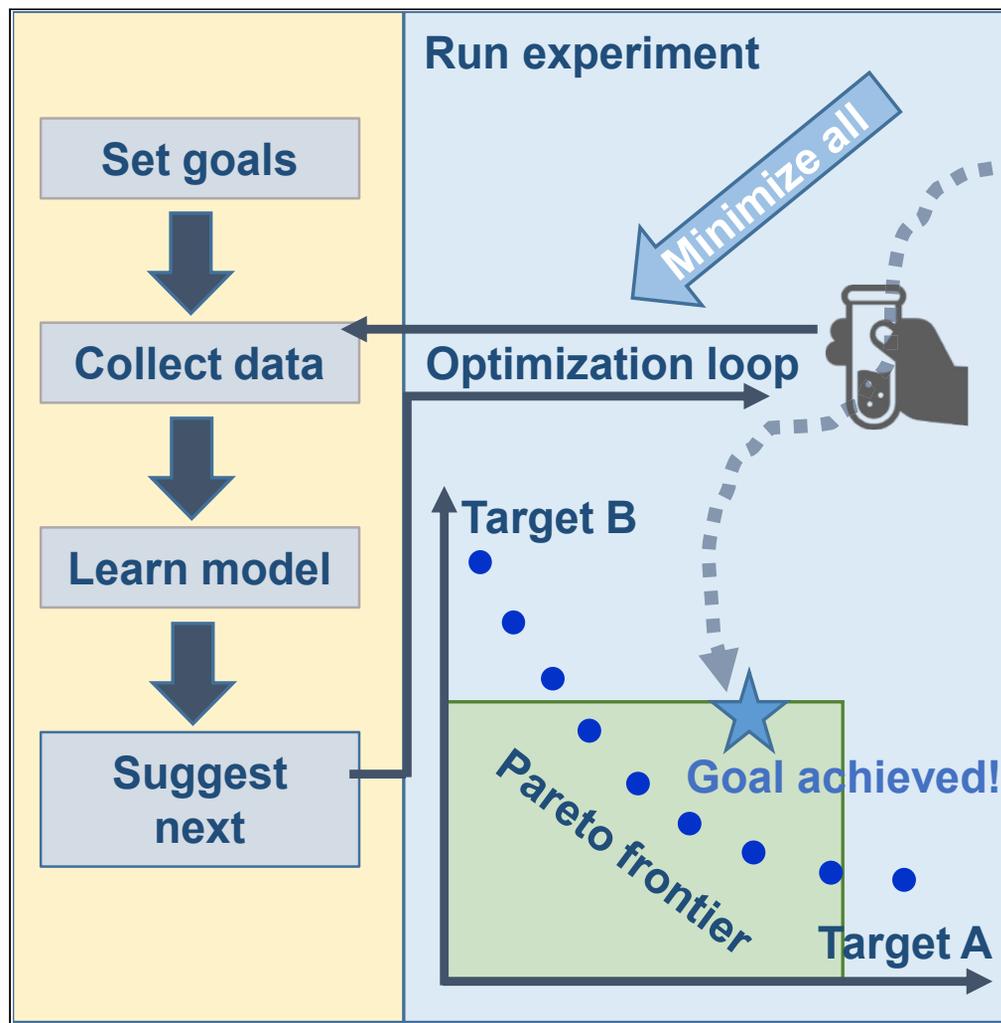


Article

Bayesian optimization for goal-oriented multi-objective inverse material design



Kyohei Hanaoka

hanaoka.kyohei.xicmq@
showadenko.com**Highlights**

Multi-objective (MO) problems with predefined goals for all objectives were studied

Fully probabilistic Bayesian optimization (BO) for goal achievement was proposed

The proposed method clearly outperformed a baseline in goal achievement efficiency

Goal-oriented BO simplifies MO problems and works with small number of experiments

Hanaoka, iScience 24, 102781
July 23, 2021 © 2021 The Author(s).
<https://doi.org/10.1016/j.isci.2021.102781>

Article

Bayesian optimization for goal-oriented multi-objective inverse material design

Kyohei Hanaoka^{1,2,*}

SUMMARY

Bayesian optimization (BO) can accelerate material design requiring time-consuming experiments. However, although most material designs require tuning of multiple properties, the efficiency of multi-objective (MO) BO in time-consuming experimental material design remains unclear, due to the complexity of handling multiple objectives. This study introduces MO BO method that efficiently achieves predefined goals and shows that by focusing on achieving the goals, BO can efficiently accelerate realistic MO design problems with small efforts. Benchmarks showed that the proposed BO method dramatically reduced the number of experiments needed to achieve goals relative to a baseline method. Virtual MO inverse design experiments with realistic material design problems were also performed, during which the proposed method could achieve goals within only around ten experiments in average and showed over 1000-fold acceleration relative to the random sampling for the most difficult case. The introduction of goal-oriented BO will precede real-world application of BO.

INTRODUCTION

Bayesian optimization (BO) (Greenhill et al., 2020; Shahriari et al., 2015) is one of the major approaches to inverse material design and involves gradually optimizing material-design parameters through repeated experiments (Balachandran et al., 2016; Doan et al., 2020; Lookman et al., 2017, 2019). Although the process resembles human-based trial-and-error, design parameters in BO are determined based on a machine-learning model, which is updated after each experiment and gets smarter through repeated updates (Shahriari et al., 2015). Accordingly, BO can effectively accelerate difficult optimization problems and is useful, particularly for material-design problems involving time-consuming experiments. Currently, there are many computational material design studies that utilize BO (Bassman et al., 2018; Fukazawa et al., 2019; Hashimoto et al., 2020; Herbol et al., 2018; Okamoto, 2017; Sakurai et al., 2019; Seko et al., 2015), and several recent real-world experimental studies have actually realized inverse material design using BO (Balachandran et al., 2018; Homma et al., 2020; Langner et al., 2020; Rouet-Leduc et al., 2016; Wakabayashi et al., 2019; Xue et al., 2016; Yuan et al., 2018). However, the targets for most such real-world inverse design studies were single properties, despite most material-design problems requiring the optimization of multiple target properties. This type of optimization problem is known as multi-objective (MO) optimization. Nowadays, the most popular class of method for MO optimization is genetic algorithm-based (Deb et al., 2002; Jung et al., 2017; Lee et al., 2017; Menou et al., 2018; Niu et al., 2018; Shrivastava et al., 2018), while the real-world implementation of more efficient BO based MO inverse material design remains in its nascent stages.

The challenge with MO optimization relative to single-objective optimization comes from the number of possible solutions arising. In single-objective optimization a single optimal solution is usually obtained after running the optimization, while MO optimization elicits many optimal solutions, as in the following. Consider MO optimization, where two objective properties, A and B were minimized by tuning design parameters, X. Figure 1 shows a schematic mapping of properties A and B corresponding to the optimal design parameters obtained by this optimization task (blue circle). Optimal design parameters achieving minimum values for both properties A and B may be preferable. However, usually the best design parameters for properties A and B (X1 and X2 in Figure 1, respectively) are not the same. For that reason, in MO optimization, optimal solutions are defined as design parameters where it is impossible to improve any properties without negatively affecting others. Such design parameters are called Pareto optimal solutions. For example, the design parameter X3, which corresponds to the red circle in Figure 1, is not Pareto

¹Advanced Technology Research & Development Center, Showa Denko Materials Co., Ltd., 48 Wadai, Tsukuba City, Ibaraki Prefecture 300-4247, Japan

²Lead contact

*Correspondence: hanaoka.kyohei.xicmq@showadenko.com

<https://doi.org/10.1016/j.isci.2021.102781>



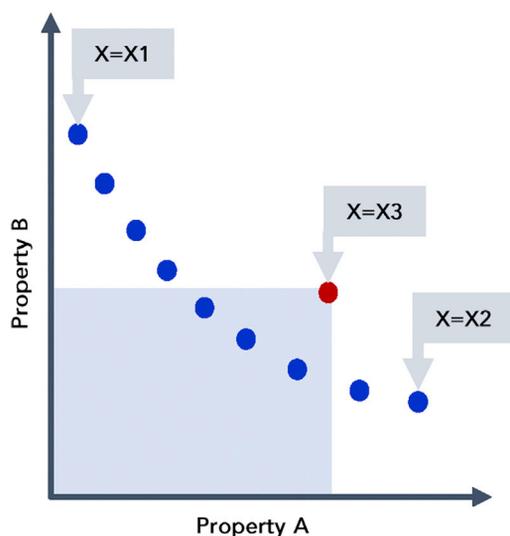


Figure 1. Schematic mapping of Pareto optimal solutions

Blue circles represent Pareto optimal solutions. The red circle represents a non-Pareto optimal design. Blue circles in the blue box have better values in both properties A and B than that with the design parameter X3.

optimal, because there are blue circles with lower values of both properties A and B (blue box in Figure 1). Conversely, parameter X1 and X2, which corresponds to blue circles, are Pareto optimal solutions, because reducing properties A and B is impossible without causing the other to increase. According to this definition, design parameters corresponding to all blue circles, which balance dual objectives in differing ways, also constitute Pareto optimal solutions. Because no general criteria exist to compare the quality of the Pareto optimal solutions, various balances of Pareto optimal solutions should be sought, but many experiments have to be performed to find them.

Most popular MO inverse design approaches comprise finding the whole Pareto optimal solution, from which the best-balanced solution is then chosen (Figure 2A). In this paper, this inverse design approach is called “many-solution-inverse-design”. Some recent computational material design studies used MO BO in many-solution-inverse-design to accelerate the finding of the Pareto optimal solutions (Janet et al., 2020; Karasuyama et al., 2020; Solomou et al., 2018; Talapatra et al., 2018; Wang et al., 2020b). However, as schematically shown in Figure 2A, finding many Pareto optimal solutions requires an excessive number of experiments, which is difficult to execute with time-consuming real-world experiments. Accordingly, this makes applying many-solution-inverse-design to real-world material design infeasible.

There are alternative MO inverse design approaches, whereby only a few Pareto optimal solutions are searched without the time-consuming need to search the entire space of Pareto optimal solutions (Figure 2B). In this paper, this inverse design approach is called “few-solution-inverse-design”. Most few-solution-inverse-design solutions use scalarization functions (SFs), which convert multiple objective properties to a single score, which, in turn, can be optimized using any single-objective optimization methods (Cummins and Bell, 2016; Wang et al., 2020a; Wheatle et al., 2020; Yamawaki et al., 2018). Among them, weight summation of multiple objective properties is most popular one. In the weight-summation SF, predefined weights for weight summation determine the balance of objective properties in the Pareto optimal solution. Using this weight-summation SF, several studies have successfully performed inverse material designs using machine-learning-based optimization (Cummins and Bell, 2016; Wang et al., 2020a; Wheatle et al., 2020). However, because the predefined weight is not directly reflected in the optimized solution, it is difficult to balance objective properties using weight-summation SF exactly as required. Conversely, there are studies using ad hoc SFs where the preference of the experimenter in terms of the balance of Pareto optimal solutions can be directly reflected (Häse et al., 2018; Walker et al., 2017). This class of methods for few-solution-inverse-design is promising, but comparing the performance of optimization methods for few-solution-inverse-design remains difficult because different optimization methods elicit different Pareto optimal solutions, and there is no quantitative measure available to fairly compare the quality of these different Pareto solutions. Therefore, an SF-method suitable for each application is unclear.

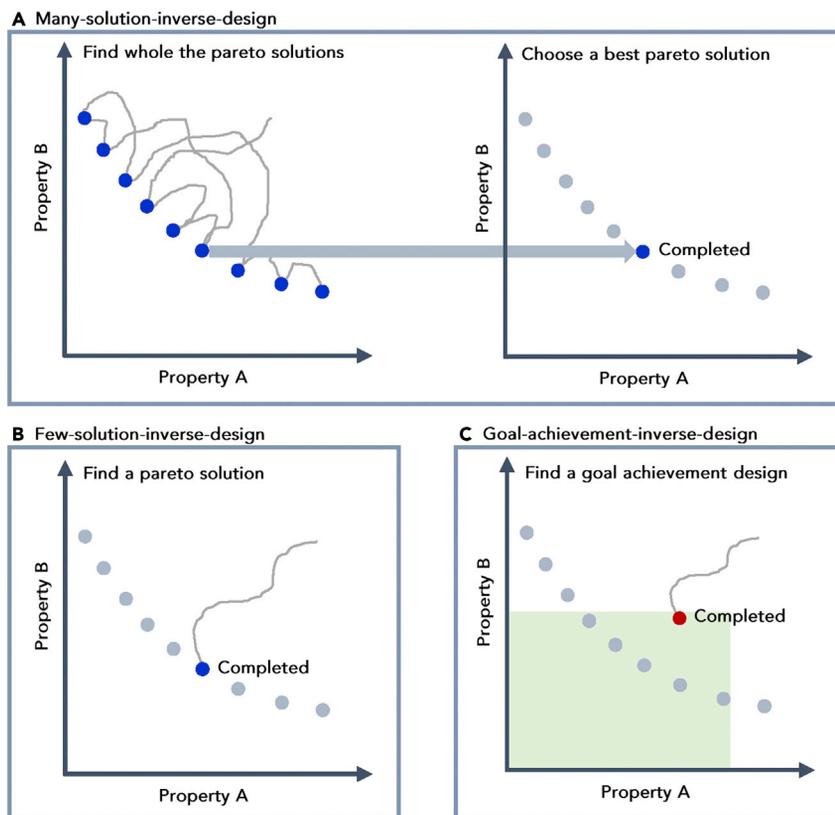


Figure 2. Schematics of multi-objective inverse design approaches

Blue and gray circles represent Pareto optimal solutions, while the red circle represents a non-Pareto optimal design. In the region colored pastel green, all the predefined goals are achieved.

Furthermore, it is often difficult to find even one Pareto optimal solution with time-consuming real-world experiments. Although finding Pareto optimal solutions represents a common goal of MO optimization studies (Gopakumar et al., 2018; Harada et al., 2020; Mannodi-Kanakithodi et al., 2016; Del Rosario et al., 2020), for real-world material designs with limited budget and available time, it is often difficult to find a true Pareto optimal solution and efforts to optimize the material property will be stopped after finding materials having properties acceptable for each application but not Pareto optimal solutions. However, although optimization methods studied in both many- and few-solution-inverse-design would also work with such a realistic design problem, the performance of BO in such realistic design problem has not been studied well and efficiency of MO BO in real-world material design requiring time-consuming experiments remains unclear.

In this study, we show that by focusing on finding a design parameter of materials achieving predefined goal values rather than the best design parameters, MO BO can efficiently accelerate realistic MO inverse design problem with small number of experiments. We considered a realistic process of MO inverse material design named goal-achievement-inverse-design. The first step of goal-achievement-inverse-design is to define quantitative goals for all target properties, then conduct goal-oriented MO BO. Finally, optimization is completed when the objective values have reached these goals rather than Pareto optimal solutions having been found (Figure 2C). Note that a similar design process is often seen in human-based real-world material design, especially when experimenters do not want to pay much experimental cost after finding a material that achieves the goals and already has properties acceptable for each application.

First of all the following section, a fully probabilistic MO BO method for goal-achievement-inverse-design is provided by extending a well-studied BO method, lower confidence bound (LCB) (Srinivas et al., 2010). Next, a rigid benchmark method capable of evaluating the performance of BO methods in the goal-achievement-inverse-design is also provided. With this benchmark method and toy problems, the

performance of the proposed MO BO method was compared with a classical SF-based baseline. Finally, to demonstrate the application of goal-oriented-inverse-design using the proposed method, virtual experiments of MO inverse material design were conducted.

RESULTS AND DISCUSSION

Goal-oriented MO BO

Underpinning BO are the machine-learning model and the acquisition function, and the role of the former is straightforward. Learning from past experimental results allows the BO method to recommend better design parameters for the next experiment. You may anticipate recommendations with design parameters featuring the best of such predicted objective properties. However, what is an excessively conservative strategy spawns locally optimal design parameters because even if better design parameters exist far from the current design parameters learned, the machine-learning model itself remains unaware of such design parameters. Conversely, an excessively challenging strategy, which always selects design parameters far from those existing, is also inefficient. Accordingly, mastering the balance of conservative and challenging experiments is crucial. Note that in the field of BO, conservative, and challenging experiments are referred to as exploitative and explorative experiments, respectively. To attack this problem in BO, the scoring function of next-design parameters, known as the acquisition function, is constructed using the machine-learning model mentioned previously, and the next-design parameters are obtained by optimizing the acquisition function. The role of the acquisition functions involves balancing the exploitative and explorative regions of design parameters based on the machine-learning model.

Among them, LCB is one of the most well-studied and used acquisition functions (Srinivas et al., 2010). Note that, this acquisition function is called the LCB for minimization problems, and also called upper confidence bound for maximization problems. As its name indicates, LCB is a function that maps design parameter X into an $(100-\alpha)\%$ LCB for the objective property value with design parameter X , where α is a predefined parameter controlling the balance of exploitative and explorative experiments. By considering the LCB, BO can be aware of not only exploitative design parameters with better expected properties but also explorative design parameters that show considerable potential for improving the objective property.

Formally, $(100-\alpha)\%$ LCB for objective values Y , under the design parameter X is written as follows:

$$\text{LCB}(X) = \text{ICDF}_X(\alpha / 100) \quad (\text{Equation 1})$$

where ICDF_X is the inverse cumulative distribution function of Y under the design parameter X . For simplicity purpose, a one-dimensional normal distribution defined by a mean μ and standard deviation σ obtained by a regression model is assumed.

$$Y \sim N(\mu(X), \sigma(X)) \quad (\text{Equation 2})$$

where Y and X are the single objective value and design parameter, respectively. In this case, using the inverse cumulative distribution function of the standard normal distribution Φ^{-1} , Equation 1 can also be written as:

$$\text{LCB}(X) = \mu(X) + \Phi^{-1}(\alpha / 100)\sigma(X) \quad (\text{Equation 3})$$

$$\text{LCB}(X) = \mu(X) - a\sigma(X), \quad (\text{Equation 4})$$

where $a = -\Phi^{-1}(\alpha / 100)$. Equation 4 is well used definition of LCB acquisition function, while in the following Equations 1 and 3 is assumed in order to discuss the probability α .

In the following, the LCB is first extended to the goal-achievement-inverse-design, then further still to the MO setting. The α in Equations 1 and 3 is the only tunable parameter of the LCB and tuning α according to the probability of goal achievement (PA) is a natural way of extending LCB to goal-achievement-inverse-design. Note that with smaller α , optimization strategy will be more explorative (Figure 3A, LCB minimum is different from the model prediction minimum.), whereas larger α will be more exploitative (Figure 3B, LCB minimum is same to the model prediction minimum.). Accordingly, it is possible to extend the LCB to the goal-achievement-inverse-design by controlling the parameter α depending on the distance from the goal as follows: If the predefined goal is far away from current designs explored i.e. the PA is small and exploitative experiments spawning small improvements are not promising, α should be set smaller. This makes optimization policy more explorative. In contrast, if the goal is proximal to current designs explored i.e. the PA is high and explorative experiments are wasteful, α should be set larger. This makes

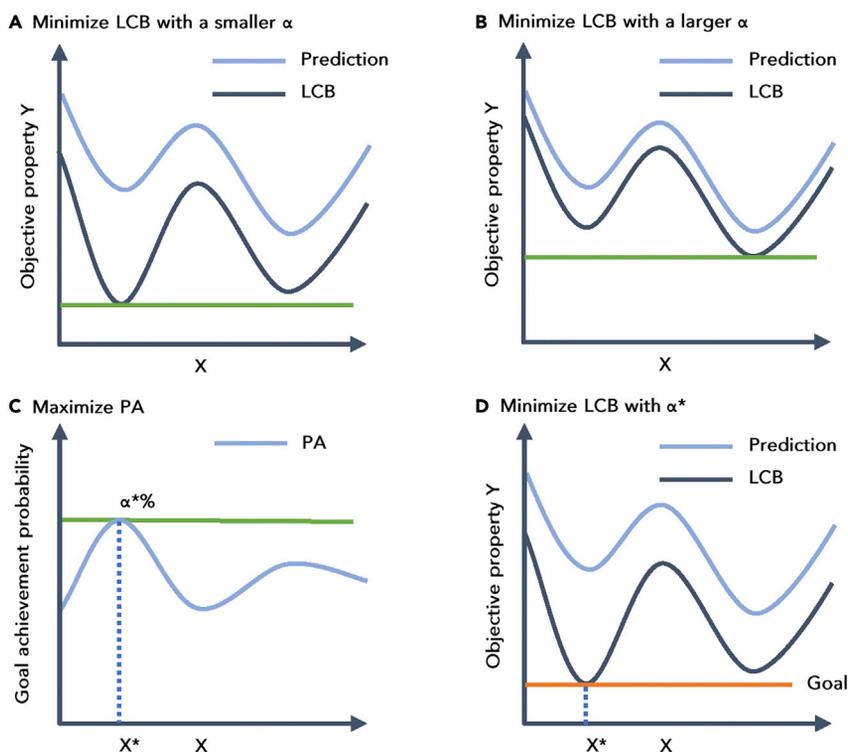


Figure 3. Schematic illustration of the optimization of acquisition functions

Green lines indicate optimal solutions for acquisition functions. An orange line indicates both the optimal solution and goal value.

(A) Minimization of the LCB with smaller α . LCB minimum is different from the model prediction minimum.

(B) Minimization of the LCB with larger α . LCB minimum is same to the model prediction minimum.

(C) Maximization of the PA. X^* is the optimal design parameter

(D) Minimization of the LCB with α^* .

optimization policy more exploitative. Indeed, such tuning of α can be easily realized by replacing the LCB acquisition function with the PA. Although LCB and PA are distinct functions, the design parameters obtained by minimizing LCB and maximizing PA are the same. This can be easily confirmed as follows: Assume, X^* is the optimal design parameter obtained by maximizing the PA and the goal achievement probability with X^* is $\alpha^*\%$ (Figure 3C) (See, Equivalence between maximization of PA and minimization of LCB in STAR Methods for the formal description).

Note that the $(100-\alpha^*)\%$ LCB under the design parameter X^* is the value where the probability of the observed property Y falling above this point under the design parameter X^* is $(100-\alpha^*)\%$, whereas the probability of the observed objective property Y falling above the goal value under the design parameter X^* is also $(100-\alpha^*)\%$.

Accordingly, the $(100-\alpha^*)\%$ LCB under the design parameter X^* is equal to the goal value. Figure 4 shows the relation between the goal value and LCB. The goal is achieved and not achieved below and above the red line indicating the goal value in Figure 4, respectively. As mentioned above, under design parameter X^* , the red line in Figure 4 also indicates the $(100-\alpha^*)\%$ LCB. If there is a design parameter X^{**} that results in a lower value than X^* in the $(100-\alpha^*)\%$ LCB, as shown in Figure 4, the $(100-\alpha^*)\%$ LCB of X^{**} is also lower than the red line indicating the goal, and the PA with design parameter X^{**} exceeds $\alpha^*\%$ by a probability $\epsilon\%$, which contradicts the assumption that the maximum value of the goal achievement probability is $\alpha^*\%$.

Accordingly, the $(100-\alpha^*)\%$ LCB contacts the line of the goal value with design parameter X^* (Figure 3D), and the true solution of minimizing LCB and maximizing PA are the same. Namely, using PA as an acquisition function corresponds to LCB with an automatically controlled α . And using the PA, as targeted, α will be smaller and the optimization strategy more explorative if the predefined goal is far from current designs

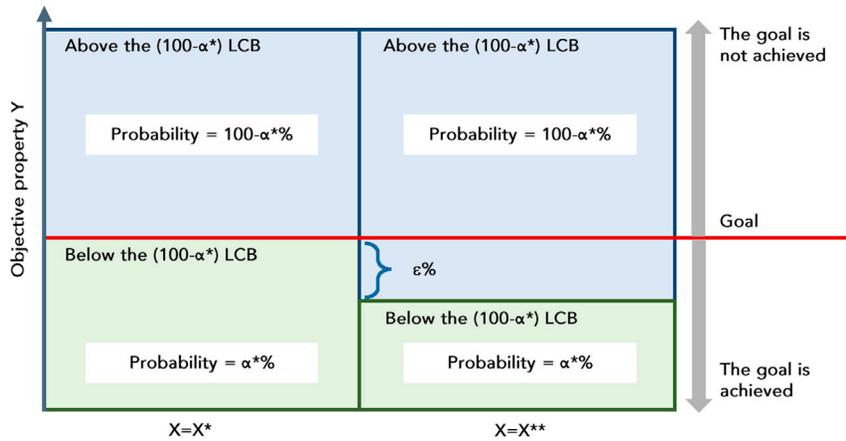


Figure 4. Relation between the lower confidence bound and the goal achievement probability

X^* is the optimal design parameter in the probability of achievement and the goal achievement probability with X^* is α^* %. The probabilities that the Y falls in the blue and green regions are $100-\alpha^*$ and α^* , respectively. A design parameter X^{**} that has a lower value of the $(100-\alpha^*)$ LCB than X^* does not exist, and optimal design parameter in the $(100-\alpha^*)$ % LCB is also X^* .

explored. Conversely, α will be larger and the optimization strategy more exploitative if the goal is proximal to current designs explored.

Extending the PA to a MO problem is simple, by leveraging the joint probability of all objective properties achieving the predefined goals. Assuming observed values of M objective properties with design parameter X are independent, the MO PA can be written as the product of the goal achievement probability for m -th property, PA_m :

$$PA(X) = \prod_{m=1}^M PA_m(X) \quad (\text{Equation 5})$$

This acquisition function can automatically balance the objectives according to the joint probability of each objective property achieving the predefined goal and no ad hoc rules or parameters are used to balance multiple objective properties. Accordingly, the LCB was extended to the MO goal-achievement-inverse-design, in a fully probabilistic manner using the PA acquisition function.

Benchmark method for goal-achievement-inverse-design

To examine the PA performance in the goal-achievement-inverse-design, a rigid benchmark method for goal-achievement-inverse-design was also developed. As things stand, the performance of optimization algorithms often depends on setting of problems. For example, an algorithm that is suitable for achieving a goal A may be unsuitable for achieving another goal B. Accordingly, evaluating the performance of algorithms for a single setting of goals is insufficient. To overcome this problem, the performance of algorithms was evaluated using optimization results obtained from 1,000 random samples of setting of balance of goals, where goals were sampled from uniform distribution with minimum and maximum values of each objective property in Pareto optimal solutions obtained by a genetic algorithm-based-MO optimization.

To evaluate the performances of optimization algorithms, a quantitative measure of quality of optimization results is also required. In goal-achievement-inverse-design, this can be simply defined as the number of experiments performed before achieving the predefined goals. Accordingly, in the following benchmarks, the performances of optimization algorithms were compared by the average number of experiments performed before achieving the predefined goals or rate of optimization trajectories that achieved goals within the same time.

Benchmarks with toy problems

In benchmarks, for comparison, the performances of an SF-based approach that employs the achievement function which is capable of treating predefined goals (Hakanen and Knowles, 2017; Wierzbicki, 2007) were

also evaluated as a baseline. The achievement function have been used in the field of the operations research. By scalarizing multiple objective properties using the achievement function, the single-objective BO can be applied for the value of the achievement function. LCB acquisition was used for this single-objective BO of the achievement function. See Calculation of the achievement function in [STAR Methods](#) for details about how the achievement function weights and merges multiple objectives, and how the LCB acquisition was calculated.

The performances of the proposed BO method with the PA and the achievement baseline were evaluated for six mathematical MO toy functions ([Figure 5](#)) with 1,000 randomly sampled predefined goals. These functions have been used to evaluate the performances of MO optimization methods ([Huband et al., 2006](#)). The toy functions included relatively simple low dimensional problems, Fonseca, Kursawe, and Viennet and also include complex high-dimensional problems, ZDT1, ZDT2, and ZDT3. Although, the goal-achievement-inverse-design will be finished after achieving the goal, 200 steps of BOs were performed for all optimization runs for comparison purposes.

[Figure 6](#) shows the rate of optimization runs having achieved goals for each step in 200 steps of optimizations, which is referred to as the goal achievement rate (GAR) in this study. See Performance metrics in [STAR Methods](#) for formal definition of the GAR. As expected for all six benchmarks, the PA dramatically reduced the optimization steps required to achieve the goals relative to the achievement baseline. For easier problems, Fonseca and Kursawe, the BO with the PA achieved most of the goals in the early stage of optimization, far outpacing the achievement baseline. For more difficult problems, Viennet, ZDT1, ZDT2, and ZDT3, even with the PA, the GAR did not reach 1, indicating that some of the randomly sampled goals remained unachieved within 200 steps of BO. However, the GAR after 200 steps of BO using the PA far exceeds that using the achievement baseline, particularly for high-dimensional problems ZDT1, ZDT2, and ZDT3, where the number of design parameters is 30 and the achievement baseline cannot be achieved most of goals randomly sampled for these high-dimensional problems.

There are two possible reasons explaining these significant performance gaps. The first possible reason is that BO did not work well with LCB and the achievement function and the second is the difficulty in properly balancing the objective properties with the achievement function, particularly with complex and high-dimensional design parameters. To investigate the first possible reason, convergences of optimization trajectories to the Pareto optimal solutions are monitored alongside optimization steps. Note that this analysis corresponds to evaluating the performances of optimization methods from the point of view of few-solution-inverse-design.

[Figure 7](#) shows the time evolutions of the average minimum distances (AMDs) between the objective values of Pareto optimal solutions and current designs explored in log scale. See Performance metrics in [STAR Methods](#) for the formal definition of the AMD.

Unlike performance evaluation according to the goal-achievement-inverse-design shown in [Figure 6](#), the performances of both methods in the few-solution-inverse-design compete. Although, for the Viennet, ZDT1, and ZDT3, the PA converged to the Pareto optimal solutions faster relative to the achievement baseline, for Fonseca, Kursawe, and ZDT2, the achievement baseline showed better convergence relative to the PA. It is notable that from the point of view of few-solution-inverse-design, the achievement baseline performed far better relative to the PA for the high-dimensional ZDT2 problem, while from the point of view of goal-achievement-inverse-design, the achievement baseline showed much worse performance relative to the PA for this problem ([Figure 6](#)). Accordingly, at least, for Fonseca, Kursawe, and ZDT2, the main reason for improved GAR on the part of PA relative to the achievement baseline is attributable to the exquisite balancing between multiple objective properties made possible by the fully probabilistic approach. And the benchmarks clearly showed that by focusing on finding a design parameter of materials achieving predefined goal values rather than the Pareto optimal solutions, BO with the PA can efficiently accelerate MO optimization problems.

Virtual MO inverse material design

Next, to demonstrate an application of goal-achievement-inverse-design by using the BO with the PA for realistic material design problem, virtual material-design experiments were conducted using regression models constructed from experimental data as a substitute for time-consuming real-world experiments.

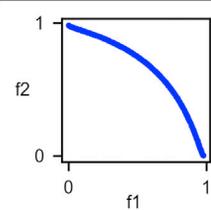
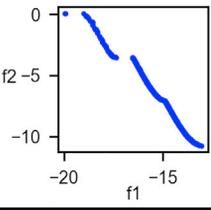
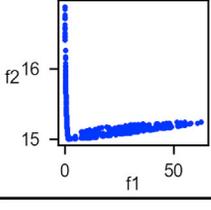
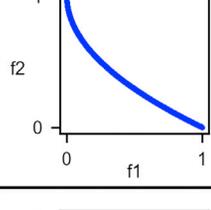
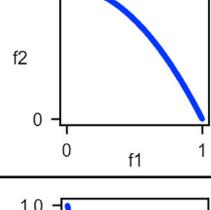
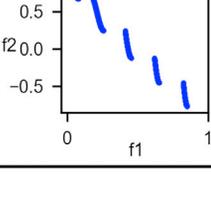
	Problem	Pareto optimal solutions
Fonseca	$\text{Minimize} = \begin{cases} f_1(X) = 1 - \exp\left[-\sum_{d=1}^D \left(x_d - \frac{1}{\sqrt{D}}\right)^2\right] \\ f_2(X) = 1 - \exp\left[-\sum_{d=1}^D \left(x_d + \frac{1}{\sqrt{D}}\right)^2\right] \\ -4 \leq x_d \leq 4, \quad D = 2 \end{cases}$	
Kursawe	$\text{Minimize} = \begin{cases} f_1(X) = \sum_{d=1}^2 \left[-10 \exp\left(-0.2 \sqrt{x_d^2 + x_{d+1}^2}\right)\right] \\ f_2(X) = \sum_{d=1}^3 \left[x_d ^{0.8} + 5 \sin x_d^3\right] \\ -5 \leq x_d \leq 5 \end{cases}$	
Viennet	$\text{Minimize} = \begin{cases} f_1(X) = 0.5(x_1^2 + x_2^2) + \sin(x_1^2 + x_2^2) \\ f_2(X) = \frac{(3x_1 - 2x_2 + 4)^2}{8} + \frac{(x_1 - x_2 + 1)^2}{27} + 15 \\ f_3(X) = \frac{1}{x_1^2 + x_2^2 + 1} - 1.1 \exp[-(x_1^2 + x_2^2)] \\ -30 \leq x_1, x_2 \leq 30 \end{cases}$	
ZDT1	$\text{Minimize} = \begin{cases} g(X) = 1 + \frac{9}{D-1} \sum_{d=2}^D x_d \\ f_1(X) = x_1 \\ f_2(X) = g(X) \left[1 - \sqrt{\frac{x_1}{g(X)}}\right] \\ 0 \leq x_d \leq 1, \quad D = 30 \end{cases}$	
ZDT2	$\text{Minimize} = \begin{cases} g(X) = 1 + \frac{9}{D-1} \sum_{d=2}^D x_d \\ f_1(X) = x_1 \\ f_2(X) = g(X) \left[1 - \left(\frac{x_1}{g(X)}\right)^2\right] \\ 0 \leq x_d \leq 1, \quad D = 30 \end{cases}$	
ZDT3	$\text{Minimize} = \begin{cases} g(X) = 1 + \frac{9}{D-1} \sum_{d=2}^D x_d \\ f_1(X) = x_1 \\ f_2(X) = g(X) \left[1 - \sqrt{\frac{x_1}{g(X)}} - \frac{x_1}{g(X)} \sin 10\pi x_1\right] \\ 0 \leq x_d \leq 1, \quad D = 30 \end{cases}$	

Figure 5. Mathematical toy problems used in the benchmarks

Objective values, f_1 and f_2 , for Pareto optimal solutions obtained by NSGA are also shown. All the Pareto optimal solutions used in this study are also provided as [Data S1](#).

The regression models were obtained from a recent MO material-design study conducted by Wang et al. (Wang et al., 2020a), where a high-performance oil sorbent material was developed by a method resembling BO with the weight-summation SF. The use of oil sorbent materials helps remove oil spilled on bodies of water, such as at sea, to mitigate ecological damage, for which materials with high contact angles, high oil absorption capacity, and high mechanical strength are known to be preferable. Wang et al. (Wang et al., 2020a) used these three parameters as objective properties in the optimization and seven design

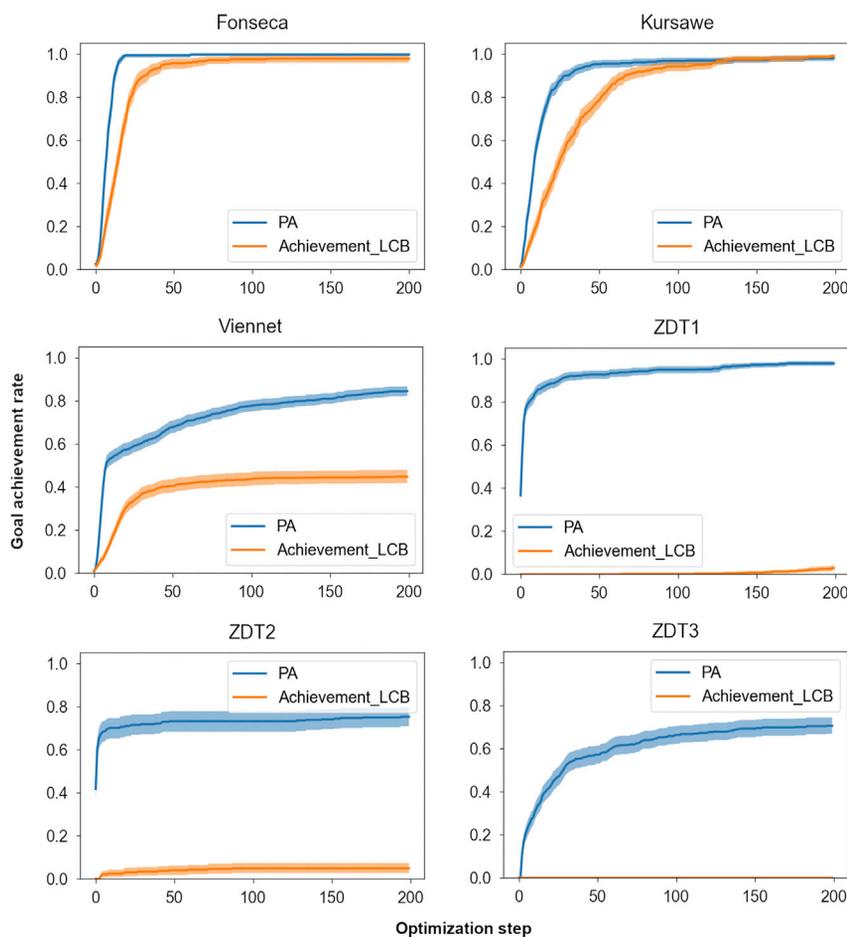


Figure 6. Time evolution of the goal achievement rate

The blue and orange lines represent the goal achievement rate, the rates of optimization runs having achieved goals for each step in 200 steps of optimization runs. And the shaded areas represent 95% confidence intervals of goal achievement rates obtained by bootstrap resampling. Note that only optimization runs with achievable predefined goals are included in this analysis, which is judged using approximated true Pareto optimal solutions obtained by a tried and tested genetic multi-objective optimization method, NSGA-II (Deb et al., 2002), with sufficient optimization steps. And the total of randomly sampled goals used to evaluate the goal achievement rate exceeds 300 for all toy problems (See also Table S1).

parameters shown in Table S2 were tuned. The design parameters include three parameters related to compositions of materials, polystyrene/polyacrylonitrile ratio, mass fraction of solute, and mass fraction of SiO₂ nanoparticles in solute and also include four parameters related to fabrication process, feed rate, receiving distance, applied voltage, and inner diameter of needle. In the following virtual material-design, the same objectives and design parameters are used. The regression models used here and further details of design parameters can be obtained from the work of Wang et al. (Wang et al., 2020a)

The goal-achievement-inverse-design started from defining the goal values of objective properties. In our virtual oil sorbent material design, five sets of goals, 1, 2, 3, 4, and 5, with different design objectives were defined using the oil sorbent material designed by Wang et al. in the previous study (Wang et al., 2020a) as a reference. Figure 8 shows values of the five sets of the goals scaled by the values of each property in the reference, and Table 1 shows the unscaled values of sets of properties for the five goals with short descriptions on their design objectives. The design objective of the goal 1 is to achieve same level of values relative to the reference in all the three properties. On the other hand, the design objective of the goal 2, 3, and 4 are to achieve higher values than the reference in the contact angle, oil-absorption capacity and mechanical strength, respectively. Finally, the design objective of the goal 5 is to achieve higher values in all the three properties relative to the reference. Table 1 also includes the number of random experiments

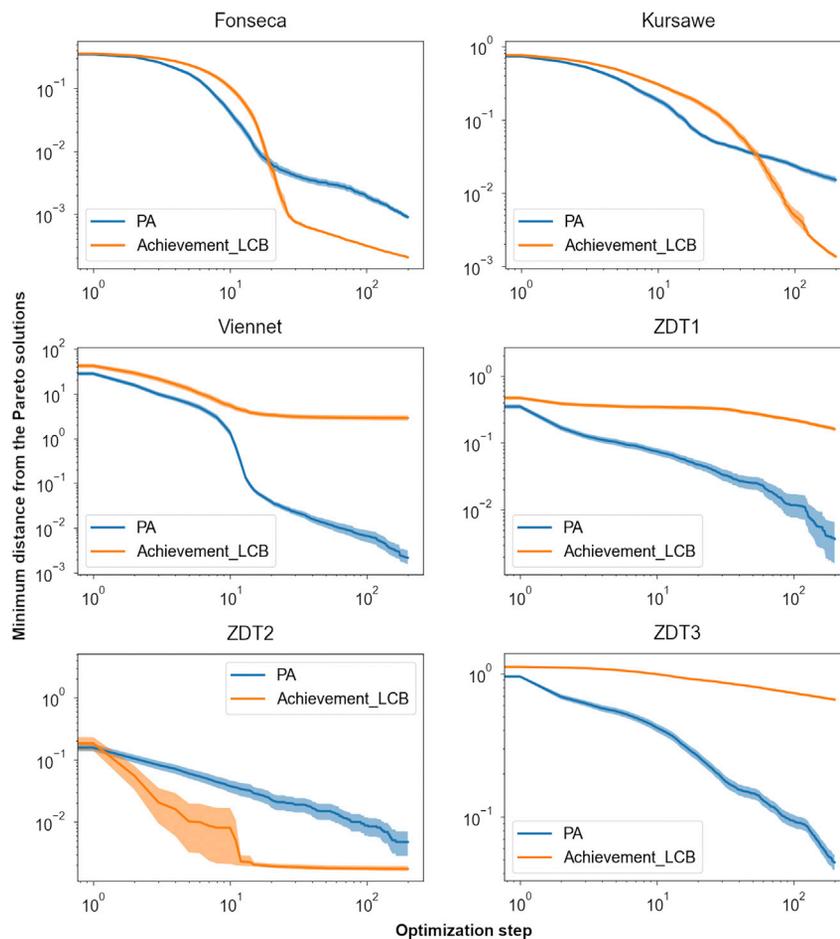


Figure 7. Time evolution of the average minimum distance from objective values in the Pareto optimal solutions Objective values were scaled by the minimum and maximum values in the Pareto optimal solutions for each problem. The blue and orange lines represent the average minimum distances from Pareto optimal solutions within the 1,000 optimization runs and shaded areas represent 95% confidence intervals of average minimum distances obtained by bootstrap resampling.

required to achieve each goal, which indicates the difficulty of each goal, and even for the easiest goal, 1, over 100 experiments are needed to achieve the goals in average. Furthermore, for the most difficult goal, 5, over 10,000 experiments are needed to achieve the goals, which is inaccessible for real world experiments.

Next, virtual experiments for ten randomly chosen design parameters were conducted, which is needed to initialize the machine-learning model. And MO BO steps were repeated pending achievement of the pre-defined goals. For statistical purposes, these processes were repeated 100 times by randomly changing the initial ten design parameters. Note that, randomly chosen design parameters that achieve the goals were not used as initial design parameters for BO. The performance of BO with the PA in the goal-achievement-inverse-design was evaluated by calculating average number of BO steps required to achieve goals, and comparison between the obtained results and the performance of random sampling, which is also shown in Table 1, are summarized in Figure 9A in log scale. The bar graphs clearly showed that the BO with the PA can efficiently achieve the goals within small numbers of experiments (5–12). Even for the easiest goal, 1, BO with the PA achieved it over 30 times faster than random experiments, while for the most difficult goal, 5, BO with the PA achieved the goal over 1,000 times faster than random experiments. Accordingly, using the PA in goal-achievement-inverse-design can pave the way for efficiently solving MO inverse material design problems that are difficult for the random experiment.

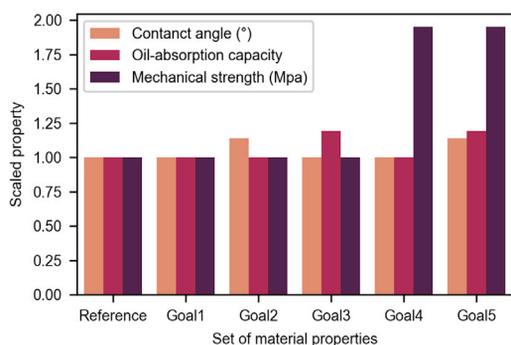


Figure 8. Scaled values of the goals used in the virtual material design

See also [Table S2](#).

In order to reconfirm the superiority of the PA relative to the achievement baseline in realistic inverse design problems, the performances of the BO using the PA and the achievement baseline were also compared and summarized in [Figure 9B](#). As with the benchmarks using the mathematical toy functions, BO with the PA clearly outperformed the achievement baseline for all five settings of goals, and consistently showed around 2 times acceleration relative to the achievement baseline.

Finally, in order to demonstrate the reason why the goal-achievement-inverse-design is suitable for real-world material design problems, experimental costs for finding a Pareto optimal solution were compared with that for achieving the goals, by extending the BO steps for all the virtual experiments, and evaluating numbers of optimization steps required to find a first Pareto optimal solution. In this analysis, only the Pareto optimal solutions that achieve the predefined goals were regarded as the solutions, and therefore the obtained Pareto optimal solutions have equivalent or better values in all the objective properties compared to the goal values. See, Judgment of Pareto optimal solutions in [STAR Methods](#) for details about how to judge whether obtained objective properties reached Pareto optimal solutions. Because, the achievement baseline showed better performances than the PA in finding the Pareto optimal solutions for this virtual experiment system ([Table S3](#)), the results for the achievement baseline were used in the following comparison.

Average numbers of experiments required to find a first Pareto optimal solution using the achievement baseline are shown in [Figure 9C](#) with that required to achieve the goals using the PA. As expected, even with the BO, finding of Pareto optimal solutions is much more difficult than achieving the goals. Finding of Pareto optimal solutions required over two times more experiments than achieving the goals in all the setting of goals. Furthermore, for easier goals (goals 1 and 2) gaps between the difficulty of achieving the goals and finding Pareto optimal solutions were especially large, and finding Pareto optimal solutions required around four times more experiments than achieving the goals. Such differences in the required numbers of experiments are critical when budget and time available for the material development are limited and experimenters do not want to pay much experimental cost after finding a material that achieves the goals and already has properties required for

Table 1. Predefined goals for the virtual material design

Goal	Contact angle (°)	Oil-absorption capacity	Mechanical strength (Mpa)	Design objective	Required number of random experiments ^a
Goal 1	140.1	83.7	4.1	Same level of the reference	158.6 (149.7-168.0)
Goal 2	160.0	83.7	4.1	Higher contact angle	256.1 (240.5-271.9)
Goal 3	140.1	100	4.1	Higher oil-absorption capacity	333.6 (313.4-354.1)
Goal 4	140.1	83.7	8.0	Higher mechanical strength	5499.6 (5136.9-5876.7)
Goal 5	160.0	100.0	8.0	Improve all properties	13760.8 (12946.5-14632.8)

^aAverage number of experiments required to achieve the each set of goals was evaluated by 1000 sequences of random experiments. 95% confidence intervals estimated by bootstrap resampling are shown in parentheses.

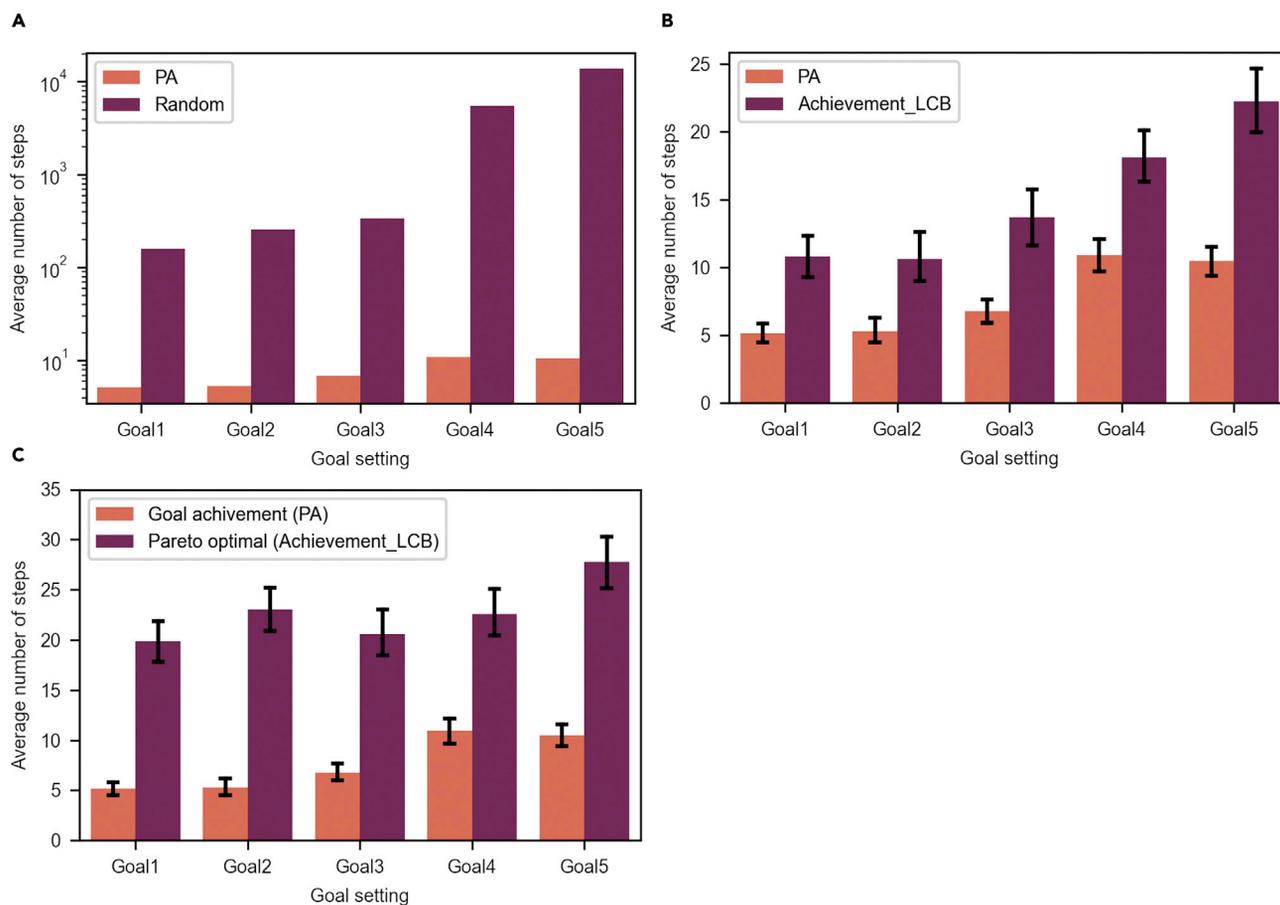


Figure 9. Average number of steps required to achieve the goals

(A) Comparison between the PA and random sampling in log scale. For random sampling, average number of experiments required to achieve the each set of goals was evaluated by 1,000 sequences of random experiments.

(B) Comparison between the PA and achievement baseline.

(C) Average number of steps required to achieve the goals by using the PA is compared with that required to find a first Pareto optimal solution by using the achievement baseline. Error bars indicate the 95% confidence interval of the average number of steps estimated by bootstrap resampling.

See also [Tables S2](#) and [S3](#).

applications. And such situations often occur in materials design processes requiring time-consuming real-world experiments, especially when experimenters are in competition with others and the speed of material design is important. Therefore, the goal-achievement-inverse-design using the PA, which efficiently accelerate solving the realistic design problem, goal achievement, is expected to be attractive for such situations. Indeed, in this demonstration, around 10 experiments were required to achieve the goals and design high-performance oil sorbent materials, and this experimental cost should be acceptable for many actual material design problems with time-consuming real-world experiments.

It is worth to mention that another merit of the goal-achievement-inverse-design compared with other methods intending to find Pareto optimal solutions is its simplicity in the criterion for stopping the optimization. In other words, experimenter conducting the goal-achievement-inverse-design can stop optimization efforts immediately after the goal achievement. While, even if methods intending to find Pareto optimal solutions are employed, it is often difficult to use the Pareto optimality as a criterion for stopping the optimization, because whether obtained solutions reached near the true Pareto optimal solutions can only be confirmed when whole the Pareto optimal solutions have been found after massive experiments. And there are no common reasonable criteria for stopping the optimization in such a situation.

Conclusion

In this study, the goal-achievement-inverse-design for the MO material design was introduced with the fully probabilistic acquisition function, PA and the rigid benchmark method. In the benchmarks, using the six mathematical functions, the performances of the PA and the achievement baseline were compared and the benchmark results showed that by focusing on finding a design parameter achieving predefined goal values rather than the Pareto optimal solution, BO with the PA dramatically outperformed the achievement baseline for all the six mathematical functions in the rate of optimization runs that achieved the goals. And performance improvement of the PA relative to the achievement baseline were much larger for complex optimization problems with more objectives or more design parameters. In addition, the application of goal-achievement-inverse-design with the PA was demonstrated using more realistic virtual material-design problems to achieve the five goals with different design objectives, where the BO with the PA again outperformed the achievement baseline, and the BO with the PA achieved the goals over 1,000 times faster than the random sampling for the most difficult case. Furthermore, in this virtual inverse material design, number of experiments required to achieve the goals were around ten, which is over two times smaller than that required to find Pareto optimal solutions and would be acceptable for most real-world material design problems with time-consuming experiments. The proposed inverse design method that works with small number of possible experiments will precede the real-world implementation of MO inverse material design, where time-consuming experiments are often required.

Limitations of the study

The inverse design approach proposed in this study can only be applied for design problems, where a quantitative and reasonable goal can be defined for each objective. And the proposed approach is not suitable for design problems where it is difficult to set reasonable goals. Additionally, the goal-achievement-inverse-design is also not suitable for the design problems, where massive number of experiments can be conducted and experimenter want to find as good solutions as possible. A typical example of such a design problem is computational materials design using molecular simulation approaches with low computational cost.

STAR★METHODS

Detailed methods are provided in the online version of this paper and include the following:

- KEY RESOURCES TABLE
- RESOURCE AVAILABILITY
 - Lead contact
 - Materials availability
 - Data and code availability
- METHOD DETAILS
 - Equivalence between maximization of PA and minimization of LCB
 - Performance metrics
- BAYESIAN OPTIMIZATION
- CALCULATION OF THE PROBABILITY OF ACHIEVEMENT
 - Calculation of the achievement function
 - Optimization of the acquisition function
 - Calculation of pareto optimal solutions
 - Regression models for virtual material design experiment
 - Judgment of Pareto optimal solutions
- QUANTIFICATION AND STATISTICAL ANALYSIS

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at <https://doi.org/10.1016/j.isci.2021.102781>.

ACKNOWLEDGMENTS

This work was supported by Showa Denko Materials Co., Ltd. The author gratefully acknowledges all the support from Showa Denko Materials Co., Ltd and his colleagues. The author also gratefully acknowledges Dr. Masanori Sakai for discussions on a traditional method for product design called the Nakazawa method.

AUTHOR CONTRIBUTIONS

The work was done by a single author.

DECLARATION OF INTERESTS

The author applied for patents related to this work and declares no other competing interest.

Received: March 28, 2021

Revised: June 1, 2021

Accepted: June 21, 2021

Published: July 23, 2021

REFERENCES

- Balachandran, P.V., Xue, D., Theiler, J., Hogden, J., and Lookman, T. (2016). Adaptive strategies for materials design using uncertainties. *Sci. Rep.* **6**, 19660.
- Balachandran, P.V., Kowalski, B., Sehirlioglu, A., and Lookman, T. (2018). Experimental search for high-temperature ferroelectric perovskites guided by two-step machine learning. *Nat. Commun.* **9**, 1668.
- Bassman, L., Rajak, P., Kalia, R.K., Nakano, A., Sha, F., Sun, J., Singh, D.J., Aykol, M., Huck, P., Persson, K., et al. (2018). Active learning for accelerated design of layered materials. *NPJ Comput. Mater.* **4**, 74.
- Cummins, D.J., and Bell, M.A. (2016). Integrating everything: the molecule selection toolkit, a system for compound prioritization in drug discovery. *J. Med. Chem.* **59**, 6999–7010.
- Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **6**, 182–197.
- Doan, H.A., Agarwal, G., Qian, H., Coughlan, M.J., Rodríguez-López, J., Moore, J.S., and Assary, R.S. (2020). Quantum chemistry-informed active learning to accelerate the design and discovery of sustainable energy storage materials. *Chem. Mater.* **32**, 6338–6346.
- Fukazawa, T., Harashima, Y., Hou, Z., and Miyake, T. (2019). Bayesian optimization of chemical composition: a comprehensive framework and its application to RFe₁₂-type magnet compounds. *Phys. Rev. Mater.* **3**, 053807.
- Gopakumar, A.M., Balachandran, P.V., Xue, D., Gubernatis, J.E., and Lookman, T. (2018). Multi-objective optimization for materials discovery via adaptive design. *Sci. Rep.* **8**, 3738.
- Greenhill, S., Rana, S., Gupta, S., Vellanki, P., and Venkatesh, S. (2020). Bayesian optimization for adaptive experimental design: a review. *IEEE Access* **8**, 13937–13948.
- Hakanen, J., and Knowles, J.D. (2017). On using decision maker preferences with ParEGO. In *Evolutionary Multi-Criterion Optimization : 9th International Conference, EMO 2017, Münster, Germany, March 19-22, 2017, Proceedings*, pp. 282–297.
- Harada, M., Takeda, H., Suzuki, S., Nakano, K., Tanibata, N., Nakayama, M., Karasuyama, M., and Takeuchi, I. (2020). Bayesian-optimization-guided experimental search of NASICON-type solid electrolytes for all-solid-state Li-ion batteries. *J. Mater. Chem. A* **8**, 15103–15109.
- Häse, F., Roch, L.M., and Aspuru-Guzik, A. (2018). Chimera: enabling hierarchy based multi-objective optimization for self-driving laboratories. *Chem. Sci.* **9**, 7642–7655.
- Hashimoto, W., Tsuji, Y., and Yoshizawa, K. (2020). Optimization of work function via Bayesian machine learning combined with first-principles calculation. *J. Phys. Chem. C* **124**, 9958–9970.
- Herbol, H.C., Hu, W., Frazier, P., Clancy, P., and Poloczek, M. (2018). Efficient search of compositional space for hybrid organic–inorganic perovskites via Bayesian optimization. *Npj Comput. Mater.* **4**, 1–7.
- Homma, K., Liu, Y., Sumita, M., Tamura, R., Fushimi, N., Iwata, J., Tsuda, K., and Kaneta, C. (2020). Optimization of a heterogeneous ternary Li₃PO₄-Li₃BO₃-Li₂SO₄ Mixture for Li-ion conductivity by machine learning. *J. Phys. Chem. C* **124**, 12865–12870.
- Huband, S., Hingston, P., Barone, L., and While, L. (2006). A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Trans. Evol. Comput.* **10**, 477–506.
- Inage, S.ichi (2019). Proposal of the “total error minimization method” for robust design. *Eng. Sci. Technol. Int. J.* **22**, 656–666.
- Janet, J.P., Ramesh, S., Duan, C., and Kulik, H.J. (2020). Accurate multiobjective design in a space of millions of transition metal complexes with neural-network-driven efficient global optimization. *ACS Cent. Sci.* **6**, 513–524.
- Jung, Y.H., Park, W.B., Pyo, M., Sohn, K.S., and Ahn, D. (2017). A multi-element doping design for a high-performance LiMnPO₄ cathode: via metaheuristic computation. *J. Mater. Chem. A* **5**, 8939–8945.
- Karasuyama, M., Kasugai, H., Tamura, T., and Shitara, K. (2020). Computational design of stable and highly ion-conductive materials using multi-objective Bayesian optimization: case studies on diffusion of oxygen and lithium. *Comput. Mater. Sci.* **184**, 109927.
- Langner, S., Häse, F., Perea, J.D., Stubhan, T., Hauch, J., Roch, L.M., Heumueller, T., Aspuru-Guzik, A., and Brabec, C.J. (2020). Beyond ternary OPV: high-throughput experimentation and self-driving laboratories optimize multicomponent systems. *Adv. Mater.* **32**, 1907801.
- Lee, J.W., Singh, S.P., Kim, M., Hong, S.U., Park, W.B., and Sohn, K.S. (2017). Metaheuristics-Assisted combinatorial screening of Eu²⁺-doped Ca-Sr-Ba-Li-Mg-Al-Si-Ge-N compositional space in search of a narrow-band green emitting phosphor and density functional theory calculations. *Inorg. Chem.* **56**, 9814–9824.
- Lookman, T., Balachandran, P.V., Xue, D., Hogden, J., and Theiler, J. (2017). Statistical inference and adaptive design for materials discovery. *Curr. Opin. Solid State Mater. Sci.* **21**, 121–128.
- Lookman, T., Balachandran, P.V., Xue, D., and Yuan, R. (2019). Active learning in materials science with emphasis on adaptive sampling using uncertainties for targeted design. *NPJ Comput. Mater.* **5**, 21.
- Mannodi-Kanakkithodi, A., Pilania, G., Ramprasad, R., Lookman, T., and Gubernatis, J.E. (2016). Multi-objective optimization techniques to design the Pareto front of organic dielectric polymers. *Comput. Mater. Sci.* **125**, 92–99.
- Menou, E., Toda-Caraballo, I., Rivera-Díaz-del-Castillo, P.E.J., Pineau, C., Bertrand, E., Ramstein, G., and Tancret, F. (2018). Evolutionary design of strong and stable high entropy alloys using multi-objective optimisation based on physical models, statistics and thermodynamics. *Mater. Des.* **143**, 185–195.
- Niu, B., Jia, M., Xu, G., Chang, Y., and Xie, M. (2018). Efficient approach for the optimization of skeletal chemical mechanisms with multiobjective genetic algorithm. *Energy Fuels* **32**, 7086–7102.
- Okamoto, Y. (2017). Applying Bayesian approach to combinatorial problem in chemistry. *J. Phys. Chem. A* **121**, 3299–3304.
- Del Rosario, Z., Rupp, M., Kim, Y., Antono, E., and Ling, J. (2020). Assessing the frontier: active learning, model accuracy, and multi-objective candidate discovery and optimization. *J. Chem. Phys.* **153**, 024112.
- Rouet-Leduc, B., Barros, K., Lookman, T., and Humphreys, C.J. (2016). Optimisation of GaN LEDs and the reduction of efficiency droop using active machine learning. *Sci. Rep.* **6**, 24862.
- Sakurai, A., Yada, K., Simomura, T., Ju, S., Kashiwagi, M., Okada, H., Nagao, T., Tsuda, K.,

- and Shiomi, J. (2019). Ultranarrow-band wavelength-selective thermal emission with aperiodic multilayered metamaterials designed by Bayesian optimization. *ACS Cent. Sci.* *5*, 319–326.
- Seko, A., Togo, A., Hayashi, H., Tsuda, K., Chaput, L., and Tanaka, I. (2015). Prediction of low-thermal-conductivity compounds with first-principles anharmonic lattice-dynamics calculations and Bayesian optimization. *Phys. Rev. Lett.* *115*, 205901.
- Shahriari, B., Swersky, K., Wang, Z., Adams, R.P., and De Freitas, N. (2015). Taking the human out of the loop: a review of Bayesian optimization. *Proc. IEEE* *104*, 148–175.
- Shrivastava, S., Mohite, P.M., Yadav, T., and Malagaudanavar, A. (2018). Multi-objective multi-laminate design and optimization of a Carbon Fibre Composite wing torsion box using evolutionary algorithm. *Compos. Struct.* *185*, 132–147.
- Solomou, A., Zhao, G., Boluki, S., Joy, J.K., Qian, X., Karaman, I., Arróyave, R., and Lagoudas, D.C. (2018). Multi-objective Bayesian materials discovery: application on the discovery of precipitation strengthened NiTi shape memory alloys through micromechanical modeling. *Mater. Des.* *160*, 810–827.
- Srinivas, N., Krause, A., and Seeger, M. (2010). Gaussian process optimization in the bandit setting: No regret and experimental design. In *Proceedings of the 27th International Conference on Machine Learning*.
- Talapatra, A., Boluki, S., Duong, T., Qian, X., Dougherty, E., and Arróyave, R. (2018). Autonomous efficient experiment design for materials discovery with Bayesian model averaging. *Phys. Rev. Mater.* *2*, 113803.
- Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., et al. (2020). SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nat. Methods* *17*, 261–272.
- Wakabayashi, Y.K., Otsuka, T., Krockenberger, Y., Sawada, H., Taniyasu, Y., and Yamamoto, H. (2019). Machine-learning-assisted thin-film growth: Bayesian optimization in molecular beam epitaxy of SrRuO₃ thin films. *APL Mater.* *7*, 101114.
- Walker, B.E., Bannock, J.H., Nightingale, A.M., and Demello, J.C. (2017). Tuning reaction products by constrained optimisation. *React. Chem. Eng.* *2*, 785–798.
- Wang, B., Cai, J., Liu, C., Yang, J., and Ding, X. (2020a). Harnessing a novel machine-learning-assisted evolutionary algorithm to Co-optimize three characteristics of an electrospun oil sorbent. *ACS Appl. Mater. Inter.* *12*, 42842–42849.
- Wang, Y., Iyer, A., Chen, W., and Rondinelli, J.M. (2020b). Featureless adaptive optimization accelerates functional electronic materials design. *Appl. Phys. Rev.* *7*, 041403.
- Wheatle, B.K., Fuentes, E.F., Lynd, N.A., and Ganesan, V. (2020). Design of polymer blend electrolytes through a machine learning approach. *Macromolecules* *53*, 9449–9459.
- Wierzbicki, A.P. (2007). Reference point Approaches and objective ranking. In *Practical Approaches to Multi-Objective Optimization, Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI)*, J. Branke, K. Deb, K. Miettinen, and R. Slowinski, eds. (Germany: Schloss Dagstuhl), p. 06501.
- Xue, D., Balachandran, P.V., Hogden, J., Theiler, J., Xue, D., and Lookman, T. (2016). Accelerated search for materials with targeted properties by adaptive design. *Nat. Commun.* *7*, 11241.
- Yamawaki, M., Ohnishi, M., Ju, S., and Shiomi, J. (2018). Multifunctional structural design of graphene thermoelectrics by Bayesian optimization. *Sci. Adv.* *4*, eaar4192.
- Yuan, R., Liu, Z., Balachandran, P.V., Xue, D., Zhou, Y., Ding, X., Sun, J., Xue, D., and Lookman, T. (2018). Accelerated discovery of large electrostrains in BaTiO₃-based piezoelectrics using active learning. *Adv. Mater.* *30*, 1702884.

STAR★METHODS

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Deposited Data		
Regression model for oil sorbent materials	Wang et al., 2020a, 2020b, supplemental information	https://pubs.acs.org/doi/abs/10.1021/acsami.0c11667
Software and algorithms		
GPy version 1.9	Sheffield Machine Learning Software	https://github.com/SheffieldML/GPy
GpyOpt version 1.2	Sheffield Machine Learning Software	https://github.com/SheffieldML/GPyOpt
Platypus version 1.0	Platypus - Multiobjective Optimization in Python	https://platypus.readthedocs.io/en/latest/
SciPy version 1.5	Virtanen et al., 2020	https://www.scipy.org/
Python version 3.6	Python Software Foundation	https://www.python.org
Code for multi-objective Bayesian optimization using the PA and achievement baseline	This paper; Mendeley Data	https://data.mendeley.com/datasets/fg5ngjism79/1

RESOURCE AVAILABILITY

Lead contact

Further information and requests for resources and reagents should be directed to and will be fulfilled by the lead contact, Kyohei Hanaoka (hanaoka.kyohei.xicmq@showadenko.com).

Materials availability

This study did not generate new reagents.

Data and code availability

The code required to reproduce the results of this study is available at Mendeley Data: <https://data.mendeley.com/datasets/fg5ngjism79/1>. This code can be freely used for scientific purposes.

METHOD DETAILS

Equivalence between maximization of PA and minimization of LCB

In the following, Y is assumed to be a random variable whose cumulative distribution function is strictly monotonically increasing. Note that the cumulative distribution function of the well-used normal distribution is also strictly monotonically increasing. And X is a set of design parameters that determines the shape of the cumulative distribution function of the random variable Y . For clarity, random variable Y under design parameter X is written as Y_X .

Assume, X^* is the optimal design parameter obtained by maximizing the PA and the goal achievement probability with X^* is α^* %. That is, the probability that random variable Y_{X^*} falls below the predefined goal g is given by

$$P(Y_{X^*} < g) = \alpha^*/100 \quad (\text{Equation 6})$$

From the definition of the LCB, the probability that random variable Y_{X^*} falls below the $(100-\alpha^*)\%$ LCB under design parameter X^* is given by

$$P(Y_{X^*} < \text{LCB}_{\alpha^*}(X^*)) = \alpha^*/100 = P(Y_{X^*} < g) \quad (\text{Equation 7})$$

, where LCB_{α^*} represents $(100-\alpha^*)\%$ LCB. From Equation 7, because, cumulative distribution function of Y is strictly monotonically increasing,

$$\text{LCB}_{\alpha^*}(X^*) = g \quad (\text{Equation 8})$$

If there is a design parameter X^{**} that results in a lower value than X^* in the $(100-\alpha^*)\%$ LCB,

$$LCB_{\alpha^*}(X^{**}) < LCB_{\alpha^*}(X^*) = g \quad (\text{Equation 9})$$

because the cumulative distribution functions of Y is strictly monotonically increasing, from Equation 9,

$$P(Y_{X^{**}} < LCB_{\alpha^*}(X^{**})) < P(Y_{X^*} < g) \quad (\text{Equation 10})$$

Again, from the definition of the LCB,

$$P(Y_{X^{**}} < LCB_{\alpha^*}(X^{**})) = \alpha^*/100 \quad (\text{Equation 11})$$

Therefore,

$$\alpha^*/100 < P(Y_{X^{**}} < g) \quad (\text{Equation 12})$$

This contradicts the assumption that the maximum value of the goal achievement probability is $\alpha^*\%$. Therefore, X^{**} cannot exist, and the solution of minimizing $(100-\alpha^*)\%$ LCB is also X^* .

Performance metrics

Goal achievement rate (GAR) in T -th optimization step (Figure 6) is defined as follows:

$$GAR(T) = \frac{\text{Number of BO runs achieving the goals within } T \text{ steps}}{\text{Total number of BO runs}} \quad (\text{Equation 13})$$

The minimum distances from the Pareto optimal solutions (Figure 7) were calculated by evaluating the distances of all the current designs explored and all approximated true Pareto solutions obtained by NSGA-II (Deb et al., 2002). Formally, the average minimum distance (AMD) from the Pareto optimal solutions in T -th step for 1,000 randomly sampled optimization trajectory is defined as follows:

$$AMD(T) = \frac{\sum_{n=1}^{1000} \text{mindist}_n(T)}{1000} \quad (\text{Equation 14})$$

$$\text{mindist}_n(T) = \min_{t=1 \dots T, m=1 \dots M} [D_2(Y_{n,t}, P_m)] \quad (\text{Equation 15})$$

where D_2 is a function mapping a pair of vectors to Euclidean distance, P_m is a vector comprising the objective values of m -th Pareto optimal solution and $Y_{n,t}$ is a vector comprising the objective values of n -th Bayesian optimization run in t -th step. M is a total number of the Pareto optimal solutions and was set to 1000. Before calculating AMD, objective values were scaled by dividing by the difference between the maximum and minimum values of each objective in the Pareto optimal solutions.

BAYESIAN OPTIMIZATION

The initial design parameters for Bayesian optimization were randomly selected. For benchmarks using mathematical toy functions, the number of the initial design parameters for each toy problem was set according to the number of design parameter dimensions plus 1. All Bayesian optimizations were performed using a GpyOpt library by implementing multi-objective functionalities, with default GpyOpt settings used unless otherwise stated. Gaussian process regression with the Matern52 kernel implemented in GPy and the normal distribution noise model were used for the machine learning model driving Bayesian optimization. Since the noise model assumed in the Gaussian process regression is the normal distribution, the probability distribution of the objective property with design parameter X , predicted by Gaussian process regression, also follows the normal distribution and acquisition functions were calculated using mean and standard deviation of this normal distribution.

CALCULATION OF THE PROBABILITY OF ACHIEVEMENT

Optimizations of the probability of achievement (PA) were performed following the logarithmic transformation. Given design parameter, X , the log of the PA for M objective properties can be obtained as follows:

$$\text{LogPA}(X) = \sum_{m=1}^M \text{Log} \left[1 - \Phi \left(\frac{\mu_m(X) - g_m}{\sigma_m(X)} \right) \right] \quad (\text{Equation 16})$$

where, g_m , μ_m and σ_m are the predefined goal, predicted mean and predicted standard deviation for the m -th objective property, respectively. Φ is the cumulative distribution function of the standard normal distribution. Note that a classical experiment navigation method for the robust product design with noisy measurements called Nakazawa method also uses a similar scoring function based on the joint probability

of goal achievement.(Inage, 2019) And Bayesian optimization with the PA can be regarded as a machine learning-based sequential implementation of this classical design method.

Calculation of the achievement function

Given the predefined goal g_m for each objective property y_m , the achievement function for M objectives is defined as follows:

$$\text{Achieve}(Y) = \sum_{m=1}^M y_m w_m \rho + \max_{m=1 \dots M} [(y_m - g_m) w_m] \quad (\text{Equation 17})$$

where w_m and ρ are predefined parameters. Note that the need for w_m arises from scale difference in objective properties. According to previous studies in the operational research field,(Hakanen and Knowles, 2017) ρ was set to 0.05 and w_m was calculated as the reciprocal number of differences between maximum and minimum values in Pareto optimal design within the design parameters explored.

Bayesian optimization with LCB acquisition, implemented in GpyOpt was used to optimize the achievement function. In GPyOpt, the LCB acquisition is implemented as Equation 4, and default value of the parameter a ($a=2$) was used.

Optimization of the acquisition function

Acquisition function optimizations were performed using the default protocol implemented in GPyOpt. In this protocol, the putative global minimum (maximum for the PA) of the acquisition functions is searched by 1000 initial random searches and subsequent optimization of the top-5 local minimum using quasi-Newton method, L-BFGS-B, implemented in SciPy.(Virtanen et al., 2020) Finally, a design parameter with the minimum searched-for value is selected for the next experiment.

Calculation of pareto optimal solutions

The multi-objective optimization methods that efficiently and thoroughly find the Pareto optimal solutions have been well studied in the field of the operations research. Among them, Non-dominated Sorting Genetic Algorithms-II (NSGA-II)(Deb et al., 2002) is one of the standard methods for optimization problems with a few objectives. In order to obtain the Pareto optimal solutions, NSGA-II implemented in Platypus library was used with sufficient number of optimization steps. The number of the obtained Pareto optimal solutions for the six mathematical benchmark problems was set to 1000, while that for the virtual inverse material design problem was set to 10000 in order to accurately evaluate experimental costs of finding Pareto optimal solutions.

Regression models for virtual material design experiment

For the virtual material design experiment, regression models constructed from experimental data were used as a substitute for time-consuming real-world experiments. The regression models can be obtained from the work of Wang et al.(Wang et al., 2020a; <https://pubs.acs.org/doi/abs/10.1021/acsami.0c11667>)

Judgment of Pareto optimal solutions

For the virtual material design experiment, average numbers of experiments before finding a first Pareto optimal solution were evaluated (Figure 9C). A set of objectives is judged as Pareto optimal when it is not Pareto-dominated by any of 10000 Pareto optimal solutions obtained by NSGA-II. Because, finding of exact Pareto optimal solutions is unnecessarily difficult, a small value δ was added to each objective value obtained by the Bayesian optimization before Judgment of Pareto optimal solutions. The small value δ for each objective was calculated as difference between the maximum and minimum values of each objective in the true Pareto optimal solutions multiplied by 0.005.

QUANTIFICATION AND STATISTICAL ANALYSIS

Performance evaluations of the Bayesian optimization methods for mathematical multi-objective toy functions were repeated 1000 times using randomly sampled settings of goals, and averages and 95% confidence intervals obtained by bootstrap resampling were reported.

Performance evaluations of the Bayesian optimization methods for the virtual material-design experiment were repeated 100 times using randomly sampled initial design parameters, and averages and 95% confidence intervals obtained by bootstrap resampling were reported.

Performance evaluations of the random sampling for the virtual material-design experiment were repeated 1000 times, and averages and 95% confidence intervals obtained by bootstrap resampling were reported.