with function value $y^* = -11.45$, and there exists a local minimum at $x = 0.263$ with $y = -10.48$.

The second function is Gramacy & Lee 2012:

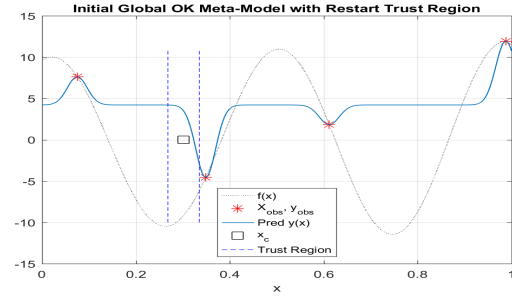
$$f_2(x) = \frac{\sin(10\pi x)}{2x} + (x-1)^4 \quad (11)$$

Where $\mathbb{X}_2 \in [0.5, 2.5]$. The global minimum and related function value are $(x^*, y^*) = (0.5486, -0.869)$, with eight local minimums $\mathbf{x}_l = [0.75, 0.95, 1.15, 1.35, 1.55, 1.74, 1.94, 2.12]$.

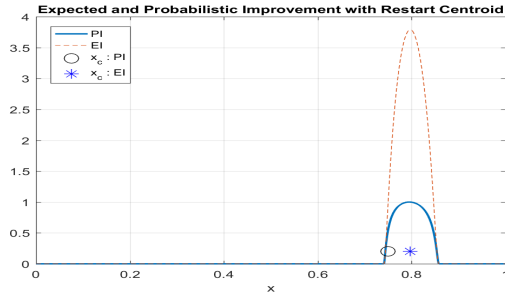
Two sequential adaptive restarts are shown for the simpler test function $f_1(x)$ in Figure 3. With $n_0 = 4$, Figure 3b shows: the initial space filling design $\mathbb{X}_{obs} = [0.075, 0.348, 0.612, 0.988]$, the initial OK model fit, and the true function $f_1(x)$. Figure 3a presents plots of the probabilistic improvement and expected improvement sampling criteria, and the two corresponding centroid selections (note Figure 3b also shows the probabilistic improvement centroid and associated trust region). Following the local optimization subroutine (not shown) the centroid appended to \mathbb{X}_{obs} is the local minimum $x_c = 0.2682$. Figure 3d shows the re-fitted OK model with the local minimum fit exactly. Figure 3c is similar to Figure 3a and shows the progression of adaptive restart sampling. As the centroid and trust region in Figure 3d allude to; the subsequent local search does converge to the global minimum, requiring 18 total simulation observations to identify the true global minimum. With this specific random number generator EGO required 37 total



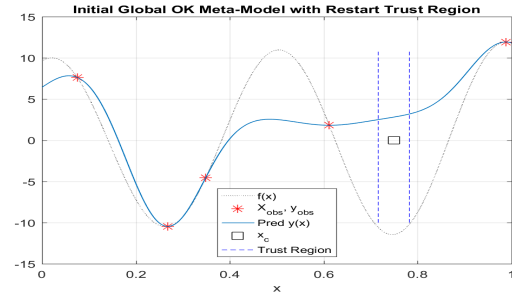
(a) Initial EI and PI evaluations.



(b) Initial OK model fit with \mathbf{x}_c identified via PI.



(c) Second EI and PI evaluations.



(d) Second OK model fit with \mathbf{x}_c identified via PI.

Figure 3: Two TBOAR adaptive restart iterations, (a) and (c) show two iterations of both sampling criteria, and (b), (d) show OK model fits.

simulations to identify the global minimum of $f_1(x)$.

Table 1 shows the results for 500 macro-replications of each of the five algorithms, with a budget of 500 simulations per replication. All algorithm macro-replications use $n_0 = 4$, and common TBOAR parameters are: $\gamma = 1.2$, $\omega = 0.5$, $\eta_1 = 0.25$, $\eta_2 = 0.75$, and $\Delta_0 = \frac{1}{15}(x_{ub} - x_{lb})$. Note that % correct identification corresponds to $\xi = 0.05$, or 5% of the total solution space; we assume that at the 5% level the ξ target region is a compact and continuous set.

Table 1: Numeric results of 500 algorithmic replications on one dimension test functions (*indicates a statistically significant difference from EGO).

Test Function	Metric	EGO	Quad EI	Quad PI	Linear EI	Linear PI
$f_1(x)$	Average $ x_{min} - x^* $	0.032	0.003*	0.009*	0.004*	0.009*
	Std Dev $ x_{min} - x^* $	0.09	0.008	0.05	0.03	0.05
	% Correct Identification	74%	97.8%*	96%*	97.8%*	96%*
$f_2(x)$	Average $ x_{min} - x^* $	0.035	0.037	0.059	0.035	0.04
	Std Dev $ x_{min} - x^* $	0.123	0.115	0.111	0.106	0.085
	% Correct Identification	91%	89.4%	74.8%*	90.4%	84.8%*

Table 1 yields promising results for TBOAR when compared with EGO. A two-sample t-test strongly indicates that all four TBOAR alternatives consistently yield solutions that are significantly better than EGO over $f_1(x)$. The same cannot be said for the Gramacy and Lee function $f_2(x)$, testing confirms that all the algorithms statistically perform the same. This is most likely due to the difficulty of this function with a plethora of local minimums over the relatively small solution space. The PI indicator performs statistically worse than EI, this is most likely due the random nature of centroid selection overshadowing the information of the indicator. Future research on alternative global model forms, centroid selection, and adaptive restart methods remain open.

With proof of concept from the one-dimensional test cases we extend the Quadratic EI TBOAR algorithm to a two-dimensional test case as this TBOAR formulation is assumed to see substantial performance increases in multiple dimensions. Table 2 summarizes numeric results of the Quadratic EI TBOAR and EGO algorithms in two dimensions. Unlike the first test suite we complete only 100 algorithmic replications and establish a finite time allocation of 200 simulated observations per replication. The two-dimensional test case is the six-hump camel function with box constraints: $x_1 \in [-2, 2]$ and $x_2 \in [-1, 1]$; and two symmetric global minimums: $\mathbf{x}_1^* = [0.0898, -0.7126]$ and $\mathbf{x}_2^* = [-0.0898, 0.7126]$. The metrics in Table 2 are the same as in Table 1 but we consider the minimum Euclidean distance from \mathbf{x}_{min} to either global minimum \mathbf{x}_1^* or \mathbf{x}_2^* .

Statistical analysis appears in Figure 4, including a quantile comparison and box plot. Note that in the results six of the 100 TBOAR replications identify a local minimum. The error distance of these six identified local minimums from the true global minimums is nearly four orders of magnitude larger than any error distance yielded by the other 94 runs. As the quantile breakdown in Figure 4 shows, the 90th worst performing Quadratic EI TBOAR replication has a error of only 0.0004, and yet the six outlying points leverage the overall error results so heavily we see the overly inflated average error of 0.097 in Table 2. This same behavior is captured in the box plot in Figure 4 and is the reason why there appears to be no quantile boxes for the Quadratic EI TBOAR.

Table 2: Numeric results of 50 macro-replications in two dimensions.

6 Hump Camel	Metric	Quad EI	EGO
$f(\mathbf{x}) = \left(4 - 2.1x_1^2 + \frac{x_1^4}{3}\right)x_2^2 + x_1x_2 + (-4 + 4x_2^2)x_2^2$	Average $\ x_{min} - x_{1,2}^*\ $	0.097	0.085
	Std Dev $\ x_{min} - x_{1,2}^*\ $	0.385	0.227
	% Correct Identification	94%*	50%

Looking at just mean and standard deviation is a bit misleading as there seems to be no statistical difference. However, the percentage of correct identification shows that 94% of the minimums identified by Quad EI TBOAR lie within the $\xi = 0.05$ target region. Of the 100 EGO runs, only 50 identified a point that resided in this target region. Indeed TBOAR's empirical precision was 6 magnitudes better than EGO. 38% of EGO's macro replications fell in a $\xi = 0.01$ target region, while 45% of TBOAR's runs fell in a

$\xi = 1 \times 10^{-8}$ target region. The degree of precision from TBOAR (a direct result from local searches) in two dimensions shows great potential for extension to higher dimensional solution spaces.

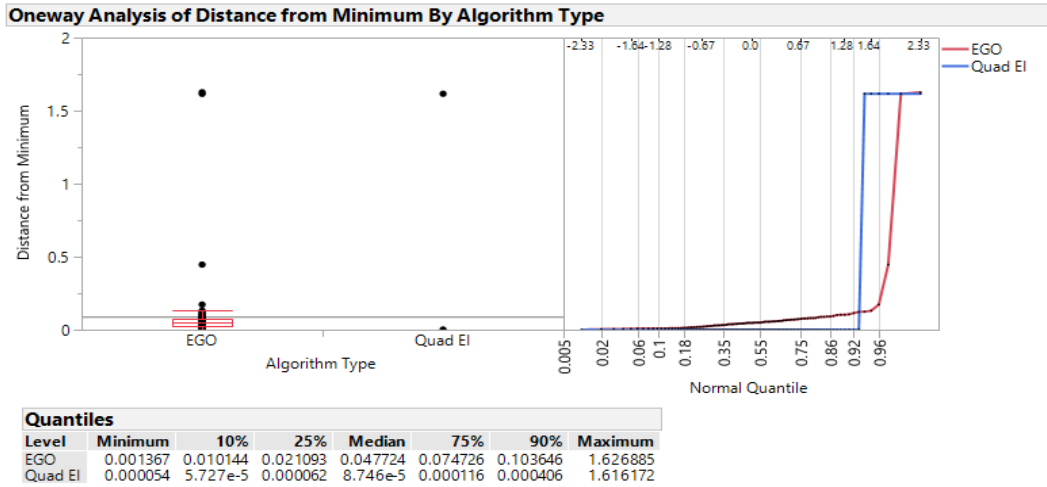


Figure 4: Quantile comparison of EGO and Quad EI TBOAR algorithms.

4 CONCLUSIONS

In this paper we proposed the algorithmic family framework of Trust region Based Optimization with Adaptive Restart. As illustrated in Section 2, the algorithmic framework allows for compartmentalization of the local and global search. Resultantly TBOAR is highly adaptable to the use of more sophisticated local search methods, and the use of alternative global meta-model structures with alternate assumptions that may more accurately fit a given problem or achieve a desired modeling characteristic. Additionally different adaptive restart criteria for both when and where to restart is yet another modularity of the TBOAR framework to achieve desired algorithmic characteristics.

This work presents a contribution to the field of simulation optimization and efficient global optimization as it is the first, to the author's knowledge, formal and iterative integration of local and global optimization methods. While GSTAR breaches the topic of combining local and global models, DMSS dynamically restarts local searches, and ISC globally finds a set of seeds to then be solved by COMPASS; TBOAR adaptively restarts sequential local searches by solving a series of global optimization problems, enabling global optimization to learn from local optimization. Through this work several other avenues for further future research include:

1. Algorithmic tests or criteria to optimally determine an appropriate initial trust region size along with growth and reduction factors.
2. Probabilistic improvement sampling provided a nice alternative to the typical expected improvement sampling criteria, but a more focused study needs to be conducted on the performance of the probabilistic improvement function.
3. An extension of the TBOAR family into higher dimensional cases to test if the gains in performance seen from the one to two-dimensional case continue to scale.
4. While TBOAR currently treats all models (local or global) as being independent of all others, a model ensemble procedure would be a logical extension allowing local model information to serve as a global model approximation in that region.
5. An extension of the TBOAR family of algorithms into the stochastic setting, in this setting TBOAR could adequately be compared to stochastic algorithms such as GSTAR and ISC.

REFERENCES

- Andradóttir, S., and A. A. Prudius. 2009. "Balanced Explorative and Exploitative Search with Estimation for Simulation Optimization". *INFORMS Journal on Computing* 21 (2): 193–208.
- Chang, K.-H., L. J. Hong, and H. Wan. 2013. "Stochastic Trust-Region Response-Surface Method (STRONG) A New Response-Surface Framework for Simulation Optimization". *INFORMS Journal on Computing* 25 (2): 230–243.
- Conn, A. R., K. Scheinberg, and L. N. Vicente. 2009. "Global Convergence of General Derivative-Free Trust-Region Algorithms to First and Second-Order Critical Points". *SIAM Journal on Optimization* 20 (1): 387–415.
- Cressie, N. A., and N. A. Cassie. 1993. *Statistics for Spatial Data*, Volume 900. Wiley New York.
- Hong, J. L., and B. L. Nelson. 2006. "Discrete Optimization via Simulation Using COMPASS". *Operations Research* 54 (1): 115–129.
- Huang, D., T. T. Allen, W. I. Notz, and N. Zeng. 2006. "Global Optimization of Stochastic Black-Box Systems via Sequential Kriging Meta-Models". *Journal of Global Optimization* 34 (3): 441–466.
- Jones, D. R., M. Schonlau, and W. J. Welch. 1998. "Efficient Global Optimization of Expensive Black-Box Functions". *Journal of Global optimization* 13 (4): 455–492.
- Kleijnen, J. P. 2009. "Kriging Metamodeling in Simulation: A Review". *European Journal of Operational Research* 192 (3): 707 – 716.
- Pedrielli, G., and S. H. Ng. 2016. "G-STAR: A New Kriging-Based Trust Region Method for Global Optimization". In *Proceedings of the 2016 Winter Simulation Conference*, edited by T. Huschka, S. Chick, J. Jimenez, P. Frazier, T. Roeder, R. Szechtman, and E. Zhou, 803–814. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.
- Quan, N., J. Yin, S. H. Ng, and L. H. Lee. 2013. "Simulation Optimization via Kriging: A Sequential Search Using Expected Improvement with Computing Budget Constraints". *IIE Transactions* 45 (7): 763–780.
- Ranjan, P., R. Haynes, and R. Karsten. 2011. "A Computationally Stable Approach to Gaussian Process Interpolation of Deterministic Computer Simulation Data". *Technometrics* 53 (4): 366–378.
- Santner, T. J., B. J. Williams, and W. I. Notz. 2013. *The Design and Analysis of Computer Experiments*. Springer Science & Business Media.
- Xu, J., B. L. Nelson, and J. Hong. 2010. "Industrial Strength COMPASS: A Comprehensive Algorithm and Software for Optimization via Simulation". *ACM Transactions on Modeling and Computer Simulation (TOMACS)* 20 (1): 3.
- Yin, G. G., and H. J. Kushner. 2003. *Stochastic Approximation and Recursive Algorithms and Applications*. Springer.
- Zabinsky, Z. B., D. Bulger, and C. Khompatraporn. 2010. "Stopping and Restarting Strategy for Stochastic Sequential Search in Global Optimization". *Journal of Global Optimization* 46 (2): 273–286.

AUTHOR BIOGRAPHIES

LOGAN MATHESSEN is a second year PhD student at Arizona State University pursuing a Degree in Industrial Engineering focused in Operations Research and Industrial Statistics. Two research interests are: efficient global optimization, and design of experiments for statistical modeling. *Email* : lmathese@asu.edu

GIULIA PEDRIELLI is a contributing member to the field of simulation optimization and an Assistant Professor at Arizona State University in the School of Computing, Informatics, and Decision Systems Engineering. *Email* : Giulia.Pedrielli@asu.edu

SZU HUI NG is an Associate Professor in the Department of Industrial and Systems Engineering at the National University of Singapore. *Email* : isensh@nus.edu.sg