High-Dimensional Bayesian Optimization with Constraints: Application to Powder Weighing

Shoki Miyagawa, Atsuyoshi Yano, Naoko Sawada, Isamu Ogawa

Information Technology R&D Center, MITSUBISHI Electric Corporation, Kamakura 247-8501, Japan,
Miyagawa.Shoki@ds.MitsubishiElectric.co.jp
Yano.Atsuyoshi@dn.MitsubishiElectric.co.jp
Sawada.Naoko@df.MitsubishiElectric.co.jp
Ogawa.Isamu@ah.MitsubishiElectric.co.jp

Abstract. Bayesian optimization works effectively optimizing parameters in black-box problems. However, this method did not work for high-dimensional parameters in limited trials. Parameters can be efficiently explored by nonlinearly embedding them into a low-dimensional space; however, the constraints cannot be considered. We proposed combining parameter decomposition by introducing disentangled representation learning into nonlinear embedding to consider both known equality and unknown inequality constraints in high-dimensional Bayesian optimization. We applied the proposed method to a powder weighing task as a usage scenario. Based on the experimental results, the proposed method considers the constraints and contributes to reducing the number of trials by approximately 66% compared to manual parameter tuning.

Keywords: Bayesian optimization, constraints, powder weighing

1 Introduction

Bayesian optimization has attracted attention because it can estimate the optimum parameters in models where the input-output relationship is a black box. This enables the exploration of the optimum parameters with fewer trials than with random or grid searches. It can be applied not only to hyperparameter optimization in deep learning but to cases such as molecular design [6][26], robot manipulation [16][24], and melody generation [27]. Many libraries [1][3][19] can automate experiments using Bayesian optimization, whose practical applications can be seen in various domains.

However, Bayesian optimization cannot estimate optimum parameters in a high-dimensional space (typically > 10) within a reasonable number of trials because of the curse of dimensionality, where the search space is explosively vast and sparse. Many studies [6][15][10][20][21][22][25][20] have focused on exploring low-dimensional space acquired by reducing dimensionality instead of directly dealing with high-dimensional space. We refer to a low-dimensional space as a latent space. Particularly, nonlinear dimensionality reduction [6][22][20] realizes
more efficient parameter exploration than linear dimensionality reduction \cite{10} or subspace exploration in the original high-dimensional space \cite{15}. They can also convert discrete into continuous variables to conduct gradient ascent in the latent space. However, they cannot consider constraints for parameters because they cannot be explicitly expressed in the latent space owing to nonlinear embedding. Although post-processing can transform the generated parameters into those that are valid for the system, dramatic transformations of the parameters decrease the exploration efficiency. Therefore, considering these constraints is a crucial issue in high-dimensional Bayesian optimization (HDBO).

To address this challenge, we propose a method that considers two types of constraints: known equality and unknown inequality constraints. We incorporate disentangled representation learning (DRL) to consider inequality constraints even with nonlinear dimensionality reduction. Although DRL is generally used to improve the interpretability of a generative model and enable users to control the model, we demonstrated that it can also be used to learn features from datasets that implicitly satisfy inequality constraints. The proposed method does not require users to provide inequality constraints explicitly. That is, it can be applied to unstructured data, such as images and natural languages, where the explicit formulation of constraints is difficult. Equality constraints can be considered by dividing the parameters into variable and fixed ones, as detailed in Section 3.

The proposed method can be widely applied to HDBO under known equality and unknown inequality constraints. Although some methods consider either high-dimensionality or constraints, few studies \cite{4} have focused on considering both. Typical benchmarks include a complex artificial function \cite{7} and robot manipulation \cite{16,24}; however, these constraints are not considered. Bergmann and Graichen \cite{4} proposed a method to consider unknown inequality constraints; however, they did not apply their method to high-dimensional parameters but to 2D artificial test functions in their evaluation. This study applied the proposed method to the practical task of precisely scaling a specific amount of powder with a powder weighing system as a usage scenario to optimize parameters with high dimensionality and constraints. Previously, the powder weighing system parameters had to be manually adjusted for each type of powder, which requires expertise and time to tune. To the best of our knowledge, this study is the first attempt to apply a machine learning technique to a powder weighing system. In this paper, Section 4 provides an overview of the powder weighing system and the dataset; Section 5 presents the preprocessing, model architecture, and hyperparameters of the system.

Two experiments were conducted using the powder weighing system. Based on the first experiment, the inequality constraints can be considered by adjusting the dimensionality of the latent space and regularization coefficient. As demonstrated by the other experiment, the proposed method achieved the target performance and reduced the number of trials by 66% compared to manual parameter tuning. The details of each experiment are presented in Section 6.

In summary, the main contributions of this study are twofold:
We designed a versatile method for HDBO under known equality and unknown inequality constraints.

By introducing DRL, Bayesian optimization can implicitly consider inequality constraints without explicit user constraints; when applied to the powder-weighing system, the proposed method allows users to determine parameters that achieve the target performance within one-third of the trials compared to manual tuning.

2 Related Work

The main approaches to dealing with HDBO are twofold. The first approach performs Bayesian optimization in a low-dimensional space embedded in high-dimensional parameters. Wang et al. [25] proposed the use of linear random matrices to embed parameters into a low-dimensional space. Kirschner et al. [15] performed Bayesian optimization to explore parameters on a one-dimensional subspace; the dimensions were selected from high dimensions. Li et al. [17] applied a deep learning technique, called dropout, which randomly dropped out some dimensions and performed Bayesian optimization on a subspace consisting of non-dropped dimensions. However, applying these methods to unstructured data such as images and natural languages is challenging. For example, in the case of images, these methods do not model the correlations between neighboring pixels because the correlations are strong. Conversely, some methods [6][9][20][21][26] use a neural network, which learns embedding into low-dimensional space and models correlations between input variables. However, a neural network cannot explicitly express constraints in the transformed space. The proposed method implicitly considers inequality constraints by introducing DRL to address the problem.

The other approach proposed by Kandasamy et al. [11] is based on the additive Gaussian process, where high-dimensional parameters are decomposed into low-dimensional parameter groups. Li et al. [18] and Rolland et al. [23] extended the approach and loosened the assumptions regarding the decomposition. Inspired by this approach, we decompose the parameters into two groups to consider the equality constraints, as described in Section 3, although previous methods do not consider constraints. In summary, the proposed method is positioned between these two conventional approaches because it combines nonlinear embedding and parameter decomposition.

3 Method

This section describes the manner in which the proposed method explores the optimum parameters. Figure 1 presents an overview of the proposed method. First, to consider equality constraints in Bayesian optimization, users need to divide the parameters into variable parameters $x_v$ without equality constraints and fixed parameters $x_f$ with equality constraints and objective values $y$. The parameters to be optimized are usually considered variable parameters, whereas
parameters for external factors, such as physical properties and temperature, are considered fixed parameters. Although more than one objective value can exist, we consider a single variable in this study for simplicity.

The proposed method optionally performs preprocessing on each parameter in the datasets, such as normalization and outlier removal, before each step. Step 1 trains encoders and decoders for dimensionality reduction. Step 2 takes user-specified fixed parameters $x_f^{\text{target}}$ and generates variable parameters $x_v^{\text{target}}$ using encoders and decoders. Finally, users try the variable parameters $x_v^{\text{target}}$ in the next trial and then obtain new objective values $y^{\text{target}}$. The input-output pair $((x_v^{\text{target}}, x_f^{\text{target}}), y^{\text{target}})$ was added to the datasets to generate variable parameters in the subsequent trials. Users iterate this process until reasonable parameters are obtained. The following sections detail each step shown in Figure 1. A specific example of the preprocessing, model architecture, and hyperparameters of the powder weighing system is described in Section 5.

3.1 Step 1: Dimensionality Reduction

The encoders and decoders for the variables and fixed parameters were trained using the datasets. To efficiently extract more complex features and reduce dimensionality, we used a variational autoencoder (VAE) for dimensionality reduction of the variable parameters. The use of nonlinear models for fixed parameters is needless because they are not optimized. In the experiment described in Section 6, we used principal component analysis (PCA) to reduce the dimensionality of the fixed parameters.

Introducing DRL enables us to consider the inequality constraints for variable parameters. The DRL decomposes high-dimensional entangled features into meaningful low-dimensional features. Therefore, users only need to check whether the inequality constraints are satisfied for the data on each axis, instead of every area of the latent space. If the datasets contain only data that satisfy the in-
equality constraints, the data on each axis are likely to meet them. Conversely, if the datasets include data that do not satisfy the inequality constraints, the data on a particular axis might not satisfy them. Therefore, we can limit the search area on the undesirable axis in Bayesian optimization. For simplicity, we remove trials whose variable parameters do not satisfy inequality constraints from the datasets, as detailed in Section 5.1.

The proposed method uses $\beta$-VAE [8] to introduce the DRL to the VAE. Equation (1) is the loss function $L(x, z)$ of $\beta$-VAE. The first term is the expected value of the log-likelihood $\log p(x_v | z_v)$ and represents a reconstruction error, whereas the second term is the Kullback-Leibler divergence $D_{\text{KL}}$ between the approximate posterior distribution $q(z_v | x_v)$ and the prior distribution $p(z_v)$, such as a standard normal distribution. The larger the coefficient $\beta$ in Equation (1), the more disentangled the representation learning is accelerated; thus, inequality constraints can be considered. Notably, a larger $\beta$ results in a more significant reconstruction error owing to the regularization effect. Large reconstruction errors generate parameters with rough-grained features, which implies that optimum parameters might not be generated even after many trials.

$$L(x_v, z_v) = \mathbb{E}_{q(z_v | x_v)}[\log p(x_v | z_v)] - \beta D_{\text{KL}}[q(z_v | x_v) || p(z_v)]$$

3.2 Step 2: Bayesian Optimization

Step 2 generates variable parameters $x_{v\text{target}}$ for the next trial, according to the user-specified fixed parameters $x_{f\text{target}}$. First, the variable and fixed parameters, $x_{v\text{dataset}}$ and $x_{f\text{dataset}}$, are encoded to the latent parameters, $z_{v\text{dataset}}$ and $z_{f\text{dataset}}$, respectively. Subsequently, the proposed method trains a Bayesian model, such as the Gaussian process regression (GPR) model, which predicts the objective values $y_{\text{dataset}}$ according to the latent parameters ($z_{v\text{dataset}}$, $z_{f\text{dataset}}$). The GPR model provides the predicted means $\mu(z_v, z_f)$ and variances $\sigma(z_v, z_f)$ of the objective values, which are used to compute the acquisition function $a(z_v, z_f)$. Bayesian optimization generates a latent variable parameter $z_{v\text{target}}$ that maximizes the acquisition function under the equality constraint $z_f = z_{f\text{target}}$ in Equation (2). The parameters $z_{v\text{target}}$ are decoded into variable parameters $x_{v\text{target}}$.

$$z_{v\text{target}} = \arg \max a(\mu(z_v, z_f = z_{f\text{target}}), \sigma(z_v, z_f = z_{f\text{target}}))$$

4 Usage Scenario: Powder Weighing System

4.1 System Overview

As illustrated in Figure 2, a powder weighing system weighs a user-specified amount of powder from the upper container into the lower container. The scale beneath the lower container weighed the amount of powder inside the lower container. The system can control the amount of powder entering the lower container by vertically moving the valve of the upper container. The degree to
which the valve opens is referred to as the valve opening degree. This system adjusts the valve opening degree according to a preset parameter. The preset parameter contains ten valve opening degrees \( \{v_i\} \) and nine switching weights \( \{s_i\} \). They configure nine pairs of them \( \{(v_i, s_i)\} \) and an initial valve opening degree \( v_0 \). The system starts by opening the valve according to the initial degree \( v_0 \). When the weighing value reaches the switching weight \( s_i \), it adjusts the valve to the corresponding degree \( v_i \). Their proper values depend on the type of powder and the user-specified required weight. It requires an average of 20 trials to obtain proper values, even for knowledgeable users. A single trial took a maximum of five min.

The valve opening degrees and switching weights must satisfy two types of inequality constraints: non-negative and monotonically decreasing constraints. The valve opening degrees generally have a larger value at the beginning of weighing and a smaller value at the end. This is because the small valve opening degrees at the beginning take a longer time to weigh the powder, and the large valve opening degrees at the end weigh more powder than expected.

4.2 Dataset

This section reviews the various parameters included in the datasets for the powder weighing system and categorizes them into variable and fixed parameters, and objective values, as shown in Table 1. The valve opening degrees and switching weights are variable parameters; that is, variable parameters have 19 dimensions. The fixed parameters include 11 physical properties such as the jetting index that explains the ease of dispersibility of the powder. The physical properties include:

- Average particle size [\( \mu m \)]
- Apparent specific gravity (loose) [-]
- Apparent specific gravity (firm) [-]
- Compressibility [%]
- Angle of repose [\( ^\circ \)]
- Spatula angle (average) [\( ^\circ \)]
- Flowability index [-]
- Collapse angle [\( ^\circ \)]
- Difference angle [\( ^\circ \)]
- Dispersion [%]
- Jetting index [-]

![Fig. 2. Powder weighing system (left) and variable parameters (right).](image-url)
parameters are related to the fall speed of the powder. For example, the larger the jetting index, the faster the fall speed of the powder is like a liquid because it easily contains air. That is, a larger jetting index complicates the control of the fall speed of the powder. The fixed parameters also include user-specified setting parameters for the powder weighing system, such as the required weight of the powder and whether to vibrate. Three parameters are also related to the fall speed of the powder: shaking, vibration, and pre-vibration. However, they are usually predetermined by knowledgeable users and rarely change; therefore, we set them as fixed parameters. The absolute value of the difference between the measured and required weights, referred to as the weighing error, was defined as the objective value.

The datasets contained 60 types of powder and consisted of 1,792 trials. The average number of trials for each powder was 31.33 ± 19.48. The variable parameters that were finally adopted were also included in the dataset. The criterion for parameter adoption was whether the ratio of the weighing error to the required weight was less than 1%.

5 Implementation

5.1 Preprocessing

This section describes the preprocessing of the powder weighing system. The following preprocessing steps are common to Steps 1 and 2, detailed in Section 3, except for duplication removal and data filtering.

Normalization Data normalization is an essential preprocessing step in machine learning. Based on a pilot study, we normalized each parameter as follows.
For the fixed parameters, we performed a min-max normalization for each dimension. For the variable parameters, we performed the same normalization for the first step of the valve opening degree \( v_0 \) and switching weight \( s_1 \) (Figure 2). However, we used the relative value of the first step in subsequent steps. This is because the scale of the parameter values depends on the value of the first step. Objective values were standardized using the mean and standard deviation.

**Outlier removal** We defined outliers as those that satisfied the following criteria: trials with large weighing errors or those whose valve opening degrees or switching weights did not decrease monotonically. Sixty-one trials were considered outliers.

**Duplication removal** Before Step 1, we removed trials whose fixed or variable parameters had the same values. Because the same powder had similar fixed or variable parameters, the powder with fewer trials had more significant reconstruction errors than those with more trials. After removing duplicates, we had 158 and 199 trials for the fixed and variable parameters, respectively.

**Data filtering** We need to define a bounding box as a search area in the latent space to perform Bayesian optimization in Step 2. To optimize the variable parameters in a few trials, we assumed that powders with similar fixed parameters have similar optimal values for variable parameters. Based on this assumption, we filtered the types of powder in the datasets whose fixed parameters were similar to those of the target powder \( x_f^{target} \) and defined a bounding box that contained their latent variable parameters \( z_v^{dataset} \). This filtering improves the efficiency of parameter exploration because the search space is localized and the new trial has a more significant impact on GPR training. We used the top seven powders whose fixed parameters were close to the target values based on Euclidean distance. After filtering, data for approximately 200 trials were retained.

**Train/Test split** We divided trials in the datasets into training and test trials to avoid overfitting. For Steps 1 and 2, 70 and 80% of the trials in the datasets were used for training, respectively.

### 5.2 Model Architecture and Hyperparameters

This section describes the architecture and hyperparameters of the powder weighing system in Steps 1 and 2.

**\( \beta \)-VAE** For variable parameters, we designed the architecture of both the encoder and decoder in the \( \beta \)-VAE with three fully connected layers with 19 units and two ReLU activation layers. We set the prior distribution in the latent space to be a standard normal distribution and the number of dimensions of the latent space to be two (detailed in Section 6.1). We used the Adam [13] optimizer and set an initial learning rate to be 0.001. Using these settings, we trained 1,000 epochs with a batch size of 32 and saved the model for the epoch when the loss was the lowest. Model training completes in 1.5 min using a single GPU (NVIDIA GeForce RTX2070).

**PCA** We used PCA as a dimensionality reduction model for the fixed parameters. Notably, the physical properties (Table SI) and the remaining parameters were encoded separately. This is because the cumulative contribution ratio was
approximately 60% when compressing together into three dimensions, whereas it was approximately 98% when compressing individually into one and two dimensions. Therefore, the correlations between the physical properties and the others are weak.

GPR A kernel function of GPR is the weighted sum of the Matern 5/2 kernel and automatic relevance determination, and the weight is estimated based on the maximum likelihood.

Acquisition function We used the upper confidence bound (UCB) [2] as the acquisition function to interactively allow users to change an optimization strategy. Users can choose one of three types of variable parameters: exploration-oriented, exploitation-oriented, or intermediate. The corresponding UCB hyperparameters were set to 0.001, 1.0, and 0.5, respectively, based on a melody design application [27].

6 Evaluation

6.1 Experiment 1: Effectiveness of Step 1

Procedure The aim of Experiment 1 is to verify whether the introduction of DRL enables us to consider inequality constraints. We randomly sampled many data samples from the latent space and counted the number of samples that did not satisfy the inequality constraints. We evaluated the correlations between the dimensionality of the latent space \( d_v \), the value of the coefficient \( \beta \), and the number of samples that did not satisfy the inequality constraints. The inequality constraints include non-negative and monotonically decreasing constraints, which users do not explicitly specify to train the dimensionality reduction models. We also visualized the variable parameters of the samples and confirmed the features acquired by DRL for a qualitative evaluation. Based on the results, we discuss the proper dimensionality of \( \beta \)-VAE and the appropriate coefficient \( \beta \) for Experiment 2.

Result Figure 3 shows the relationship between coefficient \( \beta \) and the number of samples that do not satisfy the inequality constraints ((a) non-negative constraint, (b) monotonically decreasing constraint, (c) (a) or (b)).
Fig. 4. Comparison of variable parameters for different $\beta$. The horizontal axis represents the switching weights, whereas the vertical axis represents the valve opening degrees.

(A) $\beta = 0.1$ (B) $\beta = 0.5$ (C) $\beta = 1.0$

**Fig. 4.** Comparison of variable parameters for different $\beta$. The horizontal axis represents the switching weights, whereas the vertical axis represents the valve opening degrees.

### Constraint, (b) monotonically decreasing constraint, and (c) (a) or (b)). We sampled 1,000 data samples within the hypersphere with a radius of two around the origin of the latent space. The experimental results are as follows:

- A larger coefficient $\beta$ decreases the ratio of data samples that do not satisfy the inequality constraints, implying that the introduction of DRL enables us to consider the inequality constraints.
- Larger dimensionality in the latent space increases the ratio of data samples that do not satisfy the inequality constraints. Based on our observations, samples far from the origin of the latent space do not satisfy the inequality constraints because the large dimensionality learns undesirable fine-grained features and is emphasized in the area far from the origin.
- Because (b) and (c) show almost the same trend, data samples that do not satisfy the non-negative constraint do not meet the monotonically decreasing constraint.

Next, to determine the proper coefficient $\beta$ of $\beta$-VAE, we compared the valve opening degrees and switching weights reconstructed from the two-dimensional latent space in terms of $\beta = 0.1, 0.5, 1.0$. Figure 4 shows 15 types of variable parameters sampled in the range of $[-2, 2]$ along each axis of the two-dimensional latent space. They imply that a larger $\beta$ eases the satisfaction of the inequality constraints, but produces similar variable parameters, called posterior collapse. Posterior collapse increases reconstruction errors and we may be unable to determine the proper parameters even after many trials. Therefore, posterior collapse is more severe than when inequality constraints are not satisfied. Consequently, we decided to use $\beta = 0.1$ in Experiment 2.
Table 2. User-specified fixed parameters for each powder used in Experiment 2.

| Valve diameter | Required weight | Input weight | Shaking | Vibration | Pre-vibration |
|----------------|-----------------|--------------|---------|-----------|---------------|
| A              | 150 mm          | 10 kg        | 150 kg  | False     | True          | False         |
| B              | 150 mm          | 18 kg        | 500 kg  | True      | False         | False         |
| C              | 150 mm          | 10 kg        | 200 kg  | False     | True          | False         |

Fig. 5. Mapping fixed parameters on a 2D scatter-plot.

Fig. 6. Weighing errors for each trial.

6.2 Experiment 2: Effectiveness of Step 2

Procedure Experiment 2 aims to verify whether the proposed method could determine optimum parameters within a reasonable number of trials. The number of trials required to achieve the target weighing error was less than 1% of the required weight of the powder. In this experiment, we used three types of powders not included in the datasets: Powders A, B, and C. Their features are as follows:

- The powders are highly floodable.
- Powders A and B are more fluid than powder C.
- The grain diameter is larger in the order of powders A, B, and C.

Table 2 lists the user-specified fixed parameters of each powder. Figure 5 maps 60 types of powder in the datasets on a 2D scatterplot according to their fixed parameters and indicates that powders A, B, and C can appropriately validate the versatility of the proposed method because none of them are outliers.

The parameters were explored based on the users’ experience and expertise to evaluate the system in an actual operation. The following steps were performed by users for the exploration:

- Users choose one of the three candidates as a variable parameter: exploitation-oriented, exploration-oriented, and intermediate, as detailed in Section 5.2.
- When parameters that do not satisfy the inequality constraints are generated, a minimum modification is made to the parameters to satisfy the inequality constraints before evaluation. When parameters require extensive modification or are invalid, users penalize them with a 10% weighing error without trying them.
Result. Figure 6 shows the ratio of the weighing errors to the required weight of the powder through seven trials for the three powders. We achieved the target weighing error (less than 1% of the required weight) within seven trials, which is approximately one-third of the number of trials required for manual tuning. Based on the results, the proposed method worked effectively and $\beta$-VAE and GPR models have a high generalization performance for unknown powders.

We successfully generated variable parameters that satisfy the two inequality constraints in all trials for powders B and C. In contrast, for Powder A, the proposed method generated variable parameters that did not satisfy the non-negative constraint in one trial and were penalized without evaluation in two trials. The proposed method seems to have explored areas far from the origin of the latent space, as shown in Figure 5.

7 Discussion

7.1 Hyperparameter Tuning

As illustrated in Experiment 1, two hyperparameters of $\beta$-VAE should be adjusted: the number of dimensions of the latent space $d_v$ and coefficient $\beta$. A larger $d_v$ leads to a small reconstruction error, whereas it generates variable parameters that are unlikely to satisfy the inequality constraints. However, when coefficient $\beta$ is large, a posterior collapse that increases the reconstruction error occurs and variable parameters are more likely to satisfy the inequality constraints. These contrasting effects indicate that the appropriate value of coefficient $\beta$ depends on the dimensions $d_v$. Therefore, the appropriate values of them can be determined using Figures 3 and 4 because they visualize a trade-off between the effects. The results of Experiment 2 validate that appropriate hyperparameter tuning contributes to achieving the target performance in limited trials.

However, the robustness of the setting of the two hyperparameters remains unclear because Experiment 2 was only tested for a specific combination. Other combinations of hyperparameters will be used in the experiment and methods to improve the robustness will be investigated. Particularly, we will consider applying other DRL methods, such as FactorVAE [12] and $\beta$–TCVAE [5], which improve the trade-off between the reconstruction error and the Kullback-Liebler divergence.

7.2 Search Area in Latent Space

Bayesian optimization requires defining a search area in a latent space. However, it is not trivial to determine the search area. As described in Section 5, we assumed that different types of powders with similar fixed parameters have similar optimal values for variable parameters and only the powders with similar fixed parameters are filtered. Therefore, the search area is localized compared with the case without filtering. This assumption is a slightly strong restriction because the optimal parameters may not be in the search area. The search range should
be large according to the user-acceptable number of trials. However, suppose the search area is too large. In this case, the proposed method may generate variable parameters that do not satisfy the inequality constraints because of the emphasis on fine-grained features, as shown in Figure 3(a). To address this challenge, we investigated methods to automatically determine the search area. We believe that a method \cite{22} that narrows the search area based on decoder uncertainty should be used.

7.3 Generalizability

From this empirical study, we will extend the generalizability of the proposed method from the perspective of constraints. The proposed method cannot consider unknown equality constraints, known inequality constraints, or more complicated constraints, such as the case in which inequality constraints depend on fixed parameters. We will also apply the proposed method to other tasks, such as molecule design \cite{6}\cite{26}, which handles discrete parameters.

8 Conclusion

This study designed a general framework that combines dimensionality reduction and Bayesian optimization for high-dimensional parameters with constraints. As observed, introducing DRL into dimensionality reduction enables us to consider inequality constraints, even if users do not explicitly provide them. We applied the proposed method to a powder weighing system and proved that the method could implicitly consider inequality constraints, including non-negative and monotonically decreasing constraints. Moreover, the proposed method successfully optimized parameters within a reasonable number of trials and reduced the required number of trials by 66\% compared with manual parameter tuning. Experiments also validated the importance of two hyperparameters: dimensionality in the latent space and coefficient $\beta$ of $\beta$-VAE. The robustness of the proposed method for these hyperparameters and a method to automatically determine the search area for Bayesian optimization will be examined in future studies.

Acknowledgement. The authors are grateful to Hirofumi Nitta, Masato Nakagome and Kenji Takizawa from Tsukishima Machine Sales Co Ltd and Tomoki Yoshimura from Tsukishima Kikai Co Ltd for providing the dataset and for their kind cooperation in the experiments.

References

1. Akiba, T., Sano, S., Yanase, T., Ohta, T., Koyama, M.: Optuna: A next-generation hyperparameter optimization framework. In: KDD ’19, p. 2623–2631
2. Auer, P.: Using confidence bounds for exploitation-exploration trade-offs 3, 397–422 (2003)
3. Balandat, M., Karrer, B., Jiang, D.R., Daulton, S., Letham, B., Wilson, A.G., Bakshy, E.: BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization. In: NeurIPS ’20
4. Bergmann, D., Graichen, K.: Safe bayesian optimization under unknown constraints. In: CDC '20, pp. 3592–3597
5. Chen, R.T.Q., Li, X., Grosse, R., Duvenaud, D.: Isolating sources of disentanglement in variational autoencoders. In: NeurIPS '18
6. Gómez-Bombarelli, R., Wei, J.N., Duvenaud, D., Hernández-Lobato, J.M., Sánchez-Lengeling, B., Sheberla, D., Aguilera-Iparraguirre, J., Hirzel, T.D., Adams, R.P., Aspuru-Guzik, A.: Automatic chemical design using a data-driven continuous representation of molecules. ACS Central Science 4(2), 268–276 (2018)
7. Hansen, N., Auger, A., Ros, R., Mersmann, O., Tušar, T., Brockhoff, D.: COCO: a platform for comparing continuous optimizers in a black-box setting. Optimization Methods and Software 36(1), 114–144 (2020)
8. Higgins, I., Matthey, L., Pal, A., Burgess, C.P., Glorot, X., Botvinick, M.M., Mohamed, S., Lerchner, A.: Beta-vae: Learning basic visual concepts with a constrained variational framework. In: ICLR '17
9. Jaquier, N., Rozo, L.: High-dimensional bayesian optimization via nested riemannian manifolds. In: NeurIPS '20
10. Jolliffe, I.: Principal Component Analysis. Springer Verlag (1986)
11. Kandasamy, K., Schneider, J., Poczos, B.: High dimensional bayesian optimisation and bandits via additive models. In: ICML '15, vol. 37, pp. 295–304
12. Kim, H., Mnih, A.: Disentangling by factorising. In: ICML '18, vol. 80, pp. 2649–2658
13. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR '15
14. Kingma, D.P., Welling, M.: Auto-Encoding Variational Bayes. In: ICLR '14
15. Kirschner, J., Mutny, M., Hiller, N., Ischebeck, R., Krause, A.: Adaptive and safe Bayesian optimization in high dimensions via one-dimensional subspaces. In: ICML '19, vol. 97, pp. 3429–3438
16. Letham, B., Calandra, R., Rai, A., Bakshy, E.: Re-examining linear embeddings for high-dimensional Bayesian optimization. In: NeurIPS '20
17. Li, C., Gupta, S., Rana, S., Nguyen, V., Venkatesh, S., Shilton, A.: High dimensional bayesian optimization using dropout. In: IJCAI '17, pp. 2096–2102
18. Li, C.L., Kandasamy, K., Poczos, B., Schneider, J.: High dimensional bayesian optimization via restricted projection pursuit models. In: AISTAT '16
19. Liaw, R., Liang, E., Nishihara, R., Moritz, P., Gonzalez, J.E., Stoica, I.: TUNE: A research platform for distributed model selection and training (2018)
20. Lu, X., Gonzalez, J., Dai, Z., Lawrence, N.: Structured variationally auto-encoded optimization. In: ICML '18, vol. 80, pp. 3267–3275
21. Moriconi, R., Deisenroth, M., K S, S.K.: High-dimensional bayesian optimization using low-dimensional feature spaces. Machine Learning 109, 1925–1943 (2020)
22. Notin, P., Hernández-Lobato, J.M., Gal, Y.: Improving black-box optimization in vae latent space using decoder uncertainty. In: NeurIPS '21, vol. 34, pp. 802–814
23. Rolland, P., Scarlett, J., Bogunovic, I., Cevher, V.: High-dimensional bayesian optimization via additive models with overlapping groups. In: AISTATS '18
24. Wang, Z., Li, C., Jegelka, S., Kohli, P.: Batched high-dimensional bayesian optimization via structural kernel learning. In: ICML '17, p. 3656–3664
25. Wang, Z., Zoghi, M., Hutter, F., Matheson, D., De Freitas, N.: Bayesian optimization in high dimensions via random embeddings. In: IJCAI '13, p. 1778–1784
26. Yan, C., Wang, S., Yang, J., Xu, T., Huang, J.: Re-balancing variational autoencoder loss for molecule sequence generation. In: BCB '20
27. Zhou, Y., Koyama, Y., Goto, M., Igarashi, T.: Interactive exploration-exploitation balancing for generative melody composition. In: IUI '21, p. 43–47