Resolving Undesired Bias in Optimization of Environmentally Adaptive Control Policies

Shuhei Sugiura *, Ryo Arizumi *, and Toru Asai *

Abstract: Significant research on experiment-based black-box optimization using Bayesian optimization techniques is being performed because of its usefulness in a wide range of fields. Several algorithms using Bayesian optimization for optimizing environmentally adaptive control policies have been developed. This adaptivity is expected to be crucial for applications such as mobile robots. In this work, the unbiased expected improvement metric was the key to efficiently obtain the approximated optimal policy. The purpose of the metric was to remove the bias in sample points that is inevitable if ordinary metrics, such as the expected improvement, are used. This paper clarified the mechanism that causes the bias and showed that the bias should be attenuated to achieve efficient experiments. Based on the understanding of the mechanism, a simple solution was proposed to attenuate this bias. Using numerical tests, it was shown that our method effectively attenuated the bias and that this led to better optimization performance in that it often required less samples than the existing method.

Key Words: efficient global optimization, Bayesian optimization, unbiased expected improvement, adaptive control policy.

1. Introduction

When optimizing robotic systems, it is rare that we have access to the accurate analytical form of the objective function. Therefore, we often need experiments or simulations to evaluate the objective function. Because of the economical and temporal cost of experiments, algorithms have been developed to reduce the number of necessary experiments. Bayesian optimization (BO) methods [1], which are data-driven iterative optimization algorithms, are expected to be useful for such “black-box” optimization.

In BO methods, a sample-based estimation of the objective function, which is called a surrogate function, is constructed and used instead of the true unknown objective function. Many metrics have been proposed to realize this efficiency. These metrics choose the next observation point by balancing between exploration (sampling at a region with large uncertainty) and exploitation (sampling at a region in which we are sure the objective function has a good value). Expected improvement (EI) is a metric that achieves this balance without any tuning parameters, unlike metrics such as the probability of improvement [2] or the upper confidence bound (UCB) [3]. Although some results showed the possibility of auto-tuning the parameters in UCB [3], EI is very popular because of its simplicity and efficiency. The use of EI was popularized by Jones et al.[4], and while their algorithm did not explicitly consider the noise in observations, it was shown that by using Gaussian process regression (GPR), it worked well with noisy samples [5]–[7]. Several studies have extended the algorithm to multiobjective settings [8],[9].

Among the studies that extended [4], Tesch et al.[10] proposed a method to make it adaptive to environmental change. They introduced the space of environmental parameters $X_e$ and control parameters $X_c$, and defined the system output as a mapping $f : X_e \times X_c \rightarrow \mathbb{R}$. They also defined the control policy $\gamma$ as a mapping from $X_e$ to $X_c$. The purpose of their method was to optimize the function $\gamma$ using a restricted number of experiments. In other words, it aimed at finding the best $\gamma$ that attained the maximum of $f(x_e, \gamma(x_e))$ for any $x_e \in X_e$. To this end, they defined the policy score to evaluate $\gamma$ and proposed two metrics for experimental planning: the expected policy score improvement (EPSI) and the unbiased expected improvement (UEI). The efficacy of their method has been shown by numerical tests and experiments with real robots [11].

In this paper, we showed that if a standard GPR is used to make the surrogate, there is still a bias with UEI that needs to be removed. The attenuation of this bias is important because any unintended bias results in an inefficient search. The cause of the bias was analyzed and, based on the analysis, a method to attenuate the bias was proposed. Numerical examples showed that the efficiency of the algorithm increased in that a smaller number of experiments will be required using our method than UEI-based optimization. Because of the computational cost of EPSI, only UEI was tested. However, as the problem is not in the metric itself but in the regression, we are sure that the same problem exists in EPSI and a similar solution will work. Some parts of this paper were presented in [12], and in this paper, we add a more detailed explanation on the cause of the bias and give different numerical examples.

This paper is organized as follows: In Section 2, the problem of optimization of the environmentally adaptive control policies is briefly explained. The basic algorithm and the definition of UEI are also given in this section. In Section 3, we show results of numerical tests that indicate the existence of bias in UEI-
Based experimental planning, discuss the cause of this bias, and propose a simple solution. Numerical tests with a known test function are performed in Section 4 to show the efficacy of our method, and Section 5 concludes this paper.

2. Optimization of Environmentally Adaptive Control Policies

In this section, we formalize the problem of the optimization of environmentally adaptive control policies. For notations, we follow Tesch et al.[10].

2.1 Notations and Problem Statement

First, we need to define the control parameters and the environment parameters. The control parameters are the parameters that are to be optimized, and users can specify them at any time. Control parameters include, for example, gains in a proportional-integral-differential (PID) controller. The environment parameters, in contrast, can only be measured and cannot be tuned by users except in an arranged environment like an experimental room. Environment parameters include, for example, the terrain steepness where a mobile robot moves. Although users cannot actively tune the steepness in real usage, it is assumed that in experiments we can control these parameters. For example, we assume that our experimental room is equipped with a slope with variable steepness. The spaces of these two types of parameters are defined as follows:

Definition 1 (Control Parameter) The control parameter space $X_c$ is a compact subset of $\mathbb{R}^{m_c}$, where $m_c$ is the number of control parameters.

Definition 2 (Environment Parameter) The environment parameter space $X_e$ is a compact subset of $\mathbb{R}^{m_e}$, where $m_e$ is the number of environment parameters.

The output, i.e., the objective function, of the system in our problem is then defined as follows:

Definition 3 (System Output) The system output $f : X_e \times X_c \rightarrow \mathbb{R}$ is a continuous, real-valued function which encodes the system performance.

The evaluation of the function $f$ is assumed to be time-consuming or financially expensive. The control policy is now defined as follows:

Definition 4 (Control Policy) The control policy is a mapping $\gamma : X_e \rightarrow X_c$.

Note that $\gamma$ is not necessarily surjective nor continuous.

The purpose of the optimization of the environmentally adaptive control policy is to find $\gamma^*$ that satisfies:

$$\gamma^*(x_e) = \arg\max_{x_c} f(x_e, \gamma(x_e)).$$

(1)

To evaluate the performance of $\gamma$, the policy score $S$, which is a functional of $\gamma$, is defined as follows:

$$S[\gamma] = \int_{X_e} \omega(x_e) f(x_e, \gamma(x_e)) dx_e,$$

(2)

where $\omega$ is a weight function that expresses the importance of each environment. Then, the problem is to find $\gamma^*$ that attains the maximum of $S$, and we want to achieve this with as few samples as possible.

Note that the weight function $\omega$ does not change the optimal policy. However, it is important to design $\omega$ appropriately to make the optimization procedure efficient in that a larger number of experiments is devoted to the search in more important environments. For such design to be possible, there should not be any bias in the search trend to any specific environment if $\omega = 1$ for all $x_e \in X_e$.

Figure 1 shows an example of the system output and the optimal policy $\gamma^*$. In this example, both environment and control parameters are one-dimensional and the optimal policy $\gamma^*$, which is shown by the thick lines, is not surjective nor continuous.

2.2 Black-Box Optimization Algorithm

The black-box optimization of environmentally adaptive control policies is depicted in Algorithm 1. In Line 2, the initial set of experiments are planned through Latin hypercube design [13]. The initial experiments are performed in Lines 4-6 and the input-output pairs are recorded. In Line 8, the surrogate function $\hat{f}$ is constructed by regression based on Bayesian inference: any regression that returns a predictive distribution can be used, but GPR is one of the most common choices for the regression. Based on the regression results, the estimation of the optimal control policy $\hat{\gamma}^*$ is obtained and UEI, whose details are

Algorithm 1 Black-box optimization of environmentally adaptive control policies using UEI

1: Given
   $N_i$: the number of initial sample points
   $N_{\text{max}}$: the maximum number of experiments/simulations
   $f$: the objective function evaluated through expensive experiments/simulations

2: $X \leftarrow \{(x_e^{(1)}, x_c^{(1)}), ..., (x_e^{(N_i)}, x_c^{(N_i)})\}$: use Latin hypercube design
3: for $k = 1$ to $N_i$ do
4:   $y \leftarrow f(x_e^{(k)}, x_c^{(k)})$
5:   $Y \leftarrow \{(Y, y)\}$
6: end for
7: for $j = 1$ to $N_{\text{max}} - N_i$ do
8:   Perform regression to make surrogate $\hat{f}$
9:   Calculate the estimated optimal policy $\hat{\gamma}^*$
10: $(x_e^{(\text{new}), x_c^{(\text{new})}}) \leftarrow \arg\max_{(x_e, X_e^\mu) \in X_e} \text{UEI}(x_e, X_e^\mu)$
11: $y \leftarrow f(x_e^{(\text{new}), x_c^{(\text{new})}})$
12: $X \leftarrow \{(X_e, x_e^{(\text{new}), x_c^{(\text{new})})}, Y \leftarrow \{(Y, y)\}$
13: end for
14: return $\hat{\gamma}^*$

Fig. 1 An example of the output of a system with one environment and one control parameter. Thick lines show the optimal policy.
given in Section 2.4, is calculated. Note that no model on \( y \) is required to estimate \( \hat{y}^* \). All the necessary information is given by the regression of the objective function. The maximizer of UEI evaluated in the next step. The procedure is continued until the number of experiments reaches its predefined maximum.

### 2.3 Gaussian Process Regression

To make the surrogate function \( \hat{f} \), we use GPR, which is reviewed here. For more detail, please refer to [14]. Let \( y \) and \( f \) be defined as the vectors whose \( i \)th components are \( y_i \) and \( f_i = f(x_i) \), respectively. We assume the following:

\[
y = f(x) + \epsilon,
\]

\[
f(x) \sim \mathcal{GP}(m(x), k_f(x, x')),
\]

\[
\epsilon \sim \mathcal{N}(0, \sigma^2_\epsilon),
\]

where \( x \sim \mathcal{P} \) means that a random variable \( x \) is taken from a distribution or a stochastic process \( \mathcal{P} \), \( \mathcal{GP}(m, k_f) \) is a Gaussian process with mean function \( m \) and covariance function \( k_f \), and \( \epsilon \) is a noise term. The function \( k_f \) is a user-defined kernel function that expresses the correlation between two points \( x, x' \in X \). If we have prior knowledge about \( f \), such as its periodicity, it can be incorporated by providing an appropriate kernel function. One of the most popular kernel functions is the squared exponential kernel:

\[
k_f(x_i, x_j) = \sigma_f^2 \exp \left(-\frac{1}{2} (x_i - x_j)^T M (x_i - x_j) \right),
\]

where \( M = \text{diag}(l_1^2, \ldots, l_D^2) \). Parameters \( \sigma_f, \sigma_y \), and \( l_i (i = 1, \ldots, D) \) are called the hyperparameters. In the following calculations, we use \( m(x) \equiv 0 \) to keep the calculation simple. However, in real applications, this will be used to incorporate our prior knowledge.

The regression problem finds appropriate values of the hyperparameters to explain the data. This is usually achieved through the maximization of the marginal likelihood:

\[
p(y) = \int p(y|\theta)p(\theta)\,d\theta = \mathcal{N}(0, K + \sigma^2_\epsilon I),
\]

where \( I \) is the identity matrix of dimension \( N \) and \( K \) is a kernel matrix whose \((i, j)\) component is \( k(x_i, x_j) \). Once the hyperparameters are determined, the predictive distribution of \( y \) given \( x \) is given by

\[
y\mid x \sim \mathcal{N}(\mu_y, \sigma^2_y),
\]

\[
\mu_y = k_f^T (K + \sigma^2_\epsilon I)^{-1} y,
\]

\[
\sigma^2_y = k_{xx} - k_f^T (K + \sigma^2_\epsilon I)^{-1} k_f,
\]

where \( k_{xx} = [k(x_1, x_1) \cdots k(x_h, x_h)]^T \) and \( k_{xx} = k(x, x) \).

### 2.4 Unbiased Expected Improvement

The standard EI is defined as follows:

\[
\text{EI}(x) = \int_{y_{\text{max}}}^{\infty} (y - y_{\text{max}}) p(y \mid x) \, dy
\]

\[
= (\mu - y_{\text{max}}) \Phi \left( \frac{\mu - y_{\text{max}}}{\sigma} \right) + \sigma \phi \left( \frac{\mu - y_{\text{max}}}{\sigma} \right),
\]

where \( x = (x_c, x_e) \), if the predictive distribution at \( x \) is \( \mathcal{N}(\mu, \sigma^2) \). In the above equation, \( \Phi(x) \) is the cumulative distribution function, and \( \phi(x) \) is the probability density function of the standard Gaussian distribution (i.e., \( \mathcal{N}(0, 1) \)) evaluated at \( x \). By \( y_{\text{max}} \) is either the maximum of the sampled objective values or the estimated maximum. In usual BO, the maximizer of \( \text{EI}(x) \) is selected as the query point in the next step. However, if this is used in the optimization of environmentally adaptive control policies, the search trend is inevitably biased to the environments with high objective values [10]. In other words, it will search for experiments in easy environments more often than in difficult ones.

UEI was proposed to get rid of this bias and is defined as follows [10]:

\[
\text{UEI}(x) = \omega(x_c) \int_{y_{\text{max}}}^{\infty} (y - y_{\text{max}}) \, dy
\]

\[
= \omega(x_c) \left[ (\mu - y_{\text{max}}) \Phi \left( \frac{\mu - y_{\text{max}}}{\sigma} \right) + \sigma \phi \left( \frac{\mu - y_{\text{max}}}{\sigma} \right) \right],
\]

\[
y_{\text{max}} = f(\hat{x}_c, \hat{y}^*(x_c)).
\]

The idea is that by defining EI for every environment independently from all others, we can avoid comparing environments with different difficulties on the same scale. The maximizer of UEI is selected as the query point in the next step.

### 3. Bias in UEI-based Optimization

In [12], we used a numerical example to show that there is another kind of bias left in Tesch’s method and explained the reason for the bias. Roughly speaking, Tesch’s UEI-based method prefers searching environments with smaller function ranges if the weight function is set as \( \omega(x_c) = 1 \) for all \( x_c \in X_c \). As we mentioned in Section 2.1, such bias is problematic because this makes it very difficult to design \( \omega(x_c) \) appropriately without knowing the true objective function. Here, we explain the cause of the bias in more detail and propose a method to attenuate the bias.

#### 3.1 Cause of the Bias

The bias is caused by the assumption of stationarity and homoscedasticity in the commonly used kernel functions such as (4). Stationarity means that the length scales \( l_i \) in (4) are constant, and homoscedasticity means that the noise variance \( \sigma^2_n \) is constant. Without these assumptions, the marginal likelihood becomes analytically intractable and the second equity of (5) does not hold anymore. While the assumptions make the regression problem simple, this leads to under- or over-estimation of UEI.

To demonstrate the under- or over-estimation of UEI, consider a regression of a simple function:

\[
\tau(x_c, x_e) = (1 + \sin \pi x_c + 0.1 x_e)(20 x_c + 1).
\]

In Fig. 2, we show the graph of the function \( \tau \). We evaluated the function at evenly distributed points and performed GPR. The standard deviation of the predictive distribution calculated by (6) is shown in Fig. 3. In the figure, asterisks represent the sampled inputs. It can be seen that the standard deviation depends solely on the distribution of the sampled points and not on the function value. The mechanism of the station-
The predictive density functions at \((x_e, x_e) = (0.75, 0.6)\) are almost equal, i.e. UEI should be close at those points. However, by noting that UEI is the expectation value of the improvement, it is apparent that UEI at \((x_e, x_e) = (2.25, 0.6)\) can be smaller than that at \((x_e, x_e) = (0.75, 0.6)\) because there is only a small probability for \(f(2.25, 0.6)\) to beat \(\hat{f}(2.25, x_e)\). The UEI is shown in Fig. 5, from which we can confirm that UEI was higher at \(x_e = 0.75\) than \(x_e = 2.25\).

The cause of the same level of uncertainty regardless of the function shape can be understood in the following way. Because of the assumptions of stationarity and homoscedasticity, the kernel function \(k(x, x')\) depends only on \(x - x'\) and does not depend on the absolute position of \(x\) and \(x'\). Therefore, \(k_0\) and \(K\) in (6) are determined only by the distribution of inputs \(\{x_{1}, \ldots, x_N\}\) and \(x_e\). Because the kernel function between points usually decays quickly as the distance grows, the predicted variance at \(x_e\) is determined mainly by the distribution of neighboring sample inputs. As a consequence, the predictive uncertainty becomes close for points \(x^{(1)}_k\) and \(x^{(2)}_k\) if the distribution of sampled inputs are in their vicinities.

For further analysis, consider an imaginary case where there are two sets of sample points \(D^{(1)} = \{x^{(1)}_1, \ldots, x^{(1)}_N\}\) and \(D^{(2)} = \{x^{(2)}_1, \ldots, x^{(2)}_N\}\). Assume that these two sets of points satisfy \(\max_{i,j} k(x^{(1)}_i, x^{(2)}_j) \approx 0\), i.e., the two sets are far apart. Then the kernel matrix \(K\) for \(D = D^{(1)} \cup D^{(2)}\) becomes

\[
K \approx \begin{bmatrix}
K^{(1)} & O \\
O & K^{(2)}
\end{bmatrix},
\]

\[
K^{(i)} = \begin{bmatrix}
k(x^{(1)}_1, x^{(1)}_1) & \cdots & k(x^{(1)}_1, x^{(1)}_N) \\
\vdots & \ddots & \vdots \\
k(x^{(1)}_N, x^{(1)}_1) & \cdots & k(x^{(1)}_N, x^{(1)}_N)
\end{bmatrix}, \quad i = 1, 2.
\]

Suppose that the two sets are related by \(x^{(2)}_k = x^{(1)}_k + \delta\), where \(\delta\) is a constant vector. Then because \(x^{(1)}_k - x^{(1)}_k = x^{(2)}_k - x^{(2)}_k\), we have \(K^{(1)} = K^{(2)}\). Take two test points \(x^{(1)}_k\) and \(x^{(2)}_k\) which satisfy the following conditions: (i) \(x^{(1)}_k\) is close to \(D^{(1)}\) and far from \(D^{(2)}\), i.e., for at least one of \(x^{(1)}_k\), \(k(x^{(1)}_k, x^{(1)}_l)\) is not negligible, but for all \(x^{(2)}_k\), \(k(x^{(2)}_k, x^{(2)}_l)\) are negligible; (ii) \(x^{(2)}_k = x^{(1)}_k + \delta\), i.e., the two sets \(D^{(1)} \cup \{x^{(1)}_k\}\) and \(D^{(2)} \cup \{x^{(2)}_k\}\) are related by a translation. Under these settings, because \(k(x^{(1)}_k, x^{(1)}_l) = k(x^{(2)}_k, x^{(2)}_l)\), we can conclude that the variances of the predictive distributions at \(x^{(1)}_k\) and \(x^{(2)}_k\) are the same.

In practice, things are much more complicated and the above discussion will not be applied directly. However, a similar effect will be eminent especially at the early stage of the BO procedure, where the sampled points are distributed more or less uniformly.

To see that UEI is actually higher in the environment of Fig. 4 (a) than Fig. 4 (b), suppose that in two different environ-
On the one hand, it is clear that
\[ y_i = \mu(\mathbf{x}_i) \]
and \[ y_i = y_i' = \hat{f}(\mathbf{x}_i, \gamma(\mathbf{x}_i)) = \mu(\mathbf{x}_i, \gamma(\mathbf{x}_i)). \]
Then, the following theorem holds:

**Theorem 1** Under conditions (12) and (13), it holds that

\[ \text{UEI}(\mathbf{x}_1, \mathbf{x}_2) \geq \text{UEI}(\mathbf{x}_1', \mathbf{x}_2') \]

if \( \omega(\mathbf{x}_1) = 1 \) for all \( \mathbf{x}_1 \in X_e \). The equality holds if and only if \( r = 1 \) or \( \mu(\mathbf{x}_1', \mathbf{x}_2') = \mu(\mathbf{x}_1, \gamma(\mathbf{x}_1')). \)

**Proof.** From (8), we have

\[
\text{UEI}(\mathbf{x}_1, \mathbf{x}_2) = \sigma \left[ \frac{\mu(\mathbf{x}_1) - y_1}{\sigma} + \phi \left( \left( \frac{\mu(\mathbf{x}_1) - y_1'}{\sigma} \right) \right) \right]
\]

\[
= \sigma \left[ \frac{\mu(\mathbf{x}_1) - y_1}{\sigma} + \phi \left( \left( \frac{\mu(\mathbf{x}_1) - y_1'}{\sigma} \right) \right) \right]
\]

\[
\text{UEI}(\mathbf{x}_1', \mathbf{x}_2') = \sigma \left[ \frac{\mu(\mathbf{x}_1') - y_1'}{\sigma} + \phi \left( \left( \frac{\mu(\mathbf{x}_1') - y_1'}{\sigma} \right) \right) \right]
\]

where \( z_1 = \left( \frac{\mu(\mathbf{x}_1) - y_1'}{\sigma} \right) \). From the definition of \( y_1' \), it holds that \( z_1 \leq 0 \). Let \( d(z, r) \) be defined as

\[
d(z, r) = \text{UEI}(\mathbf{x}_1, \mathbf{x}_1') - \text{UEI}(\mathbf{x}_1', \mathbf{x}_2') = \sigma \left[ \frac{\mu(\mathbf{x}_1) - y_1}{\sigma} + \phi \left( \left( \frac{\mu(\mathbf{x}_1) - y_1'}{\sigma} \right) \right) \right]
\]

\[
= \sigma \left[ \frac{\mu(\mathbf{x}_1) - y_1}{\sigma} + \phi \left( \left( \frac{\mu(\mathbf{x}_1) - y_1'}{\sigma} \right) \right) \right]
\]

On the other hand, it is clear that \( d(z, 1) = 0 \) for all \( z \leq 0 \). On the other hand, we have

\[
\frac{\partial d}{\partial r} = \sigma \left[ -z_1 \phi(r_z) + r_z \phi(r_z) - z_1 \frac{d}{dz} \phi(z) \right]_{z = z_1}
\]

\[
= -\sigma z_1 \phi(r_z) > 0,
\]

where we used \( \phi(z)/dz = \phi(z) \) and \( \phi(z)/dz = -\phi(z) \). Therefore, it is concluded that \( d(z, r) \geq 0 \). Moreover, because \( \phi(z) > 0 \) for all \( z \), it can be seen that \( d(z, r) = 0 \) \( \iff r = 1 \) or \( z = 0 \), which concludes the proof. \( \square \)

### 3.2 A Solution for the Bias

Because the cause of the bias is in the use of the same hyperparameters in all environments, a straightforward solution will be to use non-stationary GPRs [15] or heteroscedastic GPRs [16]. Non-stationary and/or heteroscedastic GPRs have already been used in BO [17] and these methods will be directly used to solve the problem. However, these non-standard GPRs necessarily have a larger number of tuning parameters than the standard GP. Therefore, if a non-stationary GPR is used, it is estimated that we need a larger number of samples to avoid overfitting. As it is crucial for our solution to reduce the required number of experiments, any solution that does not increase the complexity of the regression model will be more desirable.

Note that the direct problem is not the non-stationarity or homoscedasticity of the GPR, but the over- or under-estimation of priority for next iteration. Because UEI overestimates the priority where the range of the objective is small and underestimates it where it is large, the UEI can be adjusted according to the expected value range at each environment. Letting the estimated interquartile range at environment \( x_e \) be \( i_e(x_e) \), we use the following adjusted UEI (AUEI) instead of the standard UEI:

\[
\text{AUEI}(x_e) = i_e(x_e) \text{UEI}(x_e, x_e).
\]

The estimation of \( i_e \) is made by sampling the surrogate function at a number of control parameters that are distributed uniformly at each environment. After taking random samples, the 25 percentile and the 75 percentile of the objective values are calculated. Then the difference between them is defined as \( i_e \). Note that the evaluation of the surrogate function is easy and the computational cost for this procedure is almost negligible. Although the number of required samples will grow with the dimensionality of \( d \) of \( x_e \), empirically, around 100d points will work well for this purpose. This is because the estimate is based on Monte Carlo with independent samples and it is not necessary to estimate \( i_e \) in high precision.

The effect of multiplication by \( i_e \) is to normalize function range at every environment in some sense. For the similar purpose, order-preserving transformations are proposed in [18]. However, their methods are not applicable in our case because they need samples from \( p(x_e|x_e) \), on which we have almost no information.

### 4. Numerical Validation

To validate the efficacy of AUEI, we used four 2-input-1-output test functions \( S_1, S_2, S_3, \) and \( S_4 \). The test functions \( S_1, S_2, \) and \( S_3 \) are shown in Fig. 6 and \( S_4 \) is shown in Fig. 1. \( S_1, S_2, \) and \( S_3 \) were introduced in [10]. The test functions \( S_3 \) and \( S_4 \) were made by us for the test. The variations in function ranges for the test functions are shown in Fig. 7. In the test functions \( S_3 \) and \( S_4 \), the variation range of the functions increases approximately linearly as the environmental variable increases. In the test functions \( S_1 \) and \( S_2 \), there is no proportional relationship between the environmental variable and the variation range of the functions. Note that the variation in function range is much smaller in \( S_1 \) and \( S_2 \) compared with that of \( S_3 \) and \( S_4 \). Therefore, it is expected that the performance of the UEI-based search is relatively high for the functions \( S_1 \) and \( S_2 \) but decreases for \( S_3 \) and \( S_4 \). We performed 100 trials with these test functions using AUEI and UEI. In each trial, we selected 30 points for the initial evaluations through Latin hypercube design and performed 30 additional evaluations. We used \( \omega(\mathbf{x}_e) = 1 \) to make the effect of AUEI clearer. Note that by definition, this is applied not only to UEI but also to AUEI. To find the maximizer of UEI or AUEI, any optimization method can be used such as gradient methods with random restart. In the tests, we first calculate them at 1.0 \( \times \) 10\(^4\) points and used the maximizer of them as the initial point for Matlab \texttt{fmincon} function. For the estimation of \( i_e \), 200 uniform random points are sampled.

The histogram in Fig. 8 shows the distribution of additional samples with the test function \( S_1 \). From Fig. 8, it can be seen that UEI tended to choose points near \( x_e = 0 \), where the range

---

1 Some empirical evidences can be found at: https://nuss.nagoya-u.ac.jp/\{p5ZezykAGjB2zJ

2 The exact definition of the functions is omitted because they are quite lengthy. However, readers can find them at: https://nuss.nagoya-u.ac.jp/index.php/s/TKfSXXBV8d5kV

---
Fig. 6 Test functions $S_1$, $S_2$, and $S_3$. The thick lines refer to the optimal policy (ground truth). Note that test function $S_4$ is shown in Fig. 1.

Fig. 7 Ranges of the test functions. In every test function, the range changes as the environment changes.

Fig. 8 Distributions of additional samples.

Fig. 9 Comparison of the policy score between the AUEI- and UEI-based methods. Thick lines are medians and the colored area shows the ranges between 25 and 75 percentiles. For $S_2$, the median of the EPSI based method is also shown, which is based on 10 trials.

Table 1 Mean time consumption of one trial in optimizing function $S_2$ (s).

| Method      | UEI based | AUEI based | EPSI based |
|-------------|-----------|------------|------------|
| Time (s)    | 74.94     | 120.8      | $1.299 \times 10^5$ |

of the function is small. However, there was no excessive concentration of evaluation points when AUEI was used. From this, it can be concluded that the bias that exists in the search based on UEI was attenuated by using AUEI.

In Fig. 9, we plot the policy score calculated by (2) at each iteration. The thick lines indicate the median and the shaded areas show the ranges between 25 and 75 percent. Note that because the distributions of the policy score at each iteration can be far from symmetric, we believe that this is a better illustration than the mean and standard deviation. In the result with the test functions $S_2$, $S_3$, and $S_4$, the AUEI-based method clearly outperformed the UEI-based method. From these results, we conclude that attenuating the bias in the distribution of sampled points can improve the performance of the optimization procedure if the function range differs with the environment.

For the function $S_2$, we also run EPSI [10] based optimization for 10 trials. The mean time consumption for one trial is shown in Table 1. Clearly, EPSI is much more time-consuming than the others, which is the reason why we could not perform 100 trials. Although the simple comparison is not possible because of the shortage of the trials, we plot the median of the
policy score for the EPSI based method in Fig. 9 (b). From this figure, we cannot see any evidence that the EPSI based method outperforms the others. All the tests were performed by Matlab 2016b running on a computer with Core i7-6700 and 8 GB RAM.

With the test function $S_1$, the median of the AUEI-based method was slightly lower than with the UEI-based method. To check the significance of the difference, we performed a one-sided Wilcoxon signed-rank test. The null hypothesis is that the AUEI-based method is better than or equal to the UEI-based method in the sense of the median of the policy score after 30 additional evaluations. The test failed to reject the null hypothesis at the 5% significance level ($p = 0.12$). Moreover, it is clear that the 25th percentile for the AUEI-based method was larger than that for the UEI-based method. Therefore, it can be concluded that the AUEI-based method will not be worse than UEI-based one for the optimization of $S_1$.

To check the validity of our discussion at Section 3.1, we calculate UEI for $S_1$ after initial observations. The distribution of the median of UEI ($x_{c1}, x_{c2}$) and UEI ($x_{c}, x_{l}$) are shown in Fig. 10, where $x_{c1} = 0.4$ and $x_{c2} = 9.6$. It can be seen that there is a tendency that the median of UEI is larger in $x_{c1}$, which verifies the discussion in Section 3.1. Note that median is expected to have a larger impact on the search trend than maximum values. This is because peaks in UEI will quickly be diminished by additional several observations if the sampling is designed by UEI maximization. However, if UEI has high plateaus, whose values can be estimated by taking the median, it will require many more samples to make them lower.

5. Optimization of Snake Locomotion

In this section, we consider the maximization of the head speed of a 10-link planar snake robot based on simulations. The length and the mass of each link are 0.12 m and 0.182 kg, respectively. The robot is controlled to achieve the given reference shape, which is defined by the following equation:

$$\phi_i = \frac{2\pi T}{10} \alpha \sin \left( vt - \frac{2\pi T i}{10} \right), \quad i = 1, 2, ..., 9,$$

(20)

where $\phi_i$ is the angle of $i$th joint. There are three control parameters $\alpha$, $v$, and $T$. This is often employed as the body shape of snake robots [19],[20] and experimentally shown to be close to the shape of real snakes [21]. For the simulation model and control strategy, please see [19],[20], for example.

It is known that the ratio between friction coefficients in longitudinal and lateral directions is critical for locomotion performance [19],[20]. In this test, we set the viscosity in longitudinal direction $c_t$ and lateral direction $c_n$ as

$$c_t = x_t, \quad c_n = 1.0, \quad x_t \in [0.0, 1.0].$$

(21)

The domain of the control parameters are set as follows:

$$\alpha \in [0, \pi/2], \quad v \in [0, 1.5], \quad T \in [0.1, 2.0].$$

(22)

We run the simulation for 20 seconds and calculate the average speed of the head. The motion of the snake robot is depicted in Fig. 11, where the environment and control parameters are set as $x_t = c_t = 0.2, \alpha = \pi/4, v = 1.5$, and $T = 1$.

In Fig. 12, the resultant policy score is plotted for UEI based case and AUEI based case. It will be clear that our AUEI outperforms UEI in that it reaches higher policy score with a smaller number of samples. This suggests that, also in real situations, AUEI will be the better choice compared to UEI especially if the environment parameters known to affect the performance.

6. Conclusion

In this paper, we considered the black-box optimization of environmentally adaptive control policies. It was shown that the search trend when using UEI is biased in an unintended way if the standard GPR is used to make the surrogate function. The mechanism that causes this bias was explained and as a simple solution we proposed the AUEI metric. Although our solution is very simple, it worked well to attenuate the bias. Moreover, numerical tests and a simulation of a snake robot showed that it improved the efficiency of the optimization procedure in terms of attainable policy score at a fixed number of samples.
However, all the objective functions we used, including that of the simulation, have some variation in their range with the environment and it is not clear whether our method will perform well for functions with constant function range. Another unknown is whether a non-stationary or heteroscedastic GPR can solve the problem. We conjecture that they will be able to attenuate the bias but require more samples because they have larger complexity than the standard GPR, and this determination will be part of our future work. In the present paper, we only tested our method with known 2-input-2-output test functions. In the future, we intend to perform more tests with other functions, especially those with more inputs and with noise, and real robot experiments.

Acknowledgments

This work was supported by a research grant from The Mazda Foundation. We thank Melissa Gibbons, PhD, from Edanz Group (www.edanzediting.com/ac) for editing a draft of this manuscript.

References

[1] E. Brochu, V. Cora, and N. de Freitas: A tutorial on Bayesian optimization of expensive cost functions, with application to active user modelling and hierarchical reinforcement learning, Tech. Rep., University of British Columbia, UBC TR-2009-023, 2010.

[2] A. Zilinskas: A review of statistical models for global optimization, *J. Glob. Optim.*, Vol. 2, pp. 145–153, 1992.

[3] N. Srivivas, A. Krause, S.M. Kakade, and M.W. Seeger: Information-theoretic regret bounds for Gaussian process optimization in the bandit setting, *IEEE Trans. Information Theory*, Vol. 58, No. 5, pp. 3250–3265, 2012.

[4] D.R. Jones, M. Schonlau, and W.J. Welch: Efficient global optimization of expensive black-box functions, *J. Global Optim.*, Vol. 13, No. 4, pp. 455–492, 1998.

[5] D. Lizotte, T. Wang, M. Bowling, and D. Schuurmans: Automatic gait optimization with Gaussian process regression, *Proc. Int. Joint Conf. Artificial Intell.*, Hyderabad, India, pp. 944–949, 2007.

[6] M. Tesch, J. Schneider, and H. Choset: Using response surfaces and expected improvement to optimize snake robot gait parameters, *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, San Francisco, CA, pp. 1069–1074, 2011.

[7] R. Calandra, A. Seyfarth, J. Peters, and M.P. Deisenroth: Bayesian optimization for learning gaits under uncertainty, *Ann. Math. Artif. Intell.*, Vol. 76, pp. 5–23, 2016.

[8] J. Knowles: ParEGO: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems, *IEEE Trans. Evol. Comput.*, Vol. 10, No. 1, pp. 50–66, 2005.

[9] R. Arizumi, M. Tesch, K. Kato, H. Choset, and F. Matsuno: Multiobjective optimization based on expensive robotic experiments under heteroscedastic noise, *IEEE Trans. Robot.*, Vol. 33, No. 2, pp. 468–483, 2017.

[10] M. Tesch, J. Schneider, and H. Choset: Adapting control policies for expensive systems to changing environments, *Proc. IEEE/RSJ Int. Conf. Intel. Robot. Syst.*, pp. 357–364, 2011.

[11] C. Gong, M. Tesch, D. Rollinson, and H. Choset: Snakes on an inclined plane: Learning an adaptive sidewinding motion for changing slopes, *Proc. IEEE/RSJ Int. Conf. Intel. Robot. Syst.*, pp. 1114–1119, 2014.

[12] R. Arizumi, S. Sugiuara, and T. Asai: Bias in optimization of environmentally adaptive control policies and a solution thereof, *Proc. SICE Annual Conf.*, pp. 1533–1538, 2017.

[13] M.D. Mackay, W.J. Conover, and R.J. Beckman: A comparison of three methods for selecting values of input variables in the analysis of output from a computer code, *Technometrics*, Vol. 21 No. 2, pp. 239–245, 1979.

[14] C.E. Rasmussen and C.K.I. Williams: *Gaussian Processes for Machine Learning*, The MIT Press, 2006.

[15] C. Paciorek and M. Schervish: Nonstationary covariance functions for Gaussian process regression, *Proc. Advances in Neural Information Processing Systems*, Vol. 16, pp. 273–280, 2004.

[16] M. Lázaro-Gredilla and M.K. Titsias: Variational heteroscedastic Gaussian process regression, *Proc. Int. Conf. Machine Learning*, Bellevue, WA, pp. 841–848, 2011.

[17] J.-A.M. Assael, Z. Wang, B. Shahriari, and N. Freitas: Heteroscedastic treed Bayesian optimization, arXiv preprint arXiv:1410.7172, 2014.

[18] S. Shirakawa: Impact of invariant objective for order preserving transformation in Bayesian optimization, *Proc. Int. Congress on Evolutionary Computation*, pp. 1432–1437, 2016.

[19] R. Ariizumi and F. Matsuno: Dynamic analysis of three snake robot gaits, *IEEE Transactions on Robotics*, Vol. 33, No. 5, pp. 1075–1087, 2017.

[20] P. Liljebäck, K. Pettersen, Ø. Stavdahl, and J. Gravdahl: Controllability and stability analysis of planar snake robot locomotion, *IEEE Trans. Autom. Control*, Vol. 56, No. 6, pp. 1365–1380, 2011.

[21] S. Hirose: *Biologically Inspired Robots: Snake-Like Locomotors and Manipulators*. Oxford University Press, 1993

---

**Shuhei Sugiuara**

He received his B.S. degree from Nagoya University, Japan, in 2017. His research interests include the optimization of robotic systems.

**Ryo Arizumi (Member)**

He received his B.S., M.S., and Ph.D. degrees from Kyoto University, Kyoto, Japan, in 2010, 2012, and 2015, respectively. In 2015, he was a Postdoctoral Researcher at Kyoto University. He is currently an Assistant Professor at Nagoya University, Nagoya, Japan. His research interests include the control of redundant robots and the optimization of robotic systems.

**Toru Asai (Member)**

He received his B.E., M.E., and Ph.D. degrees from the Tokyo Institute of Technology, Japan, in 1991, 1993, and 1996, respectively. He worked as a Research Fellow at JSPS from 1996 to 1998. In 1999, he joined the faculty of Osaka University. He is presently an Associate Professor at Nagoya University, Japan. His research interests include robust control and switching control. He is a member of ISCIE and IEEE.