Efficient Global Optimization with Adaptive Target for Probability of Targeted Improvement

Anirban Chaudhuri¹, Raphael T. Haftka², Felipe A.C. Viana³

University of Florida, Gainesville, Florida, 32601

The use of Surrogate-based optimization has become increasingly prevalent in the design of engineering systems. Each optimization cycle consists of fitting a surrogate through an initial number of designs and performing optimization based on the surrogate to get a new design, where an exact simulation is performed. Algorithms like Efficient Global Optimization use uncertainty estimates available with the Kriging surrogate to guide the selection of new point(s). With access to parallel computation, adding multiple points per optimization cycle has become increasingly attractive. With EGO, Expected Improvement is commonly used as the selection criterion, but it suffers from the issue of target setting due to lack of knowledge of the true function. An adaptive target setting method is proposed in this paper, which changes the target to reflect the improvement achieved in each optimization cycle. This method is demonstrated to be highly efficient for three analytic examples. For these examples the proposed method is compared to a constant target setting and is shown to give better results. It is also seen that when multiple points are added per cycle using this method, the convergence rate becomes faster and also, the confidence in the final result becomes much higher.

Nomenclature

\[ y(x) = \text{True function at any point } x \]
\[ PI(x) = \text{Probability of targeted Improvement at any point } x \]
\[ \hat{y}(x) = \text{Kriging prediction at any point } x \]
\[ s(x) = \text{Kriging prediction standard deviation at any point } x \]
\[ y_{\text{Target}} = \text{Target value} \]
\[ y_{\text{PBS}} = \text{Present Best Solution} \]
\[ y_{\text{KRG}} = \text{Kriging Prediction} \]
\[ y_{\text{True}} = \text{True function value} \]
\[ \epsilon_{\text{S}} = \text{Exclusion radius} \]
\[ n_{\text{dim}} = \text{Number of dimensions} \]
\[ k = \text{Present EGO cycle number} \]
\[ \eta_{k} = \text{Improvement Ratio for the } k^{th} \text{ EGO cycle} \]
\[ TI = \text{Target of Improvement} \]

I. Introduction

Surrogate-based optimization is becoming increasingly popular in engineering design community due to the savings in computational time\textsuperscript{1-9}. The goal of surrogate-based optimization is to select new sampling points which would contribute towards global optimization in each cycle. Algorithms like Efficient Global Optimization (EGO)\textsuperscript{10, 11} use both the surrogate prediction and its error estimates. EGO uses Kriging prediction and prediction variance to seek the point of maximum Expected Improvement (EI)\textsuperscript{11} as the next point to be sampled in the optimization. Traditionally EGO-like algorithms add one point per cycle. Complex simulations may take weeks to complete which makes it attractive to employ parallel computation and run multiple simulations per cycle. Consequently, there has been work on including multiple points\textsuperscript{12-16}. However, selecting multiple points to maximize EI is computationally expensive\textsuperscript{12-15}. Using the Probability of targeted Improvement (that is, Probability of Improvement, PI over a given target) for finding multiple points has been shown to be much cheaper\textsuperscript{17}.  

¹ Graduate Research Assistant, Mechanical & Aerospace Engineering, a.chaudhuri@ufl.edu, AIAA Student Member
² Distinguished Professor, Mechanical & Aerospace Engineering, haftka@ufl.edu, AIAA Fellow
³ Mechanical Engineer, GE Global Research, AIAA Fellow

American Institute of Aeronautics and Astronautics

Copyright © 2012 by Anirban Chaudhuri, Raphael T. Haftka, Felipe Viana. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission.
The major problem in using PI as a selection criterion is setting the target value. If the target is too ambitious, the search is excessively global and slow to focus on promising areas. If the target is too modest, there is exhaustive search around the present best solution before moving to global search. This issue of finding the right balance for the target is why many people shirk from using EGO with PI.

In this paper we propose an adaptive target method which changes the target for every EGO cycle according to the performance of the points added in the previous cycle. The method is simple to implement and is tested with commonly used examples.

The rest of the paper is arranged as follows. Section II gives the necessary information about selection of points in EGO using PI and optimization of PI. Section III describes the adaptive target method. Section IV explains a numerical experiment using benchmark functions used to justify the method. Section V presents the results and discussions. Concluding remarks for this research are included in Section VI.

II. Background: EGO using Probability of Improvement

A brief description of Efficient Global Optimization (EGO) algorithm developed by Jones et al. is provided, followed by a description for Probability of targeted Improvement (PI) and the optimization of PI.

A. Efficient Global Optimization (EGO)

Firstly an initial set of data points is fitted with a Kriging model as a realization of a Gaussian Process with a mean of \( y(x) \) and a standard deviation of \( s(x) \). Then each cycle consists of selecting additional points based on maximizing EI or PI and refitting the surrogate. Maximizing PI is used here as the selection criteria for new sampling points. Figure 1 illustrates the EGO algorithm using a one dimensional function. After adding the new point to the existing data set, the Kriging model is updated and the process continues until a stopping criterion is met (usually number of cycles).

![Figure 1](image-url)

Figure 1. One cycle of EGO using PI for a one dimensional test function \( y(x) = (6x - 2)^2 \sin(12x - 4) \) with initial data set as \( x = [0 0.5 0.68 1]^T \). The uncertainty (amplified amplitude of \( 2s(x) \)) associated with the kriging is plotted in orange. The target, \( y_{\text{Target}} \), is set below the present best solution (PBS).
B. Probability of targeted Improvement (PI)

The probability of improving the objective beyond a target $y_{\text{target}}$ at a point $x$ is given by Equation (1)\textsuperscript{11}. This probability is the shaded area in (b) Figure 2(b) which shows the normal distribution of the kriging fit uncertainty at $x$ equal to 0.2.

$$P\,I(x) = \Phi \left( \frac{y_{\text{target}} - \hat{y}(x)}{s(x)} \right)$$

(1)

where, $\Phi(.)$ is the cumulative density function of a normal distribution, $\hat{y}(x)$ is the kriging prediction, $s(x)$ is the prediction standard deviation (square root of the kriging prediction variance).

C. Optimization of Probability of targeted Improvement

It can be seen from Figure 1 that even for a simple one dimensional function there are multiple local maxima of PI. For higher dimensions the number of local maxima could be very large. This warrants the use of a global optimizer, and here we use the stochastic Differential Evolution (DE) algorithm\textsuperscript{18}.

DE is a stochastic optimizer which gives different optimums for different runs. To increase the probability of reaching the global optimum, multiple short searches can be used rather than using the entire budget on a single long run. The basic premise is that different searches usually start with rapid initial improvement which slows down later on and may give substantially different outcomes. Schutte et al.\textsuperscript{19} showed that for stochastic global optimization it pays to use multiple starts, and this is done here.

American Institute of Aeronautics and Astronautics
D. Multiple points added per cycle

Multiple points can be added per EGO cycle to deal with the computation time associated with complex systems, especially when parallel simulations are possible. Adding multiple points increases the probability of reaching the target in any particular cycle.

Using PI multiple points can be added using a single surrogate. This is done by finding one point using the DE optimizer to maximize the PI and then constraining its nearby region using an exclusion radius, $\epsilon$, which is given by Equation (2) for the case where the design space is normalized to $[0,1]$. Then the DE optimizer is run again to find another point which lies sufficiently away from all the previous points found. This lets us add $n$ number of distinct points per cycle by repeating the process $n$ times.

$$\epsilon = 0.1 \times \sqrt{n_{\text{dim}}}$$

where, $n_{\text{dim}}$ is the number of dimensions of the function being optimized.

III. Adaptive Target method

The following section outlines the motivation behind using the proposed adaptive target method and a description of the method. The last part in this section describes an ideal target case which is used to check the effectiveness of the adaptive target method.

A. Motivation for Adaptive Target method

One way to select a target is to try for a constant percent improvement in the present best solution. This may not be reasonable as the solution nears the global optimum as is illustrated for the Hartmann 3 (Appendix A) function in Table 1. With 10% improvement target, from EGO cycle 7, the set target was beyond the global optimum of Hartmann 3 function. This makes it an impossible task to reach or better the target which in turn kept the PI values very low.

Table 1. EGO performed on a Hartmann 3 DOE with target set for 10% improvement at each cycle (Optimum for Hartmann 3: -3.86), which leads to overly ambitious targets.
B. Description of Adaptive Target method

The adaptive target methodology imitates the step-size setting in trust region methods\textsuperscript{30}. The target of improvement in cycle \(k+1\) is adjusted according to the ratio of actual to demanded improvement \(\eta_k\) in EGO cycle \(k\),

\[ \eta_k = \frac{y_{BScycle} - y_{PBS_k}}{y_{Target_k} - y_{PBS_k}} \]

where, \(y_{BScycle}\) is the best solution of the points added in the present cycle, \(y_{PBS_k}\) is the present best solution (PBS) before adding the points of the present cycle and \(y_{Target_k}\) is the target for that cycle which is given by Equation (4).

\[ y_{Target_k} = y_{PBS_k} - TI_k \]

The target of improvement (TI) is assigned an initial value for the first EGO cycle. The TI for cycle \((k+1)\) depends on the value of \(\eta_k\). If the target is met with a margin TI is increased, while if we fall short TI is decreased. This is given by Equation (5), with a linear expression being used to find value of TI in the interval.

\[ TI_{k+1} = \begin{cases} 
1.5TI_k, & \forall \eta_k > 2 \\
0.5TI_k (\eta_k + 1), & \forall \eta_k \in [0.05, 2] \\
0.5TI_k, & \forall \eta_k < 0.05
\end{cases} \]

C. Ideal Target

To check the effectiveness of the adaptive target method an ideal target case based on the known global optimum is analyzed. The target value is set below the PBS at a percentage of the distance to the global optimum. 25\% and 50\% were tried for each example and the best case was selected. Figure 3 illustrates the ideal target case for a one dimensional function.

![Figure 3. Illustration explaining Ideal target case (exaggerated scale).](image-url)
IV. Numerical Experiments

We employed the Sasena, Hartmann 3 and Hartmann 6 functions to test the adaptive target method. The Sasena function is 2 dimensional function and Hartmann functions are 3 and 6 dimensional functions which are often used as a test functions for global optimization (more description about Sasena function and Hartmann functions in Appendix A). The global optima of Sasena, Hartmann 3 and Hartmann 6 functions are known to be -1.4565, -3.86278 and -3.32237, respectively, which were used for the ideal target case.

To average out the influence of the design of experiment (DOE), 50 different Latin Hypercube DOEs were created using MATLAB function `lhsdesign`. We start sampling Sasena with 8 points, Hartmann 3 with 12 points and Hartmann 6 with 35 points and then let EGO run for 10, 18 and 35 cycles respectively. In EGO the maximization of PI is done using a Differential Evolution (DE) optimizer with 4 starts. The effectiveness of the method and the optimization process is further improved by adding multiple points per EGO cycle. The results are compared for adding 1 point, 2 points and 5 points per EGO cycle.

V. Results and Discussion

The results for Sasena, Hartmann 3 and Hartmann 6 functions compare the adaptive target method, a constant 10% improvement on the PBS as target and the ideal target case. For the adaptive target case the initial value of target of improvement (TI) is set as the absolute value of 10% of the yPBS to give it the same starting point as constant target. The ideal target uses 25% of the distance between PBS and global optimum below the PBS as the target for Hartmann functions and 50% of the distance for Sasena function. The efficiency of the method is also checked for multiple points per EGO cycle by adding 2 points and 5 points per EGO cycle. All the results are presented as a median of 50 DOEs.

The efficiency of target setting can be seen for the adaptive target and the constant target cases in Figure 4 which shows the median of target values set for each EGO cycle along with the median PBS. For Sasena function (Figure 4(a)) it can be seen that at the end of EGO cycles the targets for both adaptive target and constant target cases are very close to the global optimum which is good as the optimization is better when the set target is achievable. The advantage of the adaptive target case is evident for Hartmann 3 (Figure 4(b)) and Hartmann 6 (Figure 4(c)) functions where the target settings for the constant target case overshoot the actual global optimums by a lot. This leads to poor optimization results for both functions as it is impossible to achieve these target values. The target setting for adaptive target case on the other hand converges to the global optimum towards the end of the optimization.
Figure 4. Efficiency of target setting while adding 1 point per cycle using adaptive target and constant target for (a) Sasena, (b) Hartmann 3, and (c) Hartmann 6 functions.

The median PBS for each EGO cycle using the different methods for Sasena function when 1 point is added per cycle can be seen in Figure 5(a). It is seen that at the end of 10 EGO cycles the adaptive target method performs better than the constant target method with the optimum being comparable to the ideal target case.

For Hartmann 3 function also it is shown in Figure 5(b) that at the end of 18 EGO cycles the adaptive target method performs much better than the constant target method with the optimum being comparable to the ideal target case. The adaptive target case also shows a faster convergence as the best solution obtained using constant target is outperformed by the 9th EGO cycle using adaptive target method.

A similar trend is seen for the Hartmann 6 function in Figure 5(c). At the end of 35 EGO cycles the adaptive target method performs better than the constant target method with the optimum being comparable to the ideal target case. The best solution obtained by using constant target case is outperformed by the 23rd EGO cycle while using the adaptive target method substantiating the faster convergence seen in case of Hartmann 3.
Figure 5. Comparison of Median Present Best Solution (PBS) of constant, adaptive and ideal targets for (a) Sasena, (b) Hartmann 3, and (c) Hartmann 6 when 1 point is added per EGO cycle.

The results while adding multiple points per EGO cycle are shown in Figure 6. It is clear that adding more points per cycle substantially accelerates the convergence. In addition to the faster convergence of the median PBS, the scatter between the final global optimum obtained at the end of Ego cycles for different DOEs reduces, as shown in Table 2, Table 3 and Table 4.

American Institute of Aeronautics and Astronautics
While adding multiple points per EGO cycle ensures a much faster rate of convergence and more confidence in the final result for the global optimum obtained, there is certainly a price to pay, seen in Figure 7, as it requires a lot more expensive function evaluations as compared to adding a single point per cycle. Thus the approach pays only if the time is more important than the number of simulations and there is the ability to do parallel simulations.

Table 2. Standard deviation of PBS of last EGO cycle (10th) for different procedures for Sasena

<table>
<thead>
<tr>
<th>Method used</th>
<th>Standard Deviation of PBS of last EGO cycle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constant target of 10%</td>
<td>1.5107</td>
</tr>
<tr>
<td>Ideal target of 50%</td>
<td>1.5965</td>
</tr>
<tr>
<td>1 point adaptive target</td>
<td>1.4160</td>
</tr>
<tr>
<td>2 points adaptive target</td>
<td>0.0771</td>
</tr>
<tr>
<td>5 points adaptive target</td>
<td>0.0110</td>
</tr>
</tbody>
</table>

Table 3. Standard deviation of PBS of last EGO cycle (18th) for different procedures for Hartmann 3

<table>
<thead>
<tr>
<th>Method used</th>
<th>Standard Deviation of PBS of last EGO cycle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constant target of 10%</td>
<td>0.0369</td>
</tr>
<tr>
<td>Ideal</td>
<td>0.0136</td>
</tr>
<tr>
<td>1 point adaptive target</td>
<td>0.0257</td>
</tr>
<tr>
<td>2 points adaptive target</td>
<td>0.0014</td>
</tr>
<tr>
<td>5 points adaptive target</td>
<td>0.0011</td>
</tr>
</tbody>
</table>

Table 4. Standard deviation of PBS of last EGO cycle (35th) for different procedures for Hartmann 6

<table>
<thead>
<tr>
<th>Method used</th>
<th>Standard Deviation of PBS of last EGO cycle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constant target of 10%</td>
<td>0.0655</td>
</tr>
<tr>
<td>Ideal</td>
<td>0.0616</td>
</tr>
<tr>
<td>1 point adaptive target</td>
<td>0.1089</td>
</tr>
<tr>
<td>2 points adaptive target</td>
<td>0.0504</td>
</tr>
<tr>
<td>5 points adaptive target</td>
<td>0.0469</td>
</tr>
</tbody>
</table>
Figure 6. Median PBS for multiple points added per cycle using adaptive target for (a) Sasena, (b) Hartmann 3, and (c) Hartmann 6 functions.
VI. Concluding Remarks

We proposed an adaptive target method for EGO using probability of targeted improvement as the criterion for selecting points to be added. The adaptive target method was shown to be more efficient in global optimization as compared to a constant target for the Sasena, Hartmann 3 and Hartmann 6 functions.

Adding multiple points per EGO cycle with adaptive target method is shown to lead to a much faster convergence with more confidence in the final result as compared to adding a single point at a time for all the three functions.

Appendix A: Analytic functions

The details of Sasena function used in this research are described below. The equation for Sasena function is given by Equation (6). The number of design variables for Sasena function is 2.

\[
y(x) = 2 + 0.01(x_2 - x_1^2)^2 + (1-x_1)^2 + 2(2-x_2)^2 + 7\sin(0.5x_1)\sin(0.7x_1x_2),
\]

\[
0 \leq x_1 \leq 5, \ 0 \leq x_2 \leq 5
\]  

(6)

The Hartmann 3 and Hartmann 6 functions used in this research is described below in some more detail. The equation for Hartmann functions is given by Equation (7). For Hartmann 3 and Hartmann 6 the number of design variables, \(n_{dv}\), is 3 and 6 respectively. The parameters for Hartmann functions are shown in Table 5.

\[
y(x) = -\sum_{i=1}^{n_{dv}} a_i \exp \left\{ -\sum_{j=1}^{n_{dv}} B_{ij} x_j - D_{ij} \right\},
\]

\[
a = \begin{bmatrix} 1.0 & 1.2 & 3.0 & 3.2 \end{bmatrix},
\]

\[
0 \leq x_j \leq 1, \ j = 1, 2, \ldots, n_{dv}.
\]  

(7)
Table 5. Parameters used in Hartman functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hartman 3</td>
<td>$B = \begin{bmatrix} 3.0 &amp; 10.0 &amp; 30.0 \ 0.1 &amp; 10.0 &amp; 35.0 \ 3.0 &amp; 10.0 &amp; 30.0 \ 0.1 &amp; 10.0 &amp; 35.0 \end{bmatrix}$</td>
</tr>
<tr>
<td>Hartman 6</td>
<td>$B = \begin{bmatrix} 10.0 &amp; 3.0 &amp; 17.0 &amp; 3.5 &amp; 17.0 &amp; 8.0 \ 0.05 &amp; 10.0 &amp; 17.0 &amp; 0.1 &amp; 8.0 &amp; 14.0 \ 3.0 &amp; 3.5 &amp; 17.0 &amp; 8.0 &amp; 3.0 &amp; 10.0 \ 17.0 &amp; 8.0 &amp; 0.05 &amp; 10.0 &amp; 0.1 &amp; 14.0 \end{bmatrix}$</td>
</tr>
</tbody>
</table>

Acknowledgments

This work was supported in part by U.S. Air Force Office of Scientific Research grant FA9550-09-1-0153, National Science Foundation grant CMMI-0856431, and grant from David Stargel, Grant Monitor at the Air Force Office of Scientific Research.

References


American Institute of Aeronautics and Astronautics