This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

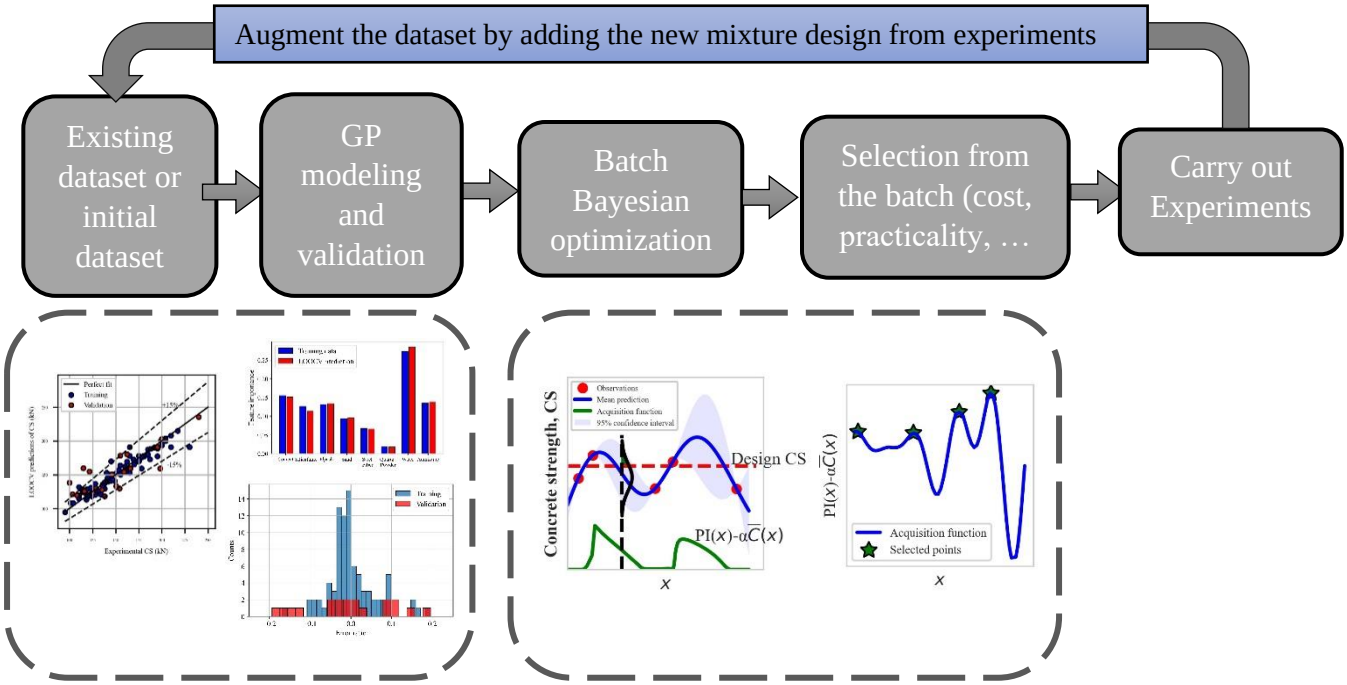
Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete

Eman Saleh[1], Ahmad Tarawneh[2], M.Z. Naser[3], M. Abedi[4], Ghassan Almasabha[5]

[1]Assistant Professor, Civil Engineering Department, Faculty of Engineering, The Hashemite University, P.O. box 3301127, Zarqa 13133, Jordan; Email: emanf_yo@hu.edu.jo
[2]Assistant Professor, Civil Engineering Department, Faculty of Engineering, The Hashemite University, P.O. box 3301127, Zarqa 13133, Jordan; Email: ahmadn@hu.edu.jo
[3]School of Civil and Environmental Engineering & Earth Sciences, Clemson University, Clemson, SC, USA
[3]AI Research Institute for Science and Engineering (AIRISE), Clemson University, Clemson, SC, USA.
E-mail: mznaser@clemson.edu, Website: www.mznaser.com
[4]M. Abedi, Ph.D. student, School of Civil and Environmental Engineering & Earth Sciences, Clemson University, Clemson, SC, USA
[5]Assistant Professor, Civil Engineering Department, Faculty of Engineering, The Hashemite University, P.O. box 3301127, Zarqa 13133, Jordan; Email: ghassans@hu.edu.jo

GRAPHICAL ABSTRACT



This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

ABSTRACT

Ultra-high-performance concrete (UHPC) has superior strength and durability, and hence it has been primarily favored in a variety of applications in structural engineering. While the open literature presents a series of predictive machine learning (ML) models to predict the strength of UHPC from its constituent materials, the reverse problem of identifying possible concrete mixtures with a targeted (pre-tailored) performance persists to exist. Unlike other works, and in an effort to bridge this knowledge gap, this study proposes a Gaussian process (GP) modeling with batch Bayesian optimization (BBO) framework (GP-BBO) to infer the mixture design of UHPC. In this framework, the GP is used as a predictive surrogate model constructed from experimental measurements. After the GP is trained and validated, BBO is used to infer the plausible formulae for the targeted strength by optimizing an acquisition function that trades off exploitation and exploration based on the optimality and variability of the surrogate model. As such, the proposed framework offers a list of possible UHPC formulae of a targeted strength. To facilitate a wide spread of the proposed framework, The ML code is shared for interested researchers to verify and expand upon. In addition, and to negate arising hurdles associated with GP-BBO programming, also an open-source and coding-free software (App) is created that can be directly deployed by UHPC fabricators. In contrast to the conventional trial-and-error-based mixture design, GP-BBO provides a self-adaptive paradigm for efficient sampling of design space for identifying the optimum sampling points. This framework can be extended to infer formulae that satisfy multiple performance objectives such as strength, workability, and durability.

Keywords: Batch Bayesian optimization, ultra-high-performance concrete, machine learning, mixture design.

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

Nomenclature

UHPC	ultra-high-performance concretes	σ_f	the output standard deviation
SCM	supplementary cementitious materials	$\boldsymbol{l}$	length scales vector of SE
HRWR	high-range water-reducing admixtures	$\delta(\boldsymbol{x}_i, \boldsymbol{x}_j)$	Kronecker delta
CS	compressive strength	σ_n	white noise standard deviation
ANN	artificial neural network	$f_X(\boldsymbol{x})$	acquisition function
SFS	sequential features selection	χ	space on input ingredients
HPC	high performance concrete	$f(\boldsymbol{x}^+)$	threshold CS value
GP	Gaussian process	$\Phi(.)$	standard CDF of normal distribution
CDF	cumulative distribution function		
BO	Bayesian optimization	$\bar{C}(\boldsymbol{x})$	normalized cost function
ML	machine learning	$\mathrm{C}(\boldsymbol{x})$	cost function
SE	square exponential	C_{min}	minimum practical cost
PI	probability of improvement	C_{max}	maximum practical cost
EI	expected improvement	α	cost-CS trade-off parameter
LCB	lower confidence bound	$\boldsymbol{D}_n$	set of the collected data
UCB	upper confidence bound	N	number of datapoints
BPI	bounded probability of improvement	$\bar{y}$	the mean of the observations
LOOCV	leave-one-out cross-validation	$\overline{y^*}$	mean of the predicted values
NRMSE	normalized root mean square error	r^2	coefficient of determination
BBO	batch Bayesian optimization	$k(x_i, x_j)$	covariance function
UCS	lower bound of the design CS	$(.)^T$	matrix transpose operator
LCS	upper bound of the design CS	$\sigma^2(\boldsymbol{x}^*)$	posterior variance of the GP
SSR	sum of the squares regression	$\mu(\boldsymbol{x}^*)$	posterior mean of the GP
SST	sum of the squares total	$m(\boldsymbol{x})$	prior mean of the GP
SE	squared exponential	ψ	set of elements in a sampled PDF probability density function batch
BBNN	back propagation neural network	ϕ	PDF of standard normal
		y	observed outcome

$f(\boldsymbol{x})$ fitted function

$\boldsymbol{\epsilon}$ observations error

$\boldsymbol{y} = \{y_i\}_{i=1}^k$ observed outcomes

$\boldsymbol{x} = \{\boldsymbol{x}_i\}_{i=1}^k$ input ingredients vector

$\boldsymbol{x}^*$ Unknown ingredients vector

y^* outcome at $\boldsymbol{x}^*$

$\boldsymbol{K}(.,.)$, $\boldsymbol{K}^*(.,.)$, and $\boldsymbol{K}^{**}(.,.)$ covariance matrix in the GP

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

1. INTRODUCTION

Ultra-high performance concrete (UHPC) characterized by its superior mechanical properties, great durability, and excellent toughness [1,2], and as a result, presents itself as an attractive solution for unique constructions, such as long-span bridges, high rise buildings, and marine structures [3,4] As widely known, the basic principle in developing UHPC is the creation of a dense particle packing structure [4,5]. Therefore, it is necessary to optimize the mixture of UHPC to obtain its excellent properties [6, 7]. Nevertheless, in much literature, the UHPC mixtures are developed without any theoretical instructions, resulting in a varying performance of UHPC [8,9,10.] Thus, to motivate the development of UHPC in construction, a more systematic and efficient mixture design method is necessary.

An efficient UHPC mixture design involves the selection of raw materials in optimum proportions to obtain concrete with desired (pre-tailored) properties (in fresh and hardened states) [1] Yet, predicting the performance of UHPC is not a simple task. The presence of different types of supplementary cementitious materials (SCMs) could change the hydration kinetics of cement [11], [12], which is one of the factors that control the strength development of UHPC. In addition, the inclusion of high-range water-reducing (HRWR) admixtures, steel fibers, and/or polyethylene fibers would trigger nonlinear effects on strength development [13]–[17] Other constitutes of UHPC such as aggregate content, water content, curing method, fine powder content (e.g., nano-silica and crushed quartzite) can also influence the performance of UHPC. As one can see, the final performance of UHPC depends on multiple factors – the alternation of each can result in significant changes in properties and performance. Therefore, producing a tailored and viable UHPC mixture is often tied with exhausting a large number of resources (i.e., acquiring constituent materials, casting different batches, performing tests, etc.) [18].

In recent years, different mixture design methods of UHPC have been developed. These methods can be divided into four categories based on their adopted design principles [19], which include methods based on the rheological properties of the paste, packing density methods that are divided into wet and dry packing, statistical design approaches, and artificial neural network (ANN)-based methods. The rheology-based mixture design employs the relationships between the mixture constitutes and the rheological properties of the paste to come up with an appropriate mixture design with target rheology and performance [19]. For example, Wang et al.[20] selected water-to-binder ratio and superplasticizers that result in the desired rheological properties of cement mortar, and controlled distribution of steel fiber in UHPC. The rheology-based mixture design requires many experiments to determine the number of constitutes to result in UHPC that meets the target rheological performance. In addition, rheological tests are very sensitive to many factors, such as environmental conditions [19]–[21].The packing density-based mixture design methods aim to develop a mixture with a maximum packing density. A higher packing density of solid constitutes a lower void ratio, thus, a lower quantity of the paste to fill the voids, and consequently,

the drying shrinkage and the rise of temperature are minimized as both are roughly correlated with paste content [22]. Several particle packing models have been used for optimizing the UHPC. For example, Larrard et al. [5,23,24] proposed the linear packing density model that was later modified to the compressive packing model (CPM). The CPM was employed by Arora et al. [25,26] to develop an eco-friendly UHPC by optimizing the aggregate and binder phases separately. Yu et al. [27,28] utilized the modified Andreasen and Andersen model for optimizing the mixture proportioning of UHPC and reported that it was efficient to develop UHPCs with excellent performance but relatively low dosage of cement by utilizing the MAA model. Most packing density methods were derived based on various assumptions that in most cases may appear unreasonable or unsuitable for the design UHPC mixture (e.g., assume all the particles are spherical); in addition, some of them rely on experimental testing without any theoretical guidance [19].

The statistical mixture design methods are used to develop objective data and decision variables by varying mixture proportions and exploring their effect on the performance. Response surface methods (RSM) and D-optimal design are the most common statistical methods used to predict the influence of mixture constitutes on UHPC performance. For example, Fan et al. [4, 8–10, 29] designed a UHPC mixture with maximum wet packing density by utilizing D-optimal design and RSM. One disadvantage of statistical-based mixture design methods is the need for a series of trial experiments to establish the necessary relationships for mixture design that are only valid for the range of the investigated parameters. ANN-based mixture design methods establish an ANN model to predict and optimize the performance of UHPC. Ghafari et al. [18] reported that ANN-based methods are more accurate and efficient than statistical methods in predicting the performance of UHPC. However, ANN methods require a large set of data to train and avoid overfitting. Another major shortcoming of all the above methods is that their optimization will only provide one or two mixture designs that are predicted to meet the targeted performances (e.g., strength, durability, workability, and economy), which may appear after testing that it cannot satisfy the predicted performances. In addition, the optimization process of these methods may not reflect the various intentions of the mixture designers. For example, the mixture designer may aim to look for the optimal global mixture that meets the target performances by exploring the space of constitute where the predictive uncertainty of performance is high.

To overcome the above, this study presents a Gaussian Process (GP) modeling with a Batch Bayesian optimization (BBO) (GP-BBO) framework for mixture design to achieve a probabilistic-based mixture design of UHPC with targeted compressive strength (CS). GP modeling with the Bayesian optimization (BO) framework has been used in many fields, such as material design and discovery [30], robotic applications [31], and experimental design of ML algorithms [32]. However, up to our knowledge, it has never been applied for the selection of optimal ingredients of a mixture, such as UHPC. GP-BBO, as opposed to the above methods, can provide a list of mixture designs of UHPC that one can choose from based on the satisfaction of the various performance requirements. In case, after testing, the selected mixture failed to meet the desired

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

performances, test results are still useful for updating the GP-BBO model, then another optimized list of mixtures can be obtained. Therefore, the proposed GP-BBO-based mixture design does not only provide a systematic method for the selection of raw materials in optimum proportions for the development of UHPC, but it also provides a more realistic and practical method for the selection process. In addition, where the points (i.e., mixture formulae) are sampled is based on an acquisition function that reflects the actual intentions of the mixture designer.

To fully describe the proposed GP-BBO-based mixture method, the mathematical background of the GP, BO, and BBO with the proposed framework is presented in Section 2. In Section 3, a description of the input data that are used to train and validate the GP model is introduced. The validation of the GP model is described in Section 4. Section 5 introduces the analysis results and inferred formulae of the UHPC mixture. Prospects for future extension of the proposed framework are then presented in Section 8.

2. MATHEMATICAL BACKGROUND AND PROPOSED GP-BBO FRAMEWORK FOR MIXTURE DESIGN

The GP is a supervised non-parametric ML method that can provide a prediction of an unknown response variable based on prior collected data. Unlike parametric regression such as least-square regression, GP can provide a more rigorous method in dealing with noisy and complex data [33] and is normally used to provide surrogate models for complex computational ML algorithms [34]–[37].

The output of GP is a posterior probability of the response variable with the input parameters used as a prior. Integrating GP with Batch Bayesian optimization (BBO) allows for identifying the optimum sampling points that represent here a set of plausible mixture designs. In particular, a prior model, a GP, of the objective is constructed from the collected experimental data and then sequentially refined as more data are obtained through optimizing an acquisition function. The acquisition function guides the selection of the optimal ingredients for the mixture and the exploration process of new mixtures design. In addition to providing multiple mixture designs for target performance, the GP-BBO framework offers a probabilistic description of each mixture design (e.g., the probability of attaining the target compressive strength); thus, the risk of any design can be quantified beforehand of the experiments; and hence reduces the need for experimental trails.

A general relationship between the inputs and outcomes can be expressed in a GP-BBO framework as can be seen in Eq. (1):

$$y = f(\boldsymbol{x}) + \boldsymbol{\epsilon} \tag{1}$$

Where y are the observed outcomes that are associated with the inputs $\boldsymbol{x}$, f is a process that describes the relationship between the inputs and outcomes, and $\boldsymbol{\epsilon}$ represents the error between the described outcome(s) ($f(\boldsymbol{x})$) and the observed outcome(s) (y). Usually, the actual expression of

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

f is unknown; however, it is possible to come up with a close expression that mimics the behavior of f using the observed data through an ML surrogate. A brief description of the GP is presented here; however, for further details about GP, the reader may refer to [38].

2.1.Gaussian process (GP)

GP is designed to estimate the most plausible outcome y^* at a new input $\boldsymbol{x}^*$ with the knowledge of the observations $\boldsymbol{y} = \{y_i\}_{i=1}^{k}$ and observational error $\boldsymbol{\epsilon} \in \mathcal{N}(0, {\sigma_k}^2)$ at inputs $\boldsymbol{x} = \{\boldsymbol{x}_i\}_{i=1}^{k}$, where $\boldsymbol{x}_i \in \mathbf{R}^{\boldsymbol{D}}$. This can be expressed as

$$\underset{\boldsymbol{x} \in \chi}{\operatorname{argmax}}\, P(y^*|\boldsymbol{y}) \tag{2}$$

Following Bayes' rule, $P(y^*|\boldsymbol{y})$ can be written as

$$P(y^*|\boldsymbol{y}) = \frac{\mathrm{P}(\boldsymbol{y}|y^*)\mathrm{P}(y^*)}{\mathrm{P}(\boldsymbol{y})} \tag{3}$$

In GP, the collection of random variables is assumed to follow a multivariate Gaussian distribution [38]; thus, $\mathrm{P}(\boldsymbol{y})$ can be written as

$$P(\boldsymbol{y}) = \mathcal{N}(m(\boldsymbol{x}), \boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x})) \tag{4}$$

And the joint probability $\mathrm{P}(\boldsymbol{y}, y^*)$ of the current observations $\boldsymbol{y}$ and the most recent one y^* can be written as

$$P(\boldsymbol{y}, y^*) = \mathcal{N}\left(\begin{bmatrix} m(\boldsymbol{x}) \\ m(\boldsymbol{x}^*) \end{bmatrix}, \begin{bmatrix} \boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x}) & \boldsymbol{K}^*(\boldsymbol{x}^*, \boldsymbol{x})^T \\ \boldsymbol{K}^*(\boldsymbol{x}^*, \boldsymbol{x}) & \boldsymbol{K}^{**}(\boldsymbol{x}^*, \boldsymbol{x}^*) \end{bmatrix}\right) \tag{5}$$

Further, the conditional distribution $P(y^*|\boldsymbol{y})$ is given as

$$P(y^*|\boldsymbol{y}) = \mathcal{N}\big(m(\boldsymbol{x}^*) + \boldsymbol{K}^*\boldsymbol{K}^{-1}\big(\boldsymbol{y} - m(\boldsymbol{x})\big), \boldsymbol{K}^{**} - \boldsymbol{K}^*\boldsymbol{K}^{-1}(\boldsymbol{K}^*)^T\big) \tag{6}$$

Where $\boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x})$, $\boldsymbol{K}^*(\boldsymbol{x}^*, \boldsymbol{x})$, and $\boldsymbol{K}^{**}(\boldsymbol{x}^*, \boldsymbol{x}^*)$ are covariance matrices that are estimated using the covariance function $k(.,.)$, which provides a measure of how variables change together. $P(y^*|\boldsymbol{y})$ is known as GP, with posterior mean and variance defined by

$$\mu(\boldsymbol{x}^*) = m(\boldsymbol{x}^*) + \boldsymbol{K}^*\boldsymbol{K}^{-1}\big(\boldsymbol{y} - m(\boldsymbol{x})\big) \tag{7}$$

$$\sigma^2(\boldsymbol{x}^*) = \boldsymbol{K}^{**} - \boldsymbol{K}^*\boldsymbol{K}^{-1}(\boldsymbol{K}^*)^T \tag{8}$$

The accuracy of the GP model depends on this covariance function and its hyperparameters. Various covariance functions have been proposed in the literature (see [38]). Here, the square exponential (SE) covariance function is selected for its excellent performance in a variety of problems, as noted in [39]. This function is expressed as

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

$$k(x_i, x_j) = \sigma_f^2 e^{[-(x_i - x_j)/\boldsymbol{l}^2]} + \sigma_n^2 \delta(\boldsymbol{x}_i, \boldsymbol{x}_j) \tag{9}$$

The hyperparameters of this covariance function are $\sigma_f, \boldsymbol{l}$, and σ_n represent the output standard deviation, length scale vector, and noise level, respectively. The output variance specifies the average distance of the fitted function away from its mean, while the length scale vector determines the amount of smoothness in the function. Smaller $\boldsymbol{l}$ corresponds to a wigglier function, while larger $\boldsymbol{l}$ means a smoother function. The noise level determines the amount of noise expected in the outcomes.

2.2.Bayesian optimization (BO)

As the GP model is trained and validated, it becomes possible to infer the plausible ingredients of UHPC with targeted compressive strength using BO. BO is a powerful approach for finding the global optimum of black-box acquisition functions and has proven successful in experimental design [40] The standard BO involves two main steps [37]: (1) estimating a probabilistic surrogate model, usually a GP, trained via experimental data; (2) optimizing an acquisition function that trades off exploitation and exploration based on the optimality and variability of the surrogate model. Typically, the high acquisition is associated with potentially optimum values of the objective function, whether because the predicted value is high, the uncertainty is high, or both. In our setting, we consider the global maximization problem

$$\boldsymbol{x}^* = \text{argmax}\, f_X(\boldsymbol{x}), \boldsymbol{x} \in \chi \tag{10}$$

Where χ is the input space of the ingredients, f_X is the acquisition function (or objective function), and $\boldsymbol{x}^*$is the vector of optimum ingredients that maximize this acquisition function based on inferences from the fitted GP. The choice of an acquisition function is a key driver of the trade-off between exploration and exploitation. *Exploitation* represents the case of choosing the best ingredients given the current information, while *exploration* prefers the choice with uncertain values to gather more information and update the current knowledge, which could result in making the best overall choice in the future. The best long-term plan involves the trade-off between both.

Four commonly used acquisition functions in the experimental design [41] are the probability of improvement (PI), expected improvement (EI), lower confidence bound (LCB), and upper confidence bound (UCB)[1].

PI selects the points with a high probability of being greater than $f(\boldsymbol{x}^+)$. PI is a pure exploitation formulation in which it favors the points that most likely exceed a threshold of $f(\boldsymbol{x}^+)$ over points that offer large values of the objective but with less certainty such that:

[1] A detailed description of these acquisition functions can be found in [42].

Please cite this paper as:

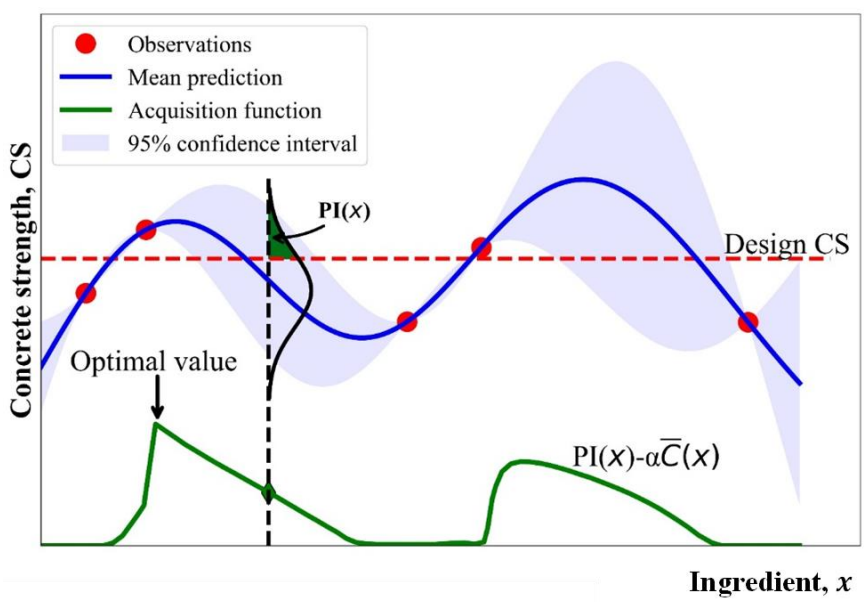
Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

$$\mathrm{PI}(\boldsymbol{x}) = P(f(\boldsymbol{x}) \geq f(\boldsymbol{x}^{+})) = \Phi\left(\frac{\mu(\boldsymbol{x})-f(\boldsymbol{x}^{+})}{\sigma(\boldsymbol{x})}\right) \tag{11}$$

To make this acquisition function fits with mixture design objectives, PI is modified by the inclusion of the cost of the UHPC mixture to force the BO to select the points that correspond to the combination of ingredients that yield the target strength or higher with a minimum mixture cost. The cost function, $\mathrm{C}(\boldsymbol{x})$, should be normalized such that it has a common scale with $P(f(\boldsymbol{x}) \geq f(\boldsymbol{x}^{+}))$; thus, both be weighted equally in the optimization problem. Normalization of cost can be done as follow

$$\bar{C}(\boldsymbol{x}) = \frac{\mathrm{C}(\boldsymbol{x})-C_{min}}{C_{max}-C_{min}} \tag{12}$$

Where C_{min} and C_{max} are the minimum and maximum practical cost of a mixture with a certain target strength, respectively. These two values can be set based on the cost of different tested mixtures from the collected dataset by subtracting and adding some marginal cost from the estimated minimum and maximum cost, respectively. Graphic descriptions of BO using this acquisition function for a 1D case (one feature or ingredient) are illustrated in Figure 1. As it can be seen from Figure 1, the threshold $f(\boldsymbol{x}^{+})$ represents the design compressive strength, and the probability of improvement is evaluated as the probability of exceeding this value where each point is assumed to follow a normal distribution (GP assumption). Further, α in the acquisition function corresponds to the trade-off parameter to balance higher strength and lower cost objectives. Assuming a linear cost function, the acquisition function $f_X(\boldsymbol{x}) = \mathrm{PI}(\boldsymbol{x})$- $\alpha\bar{C}(\boldsymbol{x})$ is computationally tractable and can be optimized without much computational cost.



Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

Figure 1. GP model optimization using $\mathrm{PI}(\boldsymbol{x})$- $\alpha\bar{C}(\boldsymbol{x})$ as the acquisition function

Another acquisition function used in mixture design, often considered an extension of PI acquisition formulation, is the bounded probability of improvement (BPI). This acquisition function selects the combinations of ingredients with the highest probability of obtaining a compressive strength within certain compressive strength bounds. The cost function can be omitted from this acquisition function because the points with much higher compressive strength than the targeted bounds, which are usually associated with a high cost, will not be selected by the BO with this acquisition function. BPI can be defined as

$$\mathrm{BPI}(\boldsymbol{x}) = P(\mathrm{LCS} \leq f(\boldsymbol{x}) \leq \mathrm{UCS})) = \Phi\left(\frac{\mu(\boldsymbol{x})-\mathrm{LCS}}{\sigma(\boldsymbol{x})}\right) - \Phi\left(\frac{\mu(\boldsymbol{x})-\mathrm{UCS}}{\sigma(\boldsymbol{x})}\right) \tag{13}$$

Where LCS and UCS are the upper and lower bound of the design compressive strength, respectively. Graphic descriptions of BO using BPI as the acquisition function for a 1D case (one feature or ingredient) is illustrated in Figure 2.

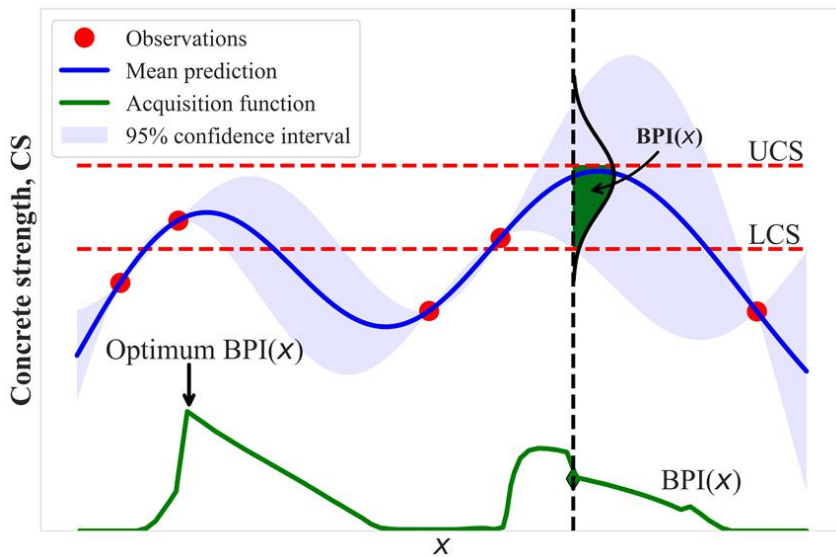


Figure 2. GP model optimization using $\mathrm{BPI}(\boldsymbol{x})$ as the acquisition function

An additional acquisition function, EI, trades off between exploration and exploitation as shown in Figure 3: first, a GP is fitted to the observations to obtain the mean prediction $\mu(x)$ as shown in the blue curve. Instead of selecting the points that maximize $\mu(x)$, EI considers the model uncertainties; mathematically, EI evaluates the possible improvement of the design compressive strength (i.e., possible improvement in $f(\boldsymbol{x}^+)$) by incorporating both $\mu(x)$ and the corresponding uncertainty $\sigma(x)$:

$$\mathrm{EI}(\boldsymbol{x}) = \mathbb{E}[\max(\mu(x) - f(\boldsymbol{x}^+), 0)]$$

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

$$\mathrm{EI}(\boldsymbol{x}) = \left(\mu(x) - f(\boldsymbol{x}^{+})\right)\Phi\left(\frac{\mu(x)-f(\boldsymbol{x}^{+})}{\sigma}\right) + \sigma\phi\left(\frac{\mu(x)-f(\boldsymbol{x}^{+})}{\sigma}\right) \tag{14}$$

Where Φ and ϕ are the cumulative distribution function (CDF) and probability density function (PDF) of the standard normal distribution, respectively.

EI is the most widely used acquisition function in many fields, and it is known to work well with deterministic functions (low uncertainties) [42], while for noisy functions, some modifications to the original formulation are required as in [43] A cost function is recommended to be added to EI to balance the higher strength and lower cost objectives.

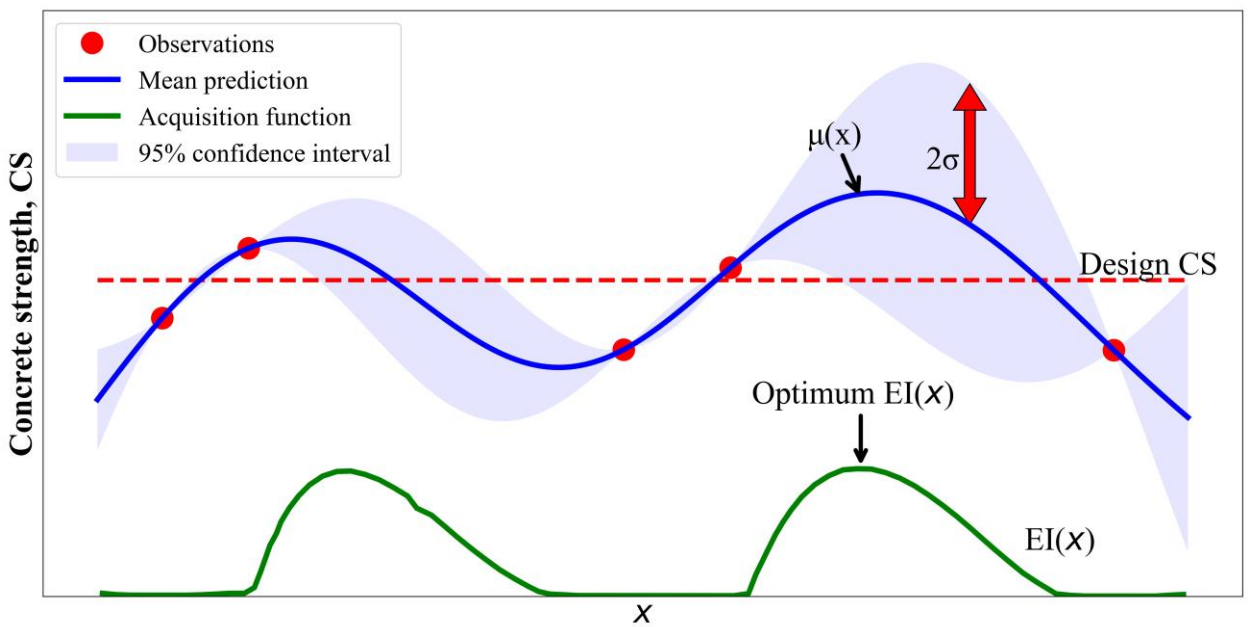


Figure 3. GP model optimization using $\mathrm{EI}(\boldsymbol{x})$ as the acquisition function

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

If no risk is desired to be taken in the mixture design process, the LCB as the acquisition function can be used; this acquisition function picks the combination of ingredients that satisfies the target strength at the lower bound of the fitted GP as shown in Figure 4.

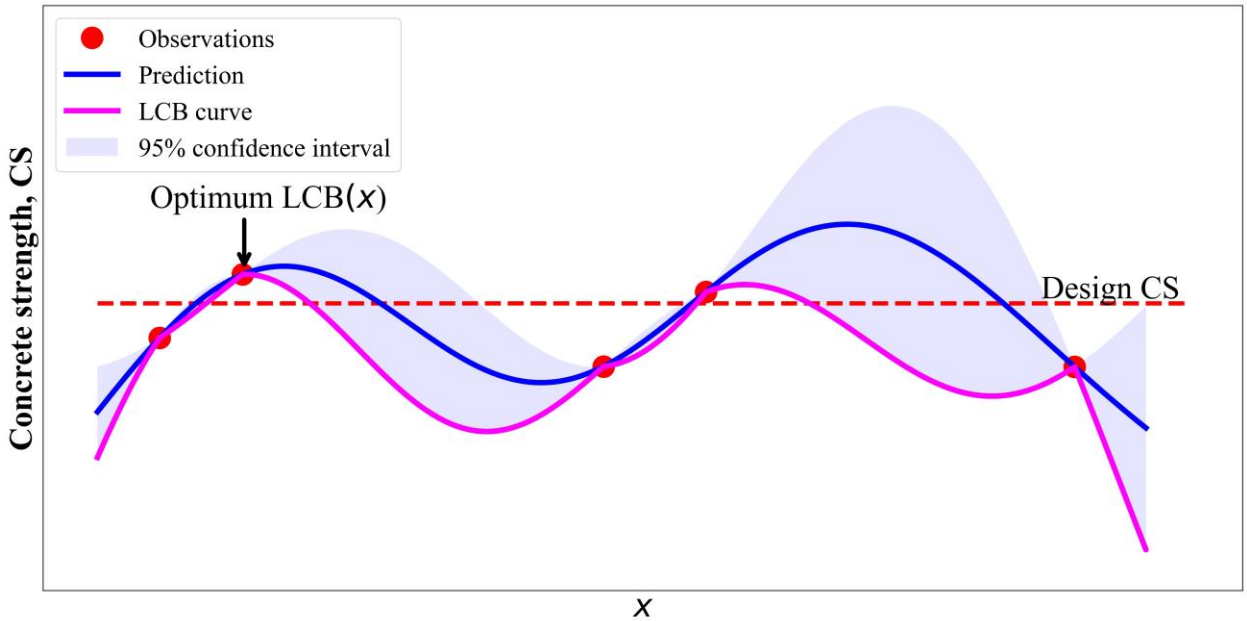
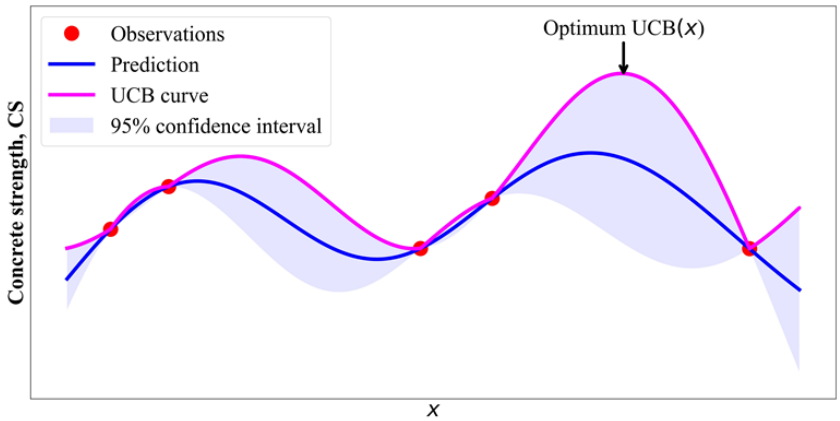


Figure 4. GP model optimization using LCB($\boldsymbol{x}$) as the acquisition function

If the goal of the mixture design is to explore different mixes that can provide the highest strength possible, then UCB as an acquisition function (also called no regret formulation) can be used in the GP-BO framework. UCB selects the combination of ingredients (or mixture constitutes) that is most plausible to provide the highest compressive strength possible. Figure 5 illustrates a graphical description of this acquisition function for a 1D case (1 feature space).



This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

Figure 5. GP model optimization using $\mathrm{UCB}(\boldsymbol{x})$ as the acquisition function

2.3.Batch Bayesian Optimization

BO offers a probabilistically targeted method for global optimization; thus, it will only provide a single sampling point (i.e., one combination of ingredients or formula). However, the sampled point may be unrealistic; for example, it may have a very low water-to-binder ratio or give much higher performance than the targeted one with a very high cost. Hence, it should be removed from the plausible mix designs. To solve this, multiple points can be sampled simultaneously using a process known as Batch Bayesian optimization (BBO) [44]; then, an assessment process of each point can be conducted to select the most appropriate point. In BBO, the goal is to sample a batch $\psi = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_k\}$ of k promising samples to estimate in parallel, as shown in Figure 6. The maximization-penalization strategy is used here to select the elements of the batch. Maximization-penalization sequentially applies a local penalizer around the most recent maxima (i.e., previous sampled batch element) to allow the BO to choose a different optimum. For further information on this strategy, the reader may consult [44].

An alternative to BBO is to sequentially change the trade-off parameters (e.g., α) in the acquisition function. This sequential maximization of the acquisition functions will result in multiple samples that could be assessed for practicality. Here, BBO with six elements in a batch will be adopted.

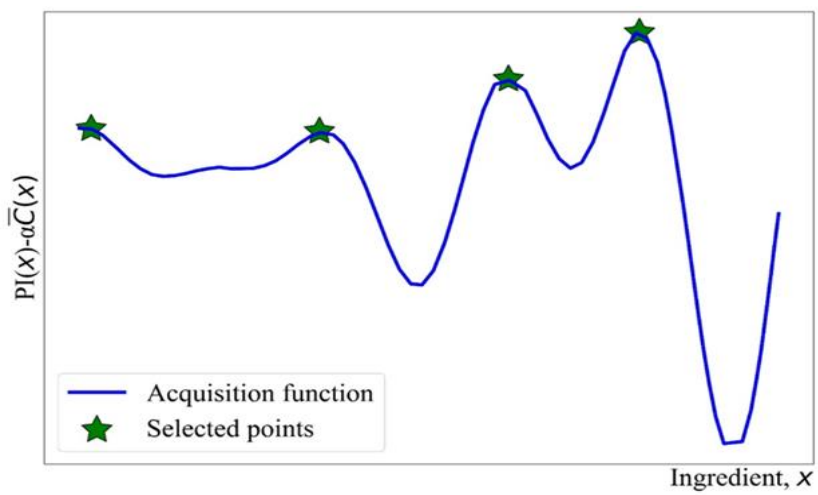


Figure 6. Graphical description of BBO for 1D case

2.4.GP-BBO framework for mixture design

Our proposed GP-BBO framework for data-driven mixture design consists of four major steps (as shown in Figure 7 and Table 1): (1) Step 1 involves developing a mixtures dataset based on the data collected from literature and lab experiments, (2) Step 2 fits a GP model using the collected dataset, and provide a quantification of the uncertainty involved in the model prediction based on

the available data, (3) Step 3 produces inference about where to take samples based on the acquisition function the trades off between exploitation (where the response appear to be optimal) and exploration (where uncertainty is high), and (4) Step 4 evaluates the sampled points (e.g., verify that the design compressive strength can be obtained experimentally). The evaluated points are used to augment the dataset and update the GP model. As the process continues, more data points are sequentially added to update the GP model and identify the global optimal solution.

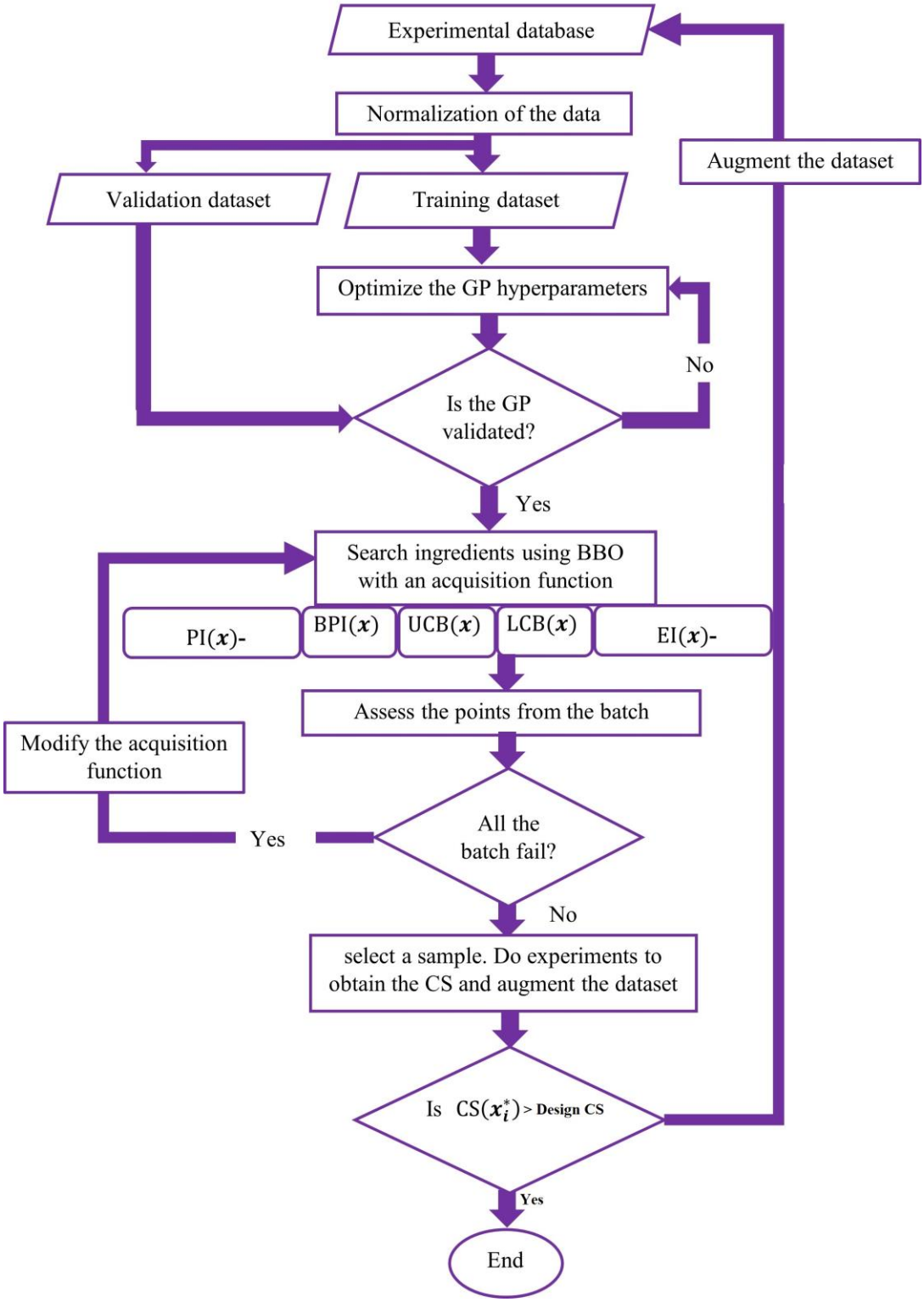
Two essential components distinguish the GP-BBO framework from the other mixture design methods, a model that provides predictions with quantification of the predictions' uncertainties, and a criterion that specifies where to take samples. The GP model provides a robust approximation of the response value (e.g., compressive strength) with multiple input variables (e.g., the proportion of cement, water, admixture…etc.) while BBO determines where to sample next based on inferences from the fitted GP model, by a measure of the quantity of information gained from sampling a certain point, called acquisition functions. Five commonly used acquisition functions that were previously described can be utilized to infer where to sample points (i.e., mixture formulas) are PI, BPI, EI, LCB, and UCB. The choice of the best acquisition function is related to the intentions of the mixture designer, as will be elaborated on later.

Until now, we have considered all the necessary details of the proposed GP-BBO mixture design framework and summarized it in Figure 7 and Algorithm 1. For the collected dataset of UHPC mixtures D_n, GP model is fitted and validated, then a list of points (i.e., UHPC mixture formulae) are sampled by optimizing an acquisition function $f_{X_i}(\boldsymbol{x})$ using BBO. After that, the sampled list (or batch) is assessed for the satisfaction of different performance requirements (e.g., packing density, workability, …, etc.) to select a point (or formula) from the sampled list for testing. If the performed experiments verified that the selected mixture could meet the targeted performances, the mixture design process is terminated, and the experimental values are used to augment the current dataset, otherwise, after augmentation of the dataset, a new fitting of the GP should be performed, and a new batch is sampled.

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270



This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

Figure 7 Flow chart of the GP-BBO framework

Table 1. GP-BBO algorithm

Step	Description
(0) (1)	Collect initial dataset $\boldsymbol{D}_n$ fit a GP to $\boldsymbol{D}_n$ and validate
(2)	Select the points (formulae) $\boldsymbol{x}_i^*$ using BBO $\boldsymbol{x}_i^* = \underset{\boldsymbol{x} \in \chi}{\operatorname{argmax}}\, f_{X_i}(\boldsymbol{x})$
(3)	Assessment of different formulae $\mu_{\boldsymbol{x}^*} \geq CS$, cost, feasibility, …etc. if all $\mu_{\boldsymbol{x}_i^*} \leq$ Design CS then, modify $f_{X_i}(\boldsymbol{x})$ Go to step (2) else Select a formula $\boldsymbol{x}_i^*$
(4)	Do experiments to obtain $\mathrm{CS}(\boldsymbol{x}_i^*)$
(5)	add to data $\boldsymbol{D}_{n+1} = \{\boldsymbol{D}_n, (\boldsymbol{x}_i^*, \mathrm{CS}(\boldsymbol{x}_i^*)\}$
(6)	if $\mathrm{CS}(\boldsymbol{x}_i^*) <$ Design CS then, Go to step (1) and fit a GP to $\boldsymbol{D}_{n+1}$
(7)	**End**

3. DESCRIPTION OF DATASET

The components of UHPC considered in the proposed GP model are cement, water, silica fume, fly ash, sand, steel fiber, Quartz Powder, and admixture – and more components can be added as well. The database used in this study is shown in Table 1A, in a total of 110 points pairing formulae with their associated compressive strength (see Table 1A). Curing age is not included in the GP model, and only the data corresponding to the 28 days of compressive strength are considered in this study. The proposed framework for mixture design is targeted to guide mixture design in practice, where 28 days strength is the reference strength used in design codes [45] For the analysis, each ingredient of UHPC is normalized by the density of the concrete, which represents the total weight of all ingredients (kg/m^3) to allow the resulting model to infer a mixture design with a predefined density. The compressive strength is normalized using the maximum compressive strength in the training dataset. Normalization is beneficial when the data has varying scales as it brings all the values to a common scale, thus, allowing for modeling the data correctly

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

[33]. Table 2 shows the ranges of input parameters for training and validation datasets, and Figure 8 shows the histograms of each variable in the dataset. It is worth noting that the presented dataset comprises realistic mixtures that can be deployed in the field.

Table 2. Ranges of input parameters and their statistical measurements

Ingredient (kg/m^3)	Mean	Standard deviation	Minimum	Maximum
Cement	879.7	329.8	383	1600
Water	197.1	54.3	0	185
Sand	980.0	513.8	0	1898
Admixture	31.9	28.2	0	185
Quartz powder	750	36.9	0	750
Steel fiber	39.0	74.8	0	470
Silica fume	192.0	94.6	0	367.95
Fly ash	33.0	72.7	0	448

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

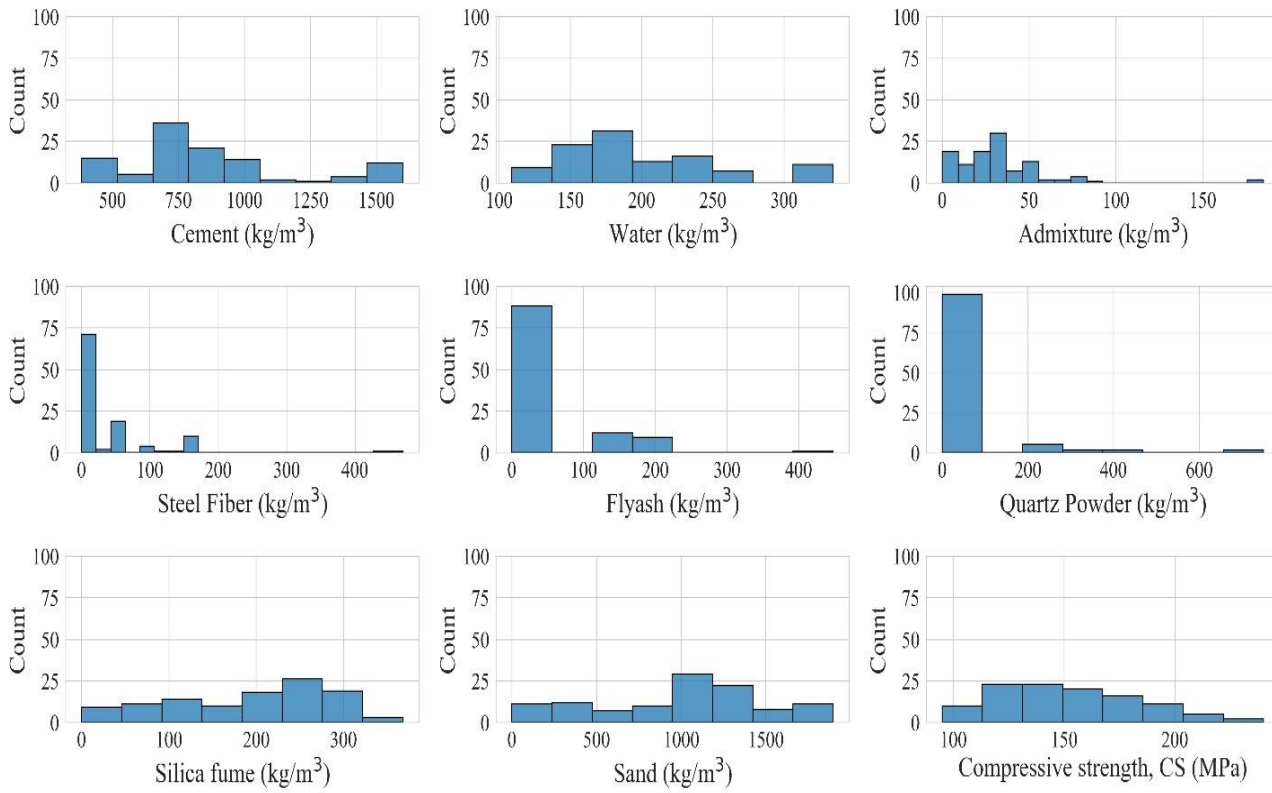


Figure 8 Frequency distributions of the parameters in the experimental dataset

4. GP FITTING AND VALIDATION

In this study, 80% of the data were randomly assigned to the training dataset, while the remaining 20% were assigned to the validation and testing dataset. Hyperparameters optimization was carried out with different initialized search values to obtain the most robust GP fit. Leave-one-out cross-validation (LOOCV) was used to learn the optimal hyperparameters of the GP fit. LOOCV is a cross-validation approach in which each training data point is assigned to the test set while the rest (N-1) data points are assigned to the training dataset. In LOOCV, the model is fitted using the N-1 observations then the fitted model is used to predict the one data point in the test set. This process is repeated N times for each data point as the test dataset. Thus, the model fit that shows the best performance among all the test datasets can be found. The quality of the fitting is measured using the following metrics:

- Normalized mean square error (NMSE):

$$\mathrm{NMSE} = \frac{1/N \sum_{i=1}^{N} (y_i - \overline{y_i^*})^2}{1/N \sum_{i=1}^{N} y_i} \tag{15}$$

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

- Coefficient of correlation (R^2):

$$R^2 = \frac{SSR}{SST} \tag{16a}$$

$$SSR = \sum_{i=1}^{N} (\overline{y_i} - \overline{y_i^*})^2 \tag{16b}$$

$$SST = \sum_{i=1}^{N} (y_i - \overline{y_i})^2 \tag{16c}$$

- Error ratio of each data point i:

$$\text{Error ratio}_i = \frac{\text{LOOCV prediction}_i - \text{observed value}_i}{\text{observed value}_i} \tag{17}$$

Where $\overline{y_i}$ is the mean of the observed compressive strength, $\overline{y_i^*}$ is the fit value at $\boldsymbol{x}_i$, and y_i is the observed compressive strength. Further validation is to evaluate these metrics for the validation data set (20% of the data) and check if the fitted GP can predict the real behavior of the data.

Figure 9 shows that most LOOCV predictions are within the $\pm$15% range of the experimental CS data. The Normalized root mean square error (NRMSE) and R^2 are 0.06 and 0.93, respectively. From the histogram of error ratio illustrated in Figure 10, it is concluded that all of the LOOCV predictions are within $\pm$20% of the training data. This indicates that the GP and the experimental data have similar statistical properties.

To ensure that the GP model can represent the actual behavior in the real dataset, an additional comparison of the feature importance derived from the LOOCV predictions and training dataset was performed, as shown in Figure 11. Feature importance represents a score assigned to input features based on their contribution to predicting the target value, here the compressive strength value. Based on Figure 11, it is clear that there is a close alignment between the feature importance derived using training data and LOOCV predictions; thus, it is reasonable to assume that the GP fit can represent the physics of the input features controlling the development of compressive strength in UHPC. Therefore, it is reasonable to use the GP to investigate the statistical features of the real process.

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

For further validation, the performance was checked against a separate validation set which represents 20% of the total dataset. The results are shown in Figure 9 and Figure 10. As expected, the performance of the validation set is very similar to the LOOCV predictions. The quantity-to-quantity plot in Figure 11 shows that the validation data are distributed on both sides of the perfect fit line, which indicates that the GP fit does not have a preference toward overestimation or underestimation of the prediction of the validation data. In addition, Figure 11 and Figure 12 demonstrate that most of the validation data landed within the 20% error range. The NRMSE and R^2 are 0.08 and 0.86, respectively. Therefore, it is fair to argue that the GP fit can provide sufficiently accurate predictions.

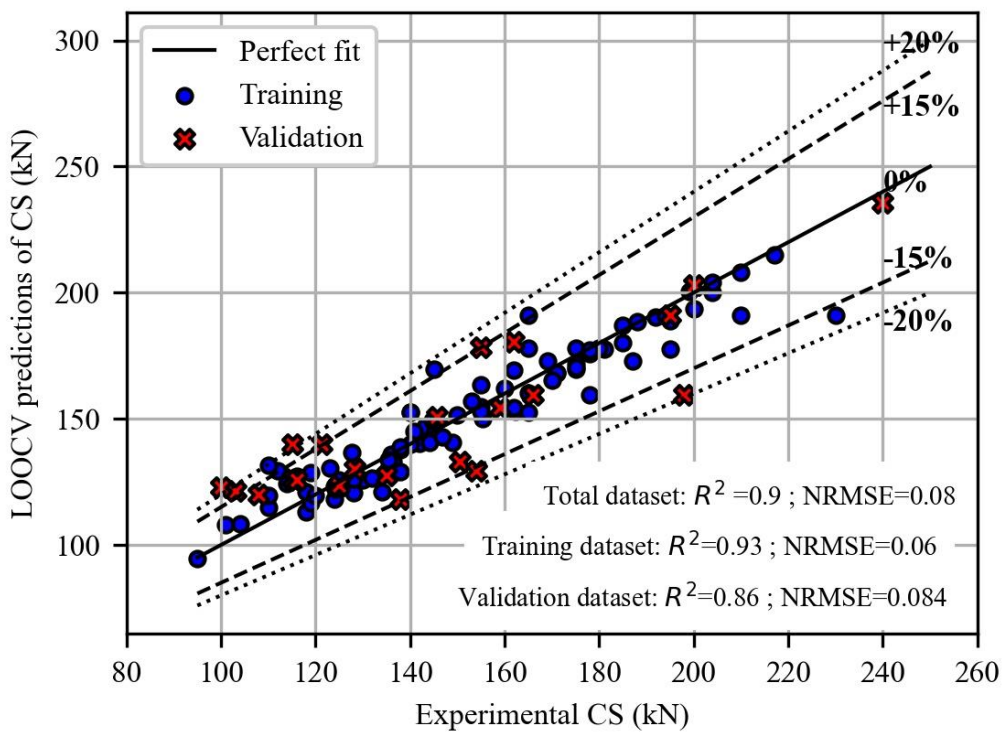


Figure 9 Percent deviation of LOOCV predicted versus experimental compressive strength.

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

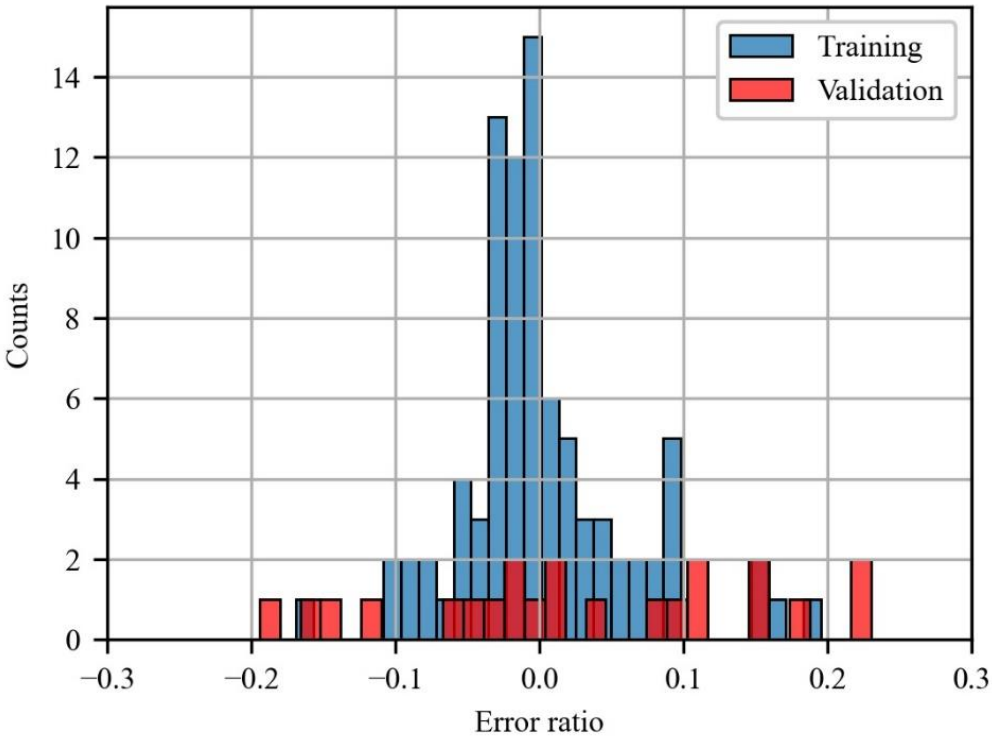


Figure 10 Histograms of error ratio of training and validation dataset

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

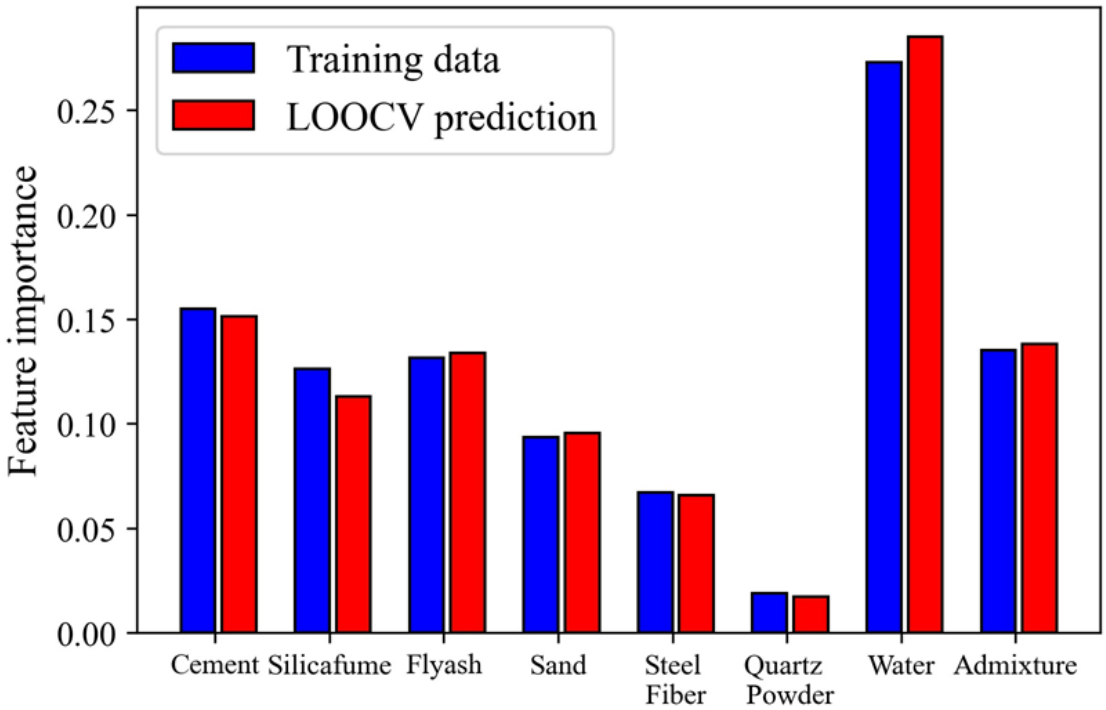


Figure 11 Features importance of ingredients to compressive strength obtained using training data and LOOCV predictions.

A Comparison of the prediction accuracy of the trained GP fit in this study with other predictive models from literature using the same UHPC dataset is shown in Table 3. Table 3 demonstrates two performance metrics: NMSE and R^2 of the total dataset. Firstly, the GP fit provides the least NMSE and the greatest R^2; this indicates that the GP model has the best performance among the other listed predictive models.

Table 3. Comparison of the prediction accuracy for UHPC using different predictive models

Predictive model	NMSE	R^2
Artificial Neural Network (ANN) without features selection [46]	0.035	0.215
ANN with features selection [46]	0.012	0.801
Nonlinear regression [46]	0.0645	0.716
Linear regression analysis	0.0209	0.07
Random forest	0.0321	0.865
XGBoost	0.012	0.876
Genetic algorithm-ANN (GA-ANN)	0.011	0.891

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

GP	0.0064	0.900

5. INFERENCE OF MIXTURE DESIGN

In mixture design, the acquisition function that favors exploitation to exploration could be recommended as it favors the ingredients with the best performance according to the current knowledge until proven otherwise (more data updates the current knowledge); thus, no, or small risk is associated with this selection. Therefore, if the main objective is to find the optimal ingredients with minimal risk involved in the selection, three acquisition functions that favor exploitation are recommended; (1) PI($\boldsymbol{x}$)- $\alpha\bar{C}(\boldsymbol{x})$, (2)BPI($\boldsymbol{x}$), and (3) LCB. If the goal of the mixture design is to explore the possibility of improving the compressive strength with some risks allowed to be taken to achieve this goal, then EI($\boldsymbol{x}$)- $\alpha\bar{C}(\boldsymbol{x})$ is more appropriate as the acquisition function in the BBO. UCB favors the high uncertainty regions; thus, it may be used to guide the process of performing experiments to augment the current database.

As an example, the proposed GP-BBO framework illustrated in Figure 7 is used to infer the plausible mixture designs of UHPC for a target strength of 150 MPa using different acquisition functions. The sampled batch was chosen to contain six inferred formulae. The results of this analysis are shown in Figure 12. This figure illustrates the first two most plausible formulae (i.e., the first two samples resulted from BBO) for a target strength of 150 MPa, assuming the unit cost of ingredients as defined in Table 4 (as suggested by [47]), C_{max} =21095.6 and C_{min} =1676.8, $\alpha = 1$, and LCS and UCS are $\pm 10\%$ of the design CS (i.e., LCS =135 MPa and UCS =165 MPa). C_{max} and C_{min} were estimated as the maximum and minimum cost of all the collected mixtures in Table 1A, respectively as follows:

$$C_{max} = \max\left(\sum_{i=1}^{N} \text{ingredient}_{\,i} \times \text{unit cost}_{\,i}\right) \tag{18}$$

$$C_{min} = \min\left(\sum_{i=1}^{N} \text{ingredient}_{\,i} \times \text{unit cost}_{\,i}\right) \tag{19}$$

Where N is the total number of ingredients that compose the UHPC mixture. The max and min functions allow finding the highest and lowest cost values in the database illustrated in Table 1A, respectively.

Table 4. The unit cost of different ingredients used in this research

Ingredient	Unit cost
Cement	2.25
Water	0.01

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

Fly ash	0.6
Silica fume	9.1
Admixtures	25.1
Sand	0.28
Quartz powder	10.5
Steel fiber	32.66

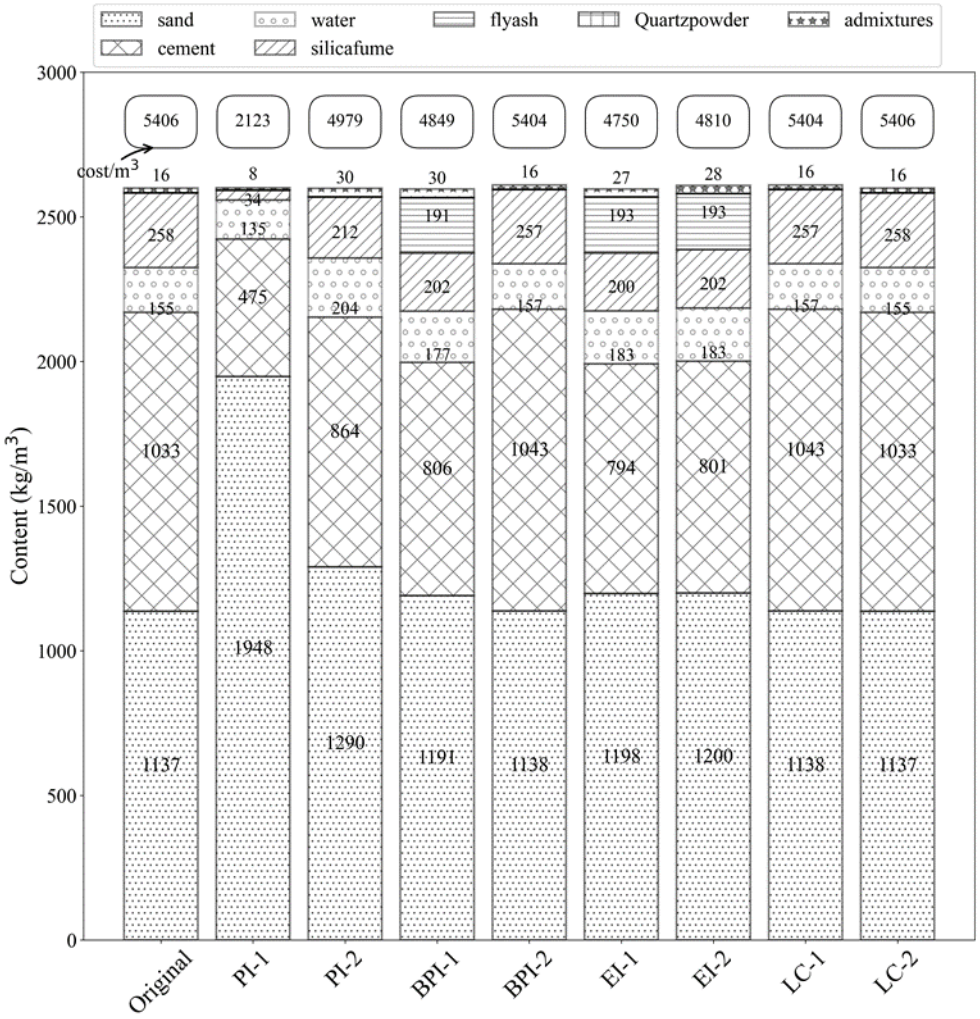


Figure 12. The first two plausible formulae of UHPC for a target strength of 150 MPa were obtained using different acquisition functions. The density of all the mixtures equals 2600 kg/m^3

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

In Figure 12, PI-1 and PI-2 represent the first two mixtures selected by the BBO using PI($\boldsymbol{x}$)-$\alpha\bar{C}(\boldsymbol{x})$ as the acquisition function, BPI-1 and BPI-2 using the BPI ($\boldsymbol{x}$) as the acquisition function, EI-1 and EI-2 using EI($\boldsymbol{x}$)- $\alpha\bar{C}(\boldsymbol{x})$ as the acquisition function, and LC-1 and LC-2 using LCB as the acquisition function while the original formula represents an experimental data point from the collected database with a compressive strength of 150 MPa that is demonstrated for comparison. As it can be observed from Figure 12, the first mixture resulting from the GP-BBO framework tends to favor the mix with a lower cost compared to the original mixture. This can be explained by the high α value (α =1) used in the framework that gives the same weight to the strength and cost objective.

Changing the α value will result in a new mixtures design that may be of more interest to the investigators/UHPC fabricators. For example, in Figure 12, it can be observed that all the formulae did not include steel fibers because its unit cost is the highest among the other ingredients (see Table 4); in addition, based on the analysis of the current dataset, it was observed that steel fibers do not have a dominant effect on the strength of UHPC (see Figure 11). However, the inclusion of steel fibers has been shown to enhance the flexural strength and other service performance objectives [48] To obtain some formulae that suggest the steel fibers as a constitute, one option is to decrease the value of α parameter to allow the optimization process to give more weight to optimizing the compressive strength than the cost objective.

To enable interested users to easily utilize our developed approach, the GP-BBO framework discussed so far is implemented into the software. The interface of this software is shown in Figure 13. This figure shows two components of this software. The first (top) component predicts the compressive strength of a given UHPC mixture based on the mixture proportion. On the other hand, the second component (bottom) suggests possible mixture candidates that can attain the same compressive strength reached by the original mixture. This software can be downloaded from the authors' websites.

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

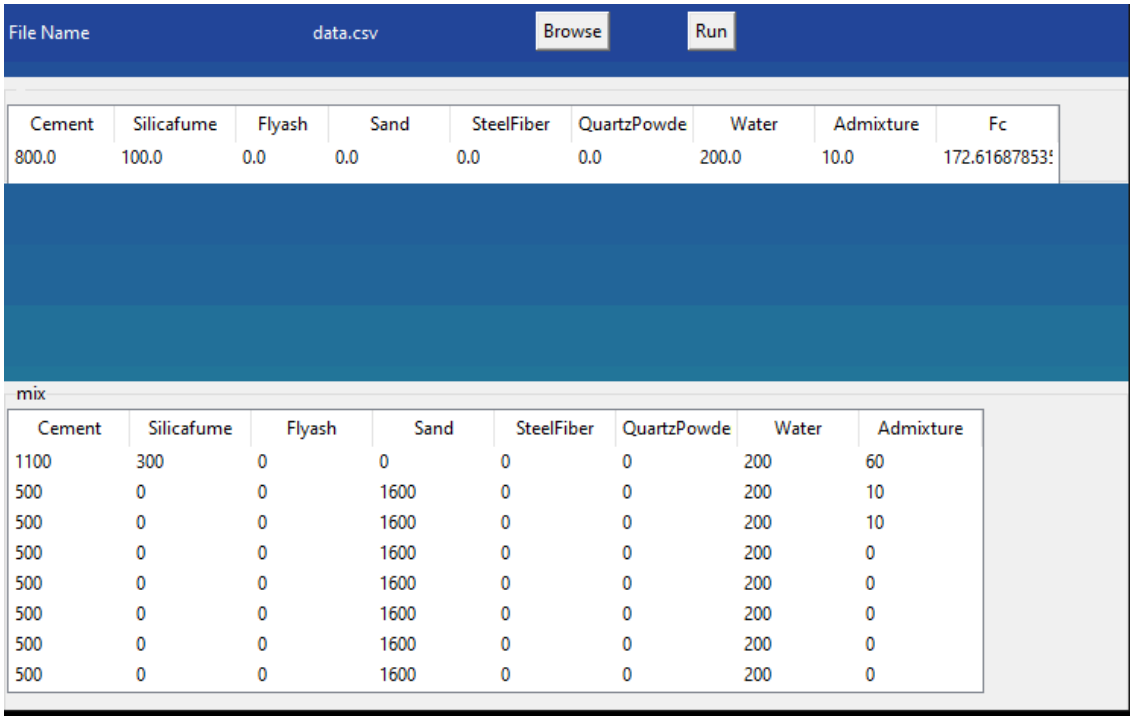


Figure 13 Graphical User Interface of the G-BBO framework [Note: This App can be downloaded at https://www.mznaser.com/fireassessmenttoolsanddatabases].

6. ASSESSMENT OF THE SAMPLED BATCH

The GP-BBO based mixture design method provides a list of formulae (i.e., batch) of UHPC mixtures. The mixture designer can select one or two mixtures to undergo experiments for the verification of satisfactory performance. This preliminary selection could be based on performance requirements such as compressive strength, cost, workability, packing density, and embodied-CO_2 of the UHPC mixture. To evaluate these performance requirements before testing, prediction models from literature could be utilized. For example, Fan et al. [8], Ghafari et al.[49], and Soliman and Tangit-Hamou [50] provide prediction models, under certain constraints of the design space (i.e., range of mixture constitute), for workability, and compressive strength of UHPC. Moreover, Wang et al. [16] and Fan et al. [29] proposed a similar prediction model for the packing density of UHPC. It is not required to obtain high accuracy predictions out of these models as they will be only used to guide the selection from the batch. To evaluate the environmental impact of the UHPC mixture, the embodied-CO_2 of UHPC is estimated. The embodied-CO_2 of the mixture can be estimated as the sum of the embodied-CO_2 of the materials that constitute the mixture,

$$\text{embodied-}CO_2 \text{ of UHPC} = \sum_{i=1}^{N} (\text{embodied} - CO_2)_i \times \text{ingredient}_i \quad (20)$$

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

Table 5 provides the embodied-CO_2 of the materials used in the mixture.

Table 5 Embodied-CO_2 of the raw materials that constitute UHPC

Constitute	Unit embodied-CO_2	Reference
Cement	0.83	[51]
Nano-silica	1.69	[52]
Water	0.0003	[53]
Silica fume	0.001	[53]
Quartz sand	0.01	[53]
Steel fiber	1.5	[54]
Fly ash	0.009	[53]
Superplasticizer (SP)	0.72	[51]
Ground granulated blast-furnace	0.019	[51]

As an example, performing this assessment on a batch designed for a target strength of 150 MPa using PI($\boldsymbol{x}$)- $\alpha\bar{C}(\boldsymbol{x})$ as the acquisition function and α value of 0.5 (four mixtures from that batch are shown in Table 6) results in the predicted values for different performances that are summarized in a radar plot (Figure 14). To select a mixture from the batch that optimizes the multiple goals, a global desirability analysis could be conducted [49]. Alternatively, the whole batch (i.e., all the mixtures), if deemed satisfactory, could undergo the experimental verification step at the same time. Based on the assessment shown in Figure 14, if workability in a range of 150 mm and 250 mm is deemed acceptable, UHPC-1 can be selected as it represents the most economical and eco-friendly mixture.

Table 6 Mixture formulae of developed UHPC (kg/m^3)

Material	UHPC-1	UHPC-2	UHPC-3	UHPC-4
Cement	445	775	700	750
Fly ash	0	0	185	170
Water	125	165	170	180
Silica fume	32.5	200	185	173
Sand	1790	1250	1110	1100
Admixtures (SP)	7.5	10	25	27
Steel fiber	0	0	0	0

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

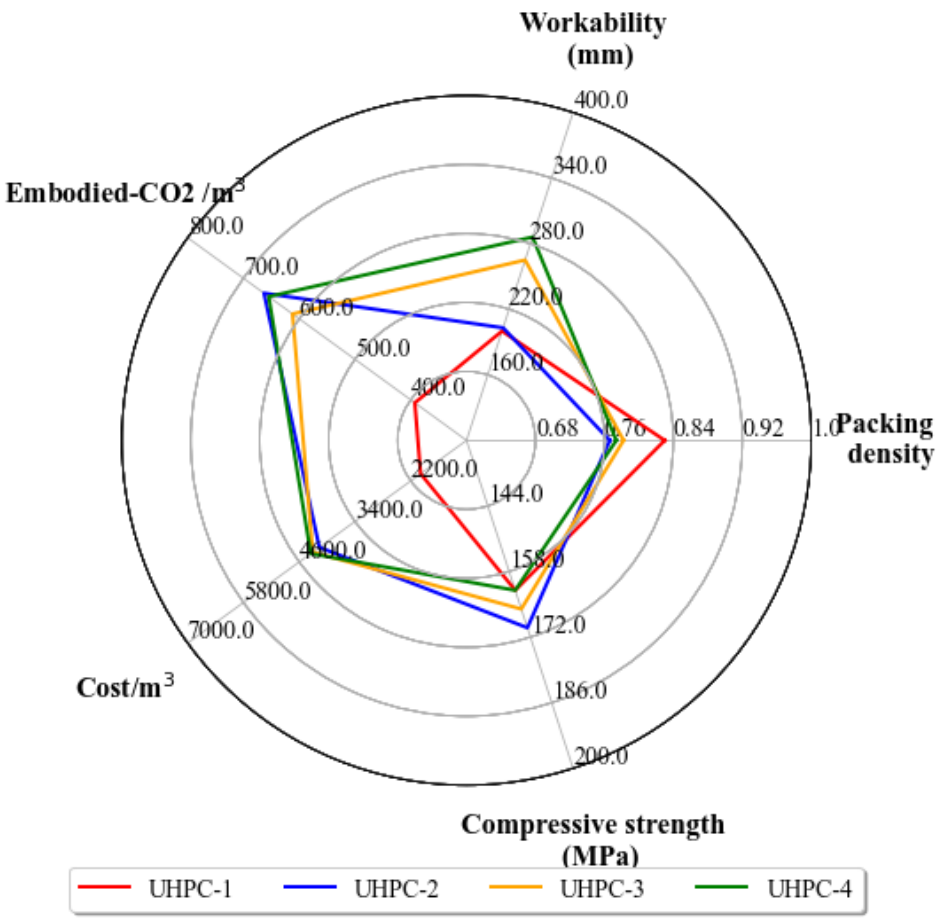


Figure 14 Comparison of the UHPC mixtures from the designed batch

6.1.Experimental verification

After selecting the desired mixtures, experimental verification should be performed to ensure that the predicted performances can be obtained experimentally. The experimental values are then used to augment the UHPC mixture dataset and update the prediction model. This ongoing process of designing and testing UHPC mixtures allows for continuous updating of the dataset and the prediction model. This will empower the proposed GP-BBO framework to identify the global optimum mixture that provides the targeted strength with the lowest cost and/or environmental impact.

To verify that the mixtures designed using the GP-BBO approach can provide the design strength, the compressive strength of the mixtures in Table 6 was estimated from experimentally verified

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

mathematical models in [8], [9], [18], [49], [50] which showed that all the estimated compressive strengths exceeded the designed strength of 150 MPa.

7. COMPARISON WITH OTHER ECO-EFFICIENT MIXTURE DESIGN METHODS

In our developed GP-BBO framework, there is an option to design an eco-friendly UHPC mixture by replacing the normalized cost objective in the acquisition function with a normalized embodied-CO_2 objective. For example, the probability of improvement with cost objective becomes PI($\boldsymbol{x}$)-$\alpha(\overline{\text{embodied} - CO_2})(\boldsymbol{x})$.

Figure 15 compares the compressive strength and embodied-CO_2 of UHPC mixtures optimized using the proposed GP-BBO method with PI($\boldsymbol{x}$)- $\alpha(\overline{\text{embodied} - CO_2})(\boldsymbol{x})$ as the acquisition function and others designed using different traditional and eco-efficient mixture design methods introduced in the relevant literature. As shown in Figure 15, it is shown that optimized UHPC mixtures in this study with an eco-efficient objective have an apparent advantage.

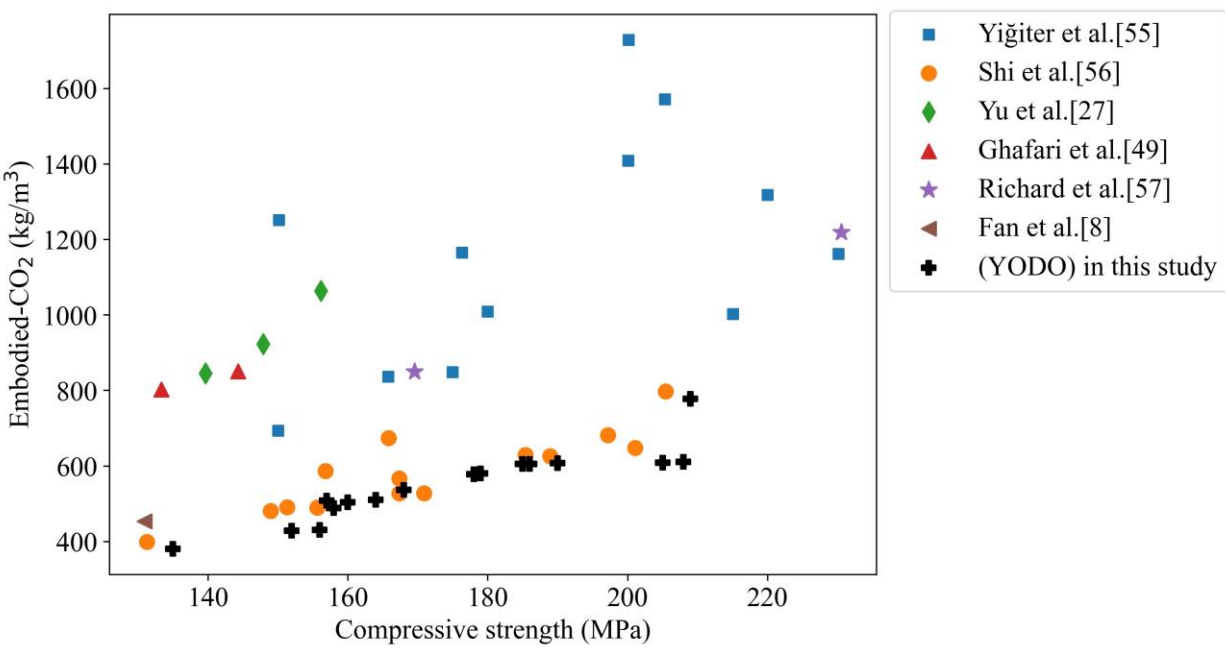


Figure 15 The compressive strength and embodied-CO_2 of different mixtures [8], [27], [49], [55]–[57] and the designed eco-friendly UHPC in this study

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

8. PROSPECTS FOR FUTURE EXTENSION OF GP-BBO FRAMEWORK FOR MIXTURE DESIGN

Future extension of the GP-BBO framework for mixture design formwork might be achieved by enlarging the concrete mixes database to cover a wide range of input ingredients such as different types of fiber reinforcements, recycled aggregate, and SCMs. Additional data on the performance of the concrete mixes, such as slump, cost, and durability characteristics, may be used in a multi-objective BO framework for mixture design. Multi-objective BO will result in multiple equally optimal solutions that form a Pareto front of the feasible samples in the input space. Then a trade-off between the objectives is performed to choose the optimal solution from the feasible set. Recently, a BO with mixed qualitative and quantitative variables has been developed in [30]; this will allow the qualitative data on the mixes to select the optimal ingredients. Qualitative data may include the ease of concrete handling, placing, finishing and type of binder.

Constraints on concrete density, water-binder content, minimum steel fiber content, and quantity of SCMs can be included in the BO using constrained BO algorithms. Constrained BO could avoid the need for BBO; however, the availability of a list of possible formulae rather than only one sampled formula may be more beneficial for the experimental stage (testing more than one trial batch in parallel).

9. CONCLUSIONS

In this study, a data-driven mixture design method for UHPC is proposed, known as GP-BBO. A description of this method is presented in detail. Based on the modeling results, some conclusions can be drawn:

(1) The GP is a robust machine learning method, which showed superiority in the prediction of UHPC compressive strength. The coefficient of correlation (R^2) of the GP model for predicting the compressive strength of UHPC are 0.9, demonstrating a satisfactory accuracy. Moreover, compared to other predictive models, including random forest, nonlinear regression, XGBoost, ANN, linear regression analysis, and GA-ANN, the GP model shows the minimum error and the highest correlation with test values due to the robust approximation provided by the GP in predicting a response variable (e.g., compressive strength) with multiple input variables.

(2) A new UHPC mixture design method is proposed, called GP-BBO, which can be briefly described as follows: at first, select an appropriate acquisition function based on the objective of the mixture design, and then the acquisition function is optimized using the fitted GP, and BBO algorithm to obtain a list of mixture formulae. Assess the list of formulae to specify a formula that meets all the performance requirements. Conduct experimental tests for the selected mixture to confirm that it satisfies the performance

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

requirements and use the experimental values to augment the original dataset of UHPC mixtures.

(3) An eco-friendly UHPC mixture can be developed utilizing the proposed framework with an environmental impact objective. Moreover, compared to other eco-efficient mixture design methods, our proposed framework showed an apparent advantage.

(4) This research provides software with a Graphical User Interface (GUI) for the mixture design of UHPC. Which can be employed for mixture design and compressive strength prediction in the field of UHPC.

(5) The proposed GP-BBO based mixture design method promotes the use of machine learning for precise prediction and mixture design of UHPC. With this, the UHPC industry can make use of new or expanded data for subsequent updates of the model, which can at some point of updating identify the global optimum UHPC mixture.

Conflict of Interest

The authors declare no conflict of interest.

References

[1] C. Shi, Z. Wu, J. Xiao, D. Wang, Z. Huang, and Z. Fang, "A review on ultra high performance concrete: Part I. Raw materials and mixture design," *Construction and Building Materials*, vol. 101, pp. 741–751, 2015.

[2] S. Ahmad, I. Hakeem, and A. K. Azad, "Effect of curing, fibre content and exposures on compressive strength and elasticity of UHPC," *Advances in Cement Research*, vol. 27, no. 4, pp. 233–239, 2015.

[3] F. U. A. Shaikh, S. Luhar, H. Ş. Arel, and I. Luhar, "Performance evaluation of Ultrahigh performance fibre reinforced concrete–A review," *Construction and Building Materials*, vol. 232, p. 117152, 2020.

[4] F. Dingqiang *et al.*, "Optimized design of steel fibres reinforced ultra-high performance concrete (UHPC) composites: Towards to dense structure and efficient fibre application," *Construction and Building Materials*, vol. 273, p. 121698, 2021.

[5] F. de Larrard and T. Sedran, "Optimization of ultra-high-performance concrete by the use of a packing model," *Cement and concrete research*, vol. 24, no. 6, pp. 997–1009, 1994.

[6] R. Yu, P. Spiesz, and H. J. H. Brouwers, "Mix design and properties assessment of ultra-high performance fibre reinforced concrete (UHPFRC)," *Cement and concrete research*, vol. 56, pp. 29–39, 2014.

[7] D.-Y. Yoo and N. Banthia, "Mechanical properties of ultra-high-performance fiber-reinforced concrete: A review," *Cement and Concrete Composites*, vol. 73, pp. 267–280, 2016.

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

[8] D. Fan *et al.*, "A new development of eco-friendly Ultra-High performance concrete (UHPC): Towards efficient steel slag application and multi-objective optimization," *Construction and Building Materials*, vol. 306, p. 124913, 2021.

[9] D. Fan *et al.*, "Precise design and characteristics prediction of Ultra-High Performance Concrete (UHPC) based on artificial intelligence techniques," *Cement and Concrete Composites*, vol. 122, p. 104171, 2021.

[10] D. Q. Fan *et al.*, "A new design approach of steel fibre reinforced ultra-high performance concrete composites: Experiments and modeling," *Cement and Concrete Composites*, vol. 110, p. 103597, 2020.

[11] M. C. G. Juenger and R. Siddique, "Recent advances in understanding the role of supplementary cementitious materials in concrete," *Cement and Concrete Research*, vol. 78, pp. 71–80, 2015.

[12] J. Skibsted and R. Snellings, "Reactivity of supplementary cementitious materials (SCMs) in cement blends," *Cement and Concrete Research*, vol. 124, p. 105799, 2019.

[13] M. Shafieifar, M. Farzad, and A. Azizinamini, "Experimental and numerical study on mechanical properties of Ultra High Performance Concrete (UHPC)," *Construction and Building Materials*, vol. 156, pp. 402–411, 2017.

[14] K. YU, Y. Jiangtao, J. G. Dai, and Z. D. Lu, "P Shah, S.(2018). Development of ultra-high performance engineered cementitious composites using polyethylene (PE) fibers," *Construction and Building Materials*, vol. 158.

[15] K. Wille, A. E. Naaman, and G. J. Parra-Montesinos, "Ultra-High Performance Concrete with Compressive Strength Exceeding 150 MPa (22 ksi): A Simpler Way.," *ACI materials journal*, vol. 108, no. 1, 2011.

[16] X. Wang *et al.*, "Optimized design of ultra-high performance concrete (UHPC) with a high wet packing density," *Cement and Concrete Research*, vol. 126, p. 105921, 2019.

[17] K. H. Khayat, W. Meng, K. Vallurupalli, and L. Teng, "Rheological properties of ultra-high-performance concrete—An overview," *Cement and Concrete Research*, vol. 124, p. 105828, 2019.

[18] E. Ghafari, M. Bandarabadi, H. Costa, and E. Júlio, "Prediction of fresh and hardened state properties of UHPC: comparative study of statistical mixture design and an artificial neural network model," *Journal of Materials in Civil Engineering*, vol. 27, no. 11, p. 04015017, 2015.

[19] M. Zhou, Z. Wu, X. Ouyang, X. Hu, and C. Shi, "Mixture design methods for ultra-high-performance concrete-a review," *Cement and Concrete Composites*, vol. 124, p. 104242, 2021.

[20] R. Wang, X. Gao, H. Huang, and G. Han, "Influence of rheological properties of cement mortar on steel fiber distribution in UHPC," *Construction and Building Materials*, vol. 144, pp. 65–73, 2017.

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

[21] L. Teng, W. Meng, and K. H. Khayat, "Rheology control of ultra-high-performance concrete made with different fiber contents," *Cement and Concrete Research*, vol. 138, p. 106222, 2020.

[22] S. Sunayana and S. v Barai, "Fly ash-incorporated recycled coarse aggregate-based concrete," in *Waste and Byproducts in Cement-Based Materials*, Elsevier, 2021, pp. 229–242.

[23] F. de Larrard, *Concrete mixture proportioning: a scientific approach*. CRC Press, 1999.

[24] T. Stovall, F. de Larrard, and M. Buil, "Linear packing density model of grain mixtures," *Powder technology*, vol. 48, no. 1, pp. 1–12, 1986.

[25] A. Arora, Y. Yao, B. Mobasher, and N. Neithalath, "Fundamental insights into the compressive and flexural response of binder-and aggregate-optimized ultra-high performance concrete (UHPC)," *Cement and Concrete Composites*, vol. 98, pp. 1–13, 2019.

[26] A. Arora, A. Almujaddidi, F. Kianmofrad, B. Mobasher, and N. Neithalath, "Material design of economical ultra-high performance concrete (UHPC) and evaluation of their properties," *Cement and Concrete Composites*, vol. 104, p. 103346, 2019.

[27] R. Yu, P. Spiesz, and H. J. H. Brouwers, "Mix design and properties assessment of ultra-high performance fibre reinforced concrete (UHPFRC)," *Cement and concrete research*, vol. 56, pp. 29–39, 2014.

[28] R. Yu, P. Spiesz, and H. J. H. Brouwers, "Development of Ultra-High Performance Fibre Reinforced Concrete (UHPFRC): Towards an efficient utilization of binders and fibres," *Construction and building materials*, vol. 79, pp. 273–282, 2015.

[29] F. Dingqiang, Y. Rui, S. Zhonghe, W. Chunfeng, W. Jinnan, and S. Qiqi, "A novel approach for developing a green Ultra-High Performance Concrete (UHPC) with advanced particles packing meso-structure," *Construction and Building Materials*, vol. 265, p. 120339, 2020.

[30] Y. Zhang, D. W. Apley, and W. Chen, "Bayesian optimization for materials design with mixed quantitative and qualitative variables," *Scientific reports*, vol. 10, no. 1, pp. 1–13, 2020.

[31] R. Calandra, A. Seyfarth, J. Peters, and M. P. Deisenroth, "Bayesian optimization for learning gaits under uncertainty," *Annals of Mathematics and Artificial Intelligence*, vol. 76, no. 1, pp. 5–23, 2016.

[32] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for hyper-parameter optimization," *Advances in neural information processing systems*, vol. 24, 2011.

[33] M. A. Chilenski *et al.*, "Improved profile fitting and quantification of uncertainty in experimental measurements of impurity transport coefficients using Gaussian process regression," *Nuclear Fusion*, vol. 55, no. 2, p. 023012, 2015.

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

[34] Y. Duan *et al.*, "Using a Gaussian process regression inspired method to measure agreement between the experiment and CFD simulations," *International Journal of Heat and Fluid Flow*, vol. 80, p. 108497, 2019.

[35] O. T. Kajero, R. B. Thorpe, T. Chen, B. Wang, and Y. Yao, "Kriging meta-model assisted calibration of computational fluid dynamics models," *AIChE Journal*, vol. 62, no. 12, pp. 4308–4320, 2016.

[36] D. D. Cox, J.-S. Park, and C. E. Singer, "A statistical method for tuning a computer code to a data base," *Computational statistics & data analysis*, vol. 37, no. 1, pp. 77–92, 2001.

[37] R. Moriconi, M. P. Deisenroth, and K. S. Sesh Kumar, "High-dimensional Bayesian optimization using low-dimensional feature spaces," *Machine Learning*, vol. 109, no. 9, pp. 1925–1943, 2020.

[38] C. E. Rasmussen, "Gaussian processes in machine learning," in *Summer school on machine learning*, 2003, pp. 63–71.

[39] S. Roberts, M. Osborne, M. Ebden, S. Reece, N. Gibson, and S. Aigrain, "Gaussian processes for time-series modelling," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 371, no. 1984, p. 20110550, 2013.

[40] R. Gómez-Bombarelli *et al.*, "Automatic chemical design using a data-driven continuous representation of molecules," *ACS central science*, vol. 4, no. 2, pp. 268–276, 2018.

[41] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *Journal of Global optimization*, vol. 13, no. 4, pp. 455–492, 1998.

[42] V. Picheny, T. Wagner, and D. Ginsbourger, "A benchmark of kriging-based infill criteria for noisy optimization," *Structural and multidisciplinary optimization*, vol. 48, no. 3, pp. 607–626, 2013.

[43] E. Vazquez, J. Villemonteix, M. Sidorkiewicz, and E. Walter, "Global optimization based on noisy evaluations: an empirical study of two statistical approaches," in *Journal of Physics: Conference Series*, 2008, vol. 135, no. 1, p. 012100.

[44] J. González, Z. Dai, P. Hennig, and N. Lawrence, "Batch Bayesian optimization via local penalization," in *Artificial intelligence and statistics*, 2016, pp. 648–657.

[45] B. Standard, "Eurocode 2: Design of concrete structures—," *Part 1*, vol. 1, p. 230, 2004.

[46] O. R. Abuodeh, J. A. Abdalla, and R. A. Hawileh, "Assessment of compressive strength of Ultra-high Performance Concrete using deep machine learning techniques," *Applied Soft Computing*, vol. 95, p. 106552, 2020.

[47] K. Wille and C. Boisvert-Cotulio, "Material efficiency in the design of ultra-high performance concrete," *Construction and Building Materials*, vol. 86, pp. 33–43, 2015.

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

[48] A. K. Akhnoukh and H. Xie, "Welded wire reinforcement versus random steel fibers in precast/prestressed ultra-high performance concrete I-girders," *Construction and Building Materials*, vol. 24, no. 11, pp. 2200–2207, 2010.

[49] E. Ghafari, H. Costa, and E. Júlio, "Statistical mixture design approach for eco-efficient UHPC," *Cement and Concrete Composites*, vol. 55, pp. 17–25, 2015.

[50] N. A. Soliman and A. Tagnit-Hamou, "Using Particle Packing and Statistical Approach to Optimize Eco-Efficient Ultra-High-Performance Concrete.," *ACI Materials Journal*, vol. 114, no. 6, 2017.

[51] G. Long, Y. Gao, and Y. Xie, "Designing more sustainable and greener self-compacting concrete," *Construction and Building Materials*, vol. 84, pp. 301–306, 2015.

[52] A. Lazaro, Q. L. Yu, and H. J. H. Brouwers, "Nanotechnologies for sustainable construction," in *Sustainability of construction materials*, Elsevier, 2016, pp. 55–78.

[53] Y. Zhang *et al.*, "Effect of compressive strength and chloride diffusion on life cycle CO2 assessment of concrete containing supplementary cementitious materials," *Journal of Cleaner Production*, vol. 218, pp. 450–458, 2019.

[54] Y. Shi, G. Long, C. Ma, Y. Xie, and J. He, "Design and preparation of ultra-high performance concrete with low environmental impact," *Journal of Cleaner Production*, vol. 214, pp. 633–643, 2019.

[55] H. Yiğiter, S. Aydın, H. Yazıcı, and M. Y. Yardımcı, "Mechanical performance of low cement reactive powder concrete (LCRPC)," *Composites Part B: Engineering*, vol. 43, no. 8, pp. 2907–2914, 2012.

[56] Y. Shi, G. Long, X. Zen, Y. Xie, and T. Shang, "Design of binder system of eco-efficient UHPC based on physical packing and chemical effect optimization," *Construction and Building Materials*, vol. 274, p. 121382, 2021.

[57] P. Richard and M. Cheyrezy, "Composition of reactive powder concretes," *Cement and concrete research*, vol. 25, no. 7, pp. 1501–1511, 1995.

[58] K. Wille and C. Boisvert-Cotulio, "Material efficiency in the design of ultra-high performance concrete," *Construction and Building Materials*, vol. 86, pp. 33–43, 2015.

[59] M. Alkaysi and S. El-Tawil, "Effects of variations in the mix constituents of ultra high performance concrete (UHPC) on cost and performance," *Materials and Structures*, vol. 49, no. 10, pp. 4185–4200, 2016.

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

[60] A. Alsalman, C. N. Dang, G. S. Prinz, and W. M. Hale, "Evaluation of modulus of elasticity of ultra-high performance concrete," *Construction and Building Materials*, vol. 153, pp. 918–928, 2017.

[61] H. Aoude, F. P. Dagenais, R. P. Burrell, and M. Saatcioglu, "Behavior of ultra-high performance fiber reinforced concrete columns under blast loading," *International Journal of Impact Engineering*, vol. 80, pp. 185–202, 2015.

[62] A. K. Azad and I. Y. Hakeem, "Flexural behavior of hybrid concrete beams reinforced with ultra-high performance concrete bars," *Construction and Building Materials*, vol. 49, pp. 128–133, 2013.

[63] O. Bonneau, M. Lachemi, E. Dallaire, J. Dugat, and P.-C. Aitcin, "Mechanical properties and durability of two industrial reactive powder concretes," *Materials Journal*, vol. 94, no. 4, pp. 286–290, 1997.

[64] A. S. El-Dieb, "Mechanical, durability and microstructural characteristics of ultra-high-strength self-compacting concrete incorporating steel fibers," *Materials & Design*, vol. 30, no. 10, pp. 4286–4292, 2009.

[65] B. A. Graybeal, "Compressive behavior of ultra-high-performance fiber-reinforced concrete," *ACI materials journal*, vol. 104, no. 2, p. 146, 2007.

[66] H. G. Russell, B. A. Graybeal, and H. G. Russell, "Ultra-high performance concrete: A state-of-the-art report for the bridge community.," United States. Federal Highway Administration. Office of Infrastructure …, 2013.

[67] K. Habel, M. Viviani, E. Denarié, and E. Brühwiler, "Development of the mechanical properties of an ultra-high performance fiber reinforced concrete (UHPFRC)," *Cement and Concrete Research*, vol. 36, no. 7, pp. 1362–1370, 2006.

[68] A. M. T. Hassan, S. W. Jones, and G. H. Mahmud, "Experimental test methods to determine the uniaxial tensile and compressive behaviour of ultra high performance fibre reinforced concrete (UHPFRC)," *Construction and building materials*, vol. 37, pp. 874–882, 2012.

[69] M. Hassan and K. Wille, "Experimental impact analysis on ultra-high performance concrete (UHPC) for achieving stress equilibrium (SE) and constant strain rate (CSR) in Split Hopkinson pressure bar (SHPB) using pulse shaping technique," *Construction and Building Materials*, vol. 144, pp. 747–757, 2017.

[70] J. Ma, J. Dietz, and F. Dehn, "Ultra high performance self compacting concrete," *Lacer*, vol. 7, pp. 33–42, 2002.

[71] C. H. Mullen, "Determining the effect of thermal treatment timing on deflections of ultra-high performance concrete beams," 2013.

[72] M. M. Reda, N. G. Shrive, and J. E. Gillott, "Microstructural investigation of innovative UHPC," *Cement and Concrete Research*, vol. 29, no. 3, pp. 323–329, 1999.

[73] M. G. Sohail *et al.*, "Advancements in concrete mix designs: High-performance and ultrahigh-performance concretes from 1970 to 2016," *Journal of Materials in Civil Engineering*, vol. 30, no. 3, p. 04017310, 2018.

[74] C. Wang, C. Yang, F. Liu, C. Wan, and X. Pu, "Preparation of ultra-high performance concrete with common technology and materials," *Cement and concrete composites*, vol. 34, no. 4, pp. 538–544, 2012.

[75] J. A. Willey, *Use of ultra-high performance concrete to mitigate impact and explosive threats*. Missouri University of Science and Technology, 2013.

[76] D. Yao, J. Jia, F. Wu, and F. Yu, "Shear performance of prestressed ultra high strength concrete encased steel beams," *Construction and Building Materials*, vol. 52, pp. 194–201, 2014.

[77] R. Zhong, K. Wille, and R. Viegas, "Material efficiency in the design of UHPC paste from a life cycle point of view," *Construction and Building Materials*, vol. 160, pp. 505–513, 2018.

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

Appendix A

Table 1A. Collected database

No.	Cement	Water	Sand	Flyash	Admixture	Silica fume	Quartz Powder	Steel Fiber	CS	Ref.
1	778	171.35	1167	189	28	194	0	0	181	[58]
2	778	185.61	1166	189	28	194	0	0	185	[58]
3	758	170.2	1138	184	28	190	0	0	155	[58]
4	753	170.88	1129	183	27	188	0	0	166	[58]
5	745	178.94	1118	181	27	186	0	0	159	[58]
6	745	178.94	1118	181	27	186	0	0	162	[58]
7	740	173.19	1110	180	27	185	0	0	200	[58]
8	815	163	0	0	0	65.2	0	8.15	194	[2]
9	664	144	1231	142	56	142	0	0	104	[48]
10	775	165	1220	0	10	194	0	0	170	[59]
11	775	165	1220	0	10	194	0	23.6	178	[59]
12	1600	320	310	0	77.22	320	0	0	124.1	[60]
13	1600	320	310	0	77.22	320	0	105.0888	127.6	[60]
14	1600	320	310	0	77.22	320	0	52.5444	128.3	[60]
15	1600	273	310	0	38.22	273	0	0	135	[60]
16	1600	320	292	0	35.37	320	0	0	135.5	[60]
17	1600	273	310	0	38.22	273	0	51.7844	135.9	[60]
18	1600	320	310	0	38.22	320	0	103.6088	143.2	[60]
19	1600	320	310	0	78.22	320	0	157.6332	144.1	[60]
20	1600	320	292	0	37.37	320	0	102.6948	144.7	[60]
21	1600	320	310	0	38.22	320	0	155.4732	145.7	[60]
22	1600	320	292	0	36.37	320	0	51.3474	146.8	[60]
23	1600	320	292	0	38.37	320	0	154.0422	162.4	[60]
24	1365	151.6	647	0	30.26	151.6	0	0	136.4	[60]
25	1365	149	647	0	30.26	149	0	51.7652	137.9	[60]
26	1365	273	647	0	30.26	273	0	103.5304	140.8	[60]
27	1365	273	647	0	30.26	273	0	155.2956	155.3	[60]
28	917	202	1443	0	0	229.2	0	55.8	145	[61]
29	917	202	1443	0	0	229.2	0	111.6	153	[61]
30	917	202	1443	0	0	229	0	167.5	165	[61]
31	900	162	1005	0	40	220	0	46.7	160	[62]
32	1107	195	0	0	61.992	343.17	0	0	217	[63]
33	900	207	1030	0	0	157.5	0	0	114	[64]

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). “You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete”. Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

No.	Cement	Water	Sand	Flyash	Admixture	Silica fume	Quartz Powder	Steel Fiber	CS	Ref.
34	900	216	1029.3	0	0	157.5	0	1.84	115	[64]
35	900	216	1028	0	0	157.5	0	2.76	116	[64]
36	900	216	1017.7	0	0	157.5	0	11.9	130	[64]
37	900	252	1293.6	0	0	135	0	0	185	[64]
38	900	252	1189	0	0	135	0	0	195	[64]
39	775	209.3	1297	0	0	116.3	0	0	192	[64]
40	712	109	1020	0	30.7	231	211	156	138	[65]
41	710	110	1020	0	30	230	210	156	119	[65]
42	712	113.2	1231	0	32	231	0	46.4	132	[66]
43	1050	190	730	0	35	275	0	137.5	169	[67]
44	1050	190	0	0	35	275	730	470	175	[67]
45	657	185	1051	0	185	119	418	0	121.32	[68]
46	657	185	1051	0	185	119	418	157	150.56	[68]
47	813	160.1	1157	0	29.3	203.3	203.3	0	204	[69]
48	665	178	1019	0	25	200	285	0	155	[70]
49	609	163	1334	0	21	183	263	0	155	[70]
50	710	150	1231	0	13.4	230	0	47.9	138	[71]
51	1115	334.5	0	0	88.085	367.95	0	0	210	[72]
52	1040	240	800	0	46.944	310	0	0	140	[72]
53	1040	240	800	0	46.944	310	0	0	140	[72]
54	1040	240	800	0	20.8	310	0	0	140	[72]
55	1040	240	800	0	46.944	310	0	0	165	[72]
56	1040	240	800	0	46.944	310	0	47.8	165	[72]
57	1040	240	800	0	20.8	310	0	48.2	165	[72]
58	1040	240	800	0	46.944	310	0	47.8	195	[72]
59	1040	240	800	0	46.944	310	0	47.8	210	[72]
60	1040	240	800	0	46.944	310	0	47.8	230	[72]
61	807	196	972	0	13	225	243	0	204	[72]
62	1033	155	1136.7	0	16.5	258	0	0	150	[57]
63	959	163	1055	0	15	239.8	0	52.3	200	[57]
64	712	109	1231	0	30.7	231	0	0	101	[13]
65	967	244	675	0	35	251	0	430	128	[73]
66	833	195.755	0	0	33.32	199.92	0	0	162	[73]
67	820	219	624	0	65	273	0	157	95	[73]

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

No.	Cement	Water	Sand	Flyash	Admixture	Silica fume	Quartz Powder	Steel Fiber	CS	Ref.
68	820	219	702	0	65	273	0	157	108	[73]
69	712	180	1142	125	25	115	0	0	125	[73]
70	711	137.2	0	0	10.665	85.32	0	0	115	[73]
71	675	180	1179	125	25	115	0	0	118	[73]
72	675	180	1178	125	25	115	0	0	128	[73]
73	637	180	1224	125	25	115	0	0	110	[73]
74	637	180	1217	125	25	115	0	0	118	[73]
75	550	137	0	0	16.5	82.5	0	0	142	[73]
76	446	126	1838	0	7.4	32.6	0	0	162	[73]
77	510	140	1700	0	46.944	65	0	0	175	[74]
78	510	140	1700	0	46.944	65	0	11.95	195	[74]
79	510	140	1700	0	46.944	65	0	11.95	240	[74]
80	450	140	1720	0	46.944	50	0	11.95	145	[74]
81	450	140	1720	0	46.944	50	0	11.95	175	[74]
82	800	230	0	0	25	150	750	20	119	[75]
83	800	247	1381	0	34	261	0	60.7	124	[75]
84	800	247	1381	0	34	261	0	60.7	138	[75]
85	790	141	1141	192	28.4	198	0	37.4	155	[75]
86	786	190	1353	0	33	256	0	0	103	[75]
87	786	227	1356	0	33	256	0	57.3	112	[75]
88	786	227	1356	0	33	256	0	58.2	154	[75]
89	784	190	1353	0	36	256	0	0	110	[75]
90	784	190	1253	0	33	256	0	0	123	[75]
91	784	190	1353	0	33	256	0	0	134	[75]
92	784	190	1353	0	33	256	0	57	149	[75]
93	750	180	1111	125	25	115	0	0	116	[75]
94	750	180	1103	125	25	115	0	0	124	[75]
95	750	180	1104	125	25	115	0	0	125	[75]
96	750	180	1104	125	25	115	0	0	128	[75]
97	731	190	1353	0	31	239	0	0	120	[75]
98	420	138	1650	120	9	60	0	0	110	[76]
99	398	145	1734.6	185	7.3	0	0	0	100	[76]
100	1327.8	254.4	0	0	47.8	332	332	0	171	[77]
101	845	175.45	0	0	0	76.1	0	12.6	188	[77]
102	715	167	0	0	30.745	157.3	0	0	178	[77]
103	550	151	1603	0	21	0	0	0	199	[77]

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

No.	Cement	Water	Sand	Flyash	Admixture	Silica fume	Quartz Powder	Steel Fiber	CS	Ref.
104	450	126	1800	0	14	0	0	0	175	[77]
105	443	228	1717	448	0.4	0	0	0	171	[77]
106	432	160.2	1636	132	7.9	29.7	0	0	178	[77]
107	412	127	1898	0	13	41	0	0	198	[77]
108	408	190	1520	0	5.6	39	0	0	178	[77]
109	406	185	1558	0	4.3	40.6	0	0	180	[77]
110	383	125	1800	0	14	67.5	0	0	187	[77]

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

Appendix B

ML code used in this study

```
#Classes and functions used in batch Bayesian optimization for mixture design #

# Copyright (c) 2016, the GPyOpt Authors
# Licensed under the BSD 3-clause license (see LICENSE.txt)
class Sequential(EvaluatorBase):
    """
    Class for standard Sequential Bayesian optimization methods.
    :param acquisition: acquisition function to be used to compute the batch.
    :param batch size: it is 1 by default since this class is only used for sequential methods.
    """

    def __init__(self, acquisition, batch_size=1):
        super(Sequential, self).__init__(acquisition, batch_size)

    def compute_batch(self, duplicate_manager=None,context_manager=None):
        """
        Selects the new location to evaluate the objective.
        """
        x, _ = self.acquisition.optimize(duplicate_manager=duplicate_manager)
        return x
# Copyright (c) 2016, the GPyOpt Authors
# Licensed under the BSD 3-clause license (see LICENSE.txt)

import numpy as np


class EvaluatorBase(object):
```

```
    """
    Base class for the evaluator of the function. This class handles both sequential and batch evaluators.
    """

    def __init__(self, acquisition, batch_size, **kwargs):
        self.acquisition = acquisition
        self.batch_size = batch_size

    def compute_batch(self, duplicate_manager=None, context_manager=None):
        raise NotImplementedError("Need to implement compute_batch.")


class SamplingBasedBatchEvaluator(EvaluatorBase):
    """
    This class handles specific types of batch evaluators, based on the sampling of anchor points (examples are random
and Thompson sampling).
    """

    def __init__(self, acquisition, batch_size, **kwargs):
        self.acquisition = acquisition
        self.batch_size = batch_size
        self.space = acquisition.space
        # The following number of anchor points is heuristically picked, to obtain good and various batches
        self.num_anchor = 5*batch_size

    def initialize_batch(self, duplicate_manager=None, context_manager=None):
        raise NotImplementedError("Need to implement initialize_batch.")

    def get_anchor_points(self, duplicate_manager=None, context_manager=None):
```

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). “You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete”. Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

```
        raise NotImplementedError("Need to implement get_anchor_points.")

    def optimize_anchor_point(self, a, duplicate_manager=None, context_manager=None):
        raise NotImplementedError("Need to implement optimize_anchor_point.")

    def compute_batch_without_duplicate_logic(self,context_manager=None):
        raise NotImplementedError("Need to implement compute_batch_without_duplicate_logic.")

    def compute_batch(self, duplicate_manager=None, context_manager=None):

        self.context_manager = context_manager

        # Easy case where we do not care about having duplicates suggested
        if not duplicate_manager:
            return self.compute_batch_without_duplicate_logic(context_manager=self.context_manager)

        batch, already_suggested_points = [], duplicate_manager.unique_points.copy()

        anchor_points = self.get_anchor_points(duplicate_manager=duplicate_manager,
context_manager=self.context_manager)

        x0 = self.initialize_batch(duplicate_manager=duplicate_manager, context_manager = self.context_manager)

        if np.any(x0):
            batch.append(x0)
            already_suggested_points.add(self.zip_and_tuple(x0))

        for a in anchor_points:
```

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

```
        x = self.optimize_anchor_point(a, duplicate_manager=duplicate_manager, context_manager = self.context_manager)

        # We first try to add the optimized anchor point; if we cannot, we then try the initial anchor point.
        zipped_x = self.zip_and_tuple(x)

        if zipped_x not in already_suggested_points:
            batch.append(x)
            already_suggested_points.add(zipped_x)
        else:
            zipped_a = self.zip_and_tuple(a)

            if zipped_a not in already_suggested_points:
                batch.append(a)
                already_suggested_points.add(zipped_a)

        if len(batch) == self.batch_size:
            break

    if len(batch) < self.batch_size:
        # Note that the case where anchor_points is empty is handled in self.get_anchor_points that would throw a FullyExploredOptimizationDomainError
        print("Warning: the batch of requested size {} could not be entirely filled in (only {} points)".format(self.batch_size, len(batch)))

    return np.vstack(batch)

  def zip_and_tuple(self, x):
    """
    convenient helper
```

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). “You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete”. Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

```
        :param x: input configuration in the model space
        :return: zipped x as a tuple
        """
        return tuple(self.space.zip_inputs(np.atleast_2d(x)).flatten())




import scipy
import numpy as np


class LocalPenalization(EvaluatorBase):
    """
    Class for the batch method on 'Batch Bayesian optimization via local penalization' (Gonzalez et al., 2016).
    :param acquisition: acquisition function to be used to compute the batch.
    :param batch size: the number of elements in the batch.
    """
    def __init__(self, acquisition, batch_size):
        super(LocalPenalization, self).__init__(acquisition, batch_size)
        self.acquisition = acquisition
        self.batch_size = batch_size

    def compute_batch(self, duplicate_manager=None, context_manager=None):
        """
        Computes the elements of the batch sequentially by penalizing the acquisition.
        """
        from ...acquisitions import AcquisitionLP
```

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

```
        assert isinstance(self.acquisition, AcquisitionLP)

        self.acquisition.update_batches(None,None,None)

        # --- GET first element in the batch
        X_batch = self.acquisition.optimize()[0]
        k=1

        if self.batch_size >1:
            # ---------- Approximate the constants of the the method
            L = estimate_L(self.acquisition.model.model,self.acquisition.space.get_bounds())
            Min = self.acquisition.model.model.Y.min()

        # --- GET the remaining elements
        while k<self.batch_size:
            self.acquisition.update_batches(X_batch,L,Min)
            new_sample = self.acquisition.optimize()[0]
            X_batch = np.vstack((X_batch,new_sample))
            k +=1

        # --- Back to the non-penalized acquisition
        self.acquisition.update_batches(None,None,None)
        return X_batch


def estimate_L(model,bounds,storehistory=True):
    """
    Estimate the Lipschitz constant of f by taking maximizing the norm of the expectation of the gradient of *f*.
    """
```

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

```
    def df(x,model,x0):
        x = np.atleast_2d(x)
        dmdx,_ = model.predictive_gradients(x)
        if dmdx.ndim>2:
            dmdx = dmdx.reshape(dmdx.shape[:2])
        res = np.sqrt((dmdx*dmdx).sum(1)) # simply take the norm of the expectation of the gradient
        return -res

    samples = samples_multidimensional_uniform(bounds,500)
    samples = np.vstack([samples,model.X])
    pred_samples = df(samples,model,0)
    x0 = samples[np.argmin(pred_samples)]
    res = scipy.optimize.minimize(df,x0, method='L-BFGS-B',bounds=bounds, args = (model,x0), options = {'maxiter': 200})
    minusL = float(res.fun)
    L = -minusL
    if L<1e-7: L=10  ## to avoid problems in cases in which the model is flat.
    return L
################################Classes and functions used in batch bayesian optamization for mixture design ######################

# Copyright (c) 2016, the GPyOpt Authors
# Licensed under the BSD 3-clause license (see LICENSE.txt)
class Sequential(EvaluatorBase):
    """
    Class for standard Sequential Bayesian optimization methods.
    :param acquisition: acquisition function to be used to compute the batch.
    :param batch size: it is 1 by default since this class is only used for sequential methods.
    """
```

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). “You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete”. Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

```

    def __init__(self, acquisition, batch_size=1):
        super(Sequential, self).__init__(acquisition, batch_size)

    def compute_batch(self, duplicate_manager=None,context_manager=None):
        """
        Selects the new location to evaluate the objective.
        """
        x, _ = self.acquisition.optimize(duplicate_manager=duplicate_manager)
        return x
# Copyright (c) 2016, the GPyOpt Authors
# Licensed under the BSD 3-clause license (see LICENSE.txt)

import numpy as np


class EvaluatorBase(object):
    """
    Base class for the evaluator of the function. This class handles both sequential and batch evaluators.
    """

    def __init__(self, acquisition, batch_size, **kwargs):
        self.acquisition = acquisition
        self.batch_size = batch_size

    def compute_batch(self, duplicate_manager=None, context_manager=None):
        raise NotImplementedError("Need to implement compute_batch.")


```

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). “You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete”. Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

```
class SamplingBasedBatchEvaluator(EvaluatorBase):
    """
    This class handles specific types of batch evaluators, based on the sampling of anchor points (examples are random
and Thompson sampling).
    """

    def __init__(self, acquisition, batch_size, **kwargs):
        self.acquisition = acquisition
        self.batch_size = batch_size
        self.space = acquisition.space
        # The following number of anchor points is heuristically picked, to obtain good and various batches
        self.num_anchor = 5*batch_size

    def initialize_batch(self, duplicate_manager=None, context_manager=None):
        raise NotImplementedError("Need to implement initialize_batch.")

    def get_anchor_points(self, duplicate_manager=None, context_manager=None):
        raise NotImplementedError("Need to implement get_anchor_points.")

    def optimize_anchor_point(self, a, duplicate_manager=None, context_manager=None):
        raise NotImplementedError("Need to implement optimize_anchor_point.")

    def compute_batch_without_duplicate_logic(self,context_manager=None):
        raise NotImplementedError("Need to implement compute_batch_without_duplicate_logic.")

    def compute_batch(self, duplicate_manager=None, context_manager=None):

        self.context_manager = context_manager

```

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

```
    # Easy case where we do not care about having duplicates suggested
    if not duplicate_manager:
        return self.compute_batch_without_duplicate_logic(context_manager=self.context_manager)

    batch, already_suggested_points = [], duplicate_manager.unique_points.copy()

    anchor_points = self.get_anchor_points(duplicate_manager=duplicate_manager, context_manager=self.context_manager)

    x0 = self.initialize_batch(duplicate_manager=duplicate_manager, context_manager = self.context_manager)

    if np.any(x0):
        batch.append(x0)
        already_suggested_points.add(self.zip_and_tuple(x0))

    for a in anchor_points:
        x = self.optimize_anchor_point(a, duplicate_manager=duplicate_manager, context_manager = self.context_manager)

        # We first try to add the optimized anchor point; if we cannot, we then try the initial anchor point.
        zipped_x = self.zip_and_tuple(x)

        if zipped_x not in already_suggested_points:
            batch.append(x)
            already_suggested_points.add(zipped_x)
        else:
            zipped_a = self.zip_and_tuple(a)

            if zipped_a not in already_suggested_points:
```

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

```
                batch.append(a)
                already_suggested_points.add(zipped_a)

            if len(batch) == self.batch_size:
                break

        if len(batch) < self.batch_size:
            # Note that the case where anchor_points is empty is handled in self.get_anchor_points that would throw a FullyExploredOptimizationDomainError
            print("Warning: the batch of requested size {} could not be entirely filled in (only {} points)".format(self.batch_size, len(batch)))

        return np.vstack(batch)

    def zip_and_tuple(self, x):
        """
        convenient helper
        :param x: input configuration in the model space
        :return: zipped x as a tuple
        """
        return tuple(self.space.zip_inputs(np.atleast_2d(x)).flatten())




import scipy
import numpy as np

```

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). “You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete”. Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

```
class LocalPenalization(EvaluatorBase):
    """
    Class for the batch method on 'Batch Bayesian optimization via local penalization' (Gonzalez et al., 2016).
    :param acquisition: acquisition function to be used to compute the batch.
    :param batch size: the number of elements in the batch.
    """
    def __init__(self, acquisition, batch_size):
        super(LocalPenalization, self).__init__(acquisition, batch_size)
        self.acquisition = acquisition
        self.batch_size = batch_size

    def compute_batch(self, duplicate_manager=None, context_manager=None):
        """
        Computes the elements of the batch sequentially by penalizing the acquisition.
        """
        from ...acquisitions import AcquisitionLP
        assert isinstance(self.acquisition, AcquisitionLP)

        self.acquisition.update_batches(None,None,None)

        # --- GET first element in the batch
        X_batch = self.acquisition.optimize()[0]
        k=1

        if self.batch_size >1:
            # ---------- Approximate the constants of the the method
            L = estimate_L(self.acquisition.model.model,self.acquisition.space.get_bounds())
            Min = self.acquisition.model.model.Y.min()
```

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

```
        # --- GET the remaining elements
        while k<self.batch_size:
            self.acquisition.update_batches(X_batch,L,Min)
            new_sample = self.acquisition.optimize()[0]
            X_batch = np.vstack((X_batch,new_sample))
            k +=1

        # --- Back to the non-penalized acquisition
        self.acquisition.update_batches(None,None,None)
        return X_batch


def estimate_L(model,bounds,storehistory=True):
    """
    Estimate the Lipschitz constant of f by taking maximizing the norm of the expectation of the gradient of *f*.
    """
    def df(x,model,x0):
        x = np.atleast_2d(x)
        dmdx,_ = model.predictive_gradients(x)
        if dmdx.ndim>2:
            dmdx = dmdx.reshape(dmdx.shape[:2])
        res = np.sqrt((dmdx*dmdx).sum(1)) # simply take the norm of the expectation of the gradient
        return -res

    samples = samples_multidimensional_uniform(bounds,500)
    samples = np.vstack([samples,model.X])
    pred_samples = df(samples,model,0)
    x0 = samples[np.argmin(pred_samples)]
```

This is a preprint draft. The published article can be found at: https://doi.org/10.1016/j.conbuildmat.2022.127270

Please cite this paper as:

Saleh E., Tarawneh A., Naser M.Z., Abedi M., Almasabha G., (2021). "You Only Design Once (YODO): Gaussian Process-Batch Bayesian Optimization framework for Mixture Design of Ultra-High-Performance Concrete". Construction and Building Materials. https://doi.org/10.1016/j.conbuildmat.2022.127270

```
    res = scipy.optimize.minimize(df,x0, method='L-BFGS-B',bounds=bounds, args = (model,x0), options = {'maxiter': 200})
    minusL = float(res.fun)
    L = -minusL
    if L<1e-7: L=10  ## to avoid problems in cases in which the model is flat.
    return L
```