AIAA SciTech Forum
6-10 January 2020, Orlando, FL
AIAA Scitech 2020 Forum

10.2514/6.2020-0903

A Delaunay-based method for optimizing infinite time averages of numerical discretizations of ergodic systems

Pooriya Beyhaghi*
ASML Inc, San Diego, CA, 92127, USA

Ryan Alimo†
Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA, 91109, USA

Muhan Zhao‡ and Thomas R. Bewley§
University of California San Diego, La Jolla, CA, 92093, USA

Delaunay-based optimization is a generalizable family of practical, efficient, and provably convergent derivative-free algorithms designed for a range of black-box optimization problems with expensive function evaluations. In many practical problems, the calculation of the true objective function is not exact for any feasible set of the parameters. For problems of this type, a variant of Delaunay-based optimization algorithms dubbed α-DOGS is designed to efficiently minimize the true objective function evaluated with sampling error, while using minimal sampling over the parameter space. In the present work, we extend α-DOGS to additionally address uncertainties of the objective function that are generated by the numerical discretization of the ODE or PDE problems of interest. For validation, this modified optimization algorithm is applied to the (chaotic) Lorenz system. Numerical results indicate that, following the new approach, most of the computational effort is spent close to the optimal solution as convergence is approached.

I. Introduction

The focus of this work is to develop a computational optimization technique that can be seamlessly applied to obtain the optimal parameters involved in stochastic physical problems for which evaluating the cost function accurately involves a significant computational expense. In particular, we are interested in complex systems that require the time averaging of large-scale computations of partial differential equations (PDEs) during the design process. We aim to solve optimization problems with the objective function $f(x) : \Omega \subseteq \mathbb{R}^n \mapsto \mathbb{R}$ in the form of:

$$f(x) = \lim_{T\to\infty} f_T(x), \qquad f_T(x) = \frac{1}{T}\int_{t=0}^{T} y(x,t)dt, \qquad \frac{dy(x,t)}{dt} = G(x,t) \tag{1}$$

where $G(x,t)$ is a nonlinear function of x and t. Note that x represents the vector of adjustable parameters in this dynamic system.

Moreover, we will assume that the analytical expression for $y(x,t)$ is not available, and we can only calculate $y(x,t)$ through numerically time marching the nonlinear function $G(x,t)$. In this way, the available measurement for the objective function is obtained via *finite* sampling defined as follows

$$f(x,h,T) = \frac{1}{N}\sum_{i=1}^{N} y(x,ih), \qquad N = \frac{T}{h}$$

Note that $f(x,h,T)$ is a noisy approximated measurement of $f(x)$ whose accuracy can be increased by decreasing h and increasing T; however, the modification of time step h will change the values of $f(x,h,T)$ and restart the simulation

*Sr. Controls Engineer, ASML Inc., p.beyhaghi@gmail.com
†Lead Machine Learning Scientist, Deep Learning Technology Group at Jet Propulsion Laboratory (JPL), sralimo@jpl.nasa.gov
‡Graduate student in Mechanical and Aerospace Engineering Department, muz021@ucsd.edu
§Professor of Mechanical and Aerospace Engineering Department, bewley@eng.ucsd.edu

from the beginning.

One of the developed algorithm in this class, dubbed α-DOGS, is designed specifically to minimize those objective functions that are given by the infinite-time average of a statistically-stationary ergodic process; in such problems, any numerical or experimental approximation of this function is characterized by sampling error, which may be reduced by additional sampling. However, α-DOGS cannot handle the discretization error, and it is only designed for problems where the uncertainty is due to the finite number of sampling.

The key idea behind the α-DOGS algorithm is that the number of sampling over different points in the feasible domain is varied. In this way, only a limited number of sampling is needed for those points that are far from the basin of global minimizer, whereas more extensive sampling is required as convergence is approached in order to more precisely quantify the objective function in regions closer to the global minimum.
The existence of the discretization error is an important issue in many practical optimization problems that are caused by the numerical discretization of the PDEs of interest. Such challenges arise, for instance, in the minimization of the drag of a numerical approximation of a turbulent flow [1]. One approach to deal with discretization error is to use the same spatial grid for all measurements, which reduces the problem to one grid. However, this approach is computationally expensive since it is not necessary to use a fine spatial grid throughout the whole optimization process.

In this paper, a new algorithm, dubbed α-DOGSX is presented to efficiently automate the trade-off between (a) additional sampling of the ergodic process, and (b) the refinement of the spatial discretization of the ODE (similar approach could be used for PDEs as well). Moreover, for a wide range of optimization problems in form of (1), a target value f_0 exist. In other words, we want to find the control parameters x such that $f(x) < f_0$, rather than actually minimizing the $f(x)$. Therefore, in this paper, we also modify the α-DOGS to optimize the algorithm using the target value.

The structure of this paper is organized as follows: Section II briefly explains the essential elements of the α-DOGS for problems that a target value exist. Section III presents the modified algorithm to deal with discretization error efficiently. Section IV shows the performance of the proposed algorithm on a model problem based on the Lorenz equation. Some conclusions are drawn in Section V.

II. α-DOGS for Problems with Target Value

A. Algorithm

In this section, the general framework of α-DOGS algorithm is presented which is designed to minimize the objective function of the form:

$$f(x) = \lim_{N\to\infty} \frac{1}{N} f_N(x), \;\; f_N(x) = \sum_{i=1}^{N} f_i(x) \tag{2}$$

where $f_i(x)$ is considered as a stationary and ergodic random process at each x.

Remark 1. *Estimating the value of the uncertainty associated with $f_N(x)$ is the classical uncertainty quantification (UQ) problem that is studied in [2], [3] and [4]. In this section, we will assume that this quantity is known, and it is denoted by $\sigma_N(x)$.*

The original algorithm [5] is designed to optimize the objective function of form 2 in general, but for a wide range of practical optimization problems, we seek a point such that $f(x) \le f_0$. In this section, a modified version of algorithm in [5] is presented which more efficiently solves those optimization problems that have a target value f_0. This approach is similar to the EI (Expected Improvement) approach, which is considered as a major relevant criterion in global optimization.

Definition II.1. *Let S be a set of points in the feasible domain L and $Y = \{f_{N_x}(x)|x \in S\}$ as a set of approximated measurements of $f(x)$ at these points. N_x denotes the length of averaging sampling at point x. $p(x)$ denotes the regression that passed through these measurements. Suppose f_0 is a target value for $f(x)$, then the* ***continuous search function*** *$s_c(x)$ is defined as follows:*

$$s_c(x) = \begin{cases} \frac{p(x)-f_0}{e(x)} & \text{if } p(x) \ge f_0, \\ p(x) - f_0 & \text{otherwise} \end{cases} \tag{3}$$

In this framework, the regression $p(x)$ could be a user defined robust regression process which satisfies

$$|p(x) - f_{N_x}(x)| \leq \beta\sigma_{N_x}(x), \forall x \in S \tag{4}$$

here $\sigma_{N_x}(x)$ is the uncertainty with the function evaluation $f_{N_x}(x)$.

$e(x)$ is the Delaunay-based uncertainty function defined in [5]. The piecewise quadratic function $e(x)$ is non-negative everywhere in L and remains zero at the evaluated points in S.

Definition II.2. *Consider S as a set of feasible points in L, and $Y = \{f_{N_x}(x)|x \in S\}$ as a set of measurements of $f(x)$ at these points. Moreover, $\sigma_{N_x}(x)$ is considered as the uncertainty quantification of $f_{N_x}(x)$. Furthermore, $p(x)$ is considered as a regression for these measurement. Then the* ***discrete search function*** *$s_d(x)$ for each $x \in S$, is defined as*

$$s_d(x) = \begin{cases} \frac{\min\{p(x), 2f_{N_x}(x) - p(x)\} - f_0}{\sigma_{N_x}(x)} & \text{if } p(x) \geq f_0 \\ p(x) - f_0 & \text{otherwise} \end{cases} \tag{5}$$

In addition to the discrete and continuous search functions, the concept of the Cartesian grid is needed to present the α-DOGS algorithm which is defined in [6], [7] and [8].

Definition II.3. *The Cartesian grid of level ℓ for the feasible domain $B = \{x|a \leq x \leq b\}$, denoted as B_ℓ, is defind as:*

$$B_\ell = \left\{ x \middle| x = a + \frac{1}{2^\ell}(b-a) \otimes z, z \in \{0, 1, ..., 2^\ell\} \right\} \tag{6}$$

A quantizer of point $x \in B$ onto B_ℓ is the point x_q, which has the minimum distance to x from the B_ℓ grid. Note that quantizer of point x is not necessarily unique. The maximum discretization error is defined as follows:

$$\delta_\ell = \max_{x \in B} ||x - x_q|| \tag{7}$$

Remark 2. *Scaling the function evaluations and the domain of the parameters is required in order to have an efficient optimization algorithm. We consider that the normalized variables x_i within the range $0 \leq x_i \leq 1$, and $0 \leq f(x) \leq 1$ after scaling.*

Now we present the modified Adaptive-K α-DOGS algorithm to find a point such that $f(x) \leq f_0$. The steps of this algorithm are summarized in Algorithm 1. The detailed convergence proof for Adaptive-K α-DOGS is available in the full version paper of this work.

One of the challenges in Algorithm 1 is to determine the most promising points for the available measurements at the the finite iteration k. This point is called the candidate point which is determined as follows:

- If $s_c(x_k) \leq s_d(w_k)$, then the candidate point is defined as the minimizer x_k of $s_c(x)$. The iteration k is called the *identifying sampling iteration*.
- If $s_c(x_k) \geq s_d(w_k)$, then the candidate point is defined as the minimizer w_k of $s_d(x)$. The iteration k is called the *addtional sampling iteration*.

Remark 3. *An important factor in Algorithm 1 is the construction of the regression $p^k(x)$. Generally, any well-behaved regression that is robust [5] can be used. However, in the simulation of this paper, we have used the efficient polyharmonic spline regression process (see [9, 10]).*

Remark 4. *Minimizing the search function $s_c^k(x)$ is an important sub-problem in Algorithm 1, as explained in [6].*

In addition to the regression process, there are a few parameters, which play a key role in Algorithm 1, and are summarized as follows:

a. β, which controls the amount of violation of the regression from the measurement at the available datapoints.
b. ℓ_0, which quantifies the mesh level at the first level. In the simulations of this paper, we have used $\ell_0 = 3$.
c. The initial averaging length N_0, and incrementing length N_1, which are problem-dependent parameters and control the minimum averaging length that should be considered between each two iterations of Algorithm 1.

One of the challenges we face in Algorithm 1 is determining the most promising point for the available measurements at the finite iteration k. This point is called the candidate and is determined as follows:

1. For $k = 0$, it is the point in the initial set whose measured value is minimized.

Algorithm 1 Adaptive-K α-DOGS algorithm to find a point such that $f(x) \leq f_0$, where $f(x)$ is in form of (2)

Input: Set $k = 0$ and initialize the algorithm $\ell = \ell_0$ as the initial grid level. Initialize the algorithm with a set of points S_0 on the initial grid of level ℓ_0. Afterwards, calculate an initial estimate with averaging length of N_0 at the initial points. Initialize the amount of additional sampling N.
repeat
  Calculate (or, update) the regression $p_k(x)$ that satisfies (4) over S_k, Delaunay triangulation Δ, and $s_c(x)$, $s_d(x)$.
  If a point $x \in S_k$ exists such that $\frac{p_k(x)-f_0}{\sigma_{N_x}(x)} \leq 1$, then increment the averaging length at x, N_x by N, increment k and repeat from 1.
  Find x_k and w_k as the minimizer of $s_c(x)$ and $s_d(x)$ respectively.
  if $s_c(x_k) \leq s_d(w_k)$ **then**
    Determine q_k as a quantizer of x_k on the grid of level ℓ_k.
    if $q_k \notin S_k$ **then**
      Add q_k to S_k calculate an initial estimate of length N_0 at q_k;
    else
      Refine the grid by setting $\ell_{k+1} = \ell_k + 1$ and increment k.
    end if
  else
    $s_c(x_k) \geq s_d(w_k)$ and increase the averaging length at w_k by N.
  end if
until ℓ_k achieves $\ell_{\max}$ or target value achieved.

2. If step k is decreasing the mesh, then $\eta_k = q_k$.
3. If step k is improving iteration and w_k has the maximum averaging length over all points in S, then $\eta_k = w_k$.
4. If neither case 2 nor case 3 happens, $\eta_k = \eta_{k-1}$ for $k > 0$.

In the next section, we will analyze the convergence properties of Algorithm 1, and show that the value of the truth function at the candidate point will ultimately achieve the target value, if the target value f_0 is achievable.

B. Convergence analysis of Algorithm 1

The analysis presented in this section is similar to the one given in section 4 of [9]. Note that Algorithm 1 is a modified version of the Algorithm 1 described in [5] for the range of problems where a target value is known; therefore, the analysis is similar.

In order to analyze the convergence properties of Algorithm 1, the following assumptions are made:

1. The truth function $f(x)$ and the regressions $p^k(x)$ are twice differentiable functions, and

$$-2\,\hat{K}I \leq \nabla^2 f(x) \leq 2\,\hat{K}I, \quad \forall x \in L, \tag{8}$$

$$-2\,\hat{K}I \leq \nabla^2 p^k(x) \leq 2\,\hat{K}I, \quad \forall x \in L. \tag{9}$$

2. The truth function $f(x)$ and the interpolating function $p^k(x)$ are Lipschitz with constant $\hat{L}$.
3. There is a constant γ such that for all measurements $f_N(x)$ at point x with uncertainty σ_N, then

$$|f_N(x) - f(x)| \leq \gamma\sigma_N(x). \tag{10}$$

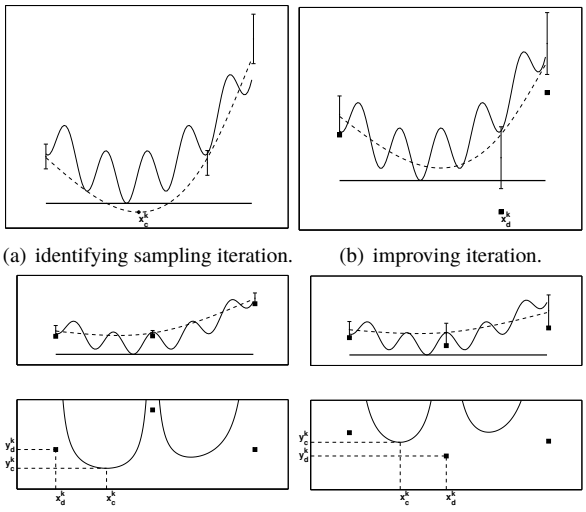
   This is a restrictive assumption for Algorithm 1. In practice, Algorithm 1 works well even if (10) is not verified, but in order to simplify the analysis, we make this additional assumption*.
4. Defining $\sigma_N(x)$ as an uncertainty associated with the measurement $f_N(x)$, we have

$$0 < \sigma_N(x) \leq E(N), \tag{11}$$

$$\lim_{N\to\infty} E(N) = 0. \tag{12}$$

   Note that $E(N)$ is considered to be a positive and monotonically decreasing function.
5. The target value f_0 is achievable.

*In [5], a less restrictive assumption for the measurement process is imposed.



(a) identifying sampling iteration. (b) improving iteration.

(c) identifying sampling iteration. (d) improving iteration.

Fig. 1 Representation of essential elements of Algorithm 1 in improving iteration and identifying sampling iteration. Subfigures a), b) **indicate (dashed curve) the truth function** $f(x)$**, (black lower line) the target value** f_0**, (black curve) the regression model** $p^k(x)$**, and (error bars)** $\sigma^k(x)$**. Subfigures** c), d) **indicate the continuous search function** $s_c^k(x)$ **and discrete search function** $s_d^k(x)$**, and their minimizers in** c) **identifying sampling iteration and** d) **improving iteration.**

Before analyzing the convergence of Algorithm 1, we present some preliminary lemmas.

Lemma II.1. *At each iteration k of Algorithm 1, we have:*

$$\min\{s_c^k(x_k), s_d^k(w_k)\} \leq \max\{3\hat{K}, 2\gamma\} \tag{13}$$

Proof. In Lemma 4 of [5], it is shown that using Assumption 1, we have:

$$\min_{\{z\in S^k\}} \{2\, f(z) - p^k(z)\} + p^k(x) - 2f(x) - 3\,\hat{K} e^k(x) \leq 0, \tag{14}$$

for all $x \in L$. Note that $e^k(x)$ is the uncertainty function based on the Delaunay triangulation.

On the other hand, according to Assumption 4, for each point $z \in S^k$ with a measurement $f_{N_z^k}(z)$ and an uncertainty $\sigma_{N_z}(z)$,

$$|f_{N_z^k}(z) - f(z)| \leq \gamma\sigma_{N_z}(z). \tag{15}$$

Thus, we have:

$$2\, f_{N_z^k}(z) - p^k(z) \leq 2\, f(z) - p^k(z) + 2\,\gamma\sigma_{N_z^k}(z) \tag{16}$$

where $f_{N_z^k}(z)$ is the measured value of z at iteration k with uncertainty $\sigma(z)$. Thus, using (14) and (16), we have:

$$\begin{aligned}\min_{\{z\in S^k\}} &\{2\, f_{N_z^k}(z) - p^k(z) - 2\gamma\sigma_{N_z}(z)\} \\ &+ p^k(x) - 2f(x) - 3\,\hat{K} e^k(x) \leq 0.\end{aligned}$$

Now let us consider x^*, the global minimizer of $f(x)$ in L. Since f_0 is achievable (Assumption 4), then $f(x^*) \leq f_0$; therefore,

$$\begin{aligned}\min_{\{z\in S^k\}} &\{2\, f_{N_z^k}(z) - p^k(z) - 2\gamma\sigma(z) - f_0\} \\ &+ p^k(x^*) - f_0 - 3\,\hat{K} e^k(x^*) \leq 0,\end{aligned}$$

$$\min\{\min_{\{z\in S^k\}}\{2\,f_{N_z^k}(z) - p^k(z) - 2\gamma\sigma_{N_z}(z) - f_0\},$$
$$p^k(x^*) - f_0 - 3\,\hat{K}e^k(x^*)\} \le 0.$$

According to the construction of the discrete search function (5), to the fact that w_k is its minimizer, and to the fact that $\sigma_{N_z}(z) > 0$,

$$\min\{\sigma(z)(s_d^k(w_k) - 2\gamma), p^k(x^*) - f(x^*) - 3\,\hat{K}e^k(x^*)\} \le 0. \tag{17}$$

However, if $s_d^k(w_k) - 2\gamma \le 0$, then equation (13) is satisfied; thus, we will assume that $p^k(x^*) - f(x^*) - 3\,\hat{K}e^k(x^*) \le 0$. Now, if $e^k(x^*) > 0$, then according to the construction of the discrete search function (5),

$$s_c^k(x^*) - 2\hat{K} \le 0. \tag{18}$$

Furthermore, x_k is the minimizer of $s_c^k(x)$; thus, (13) is satisfied. The only case that is left is when $e^k(x^*) = 0$ and $p^k(x^*) - f_0 \le 0$. Note that since $e^k(x^*) = 0$ by construction of the uncertainty function [6], with $x^* \in S^k$, and since $p^k(x^*) - f_0 \le 0$, then $s_d^k(w_k) \le s_d^k(x^*) \le 0$. □

Lemma II.2. *If iteration k of Algorithm 1 is a mesh refinery step, then*

$$f(y_{q_k}) - f_0 \le (1 + \beta + \gamma)T, \tag{19}$$
$$T = \max\{E\,\delta_k^2, F\,\delta_k\},$$
$$E = 4\,\hat{K} + 2\,\gamma, \quad F = \hat{L}$$

where y_{q_k} and σ_k are the measurement and its uncertainty at point q_k. Note that since this step is refines the mesh, q_k is in S^k. Moreover, δ_k is the quantization error at iteration k.

Proof. First, we will show that

$$y_{q_k} - f_0 \le (1 + \beta)T, \text{ and } \quad \sigma_{q_k} \le T. \tag{20}$$

Since step k is mesh-decreasing, according to the construction of Algorithm 1, $R = s_c^k(x_k) \le s_d^k(w_k)$. Then using Lemma II.1, $R \le \max\{3\hat{K}, 2\gamma\}$. If $R \le 0$, then $p^k(x_k) \le f_0$. Since $p^k(x_k)$ is Lipschitz, then

$$p^k(q_k) - f_0 \le \hat{L}\delta_k,$$
$$\sigma_k \le p^k(q_k) - f_0,$$
$$\sigma_k \le \hat{L}\delta_k,$$
$$y_{q_k} - f_0 \le \hat{L}(1 + \beta)\delta_k.$$

Thus, equation (20) is satisfied. We will now consider the case where $R > 0$. In this case, since R is the minimizer of $s_c^k(x) = \frac{p^k(x) - f_0}{e^k(x)}$, then by construction x_k is also the minimizer of $G(x) = p^k(x) - Re^k(x)$. Using Assumption 1, equality $\nabla^2 e^k(x) = 2\,I$ (see [6]) and Lemma 3 in paper [5],

$$G(q_k) - G(x_k) \le (\hat{K} + R)\delta_k^2 \le (4\,\hat{K} + 2\,\gamma)\delta_k^2. \tag{21}$$

Moreover, $G(x_k) = f_0$ and since $q^k \in S^k$, then $e^k(q_k) = 0$.

$$p^k(q_k) - f_0 \le (4\,\hat{K} + 2\,\gamma)\delta_k^2, \tag{22}$$
$$\sigma_{N_{q_k}}^k \le p^k(q_k) - f_0, \tag{23}$$
$$f_{N_{q_k}^k}(q_k) - f_0 \le (4\,\hat{K} + 2\,\gamma)(1 + \beta)\delta_k^2. \tag{24}$$

Now using equation (20) and Assumption 3, (19) is satisfied. □

Lemma II.3. *If iteration k of Algorithm 1 is an improving iteration, and thus increases the uncertainty of the maximum averaging length, then*

$$f(w_k) - f_0 \le (\beta + \max\{3\hat{K}, 2\gamma\})E(N^k w_k), \tag{25}$$

where $N_{w_k}^k$ is the averaging length at w_k.

Proof. Since iteration k is improving, $s_d^k(w_k) \leq \max\{3\hat{K}, 2\gamma, 1\}$, using Lemma II.1, then

$$\min\{p^k(w_k), 2\hat{f}(w_k) - p^k(w_k)\} - f_0 \leq \max\{3\hat{K}, 2\gamma\}\sigma_{w_k}^k(y_k). \tag{26}$$

Since the regression is robust,

$$f_{N_{w_k}^k}(w_k) - f_0 \leq (\beta + \max\{3\hat{K}, 2\gamma\})\sigma_{N_k}^k(w_k). \tag{27}$$

Now, according to Assumption 3, we have

$$f(w_k) - f_0 \leq (\gamma + \beta + \max\{3\hat{K}.2\gamma\})\sigma_{N_k}^k(w_k), \tag{28}$$

Thus, using assumption 5, equation (25) is shown. □

Theorem II.4. *Let us consider η_k, the candidate point at iteration k. Then*

$$\limsup_{k\to\infty} f(\eta_k) - f_0 \leq 0. \tag{29}$$

Proof. According to the construction of the candidate point, its value is changed each time that we have either a mesh refinery step or an improving iteration, which maximizes the averaging length. Therefore, using Lemmas II.2 and II.3, equation (29) is satisfied if there is an infinite number of modifications in the value of η_k.

This theorem is shown by contradiction. Assuming that we have a finite number of mesh refinery steps, Algorithm 1 will obtain only a finite number of datapoints; thus, there must be an infinite number of improving iterations at a finite number of points. As a result, there is an infinite number of iterations that increase the averaging length at one of the available datapoints which has the maximum averaging length; this is a contradiction with our contradictory assumption; hence the proof of our theorem. □

III. Modified Optimization Algorithm Dealing With Discretization Error: α-DOGSX

In this section, we modify Algorithm 1 to solve more general problems in the form of (1). The main difference between the objective function of form (1) and (2) is that the convergence to $f(x)$ in (1) can be achieved only when the time step h reduces to zero.

Recall that during each addtional sampling iteration of Algorithm 1, the averaging length at point w_k is simply incremented by N_1. However, in α-DOGSX, there are two possible options for improving the accuracy of the available measurement at point w_k:

- Increasing the value of the averaging length, T.
- Decreasing the mesh size, h.

To develop an efficient policy between increasing T and decreasing h, the following factor are important:

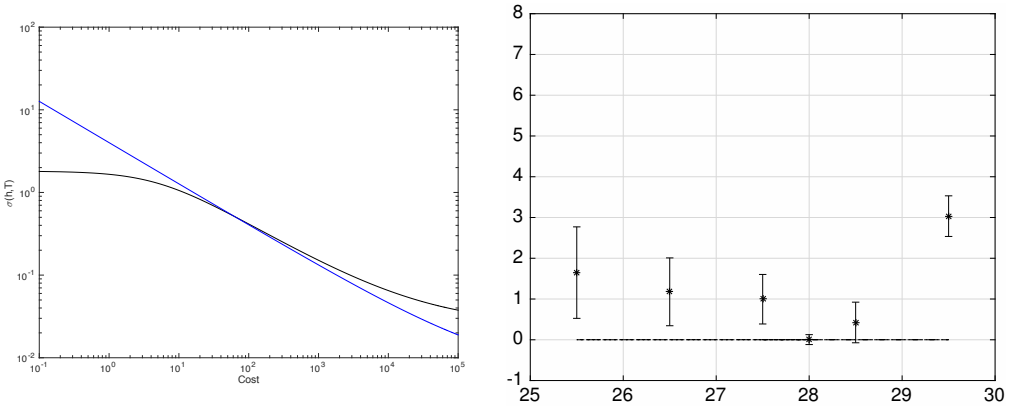
- The value of the uncertainty associated the measurement $f(w_k, h_k, T_k)$ for $f(w_k)$, denoted by $\sigma(w_k, h_k, T_k)$.
- The required amount of improvement at point w_k.
- The computational cost of the improvement for objective function measurement.

Remark 5. *The required amount of improvement plays a key role in our measurement improvement process. Fig. 2(a) illustrates the role of the value of required uncertainty. It is observed that to achieve a small amount of improvement, it is typically more efficient to increase the averaging length, T. However, for greater improvement, the measurement should be improved by modifying both h and T.*

Now we will develop a procedure to estimate the required amount of reduction at point w_k. There are two possible cases for an improving step of the iteration k of Algorithm 1:

1. Iteration k is improving since $s_d(w_k) \leq s_c(x_k)$. In this case, we expect to improve the uncertainty until $s_d(w_k) = s_c(x_k)$. Therefore, the required improvement at w_k is denoted by $\sigma_k^R(w_k)$,

$$\sigma_k^R(w_k) = \frac{\min\{p(w_k), 2f_N(w_k) - p(w_k)\}}{s_c(x_k)} \tag{30}$$



(a) $k = 1$ (b) $k = 8$

Fig. 2 (a) Illustration of the effect of the required uncertainty on the process of improvement. Black curve show the trend of the uncertainty function at the current h as a function of computational time, and the blue curve shows the behaviour of this function for a reduced h. It is observed that for small value of σ_R, it is more efficient to increase averaging length, but for further improvement, it is more efficient to reduce h; (b) Illustration of the generated points by α-DOGSX on Lorenz system.

2. Iteration k is improving since a point $x \in S$ exists such that $p(x) - f_0 \leq \sigma_{N_x}(x)$. In this case, the required improvement at point x is obtained by

$$\sigma_k^R(w_k) = \max\{p(x^*) - f_0, 0\} \tag{31}$$

Note that expressions in (30) and (31) estimates $\sigma_k^R(w_k)$ assuming $p^k(x)$ is constant during the measurement improvement process. This assumption may not be true, especially if the reduction of the uncertainty is high. Thus, we will limit the reduction of the uncertainty by the half of the current uncertainty.

Now based on the available required improvement, the optimal procedure for the measurement improvement can be developed through matching the uncertainty at the improving point z. The notation $\sigma(z, h, T)$ denotes the quantity of uncertainty at the point z with time length T and discretization step h. For a wide range of practical problems, [3], the expression of $\sigma(z, h, T)$ can be formulated as follows

$$\sigma(z, h, T) = \sqrt{(C_0(z)h^p)^2 + (\frac{\sigma_0(z)}{\sqrt{T}})^2} \tag{32}$$

here p is the order of accuracy of the time marching scheme used in practice. The first term in (32) works as the discretization error and the second term serves as the sampling error computed from time averaging. Using equation (32) as a model for the value of the uncertainty, the constant $C_0(z)$ and $\sigma_0(z)$ are determined empirically, based on the available measurement [3].

The main modification in Algorithm 1 is the determination of time length T and time step h for the point z in each improving iteration. Notice that there are two sources of improving point z in Algorithm 1. The first one is from line 4 and the other one is from line 14.

Algorithm 2 α-DOGSX algorithm to determine the time length T and time step h

1: If $z = w_k$, calculate σ_k^R from (30); otherwise, z is found from line 4 of Algorithm 1, compute σ_k^R using (31).
2: If $\sigma_k^R \leq \frac{\sigma(z)}{2}$, replace σ_k^R with $\frac{\sigma(z)}{2}$
3: For each $h_l = h_k 2^l$, $l \in \{0, 1, ..., L\}$, calculate T_l such that

$$\sigma(z, h_k 2^l, T_l) = \sigma_k^R \tag{33}$$

where $\sigma(z, h_k 2^l, T_l)$ is the uncertainty associated with the measurement at point z with averaging length Tl and mesh size of hl .
4: For each $l \in \{0, 1, ..., L\}$, calculate the cost associated with each required measurement as follows:

$$Cost_l = \frac{T_l}{h_k 2^l}, \ \forall l \geq 1 \tag{34}$$

Note that in the above expression, we estimate the cost of the measurement process by the number of time marching that are required.
5: Calculate l_{opt} as the minimizer of $Cost_l$, take $h_{k+1} = h_k 2^l$ and $T_{k+1} = T_l$.

IV. Implementation of α-DOGSX on Synthetic Lorenz problem

In this section, the developed Algorithm 2 is improved from [7] and applied on a synthetic model problem based on Lorenz system. The Lorenz system is a strange attractor that arises in a system of equations describing the 2-dimensional flow of a fluid of uniform depth, with an imposed vertical temperature difference. The chaotic behavior of a simplified 3-dimensional system of this problem, known as the Lorenz equations [11], is given below:

$$\frac{d}{dt}X = s(Y - X) \tag{35}$$

$$\frac{d}{dt}Y = -XZ + rX - Y \tag{36}$$

$$\frac{d}{dt}Z = XY - bZ \tag{37}$$

In the Lorenz system, the values of r, b, and s are positive. The Lorenz system becomes chaotic if [11]

$$r > s\frac{s + b + 3}{s - b - 1}. \tag{38}$$

Moreover, r is related to the finite time averaging quantity of Z in (35), denoted as $\bar{Z}$. Using this relationship, the upper bound for r can be estimated by:

$$\bar{Z}^i \leq (r - 1)^i, \quad i = 1, 2, 3. \tag{39}$$

Imposing (39) and (38), we choose $24 \leq r \leq 30$, $b = 8/3$ and $s = 10$. It has been reported in [3] that the discretization error of $\bar{Z}$, ε_h, can be modeled as $C_0\, h^p$ for the Lorenz system where $p = 3$ and $C_0 = 0.8$ approximately.

Consider an estimation problem, in which values of various moments of the chaotic attractor at the nominal values of the parameters are taken as the target. The goal is to search over the parameter space of in an attempt to determine the optimal parameter values $r^* = 28$ through minimizing the objective function

$$J(r) = \sum_{i=1}^{K} (f_i - f_{target})^2 \tag{40}$$

where f_i is considered to be different moments of Z direction. The desired value for f_{target} is 23.57 using RK4 scheme. Fig. 2(b) shows the points that are generated during the optimization algorithm. The initialization is performed with an initial time averaging of $T_0 = 50$, and the solver grid size of $h_0 = 2.5 \times 10^{-3}$ for α-DOGS and $h_0 = 0.05$ for α-DOGSX.

Table 1 and Fig. 3 show the points generated during the optimization algorithm. The initialization is performed with an initial time averaging of $T_0 = 50$, and the solver grid size is $h_0 = 2.5 \times 10^{-3}$ for α-DOGS and $h_0 = 0.05$ for α-DOGSX.

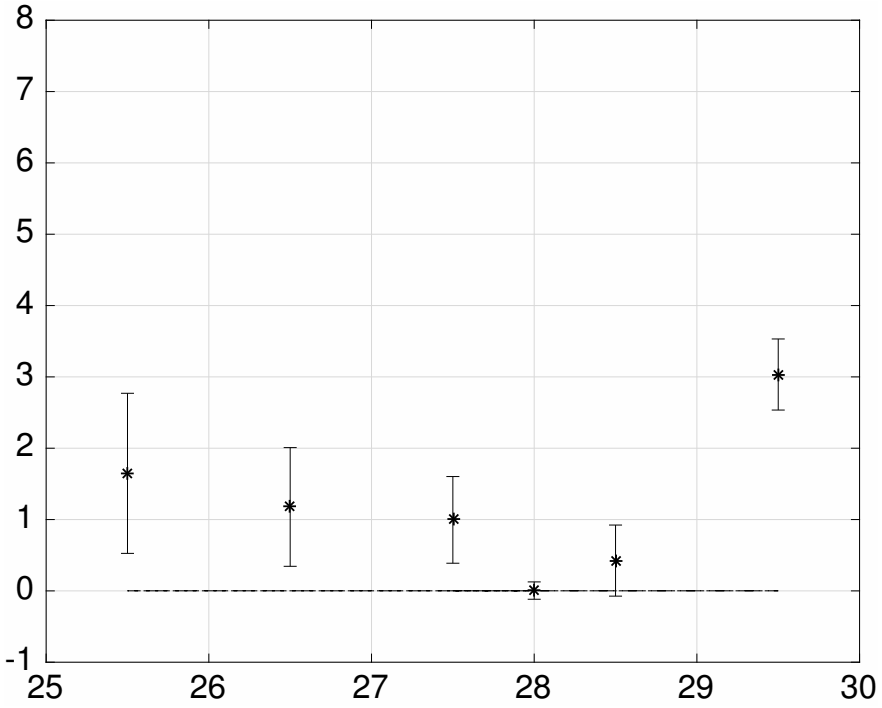


Fig. 3 Illustration of the points generated by α-DOGSX on Lorenz problem.

Table 1 The points generated by applying the optimization algorithm to the Lorenz system.

r	27.5	28.5	29.5	25.5	26.5	28.
J	0.9	0.42	3.0	1.65	1.17	10^{-3}
T	50	360	280	210	300	570
h	0.05	2×10^{-4}	1×10^{-3}	3×10^{-3}	8×10^{-3}	5×10^{-4}
Cost	10^3	10^6	10^5	10^4	10^5	10^7

V. Conclusion

In this paper, we developed a new algorithm, dubbed α-DOGSX, for solving optimization problems, whose functions are the infinite time-averaged statistics of a continuous stationary process without discretization error. The method we presented is based on the Delaunay-based optimization algorithm which is developed in [5].

We used the finite-time averaging value, which is an estimate for the truth objective value at each given resolution of the mesh grid. The novelty of our algorithm is that the different averaging lengths and mesh grid resolutions are used at different sets of design parameters. The flexibility in the calculation of function evaluations that are less accurate far from the solution allows us to get an efficient algorithm for solving problems whose function values are given.

Our algorithm has a tuning parameter α, which plays a key role in the optimization process. It is observed that as α decreases, we use more data points for the convergence, but the total amount of averaging decreases. Note, that this flexibility is useful since for some applications (e.g., turbulence simulations), it is more efficient to use fewer datapoints, as the examination of each data point is costly (because of the complexity in setting up a measurement process at each point). Note also that in the limit where the initialization cost is too expensive, it is better to use the same averaging length for all data points.

Another important issue which has to be considered is the role of the desired accuracy in our algorithm. It is observed that as the desired accuracy increases, the efficiency of the optimization process increases as well. In other words, in

the limit where the desired accuracy goes to 0, the computational cost of the optimization process converges to the computational cost of the most expensive measurement.

Furthermore, since the stated problem (1) is stochastic, the performance of the presented algorithms are also stochastic. In other words, based on the measured values that are derived as the algorithm proceeds, the convergence speed changes; however, the main advantage of this algorithm is that the convergence is guaranteed under some conditions.

Although this new method works well for the test functions we presented in this paper, it cannot be practically used for high-dimensional problems yet, due to the exponential growth of the number of simplices with the dimension. This is an important limitation of our optimization algorithm.

In our future work, we intend to implement our algorithm for some practical optimization problems in turbulence simulations.

Acknowledgement

The authors also gratefully acknowledge funding from Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration (NASA) and funding from AFOSR FA 9550-12-1-0046 in support of this work.

References

[1] Marsden, A. L., Wang, M., Dennis, J. E., and Moin, P., "Optimal aeroacoustic shape design using the surrogate management framework," *Optimization and Engineering*, Vol. 5, No. 2, 2004, pp. 235–262.

[2] Beyhaghi, P., Alimo, S., and Bewley, T., "Uncertainty Quantification of the time averaging of a Statistics Computed from Numerical Simulation of Turbulent Flow," *arXiv preprint arXiv:1802.01056*, 2018.

[3] Oliver, T. A., Malaya, N., Ulerich, R., and Moser, R. D., "Estimating uncertainties in statistics computed from direct numerical simulation," *Physics of Fluids*, Vol. 26, No. 3, 2014, p. 035101.

[4] Salesky, S. T., Chamecki, M., and Dias, N. L., "Estimating the random error in eddy-covariance based fluxes and other turbulence statistics: the filtering method," *Boundary-layer meteorology*, Vol. 144, No. 1, 2012, pp. 113–135.

[5] Beyhaghi, P., Alimo, R., and Bewley, T., "A derivative-free optimization algorithm for the efficient minimization of functions obtained via statistical averaging," *arXiv preprint arXiv:1910.12393*, 2019.

[6] Beyhaghi, P., Cavaglieri, D., and Bewley, T., "Delaunay-based derivative-free optimization via global surrogates, part I: linear constraints," *Journal of Global Optimization*, Vol. 66, No. 3, 2016, pp. 331–382.

[7] Alimo, S. R., *Efficient New Approaches for Data-Driven Global Optimization, with Applications in Computer-Aided Design*, University of California, San Diego, 2017.

[8] Zhao, M., Alimo, S. R., and Bewley, T. R., "An active subspace method for accelerating convergence in Delaunay-based optimization via dimension reduction," *2018 IEEE Conference on Decision and Control (CDC)*, IEEE, 2018, pp. 2765–2770.

[9] Beyhaghi, P., and Bewley, T., "Implementation of Cartesian grids to accelerate Delaunay-based derivative-free optimization," *Journal of Global Optimization*, Vol. 69, No. 4, 2017, pp. 927–949.

[10] Wahba, G., *Spline models for observational data*, Vol. 59, Siam, 1990.

[11] Goluskin, D., "Bounding averages rigorously using semidefinite programming: mean moments of the Lorenz system," *Journal of Nonlinear Science*, Vol. 28, No. 2, 2018, pp. 621–651.

[12] Alimo, S. R., Beyhaghi, P., and Bewley, T. R., "Delaunay-based Derivative-free Optimization via Global Surrogates, Part III: Nonconvex Constraints," *Journal of Global Optimization*, in press.

[13] Alimo, S. R., Cavaglieri, D., Beyhaghi, P., and Bewley, T. R., "Design of IMEXRK time integration schemes via Delaunay-based derivative-free optimization with nonconvex constraints and grid-based acceleration," *Journal of Global Optimization*, in press.

[14] Zhao, M., Alimo, S. R., Beyhaghi, P., and Bewley, T. R., "Delaunay-based Derivative-free Optimization via Global Surrogates with Safe and Exact Function Evaluations," *2019 IEEE Conference on Decision and Control (CDC)*, IEEE, To appear.

[15] Beyhaghi, P., and Bewley, T. R., "Delaunay-based derivative-free optimization via global surrogates, part II: convex constraints," *Journal of Global Optimization*, Vol. 66, No. 3, 2016, pp. 383–415.

[16] Sankaran, S., Audet, C., and Marsden, A. L., "A method for stochastic constrained optimization using derivative-free surrogate pattern search and collocation," *Journal of Computational Physics*, Vol. 229, No. 12, 2010, pp. 4664–4682.

[17] Talgorn, B., Le Digabel, S., and Kokkolaras, M., "Statistical surrogate formulations for simulation-based design optimization," *Journal of Mechanical Design*, Vol. 137, No. 2, 2015, p. 021405.

[18] Snoek, J., Larochelle, H., and Adams, R. P., "Practical bayesian optimization of machine learning algorithms," *Advances in neural information processing systems*, 2012, pp. 2951–2959.

[19] Jalali, M., VanDine, A., Chalamalla, V. K., and Sarkar, S., "Oscillatory stratified flow over supercritical topography: Wave energetics and turbulence," *Computers & Fluids*, Vol. 158, 2017, pp. 39–48.

[20] Luchini, P., "Immersed-boundary simulations of turbulent flow past a sinusoidally undulated river bottom," *European Journal of Mechanics-B/Fluids*, Vol. 55, 2016, pp. 340–347.

[21] Foias, C., Jolly, M., and Manley, O., "Kraichnan turbulence via finite time averages," *Communications in mathematical physics*, Vol. 255, No. 2, 2005, pp. 329–361.

[22] Alimo, S., and He, D., "Multi-stage algorithm for uncertainty analysis of solar power forecasting," *2016 IEEE Power and Energy Society General Meeting (PESGM)*, IEEE, 2016, pp. 1–5.

[23] Motlagh, Y. G., Ahn, H. T., Hughes, T. J., and Calo, V. M., "Simulation of laminar and turbulent concentric pipe flows with the isogeometric variational multiscale method," *Computers & Fluids*, Vol. 71, 2013, pp. 146–155.

[24] Gramacy, R. B., and Lee, H. K. H., "Bayesian treed Gaussian process models with an application to computer modeling," *Journal of the American Statistical Association*, Vol. 103, No. 483, 2008, pp. 1119–1130.

[25] Jones, D. R., "A taxonomy of global optimization methods based on response surfaces," *Journal of global optimization*, Vol. 21, No. 4, 2001, pp. 345–383.

[26] Bazilevs, Y., Beirao da Veiga, L., Cottrell, J. A., Hughes, T. J., and Sangalli, G., "Isogeometric analysis: approximation, stability and error estimates for h-refined meshes," *Mathematical Models and Methods in Applied Sciences*, Vol. 16, No. 07, 2006, pp. 1031–1090.

[27] Alimo, S., and He, D., "Multi-stage algorithm for uncertainty analysis of solar power forecasting," *2016 IEEE Power and Energy Society General Meeting (PESGM)*, IEEE, 2016, pp. 1–5.

[28] Jalali, M., VanDine, A., Chalamalla, V. K., and Sarkar, S., "Oscillatory stratified flow over supercritical topography: Wave energetics and turbulence," *Computers & Fluids*, Vol. 158, 2017, pp. 39–48.