Nested Bayesian Optimization for Computer Experiments

Yan Wang, Meng Wang, Areej AlBahar, and Xiaowei Yue, *Senior Member, IEEE*

***Abstract*—Computer experiments can emulate the physical systems, help computational investigations, and yield analytic solutions. They have been widely employed with many engineering applications (e.g., aerospace, automotive, energy systems). Conventional Bayesian optimization did not incorporate the nested structures in computer experiments. This article proposes a novel nested Bayesian optimization method for complex computer experiments with multistep or hierarchical characteristics. We prove the theoretical properties of nested outputs given that the distribution of nested outputs is Gaussian or non-Gaussian. The closed forms of nested expected improvement are derived. We also propose the computational algorithms for nested Bayesian optimization. Three numerical studies show that the proposed nested Bayesian optimization method outperforms the five benchmark Bayesian optimization methods that ignore the intermediate outputs of the inner computer code. The case study shows that the nested Bayesian optimization can efficiently minimize the residual stress during composite structures assembly and avoid convergence to local optima.**

***Index Terms*—Bayesian optimization, Gaussian process, multistage manufacturing, nested computer experiment, surrogate modeling.**

I. Introduction

COMPUTER experiments have become increasingly used in engineering simulations due to the development of information technology and computing power. Especially for the scenarios where physical experiments are difficult, expensive, or impossible to implement, computer experiments can serve as proxy surrogates for and adjuncts to physical experiments [1]. In advanced manufacturing and mechatronics, typical computer experiments may rely on finite element analysis (FEA), computational fluid dynamics (CFD), multiphysics simulation, variation propagation analysis, etc. Widely used engineering simulation software includes ANSYS, MATLAB/Simulink, COMSOL Multiphysics, Solidworks, 3DCS. Sophisticated computer codes can model the multistep or multiphysics processes accurately, thereby improving the efficiency of engineering design, system optimization, and quality control.

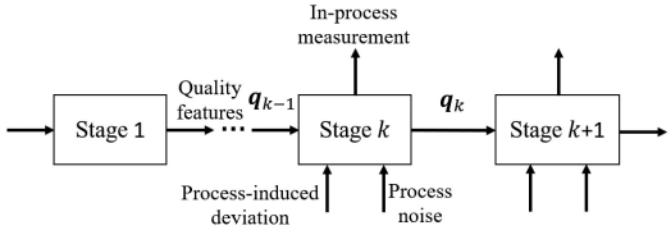

Fig. 1. Variation propagation in multistage manufacturing systems.

A. Nested Computer Experiments

First, we will illustrate what is *nested computer experiment*, and why the nested effect is very critical for engineering simulations, in particular for advanced manufacturing. If one model or system contains the outputs of the other model or system, we call them *nested*. Nested property usually comes from the hierarchical structures of systems and multiphysics phenomena. In practice, one system often contains a few subsystems; the output of one subsystem could be the input for the sequential subsystem. Nested structures are ubiquitous in engineering simulation. Suppose one computer experiment includes multilayer sequential operations/codes, and outputs from one computer code may serve as the inputs for the other level of computer code. In that case, we call it a nested computer experiment. The *nested computer experiment codes* are also called *System of Solvers* in engineering.

Most computer simulations and digital twins for multistage manufacturing processes (MMP) are nested, because of the natural multistep structure and inherent hierarchy in advanced manufacturing systems. In MMP, multiple operations/stations are involved to produce one product [2], [3], as shown in Fig. 1.

Manuscript received 10 December 2021; revised 6 March 2022, 27 May 2022, and 15 July 2022; accepted 20 August 2022. Recommended by Technical Editor S. Jeon and Senior Editor R. Gao. The work of Yue's research was supported by the Grainger Frontiers of Engineering Grant Award from the U.S. National Academy of Engineering (NAE) and the National Science Foundation (2035038); The work of Wang's research was supported by the National Natural Science Foundation of China (12101024) and the Natural Science Foundation of Beijing Municipality (1214019). *(Corresponding author: Xiaowei Yue.)*

Yan Wang and Meng Wang are with the School of Statistics and Data Science, Faculty of Science, Beijing University of Technology, Beijing 100124, China (e-mail: yanwang@bjut.edu.cn; wangmeng0230@163.com).

Areej AlBahar is with the Grado Department of Industrial and Systems Engineering, Virginia Tech, Blacksburg, VA 24061 USA, and also with the Department of Industrial and Management Systems Engineering, Kuwait University, Kuwait City 12037, Kuwait (e-mail: areejaa3@vt.edu).

Xiaowei Yue is with the Grado Department of Industrial and Systems Engineering, Virginia Tech, Blacksburg, VA 24061 USA (e-mail: xwy@vt.edu).

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/TMECH.2022.3202079.

Digital Object Identifier 10.1109/TMECH.2022.3202079

The product quality variations can propagate from one station to its downstream station. Stream of variation methodologies have been developed to model and reduce the variation and improve the quality control [2], [4]. When simulating the MMP in Fig. 1, the inputs for stage k include two types: input quality features q_{k-1} from the upstream stage $k-1$, and the new process-induced deviations and noise at the current stage. Similarly, the outputted quality features q_k of Stage k will also serve as inputs for downstream stage $k+1$. Wen et al. developed a computer simulation for composite aircraft assembly process [5], [6], where the simulation needs multiple steps even for a single-stage assembly, as shown in Fig. 9. Therefore, the omnipresent nested structure needs to be incorporated when modeling computer experiments.

B. Literature Review

In this section, we conduct the literature review from three fields: mechatronics, advanced statistics, and manufacturing systems.

In the mechatronics field, Rodriguez et al. developed one hybrid control scheme with two nested loops for twisted string actuators [7]. Nested design techniques have been used for codesign of controlled systems [8]. Zeng et al. proposed a nested optimization strategy to guarantee cost control for a motor driving system [9]. The performance-based nested Kriging model was constructed to interpolate the Antenna characteristics data [10]. Nested long-short term memory (LSTM) networks were incorporated into deep learning architecture for multivariate air quality prediction [11]. A nested tensor product model transformation was used to analyze the Takagi–Sugeno fuzzy system for system control design [12]. These approaches make full use of the nested structure for various objectives (control, design, prediction, etc.) and achieve excellent performance.

In the advanced statistics field, researchers investigated nested effects in computer experiments. Nested space-filling designs were constructed for computer experiments with two levels of simulation accuracy [13]. Next, nested Latin hypercube designs with sliced structures were proposed for experimental data collection [14]. Hung et al. developed the optimal Latin hypercube designs and kriging methods incorporating nested factors and branching factors [15]. Marque-Pucheu et al. proposed an efficient dimension reduction method for Gaussian process emulation of two nested codes [16]. Keogh and White investigated nested case-control and case-cohort study on exposure-disease association [17]. These methods significantly improve the efficiency and effectiveness of data collection, model emulation, and association analysis in advanced statistics.

In the advanced manufacturing field, nested systems have also been investigated. Gibson et al. used multivariate nested distributions to model semiconductor process variability [18]. Similarly, Tian et al. analyzed the nested variation pattern in the batch processes of semiconductor manufacturing, and proposed a two-level nested control chart for process monitoring [19]. Jin and Shi developed a reconfigured piecewise linear regression tree to model the nested structure for process control in multistage manufacturing [20]. Savin and Vorochaeva developed a quadratic programming based controller with nested structure, and it achieved excellent performance in planar pipeline robots [21]. Wang et al. proposed multiresolution and multisensor fusion network for fault diagnosis, with integration of multiple network structures [22]. These methods enhanced variability modeling, process control, and quality assurance by accommodating the nested structure.

C. Novelty and Contributions

Although numerous techniques have been investigated in studying and using nested effect, as mentioned in the literature review above, global optimization for nested computer experiments still lacks a systematic science base. This article focuses on the global optimization of nested computer experiments. We mainly use *two-layer nested computer models* as one example for nested computer experiments. The first-layer code is denoted as the inner computer model, and the second one as the outer computer model. The nested structure indicates that the outputs of the inner computer model are part of inputs of the outer computer model. The inner computer model and outer computer model are very complex and they are assumed to be black-box.

Bayesian optimization is an efficient approach to obtain the global optimal solution for complex computer experiments given specific objectives. This approach has proven to be successful in many real-world engineering optimization problems, such as the robust parameter design [23], image detection [24], the multiobjective optimization problems [25], [26], [27], and the constrained optimization problems [28]. The main steps of a standard Bayesian optimization method include: i) Build a statistical surrogate model based on previous computer outputs; ii) Choose an acquisition function and sequentially query the objective function at points which maximize the acquisition. For step (i), the most popular stochastic surrogate model is the Gaussian process (GP) model [1] or its variants [29]. For step (ii), commonly used acquisition functions include the expected improvement (EI) [30], [31], the lower/upper confidence bound (LCB) [32], and the expected quantile improvement (EQI) acquisition functions [33]. Despite the wide applications of Bayesian optimization methods, these existing methods ignored the outputs of the inner computer model and treated all the inputs characterizing the system of interest as a single input vector. When trying to find the global optimal solution of nested computer experiments, these existing Bayesian optimization methods are less efficient, since the nested structure information is ignored in the optimization. Astudillo and Frazier [34] considered Bayesian optimization of composite functions and took the outputs of the inner part of a composite function into account. This method performs excellently when the outer part of a composite function is a known, cheap-to-evaluated, and real-valued function. It does not work well for the complex black-box functions with nested structure, which is more common in engineering computer experiments.

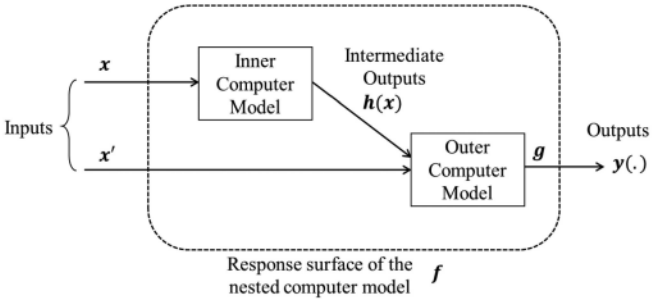

Fig. 2. Nested computer experiments.

In this work, we proposed a novel and systematic Bayesian optimization method for nested computer experiments. We assume that both the inner and outer computer models are deterministic, but expensive-to-evaluate. Our contributions can be summarized as follows:

1) The nested Bayesian optimization method is proposed to incorporate the nested structures in complex computer experiments. This method can learn the global optimum more efficiently and avoid convergence to the local optimum.
2) We investigated the theoretical properties of the nested Gaussian process for two cases: i) it can be approximated by a Gaussian process and ii) it cannot be approximated by a Gaussian process. Furthermore, we derive the closed forms of nested expected improvement and propose a computational algorithm for nested Bayesian optimization.
3) Based on the composite structures assembly case study, we show that nested Bayesian optimization can minimize the residual stress after assembly. We also show the proposed nested Bayesian optimization performs better than five benchmark methods via numerical studies.

The rest of this article organized is as follows: Section II introduces the optimization problem of two-nested computer experiments. Section III proposes the nested Bayesian optimization method. Sections IV and V compare the proposed method with the standard Bayesian optimization method by using three numerical studies and a real case study. Finally, Section VI concludes this article. Appendices contain detailed proofs of the theorems and selection of correlation functions.

II. Problem Setting

In this section, we use mathematical models to describe the problem setting. Denote $f : \mathcal{X} \to \mathbf{R}$ to be a nested computer model, which is defined as

$$f(\tilde{\boldsymbol{x}}) = g(\boldsymbol{h}^T(x), x'); \tilde{\boldsymbol{x}} = (x, x')^T \in \mathcal{X} \subset \mathbf{R}^d \qquad (1)$$

where $\boldsymbol{h}(x) = (h_1(x), \ldots, h_p(x))^T, p \geq 1$ is a vector of inner computer model outputs. $g(\cdot)$ is the outer computer model whose inputs include outputs of the inner computer model $\boldsymbol{h}(x)$ and the additional control variable x'. There is a serial relationship between the inner computer model and outer computer model. Intermediate outputs $\boldsymbol{h}(x)$ and x' are parallel inputs. Fig. 2 shows the framework of nested computer experiments:

Suppose these two computer models are black-box, deterministic, expensive-to-evaluate, and the gradient information is not available. With the help of a limited number of outputs from both computer models, we consider the problem of finding a minimizer of the entire response surface of the nested computer model f:

$$\tilde{\boldsymbol{x}}^* = \underset{\tilde{\boldsymbol{x}} \in \mathcal{X}}{\operatorname{argmin}} f(\tilde{\boldsymbol{x}}). \qquad (2)$$

Specifically, suppose the nested computer experiments are conducted at the points $\tilde{X}_n = (\tilde{\boldsymbol{x}}_1, \ldots, \tilde{\boldsymbol{x}}_n)^T$, which contains the collections of $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ and $\{\tilde{\boldsymbol{x}}_1, \ldots, \tilde{\boldsymbol{x}}_n\}$. The first-layer computer model generates intermediate outputs $H_n = (\boldsymbol{h}(x_1), \ldots, \boldsymbol{h}(x_n))^T$, and the second-layer computer model generates the outputs $Y_n = (g(\boldsymbol{h}^T(x_1), x_1'), \ldots, g(\boldsymbol{h}^T(x_n), x_n'))^T$. These computer experiments yield data $D_n = \{\tilde{X}_n, H_n, Y_n\}$. The goal of this work is to query $\tilde{\boldsymbol{x}}^*$ by making full use of the dataset D_n.

As discussed above, the standard Bayesian optimization method can be used to solve the optimization problem (2). This approach can query the optimal point of f sequentially by optimizing an acquisition function. In this work, we focus on the EI criterion [1], [30]. Detailed comparisons are conducted between EI, LCB, and EQI-based approaches in Sections IV and V.

The main idea of EI is to sample the point offering the greatest expected improvement over the current best sampled point. Let $f_n^* = \min_{i=1}^n \{y_i\}$ be the current best objective value, given data $\{\tilde{X}_n, Y_n\}$, the EI function becomes

$$\mathrm{EI}_n(\tilde{\boldsymbol{x}}) = E_{f|\tilde{X}_n, Y_n}(f_n^* - f(\tilde{\boldsymbol{x}}))_+ \qquad (3)$$

where $(f_n^* - f(\tilde{\boldsymbol{x}}))_+ = \max\{f_n^* - f(\tilde{\boldsymbol{x}}), 0\}$ is the improvement utility function.

It can be known that the evaluation of EI depends on the posterior distribution $f|\tilde{X}_n, Y_n$. Since the posterior distribution $f|\tilde{X}_n, Y_n$ in standard Bayesian optimization method ignores the outputs of the inner computer model, it leads to low optimization efficiency or even getting stuck in a local optimum when the number of samples is limited. To overcome this limitation, we will develop a new Bayesian optimization method to incorporate the nested structure and identify the optimal solution for complex computer experiments.

III. Nested Bayesian Optimization

Nested computer experiments are ubiquitous when running engineering simulations, digital twin or finite element analysis. Conventional Bayesian optimization approaches consider the entire system as a whole and try to identify the global optimum for black-box functions. They are less efficient in complex systems optimization when nested structures exist. The nested structures usually can be determined according to the system configurations or engineering knowledge. By incorporating the nested structures of complex systems, we can make full use of more information in Bayesian optimization, intuitively avoid getting stuck in some local optima, and have the potential to

improve optimization efficiency. In this section, we propose a novel method, named as *Nested Bayesian Optimization (NBO)*, to query the global optimal solution of nested computer experiments. To approximate the outputs of nested computer experiments, we first introduce nested Gaussian Process (NGP) models in Section III-A. Next, we derive the closed forms of the expected improvement acquisition function for nested computer experiments in Section III-B, under the cases that the NGP models are Gaussian and non-Gaussian. Section III-C provides a detailed algorithm of the NBO method.

A. Nested Gaussian Process Models

In this work, Gaussian Process (GP) models [1] are used to mimic the inner and the outer computer models. Suppose h and g are realizations of two Gaussian Processes. Given data D_n, the posterior distribution of the inner computer model at an unobserved input x is

$$\boldsymbol{h}(x)|D_n \sim N(\hat{\boldsymbol{h}}_n(x), s_h^2(x)) \tag{4}$$

where $\hat{\boldsymbol{h}}_n(x)$ is a $p \times 1$ mean vector, and $s_h^2(x)$ is a $p \times p$ covariance matrix. The posterior distribution of the outer computer model at an unobserved input $x^{out} = (\boldsymbol{h}^T, x')$ is

$$g(x^{out})|D_n \sim N(\hat{g}_n(x^{out}), s_g^2(x^{out})). \tag{5}$$

Formulations of the posterior mean and posterior variance function are given by (18) and (19), respectively, in Appendix A. More Details can be found in Appendix A. All the appendices are in the supplementary materials associated with this paper.

The *nested Gaussian Process (NGP) model* is expressed as

$$f(\tilde{\boldsymbol{x}})|D_n = \hat{g}_n(\Psi^T(x), x') + s_g(\Psi^T(x), x')\xi_g \tag{6}$$

where $\Psi(x) = \boldsymbol{h}(x)|D_n$, ξ_g is a standard normal random variable. From the posterior distribution of the inner computer model (4), $\Psi(x)$ can be represented as $\Psi(x) = \hat{\boldsymbol{h}}_n(x) + s_h(x)\boldsymbol{\xi}_h$, where $\boldsymbol{\xi}_h$ is a $p \times 1$ random vector that follows the normal distribution and it is independent from ξ_g. By numerical calculations, we have that, the posterior variance of $f(\tilde{\boldsymbol{x}})|D_n$ is zero for any $i = 1, \ldots, n$, and the posterior mean is interpolating the observed data values $(\tilde{X}_n, Y_n)$.

From (6), we can see that $\Psi(x)$ follows a normal distribution when $s_h(x) \neq 0$. As a function of $\Psi(x)$, the posterior distribution of $f(\tilde{\boldsymbol{x}})|D_n$ may not be normal. Therefore, we will investigate two cases, Gaussian and non-Gaussian in the following part.

Theorem 1 focuses on the Gaussian case, while Theorem 2 analyzes the non-Gaussian case.

Theorem 1: Denote $\mu_Z(\tilde{\boldsymbol{x}}) = \hat{g}_n(\hat{\boldsymbol{h}}_n^T(x), x')$ and $s_Z^2(\tilde{\boldsymbol{x}}) = s_g^2(\hat{\boldsymbol{h}}_n^T(x), x')$. The NGP model (6) is the following GP model:

$$GP(\mu_Z(\tilde{\boldsymbol{x}}), s_Z^2(\tilde{\boldsymbol{x}})) \tag{7}$$

if and only if for all $\tilde{\boldsymbol{x}} \in \mathcal{X}$, there is $s_h(x) = \boldsymbol{0}_{1\times p}$.

For ease of understanding, here we give the brief proof of Theorem 1. First, $s_h(x) = \boldsymbol{0}_{1\times p}$ indicates that the surrogate of inner computer model is deterministic. By plugging $\Psi(x) = \hat{\boldsymbol{h}}_n(x)$ into (6), we can derive that $f(\tilde{\boldsymbol{x}})|D_n$ follows a normal distribution for fixed $\tilde{\boldsymbol{x}}$. In addition, the NGP model is gaussian, implying that at least one of the following two conditions holds:

1) The outer computer model is independent on the inner computer outputs, i.e., the NGP model (6) can be expressed as $\hat{g}_n(x') + s_g(x')\xi_g$. Due to the nested structure, both $\hat{g}_n$ and s_g depend on Ψ. This condition is not true.
2) $\Psi(x) = \hat{\boldsymbol{h}}_n(x)$. It indicates that $s_h(x)$ equals to zero and the surrogate of inner computer model is deterministic.

Theorem 1 states that for a nested computer model, the NGP is a GP model if and only if the surrogate of inner computer model is deterministic. This condition is hard to achieve or even unattainable in some cases. Indeed, from Corollary 1, when s_h is close to $\boldsymbol{0}$, i.e., the inner GP model can achieve satisfactory prediction accuracy, the GP model (7) can be used to mimic the nested computer experiments.

Theorem 2: Denote $\boldsymbol{c}_h^T(\tilde{\boldsymbol{x}}) = \frac{\partial \hat{g}_n}{\partial \boldsymbol{h}}(\hat{\boldsymbol{h}}_n^T(x), x')s_h(x)$, $c_g(\tilde{\boldsymbol{x}}) = s_g(\hat{\boldsymbol{h}}_n^T(x), x')$, and $\boldsymbol{c}_{h,g}^T(\tilde{\boldsymbol{x}}) = \frac{\partial s_g}{\partial \boldsymbol{h}}(\hat{\boldsymbol{h}}_n^T(x), x')s_h(x)$. Assume that the second-order derivatives of $\hat{g}_n$ and s_g with respect to $\boldsymbol{h}$ are uniformly bounded. The NGP model (6) is a non-Gaussian Process model if and only if there is $\tilde{\boldsymbol{x}} \in \mathcal{X}$, such that $s_h(x) \neq \boldsymbol{0}_{1\times p}$. Specifically, in this case, the NGP model (6) can be approximated by

$$Z(\tilde{\boldsymbol{x}}) = Z_1(\tilde{\boldsymbol{x}})Z_2(\tilde{\boldsymbol{x}}) + z_0(\tilde{\boldsymbol{x}}). \tag{8}$$

Here, $Z_1(\tilde{\boldsymbol{x}})$ and $Z_2(\tilde{\boldsymbol{x}})$ are independent Gaussian Processes with mean functions $\mu_1(\tilde{\boldsymbol{x}}) = c_g(\tilde{\boldsymbol{x}})/\sqrt{\boldsymbol{c}_{h,g}^T(\tilde{\boldsymbol{x}})\boldsymbol{c}_{h,g}(\tilde{\boldsymbol{x}})}$, $\mu_2(\tilde{\boldsymbol{x}}) = \sqrt{\boldsymbol{c}_h^T(\tilde{\boldsymbol{x}})\boldsymbol{c}_h(\tilde{\boldsymbol{x}})}$, respectively, and variance functions $\sigma_1^2(\tilde{\boldsymbol{x}}) = 1$, $\sigma_2^2(\tilde{\boldsymbol{x}}) = \boldsymbol{c}_{h,g}^T(\tilde{\boldsymbol{x}})\boldsymbol{c}_{h,g}(\tilde{\boldsymbol{x}})$, respectively; $z_0(\tilde{\boldsymbol{x}}) = \mu_Z(\tilde{\boldsymbol{x}}) - \mu_1(\tilde{\boldsymbol{x}})\mu_2(\tilde{\boldsymbol{x}})$. In addition, the mean and variance functions of $Z(\tilde{\boldsymbol{x}})$ are

$$\begin{aligned} \mathrm{E}[Z(\tilde{\boldsymbol{x}})] &= \mu_Z(\tilde{\boldsymbol{x}}) = \hat{g}_n(\hat{\boldsymbol{h}}_n^T(x), x') \\ \mathrm{Var}[Z(\tilde{\boldsymbol{x}})] &= \boldsymbol{c}_h^T(\tilde{\boldsymbol{x}})\boldsymbol{c}_h(\tilde{\boldsymbol{x}}) + c_g^2(\tilde{\boldsymbol{x}}) + \boldsymbol{c}_{h,g}^T(\tilde{\boldsymbol{x}})\boldsymbol{c}_{h,g}(\tilde{\boldsymbol{x}}). \end{aligned} \tag{9}$$

Remark 1: For a fixed $\tilde{\boldsymbol{x}} \in \mathcal{X}$, $Z(\tilde{\boldsymbol{x}})$ is a non-Gaussian random variable. The exact probability density function of $Z(\tilde{\boldsymbol{x}})$ is given by (22) in Appendix B. If $z_0(\tilde{\boldsymbol{x}}) = 0$, $Z(\tilde{\boldsymbol{x}})$ follows a normal product (NP) distribution [35], which is in general non-Gaussian. Especially, if $Z_1(\tilde{\boldsymbol{x}}) \sim N(0,1)$ and $Z_2(\tilde{\boldsymbol{x}}) \sim N(0,1)$, then density function of $Z_1(\tilde{\boldsymbol{x}})Z_2(\tilde{\boldsymbol{x}})$ is

$$p_Z(z) = \frac{K_0(|z|)}{\pi}, \infty < z < +\infty.$$

Here, K_0 denotes the modified Bessel function of the second kind with order 0. This density function exhibits a sharp peak at the origin and heavy tails.

Detailed proof of Theorem 2 can be found in Appendix B. Theorem 2 states that the NGP model can be approximated by a non-Gaussian process model $Z(\tilde{\boldsymbol{x}})$. The global trend of $Z(\tilde{\boldsymbol{x}})$ is the same as the posterior mean of (6). The variance of $Z(\tilde{\boldsymbol{x}})$ involves three kinds of uncertainty: $\boldsymbol{c}_h^T(\tilde{\boldsymbol{x}}) = \frac{\partial \hat{g}_n}{\partial \boldsymbol{h}}(\hat{\boldsymbol{h}}_n^T(x), x')s_h(x)$ is the uncertainty due to the inner GP model; $c_g(\tilde{\boldsymbol{x}}) = s_g(\hat{\boldsymbol{h}}_n^T(x), x')$ is the uncertainty due to the

outer GP model; $c^T_{h,g}(\tilde{x}) = \frac{\partial s_g}{\partial h}(\hat{h}^T_n(x), x')s_h(x)$ is the uncertainty arising from the combined effect of the inner and outer models. In addition, from Theorem 2, we have that, there is a great difference between the NGP and composite GP [36]. The composite GP model is an addition of two Gaussian Processes, where the first one captures the smooth global trend and the second one models local details. Thus, the composite GP is still a Gaussian Process. However, the NGP may no longer be a Gaussian Process.

Corollary 1: If $s_h(x)$ converges to $\mathbf{0}_{1\times p}$ for all $\tilde{x} \in \mathcal{X}$, $Z_2(\tilde{x})$ tends to be a deterministic function. In this case the model (8) converges to the GP model (7).

Corollary 1 shows that the NGP model (6) can be approximated by the GP model (7), if $s_h(x)$ is small for all $\tilde{x} \in \mathcal{X}$. It relaxes the condition for an NGP model able to be approximated by a GP model in Theorem 1.

From Theorems 1 and 2, we can see that, the posterior mean and variance function of the NGP model depend only on the posterior mean and variance of the inner and the outer GP models. Given the fact that the computational complexity for the outer GP model is $O(n^3)$, and for the inner GP model is $O((pn)^3))$ [1], the computational complexity for the NGP model is $O((pn)^3))$.

B. Closed Forms of the Nested Expected Improvement (NEI)

To distinguish from the standard Bayesian optimization method, the EI function where NGP is used to approximate the nested computer experiments is called *Nested Expected Improvement (NEI)* function:

$$\mathrm{NEI}_n(\tilde{x}) = E_{f|D_n}(f^*_n - f(\tilde{x}))_+ \tag{10}$$

A new queried point $\tilde{x}_{n+1}$ is selected by maximizing the $\mathrm{NEI}_n(\tilde{x})$ function

$$\tilde{x}_{n+1} = \underset{\tilde{x}\in\mathcal{X}}{\operatorname{argmax}}\, \mathrm{NEI}_n(\tilde{x}). \tag{11}$$

We can see that values of NEI_n depend on the posterior distribution $f(\tilde{x})|D_n$. Given two cases depending on whether NGP model can be approximated by a Gaussian process, the NEI acquisition function also has different expressions. Specifically,

1) If the NGP model can be approximated by the GP model (7), denote $v(\tilde{x}) = \frac{f^*_n - \mu_Z(\tilde{x})}{s_Z(\tilde{x})}$, the NEI acquisition function has the closed-form expression:

$$(f^*_n - \mu_Z(\tilde{x}))\Phi_N\left(v(\tilde{x})\right) + s_Z(\tilde{x})\phi_N\left(v(\tilde{x})\right). \tag{12}$$

2) If the NGP model cannot be approximated by a GP model, the NEI acquisition function can be evaluated by

$$\begin{aligned}\int_{-\infty}^{\infty} &(f^*_n - z_0(\tilde{x}) - t\mu_2(\tilde{x}))\phi_N\left(u_1(t,\tilde{x})\right)\\ &\times \Phi_N\left(u_2(f^*_n, t, \tilde{x})\right)\\ &+ |t|\sigma_2(\tilde{x})\phi_N\left(u_1(t,\tilde{x})\right)\phi_N\left(u_2(f^*_n, t, \tilde{x})\right) dt.\end{aligned} \tag{13}$$

Detailed derivation of (12) and (13) can be found in [30] and Appendix B, respectively.

Algorithm 1: Nested Bayesian Optimization.

1: Obtain an initial design $\tilde{X}_{n_0}$ with n_0 points, and run the nested computer models at these points, yielding corresponding simulator outputs H_{n_0}, Y_{n_0}.
2: **for** iteration $n = n_0, \ldots, N-1$ **do**
3: Evaluate the current best optimal point $\tilde{x}^*_n = \operatorname{argmin} Y_n$ and the corresponding function value $f^*_n = \min Y_n$.
4: Build GP models (4) and (5) to mimic the inner and the outer computer models, respectively.
5: Test whether the NGP model is a GP model by using a cross-validation method.
6: **if** NGP model is Gaussian **then**
7: Identify the maximizer $\tilde{x}_{n+1}$ of NEI_n (12).
8: **else**
9: Identify the maximizer $\tilde{x}_{n+1}$ of NEI_n (13).
10: **end if**
11: Run the nested computer models at $\tilde{x}_{n+1}$, augment $\tilde{X}_n$, H_n and Y_n with $\tilde{x}_{n+1}$, $h(x_{n+1})$ and $f(\tilde{x}_{n+1})$.
12: **end for**
13: **Return** the current best optimal point $\tilde{x}^*_N = \operatorname{argmin} Y_N$ and the corresponding function value $f^*_N = \min Y_N$.

Remark 2: The NEI acquisition function (12) implicitly encodes a tradeoff between exploration of the feasible region and exploitation near the current best solution. The first term in (12) encourages exploitation, by assigning larger values for points with smaller predicted values; the second term in (12) encourages exploration, by assigning greater values for points with larger estimated posterior variance.

Remark 3: Markov Chain Monte Carlo (MCMC) method can be used to estimate NEI_n (13). Because $\phi_N(u_1(t,\tilde{x})) = 0$ as $u_1(t,\tilde{x})$ tends to infinity, the interval of integration $t \in (-\infty, \infty)$ can be shrunk to $t \in [L_t(\tilde{x}), U_t(\tilde{x})]$, where $L_t(\tilde{x})$ and $U_t(\tilde{x})$ are prespecified, such as $L_t = -10\sigma_1(\tilde{x}) + \mu_1(\tilde{x})$ and $U_t = 10\sigma_1(\tilde{x}) + \mu_1(\tilde{x})$, respectively.

Remark 4: Sampled expected improvement (SEI) as suggested in [37] is a commonly used method to estimate EI values when $f(\tilde{x})|D_n$ is non-Gaussian. SEI estimates EI values based on a large number of posterior samples of $f(\tilde{x})|D_n$ and only the prediction posterior samples that are smaller than the current best value are taken in the calculation. Since generating posterior samples of $f(\tilde{x})|D_n$ by using the posterior density function (22) is rather time-consuming, this method loses attraction.

C. Algorithm

In this section, we develop the computational algorithm for nested Bayesian optimization. Algorithm 1 provides detailed steps of the NBO method.

We can explain this algorithm as follows. First, initial data are collected based on a maximin Latin hypercube design. Here, the number of initial points n_0 is set at 10 d, as recommended in [38]. Next, Gaussian Process models are built to mimic the

inner model and the outer model by using (4) and (5). Then, K-fold cross-validation method is used to exam whether the NGP is a GP or not. More specifically, build GP model (7) to approximate the nested computer outputs and then examine the prediction accuracy of this GP model by K-fold cross-validation method. Here, choice of K follows the criterion below [39]

$$K \approx \log(n) \text{ and } \quad n/K > 3\ d.$$

Finally, query the sequential points by maximizing (12) (when NGP is Gaussian) or by maximizing (13) (when NGP is non-Gaussian), until the sample size budget N is reached.

IV. Numerical Studies

In this section, we compare the proposed NEI method with five benchmark methods. The five benchmark methods include

1) EI-GP: the Expected Improvement (EI) method under the one-GP model;
2) LCB-GP: the Lower Confidence Bound (LCB) method under the one-GP model;
3) LCB-NGP: the Lower Confidence Bound (LCB) method under the NGP model;
4) EQI-GP: the Expected Quantile Improvement (EQI) method under the one-GP model; and
5) EQI-NGP: the Expected Quantile Improvement (EQI) method under the NGP model.

The tuning parameter for the LCB function is selected as in the following [32], [40].

The simulation setup is as follows. We generate the inputs $\tilde{X}_{n_0}$, where $n_0 = 10\ d$, according to a maximin Latin hypercube design via the R package *maximinLHS*. Then, we collect the inner computer model outputs H_{n_0}, and the outer computer model outputs Y_{n_0} on H_{n_0} and $\tilde{X}_{n_0}$.

To obtain the NGP predictor, two GP models are built to mimic the inner and outer computer models, respectively. Here, the GP models are fitted using the R package *DiceKriging* [41]. The log-optimality gap is used to compare the performance of different methods, which is defined as

$$log_{10}(f_n^* - f^*).$$

All results about the log-optimality gap are averaged over 50 replications.

A. 1-D GP Model

Suppose the inner computer model and the outer computer model are both commonly used one-dimension (1-D) test functions in the literature on GP models [1]:

$$h(x) = \exp(-1.4x)\cos(7\pi x/2) - 1.4x, x \in [0, 1]$$

$$g(h) = h\sin(\pi h/2).$$

The global minimum of $f(x) = g(h(x))$ is at $x^* = 0.124$ and the corresponding function value is 0.

These two GP models are built to mimic the inner and outer computer models, respectively. To illustrate the reasons we the choose Gaussian correlation functions, a detailed comparison of the model accuracy between the one-GP model and the NGP model under different correlation functions is given in Appendix C.

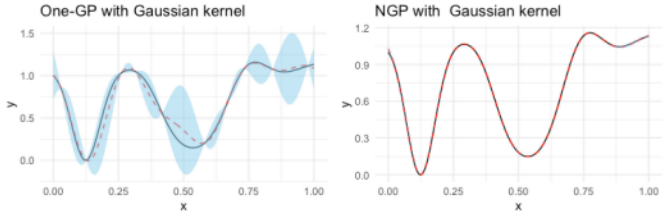

Fig. 3. Left: predictions (red dotted line) and 95% confidence intervals of the one-GP model build by using $(\tilde{X}_{n_0}, Y_{n_0})$, with $n_0 = 10$; Right: predictions (red dotted line) and 95% confidence intervals of the NGP model.

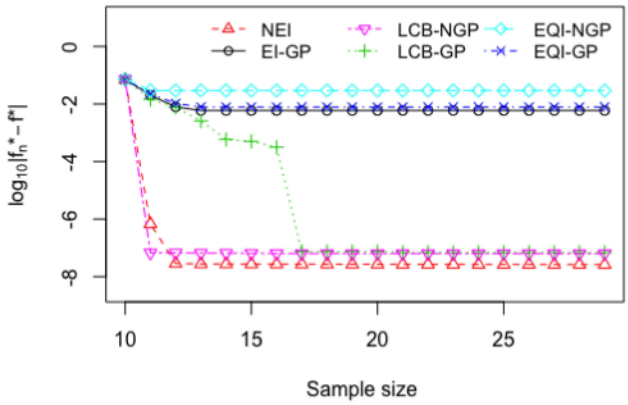

Fig. 4. Average optimality gap over 50 replications by different methods.

By the threefold cross-validation (CV) method, we have that, the NGP model is a GP model. Fig. 3 compares the performance of the one-GP build by using $(\tilde{X}_{n_0}, Y_{n_0})$ and the NGP model approximated by a composite GP model. It can be seen that, both mean functions of the one-GP model and the NGP model match the true function accurately, but the 95% confidence intervals indicate that, the NGP predictor has smaller variance than the one-GP predictor. The reason for this result is that, f is a realization from a nonstationary GP. Compared to the stationary one-GP model, the NGP model can approximate f more accurately and can also improve the prediction intervals, especially when the experimental design is sparse [36].

Fig. 4 shows the log-optimality gap against the number of samples for the six methods. From Fig. 4, we can see that, the optimality gaps for NEI, LCB-NGP, and LCB-GP enjoy steady improvements as n increases, whereas the optimality gap for the other methods stagnates for larger sample sizes. The proposed method outperforms other methods. The NGP-based approaches outperform the one GP-based approaches under the same acquisition function. This is a very direct result of the more accurate predictions for the NGP model.

B. 1-D Non-GP Model

Suppose the inner computer model is

$$h(x) = (1 + |x|)^{-4}, x \in [-1, 1]$$

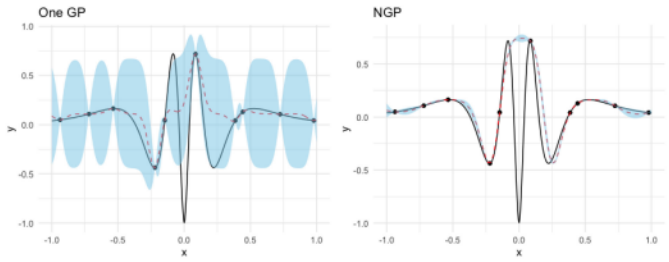

Fig. 5. Left: predictions (red dotted line) and 95% confidence intervals of the one-GP model build by using $(\tilde{X}_{n_0}, Y_{n_0})$, with $n_0 = 10$; Right: predictions (red dotted line) and 95% confidence intervals of the NGP model.

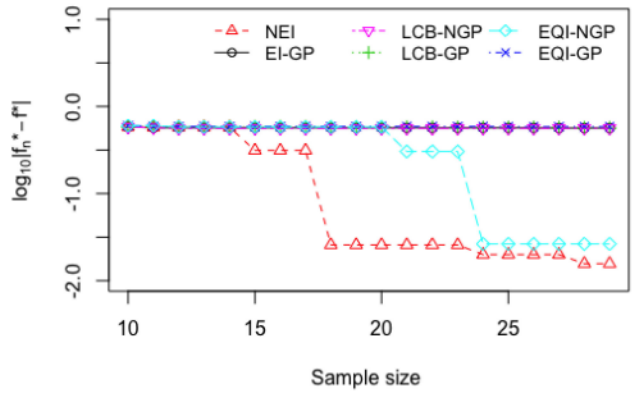

Fig. 6. Average optimality gap over 50 replications by different methods.

and the outer computer model is

$$g(h) = h\sin(7\pi h/2).$$

The global minimum of this nested computer experiment is $(0, -1)$. Fig. 5 compares the performance of one GP model and NGP model with $n_0 = 10$.

Fig. 5 shows that both the one-GP model and the NGP model perform poor in $x \in [-0.1, 0.1]$. The reason is that, values of the true function changes fast in $x \in [-0.1, 0.1]$, but the design is sparse in $[-0.1, 0.1]$. Except at the points that belong to $[-0.1, 0.1]$, the NGP model outperforms the one-GP model. Via the threefold CV test, we can find that the NGP model is not Gaussian. Therefore, in the NBO algorithm, the sequential point is collected by maximizing NEI_n (13). Set $L_t = -10\sigma_1(x) + \mu_1(x)$ and $U_t = 10\sigma_1(x) + \mu_1(x)$, MCMC method is used to evaluate (13) and the EQI function. The log-optimality gaps against the number of samples for the six methods are shown in Fig. 6.

From Fig. 6, we can conclude that the optimality gaps for NEI and EQI-NGP enjoy steady improvements as n increases. However, the other methods fall into a local optimal point, which is included in the initial design. This shows that the proposed method balances the optimal point of the fitted model with the exploration of other regions.

It is worth noting that, since the LCB depends only on the posterior mean and variance of $f(\tilde{x})$, this acquisition function lose its advantage when the posterior distribution of $f(\tilde{x})$ is non-Gaussian.

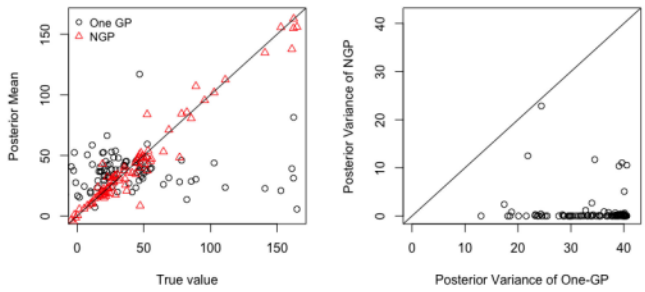

Fig. 7. Left: Posterior mean of the one-GP (black circles) and NGP (red triangles); Right: Posterior variance of the one-GP and NGP.

C. 4-D GP Model

Suppose the inner computer model includes two functions: the three-hump camel function

$$h_1(x) = 2x_1^2 - 1.05x_1^4 + x_1^6/6 + x_1x_2 + x_2^2$$

and the six-hump camel function

$$h_2(x) = (4 - 2.1x_3^2 + x_3^4/3)x_3^2 + x_3x_4 + (-4 + 4x_4^2)x_4^2$$

Here, $x = (x_1, x_2, x_3, x_4) \in [-1, 1]^4$. Suppose the outer computer model is the Branin function

$$g(h) = \frac{1}{51.95}\left[g_1(h) + \left(10 - \frac{10}{8\pi}\right)\cos(\bar{h}_1) - 44.81\right]$$

where $g_1(h) = (\bar{h}_2 - \frac{5.1\bar{h}_1^2}{4\pi^2} + \frac{5\bar{h}_1}{\pi} - 6)^2$, $\bar{h}_1 = 5(h_1 - 1)$, $\bar{h}_2 = 5(h_2 + 1)$. The global minimum of $f = g(h(x))$ is at $x^* = (-0.121, 0.547, 0.915, 0.715)$ and the corresponding function value is -16.644. Let $n_0 = 40$, we still use the maximin Latin hypercube design to collect data. Then, we build GP models for inner and outer computer models. Via the threefold CV test, we have that the NGP model is Gaussian.

Fig. 7 compares the prediction performance of one GP model and NGP model at 100 unobserved locations. These 100 testing locations are sampled by the maximin Latin hypercube design. Left of Fig. 7 shows the comparison between predictions of different models and the true outputs of the nested computer experiment. We see that, predictions given by the NGP model at these testing locations are much closer to the true values. The 100 points (black circles) in Fig. 7 right compare the posterior variances given by the one-GP model and the NGP model. Because all 100 points are under the line "$y = x$," it indicates that posterior variances given by the NGP model are smaller than posterior variances given by the one-GP model.

Fig. 8 shows the log-optimality gap $\log_{10}(f_n^* - f^*)$ against the number of samples n. Results of the log-optimality gap are averaged over 50 replications. We can see from Fig. 8 that the proposed method outperforms other methods: the optimality gap for the latter methods stagnates for larger sample sizes, whereas the former enjoys steady improvements as n increases.

In summary, results of the numerical simulations show that the proposed NBO method has three advantages:

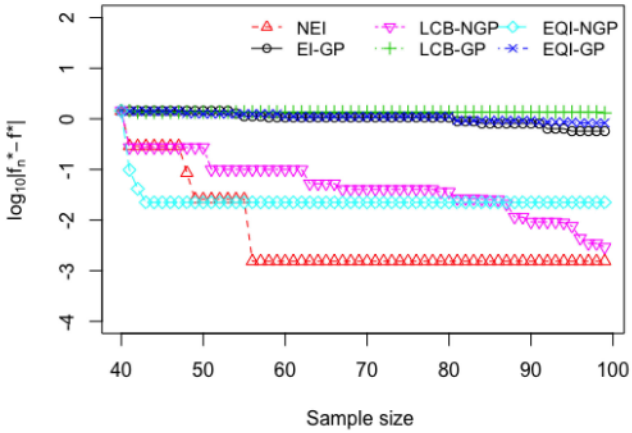

Fig. 8. Average optimality gap over 50 replications by different methods.

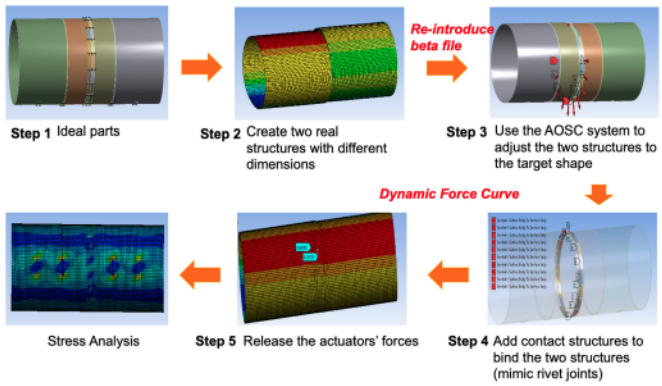

Fig. 9. Computer experiment mimics the composite structures assembly process.

1) it incorporates the nested structure information and makes full use of the inner computer model outputs;
2) it improves the prediction accuracy significantly;
3) it avoids the convergence to local minimum and identifies the global optimum more efficiently.

V. Case Study via Composite Structures Assembly

Composite structures have become increasingly used in many major products (e.g., fuselages, wings, car bodies, solar panels, spacecraft) due to their superior characteristics including high strength-to-weight ratio, high stiffness-to-weight ratio, potential long life, and low life-cycle cost. However, fabrication deviations are inevitable in composite structures. It is timely important to address the quality control in composite structures assembly.

One digital twin simulation platform for composite structures assembly was developed to mimic the fabrication process of carbon-fiber reinforced composites [5], [6]. This computer simulation platform was built based on ANSYS PrepPost Composites workbench, and it was calibrated and validated via physical experiments. The calibration process refers to [42]. The digital twin simulation can conduct virtual assembly to illustrate detailed composite structures joint. As shown in Fig. 9, the virtual assembly simulation includes multiple steps:

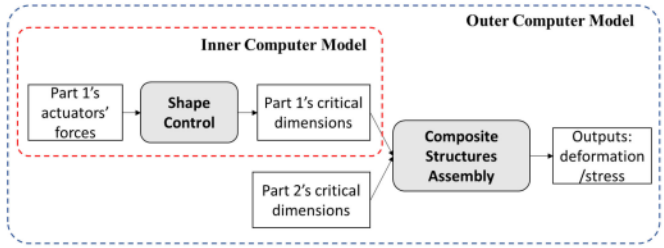

Fig. 10. Nested computer experiments in composite structures assembly.

TABLE I
Inputs and Outputs for the Nested Computer Experiments

Inner computer model	Name of variable	Dimension
Inputs	Part 1's actuators' forces (x)	10
Outputs	Part 1's critical dimensions ($\boldsymbol{h}(x)$)	5
Outer computer model	**Name of variable**	**Dimension**
Inputs	Part 1's critical dimensions ($\boldsymbol{h}(x)$)	5
	Part 2's critical dimensions (x')	5
Outputs	Mean of Stress	1

1) generate composite structures with deviations;
2) apply Automatic Optimal Shape Control technique [43] to adjust the dimensions;
3) add revit joins and then release actuators' forces;
4) do dimensional analysis and stress analysis.

This multistep computer simulation for composite structure assembly has nested structure. As shown in Fig. 10, the inner computer model simulates the shape control of a single composite structure. It can be modeled by Gaussian process [44]. The automatic optimal shape control can adjust the dimensional deviations of one composite fuselage and make it align well with the other fuselage to be assembled. The outer computer model simulates the process of composite structures assembly, where the inputs are critical dimensions from two parts, and the outputs are internal stress after assembly. Table I summarizes the inputs and outputs information in computer experiments. We will conduct nested Bayesian optimization for this nested computer experiment to identify the optimal assembly that can minimize the residual stress after assembly.

Let $n_0 = 100$, we collect the inner computer model outputs H_{n_0} on a maximin Latin hypercube design X_{n_0}, and the outer computer model outputs Y_{n_0} on (H_{n_0}, X_{n_0}). We conduct the twofold CV test and find that the NGP model is non-Gaussian. We split the initial data into 70% as training and 30% as a testing set randomly, and use the training data to build the GP and NGP models. The testing data are used to compare the prediction accuracy of different models.

Fig. 11 shows that the NGP model outperforms the one-GP model. Because the dimension of the inputs is 15, it is time-consuming to search the optimal point of EI and NEI function in Bayesian optimization. Following [45], instead of directly

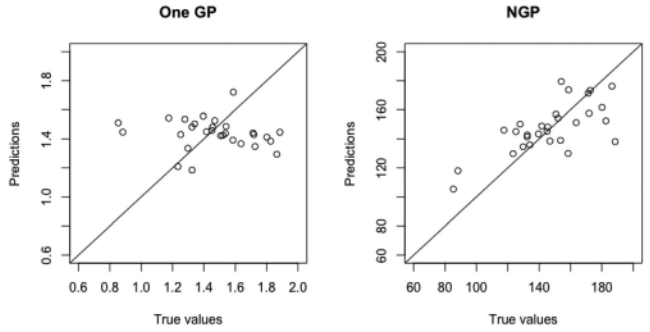

Fig. 11. Predictions given by the GP (left) and NGP (right) versus the true outputs.

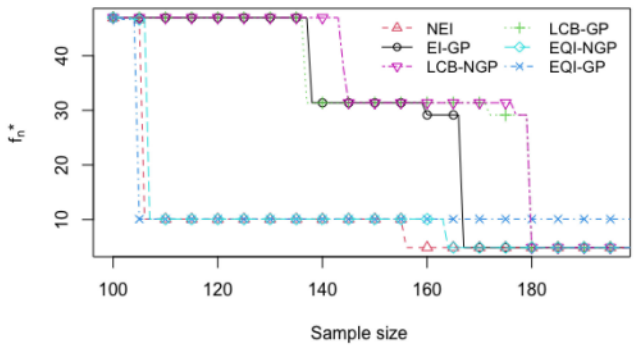

Fig. 12. Optimal results given by different methods.

optimizing the acquisition functions over $\mathcal{X}$, we choose a set of candidate point $\mathcal{X}_{cand}$ from the whole search domain and then find the next point in $\mathcal{X}_{cand}$. In this work, we select $\mathcal{X}_{cand}$ on a maximin Latin hypercube design and the sample size of $\mathcal{X}_{cand}$ is set to be 1000. Let $N = 200$, Fig. 12 shows the optimal results given by different methods.

From Fig. 12, we have that except for the EQI method under one-GP model, the others obtain the same minimum of residual stress with 4.885 psi. Moreover, the proposed method identifies this residual stress with a minimum number of sequential points, which indicates the high effectiveness of the proposed method.

VI. Summary and Discussions

Computer experiments and digital twins have ubiquitous influence on engineering systems. Since the multistep simulations or hierarchical structure of systems, many computer experiments have nested structures. This article proposed a novel Bayesian optimization method for nested computer experiments. We first derived the nested Gaussian process models to serve as surrogates for the computer models. We proved the distribution of nested outputs given it is Gaussian or non-Gaussian. We also deduced the closed forms of nested expected improvement, and proposed one new algorithm for nested Bayesian optimization. The proposed NBO method can make full use of the nested structure and intermediate outputs to identify the global optimum efficiently. It avoids convergence to the local optimum which may occur in standard Bayesian optimization. We validated the performance of NBO based on three numerical studies and one case study. In the case study, the proposed NBO can minimize the residual stress for composite structures assembly, and achieve a much better result than the conventional Bayesian optimization methods.

The proposed method may be faced with generalizability challenge when the system has multiple connected models. Specifically, approximating the multiple nested computer models by a suitable surrogate model needs to estimate more hyperparameters. More training samples will be required for accurate parameter learning. High dimensionality of parameters may result in high computational cost of Bayesian optimization. Furthermore, the fitting multiple connected computer models by a nested GP may have non-identifiability issue. In future research, we will investigate the identifiability conditions and new nested Bayesian optimization methods for complex multiple connected systems.

References

[1] T. J. Santner, B. J. Williams, and W. I. Notz, *The Design and Analysis of Computer Experiments*, 2nd ed. Berlin, Germany: Springer, 2018.
[2] J. Shi, *Stream of Variation Modeling and Analysis for Multistage Manufacturing Processes*. Boca Raton, FL, USA: CRC Press, 2006.
[3] J. Shi, "In-process quality improvement: Concepts, methodologies, and applications," *IISE Trans.*, pp. 1–20, 2022, doi: 10.1080/24725854.2022.2059725.
[4] T. Zhang and J. Shi, "Stream of variation modeling and analysis for compliant composite part assembly—Part II: Multistation processes," *J. Manuf. Sci. Eng.*, vol. 138, no. 12, 2016, Art. no. 121004.
[5] Y. Wen, X. Yue, J. H. Hunt, and J. Shi, "Feasibility analysis of composite fuselage shape control via finite element analysis," *J. Manuf. Syst.*, vol. 46, pp. 272–281, 2018.
[6] Y. Wen, X. Yue, J. H. Hunt, and J. Shi, "Virtual assembly and residual stress analysis for the composite fuselage assembly process," *J. Manuf. Syst.*, vol. 52, pp. 55–62, 2019.
[7] A. S.-M. Rodriguez, M. Hosseini, and J. Paik, "A hybrid control strategy for force and precise end effector positioning of a twisted string actuator," *IEEE/ASME Trans. Mechatronics*, vol. 26, no. 5, pp. 2791–2802, Oct. 2021.
[8] A. Kamadan, G. Kiziltas, and V. Patoglu, "Co-design strategies for optimal variable stiffness actuation," *IEEE/ASME Trans. Mechatronics*, vol. 22, no. 6, pp. 2768–2779, Dec. 2017.
[9] T. Zeng, X. Ren, Y. Zhang, G. Li, and J. Na, "An integrated optimal design for guaranteed cost control of motor driving system with uncertainty," *IEEE/ASME Trans. Mechatronics*, vol. 24, no. 6, pp. 2606–2615, Dec. 2019.
[10] S. Koziel and A. Pietrenko-Dabrowska, "Performance-based nested surrogate modeling of antenna input characteristics," *IEEE Trans. Antennas Propag.*, vol. 67, no. 5, pp. 2904–2912, May 2019.
[11] N. Jin, Y. Zeng, K. Yan, and Z. Ji, "Multivariate air quality forecasting with nested LSTM neural network," *IEEE Trans. Ind. Inform.*, vol. 17, no. 12, pp. 8514–8522, Dec. 2021.
[12] Y. Yu et al., "A nested tensor product model transformation," *IEEE Trans. Fuzzy Syst.*, vol. 27, no. 1, pp. 1–15, Jan. 2019.
[13] P. Z. Qian, B. Tang, and C. J. Wu, "Nested space-filling designs for computer experiments with two levels of accuracy," *Statistica Sinica*, vol. 19, pp. 287–300, 2009.
[14] H. Chen and M.-Q. Liu, "Nested latin hypercube designs with sliced structures," *Commun. Statist. Theory Methods*, vol. 44, no. 22, pp. 4721–4733, 2015.
[15] Y. Hung, V. R. Joseph, and S. N. Melkote, "Design and analysis of computer experiments with branching and nested factors," *Technometrics*, vol. 51, no. 4, pp. 354–365, 2009.
[16] S. Marque-Pucheu, G. Perrin, and J. Garnier, "An efficient dimension reduction for the Gaussian process emulation of two nested codes with functional outputs," *Comput. Statist.*, vol. 35, no. 3, pp. 1059–1099, 2020.

[17] R. H. Keogh and I. R. White, "Using full-cohort data in nested case-control and case-cohort studies by multiple imputation," *Statist. Med.*, vol. 32, no. 23, pp. 4021–4043, 2013.
[18] D. S. Gibson, R. Poddar, G. S. May, and M. A. Brooke, "Using multivariate nested distributions to model semiconductor manufacturing processes," *IEEE Trans. Semicond. Manuf.*, vol. 12, no. 1, pp. 53–65, Feb. 1999.
[19] W. Tian, H. You, K. Gu, C. Zhang, and X. Jia, "Two-level nested control chart for batch process in the semiconductor manufacturing," *IEEE Trans. Semicond. Manuf.*, vol. 29, no. 4, pp. 399–410, Nov. 2016.
[20] R. Jin and J. Shi, "Reconfigured piecewise linear regression tree for multistage manufacturing process control," *IIE Trans.*, vol. 44, no. 4, pp. 249–261, 2012.
[21] S. Savin and L. Vorochaeva, "Nested quadratic programming-based controller for pipeline robots," in *Proc. Int. Conf. Ind. Eng. Appl. Manuf.*, 2017, pp. 1–6.
[22] J. Wang, P. Fu, L. Zhang, R. X. Gao, and R. Zhao, "Multilevel information fusion for induction motor fault diagnosis," *IEEE/ASME Trans. Mechatronics*, vol. 24, no. 5, pp. 2139–2150, Oct. 2019.
[23] M. H. Tan, "Bayesian optimization of expected quadratic loss for multiresponse computer experiments with internal noise," *SIAM/ASA J. Uncertainty Quantification*, vol. 8, no. 3, pp. 891–925, 2020.
[24] A. AlBahar, I. Kim, and X. Yue, "A robust asymmetric kernel function for Bayesian optimization, with application to image defect detection in manufacturing systems," *IEEE Trans. Automat. Sci. Eng.*, pp. 1–12, 2021, doi: 10.1109/TASE.2021.3114157.
[25] L. Shu, P. Jiang, X. Shao, and Y. Wang, "A new multi-objective Bayesian optimization formulation with the acquisition function for convergence and diversity," *J. Mech. Des.*, vol. 142, no. 9, 2020, Art. no. 091703.
[26] A. Biswas, C. Fuentes, and C. Hoyle, "A multi-objective Bayesian optimization approach using the weighted Tchebycheff method," *J. Mech. Des.*, vol. 144, no. 1, 2022, Art. no. 011703.
[27] A. Mathern et al., "Multi-objective constrained Bayesian optimization for structural design," *Struct. Multidisciplinary Optim.*, vol. 63, no. 2, pp. 689–701, 2021.
[28] A. Tran, M. Tran, and Y. Wang, "Constrained mixed-integer Gaussian mixture Bayesian optimization and its applications in designing fractal and auxetic metamaterials," *Struct. Multidisciplinary Optim.*, vol. 59, no. 6, pp. 2131–2154, 2019.
[29] C. Lee, J. Wu, W. Wang, and X. Yue, "Neural network Gaussian process considering input uncertainty for composite structure assembly," *IEEE/ASME Trans. Mechatronics*, vol. 27, no. 3, pp. 1267–1277, Jun. 2022.
[30] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *J. Glob. Optim.*, vol. 13, no. 4, pp. 455–492, 1998.
[31] P. Ranjan, "Comment: EI criteria for noisy computer simulators," *Technometrics*, vol. 55, no. 1, pp. 24–28, 2013.
[32] N. Srinivas, A. Krause, S. Kakade, and M. Seeger, "Gaussian process optimization in the bandit setting: No regret and experimental design," in *Proc. 27th Int. Conf. Mach. Learn.*, 2010, pp. 1015–1022.
[33] V. Picheny, D. Ginsbourger, Y. Richet, and G. Caplin, "Quantile-based optimization of noisy computer experiments with tunable precision," *Technometrics*, vol. 55, no. 1, pp. 2–13, 2013.
[34] R. Astudillo and P. Frazier, "Bayesian optimization of composite functions," in *Proc. 36th Int. Conf. Mach. Learn.*, K. Chaudhuri and R. Salakhutdinov, Eds. Jun. 2019, vol. 97, pp. 354–363.
[35] G. Cui, X. Yu, S. Iommelli, and L. Kong, "Exact distribution for the product of two correlated gaussian random variables," *IEEE Signal Process. Lett.*, vol. 23, no. 11, pp. 1662–1666, Nov. 2016.
[36] S. Ba et al., "Composite Gaussian process models for emulating expensive functions," *Ann. Appl. Statist.*, vol. 6, no. 4, pp. 1838–1860, 2012.
[37] R.-B. Chen, Y. Wang, and C. J. Wu, "Finding optimal points for expensive functions using adaptive RBF-based surrogate model via uncertainty quantification," *J. Glob. Optim.*, vol. 77, pp. 919–948, 2020.
[38] J. L. Loeppky and S. W. J. Welch, "Special issue on computer modeling—choosing the sample size of a computer experiment: A practical guide," *Technometrics*, vol. 51, no. 4, pp. 366–376, 2009.
[39] Y. Jung, "Multiple predicting K-fold cross-validation for model selection," *J. Nonparametric Statist.*, vol. 30, no. 1, pp. 197–215, 2018.
[40] E. Brochu, V. M. Cora, and N. De Freitas, "A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning," 2010. [Online]. Available: http://www.cs.ox.ac.uk/publications/publication7472-abstract.html
[41] O. Roustant, D. Ginsbourger, Y. D. Contributors, and M. O. Roustant, "Package 'dicekriging'," 2015. [Online]. Available: https://cran.r-project.org/web/packages/DiceKriging/index.html
[42] Y. Wang et al., "Effective model calibration via sensible variable identification and adjustment with application to composite fuselage simulation," *Ann. Appl. Statist.*, vol. 14, no. 4, pp. 1759–1776, 2020.
[43] X. Yue, Y. Wen, J. H. Hunt, and J. Shi, "Surrogate model-based control considering uncertainties for composite fuselage assembly," *J. Manuf. Sci. Eng.*, vol. 140, no. 4, 2018, Art. no. 041017.
[44] W. Wang, X. Yue, B. Haaland, and C. F. Jeff Wu, "Gaussian processes with input location error and applications to the composite parts assembly process," *SIAM/ASA J. Uncertainty Quantification*, vol. 10, no. 2, pp. 619–650, 2022.
[45] R.-B. Chen, Y. Wang, and C. F. J. Wu, "Finding optimal points for expensive functions using adaptive RBF-based surrogate model via uncertainty quantification," *J. Glob. Optim.*, vol. 77, no. 4, pp. 919–948, 2020.

Yan Wang received the Ph.D. degree in statistics from the Chinese Academy of Sciences, Beijing, China, in 2018.

She was a Visiting Scholar supervised by Professor Jeff Wu at Georgia Institute of Technology and Professor Tan Matthias Hwai-yong Tan at City University of Hong Kong. She is currently an Assistant Professor with the School of Statistics and Data Science, Beijing University of Technology, Beijing, China. Her research interests include data analytics, computer experiments, and uncertainty quantification for computer models.

Dr. Wang is a recipient of INFORMS QSR Best Paper Finalist Award.

Meng Wang received the B.S. degree in applied statistics from the North China University of Science and Technology, Hebei, China, in 2018. She is currently working toward the master's degree with the School of Statistics and Data Science, Beijing University of Technology, Beijing, China.

Her research interests include computer experiments and uncertainty quantification for computer models.

Areej AlBahar received the B.S. degree in industrial and management systems engineering from Kuwait University, Kuwait, in 2012, the M.S. degree in industrial and systems engineering from the University of Florida, Gainesville, FL, USA, in 2016, and the Ph.D. degree in industrial and systems engineering from Virginia Tech, Blacksburg, VA, USA, in 2022.

She is currently an Assistant Professor with the Department of Industrial and Management Systems Engineering, Kuwait University, Kuwait. Her research interests include machine learning and advanced statistics for monitoring, optimization and quality improvement of manufacturing processes.

Xiaowei Yue (Senior Member, IEEE) received the B.S. degree in mechanical engineering from the Beijing Institute of Technology, Beijing, China, in 2011, the M.S. degree in power engineering and engineering thermophysics from the Tsinghua University, Beijing, China, in 2013, the M.S. degree in statistics, the Ph.D. degree in industrial engineering with minor machine learning from the Georgia Institute of Technology, Atlanta, USA, in 2016 and 2018.

He is currently an Assistant Professor with the Grado Department of Industrial and Systems Engineering, Virginia Tech, Blacksburg, USA. His research interests are focused on engineering-driven data analytics for advanced manufacturing.

Dr. Yue is a recipient of SME Outstanding Young Manufacturing Engineer Award, IISE Manufacturing and Design Division Outstanding Young Investigator Award, more than ten best paper awards. He is a DOD MEEP Faculty fellow. He is a Senior Member of ASQ, IISE, and IEEE, and a member of SME and ASME.