

Article

Not peer-reviewed version

Multivariate And Multidimensional Quality Gain-Loss Function and Its Applications Based on Nonseparable Gaussian Processes

[Aili Wang](#), [Xianfei Chen](#), Jiahang Liu, [Shunan Tong](#), [Yizhou Li](#), [Tianyu Fan](#)*

Posted Date: 16 April 2026

doi: 10.20944/preprints202604.1225.v1

Keywords: quality gain-loss function; nonseparable Gaussian process; spatiotemporal interaction; signal-to-noise ratio; quality compensation



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a [Creative Commons CC BY 4.0 license](#), which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

Multivariate And Multidimensional Quality Gain-Loss Function and Its Applications Based on Nonseparable Gaussian Processes

Aili Wang ¹, Xianfei Chen ¹, Jiahang Liu ², Shunan Tong ³, Yizhou Li ³ and Tianyu Fan ^{1,*}

¹ School of Water Conservancy, North China University of Water Resources and Electric Power, Zhengzhou 450046, China

² Langfang Water Conservancy Bureau, Langfang 065000, China

³ Langfang Water Development Group Co., Ltd., Langfang 065000, China

* Correspondence: fantianyu92@126.com

Abstract

Existing research on quality gain-loss functions predominantly focuses on single variables or separable quality characteristics, overlooking the correlations among multiple quality attributes and the complexity of spatiotemporal factors. This paper employs the Matérn kernel to construct spatiotemporal interaction terms, incorporates Kalman filtering and smoothing algorithms to enhance computational efficiency, and establishes joint gain-loss weights using the signal-to-noise ratio method. Consequently, a multivariate multidimensional quality gain-loss function model based on the Non-Separable Gaussian Process (NSGP) is developed. The NSGP model is applied to simulation cases and dam concrete production scenarios. Comparative optimization with machine learning methods such as Gaussian processes and linear regression validates the robustness of the NSGP model. Crucially, it eliminates the computational requirement for determining covariance separability, thereby reducing computational costs. This provides robust case support for quality management in hydraulic concrete construction.

Keywords: quality gain-loss function; nonseparable Gaussian process; spatiotemporal interaction; signal-to-noise ratio; quality compensation

1. Introduction

In the 1960s, Taguchi [1] associated product quality with economic loss and proposed that the quality loss function attains its minimum value of zero when quality characteristics exactly match their target values. Any deviation of the quality characteristic value from the target value will result in quality loss. Spiring [23] proposed an asymmetric quality loss function to capture the asymmetric nature of losses in production processes. To address the unboundedness of the loss function, he further developed an inverted normal quality loss function. Building upon Taguchi's quality loss function, M Abolmohammadi et al. [4] extended the loss function by incorporating linear, quadratic, exponential, and Linex functions, thereby establishing the Linex predicted loss function. Artiles-leon [5] proposed a dimensionless "standardized" multi-parameter quality loss function. Pignatiello [6] introduced a multivariate quality loss function to address scenarios where the overall quality of products is jointly influenced by multiple quality characteristics. **Error! Reference source not found.** developed Type I and Type II exponential quality loss function models which is more suitable for the real scene, accounting for the practical tendency to replace rather than repair scrapped products and the asymmetric nature of quality losses. Wang et al. [7] optimized the quality cost curve by incorporating the Cobb-Douglas function into the Taguchi quality loss function. Cao et al. [8] developed a fuzzy quality loss function model addressing the fuzzy nature of product quality. Wang et al. [9] established a robust parameter optimization model for multiple characteristics based on

Bayesian posterior probability theory. Ni et al. 11 conducted an in-depth research on process mean estimation under asymmetric loss functions.

To address mutual compensation and adaptive coupling phenomena between processes in engineering construction—where subsequent processes compensate for losses induced by preceding ones, and even processes within the same sequence complement each other to reduce overall quality loss, thereby producing a “quality compensation” effect. Wang et al. 13 defined the constant term in the Taylor series expansion as quality compensation and introduced the concept of a quality gain-loss function. Assuming constant compensation while retaining linear loss terms, they developed “larger the better” and “smaller the better” quality gain-loss function models. 14 Subsequently, fuzzy quality gain-loss function models 15 and grey quality gain-loss function models 17 were proposed based on fuzzy theory and grey system theory. To address the boundedness of quality gains and losses, an inverted normal quality gain-loss function was proposed, leading to the construction of a multivariate inverted normal model 19. By retaining the third-order Taylor expansion, cubic and asymmetric cubic quality gain-loss functions were proposed 20 . To address asymmetry in quality gain-loss functions, a quadratic exponential quality gain-loss function model was developed 22 . To correct loss bias arising from omitting the cubic term, a cubic-term loss-considering “larger the better” and “smaller the better” quality gain-loss function model was proposed 23 . In engineering projects, quality losses are typically caused by the combined effects of multiple quality characteristics. Accordingly, a wide range of multivariate modeling and analysis methods are available, including Bayesian networks, multiple linear regression, random forests, nearest neighbor methods, and Gaussian processes.

Non-separable Gaussian Processes (NSGP) is an extension of Gaussian Processes (GP) designed to model complex spatial and temporal data where the correlation between spatial and temporal factors is relatively complex. Fricker Thomas E et al. 24 highlighted the flexibility and predictive strength of non-separable covariance structures; Ilias Billionis et al. 25 proposed a posteriori probability measure for potential surrogate function spaces defined by Gaussian Process models; Datta Abhirup et al. 26 developed a scalable dynamic k-nearest neighbor Gaussian process model capable of providing sparse approximations for arbitrary spatiotemporal GPs. Dionissios T. et al. 27 employed a hybrid spectral approach to develop a new non-separable covariance kernel, validating it under various damping conditions; Gu et al. 28 proposed a fast non-separable Gaussian process, combined it with linear regression to predict methylation levels, and demonstrated its superiority; Zhang et al. 29 introduced a multi-output Gaussian process simulator with a non-separable auto-covariance function; Cui et al. 30 applied the fast non-separable Gaussian process model to robust parameter design. In summary, researchers worldwide have investigated non-separable Gaussian processes from the perspectives of kernel construction and algorithmic optimization, achieving several practical applications. Nevertheless, this line of research has not yet received widespread attention.

This paper extends the quality gain-loss function by incorporating the correlation between spatiotemporal factors and the weight distribution of multi-dimensional quality characteristics on the basis of overall quality gain-loss theory. In response to the limitations of existing related research—which often neglects factor interactions and struggles to balance robustness with computational efficiency when handling multidimensional quality characteristics—this study introduces the NSGP, thereby overcoming the assumption of separability between spatial and temporal factors. Within this framework, a multidimensional quality gain-loss function model reflecting spatiotemporal interaction effects is constructed. The proposed approach not only improves predictive accuracy and computational efficiency but also enables the optimized design of multiple quality characteristics in complex engineering projects. It provides new theoretical support and practical application pathways for quality control in hydraulic concrete construction and related engineering fields.

2. Nonseparable Gaussian Process Regression Model

2.1. Quality Characteristic Data

Let $\{y_i(s_j) | i = 1, 2, \dots, K, j = 1, 2, \dots, N\}$ denote the quality characteristic data set of $K = k + k^*$ samples across $N = n + n^*$ observation locations/time points. Here $y_i(s_j)$ represents the quality characteristic value of the i th sample at the j th location/time point.

Define two sets of observation locations or time points, denoted as $S^D = \{s_1^D, s_2^D, \dots, s_n^D\}$ and $S^* = \{s_1^*, s_2^*, \dots, s_{n^*}^*\}$. For S^D , the quality characteristic level $S^D = \{s_1^D, s_2^D, \dots, s_n^D\}$ can be observed for all $K = k + k^*$ samples. In contrast, for S^* , the quality characteristic level $S^* = \{s_1^*, s_2^*, \dots, s_{n^*}^*\}$ is only observed for the first k samples, while it is unobservable for the remaining k^* samples. Therefore, for S^D , the quality characteristic values $y(s^D)_{k \times n}$ and $y^*(s^D)_{k^* \times n}$ are available for all samples. For S^* , only the quality characteristic values $y(s^*)_{k \times n^*}$ for the first k samples can be obtained, while the quality characteristic values $y^*(s^*)_{k^* \times n^*}$ for the remaining k^* samples are unobservable. The objective of this paper is to predict the distribution of $y^*(s^*)$, given the known quality characteristic values $y(s^D)$, $y(s^*)$, and $y^*(s^D)$.

2.2. Nonseparable Gaussian Process Modeling

Starting from the common output region $Y(s)$, the $K \times 1$ -dimensional output vector at each location or time point $s \in \varphi$ can be expressed as:

$$Y(s) = A\tilde{v}(s) + \varepsilon_0 \quad (1)$$

Where $\varepsilon_0 \sim \text{MN}(0, \sigma_0^2 I_K)$, and $A = (a_1, \dots, a_K)$ are the $K \times K$ matrix, and a_i is the i -th factor loading vector. The vector $\tilde{v}(s) = (\tilde{v}_1(s), \dots, \tilde{v}_K(s))^T$ denotes the random factor vector at observation location or time point s . If each random factor process $\tilde{v}_i(\cdot)$ is independently modeled as a zero-mean GP model, with its noise term belonging to the class $s \in \varphi$, then for any observation location or time point s in φ , we have:

$$\begin{aligned} \tilde{v}_i(s) &= v_i(s) + \varepsilon_i \\ v_i(\cdot) &\sim \text{GP}(0, \tau_i^2 c_i(\cdot, \cdot)) \end{aligned} \quad (2)$$

Among these, $\varepsilon_i \sim N(0, \sigma_i^2)$ represents the independent noise, and $\tau_i^2 c_i(s_a, s_b)$ denotes the covariance between any pair of location or time point $s_a, s_b \in \varphi$. The parameter σ_i^2 , along with τ_i^2 , correspond to their respective unknown variances.

Let the signal-to-noise ratio parameter be denoted as $\eta_i = \frac{\sigma_i^2}{\tau_i^2}$. From equation (2), the marginal distribution of $\tilde{\mathbf{v}}_i(\mathbf{s}^D) = (\tilde{v}_i(s_1^D), \dots, \tilde{v}_i(s_n^D))^T$ follows a multivariate normal distribution, expressed as:

$$\tilde{\mathbf{v}}_i(\mathbf{s}^D) | \tau_i^2, \eta_i \sim MN(0, \tau_i^2 (R_i + \eta_i I_n)) \quad (3)$$

Where the correlation matrix R_i has the following entries for $(l, m) : c_i(s_l^D, s_m^D)$, $i = 1, 2, \dots, K$, I_n is the identity matrix for $n \times n$. Using the Matérn kernel to establish a correlation model yields:

$$c_i(d) = \frac{1}{2^{v_i-1} \Gamma(v_i)} \left(\frac{d}{\gamma_i} \right)^{v_i} \kappa_{v_i} \left(\frac{d}{\gamma_i} \right) \quad (4)$$

Where $\Gamma(v)$ is the gamma function, $\kappa_{v_i}(\cdot)$ is a modified Bessel function of the second kind with smoothing parameter, γ_i is the range parameter. For any $s_a, s_b \in \Phi$, there is $d = |s_a - s_b|$.

Let $Y(s^D) = (y(s^D)^T; y^*(s^D)^T)^T$. Following Higdon's method, apply singular value decomposition(SVD) to $Y(s^D) = UDV$. Furthermore, the computational order for estimating A is linearly related to n , and when $i \neq j$, there is $\mathbf{a}_i^T \mathbf{a}_j = 0$. Thus, $A^T A$ becomes a diagonal matrix, and therefore this paper estimates A as $A = \frac{UD}{\sqrt{n}}$.

For convenience, vectorizing the outputs of $\mathbf{Y}_v(s^D)$ and $\tilde{\mathbf{v}}_v(s^D)$ yields $\mathbf{Y}_v(s^D) := \text{vec}(\mathbf{Y}(s^D))$ and $\tilde{\mathbf{v}}_v(s^D) := \text{vec}(\tilde{\mathbf{v}}(s^D)^T)$. Furthermore, defining a matrix $Kn \times Kn$ as $\mathbf{A}_v := [\mathbf{I}_n \otimes \mathbf{a}_1; \dots; \mathbf{I}_n \otimes \mathbf{a}_K]$, $Y(s^D)$ can be expressed as:

$$\mathbf{Y}_v(s^D) = \mathbf{A}_v \tilde{\mathbf{v}}_v(s^D) + \boldsymbol{\varepsilon}_{0v} \quad (5)$$

Among these, $\boldsymbol{\varepsilon}_{0v} \sim MN(0, \sigma_0^2 I_{Kn})$, $(\tilde{\mathbf{v}}_v(s^D) | \tau_1^2, \dots, \tau_K^2, \tilde{\mathbf{R}}_1, \dots, \tilde{\mathbf{R}}_K) \sim MN(0, \Sigma_v)$, and the (l, m) item is defined as $\tilde{c}_i(s_l^D, s_m^D) = c_i(s_l^D, s_m^D) + \eta_i 1_{l=1}$. In addition, $\sum_v = \text{blkdiag}(\tau_1^2 \tilde{\mathbf{R}}_1, \dots, \tau_k^2 \tilde{\mathbf{R}}_k)$, and $\text{blkdiag}(\cdot)$ denotes the block-diagonal matrix associated with the observation locations or time points. Marginalizing directly over $\tilde{\mathbf{v}}_v(s^D)$ yields:

$$\mathbf{Y}_v(s^D) | \sigma_0^2, \tau_1^2, \dots, \tau_K^2, \tilde{\mathbf{R}}_1, \dots, \tilde{\mathbf{R}}_K \sim MN(0, \mathbf{A}_v \Sigma_v \mathbf{A}_v^T + \sigma_0^2 \mathbf{I}_{Kn}) \quad (6)$$

However, directly computing the likelihood function in (6) requires repeated manipulations of the $Kn \times Kn$ covariance matrix $\mathbf{A}_v \Sigma_v \mathbf{A}_v^T + \sigma_0^2 \mathbf{I}_{Kn}$, resulting in computational complexity of $O(Kn^3)$.

Lemma 1: Let $A = UD/\sqrt{n}$, $A(A^T A)^{-1} A^T = I_K$. Then the marginal likelihood function of $Y_v(s^D)$ is:

$$\begin{aligned}
p(Y_v(s^D) | v_v(s^D), \sigma_0^2) &= (2\pi\sigma_0^2)^{-nk/2} \exp \left\{ -\frac{(Y_v(s^D) - A_v v_v(s^D))^T (Y_v(s^D) - A_v v_v(s^D))}{2\sigma_0^2} \right\} \\
&= (2\pi\sigma_0^2)^{-nk/2} \exp \left\{ -\frac{(\hat{v}_v(s^D) - v_v(s^D))^T A_v^T A_v (\hat{v}_v(s^D) - v_v(s^D))}{2\sigma_0^2} \right\}
\end{aligned} \quad (7)$$

Among these, $\hat{v}(s^D) = (A_v^T A_v)^{-1} A_v^T Y_v(s^D)$. Marginalizing over $v_v(s^D)$ yields:

$$\begin{aligned}
p(Y_v(s^D) | \sigma_0^2, \tau_{1:k}^2, \gamma_{1:k}, \eta_{1:k}) &= \int p(Y_v(s^D) | \hat{v}_v(s^D), \sigma_0^2) p(\hat{v}_v(s^D) | \tau_{1:k}^2, \gamma_{1:k}, \eta_{1:k}) d\hat{v}_v(s^D) \\
&= |A_v^T A_v|^{-1/2} (2\pi)^{-nk/2} |\Sigma_v + \sigma_0^2 (A_v^T A_v)^{-1}|^{-1/2} \times \exp \left\{ -\frac{1}{2} \hat{v}_v(s^D)^T (\Sigma_v + \sigma_0^2 (A_v^T A_v)^{-1})^{-1} \hat{v}_v(s^D) \right\} \\
&= |A_v^T A_v|^{-1/2} \prod_{i=1}^K \left\{ (2\pi)^{-n/2} |\sigma_i^2 \tilde{R}_i + \sigma_0^2 (a_i^T a_i)^{-1} \mathbf{I}_n|^{-1/2} \times \exp \left[-\frac{1}{2} \hat{v}_i(s^D)^T (\sigma_i^2 \tilde{R}_i + \sigma_0^2 (a_i^T a_i)^{-1} \mathbf{I}_n)^{-1} \hat{v}_i(s^D) \right] \right\}
\end{aligned} \quad (8)$$

Thus, the marginal likelihood in model (1) can be expressed as the product of K independent multivariate normal distributions:

$$p(\mathbf{Y}(s^D) | \sigma_0^2, \tau_1^2, \dots, \tau_k^2, \tilde{\mathbf{R}}_1, \dots, \tilde{\mathbf{R}}_k) = |\mathbf{A}^T \mathbf{A}_v|^{-1/2} \prod_{i=1}^K p_{MN}(\hat{v}_i(s^D); 0, \tau_i^2 \tilde{\mathbf{R}}_i + \sigma_0^2 (\mathbf{a}_i^T \mathbf{a}_i)^{-1} \mathbf{I}_n) \quad (9)$$

Where $p_{MN}(\cdot; \mu, \Sigma)$ follows a multivariate normal distribution with mean μ and covariance matrix Σ , and $\hat{v}_i(s^D)$ denotes the transpose of the i-th row of the data table:

$$\hat{\mathbf{v}}(s^D) = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{Y}(s^D) \quad (10)$$

Lemma 1 indicates that the marginal likelihood function of model (1) can be expressed as the product of K independent multivariate normal distributions, thereby greatly simplifying the computational burden. Specifically, instead of evaluating the density of a multivariate normal distribution with a covariance matrix of dimension $Kn \times Kn$, the density can be estimated by computing K independent multivariate normal distributions, each with a covariance matrices of dimension $n \times n$, rather than using an independent multivariate normal distribution with a covariance matrix of dimension $Kn \times Kn$. However, when n is large, the computational cost of directly estimating the density via independent multivariate normal distributions with $n \times n$ covariance matrices remains prohibitively high. Suppose:

$$\mathbf{Y}(s_j^*) = (\mathbf{y}(s_j^*)^T; \mathbf{y}^*(s_j^*)^T)^T \quad (11)$$

Among these, $\mathbf{y}(s_j^*)$ and $\mathbf{y}^*(s_j^*)$ denote the j-th row of $\mathbf{Y}(s^*)$ and $\mathbf{Y}^*(s^*)$ respectively.

Lemma 2: Let $A = UD/\sqrt{n}$, $A(A^T A)^{-1} A^T = I_K$, $Y_v(s^D, s_j^*) := \text{vec}(Y(s^D, s_j^*))$ and $v_v(s^D) := \text{vec}(v(s^D)^T)$, where all of these are $k \times (n+1)$ -dimensional matrices. Vectorizing the outputs of $Y_v(s^D, s_j^*)$ and $v_v(s^D)$ yields $Y_v(s^D, s_j^*) := \text{vec}(Y(s^D, s_j^*))$ and $v_v(s^D) := \text{vec}(v(s^D)^T)$, respectively. Accordingly, equation (1) can be written as:

$$Y_v(s^D, s_j^*) = A_v v_v(s^D, s_j^*) + \boldsymbol{\varepsilon} \quad (12)$$

Among these, $\boldsymbol{\varepsilon} \sim MN(0, \sigma_0^2 \mathbf{I}_{(n+1)K})$.

1) For each s_j^* , with $j = 1, 2, \dots, n^*$. Setting $\mathbf{r}_i(s_j^*) = (c_i(s_j^*, s_1^*), \dots, c_i(s_j^*, s_n^*))^T$ yields:

$$\mathbf{Y}(s_j^*) | \mathbf{Y}(s^D), \sigma_0^2, \tau_{1:k}^2, \eta_{1:k}, \boldsymbol{\kappa} \sim MN(\hat{\boldsymbol{\mu}}(s_j^*), \hat{\boldsymbol{\Sigma}}(s_j^*)) \quad (13)$$

Among these , $\hat{\mu}(s_j^*) = \mathbf{A}\hat{\mathbf{v}}(s_j^*)$, and the i-th term of $\hat{\mathbf{v}}(s_j^*)$ is $\hat{v}_i(s_j^*) = \mathbf{r}_i^T(s_j^*)$.

Additionally , for $(\tilde{\mathbf{R}}_i + \frac{\sigma_0^2 (a_i^T a_i)^{-1}}{\tau_i} \mathbf{I}_n)^{-1} \hat{v}_i(s_j^D)$, $\hat{v}_i(s_j^D)$ is the transpose of the i-th term defined in formula(6). Furthermore , $\hat{\Sigma}(s_j^*) = \mathbf{A}D^*(s_j^*)\mathbf{A}^T + \sigma_0^2 \mathbf{I}_K$, where $D^*(s_j^*)$ is a diagonal matrix whose i-th diagonal entry is:

$$\tau_i^2 \left[\hat{c}_i(s_j^*, s_j^*) - \mathbf{r}_i^T(s_j^*) \left(\hat{\mathbf{R}}_i + \frac{\sigma_0^2 (a_i^T (a_i^T a_i)^{-1} \alpha_i)}{\tau_i^2} \mathbf{I}_n \right)^{-1} \mathbf{r}_i(s_j^*) \right], \quad i = 1, 2, \dots, K \quad (14)$$

2) Let the partition be defined as $\hat{\mu}(s_j^*) = (\hat{\mu}_0^T(s_j^*), \hat{\mu}_*^T(s_j^*))^T$ and $\hat{\Sigma}(s_j^*) = \begin{pmatrix} \hat{\Sigma}_{00}(s_j^*) & \hat{\Sigma}_{0*}(s_j^*) \\ \hat{\Sigma}_{*0}(s_j^*) & \hat{\Sigma}_{**}(s_j^*) \end{pmatrix}$. Then, for any s_j^* , the prediction distribution of $y^*(s_j^*)$ is :

$$y^*(s_j^*) | y(s_j^D), y(s_j^*), y^*(s_j^D), \sigma_0^2, \tau_{1:K}^2, \gamma_{1:K}, \eta_{1:K} \sim MN(\hat{\mu}_{*0}(s_j^*), \hat{\Sigma}_{*0}(s_j^*)) \quad (15)$$

Among these , $\hat{\mu}_*(s_j^*)$ corresponds to $\hat{\mu}_{*0}(s_j^*) = \hat{\mu}_*(s_j^*) + \hat{\Sigma}_{*0}(s_j^*) \hat{\Sigma}_{00}^{-1}(s_j^*) (\mathbf{y}(s_j^*) - \hat{\mu}_0(s_j^*))$ while $\hat{\Sigma}_{*0}(s_j^*)$ corresponds to $\hat{\Sigma}_{*0}(s_j^*) = \hat{\Sigma}_{**}(s_j^*) - \hat{\Sigma}_{*0}(s_j^*) \hat{\Sigma}_{00}^{-1}(s_j^*) \hat{\Sigma}_{0*}(s_j^*)$.

2.3. Model Estimation and Computational Strategy

Computing the likelihood function in model (1) requires evaluating the inverse of \mathbf{R}_i , where each inversion incurs a computational complexity cost of $O(n^3)$. Consequently , when $n=K$ is large, the overall computational complexity escalates to $O(Kn^3)$, rendering direct implementation computationally infeasible. Therefore, this paper introduces a computationally efficient algorithm grounded in Gaussian random fields and Gaussian Markov random fields (GMRF), inspired by the work of Hartikainen and Särkkä . Compared with existing approaches, this proposed approach offers the notable advantage of avoiding any approximation of the likelihood function.

Consider a continuous regression model of order p defined by the following stochastic differential equation (SDE):

$$c_p v^{(p)}(s) + c_{p-1} v^{(p-1)}(s) + \dots + c_0 v(s) = b_0 z(s) \quad (16)$$

Where $v^{(l)}(s)$ denotes the l-th derivative coefficient of the process $v(s)$, and $z(s)$ represents a standard Gaussian white noise process defined on the domain $s \in \mathcal{R}$. To avoid identifiability issues , we set $c_p = 1$ without loss of generality. According to the work of Whittle , the spectral density of a Gaussian process equipped with a Matérn kernel is given by:

$$S_{\text{Mat}}(t) \propto \frac{1}{(\lambda^2 + t^2)^{(\nu+1/2)}} \quad (17)$$

Among these, $\lambda = \frac{\sqrt{2\nu}}{\gamma}$, γ is the coverage parameters , and ν is the smoothing parameter.

2.4. Computation of Continuous-Time Stochastic Processes

By Lemma 1, the likelihood of $Y(s^D)$ in model(1) is proportional to the product of K-multivariate normal distributions $\hat{v}_i(s^D)$. Here, $\hat{v}_i(s^D)$ denotes the transpose of the i-th row of $\hat{v}(s^D) = (A^T A)^{-1} A^T Y(s^D)$. Furthermore, due to the identifiable row property, we focus on $\hat{v}(s^D)$ when $v_0^2 = 0$.

Choose The Matérn kernel function when $\nu = 2/5$ as the latent factor process:

$$c(d) = \left(1 + \frac{\sqrt{5}d}{\gamma} + \frac{5d^2}{3\gamma^2} \right) \exp\left(-\frac{\sqrt{5}d}{\gamma} \right) \quad (18)$$

Define $\theta_i(s) := (v_i(s), v_i^{(1)}(s), v_i^{(2)}(s))^T$, and for $\forall s \in \varphi$, each $v_i^{(l)}(s)$ corresponds to the l-th order derivative (with $l=1,2$) of the underlying latent process with respect to s .

Therefore, for any $s \in \varphi$, equation (2) can be expressed as:

$$\begin{aligned} \hat{v}_i(s) &= v_i(s) + \varepsilon_i \\ v_i(\cdot) &\sim \text{GP}(0, \tau_i^2 c_i(\cdot, \cdot)) \end{aligned} \quad (19)$$

Combining the Gaussian process model in equation (21) with the correlation structure implied by the stochastic differential equation (SDE) in equation (18) yields the following unified representation:

$$\frac{d\theta_i(s)}{ds} = J_i \theta_i(s) + L z_i(s) \quad (20)$$

Among these, $z_i(s)$ is a Gaussian white noise process with mean = 0 and variance = τ_i^2 . And,

$$J_i = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -\lambda_i^3 & -\lambda_i^2 & -3\lambda_i \end{bmatrix}, \quad \lambda_i = \sqrt{2\nu_i} / \gamma_i, i = 1, 2, \dots, K, \quad L = (0, 0, 1)^T.$$

For $j = 1, \dots, N-1$, let $d_j = |s_{j+1} - s_j|$. Then the expression for the solution of the SDE in equation (20) is:

$$G_i(s_j) = e^{\lambda_i d_j} = \frac{e^{-\lambda_i d_j}}{2} \begin{bmatrix} \lambda_i^2 d_j^2 + 2\lambda_i + 2 & 2(\lambda_i^2 d_j^2 + d_j) & d_j^2 \\ -\lambda_i^3 d_j^2 & -2(\lambda_i^3 d_j^2 - \lambda_i d_j - 1) & 2d_j - \lambda_i d_j^2 \\ \lambda_i^4 - 2\lambda_i^3 d_j & 2(\lambda_i^3 d_j^2 - 3\lambda_i^2 d_j) & \lambda_i^2 d_j^2 - 4\lambda_i d_j + 2 \end{bmatrix}$$

$$W_i(s_j) = \frac{4\sigma^2 \lambda_i^5}{3} \begin{bmatrix} W_{1i}(s_j) & W_{2i}(s_j) & W_{3i}(s_j) \\ W_{4i}(s_j) & W_{5i}(s_j) & W_{6i}(s_j) \\ W_{7i}(s_j) & W_{8i}(s_j) & W_{9i}(s_j) \end{bmatrix}$$

Among these:

$$W_{1i}(s_j) = \frac{e^{-2\lambda_i d_j} (3 + 6\lambda_i d_j + 6\lambda_i^2 d_j^2 + 4\lambda_i^3 d_j^3 + 2\lambda_i^4 d_j^4) - 3}{-4\lambda_i^5}$$

$$W_{2i}(s_j) = W_{4i}(s_j) = \frac{e^{-2\lambda_i d_j} t^4}{2}$$

$$W_{3i}(s_j) = W_{7i}(s_j) = \frac{e^{-2\lambda_i d_j} (1 + 2\lambda_i d_j + 2\lambda_i^2 d_j^2 + 4\lambda_i^3 d_j^3 - 2\lambda_i^4 d_j^4) - 1}{4\lambda_i^3}$$

$$W_{5i}(s_j) = \frac{e^{-2\lambda_i d_j} (1 + 2\lambda_i d_j + 2\lambda_i^2 d_j^2 - 4\lambda_i^3 d_j^3 + 2\lambda_i^4 d_j^4) - 1}{-4\lambda_i^3}$$

$$W_{6i}(s_j) = W_{8i}(s_j) = \frac{e^{-2\lambda_i d_j} d_j^2 (4 - 4\lambda_i d_j + \lambda_i^2 d_j^2)}{2}$$

$$W_i(s_0) = \begin{bmatrix} \sigma_i^2 & 1 & -\sigma_i^2 \lambda_i^2 / 3 \\ 0 & \sigma_i^2 \lambda_i^2 / 3 & 1 \\ -\sigma_i^2 \lambda_i^2 / 3 & 0 & \sigma_i^2 \lambda_i^4 \end{bmatrix}$$

For $j = 1, \dots, N$, the joint distribution of $\theta_i(s_{0:n})$ is:

$$\begin{bmatrix} \theta_i(s_0) \\ \theta_i(s_1) \\ \theta_i(s_2) \\ \vdots \\ \theta_i(s_N) \end{bmatrix} \sim MN \left(\begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} W_i(s_0)^{-1} & -G_i^T(s_1)W_i(s_1)^{-1}G_i(s_1) & \dots & \dots \\ -G_i^T(s_1)W_i(s_1)^{-1}G_i(s_1) & W_i^{-1}(s_1) & \dots & \dots \\ \vdots & -G_i^T(s_2)W_i(s_2)^{-1}G_i(s_2) & \dots & \dots \\ \vdots & \dots & \dots & W_i^{-1}(s_N) \end{bmatrix}^{-1} \right)$$

Let $q_i = \frac{16}{3} \sigma_i^2 \lambda_i^5$, $F = (1, 0, 0)$. Then the SDE solution for model (2) can be expressed as:

$$\begin{aligned} \tilde{v}_{i(s_{j+1})} &= F\theta_i(s+1) + \varepsilon_i \\ \theta_i(s_j+1) &= G_i(s_j)\theta_i(s_j) + w_i(s_j) \end{aligned} \quad (21)$$

Among these, $G_i(s_j) = e^{J_i(s_{j+1}-s_j)}$, $w_i(s_j) \sim MN(0, W_i(s_j))$ ($i = 1, \dots, K$, $j = 1, \dots, N-1$) and $W_i(s_j) = \int_0^{s_{j+1}-s_j} e^{J_i t} L q_i L^T e^{J_i^T t} dt$. Thus, the stationary distribution at θ_i is:

$$\theta_i(s_0) \sim MN(0, W_i(s_0)) \quad (22)$$

Among these, $W_i(s_0) = \int_0^\infty e^{J_i t} L q_i L^T e^{J_i^T t} dt$.

Lemma 3: (Kalman Filter) Consider the continuous-time state-space model specified by equation (21), assume all distributions in this lemma are conditioned on $(\sigma_i^2, \tau_i^2, \gamma_i)$. Furthermore, for any $j \geq 1$ and $i = 1, 2, \dots, K$, there is:

$$\theta_i(s_{j-1}^D) | \hat{v}_i(s_{1:j-1}^D) \sim MN(m_i(s_{j-1}^D), C_i(s_{j-1}^D)) \quad (23)$$

(1) Given $\hat{v}_i(s_{1:j-1}^D)$, the one-step-ahead prediction distribution of $\theta_i(s_j^D)$ is:

$$\theta_i(s_j^D) | \hat{v}_i(s_{1:j-1}^D) \sim MN(\mu_{\theta_i}(s_j^D), \theta_i(s_j^D)) \quad (24)$$

Among these, $\mu_{\theta_i}(s_j^D) = E\theta_i(s_j^D | \hat{v}_i(s_{1:j-1}^D)) = G_i(s_j^D)m_i(s_{j-1}^D)$, $\theta_i(s_j^D) = \text{var}(\theta_i(s_j^D) | \hat{v}_i(s_{1:j-1}^D)) = G_i(s_j^D)C_i(s_{j-1}^D)G_i^T(s_j^D) + W_i(s_j^D)$.

(2) Given $\hat{v}_i(s_j^D)$, the one-step-ahead prediction distribution of $\hat{v}_i(s_j^D)$ is:

$$\hat{v}_i(s_j^D) | \hat{v}_i(s_{1:j-1}^D) \sim N(f_i(s_j^D), Q_i(s_j^D)) \quad (25)$$

(3) Given $\hat{v}_i(s_{1:j}^D)$, the filtering distribution of $\theta_i(s_j^D)$ is:

$$\theta_i(s_j^D) | \hat{v}_i(s_{1:j}^D) \sim MN(m_i(s_j^D), C_i(s_j^D)) \quad (26)$$

Among these, $m_i(s_j^D) = E[\theta_i(s_j^D) | \hat{v}_i(s_{1:j}^D)] = \mu_{\theta_i}(s_j^D) + \Sigma_{\theta_i}(s_j^D)F^T Q_i^{-1}(s_j^D)e_i(s_j^D)$, $C_i(s_j^D) = \text{var}[\theta_i(s_j^D) | \hat{v}_i(s_{1:j}^D)] = \Sigma_{\theta_i}(s_j^D) - \Sigma_{\theta_i}(s_j^D)F^T Q_i^{-1}(s_j^D)F \Sigma_{\theta_i}(s_j^D)$. The prediction error is $e_i(s_j^D) = \hat{v}_i(s_j^D) - f_i(s_j^D)$.

To compute the likelihood function, the Kalman filter on s^D is defined in Lemma3, the estimate of $(\sigma_i^2, \tau_i^2, \gamma_i)$ when $i=1,2,\dots,K$ is given, then the Kalman filtering perform on all observation locations or time points $S_{1:N}^*$ (including s^*).

Lemma 4: (Kalman smoother) Consider the continuous-time state-space model specified by equation (21). For simplicity, all distributions in this lemma are assumed to be conditioned on $(\sigma_i^2, \tau_i^2, \gamma_i)$. Furthermore, for any $i = 1, 2, \dots, K$ and $j = 1, 2, \dots, N-1$, let $\theta_i(s_{j+1}) | \hat{v}_i(s^D) \sim N(\mu_{\theta_i}^*(s_{j+1}), \Sigma_{\theta_i}^*(s_{j+1}))$, then:

$$\theta_i(s_j) | \hat{v}_i(s_j^D) \sim N(\mu_{\theta_i}^*(s_j), \Sigma_{\theta_i}^*(s_j)) \quad (27)$$

Among these, $\mu_{\theta_i}^*(s_j) = \mathbf{m}_i(s_j) + C_i(s_j)G_i^T(s_{j+1})\Sigma_{\theta_i}^{-1}(s_{j+1})(\mu_{\theta_i}^*(s_{j+1}) - \mu_{\theta_i}(s_j))$, $\Sigma_{\theta_i}^*(s_j) = C_i(s_j) - C_i(s_j)G_i^T(s_{j+1})\Sigma_{\theta_i}^{-1}(s_{j+1})(\Sigma_{\theta_i}(s_{j+1}) - \Sigma_{\theta_i}^*(s_{j+1}))\Sigma_{\theta_i}^{-1}(s_{j+1})G_i(s_{j+1})C_i(s_j)$.

Based on the above settings, the posterior distribution of $\theta_i(s) (i=1,2,\dots,K)$ can be efficiently computed via the forward filtering and backward smoothing (FFBS) algorithm, requiring only $O(n)$ operations—substantially reducing the computational burden compared with $O(n^3)$ operations cost of directly evaluating of the Gaussian process model. Predictions at s^* are computed linearly with the number of observed locations or time points, resulting in an overall computational complexity of only $O(n)$.

It is worth noting that when $\nu_i = 2/5$, both the Matérn covariance matrix and its inverse in equation(18) are rank-n matrices of size $n \times n$. However, the covariance matrix $\theta_i(s) = (\mathbf{v}_i(s), \mathbf{v}_i^{(1)}(s), \mathbf{v}_i^{(2)}(s))^T$ of the latent states is sparse. Therefore, the result holds for all Matérn kernels when $\nu_i = (2m+1)/2(m \in N)$.

2.5. Parameter Estimation

The computationally intensive part of the FFBS algorithm involves sampling the latent states and marginalizing them:

$$p(\hat{v}(s^D) | \sigma_{1:K}^2, \tau_{1:K}^2, \gamma_{1:K}) = \sum_{i=1}^K \left\{ p(\hat{v}_i(s_1^D) | \sigma_i^2, \tau_i^2, \gamma_i) \prod_{j=2}^n p(\hat{v}_i(s_j^D) | \hat{v}_i(s_{j-1}^D), \sigma_i^2, \tau_i^2, \gamma_i) \right\} \quad (28)$$

Among these, each term follows the normal distribution specified in the one-step-ahead prediction of the FFSE algorithm.

Lemma 5: (Likelihood by the Kalman filter) For each $i = 1, 2, \dots, K$, there is:

$$p(\hat{v}_i(s_{1:j}^D) | \sigma_i^2, \tau_i^2, \gamma_i) = (2\pi)^{-n/2} \prod_{j=1}^N Q_i^{-1/2}(s_j^D) \exp \left\{ -\sum_{j=1}^N \frac{(\hat{v}_i(s_j^D) - f_i(s_j^D))^2}{Q_i(s_j^D)} \right\} \quad (29)$$

Among these, $f_i(s_j^D)$ and $Q_i(s_j^D)$ are defined as in lemma 3. To complete the model, assume the following prior:

$$\pi(\sigma_{1:K}^2, \tau_{1:K}^2, \gamma_{1:K}) \propto \prod_{i=1}^K \left(\frac{\pi(\tau_i^2, \gamma_i)}{\sigma_i^2} \right) \quad (30)$$

Let $\zeta_i := 1/\gamma_i$ and $\eta_i := \tau_i/\sigma_i^2$. For $i = 1, 2, \dots, K$, assume the joint robust prior for the transformation parameters is: $\pi(\zeta_i, \eta_i) \propto (C_i \zeta_i + \eta_i)^{a_i} \exp(-b_i(C_i \zeta_i + \eta_i))$. Here, a_i , b_i and C_i are prior parameters. If set $a_i = 1/2$, $b_i = 1$ and $C_i = |\varphi|n$, where $|\varphi|$ is the length of φ . When n is large, sampling the posterior distribution via MCMC remains slow. Parameters can be estimated using the posterior mode:

$$(\hat{\zeta}_i, \hat{\eta}_i) = \arg \max_{\zeta_i, \eta_i} \left\{ p(\hat{v}(s^D) | \zeta_i, \eta_i) \pi(\zeta_i, \eta_i) \right\} \quad (31)$$

By applying Lemma 5, the computation can be simplified to $O(n)$ operations. The NSGP model construction flowchart is shown in Figure 1.

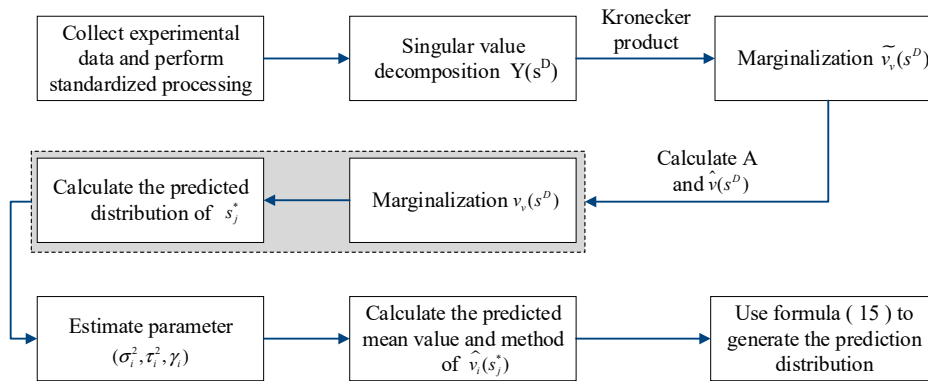


Figure 1. NSGP Model Construction Flowchart.

3. Multivariate and Multidimensional Quality Profit and Loss Function Based on Non-separable Gaussian Process

3.1. Signal-to-Noise Ratio

The quality characteristic value of a product is a random variable influenced by multiple factors. Assuming the mathematical expectation is μ and the variance is σ^2 . For nominal the best, the closer μ is to the target value T , the better; the smaller σ^2 is, the better. In robust design, let the coefficient of variation be y and the signal-to-noise ratio be η , then:

$$\gamma = \frac{\sigma}{\mu} \quad (32)$$

$$\eta = \frac{\mu^2}{\sigma^2} \quad (33)$$

The signal-to-noise ratio η_T for nominal the best is:

$$\eta_T = 10 \log \frac{\mu^2}{\sigma^2} \quad (34)$$

Setting $\bar{y} = \frac{1}{n} \sum_{i=1}^n y_i$, $S^2 = \frac{1}{n-1} \sum_{i=1}^n (y_i - \bar{y})^2$, the estimated value $\hat{\eta}_T$ for η_T is:

$$\hat{\eta}_T = 10 \log \left(\frac{\bar{y}^2}{S^2} \right) \quad (35)$$

Similarly, larger the better signal-to-noise ratio η_L and its estimated value $\hat{\eta}_L$ are:

$$\eta_L = 10 \log (\sigma^2 + \mu^2) \quad (36)$$

$$\hat{\eta}_L = -10 \log \left(\frac{1}{n} \sum_{i=1}^n \frac{1}{y_i^2} \right) \quad (37)$$

Smaller the better signal-to-noise ratio η_s and its estimate $\hat{\eta}_s$ are:

$$\eta_s = 10 \log \frac{1}{\mu^2 + \sigma^2} \quad (38)$$

$$\hat{\eta}_s = -10 \log \left(\frac{1}{n} \sum_{i=1}^n y_i^2 \right) \quad (39)$$

The signal-to-noise ratio is a monotonic function of the coefficient of variation γ , serving as a crucial indicator of product quality stability. A higher η value indicates greater product stability and reduced losses.

Both the coefficient of variation γ and the signal-to-noise ratio η are dimensionless data. They both simultaneously consider the effects of mean and variance on quality stability. However, η is closer to a normal distribution, enabling linear additivity between quality characteristics and allowing interaction effects to be neglected.

3.2. Quality Gain-Loss Function Model

Leveraging the advantages of Taguchi's signal-to-noise ratio method and Pignatiello's multivariate quadratic loss function approach in robust parameter design, a multivariate quality gain-loss function suitable for spatiotemporal responses is constructed.

Based on the signal-to-noise ratio method, the loss weights α_i and β_i for each spatial factor and temporal factor level under the quality characteristic y_i are calculated as:

$$\alpha_i = \frac{1}{\sum_{l=1}^K \frac{1}{\eta_l}}, i = 1, 2, \dots, K \quad (40)$$

$$\beta_j = \frac{1}{\sum_{m=1}^N \frac{1}{\eta_m}}, j = 1, 2, \dots, N \quad (41)$$

Among these, α_i and β_j represent the signal-to-noise ratios for the quality characteristics located in row i and column j respectively. Since a higher signal-to-noise ratio indicates greater stability in the quality characteristic and consequently smaller quality losses, α_i and β_j can respectively determine the contribution levels of the spatial factor i and the temporal factor j to the fluctuations.

Construct the joint weight matrix using gain-loss weights:

$$\omega = (\alpha_1, \alpha_2, \dots, \alpha_K)^T \times (\beta_1, \beta_2, \dots, \beta_N) = \begin{pmatrix} \alpha_1\beta_1 & \alpha_1\beta_2 & \dots & \alpha_1\beta_N \\ \alpha_2\beta_1 & \alpha_2\beta_2 & \dots & \alpha_2\beta_N \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_K\beta_1 & \alpha_K\beta_2 & \dots & \alpha_K\beta_N \end{pmatrix}_{K \times N} \quad (42)$$

Among these, the element (i,j) of ω , denoted as $\omega_{ij} = \alpha_i\beta_j$, represents the joint loss weight under the combined effect of the i th spatial factor level and the j th temporal factor level.

The multi-dimensional quality gain-loss functions for larger the better ($G_{GL}(y(x))$), smaller the better ($G_{GS}(y(x))$), and nominal the best ($G_{GT}(y(x))$) quality characteristics are respectively:

$$\begin{aligned} G_{GL}(y(x)) &= \begin{bmatrix} k_1 y_1(x)^{-1} \\ k_2 y_2(x)^{-1} \\ \vdots \\ k_K y_K(x)^{-1} \end{bmatrix}^T \begin{pmatrix} \alpha_1\beta_1 & \alpha_1\beta_2 & \dots & \alpha_1\beta_N \\ \alpha_2\beta_1 & \alpha_2\beta_2 & \dots & \alpha_2\beta_N \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_K\beta_1 & \alpha_K\beta_2 & \dots & \alpha_K\beta_N \end{pmatrix} \begin{bmatrix} y_1(x)^{-1} \\ y_2(x)^{-1} \\ \vdots \\ y_K(x)^{-1} \end{bmatrix} + \begin{bmatrix} g(y_1(x_2)^{-1}) \\ g(y_2(x_2)^{-1}) \\ \vdots \\ g(y_K(x_2)^{-1}) \end{bmatrix}^T \begin{pmatrix} \alpha_1\beta_1 & \alpha_1\beta_2 & \dots & \alpha_1\beta_N \\ \alpha_2\beta_1 & \alpha_2\beta_2 & \dots & \alpha_2\beta_N \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_K\beta_1 & \alpha_K\beta_2 & \dots & \alpha_K\beta_N \end{pmatrix} \\ &= \begin{bmatrix} k_1 y_1(x)^{-1} \\ k_2 y_2(x)^{-1} \\ \vdots \\ k_K y_K(x)^{-1} \end{bmatrix}^T \begin{pmatrix} \alpha_1\beta_1 & \alpha_1\beta_2 & \dots & \alpha_1\beta_N \\ \alpha_2\beta_1 & \alpha_2\beta_2 & \dots & \alpha_2\beta_N \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_K\beta_1 & \alpha_K\beta_2 & \dots & \alpha_K\beta_N \end{pmatrix} \begin{bmatrix} y_1(x)^{-1} \\ y_2(x)^{-1} \\ \vdots \\ y_K(x)^{-1} \end{bmatrix} \\ &= \begin{bmatrix} k_1 y_1(x)^{-1} \\ k_2 y_2(x)^{-1} \\ \vdots \\ k_K y_K(x)^{-1} \end{bmatrix}^T \begin{pmatrix} \alpha_1\beta_1 & \alpha_1\beta_2 & \dots & \alpha_1\beta_N \\ \alpha_2\beta_1 & \alpha_2\beta_2 & \dots & \alpha_2\beta_N \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_K\beta_1 & \alpha_K\beta_2 & \dots & \alpha_K\beta_N \end{pmatrix} \begin{bmatrix} y_1(x)^{-1} \\ y_2(x)^{-1} \\ \vdots \\ y_K(x)^{-1} \end{bmatrix} \\ &= \sum_{i=1}^K \sum_{j=1}^N \omega_{ij} [k_i (y_i(x_j)^{-1})^2 + g(y_i(x_j)^{-1})] \end{aligned} \quad (43)$$

$$\begin{aligned}
G_{GS}(y(x)) &= \begin{bmatrix} k_1 y_1(x) \\ k_2 y_2(x) \\ \vdots \\ k_K y_K(x) \end{bmatrix}^T \begin{pmatrix} \alpha_1 \beta_1 & \alpha_1 \beta_2 & \cdots & \alpha_1 \beta_N \\ \alpha_2 \beta_1 & \alpha_2 \beta_2 & \cdots & \alpha_2 \beta_N \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_K \beta_1 & \alpha_K \beta_2 & \cdots & \alpha_K \beta_N \end{pmatrix} \begin{bmatrix} y_1(x) \\ y_2(x) \\ \vdots \\ y_K(x) \end{bmatrix} + \begin{bmatrix} g(y_1(x_N)) \\ g(y_2(x_N)) \\ \vdots \\ g(y_K(x_N)) \end{bmatrix}^T \begin{pmatrix} \alpha_1 \beta_1 & \alpha_1 \beta_2 & \cdots & \alpha_1 \beta_N \\ \alpha_2 \beta_1 & \alpha_2 \beta_2 & \cdots & \alpha_2 \beta_N \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_K \beta_1 & \alpha_K \beta_2 & \cdots & \alpha_K \beta_N \end{pmatrix} \\
&= \begin{bmatrix} k_1 y_1(x) - T_1 \\ k_2 y_2(x) - T_2 \\ \vdots \\ k_K y_K(x) - T_K \end{bmatrix}^T \begin{pmatrix} \alpha_1 \beta_1 & \alpha_1 \beta_2 & \cdots & \alpha_1 \beta_N \\ \alpha_2 \beta_1 & \alpha_2 \beta_2 & \cdots & \alpha_2 \beta_N \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_K \beta_1 & \alpha_K \beta_2 & \cdots & \alpha_K \beta_N \end{pmatrix} \begin{bmatrix} y_1(x) - T_1 \\ y_2(x) - T_2 \\ \vdots \\ y_K(x) - T_K \end{bmatrix} \\
&+ \begin{bmatrix} g(y_1(x_1)) \\ g(y_2(x_1)) \\ \vdots \\ g(y_K(x_1)) \end{bmatrix}^T \begin{bmatrix} \alpha_1 \beta_1 \\ \alpha_2 \beta_1 \\ \vdots \\ \alpha_K \beta_1 \end{bmatrix} + \begin{bmatrix} g(y_1(x_2)) \\ g(y_2(x_2)) \\ \vdots \\ g(y_K(x_2)) \end{bmatrix}^T \begin{bmatrix} \alpha_1 \beta_2 \\ \alpha_2 \beta_2 \\ \vdots \\ \alpha_K \beta_2 \end{bmatrix} + \cdots + \begin{bmatrix} g(y_1(x_N)) \\ g(y_2(x_N)) \\ \vdots \\ g(y_K(x_N)) \end{bmatrix}^T \begin{bmatrix} \alpha_1 \beta_N \\ \alpha_2 \beta_N \\ \vdots \\ \alpha_K \beta_N \end{bmatrix} \\
&= \sum_{i=1}^K \sum_{j=1}^N \omega_{ij} [k_i (y_i(x_j))^2 + g(y_i(x_j))]
\end{aligned} \tag{44}$$

$$\begin{aligned}
G_{GT}(y(x)) &= \begin{bmatrix} k_1 y_1(x) - T_1 \\ k_2 y_2(x) - T_2 \\ \vdots \\ k_K y_K(x) - T_K \end{bmatrix}^T \begin{pmatrix} \alpha_1 \beta_1 & \alpha_1 \beta_2 & \cdots & \alpha_1 \beta_N \\ \alpha_2 \beta_1 & \alpha_2 \beta_2 & \cdots & \alpha_2 \beta_N \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_K \beta_1 & \alpha_K \beta_2 & \cdots & \alpha_K \beta_N \end{pmatrix} \begin{bmatrix} y_1(x) - T_1 \\ y_2(x) - T_2 \\ \vdots \\ y_K(x) - T_K \end{bmatrix} + \begin{bmatrix} g(y_1(x_N) - T_1) \\ g(y_2(x_N) - T_2) \\ \vdots \\ g(y_K(x_N) - T_K) \end{bmatrix}^T \begin{pmatrix} \alpha_1 \beta_1 & \alpha_1 \beta_2 & \cdots & \alpha_1 \beta_N \\ \alpha_2 \beta_1 & \alpha_2 \beta_2 & \cdots & \alpha_2 \beta_N \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_K \beta_1 & \alpha_K \beta_2 & \cdots & \alpha_K \beta_N \end{pmatrix} \\
&= \begin{bmatrix} k_1 y_1(x) - T_1 \\ k_2 y_2(x) - T_2 \\ \vdots \\ k_K y_K(x) - T_K \end{bmatrix}^T \begin{pmatrix} \alpha_1 \beta_1 & \alpha_1 \beta_2 & \cdots & \alpha_1 \beta_N \\ \alpha_2 \beta_1 & \alpha_2 \beta_2 & \cdots & \alpha_2 \beta_N \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_K \beta_1 & \alpha_K \beta_2 & \cdots & \alpha_K \beta_N \end{pmatrix} \begin{bmatrix} y_1(x) - T_1 \\ y_2(x) - T_2 \\ \vdots \\ y_K(x) - T_K \end{bmatrix} \\
&+ \begin{bmatrix} g(y_1(x_1) - T_1) \\ g(y_2(x_1) - T_2) \\ \vdots \\ g(y_K(x_1) - T_K) \end{bmatrix}^T \begin{bmatrix} \alpha_1 \beta_1 \\ \alpha_2 \beta_1 \\ \vdots \\ \alpha_K \beta_1 \end{bmatrix} + \begin{bmatrix} g(y_1(x_2) - T_1) \\ g(y_2(x_2) - T_2) \\ \vdots \\ g(y_K(x_2) - T_K) \end{bmatrix}^T \begin{bmatrix} \alpha_1 \beta_2 \\ \alpha_2 \beta_2 \\ \vdots \\ \alpha_K \beta_2 \end{bmatrix} + \cdots + \begin{bmatrix} g(y_1(x_N) - T_1) \\ g(y_2(x_N) - T_2) \\ \vdots \\ g(y_K(x_N) - T_K) \end{bmatrix}^T \begin{bmatrix} \alpha_1 \beta_N \\ \alpha_2 \beta_N \\ \vdots \\ \alpha_K \beta_N \end{bmatrix} \\
&= \sum_{i=1}^K \sum_{j=1}^N \omega_{ij} [k_i (y_i(x_j) - T_i)^2 + g(y_i(x_j) - T_i)]
\end{aligned} \tag{45}$$

Let:

$$G(y_i(x_j)) = \begin{cases} k_i (y_i(x_j) - T_i)^2 + g(y_i(x_j) - T_i) & NTB \\ k_i (y_i(x_j) - T_i)^{-1} + g(y_i(x_j) - T_i) & LTB \\ k_i (y_i(x_j))^2 + g(y_i(x_j)) & STB \end{cases} \tag{46}$$

Then Equations (43) to (45) can be unified into the following form:

$$G(y(x)) = \sum_{i=1}^K \sum_{j=1}^N \omega_{ij} G(y_i(x_j)) \tag{47}$$

Among these, $y_i(x_j)$ represents the observed value of quality characteristic i at time point j for sample y_i . ω_{ij} and $G(y_i(x_j))$ denote the joint gain-loss weight and the gain-loss function, respectively, under the combined influence of spatial factor i and time factor j .

3.3. Two-Stage Parameter Optimization

3.3.1. Time Factor Optimization

Using the NSGP model and the quality gain-loss function, the spatiotemporal response model is constructed as:

$$\arg \min G(\mathbf{y}_{1:K}(\mathbf{s}_{1:N})) = \sum_{i=1}^K \sum_{j=1}^N \omega_{ij} G(y_i(s_j)) \quad (48)$$

A nonlinear optimization algorithm is employed to find the globally optimal parameter design value $\mathbf{s}^{\text{opt}} = (s_1^{\text{opt}}, s_2^{\text{opt}}, \dots, s_N^{\text{opt}})$ for the time factor s in equation(48). The FFBS algorithm calculates the optimal solution \mathbf{s}^{opt} corresponding to the overall predicted mean.

3.3.2. Spatial Factor Optimization

Use the overall prediction mean from the first stage as the observation value for this section. Let z_j denote the mass characteristic corresponding to the j th position/time point, and β_j denote the gain/loss weight for mass characteristic z_j . Then there is:

$$G(z_j(\mathbf{x})) = \begin{cases} k_i(z_j(x) - T_j)^2 + g(z_j(x) - T_i) & NTB \\ k_i(z_j(x))^{-2} + g(z_j(x)^{-1}) & LTB \\ k_i(z_j(x))^2 + g(z_j(x)) & STB \end{cases} \quad (49)$$

Employing multi-response robust parameter modeling and its design methodology, construct the multi-response optimization model:

$$\arg \min G(z(x)) = \sum_{j=1}^N \beta_j G(z_j(x)) \quad (50)$$

Among these, β_j and $G(z_j(x))$ represent the quality gain/loss weight and gain/loss function for the j th quality characteristic z_j , respectively. Employ a nonlinear optimization algorithm to find the globally optimal parameter design values $\mathbf{x}^{\text{opt}} = (x_1^{\text{opt}}, x_2^{\text{opt}}, \dots, x_N^{\text{opt}})$ for the space factor x in equation (50).

By integrating the optimal solutions from equations (48) and (50), the combined optimal parameter combination for the temporal and spatial factors is obtained: $(\mathbf{x}^{\text{opt}}, \mathbf{s}^{\text{opt}})$.

The flowchart integrating NSGP with the multi-objective, multi-dimensional quality gain-loss function model is shown in Figure 2.

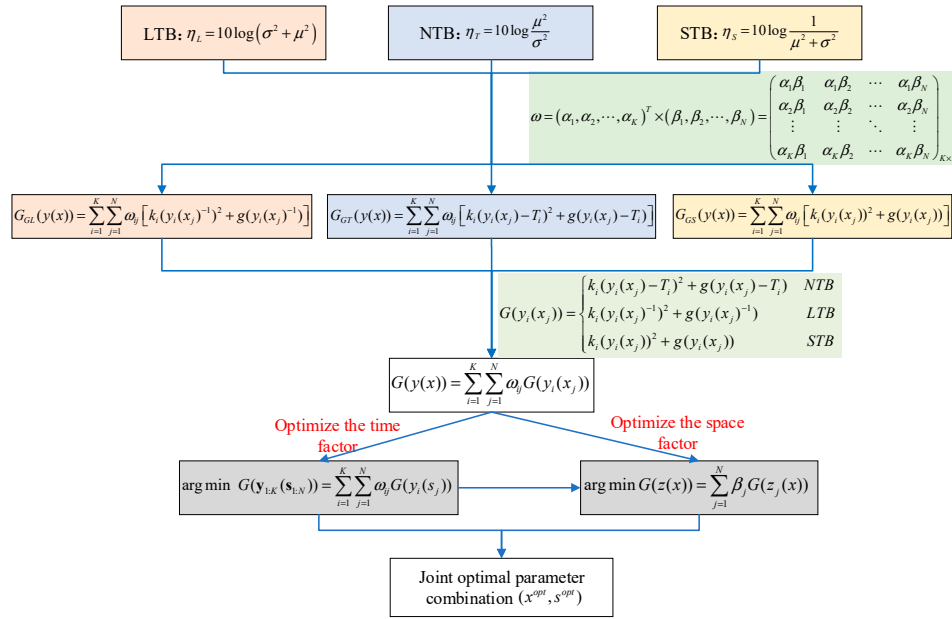


Figure 2. Flowchart of NSGP Combined with Multivariate Multidimensional Quality Gain-Loss Function Model.

4. Simulation and Case Studies

4.1. Prediction Indicators

This paper employs simulation cases and concrete strength prediction examples to demonstrate the superiority of the proposed method. The following models were selected for comparison: univariate Gaussian process (GP), separable Gaussian process (SGP), two linear regression (LR) strategies (sample-based and time-point-based), two random decision forests (RF) strategies (sample-based and time-point-based), and nearest neighbor (NN) method. The predictive performance of these models was evaluated using the following metrics:

(1) Root mean square error:

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{k^*} \sum_{j=1}^{n^*} (y_i^*(s_j^*) - \hat{y}_i^*(s_j^*))^2}{k^* n^*}} \quad (51)$$

(2) 95% posterior confidence interval coverage rate:

$$P_{CI}(95\%) = \frac{1}{k^* n^*} \sum_{i=1}^{k^*} \sum_{j=1}^{n^*} 1\{y_i^*(s_j^*) \in \text{CL}_{ij}(95\%)\} \quad (52)$$

(3) 95% posterior confidence interval length:

$$L_{CI}(95\%) = \frac{1}{k^* n^*} \sum_{i=1}^{k^*} \sum_{j=1}^{n^*} \text{length}\{\text{CI}_{ij}(95\%)\} \quad (53)$$

(4) Multivariate quality gain-loss function:

$$G(y^*(s^*)) = \sum_{i=1}^{k^*} \sum_{j=1}^{n^*} \omega_{ij} G(y_i^*(s_j^*)) \quad (54)$$

For $1 \leq i \leq k^*$ and $1 \leq j \leq n^*$, $y_i^*(s_j^*)$ is the observation value of the s_j^* th sample at the j th position/timestamp s_j^* , $\hat{y}_i^*(s_j^*)$ is the observation value of the i th sample at the j th position/timestamp s_j^* , and ω_{ij} is the joint gain-loss weight of the i th sample at the j th position/timestamp s_j^* . $CI_{ij}(95\%)$ represents the 95% posterior confidence interval for the predicted mean.

4.2. Simulation Case

4.2.1. Simulation Function

In many complex engineering and scientific problems, the optimal response of a system —such as the maximum strength of concrete or the highest yield of crops—is often not a constant value but is concentrated within a specific region that dynamically varies with spatiotemporal factors. To capture the correlation between dynamic peak characteristics and process parameters, shape parameters and time parameters, this paper constructs a simulation function based on the Space-time Gaussian Pulse function, with the specific form as follows:

$$f(x_1, x_2, t) = [8e^{-0.05(x_1 - x_{10})^2}] e^{\frac{-(t-t_0)^2}{2w_t^2} + \frac{-(x_2 - x_{20})^2}{2w_{x_2}^2}} + 0.02t - 0.01x_2 + \varepsilon \quad (55)$$

Assume $x_1 \in [5, 10]$ is the process parameter, $x_2 \in [40, 100]$ is the shape parameter with $x_2 \sim N(x_{20}, w_{x_2}^2)$, $t \in [120, 180]$ is the time parameter with $t \sim N(t_0, w_t^2)$, and the random noise is $\varepsilon \sim N(0, 0.0001^2)$. Using a maximum-minimum Latin hypercube design, 60 spatial samples of x_1 and x_2 were randomly generated on $[5, 10] \times [40, 100]$. Additionally, 600 temporal observation points of t were randomly generated on $[120, 180]$. Different values of (x_1, x_2) and t were randomly combined to generate spatio-temporal data of size 60×600 . For this data, inference based on the Gaussian process model involves the inverse operation of a 600×600 -dimensional covariance matrix, resulting in an overall computational complexity of $O((60 \times 600)^3)$. Therefore, this simulation data demonstrates the feasibility and effectiveness of the proposed method.

To investigate the empirical correlation of output levels in both sample and time dimensions within the simulation data, R language was employed to generate and visualize the data: (a) correlation of output levels across samples; (b) empirical correlation of output levels aggregated by time intervals. Figure 3 reveals that, in the spatial dimension, output levels of the same spatial parameter are correlated with each other, and output levels between adjacent spatial parameters are highly correlated. In the temporal dimension, output levels at adjacent time points are correlated with each other, and this correlation weakens as the time interval increases.

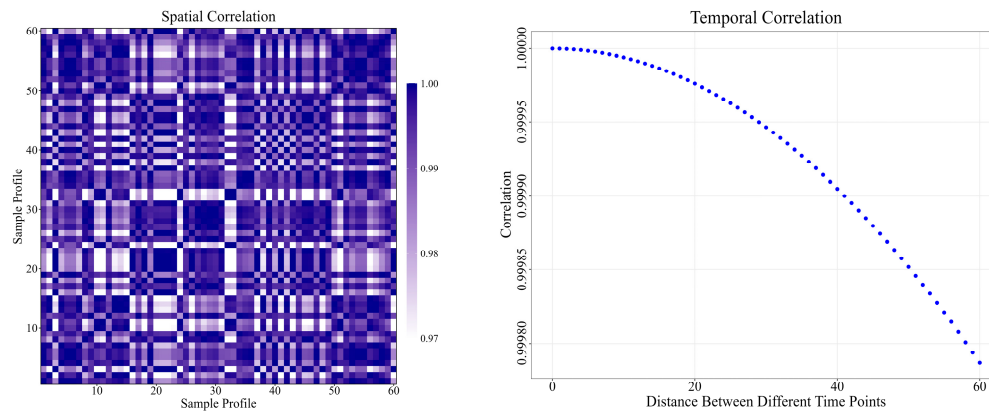


Figure 3. Empirical correlation of output levels across samples (left) and across time points (right).

To visually demonstrate the dynamic changes of the simulation function with respect to the spatial factor (x_1, x_2) and the temporal factor t , the surface plot and contour plots shown in Figure 4 were generated. As shown in Figure 4, the output level peaks at $x_1 = x_{1_0}$ and decreases with increasing $|x - x_{1_0}|$. Along the x_2 dimension, the output exhibits a unimodal structure centered at $x_2 = x_{2_0}$, overlaid with a weak negative linear correlation trend overall. Along the t dimension, the output shows a unimodal structure centered at $x_2 = x_{2_0}$, overlaid with a weak positive linear correlation trend overall.

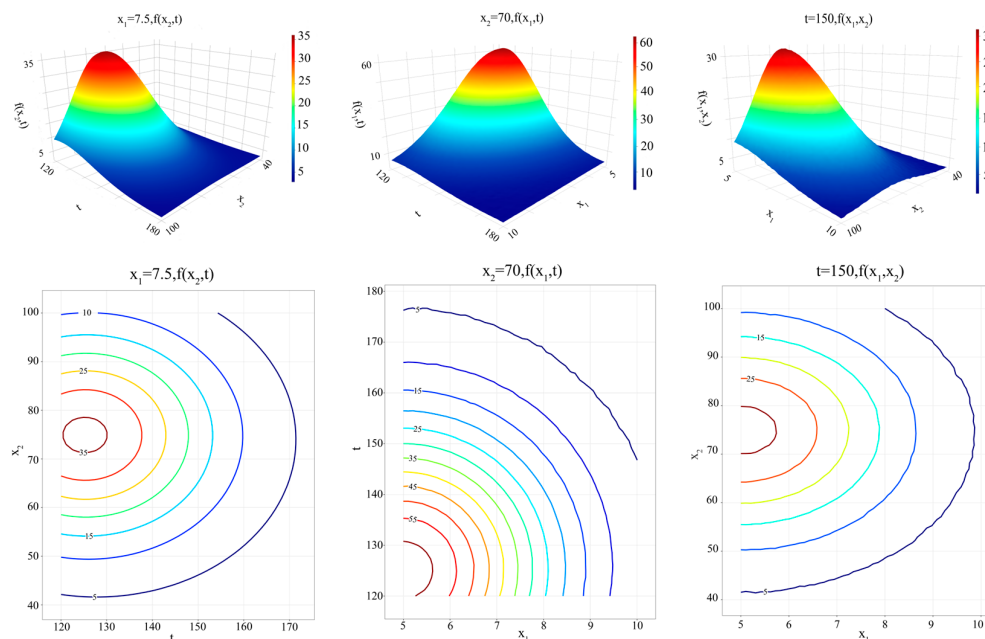


Figure 4. Surface plot and contour plot of the simulation function.

The results in Figures 3 and 4 collectively indicate that the output level is influenced by the combined effects of spatial and temporal factors. Therefore, identifying and quantifying the correlation between dynamic peaks and spatio-temporal factors is crucial for regulating spatial and temporal factors and achieving optimal system response.

4.2.2. Simulation Results

This paper randomly divides 60 sets of simulation data into a training set (50 sets) and a test set (10 sets), with the former used for model fitting and the latter for out-of-sample prediction. Use R language to perform hold-out tests on each data set at 25%, 50%, 75%, and 90% ratios among 600 observation points respectively. Each proportion is randomly repeated three times, and the average is taken. Simulation results for each method are presented in Table 1.

The comparison results in Table 1 show that compared with other models, the NSGP model exhibits the lowest root mean square error and 95% posterior confidence interval length across all test scenarios. Compared to the NSGP model, the 95% posterior confidence interval coverage rate of the “point-in-time LR” model is closer to the nominal 95% level, primarily due to its wider 95% posterior confidence interval length. However, this model's RMSE is higher than NSGP's across all test cases. In summary, the NSGP model constructed in this study demonstrates superior performance in prediction accuracy, interval efficiency, and overall stability.

Table 1. Comparison of Simulation Results Across Different Methods.

25% data reserved	RMSE	P_{CI}(95%)	L_{CI}(95%)
NSGP	0.0121	0.8972	0.0039
GP	0.0378	0.2514	0.0287
SGP	0.0127	0.1463	0.0042
LR by site	0.0498	—	—
LR by time	0.0378	0.9866	0.1608
RF by site	0.0498	—	—
RF by time	0.0189	—	—
NN	0.0157	—	—
50% data reserved	RMSE	P_{CI}(95%)	L_{CI}(95%)
NSGP	0.0137	0.9140	0.0056
GP	0.0441	0.3771	0.0502
SGP	0.0144	0.1769	0.0058
LR by site	0.0590	—	—
LR by time	0.0441	0.9262	0.1562
RF by site	0.0590	—	—
RF by time	0.0284	—	—
NN	0.0191	—	—
75% data reserved	RMSE	P_{CI}(95%)	L_{CI}(95%)
NSGP	0.0242	0.8605	0.0058
GP	0.0420	0.4794	0.0679
SGP	0.0254	0.1538	0.0061
LR by site	0.0554	—	—
LR by time	0.0474	0.9183	0.1851
RF by site	0.0554	—	—
RF by time	0.0320	—	—
NN	0.0304	—	—
90% data reserved	RMSE	P_{CI}(95%)	L_{CI}(95%)
NSGP	0.0358	0.8478	0.0098
GP	0.0549	0.3919	0.0565
SGP	0.0396	0.1307	0.0103
LR by site	0.0578	—	—
LR by time	0.0561	0.9389	0.2774
RF by site	0.0578	—	—
RF by time	0.0578	—	—
NN	0.0457	—	—

Figure 5 shows the predicted mean values obtained from direct covariance inversion and the FFBS algorithm. The points represent the output of $\hat{v}_i(s_{1:250}^*) | \hat{v}_i(s_{1:750}^D)$ calculated using equation (21) for the given parameter set $((\sigma_i^2, \gamma_i, \tau_i))$, and the left panel represents the case when $i = 1$ and the right panel represents the case when $i = 2$. The covariance inversion and FFBS algorithms compute the same quantity, so their predicted means are identical—as shown in Figure 5, the red triangle and blue dots overlap. Although the difference between the two computational methods is negligible, the FFBS algorithm enables precise calculation of numerous statistics, such as the posterior predicted mean, variance, and marginal likelihood function. Furthermore, these two algorithms exhibit significant differences in computational time. As shown in Figure 6, triangles and dots represent the FFBS algorithm and direct computation, respectively. When $n=5000$, estimating the likelihood function via the FFBS algorithm in R takes less than 1 second, whereas direct computation requires over 100 seconds. Clearly, the FFBS algorithm is more computationally efficient.

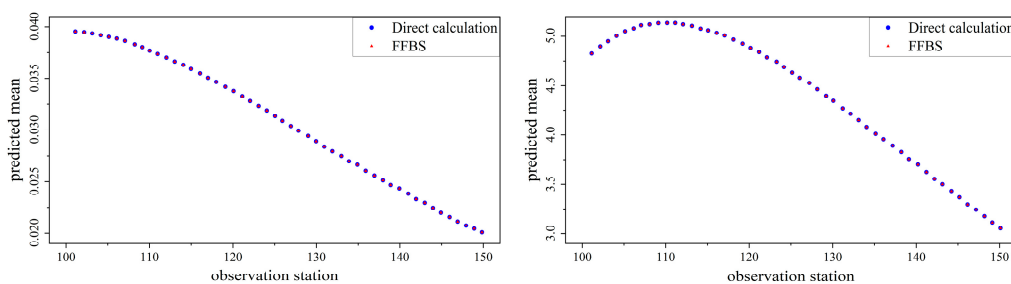


Figure 5. Predicted means from direct computation and FFBS algorithm.

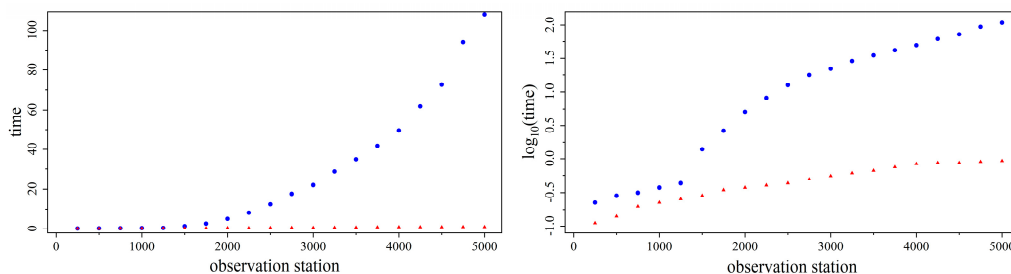


Figure 6. Computational time for likelihood function estimation via direct calculation and FFBS algorithm.

4.3. Case Study

Concrete production is a critical phase in dam construction. Concrete raw materials such as cement, coarse aggregates, and fine aggregates are mixed in a mixer or batching plant and then transported to the construction site for pouring. The slump of concrete mixture at the outlet reflects the workability of the concrete mixture in its initial state after mixing, and is an important indicator for evaluating fluidity, compactness, and construction performance. Resistance to segregation is a key indicator of concrete stability during transportation. Strong resistance prevents layering of aggregates and mortar, ensuring mixture uniformity. Mortar loss rate reflects the degree of mortar loss during transport; excessive loss leads to poor workability and insufficient strength in concrete. Therefore, properly controlling concrete slump, segregation resistance, and mortar loss rate is crucial for enhancing project quality.

This paper uses the relevant testing standards for concrete mixtures during the production of a dam as an example. Classified according to quality characteristics, concrete slump is the nominal the best, segregation resistance is the larger the better, and mortar loss rate is the smaller the better. The slump (response y_1), segregation resistance (response y_2), and mortar loss rate (response y_3) of 70

concrete mixes were observed under varying transport and waiting times (time factor $s \in [20, 40]$). The two parameters in this experiment were water-cement ratio (spatial factor $x_1 = 30\%, 35\%, 40\%, 45\%, 50\%, 55\%$) and cement temperature (spatial factor $x_2 = 12, 15, 18, 21, 24, 27, 30, 33$). Assume the target value for response y_1 is 5cm; the quality compensation resulting from interactions between spatiotemporal factors is 10 yuan/m³. The objective of this experiment is to achieve optimal concrete strength by selecting appropriate transport and waiting times, water-cement ratio, and cement temperature.

Figure 7 displays the empirical strength correlation of 70 concrete samples across the transport and waiting time range of [20,40]. Generated using R-Language's construction- "Shiny," this figure visually illustrates the associative patterns of strength across samples and the temporal dimension. Results indicate that in the spatial dimension, output levels between adjacent spatial parameters are highly correlated. In the temporal dimension, the average mass strengths at proximate time points are correlated with each other, and this correlation weakens as the time interval increases.

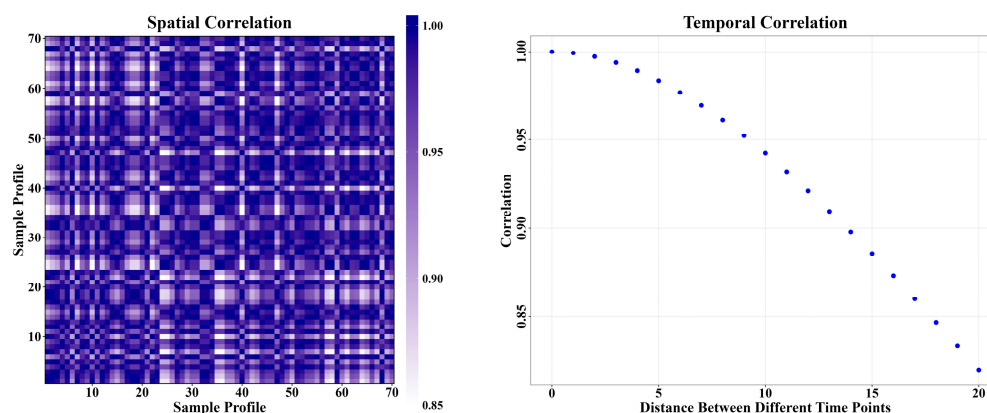


Figure 7. Empirical correlations of mass intensity across samples (left) and across time points (right) in the case study.

As shown in Figure 7, quality intensity is jointly influenced by spatial and temporal parameters. This study randomly selected 18 samples to test the models under conditions where 25% of the measurement values were excluded from the analysis. To evaluate model performance, a comparative analysis of modeling and prediction capabilities was conducted within the R framework for the NSGP, GP, and SGP methods. The optimization results for each method are detailed in Table 2. The core functions invoked during model construction were consistent across all three approaches, utilizing the "ginv" function from the MASS package, the "dist" and 'optim' functions from the stats package, and the "expand.grid" functions from the base package in R.

Table 2. Optimization Results for Dam Concrete Production Analysis Using Different Research Methods.

Research Methods	Optimal Settings for Controllable Factors			RMSE	P _{CI} (95%)	L _{CI} (95%)	Optimize the accumulation of predicted values	Quality Gain-Loss values
	x_1	x_2	s^1					
NSGP	0.4208	22.4000	×	0.8847	0.7233	1.8845	150.4114	24.5306
GP	0.4500	24.0000	×	24.2478	0.1233	4.7859	150.2646	59.0512
SGP	0.4000	21.0000	√	16.0889	0.1000	2.3944	149.7374	38.6978

Note: 1 × indicates that the controllable factor parameter design does not consider the time factor s , while √ indicates that the controllable factor parameter design does consider the time factor s .

A comparison of the optimization results in Table 2 clearly shows that the NSGP method achieves RMSE of 0.8847 and a 95% posterior confidence interval length of 1.8845, both lower than

those of other methods. The 95% posterior confidence interval coverage rate of the NSGP method is 0.7233, which is closer to 0.95 compared to other methods. The proposed method yields the optimal combined parameter combination for spatio-temporal factors, with an optimized prediction cumulative sum of 150.4114, exceeding the other two methods. The multivariate multidimensional quality gain-loss value obtained by this method is 24.5306, significantly lower than the optimized values of other methods, enabling superior concrete strength design.

5. Conclusions

This paper establishes an engineering robust design scheme based on non-separable Gaussian processes and Quality Gain-Loss functions theory. First, considering the correlation between spatial and temporal factors, the Matérn kernel function is constructed to establish a functional model between temporal factors and quality characteristics. The fast and accurate algorithms of Kalman filters and Kalman smoothing are employed to estimate and predict model data. Second, the gain-loss weights for each quality characteristic were determined based on the signal-to-noise ratio, forming a joint weight matrix. Multivariate, multidimensional quality gain-loss function models were then constructed for the larger the better, smaller the better, and nominal the best, respectively. Finally, the time and spatial factors were optimized to obtain the combined optimal parameter combination. In simulation examples, simulation data were substituted into NSGP, GP, SGP, LR, FR, and NN models for comparative analysis, with the NSGP model yielding superior simulation results. Through calculating gains and losses based on slump at discharge, segregation resistance, mortar loss rate, and relevant spatio-temporal factors during dam concrete production, the optimization results combining NSGP with three quality gain-loss functions — Artiles-leon's multivariate quality loss function, Pignatiello's multivariate quadratic quality loss function, and the traditional multivariate quality gain-loss function — all yielded optimal outcomes. This methodology not only achieves joint optimization of spatiotemporal factors but also reduces reliance on the covariance separability assumption, offering novel insights and technical pathways for hydraulic concrete quality control and robust engineering design.

Author Contributions: A.W. and X.C. contributed to the study conception, design, and material preparation. Data collection and analysis were performed by A.W. and X.C. The first draft of the manuscript was written by A.W., X.C. and T.F., J.L., S.T. and Y.L. provided input towards the development of ideas, revisions, and editorial oversight. X.C. organized the manuscript and formed the final manuscript. All authors reviewed the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Research on Digitalization and Intelligentization-Driven Collaborative Governance in the Wen'anwa Flood Storage(2025-85), the Training Programme for Young Backbone Teachers of Higher Education Institutions in Henan Province(2024GGJS061), the Natural Science Foundation of Henan(Grant No. 252300420469) and the High-level Talent Research Start-up Project of North China University of Water Resources and Electric Power(Grant No. 202310024).

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding authors.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Taguchi, G. Introduction to Quality Engineering [M]. China Foreign Languages Publishing House, Dec.1985.
2. Spiring, F. A. A General Class of Loss Functions with Industrial Applications. *Journal of Quality Technology*, 1998,30(2):152-162.
3. Spiring, F. A. The Reflected Normal Loss Function [J]. *The Canadian Journal of Statistics / La Revue Canadienne de Statistique*, 1993,21(3):321- 330.

4. Abolmohammadi M, Seif A, Behzadi M H, et al. Economic statistical design of adaptive Y control charts based on quality loss functions[J]. *Operational Research*, 2019,21.
5. N Artiles-León. A Pragmatic Approach to Multiple-Response Problems Using Loss Functions[J]. *Quality Engineering*, 1996.
6. J J P JR. Strategies for Robust Multi-Response Quality Engineering[J]. *IIE Transactions*, 1993.
7. Shaibu, A-B, Yang, et al. Development and Implementation of Type I and II Exponential Quality Loss Functions[J]. *IIE Annual Conference. Proceedings*, 2008,566-571.
8. Wang B J, Wang X, Shang G J. Quality Cost Equation and Economic Analysis[J]. *Aviation Standardization and Quality*, 2004,(04):12-16.
9. Cao, Y L., Yang, J X., Wu, Z T., et al. Research on Tolerance Robust Design Method Based on Fuzzy Quality Loss[J]. *Journal of Zhejiang University (Engineering Science)*, 2004,(01):2-511.
10. Wang J J, Ma Y Z, Ouyang L H, et al. Bayesian Modeling and Optimization for Robust Parameter Design with Multiple Responses[J]. *Journal of Management Science*, 2016,19(02):85-94.
11. Ni Ziyin, Wei Shizhen, Han Yuqi. Research on Process Mean Design Based on Asymmetric Losses [J]. *Operations Research and Management Science*, 2004(03):126-131.
12. Wang B. Research on Quality Control Technology of Dam Concrete Construction and Its Engineering Application [M]. China Water Resources and Hydropower Press: 201908:194.
13. Wang B, Fan T Y, Tian J, et al. Design of Quality Gain-Loss Function with Over- and Under-Estimation Characteristics when first-order loss is not neglected and compensation amount is constant[J]. *Mathematical Practice and Understanding*, 2019,49(18):153-160.
14. Nie X T, Liu C, Guo W J, et al. Design Optimization of Fuzzy Quality Loss-Gain Function Model and Its Process Mean[J]. *Henan Science*, 2020,38(09):1377-1386.
15. Liu M Q. Research on Concrete Construction Quality Control of Dams Based on Fuzzy Quality Loss Function [D]. North China University of Water Resources and Electric Power, 2020.
16. Bo W, Qikai L, Chen L, et al. Establishment and Application of a Grey Quality Gain-Loss Function Model[J]. *Processes*, 2022,10(3):495-495.
17. Liu Chen. Research on Concrete Construction Quality Control of Dams Based on Grey Quality Gain-Loss Function[D]. Henan Province:North China University of Water Resources and Electric Power, 2021.
18. Xiangtian N, Chen L, Bo W. Inverted Normal Quality Gain-Loss Function and Its Application in Water Project Construction[J]. *Journal of Coastal Research*, 2020, 104(sp1):415-420.
19. Bo W, Qi Y, Chen L, et al. Optimization Model of Engineering Specifications Based on Grey Quality Gain-Loss Function[J]. *Coatings*, 2021,11(11):1327-1327.
20. Bo W, Xiaojuan L, Jianyou S, et al. The establishment and application of Quality Gain-Loss Function When the loss of primary and cubic term is not ignored and the compensation quantity is constant.[J]. *PloS one*, 2023,18(12):e0295949-e0295949.
21. Wang B, Li Q, Yang Q, et al. Establishment of quadratic exponential quality gain-loss function model and design of process mean[J]. *Heliyon*, 2024,10(16):e36481-e36481.
22. Wang B, Yang R, Li P, et al. Design of Quality Gain-Loss Function with the Cubic Term Consideration for Larger-the-Better Characteristic and Smaller-the-Better Characteristic[J]. *Mathematics*, 2025,13(5):777-777.
23. E. T F, E. J O, M. N U. Multivariate Gaussian Process Emulators With Nonseparable Covariance Structures[J]. *Technometrics*, 2013,55(1):47-56.
24. Bilonis I, Zabarar N, Konomi A B, et al. Multi-output separable Gaussian process: Towards an efficient, fully Bayesian paradigm for uncertainty quantification[J]. *Journal of Computational Physics*, 2013,241212-239.
25. Abhirup D, Sudipto B, O A F, et al. Nonseparable Dynamic Nearest Neighbor Gaussian Process Models For Large Spatio-Temporal Data with an Application To Particulate matter analysis.[J]. *The Annals of Applied Statistics*, 2016,10(3):1286-1316
26. DIONISSIOS T. HRISTOPULOS. Non-Separable Covariance Kernels for Spatiotemporal Gaussian Processes Based on a Hybrid Spectral Method and the Harmonic Oscillator[J]. *IEEE Transactions on Information Theory*, 2024,70(2):1268-1283.

27. Mengyang G, Yanxun X. Fast Nonseparable Gaussian Stochastic Process With Application to Methylation Level Interpolation[J]. *Journal of Computational and Graphical Statistics*, 2020,29(2):250-260.
28. Zhang B, Konomi A B, Sang H, et al. Full-scale multi-output Gaussian process emulator with nonseparable autocovariance functions[J]. *Journal of Computational Physics*,2015,300623-642.
29. Cui C H, Wang J J, Ma Y Z, et al. Robust parameter design for spatio temporal response based on Gaussian process models[J]. *Theory and Practice of Systems Engineering*, 2023,43(02):537-555.
30. Rasmussen E C, Nickisch H. Gaussian Processes for Machine Learning (GPML) Toolbox.[J]. *Journal of Machine Learning Research*, 2010,113011 -3015.
31. Higdon D, Gattiker J, Williams B, et al. Computer Model Calibration Using High-Dimensional Output [J]. *Journal of the American Statistical Association*, 2008, 103(482):570-583.
32. HARTIKAINEN, JOUNI, SARKKA, SIMO. Kalman filtering and smoothing solutions to temporal Gaussian process regression models[C]//2010 IEEE International Workshop on Machine Learning for Signal Processing: MLSP 2010, Kittila, Finland, 29 August 1 September 2010.:IEEE,2010:379-384
33. PENG, Dingcong. Basic Principles and Applications of Kalman Filtering [J]. *Software Guide*, 2009,8(11):32-34.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.