SIAM/ASA J. UNCERTAINTY QUANTIFICATION
Vol. 9, No. 4, pp. 1615–1642

Linked Gaussian Process Emulation for Systems of Computer Models Using Matérn Kernels and Adaptive Design*

Deyu Ming† and Serge Guillas‡

Abstract. The state-of-the-art linked Gaussian process offers a way to build analytical emulators for systems of computer models. We generalize the closed form expressions for the linked Gaussian process under the squared exponential kernel to a class of Matérn kernels that are essential in advanced applications. An iterative procedure to construct linked Gaussian processes as surrogate models for any feed-forward systems of computer models is presented and illustrated on a feed-back coupled satellite system. We also introduce an adaptive design algorithm that could increase the approximation accuracy of linked Gaussian process surrogates with reduced computational costs on running expensive computer systems, by allocating runs and refining emulators of individual submodels based on their heterogeneous functional complexity.

Key words. multidisciplinary, multiphysics, surrogate model, sequential design

AMS subject classifications. 60G15, 62K99, 62M20, 62P30, 65R20

DOI. 10.1137/20M1323771

1. Introduction.

Systems of computer models constitute the new frontier of many scientific and engineering simulations. These can be multiphysics systems of computer simulators such as coupled tsunami simulators with earthquake and landslide sources [23, 32], coupled multiphysics model of the human heart [26], and multidisciplinary systems such as automotive and aerospace systems [7, 16, 34]. Other examples include climate models where climate variability arises from atmospheric, oceanic, land, and cryospheric processes and their coupled interactions [12, 15], or highly multidisciplinary future biodiversity models [31] using combinations of species distribution models, dispersal strategies, climate models, and representative concentration pathways. The number and complexity of computer models involved can hinder the analysis of such systems. For instance, the engineering design optimization of an aerospace system typically requires hundreds of thousands of system evaluations. When the system has feed-backs across computer models, the number of simulations becomes computationally prohibitive [4]. Therefore, building and using a surrogate model is crucial: the system outputs can be predicted at little computational cost, and subsequent sensitivity analysis, uncertainty

*Received by the editors March 13, 2020; accepted for publication (in revised form) June 7, 2021; published electronically November 29, 2021.
https://doi.org/10.1137/20M1323771
Funding: The work of the second author was supported by the Alan Turing Institute project "Uncertainty Quantification of Multi-scale and Multiphysics Computer Models: Applications to Hazard and Climate Models" under the Data-Centric Engineering programme, as part of the grant EP/N510129/1 made to the Alan Turing Institute by EPSRC.

†Department of Statistical Science, University College London, London WC1E 6BT, UK (deyu.ming.16@ucl.ac.uk).
‡Department of Statistical Science, University College London, London WC1E 6BT, UK (s.guillas@ucl.ac.uk), and The Alan Turing Institute, London NW1 2DB, UK (sguillas@turing.ac.uk).

propagation, or inverse modeling can be conducted in a computationally efficient manner.

Gaussian Stochastic process or Gaussian process (GaSP or GP) emulators have gained popularity as surrogate models of systems of computer models in fields including environmental science, biology, and geophysics because of their attractive statistical properties. However, many studies [13, 14, 23, 28, 30] construct global GaSP emulators (named as composite emulators hereinafter) of such systems based on global inputs and outputs without consideration of system structures. One major drawback of such a structural ignorance is that designing experiments can be expensive because system structures may induce high nonlinearity between global inputs and outputs [25]. Furthermore, runs of the whole system are required to produce new training points, even though the overall functional complexity global inputs and outputs originates from a few computer models. This pitfall is particularly undesirable because modern engineering and physical systems can include multiple computer models.

To overcome the disadvantages of the composite emulator, one could construct the surrogate for a system of computer models by integrating GaSP emulators of individual computer models. The idea of integrating GaSP emulators has been explored by [25] in a feed-forward system, but only using the Monte Carlo simulation to approximate the predictive mean and variance of the system output. The Monte Carlo method suffers from a low convergence rate and heavy computational cost, especially when the number of layers in a system is high [20] and the number of new input positions to be evaluated is large, making it prohibitive for complex systems.

Recently, [18] presents a nested emulator that works for systems of two computer models, while [17] derived a more flexible emulator, called linked GaSP, for two-layered feed-forward systems of computer models in analytical form (i.e., closed form expressions for mean and variance of the predicted output of the system at an unexplored input position). However, both of the works are carried out under the assumption that every computer model in the system is represented by a GaSP with a product of one-dimenional (1-D) squared exponential kernels over different input dimensions. Indeed, the squared exponential kernel has been criticized for its oversmoothness [29] and associated ill-conditioned problem [5, 11]. Thus, the generalization of the kernel assumption is necessary. In this study, we generalize the linked GaSP to a class of Matérn kernels for its wider applications in practice. We also demonstrate an iterative procedure, by which the linked GaSP can be constructed for any feed-forward computer systems.

Careful experimental design is important to construct an efficient linked GaSP surrogate under limited computational resources. Poor designs can cause inaccurate linked GaSP with excessive designing cost, and numerical instabilities in training GaSP emulators of individual computer models. Particularly, the linked GaSP is more prone to the latter issue than the composite emulator because the design (e.g., the Latin hypercube design) of the global input can produce poor designs for GaSP emulators of internal computer models. Therefore, we discuss in this work several possible design strategies that can be used for linked GaSP emulation, and introduce an adaptive design algorithm that has the potential to effectively enhance the approximation accuracy of the linked GaSP with improved designs and reduced overall simulation cost.

The remainder of the manuscript is organized as follows. In section 2, we review basics of the GaSP emulator and the linked GaSP. The extension of linked GaSP to Matérn kernels is

then formulated with a synthetic experiment in section 3. An iterative procedure to produce linked GaSPs for any feed-forward computer system is demonstrated with a feed-back coupled satellite model in section 4. In section 5, we introduce an adaptive design strategy for the linked GaSP emulation and discuss its advantages and disadvantages in relating to other alternative designs. Limitations of the linked GaSP are discussed in section 6. We conclude in section 7. Key closed form expressions for the linked GaSP under different kernels and associated proofs are contained in the appendices and supplementary materials, respectively.

2. Review of GaSP emulator and linked GaSP.

In this section, we first give a brief description of GaSP emulators for individual computer models in a computer system. Then the linked GaSP introduced in [17] is reviewed. Note that we present the linked GaSP using our own notation for the benefit of deriving kernel extensions in section 3.

2.1. GaSP emulators for individual computer models.

The GaSP emulator of a computer model considered in this work is itself a collection of GaSP emulators, approximating the functional dependence between the inputs of the computer model and its 1-D outputs. Each 1-D output emulator is constructed independently without the consideration of cross-output dependence, as in [10, 17].

Let $\mathbf{X} \in \mathbb{R}^p$ be a p-dimensional vector of inputs of a computer model, and let $Y(\mathbf{X})$ be the corresponding scalar-valued output. Then, given m sets of inputs $\{\mathbf{X}_1, \ldots, \mathbf{X}_m\}$, the GaSP model is defined by

$$Y(\mathbf{X}_i) = t(\mathbf{X}_i, \mathbf{b}) + \varepsilon_i, \quad i = 1, \ldots, m,$$

where $t(\mathbf{X}_i, \mathbf{b}) = \mathbf{h}(\mathbf{X}_i)^\top \mathbf{b}$ is the trend function with q basis functions $\mathbf{h}(\mathbf{X}_i) = [h_1(\mathbf{X}_i), \ldots, h_q(\mathbf{X}_i)]^\top$ and $\mathbf{b} = [b_1, \ldots, b_q]^\top$; $(\varepsilon_1, \ldots, \varepsilon_m)^\top \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{R})$ with ijth element of the correlation matrix $\mathbf{R}$ given by $R_{ij} = c(\mathbf{X}_i, \mathbf{X}_j) + \eta \mathbb{1}_{\{\mathbf{X}_i = \mathbf{X}_j\}}$, where $c(\cdot, \cdot)$ is a given kernel function; η is the nugget term; and $\mathbb{1}_{\{\cdot\}}$ is the indicator function.

The specification of the kernel function $c(\cdot, \cdot)$ plays an important role in GaSP emulation as it characterizes the sample paths of a GaSP model [29]. In this study we consider the kernel function with the following multiplicative form:

$$c(\mathbf{X}_i, \mathbf{X}_j) = \prod_{k=1}^{p} c_k(X_{ik}, X_{jk}),$$

where $c_k(\cdot, \cdot)$ is a 1-D kernel function for the kth input dimension. Popular candidates for $c_k(\cdot, \cdot)$ are summarized in Table 1. In section 3, we will show that the linked GaSP is applicable to all these aforementioned choices. In the proofs of the supplementary materials, we also consider the additive form of $c(\cdot, \cdot)$.

Assume that the GaSP model parameters σ^2, η, and $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_p)^\top$ are known but $\mathbf{b}$ is a random vector that has a Gaussian distribution with mean $\mathbf{b}_0$ and variance $\tau^2\mathbf{V}_0$. Then, given m inputs $\mathbf{x}^\mathcal{T} = (\mathbf{x}_1^\mathcal{T}, \ldots, \mathbf{x}_m^\mathcal{T})^\top$ and the corresponding outputs $\mathbf{y}^\mathcal{T} = (y_1^\mathcal{T}, \ldots, y_m^\mathcal{T})^\top$, the GaSP emulator of the computer model is defined by the predictive distribution of $Y(\mathbf{x}_0)$ (i.e., conditional distribution of $Y(\mathbf{x}_0)$ given $\mathbf{y}^\mathcal{T}$) at a new input position $\mathbf{x}_0$ [27], which is

$$Y(\mathbf{x}_0)|\mathbf{y}^\mathcal{T} \sim \mathcal{N}(\mu_0(\mathbf{x}_0), \sigma_0^2(\mathbf{x}_0)) \tag{2.1}$$

Table 1

Choices of $c_k(\cdot,\cdot)$. $\gamma_k > 0$ is the range parameter for the kth input dimension.

Exponential	$c_k(\cdot,\cdot) = \exp\left\{-\frac{\lvert X_{ik}-X_{jk}\rvert}{\gamma_k}\right\}$,
Squared Exponential	$c_k(\cdot,\cdot) = \exp\left\{-\frac{(X_{ik}-X_{jk})^2}{\gamma_k^2}\right\}$,
Matérn-1.5	$c_k(\cdot,\cdot) = \left(1+\frac{\sqrt{3}\lvert X_{ik}-X_{jk}\rvert}{\gamma_k}\right)\exp\left\{-\frac{\sqrt{3}\lvert X_{ik}-X_{jk}\rvert}{\gamma_k}\right\}$,
Matérn-2.5	$c_k(\cdot,\cdot) = \left(1+\frac{\sqrt{5}\lvert X_{ik}-X_{jk}\rvert}{\gamma_k}+\frac{5(X_{ik}-X_{jk})^2}{3\gamma_k^2}\right)\exp\left\{-\frac{\sqrt{5}\lvert X_{ik}-X_{jk}\rvert}{\gamma_k}\right\}$

with

$$\mu_0(\mathbf{x}_0) = \mathbf{h}(\mathbf{x}_0)^\top \widehat{\mathbf{b}} + \mathbf{r}(\mathbf{x}_0)^\top \mathbf{R}^{-1}\left(\mathbf{y}^{\mathcal{T}} - \mathbf{H}(\mathbf{x}^{\mathcal{T}})\widehat{\mathbf{b}}\right), \tag{2.2}$$

$$\begin{aligned}\sigma_0^2(\mathbf{x}_0) = \sigma^2\Big[1+\eta-\mathbf{r}(\mathbf{x}_0)^\top\mathbf{R}^{-1}\mathbf{r}(\mathbf{x}_0)+\Big(\mathbf{h}(\mathbf{x}_0)-\mathbf{H}(\mathbf{x}^{\mathcal{T}})^\top\mathbf{R}^{-1}\mathbf{r}(\mathbf{x}_0)\Big)^\top \\ \times\left(\mathbf{H}(\mathbf{x}^{\mathcal{T}})^\top\mathbf{R}^{-1}\mathbf{H}(\mathbf{x}^{\mathcal{T}})+\frac{\sigma^2}{\tau^2}\mathbf{V}_0^{-1}\right)^{-1}\Big(\mathbf{h}(\mathbf{x}_0)-\mathbf{H}(\mathbf{x}^{\mathcal{T}})^\top\mathbf{R}^{-1}\mathbf{r}(\mathbf{x}_0)\Big)\Big],\end{aligned} \tag{2.3}$$

where $\mathbf{r}(\mathbf{x}_0) = [c(\mathbf{x}_0,\mathbf{x}_1^{\mathcal{T}}),\ldots,c(\mathbf{x}_0,\mathbf{x}_m^{\mathcal{T}})]^\top$, $\mathbf{H}(\mathbf{x}^{\mathcal{T}}) = [\mathbf{h}(\mathbf{x}_1^{\mathcal{T}}),\ldots,\mathbf{h}(\mathbf{x}_m^{\mathcal{T}})]^\top$ and

$$\widehat{\mathbf{b}} \overset{\text{def}}{=\!=} \left(\mathbf{H}(\mathbf{x}^{\mathcal{T}})^\top\mathbf{R}^{-1}\mathbf{H}(\mathbf{x}^{\mathcal{T}})+\frac{\sigma^2}{\tau^2}\mathbf{V}_0^{-1}\right)^{-1}\left(\mathbf{H}(\mathbf{x}^{\mathcal{T}})^\top\mathbf{R}^{-1}\mathbf{y}^{\mathcal{T}}+\frac{\sigma^2}{\tau^2}\mathbf{V}_0^{-1}\mathbf{b}_0\right).$$

Let $\tau^2 \to \infty$ (i.e., the Gaussian distribution of $\mathbf{b}$ gets more and more noninformative), then all terms associated with $\mathbf{b}_0$ and $\mathbf{V}_0$ in (2.2) and (2.3) become increasingly insignificant and thus we obtain the GaSP emulator defined by the predictive distribution of $Y(\mathbf{x}_0)$ with its mean and variance given by

$$\mu_0(\mathbf{x}_0) = \mathbf{h}(\mathbf{x}_0)^\top \widehat{\mathbf{b}} + \mathbf{r}(\mathbf{x}_0)^\top \mathbf{R}^{-1}\left(\mathbf{y}^{\mathcal{T}} - \mathbf{H}(\mathbf{x}^{\mathcal{T}})\widehat{\mathbf{b}}\right), \tag{2.4}$$

$$\begin{aligned}\sigma_0^2(\mathbf{x}_0) = \sigma^2\Big[1+\eta-\mathbf{r}(\mathbf{x}_0)^\top\mathbf{R}^{-1}\mathbf{r}(\mathbf{x}_0)+\Big(\mathbf{h}(\mathbf{x}_0)-\mathbf{H}(\mathbf{x}^{\mathcal{T}})^\top\mathbf{R}^{-1}\mathbf{r}(\mathbf{x}_0)\Big)^\top \\ \times\left(\mathbf{H}(\mathbf{x}^{\mathcal{T}})^\top\mathbf{R}^{-1}\mathbf{H}(\mathbf{x}^{\mathcal{T}})\right)^{-1}\Big(\mathbf{h}(\mathbf{x}_0)-\mathbf{H}(\mathbf{x}^{\mathcal{T}})^\top\mathbf{R}^{-1}\mathbf{r}(\mathbf{x}_0)\Big)\Big]\end{aligned} \tag{2.5}$$

with $\widehat{\mathbf{b}} \overset{\text{def}}{=\!=} \left[\mathbf{H}(\mathbf{x}^{\mathcal{T}})^\top\mathbf{R}^{-1}\mathbf{H}(\mathbf{x}^{\mathcal{T}})\right]^{-1}\mathbf{H}(\mathbf{x}^{\mathcal{T}})^\top\mathbf{R}^{-1}\mathbf{y}^{\mathcal{T}}$, where $\mu_0(\mathbf{x}_0)$ and $\sigma_0^2(\mathbf{x}_0)$ match the best linear unbiased predictor (BLUP) of $Y(\mathbf{x}_0)$ and its mean squared error [29]. In the remainder of the study we use the predictive distribution with mean and variance given in (2.4) and (2.5) as the GaSP emulator of a computer model. Note that the GaSP model parameters σ^2, η, and $\boldsymbol{\gamma} = (\gamma_1,\ldots,\gamma_p)^\top$ in (2.4) and (2.5) are typically unknown and need to be estimated. One may estimate these parameters by solving the objective function

$$(\widehat{\eta},\widehat{\boldsymbol{\gamma}}) = \underset{\eta,\boldsymbol{\gamma}}{\operatorname{argmax}}\,\mathcal{L}(\widehat{\sigma^2},\eta,\boldsymbol{\gamma}),$$

where

$$\mathcal{L}(\widehat{\sigma^2}, \eta, \boldsymbol{\gamma}) = \frac{|\mathbf{R}|^{-\frac{1}{2}}|\mathbf{H}(\mathbf{x}^{\mathcal{T}})^{\top}\mathbf{R}^{-1}\mathbf{H}(\mathbf{x}^{\mathcal{T}})|^{-\frac{1}{2}}}{(2\pi\widehat{\sigma^2})^{\frac{m-q}{2}}} \times \exp\left\{-\frac{1}{2\widehat{\sigma^2}}\left(\mathbf{y}^{\mathcal{T}} - \mathbf{H}(\mathbf{x}^{\mathcal{T}})\widehat{\mathbf{b}}\right)^{\top}\mathbf{R}^{-1}\left(\mathbf{y}^{\mathcal{T}} - \mathbf{H}(\mathbf{x}^{\mathcal{T}})\widehat{\mathbf{b}}\right)\right\}$$

is the marginal likelihood obtained by integrating out $\mathbf{b}$ from the full likelihood function $\mathcal{L}(\mathbf{b}, \sigma^2, \eta, \boldsymbol{\gamma})$ and have σ^2 replaced by its maximum likelihood estimator

$$\widehat{\sigma^2} = \frac{1}{m-q}\left(\mathbf{y}^{\mathcal{T}} - \mathbf{H}(\mathbf{x}^{\mathcal{T}})\widehat{\mathbf{b}}\right)^{\top}\mathbf{R}^{-1}\left(\mathbf{y}^{\mathcal{T}} - \mathbf{H}(\mathbf{x}^{\mathcal{T}})\widehat{\mathbf{b}}\right) \tag{2.6}$$

with $\widehat{\mathbf{b}} \overset{\text{def}}{=\!=} \left[\mathbf{H}(\mathbf{x}^{\mathcal{T}})^{\top}\mathbf{R}^{-1}\mathbf{H}(\mathbf{x}^{\mathcal{T}})\right]^{-1}\mathbf{H}(\mathbf{x}^{\mathcal{T}})^{\top}\mathbf{R}^{-1}\mathbf{y}^{\mathcal{T}}$. Alternatively, the maximum a posterior (MAP) method is a more robust estimation technique [11]. It maximizes the marginal posterior mode with respect to the objective function

$$(\widehat{\eta}, \widehat{\boldsymbol{\gamma}}) = \underset{\eta, \boldsymbol{\gamma}}{\operatorname{argmax}}\, \mathcal{L}(\widehat{\sigma^2}, \eta, \boldsymbol{\gamma})\pi(\eta, \boldsymbol{\gamma}), \tag{2.7}$$

where $\pi(\eta, \boldsymbol{\gamma})$ is the reference prior; see [11] for different choices and parameterizations.

After the estimates of σ^2, η, and $\boldsymbol{\gamma}$ are obtained, they are plugged into the predictive distribution mean (2.4) and variance (2.5), forming the empirical GaSP emulator of a computer model. In the remainder of the study, all GaSP models of individual computer models are estimated using the MAP method via the `R` package `RobustGaSP`. Note that `RobustGaSP` in fact estimates η and $\boldsymbol{\gamma}$ with the marginal likelihood obtained by integrating out both $\mathbf{b}$ and σ^2. However, as demonstrated in [1] the estimates of η and $\boldsymbol{\gamma}$ are not influenced by the integration of σ^2. As a result, we can implement `RobustGaSP` to obtain the estimates of η and $\boldsymbol{\gamma}$ produced by the discussed MAP method and then have them plugged into (2.6) to obtain the estimate of σ^2.

2.2. Linked GaSP. Consider a two-layered system of computer models, where the computer models in the first layer produce collectively d-dimensional output that feeds into a computer model in the second layer. Let $\mathbf{W} = [W_1(\mathbf{x}_1), \ldots, W_d(\mathbf{x}_d)]^{\top}$ be the collection of the d-dimensional output produced by d GaSP emulators $\widehat{f}_1, \ldots, \widehat{f}_d$ of computer models in the first layer given the input positions $\mathbf{x}_1, \ldots, \mathbf{x}_d$. Denote $\widehat{g}$ as the GaSP emulator of the computer model g in the second layer, producing $Y(\mathbf{W}, \mathbf{z})$ that approximates a scalar-valued output of g at inputs $\mathbf{W}$ from $\widehat{f}_1, \ldots, \widehat{f}_d$ and exogenous inputs $\mathbf{z} = (z_1, \ldots, z_p)^{\top}$. Then the emulation of the two-layered system aims to link GaSP emulators connected as shown in Figure 1.

Perhaps the most straightforward way to build an emulator of the system is to obtain the predictive distribution of $Y(\mathbf{x}_1, \ldots, \mathbf{x}_d, \mathbf{z})$, given the global inputs $\mathbf{x}_1, \ldots, \mathbf{x}_d$ and $\mathbf{z}$. This predictive distribution, named as linked emulator by [17], is naturally defined by the probability density function

$$p(y|\mathbf{x}_1, \ldots, \mathbf{x}_d, \mathbf{z}) = \int_{\mathbf{w}} p(y|\mathbf{w}, \mathbf{z})\, p(\mathbf{w}|\mathbf{x}_1, \ldots, \mathbf{x}_d)\, \mathrm{d}\mathbf{w}, \tag{2.8}$$

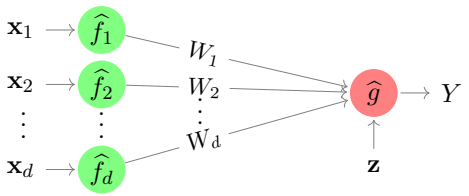


Figure 1. *The connections of emulators to be linked for emulating a two-layered computer system. $\widehat{f}_1, \widehat{f}_2 \ldots, \widehat{f}_d$ are 1-D emulators approximating d outputs from computer models in the first layer; $\widehat{g}$ is a 1-D GaSP emulator approximating a scalar-valued output of the computer model g in the second layer of the system.*

where $\mathbf{w} = (w_1, \ldots, w_d)^\top$. However, $p(y|\mathbf{x}_1, \ldots, \mathbf{x}_d, \mathbf{z})$ is neither analytically tractable nor Gaussian in general. One might compute the integral in (2.8) numerically or simply generate realizations of $Y(\mathbf{x}_1, \ldots, \mathbf{x}_d, \mathbf{z})$ by sampling sequentially from Gaussian densities $p(y|\mathbf{w}, \mathbf{z})$ and $p(\mathbf{w}|\mathbf{x}_1, \ldots, \mathbf{x}_d)$, and then use the resulting density or sampled realizations as the linked emulator. However, such approaches are computationally expensive and can soon become prohibitive for many uncertainty analysis as the dimensions of $\mathbf{x}_{i=1,\ldots,d}$ and $\mathbf{w}$ increase. Fortunately, [17] shows that under some mild conditions, the mean, and variance of the linked emulator can be calculated analytically, and its Gaussian approximation, called linked GaSP, is a Gaussian distribution with matching mean and variance. One of the key conditions that [17] makes for the closed form mean and variance of the inked emulator is that the GaSP emulator $\widehat{g}$ is constructed under the squared exponential kernel. However, it is well known that the squared exponential kernel can have computational difficulties both in theory and practice [29, 5, 11], limiting broader applications of the linked GaSP. In section 3, we relax this kernel limitation and show that there exists closed form expressions for the mean and variance of the linked emulator under a class of Matérn kernels.

3. Generalization of linked GaSP to Matérn kernels.

Assume that the GaSP emulator $\widehat{g}$ is built with m training points $\mathbf{w}^{\mathcal{T}} = (\mathbf{w}_1^{\mathcal{T}}, \ldots, \mathbf{w}_m^{\mathcal{T}})^\top$, $\mathbf{z}^{\mathcal{T}} = (\mathbf{z}_1^{\mathcal{T}}, \ldots, \mathbf{z}_m^{\mathcal{T}})^\top$, and $\mathbf{y}^{\mathcal{T}} = (y_1^{\mathcal{T}}, \ldots, y_m^{\mathcal{T}})^\top$, where $\mathbf{w}_i^{\mathcal{T}} = (w_{i1}^{\mathcal{T}}, \ldots, w_{id}^{\mathcal{T}})^\top$ and $\mathbf{z}_i^{\mathcal{T}} = (z_{i1}^{\mathcal{T}}, \ldots, z_{ip}^{\mathcal{T}})^\top$ for all $i = 1, \ldots, m$. We further make the following two assumptions.

Assumption 3.1. The trend function $t(\mathbf{W}, \mathbf{z}, \boldsymbol{\theta}, \boldsymbol{\beta})$ in the GaSP model for the computer model g is specified by $t(\mathbf{W}, \mathbf{z}, \boldsymbol{\theta}, \boldsymbol{\beta}) = \mathbf{W}^\top \boldsymbol{\theta} + \mathbf{h}(\mathbf{z})^\top \boldsymbol{\beta}$, where

- $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_d)^\top$ and $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_q)^\top$;
- $\mathbf{h}(\mathbf{z}) = [h_1(\mathbf{z}), \ldots, h_q(\mathbf{z})]^\top$ are basis functions of $\mathbf{z}$.

Assumption 3.2. $W_k(\mathbf{x}_k) \overset{ind}{\sim} \mathcal{N}(\mu_k(\mathbf{x}_k), \sigma_k^2(\mathbf{x}_k))$ for $k = 1, \ldots, d$.

We can then derive in closed form the mean and variance of linked emulator subject to the choice of 1-D kernel functions used in GaSP emulator $\widehat{g}$.

Theorem 3.3. *Under Assumptions 3.1 and 3.2, the output $Y(\mathbf{x}_1, \ldots, \mathbf{x}_d, \mathbf{z})$ of the linked emulator at the input positions $\mathbf{x}_1, \ldots, \mathbf{x}_d$ and $\mathbf{z}$ has analytical mean μ_L and variance σ_L^2*

given by

$$\mu_L = \boldsymbol{\mu}^\top \widehat{\boldsymbol{\theta}} + \mathbf{h}(\mathbf{z})^\top \widehat{\boldsymbol{\beta}} + \mathbf{I}^\top \mathbf{A}, \tag{3.1}$$

$$\begin{aligned}\sigma_L^2 =& \mathbf{A}^\top \left(\mathbf{J} - \mathbf{I}\mathbf{I}^\top\right) \mathbf{A} + 2\widehat{\boldsymbol{\theta}}^\top \left(\mathbf{B} - \boldsymbol{\mu}\mathbf{I}^\top\right) \mathbf{A} + \operatorname{tr}\left\{\widehat{\boldsymbol{\theta}}\widehat{\boldsymbol{\theta}}^\top \boldsymbol{\Omega}\right\} \\ &+ \sigma^2 \left(1 + \eta + \operatorname{tr}\{\mathbf{Q}\mathbf{J}\} + \mathbf{G}^\top \mathbf{C}\mathbf{G} + \operatorname{tr}\left\{\mathbf{C}\mathbf{P} - 2\mathbf{C}\widetilde{\mathbf{H}}^\top \mathbf{R}^{-1}\mathbf{K}\right\}\right),\end{aligned} \tag{3.2}$$

where

- $\boldsymbol{\mu} = [\mu_1(\mathbf{x}_1), \ldots, \mu_d(\mathbf{x}_d)]^\top$ *and* $\left[\widehat{\boldsymbol{\theta}}^\top, \widehat{\boldsymbol{\beta}}^\top\right]^\top \overset{\text{def}}{=\!=} \left(\widetilde{\mathbf{H}}^\top \mathbf{R}^{-1} \widetilde{\mathbf{H}}\right)^{-1} \widetilde{\mathbf{H}}^\top \mathbf{R}^{-1} \mathbf{y}^{\mathcal{T}}$;
- $\boldsymbol{\Omega} = \operatorname{diag}(\sigma_1^2(\mathbf{x}_1), \ldots, \sigma_d^2(\mathbf{x}_d))$ *and* $\mathbf{P} = \operatorname{blkdiag}(\boldsymbol{\Omega}, \mathbf{0})$;
- $\mathbf{A} = \mathbf{R}^{-1}\left(\mathbf{y}^{\mathcal{T}} - \mathbf{w}^{\mathcal{T}}\widehat{\boldsymbol{\theta}} - \mathbf{H}(\mathbf{z}^{\mathcal{T}})\widehat{\boldsymbol{\beta}}\right)$ *with* $\mathbf{H}(\mathbf{z}^{\mathcal{T}}) = [\mathbf{h}(\mathbf{z}_1^{\mathcal{T}}), \ldots, \mathbf{h}(\mathbf{z}_m^{\mathcal{T}})]^\top$;
- $\mathbf{Q} = \mathbf{R}^{-1}\widetilde{\mathbf{H}}\left(\widetilde{\mathbf{H}}^\top \mathbf{R}^{-1}\widetilde{\mathbf{H}}\right)^{-1} \widetilde{\mathbf{H}}^\top \mathbf{R}^{-1} - \mathbf{R}^{-1}$ *with* $\widetilde{\mathbf{H}} = \left[\mathbf{w}^{\mathcal{T}}, \mathbf{H}(\mathbf{z}^{\mathcal{T}})\right]$;
- $\mathbf{G} = [\boldsymbol{\mu}^\top, \mathbf{h}(\mathbf{z})^\top]^\top$, $\mathbf{C} = \left(\widetilde{\mathbf{H}}^\top \mathbf{R}^{-1}\widetilde{\mathbf{H}}\right)^{-1}$ *and* $\mathbf{K} = \left[\mathbf{B}^\top, \mathbf{I}\mathbf{h}(\mathbf{z})^\top\right]$;
- $\mathbf{I}$ *is a* $m \times 1$ *column vector with the* i*th element given by*

$$I_i = \prod_{k=1}^{p} c_k(z_k, z_{ik}^{\mathcal{T}}) \prod_{k=1}^{d} \xi_{ik},$$

  where $\xi_{ik} \overset{\text{def}}{=\!=} \mathbb{E}\left[c_k(W_k(\mathbf{x}_k), w_{ik}^{\mathcal{T}})\right]$;
- $\mathbf{J}$ *is a* $m \times m$ *matrix with the* ij*th element given by*

$$J_{ij} = \prod_{k=1}^{p} c_k(z_k, z_{ik}^{\mathcal{T}})\, c_k(z_k, z_{jk}^{\mathcal{T}}) \prod_{k=1}^{d} \zeta_{ijk},$$

  where $\zeta_{ijk} \overset{\text{def}}{=\!=} \mathbb{E}\left[c_k(W_k(\mathbf{x}_k), w_{ik}^{\mathcal{T}})\, c_k(W_k(\mathbf{x}_k), w_{jk}^{\mathcal{T}})\right]$;
- $\mathbf{B}$ *is a* $d \times m$ *matrix with the* lj*th element given by*

$$B_{lj} = \psi_{jl} \prod_{\substack{k=1 \\ k \neq l}}^{d} \xi_{jk} \prod_{k=1}^{p} c_k(z_k, z_{jk}^{\mathcal{T}}),$$

  where $\psi_{jl} \overset{\text{def}}{=\!=} \mathbb{E}\left[W_l(\mathbf{x}_l)\, c_l(W_l(\mathbf{x}_l), w_{jl}^{\mathcal{T}})\right]$.

Proof. The proof is in section SM4 of the supplementary materials. ∎

Proposition 3.4. *The three expectations* ξ_{ik}, ζ_{ijk}, *and* ψ_{jl} *defined in Theorem 3.3 have closed form expressions for the squared exponential kernel and a class of Matérn kernels [21] defined by*

$$c_k(d_{ij,k}) = \exp\left(-\frac{\sqrt{2p+1}\, d_{ij,k}}{\gamma_k}\right) \frac{p!}{(2p)!} \sum_{i=0}^{p} \frac{(p+i)!}{i!(p-i)!} \left(\frac{2d_{ij,k}\sqrt{2p+1}}{\gamma_k}\right)^{p-i}, \tag{3.3}$$

where $d_{ij,k} = X_{ik} - X_{jk}$ *and* p *is a nonnegative integer.*

Proof. Derivations for the squared exponential kernel, Matérn kernels (3.3) with $p = 0$ (exponential), $p = 1$ (Matérn-1.5), and $p = 2$ (Matérn-2.5) are detailed in section SM5 of the supplementary materials. The corresponding closed form expressions are summarized in Appendices A to D. The closed form expressions for Matérn kernels with $p \geq 3$ can be obtained straightforwardly by invoking Lemma SM5.1 of the supplementary materials and using same arguments in proofs of Matérn-1.5 and Matérn-2.5. Note that we reproduce the result for the squared exponential kernel given in [17] using our own notation for completeness. ∎

3.1. A synthetic experiment. Consider the computer system shown in Figure 2, which consists three computer models with the following analytical functional forms:

$$f_1 = 30 + 5x_1 \sin(5x_1), \quad f_2 = 4 + \exp(-5x_2), \quad \text{and} \quad f_3 = (w_1 w_2 - 100)/6$$

with $x_1 \in [0, 2]$ and $x_2 \in [0, 2]$.

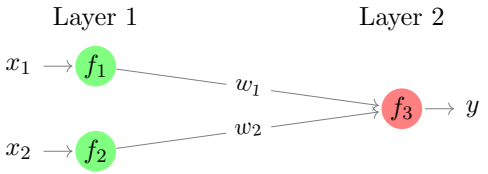


Figure 2. *The computer system in the synthetic experiment where f_1 and f_2 are two computer models with 1-D input and output, and f_3 is a computer model with two-dimensional (2-D) input and 1-D output.*

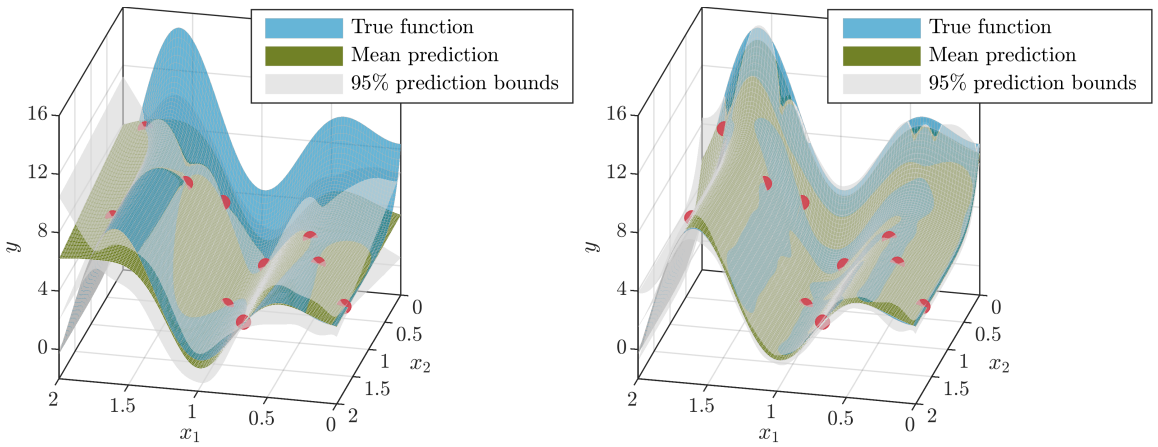
We generate ten training points from the maximin Latin hypercube and construct the composite emulator (Figure 3a) and linked GaSP (Figure 3b) of the system with Matérn-2.5 kernel. Figure 3b indicates that the Matérn extension to the linked GaSP is valid because the constructed linked GaSP interpolates training points with sensible predictive mean and bounds.

We further compare the linked GaSP with composite emulator with Matérn-2.5 kernel at different training sizes in Figure 4a. At each selected training set size, normalized root mean squared error of prediction (NRMSEP) of both composite emulator and linked GaSP are calculated, where

$$\text{NRMSEP} = \frac{\sqrt{\frac{1}{nT}\sum_{t=1}^{T}\sum_{i=1}^{n}(y(\mathbf{x}_i) - \mu_Y^t(\mathbf{x}_i))^2}}{\max\{y(\mathbf{x}_i)_{i=1,\ldots,n}\} - \min\{y(\mathbf{x}_i)_{i=1,\ldots,n}\}}, \tag{3.4}$$

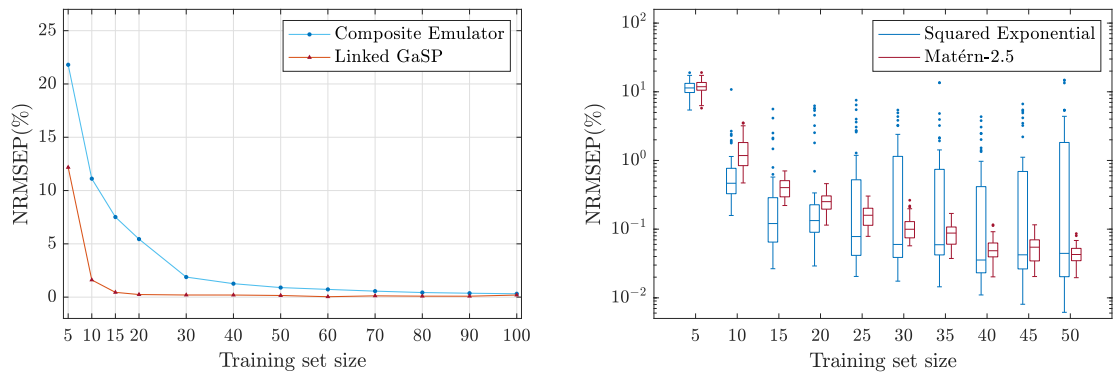
in which $y(\mathbf{x}_i)$ denotes the true global output of the system evaluated at the testing input position $\mathbf{x}_i$ for $i = 1, \ldots, n$ with $n = 2500$, which are equally spaced over the global input domain $[0, 2]\times[0, 2]$; $\mu_Y^t(\mathbf{x}_i)$ is the mean prediction of the respective emulator built with the tth design of total $T = 100$ designs sampled from the maximin Latin hypercube. Both Figures 3 and 4a show that the linked GaSP outperforms (in terms of mean predictions, prediction bounds, NRMSEP, and training cost) the composite emulator under the Matérn-2.5 kernel.

In Figure 4b, NRMSEP between linked GaSPs with squared exponential and Matérn-2.5 kernels are compared under ten different training set sizes. At each selected training set size, NRMSEPs are computed (without averaging over T in (3.4)) for $T = 50$ random designs drawn



(a) Composite emulator (b) Linked GaSP

Figure 3. *The composite emulator and linked GaSP of the system in Figure 2. The filled circles are training points used to construct the emulators.*



(a) Composite emulator versus Linked GaSP (b) Squared exponential versus Matérn-2.5

Figure 4. *Emulation results for the system in Figure 2.* (a) *NRMSEP of composite emulator and linked GaSP with Matérn-2.5 kernel;* (b) *NRMSEP of linked GaSPs with squared exponential and Matérn-2.5 kernels, both with a small nugget to handle ill-conditioned correlation matrices whenever necessary. NRMSEP in* (b) *is shown under the log-scale.*

from the maximin Latin hypercube. The NRMSEP of the linked GaSP with Matérn-2.5 kernel decays steadily as the training set size increases and its predictive performance is robust across different designs. On the contrary, NRMSEP of the linked GaSP with squared exponential kernel decreases with increasing oscillations over designs. Particularly, as the training set size increases beyond 15, the linked GaSP with squared exponential kernel exhibits increasing chances of NRMSEPs over 1.0% with extreme NRMSEPs reaching 5–10% for some designs, whereas the linked GaSP with Matérn-2.5 kernel consistently provides NRMSEPs lower than 0.5–1.0%. The large fluctuations of NRMSEPs displayed in the squared exponential case

are due to the GaSP emulator $\widehat{f_3}$ that cannot capture adequately the true functional form of f_3 under some designs with the squared exponential kernel. It is also worth noting that in constructing GaSP emulators of individual computer models we experience ill-conditioned correlation matrices (which are subsequently addressed by enhancing their diagonal elements with a small nugget term) more frequently with the squared exponential kernel than the Matérn-2.5 kernel. These results stress the importance of Matérn extensions to the linked GaSP, in agreement with [11, 9] that Matérn kernels are less vulnerable to ill-conditioning issues, provide reasonably adequate choices on the smoothness, and have both attractive theoretical properties and good practical performance. Furthermore, in practice, Matérn-1.5 and Matérn-2.5 are included in several computer emulation packages, such as `DiceKriging` and `RobustGaSP`, where Matérn-2.5 is the default kernel choice. In the remainder of the study, Matérn-2.5 is thus used for all GaSP emulator constructions.

4. Construction of linked GaSP for multilayered computer systems. In this section, we demonstrate how to construct linked GaSP for a multilayered system with feed-forward hierarchy, in which the outputs of lower-layer computer models act as the inputs of higher-layer ones.

It is a challenging analytical work to construct linked GaSP for a multilayered feed-forward system in one-shot because there exists no closed form expressions for the mean and variance of the linked emulator, whose density function involves integration of GaSP emulators across a large number of layers. However, one could collapse a complex feed-forward system into a sequence of two-layered computer systems, and then successively construct linked GaSPs across two layers.

Consider a general feed-forward system of computer models, denoted by $e_{1\to L}$, with L layers. The system can be decomposed into a sequence of $L-1$ subsystems: $e_{1\to(i+1)}$ for $i = 1, \ldots, L-1$. Then, the linked GaSP of the whole system ($e_{1\to L}$) is built by the following steps:

1. Construct the linked GaSP of $e_{1\to 2}$ by applying Theorem 3.3 to GaSP emulators of computer models in the first and second layers of $e_{1\to L}$;
2. For $i = 2, \ldots, L-1$, construct the linked GaSP of $e_{1\to i+1}$ by applying Theorem 3.3 to the linked GaSP of $e_{1\to i}$ and GaSP emulators of computer models in the $(i+1)$th layer of $e_{1\to L}$;

For example, the system in Figure 5 can be decomposed into three recursive systems: $e_{1\to 2}$, $e_{1\to 3}$, and $e_{1\to 4}$, and the linked GaSP of the whole system $e_{1\to 4}$ takes three iterations to be produced. It is noted that the above iterative procedure works because Assumption 3.2 only requires normality while it has no constraints on specific forms of corresponding mean and variance.

4.1. Linked GaSP for a feed-back coupled satellite model. In this section, we show the construction of the linked GaSP for a multilayered fire-detection satellite model studied in [24]. This satellite is designed to conduct near-real-time detection, identification, and monitoring of forest fires. The satellite system consists of three submodels, namely the orbit analysis, the attitude control, and power analysis. The satellite system is shown in Figure 6. It can be seen from Figure 6 that there are nine global input variables H, F_s, θ, L_{sp}, q, R_D, L_a, C_d, P_{other} and three global output variables of interest τ_{tot}, P_{tot}, A_{sa}. The coupling variables are Δt_{orbit},

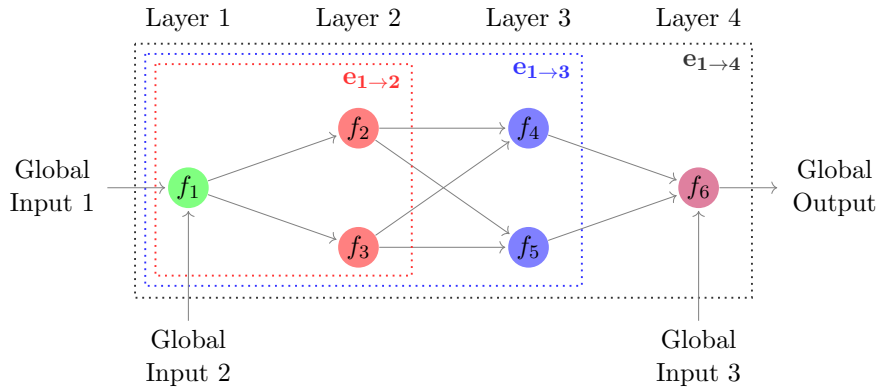


Figure 5. *An illustration on the iterative procedure to construct linked GaSP for a 4-layered feed-forward computer system.*

$\Delta t_{eclipse}$, ν, θ_{slew}, P_{ACS}, I_{max}, and I_{min}. Since Δt_{orbit} is the input to both power analysis and attitude control, there are a total of eight coupling variables. Note that the system has feed-back coupling because the coupling variables P_{ACS}, I_{max}, and I_{min} form an internal loop between power analysis and attitude control. Therefore, to implement the iterative procedure to build the linked GaSP of the system, we first convert the system to a feed-forward one by applying the decoupling algorithm proposed in [2]. The decoupling algorithm identifies four weakly coupled variables Δt_{orbit} (between orbit analysis and attitude control), θ_{slew}, I_{max}, and I_{min}. Since the weakly coupled variables have insignificant impact on the accuracy of global outputs, they are neglected from the interaction terms between submodels, producing a feed-forward system (see Figure 6 without the dashed arrows). Table 2 gives the domains of global inputs considered for the emulation.

Maximin Latin hypercube sampling is then used to generate inputs positions for seven training sets, with sizes of 10, 15, 20, 25, 30, 35, and 40 respectively. The corresponding output positions are consequently obtained by running the satellite model. For each of the seven training sets and each of the three global output variables, we build the composite emulator and linked GaSP. Leave-one-out cross-validation is utilized for assessing the predictive performance of the two emulators. For example, in the case of the composite emulation of the output variable P_{tot} with training set size of 10, we build ten composite emulators, each based on nine training points by dropping one training point out of the set. The dropped training point is then serves as the testing point to assess the associated composite emulator. The performance of the emulator (composite emulator or linked GaSP) of a global output variable given a certain training set is ultimately summarized by

$$\text{NRMSEP} = \frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(f(\mathbf{x}_i) - \mu^{-i}(\mathbf{x}_i))^2}}{\max\{f(\mathbf{x}_i)_{i=1,\ldots,n}\} - \min\{f(\mathbf{x}_i)_{i=1,\ldots,n}\}},$$

where $\mathbf{x}_i$ is the ith input position of a training set with size n, $f(\mathbf{x}_i)$ is the value of the output variable of interest produced by the satellite model at the input $\mathbf{x}_i$, the mean prediction $\mu^{-i}(\mathbf{x}_i)$ at input $\mathbf{x}_i$ is provided by the corresponding emulator constructed using all n training

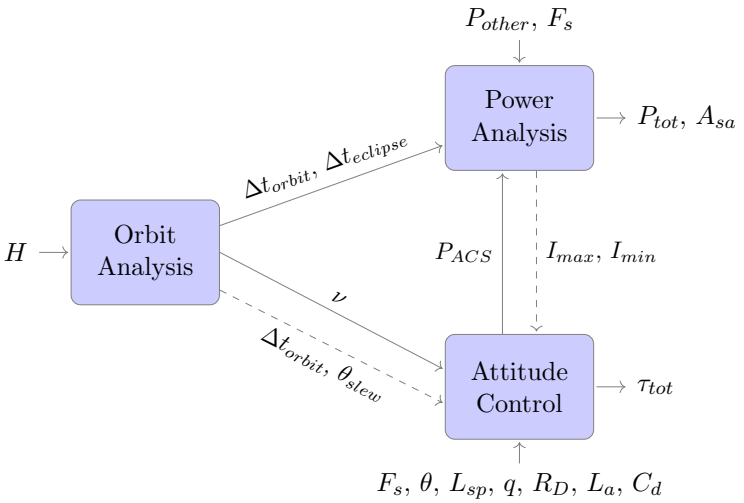


Figure 6. *Fire-detection satellite model from [24], where H is altitude; Δt_{orbit} is orbit period; $\Delta t_{eclipse}$ is eclipse period; ν is satellite velocity; θ_{slew} is maximum slewing angel; P_{other} represents other sources of power; P_{ACS} is power of attitude control system; I_{max}, I_{min} are maximum and minimum moment of inertia respectively; F_s, θ, L_{sp}, q, R_D, L_a, C_d represent average solar flux, deviation of moment axis from vertical, moment arm for the solar radiation torque, reflectance factor, residual dipole, moment arm for aerodynamic torque, and drag coefficient respectively; P_{tot} is total power; A_{sa} is area of solar array; and τ_{tot} is total torque. The dashed arrows indicate the connections that can be decoupled between sub-models, according to the decoupling algorithm from [2].*

Table 2
Domains of the nine global input variables to be considered for the emulation.

Global input variable (unit)	Symbol	Domain
Altitude (m)	H	$[1.50 \times 10^{17}, 2.10 \times 10^{17}]$
Other sources of power (W)	P_{other}	$[8.50 \times 10^{2}, 1.15 \times 10^{3}]$
Average solar flux (W/m^2)	F_s	$[1.34 \times 10^{3}, 1.46 \times 10^{3}]$
Deviation of moment axis from vertical ($^\circ$)	θ	[12.00, 18.00]
Moment arm for the solar radiation torque (m)	L_{sp}	[0.80, 3.20]
Reflectance factor	q	[0, 1]
Residual dipole ($A \cdot m^2$)	R_D	[2.00, 8.00]
Moment arm for aerodynamic torque (m)	L_a	[0.80, 3.20]
Drag coefficient	C_d	[0.10, 1, 90]

points except for $\mathbf{x}_i$.

The NRMSEP of the composite emulators and linked GaSPs of the three global output variables τ_{tot}, P_{tot}, and A_{sa} against seven different training sizes are presented on the top row of Figure 7. It can be seen that for the output variable τ_{tot}, the linked GaSP is only marginally better than the composite emulator. For the output variables P_{tot} and A_{sa}, the linked GaSPs present better predictive performance than the composite ones when the training set size is small. The superiority of the linked GaSP soon vanishes when the training set size

increases over 20. To investigate the possible cause for this quick depreciation, we construct GaSP emulators for outputs produced by the three submodels. The NRMSEP of these GaSP emulators across different training sizes are summarized on the bottom row of Figure 7. We observe that the GaSP emulator of the attitude control with respect to τ_{tot} requires around 35 training points to reach a low NRMSEP, while the GaSP emulator of the orbit analysis with respect to ν can reach such a level with only 10 training points. This indicates that the functional complexity between the global inputs and the output τ_{tot} is dominated by the submodel attitude control, and thus the linked GaSP of τ_{tot} shows no obvious superiority over the corresponding composite emulator. Although the attitude control still dominates the functional complexity between the global inputs and P_{tot} and A_{sa} (see Figure 7f), P_{tot} and A_{sa} are produced not only by the orbit analysis and attitude control, but also by the power analysis. This extra submodel increases the input dimension that the composite emulators need to explore, and thus cause the composite emulators slow to learn the functional dependence of P_{tot} and A_{sa} to the global inputs when training data size is small.

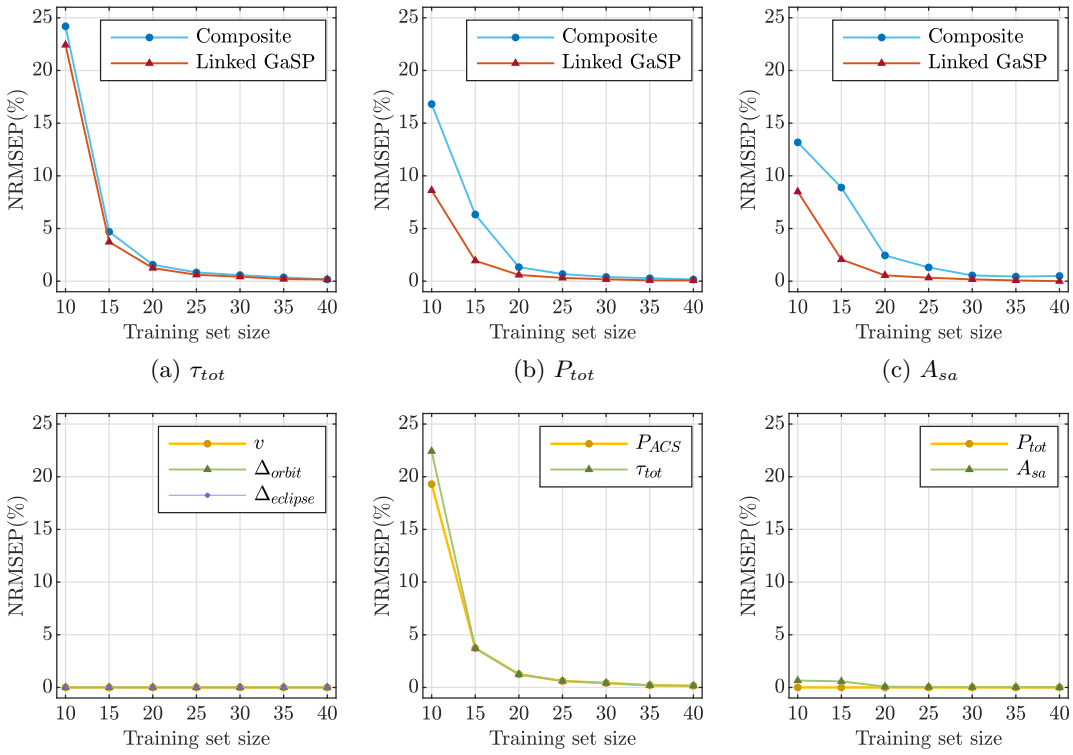


Figure 7. (Top) *NRMSEP of the composite emulators and linked GaSPs of the three global output variables τ_{tot}, P_{tot}, and A_{sa} against different training set sizes.* (Bottom) *NRMSEP of the GaSP emulators of outputs produced by the three subsystems: orbit analysis, attitude control, and power analysis.*

5. Experimental designs for linked GaSP. The linked GaSP is so far constructed using the Latin hypercube design (LHD) [27] in a sequential fashion. It means that a one-shot LHD is applied only to the global inputs (i.e., the inputs to the computer models in the first layer of the system) and designs for the inputs to the computer models in higher layers are automatically determined by the outputs from the lower-layer computer models. This design, called sequential LHD hereinafter, is a simple strategy and has the benefit that it only explores input spaces of individual computer models that have impact on the global outputs. However, the complexity of system structures and nonlinearity of individual computer models can produce poor designs for submodels in higher layers when the LHD of the global input is propagated through the system hierarchy. This issue can be seen from the sequential LHD (see Figure 8) that we used for the synthetic experiment in subsection 3.1. Figure 8 shows that although the LHD gives satisfactory input exploration for the global inputs x_1 and x_2, the design for the computer model f_3 is poor. This is because of the steep decrease of f_2 over $x_2 \in [0, 0.5]$, which concentrates most of the design points for f_3 on the border of its input w_2 while few of them locate over $w_2 \in [4.1, 5.0]$. Indeed, such an issue could be alleviated by increasing the size of the sequential LHD or implementing adaptive design strategies (e.g., [3]) over the global inputs. However, these solutions can result in excessive design points that contain similar information about the underlying computer model. In addition, such sequential designs require full runs of entire systems, and thus can be computationally expensive and inefficient when the designs for some submodels are already satisfactory and no further enhancements are needed.

The work in [17] suggests an independent design strategy where the designs of submodels are developed (by either one-shot LHD or adaptive designs) separately without considering their structural dependence. This design strategy is useful because the construction of the linked GaSP does not require realizations generated by running the whole system and thus different computer models can be ran in parallel rather than in sequence; one can even use existing realizations (with different sizes) from individual computer models to build the linked GaSP; the experimental design can be tailor-made for each computer model and thus one avoids issues related to the aforementioned sequential designs.

While it is desirable to construct accurate GaSP emulators of individual computer models via the independent design and then integrate them to have a well-behaved linked GaSP, ignoring the structure dependence can cause *unnecessary refinements* of GaSP emulators (and thus excessive experimental costs) over input spaces of computer models that are insignificant to the global output. Similarly, the ignorance of structural dependence may also cause GaSP emulators to be accurate only in part of input spaces that are significant to the global output. We illustrate such an issue in section SM1 of the supplementary materials. In subsection 5.1, we introduce an adaptive design strategy for the linked GaSP that utilizes the analytical variance decomposition of linked emulators. As we will show, this design not only takes system structures into account but also shares some advantages of the independent design.

5.1. A variance-based adaptive design for linked GaSP. The adaptive design introduced in this section extends the simulation-based Single Model Selection training strategy given in [25]. At each iteration, the adaptive design conducts the follow three steps:

1. Select one submodel and determine the input position to run the model.

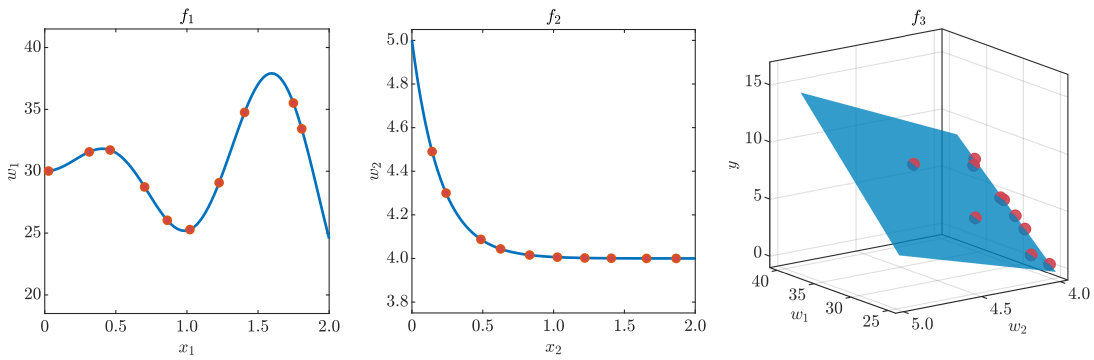


Figure 8. *The sequential LHD used to build the linked GaSP for the synthetic experiment in subsection 3.1. The solid lines and surface represent the true functional forms of each computer model; the filled circles are design points.*

2. Run the selected submodel and refine its GaSP emulator given the new run.
3. Construct the linked GaSP of the system.

It can be seen that at each iteration the adaptive design only requires a single run of one submodel. Therefore, one can save computational resources by avoiding runs of the whole system and only refining the GaSP emulator of one submodel to improve the overall accuracy of the linked GaSP. We select the target submodel at each iteration by searching for the submodel whose GaSP emulator contributes the most to the variance of the linked GaSP. We demonstrate the approach on a two-layered system whose submodels have their GaSP emulators connected as in Figure 1. Note (see section SM4 of the supplementary materials) that the variance of linked emulator in equation (3.2) of Theorem 3.3 can be written as

$$\sigma_L^2 = \mathrm{Var}\left(\mu_g(\mathbf{W},\mathbf{z})\right) + \mathbb{E}\left[\sigma_g^2(\mathbf{W},\mathbf{z})\right],$$

where

$$\mathrm{Var}\left(\mu_g(\mathbf{W},\mathbf{z})\right) = \mathbf{A}^\top\left(\mathbf{J}-\mathbf{I}\mathbf{I}^\top\right)\mathbf{A} + 2\widehat{\boldsymbol{\theta}}^\top\left(\mathbf{B}-\boldsymbol{\mu}\mathbf{I}^\top\right)\mathbf{A} + \mathrm{tr}\left\{\widehat{\boldsymbol{\theta}}\widehat{\boldsymbol{\theta}}^\top\boldsymbol{\Omega}\right\},$$
$$\mathbb{E}\left[\sigma_g^2(\mathbf{W},\mathbf{z})\right] = \sigma^2\left(1+\eta+\mathrm{tr}\left\{\mathbf{Q}\mathbf{J}\right\}+\mathbf{G}^\top\mathbf{C}\mathbf{G}+\mathrm{tr}\left\{\mathbf{C}\mathbf{P}-2\mathbf{C}\widetilde{\mathbf{H}}^\top\mathbf{R}^{-1}\mathbf{K}\right\}\right)$$

with $\mu_g(\mathbf{W},\mathbf{z})$ and $\sigma_g^2(\mathbf{W},\mathbf{z})$ being the mean and variance of $\widehat{g}$.

Define $V_1 = \mathrm{Var}\left(\mu_g(\mathbf{W},\mathbf{z})\right)$ and $V_2 = \mathbb{E}\left[\sigma_g^2(\mathbf{W},\mathbf{z})\right]$, then V_1 represents the overall contribution of GaSP emulators $\widehat{f}_1,\ldots,\widehat{f}_d$ to σ_L^2, and V_2 represents the contribution of $\widehat{g}$ to σ_L^2. Analogously, the variance contribution of GaSP emulators $\widehat{f}_{k\in\mathbb{S}}$ for $\mathbb{S}\subseteq\{1,\ldots,d\}$ can be defined by $V_1(\mathbb{S}) = \mathrm{Var}_{W_{k\in\mathbb{S}}}\left(\mathbb{E}_{W_{k\in\mathbb{S}^c}}\left[\mu_g(\mathbf{W},\mathbf{z})\right]\right)$, where $\mathbb{S}^c$ is the complement of $\mathbb{S}$. One can compute $V_1(\mathbb{S})$ analytically according to Proposition 5.1.

Proposition 5.1. *Under the same conditions of Theorem 3.3, $V_1(\mathbb{S})$ has the closed form expression given by*

$$V_1(\mathbb{S}) = \mathbf{A}^\top\left(\widetilde{\mathbf{J}}-\mathbf{I}\mathbf{I}^\top\right)\mathbf{A} + 2\widehat{\boldsymbol{\theta}}^\top\left(\widetilde{\mathbf{B}}-\boldsymbol{\mu}\mathbf{I}^\top\right)\mathbf{A} + \mathrm{tr}\left\{\widehat{\boldsymbol{\theta}}\widehat{\boldsymbol{\theta}}^\top\widetilde{\boldsymbol{\Omega}}\right\},$$

where

- $\widetilde{\mathbf{\Omega}}$ *is a* $d \times d$ *diagonal matrix with* k*th diagonal element given by* $\sigma_k^2(\mathbf{x}_k)\mathbb{1}_{\{k\in\mathbb{S}\}}$;
- $\widetilde{\mathbf{J}}$ *is a* $m \times m$ *matrix with the* ij*th element given by*

$$\widetilde{J}_{ij} = \prod_{k\in\mathbb{S}} \zeta_{ijk} \prod_{k\in\mathbb{S}^c} \xi_{ik}\xi_{jk} \prod_{k=1}^{p} c_k(z_k, z_{ik}^{\mathcal{T}})\, c_k(z_k, z_{jk}^{\mathcal{T}});$$

- $\widetilde{\mathbf{B}}$ *is a* $d \times m$ *matrix with the* lj*th element given by*

$$\widetilde{B}_{lj} = \begin{cases} \psi_{jl} \displaystyle\prod_{\substack{k=1\\k\neq l}}^{d} \xi_{jk} \prod_{k=1}^{p} c_k(z_k, z_{jk}^{\mathcal{T}}), & l \in \mathbb{S}, \\ \mu_l \displaystyle\prod_{k=1}^{d} \xi_{jk} \prod_{k=1}^{p} c_k(z_k, z_{jk}^{\mathcal{T}}), & l \in \mathbb{S}^c. \end{cases}$$

Proof. The proof is in section SM6 of the supplementary materials. ∎

Thanks to the closed form expressions of V_1, V_2, and $V_1(\mathbb{S})$, the adaptive design can quickly locate the submodel and determine the input position to run the model. To show the performance we implement the adaptive design on the synthetic example in subsection 3.1 via Algorithm 5.1, where the optimization problem in line 3 is done by grid search due to the low global input dimension. The linked GaSP built by the adaptive design is summarized in Figure 9. It can be observed from Figure 9 that the linked GaSP built via the adaptive design can achieve lower NRMSEP than that built via the sequential LHD, with a smaller number of computer model runs. This is because, in contrast to the poor design for f_3 created by the sequential LHD (see Figure 8), the adaptive design creates a satisfactory design by adding extra design points to the input space of f_3 that is not well explored by the sequential LHD but still significant to the global output. It can also be seen that the adaptive design leads to more runs of f_1, whose functional form is more complex than other models and thus needs to generate more realizations to be emulated adequately. Thus the adaptive design is able to improve the emulation performance of the linked GaSP with reduced experimental costs by allocating runs to computer models according to their heterogeneous functional complexity. We also report in Figure 9 the NRMSEP of the linked GaSP trained with the independent design, by which GaSP emulators of individual computer models are built separately with their own training points independently generated from the LHD. Although the linked GaSP with the independent design achieves a low NRMSEP, its accuracy is overestimated because we assume that the input domain of f_3 that is significant to the global output is perfectly known or can be determined in a cost efficient way, e.g., we were able to determine the important input domain of f_3 by evaluating f_1 and f_2 exhaustively over the entire domain of the global input thanks to the cheap cost of the synthetic models. However, in practice it is rarely possible to gain perfect knowledge about the important input domain of a computer model or feasible to evaluate models thoroughly without constraints.

Although the adaptive design is a desirable design strategy, it has its own limitations. First, the adaptive design updates the GaSP emulator of one submodel iteratively. Therefore,

Algorithm 5.1. Adaptive design for the synthetic system illustrated in subsection 3.1.

Choose K number of enrichment (i.e., iterations) to the initial design.
for $k = 1, \ldots, K$ **do**
  Find $\widehat{\mathbf{x}}$ and $\widehat{l}$ such that $(\widehat{\mathbf{x}}, \widehat{l}) = \text{argmax}_{\mathbf{x}, l \in \{1,2\}} V_l(\mathbf{x})$, where $\mathbf{x} = (x_1, x_2)$, and $V_1(\mathbf{x})$ and $V_2(\mathbf{x})$, respectively, are contributions of $\widehat{e}_1$ (i.e., GaSP emulators $\widehat{f}_1$ and $\widehat{f}_2$ in the first layer) and $\widehat{f}_3$ to the variance of the linked GaSP;
  if $\widehat{l} = 1$ **then**
    Compute $V_{1k}(\widehat{\mathbf{x}})$ for $k \in \{1, 2\}$ according to Proposition 5.1, where $V_{1k}(\widehat{\mathbf{x}})$ is the contribution of $\widehat{f}_k$ to the variance of linked GaSP;
    if $V_{11}(\widehat{\mathbf{x}}) > V_{12}(\widehat{\mathbf{x}})$ **then**
      Enrich the training points for $\widehat{f}_1$ by evaluating f_1 at the input position $\widehat{x}_1$;
    else
      Enrich the training points for $\widehat{f}_2$ by evaluating f_2 at the input position $\widehat{x}_2$;
    end if
  else
    Enrich the training points for $\widehat{f}_3$ by evaluating f_3 at the input position $(\mu_1(\widehat{x}_1), \mu_2(\widehat{x}_2))$, obtained by evaluating the predictive mean μ_1 and μ_2 of $\widehat{f}_1$ and $\widehat{f}_2$ at the input position $\widehat{x}_1$ and $\widehat{x}_2$, respectively;
  end if
  Update the GaSP emulator $\widehat{f}_1$, $\widehat{f}_2$, or $\widehat{f}_3$ and construct the linked GaSP.
end for

unlike the independent design, it does not allow submodels of a system to run simultaneously during the experimental design. Beside, the adaptive design is still a sequential method because the input location at which the selected submodel needs to run is determined by propagating the determined global input location through the GaSP emulators of those submodels in lower layers. As a result, inaccurate GaSP emulators in lower layers may produce suboptimal input positions to improve the GaSP emulators in higher layers. One thus need to implement the adaptive design with more iterations, and in turn spend more computational resources, to improve the linked GaSP sufficiently. Furthermore, the maximization problem involved in the adaptive design to search for the submodel whose GaSP emulator needs to be updated is a challenging task especially when the global input dimension is high. Therefore, developing a fast and efficient searching algorithm is essential. Fortunately, the closed form expressions for the variance decomposition given in Proposition 5.1 render the exact evaluation of their derivatives respect to the input positions, thus many existing optimization algorithms (e.g., gradient ascent) could be applied. We leave this aspect as a future development without exploring further in this study.

6. Discussion. The development of Theorem 3.3 in section 3 depends on Assumption 3.2, which asks for independence of input variables to the GaSP emulator of g in the second layer. This independence assumption helps reduce analytical efforts in deriving the closed form mean and variance of the linked emulator. In addition, the consideration of dependence between input variables requires specification of their dependence structures, which can be a difficult

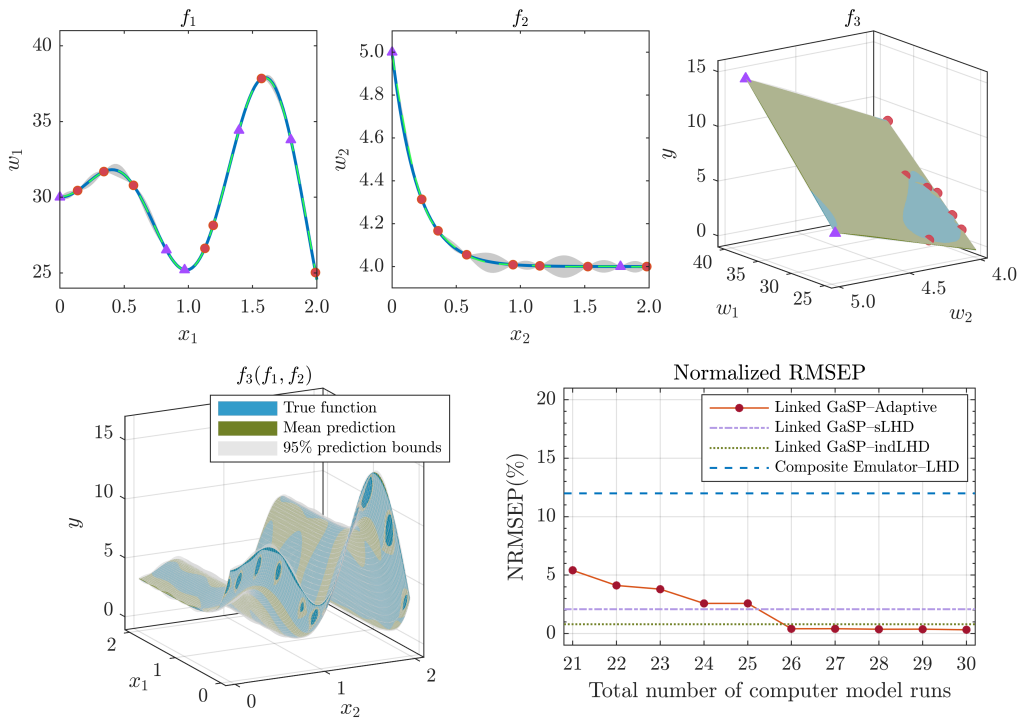


Figure 9. *The adaptive design for the synthetic experiment in subsection 3.1.* (Top-left) *GaSP emulator of* f_1. (Top-middle) *GaSP emulator of* f_2. (Top-right) *GaSP emulator of* f_3. (Bottom-left) *linked GaSP of the system.* (Bottom-right) *Comparison of NRMSEP between the linked GaSP with the adaptive design, the linked GaSP with the sequential LHD (sLHD), the linked GaSP with the independent LHD (indLHD), and the composite emulator with the LHD. The linked GaSP with the sLHD and the composite emulator are trained with* 30 *computer runs (i.e.,* 10 *full runs of the entire system). The linked GaSP with the indLHD is trained with* 10 *runs for each submodel. The linked GaSP with the adaptive design is trained with* 21 *initial computer model runs determined by the sLHD (i.e., seven runs of the whole system, corresponding to the filled circles in the top panels) and nine additional submodel runs (corresponding to the filled triangles in the top panels) over nine iterations.*

task as careful dependence modeling, model training, and predictions are needed. Nevertheless, ignoring the dependence structure between input variables feeding to the second layer can cause biased mean and variance of the linked emulator if the dependence is nonnegligible. The work in [17] explores the impact of such dependence ignorance and conclude that in the case of Gaussian dependence under the squared exponential kernel, one could diagnose the significance of dependence by calculating the following ratios $r_k = \widehat{\gamma}_k^2/\sigma_k^2$ for all $k = 1, \ldots, d$, where $\widehat{\gamma}_k$ is the estimated range parameter of the kth input to the GaSP emulator $\widehat{g}$. If r_k is large (e.g., in the order of hundreds or thousands) for all k, the difference between the linked GaSPs with and without the dependence structure is then negligible. Note that given $\widehat{\gamma}_k^2$, r_k increases as predictive variance σ_k^2 decreases. Thus, one could safely neglect the impact of dependence by improving GaSP emulators in the feeding layer. We review these results in section SM2 of the supplementary materials. Since r_k is calculated without the consider-

ation of dependence and before invoking Theorem 3.3, it can be used as a measurement to determine whether one should consider the dependence before explicitly incorporating it to the emulation.

However, r_k may not be a valid measurement when kernels other than the squared exponential are used. It is also difficult in practice to have GaSP emulators producing sufficiently small predictive variances at the evaluated input positions to rule out the impact of dependence. Therefore, one may have to consider specifying the dependence structure between outputs of GaSP emulators from the feeding layer. One option for the dependence specification is to build multivariate GaSP emulators [22, 8, 33]. However, existing literature on multivariate GaSP only consider the dependence among outputs from a single computer model, which means that in each layer of a system one has to treat all computer models, whose outputs are correlated, as a single model for the multivariate GaSP emulation, This is apparently an unpleasant feature because it reduces the benefit of system order reduction (i.e., GaSP emulators are constructed for individual computer models) offered by the linked GaSP emulation. A possible solution to this issue is to first build GaSP emulators ignoring the dependence and then model dependence structure separately, e.g., utilizing copulas [6]. Nevertheless, one still need to conduct extra analytical efforts to derive more sophisticated closed form expressions for the mean and variance of linked emulator under the multivariate setting for different kernel choices.

Linked emulator gives the true distributional representation of coupled GaSP emulators of computer models in a system. Linked GaSP then serves as a Gaussian approximation to the analytically intractable linked emulator. The use of linked GaSP in replacement of linked emulator can be justified from two aspects. First, with Gaussian distribution, one can construct closed form linked GaSP successively via the iterative procedure in section 4. Second, linked GaSP with its mean and variance matching to the linked emulator minimizes the Kullback–Leibler (KL) divergence (i.e., information loss) between the linked emulator and a Gaussian density [19].

The approximation accuracy of the linked GaSP to the linked emulator for a two-layered system is explored in [17], which indicate that the linked GaSP converges to the linked emulator when the predictive variances of GaSP emulators in the first layer reduce to zero. This statement is intuitive because GaSP emulators tend to be deterministic as their predictive variances drop. Consequently, the linked emulator decays to a Gaussian distribution that is equivalent to the corresponding linked GaSP. However, it is often not possible to ensure this condition for multilayered systems, especially when systems are complex and the computational budget is limited. We explore provisionally the approximating performance of the linked GaSP in a three-layered synthetic system with a fairly small number of training points in section SM3 of the supplementary materials. We found, and we also conjecture for systems with a moderate number of layers, that the linked GaSP approximates well the mean and variance of the linked emulator, while is unable to reconstruct sufficiently the full probabilistic distribution of the linked emulator. Therefore, the linked GaSP can be a good analytical replacement of a linked emulator for analysis, such as the history matching, where mean and variance are the key quantities of interest. However, if the full uncertainty description of an emulator is of concern (e.g., if tails are of specific interest), the linked GaSP may not be a fully adequate surrogate model.

Like all data-driven emulators, the linked GaSP is a simplified approximation to the underlying computer system, which can be both high dimensional and extremely nonlinear. Thus, careful plans and implementations, such as computational budget allocation, design consideration, and model validation, are essential for efficient emulation on systems of computer models. In addition, the accuracy of linked GaSPs is not only constrained by the assumptions listed in section 3, but also limited by those (e.g., stationarity) made for GaSP emulators. Therefore, further methodological and empirical advancements on both GaSP emulator and linked GaSP are required for robust uncertainty quantification of sophisticated real-world systems of computer models.

7. Conclusion.

In this study, we generalize the linked GaSP to a class of Matérn kernels. The ability to use Matérn kernels is essential for wider applications of the linked GaSP on uncertainty quantification of systems of computer models. The linked GaSP emulation can also be applied to any feed-forward systems with an iterative procedure. In combination with decoupling techniques, the linked GaSP can even be utilized for systems with internal loops.

The linked GaSP emulation can be further enhanced, in terms of the approximating accuracy and computational cost, via careful implementation of design strategies. We discuss pros and cons of several alternative designs and introduce an adaptive design that improves the accuracy of the linked GaSP with reduced computational by allocating runs to different computer models in a system based on their heterogeneous functional complexity. The benefits of the adaptive design are illustrated via a synthetic example. Further refinements of the design and how it performs in real systems are directions worth exploring.

The linked GaSP outperforms the composite emulator by a "divide-and-conquer" strategy [17], which converts the emulation of a bulky system into emulations of a number of simpler elements. However, when a single computer model dominates the functional complexity of the whole system the linked GaSP may not show a significant improvement over the composite emulator. Particularly, if the dimension of input to individual computer models is remarkably higher than that of global input, one might resort to dimension reduction techniques to construct GaSP emulators of individual computer models. Whether the benefits offered by the linked GaSP can overweight the approximation error induced by the dimension reduction methods needs to be studied in the future. Since the uncertainty quantification is now an integrated module in many research of multiphysics systems, one may consider split processes during the system development to facilitate surrogate modeling.

Overall, we demonstrate both the effectiveness and efficiency of our new strategies to build linked GaSPs for systems of computer models. Another ambitious, but needed, task would be to investigate how our results can be exploited to emulate more complex feed-back coupled systems, such as climate models, than the one considered in this study.

Appendix A. Exponential case.

The three expectations ξ_{ik}, ζ_{ijk}, and ψ_{jl} defined in

Theorem 3.3 are given by

$$\xi_{ik} = \exp\left\{\frac{\sigma_k^2 + 2\gamma_k\left(w_{ik}^{\mathcal{T}} - \mu_k\right)}{2\gamma_k^2}\right\}\Phi\left(\frac{\mu_A - w_{ik}^{\mathcal{T}}}{\sigma_k}\right) + \exp\left\{\frac{\sigma_k^2 - 2\gamma_k\left(w_{ik}^{\mathcal{T}} - \mu_k\right)}{2\gamma_k^2}\right\}\Phi\left(\frac{w_{ik}^{\mathcal{T}} - \mu_B}{\sigma_k}\right),$$

$$\zeta_{ijk} = \begin{cases} h_\zeta\left(w_{ik}^{\mathcal{T}}, w_{jk}^{\mathcal{T}}\right), & w_{jk}^{\mathcal{T}} \geq w_{ik}^{\mathcal{T}}, \\ h_\zeta\left(w_{jk}^{\mathcal{T}}, w_{ik}^{\mathcal{T}}\right), & w_{jk}^{\mathcal{T}} < w_{ik}^{\mathcal{T}}, \end{cases}$$

$$\psi_{jk} = \exp\left\{\frac{\sigma_k^2 + 2\gamma_k\left(w_{jk}^{\mathcal{T}} - \mu_k\right)}{2\gamma_k^2}\right\} \times \left[\mu_A\Phi\left(\frac{\mu_A - w_{jk}^{\mathcal{T}}}{\sigma_k}\right) + \frac{\sigma_k}{\sqrt{2\pi}}\exp\left\{-\frac{\left(w_{jk}^{\mathcal{T}} - \mu_A\right)^2}{2\sigma_k^2}\right\}\right] - \exp\left\{\frac{\sigma_k^2 - 2\gamma_k\left(w_{jk}^{\mathcal{T}} - \mu_k\right)}{2\gamma_k^2}\right\} \times \left[\mu_B\Phi\left(\frac{w_{jk}^{\mathcal{T}} - \mu_B}{\sigma_k}\right) - \frac{\sigma_k}{\sqrt{2\pi}}\exp\left\{-\frac{\left(w_{jk}^{\mathcal{T}} - \mu_B\right)^2}{2\sigma_k^2}\right\}\right],$$

where $\Phi(\cdot)$ denotes the cumulative density function of the standard normal distribution, and

$$h_\zeta(x_1, x_2) = \exp\left\{\frac{2\sigma_k^2 + \gamma_k(x_1 + x_2 - 2\mu_k)}{\gamma_k^2}\right\}\Phi\left(\frac{\mu_C - x_2}{\sigma_k}\right) + \exp\left\{-\frac{x_2 - x_1}{\gamma_k}\right\}\left[\Phi\left(\frac{x_2 - \mu_k}{\sigma_k}\right) - \Phi\left(\frac{x_1 - \mu_k}{\sigma_k}\right)\right] + \exp\left\{\frac{2\sigma_k^2 - \gamma_k(x_1 + x_2 - 2\mu_k)}{\gamma_k^2}\right\}\Phi\left(\frac{x_1 - \mu_D}{\sigma_k}\right),$$

and $\mu_A = \mu_k - \sigma_k^2/\gamma_k$, $\mu_B = \mu_k + \sigma_k^2/\gamma_k$, $\mu_C = \mu_k - 2\sigma_k^2/\gamma_k$, and $\mu_D = \mu_k + 2\sigma_k^2/\gamma_k$.

For notational convenience, in the above result we replace the index variable l in the subscript of ψ_{jl} by k, and $\mu_k(\mathbf{x}_k)$ and $\sigma_k(\mathbf{x}_k)$ by μ_k and σ_k. This change of notation is also applied in the rest of appendices and proofs of the supplementary materials.

Appendix B. Squared exponential case. The three expectations ξ_{ik}, ζ_{ijk}, and ψ_{jl} defined

in Theorem 3.3 are given by

$$\xi_{ik} = \frac{1}{\sqrt{1+2\sigma_k^2/\gamma_k^2}} \exp\left\{-\frac{\left(\mu_k - w_{ik}^{\mathcal{T}}\right)^2}{2\sigma_k^2+\gamma_k^2}\right\},$$

$$\zeta_{ijk} = \frac{1}{\sqrt{1+4\sigma_k^2/\gamma_k^2}} \exp\left\{-\frac{\left(\frac{w_{ik}^{\mathcal{T}}+w_{jk}^{\mathcal{T}}}{2} - \mu_k\right)^2}{\gamma_k^2/2+2\sigma_k^2} - \frac{\left(w_{ik}^{\mathcal{T}} - w_{jk}^{\mathcal{T}}\right)^2}{2\gamma_k^2}\right\},$$

$$\psi_{jk} = \frac{1}{\sqrt{1+2\sigma_k^2/\gamma_k^2}} \exp\left\{-\frac{\left(\mu_k - w_{jk}^{\mathcal{T}}\right)^2}{2\sigma_k^2+\gamma_k^2}\right\} \frac{2\sigma_k^2 w_{jk}^{\mathcal{T}} + \gamma_k^2\mu_k}{2\sigma_k^2+\gamma_k^2}.$$

Appendix C. Matérn-1.5 case. The three expectations ξ_{ik}, ζ_{ijk}, and ψ_{jl} defined in Theorem 3.3 are given by

$$\begin{aligned}
\xi_{ik} =& \exp\left\{\frac{3\sigma_k^2 + 2\sqrt{3}\gamma_k\left(w_{ik}^{\mathcal{T}} - \mu_k\right)}{2\gamma_k^2}\right\} \\
&\times \left[\mathbf{E}_1^\top \mathbf{\Lambda}_{11} \Phi\left(\frac{\mu_A - w_{ik}^{\mathcal{T}}}{\sigma_k}\right) + \mathbf{E}_1^\top \mathbf{\Lambda}_{12} \frac{\sigma_k}{\sqrt{2\pi}} \exp\left\{-\frac{(w_{ik}^{\mathcal{T}} - \mu_A)^2}{2\sigma_k^2}\right\}\right] \\
&+ \exp\left\{\frac{3\sigma_k^2 - 2\sqrt{3}\gamma_k\left(w_{ik}^{\mathcal{T}} - \mu_k\right)}{2\gamma_k^2}\right\} \\
&\times \left[\mathbf{E}_2^\top \mathbf{\Lambda}_{21} \Phi\left(\frac{w_{ik}^{\mathcal{T}} - \mu_B}{\sigma_k}\right) + \mathbf{E}_2^\top \mathbf{\Lambda}_{22} \frac{\sigma_k}{\sqrt{2\pi}} \exp\left\{-\frac{(w_{ik}^{\mathcal{T}} - \mu_B)^2}{2\sigma_k^2}\right\}\right], \\
\zeta_{ijk} =& \begin{cases} h_\zeta\left(w_{ik}^{\mathcal{T}}, w_{jk}^{\mathcal{T}}\right), & w_{jk}^{\mathcal{T}} \geq w_{ik}^{\mathcal{T}}, \\ h_\zeta\left(w_{jk}^{\mathcal{T}}, w_{ik}^{\mathcal{T}}\right), & w_{jk}^{\mathcal{T}} < w_{ik}^{\mathcal{T}}, \end{cases} \\
\psi_{jk} =& \exp\left\{\frac{3\sigma_k^2 + 2\sqrt{3}\gamma_k\left(w_{jk}^{\mathcal{T}} - \mu_k\right)}{2\gamma_k^2}\right\} \\
&\times \left[\mathbf{E}_1^\top \mathbf{\Lambda}_{61} \Phi\left(\frac{\mu_A - w_{jk}^{\mathcal{T}}}{\sigma_k}\right) + \mathbf{E}_1^\top \mathbf{\Lambda}_{62} \frac{\sigma_k}{\sqrt{2\pi}} \exp\left\{-\frac{(w_{jk}^{\mathcal{T}} - \mu_A)^2}{2\sigma_k^2}\right\}\right] \\
&- \exp\left\{\frac{3\sigma_k^2 - 2\sqrt{3}\gamma_k\left(w_{jk}^{\mathcal{T}} - \mu_k\right)}{2\gamma_k^2}\right\} \\
&\times \left[\mathbf{E}_2^\top \mathbf{\Lambda}_{71} \Phi\left(\frac{w_{jk}^{\mathcal{T}} - \mu_B}{\sigma_k}\right) + \mathbf{E}_2^\top \mathbf{\Lambda}_{72} \frac{\sigma_k}{\sqrt{2\pi}} \exp\left\{-\frac{(w_{jk}^{\mathcal{T}} - \mu_B)^2}{2\sigma_k^2}\right\}\right],
\end{aligned}$$

where

$$
\begin{aligned}
&h_\zeta(x_1, x_2)\\
&= \exp\left\{\frac{6\sigma_k^2 + \sqrt{3}\gamma_k(x_1 + x_2 - 2\mu_k)}{\gamma_k^2}\right\}\\
&\quad\times\left[\mathbf{E}_3^\top\mathbf{\Lambda}_{31}\Phi\left(\frac{\mu_C - x_2}{\sigma_k}\right) + \mathbf{E}_3^\top\mathbf{\Lambda}_{32}\frac{\sigma_k}{\sqrt{2\pi}}\exp\left\{-\frac{(x_2-\mu_C)^2}{2\sigma_k^2}\right\}\right]\\
&\quad+\exp\left\{-\frac{\sqrt{3}(x_2 - x_1)}{\gamma_k}\right\}\left[\mathbf{E}_4^\top\mathbf{\Lambda}_{41}\left(\Phi\left(\frac{x_2-\mu_k}{\sigma_k}\right) - \Phi\left(\frac{x_1-\mu_k}{\sigma_k}\right)\right)\right.\\
&\quad\left.+\mathbf{E}_4^\top\mathbf{\Lambda}_{42}\frac{\sigma_k}{\sqrt{2\pi}}\exp\left\{-\frac{(x_1-\mu_k)^2}{2\sigma_k^2}\right\} - \mathbf{E}_4^\top\mathbf{\Lambda}_{43}\frac{\sigma_k}{\sqrt{2\pi}}\exp\left\{-\frac{(x_2-\mu_k)^2}{2\sigma_k^2}\right\}\right]\\
&\quad+\exp\left\{\frac{6\sigma_k^2 - \sqrt{3}\gamma_k(x_1 + x_2 - 2\mu_k)}{\gamma_k^2}\right\}\\
&\quad\times\left[\mathbf{E}_5^\top\mathbf{\Lambda}_{51}\Phi\left(\frac{x_1-\mu_D}{\sigma_k}\right) + \mathbf{E}_5^\top\mathbf{\Lambda}_{52}\frac{\sigma_k}{\sqrt{2\pi}}\exp\left\{-\frac{(x_1-\mu_D)^2}{2\sigma_k^2}\right\}\right]
\end{aligned}
$$

and

- $\mathbf{\Lambda}_{11} = [1, \mu_A]^\top$, $\mathbf{\Lambda}_{12} = [0, 1]^\top$, $\mathbf{\Lambda}_{21} = [1, -\mu_B]^\top$, and $\mathbf{\Lambda}_{22} = [0, 1]^\top$;
- $\mathbf{\Lambda}_{31} = [1, \mu_C, \mu_C^2 + \sigma_k^2]^\top$ and $\mathbf{\Lambda}_{32} = [0, 1, \mu_C + x_2]^\top$;
- $\mathbf{\Lambda}_{41} = [1, \mu_k, \mu_k^2 + \sigma_k^2]^\top$, $\mathbf{\Lambda}_{42} = [0, 1, \mu_k + x_1]^\top$, and $\mathbf{\Lambda}_{43} = [0, 1, \mu_k + x_2]^\top$;
- $\mathbf{\Lambda}_{51} = [1, -\mu_D, \mu_D^2 + \sigma_k^2]^\top$ and $\mathbf{\Lambda}_{52} = [0, 1, -\mu_D - x_1]^\top$;
- $\mathbf{\Lambda}_{61} = [\mu_A, \mu_A^2 + \sigma_k^2]^\top$ and $\mathbf{\Lambda}_{62} = [1, \mu_A + w_{jk}^{\mathcal{T}}]^\top$;
- $\mathbf{\Lambda}_{71} = [-\mu_B, \mu_B^2 + \sigma_k^2]^\top$ and $\mathbf{\Lambda}_{72} = [1, -\mu_B - w_{jk}^{\mathcal{T}}]^\top$;
- $\mathbf{E}_1 = \left[1 - \frac{\sqrt{3}w_{ik}^{\mathcal{T}}}{\gamma_k}, \frac{\sqrt{3}}{\gamma_k}\right]^\top$ and $\mathbf{E}_2 = \left[1 + \frac{\sqrt{3}w_{ik}^{\mathcal{T}}}{\gamma_k}, \frac{\sqrt{3}}{\gamma_k}\right]^\top$;
- $\mathbf{E}_3 = \left[1 + \frac{3x_1x_2 - \sqrt{3}\gamma_k(x_1+x_2)}{\gamma_k^2}, \frac{2\sqrt{3}\gamma_k - 3(x_1+x_2)}{\gamma_k^2}, \frac{3}{\gamma_k^2}\right]^\top$;
- $\mathbf{E}_4 = \left[1 + \frac{\sqrt{3}\gamma_k(x_2-x_1) - 3x_1x_2}{\gamma_k^2}, \frac{3(x_1+x_2)}{\gamma_k^2}, -\frac{3}{\gamma_k^2}\right]^\top$;
- $\mathbf{E}_5 = \left[1 + \frac{3x_1x_2 + \sqrt{3}\gamma_k(x_1+x_2)}{\gamma_k^2}, \frac{2\sqrt{3}\gamma_k + 3(x_1+x_2)}{\gamma_k^2}, \frac{3}{\gamma_k^2}\right]^\top$;
- $\mu_A = \mu_k - \frac{\sqrt{3}\sigma_k^2}{\gamma_k}$, $\mu_B = \mu_k + \frac{\sqrt{3}\sigma_k^2}{\gamma_k}$, $\mu_C = \mu_k - \frac{2\sqrt{3}\sigma_k^2}{\gamma_k}$, $\mu_D = \mu_k + \frac{2\sqrt{3}\sigma_k^2}{\gamma_k}$.

Appendix D. Matérn-2.5 case. The three expectations ξ_{ik}, ζ_{ijk}, and ψ_{jl} defined in

Theorem 3.3 are given by

$$\begin{aligned}\xi_{ik} = &\exp\left\{\frac{5\sigma_k^2 + 2\sqrt{5}\gamma_k\left(w_{ik}^{\mathcal{T}} - \mu_k\right)}{2\gamma_k^2}\right\} \\ &\times\left[\mathbf{E}_1^\top \mathbf{\Lambda}_{11}\Phi\left(\frac{\mu_A - w_{ik}^{\mathcal{T}}}{\sigma_k}\right) + \mathbf{E}_1^\top \mathbf{\Lambda}_{12}\frac{\sigma_k}{\sqrt{2\pi}}\exp\left\{-\frac{(w_{ik}^{\mathcal{T}} - \mu_A)^2}{2\sigma_k^2}\right\}\right] \\ &+ \exp\left\{\frac{5\sigma_k^2 - 2\sqrt{5}\gamma_k\left(w_{ik}^{\mathcal{T}} - \mu_k\right)}{2\gamma_k^2}\right\} \\ &\times\left[\mathbf{E}_2^\top \mathbf{\Lambda}_{21}\Phi\left(\frac{w_{ik}^{\mathcal{T}} - \mu_B}{\sigma_k}\right) + \mathbf{E}_2^\top \mathbf{\Lambda}_{22}\frac{\sigma_k}{\sqrt{2\pi}}\exp\left\{-\frac{(w_{ik}^{\mathcal{T}} - \mu_B)^2}{2\sigma_k^2}\right\}\right],\end{aligned}$$

$$\zeta_{ijk} = \begin{cases} h_\zeta\left(w_{ik}^{\mathcal{T}}, w_{jk}^{\mathcal{T}}\right), & w_{jk}^{\mathcal{T}} \geq w_{ik}^{\mathcal{T}}, \\ h_\zeta\left(w_{jk}^{\mathcal{T}}, w_{ik}^{\mathcal{T}}\right), & w_{jk}^{\mathcal{T}} < w_{ik}^{\mathcal{T}}, \end{cases}$$

and

$$\begin{aligned}\psi_{jk} = &\exp\left\{\frac{5\sigma_k^2 + 2\sqrt{5}\gamma_k\left(w_{jk}^{\mathcal{T}} - \mu_k\right)}{2\gamma_k^2}\right\} \\ &\times\left[\mathbf{E}_1^\top \mathbf{\Lambda}_{61}\Phi\left(\frac{\mu_A - w_{jk}^{\mathcal{T}}}{\sigma_k}\right) + \mathbf{E}_1^\top \mathbf{\Lambda}_{62}\frac{\sigma_k}{\sqrt{2\pi}}\exp\left\{-\frac{(w_{jk}^{\mathcal{T}} - \mu_A)^2}{2\sigma_k^2}\right\}\right] \\ &- \exp\left\{\frac{5\sigma_k^2 - 2\sqrt{5}\gamma_k\left(w_{jk}^{\mathcal{T}} - \mu_k\right)}{2\gamma_k^2}\right\} \\ &\times\left[\mathbf{E}_2^\top \mathbf{\Lambda}_{71}\Phi\left(\frac{w_{jk}^{\mathcal{T}} - \mu_B}{\sigma_k}\right) + \mathbf{E}_2^\top \mathbf{\Lambda}_{72}\frac{\sigma_k}{\sqrt{2\pi}}\exp\left\{-\frac{(w_{jk}^{\mathcal{T}} - \mu_B)^2}{2\sigma_k^2}\right\}\right],\end{aligned}$$

where

$$
\begin{aligned}
&h_\zeta(x_1, x_2)\\
&= \exp\left\{\frac{10\sigma_k^2 + \sqrt{5}\gamma_k(x_1 + x_2 - 2\mu_k)}{\gamma_k^2}\right\}\\
&\quad\times\left[\mathbf{E}_3^\top \mathbf{\Lambda}_{31}\Phi\left(\frac{\mu_C - x_2}{\sigma_k}\right) + \mathbf{E}_3^\top \mathbf{\Lambda}_{32}\frac{\sigma_k}{\sqrt{2\pi}}\exp\left\{-\frac{(x_2-\mu_C)^2}{2\sigma_k^2}\right\}\right]\\
&\quad+\exp\left\{-\frac{\sqrt{5}(x_2 - x_1)}{\gamma_k}\right\}\left[\mathbf{E}_4^\top \mathbf{\Lambda}_{41}\left(\Phi\left(\frac{x_2-\mu_k}{\sigma_k}\right) - \Phi\left(\frac{x_1-\mu_k}{\sigma_k}\right)\right)\right.\\
&\quad\left.+\mathbf{E}_4^\top \mathbf{\Lambda}_{42}\frac{\sigma_k}{\sqrt{2\pi}}\exp\left\{-\frac{(x_1-\mu_k)^2}{2\sigma_k^2}\right\} - \mathbf{E}_4^\top \mathbf{\Lambda}_{43}\frac{\sigma_k}{\sqrt{2\pi}}\exp\left\{-\frac{(x_2-\mu_k)^2}{2\sigma_k^2}\right\}\right]\\
&\quad+\exp\left\{\frac{10\sigma_k^2 - \sqrt{5}\gamma_k(x_1 + x_2 - 2\mu_k)}{\gamma_k^2}\right\}\\
&\quad\times\left[\mathbf{E}_5^\top \mathbf{\Lambda}_{51}\Phi\left(\frac{x_1-\mu_D}{\sigma_k}\right) + \mathbf{E}_5^\top \mathbf{\Lambda}_{52}\frac{\sigma_k}{\sqrt{2\pi}}\exp\left\{-\frac{(x_1-\mu_D)^2}{2\sigma_k^2}\right\}\right]
\end{aligned}
$$

and

- $\mathbf{\Lambda}_{11} = [1,\, \mu_A,\, \mu_A^2 + \sigma_k^2]^\top$ and $\mathbf{\Lambda}_{12} = [0,\, 1,\, \mu_A + w_{ik}^{\mathcal{T}}]^\top$;
- $\mathbf{\Lambda}_{21} = [1,\, -\mu_B,\, \mu_B^2 + \sigma_k^2]^\top$ and $\mathbf{\Lambda}_{22} = [0,\, 1,\, -\mu_B - w_{ik}^{\mathcal{T}}]^\top$;
- $\mathbf{\Lambda}_{31} = [1,\, \mu_C,\, \mu_C^2 + \sigma_k^2,\, \mu_C^3 + 3\sigma_k^2\mu_C,\, \mu_C^4 + 6\sigma_k^2\mu_C^2 + 3\sigma_k^4]^\top$;
- $\mathbf{\Lambda}_{32} = [0,\, 1,\, \mu_C + x_2,\, \mu_C^2 + 2\sigma_k^2 + x_2^2 + \mu_C x_2,\, \mu_C^3 + x_2^3 + x_2\mu_C^2 + \mu_C x_2^2 + 3\sigma_k^2 x_2 + 5\sigma_k^2\mu_C]^\top$;
- $\mathbf{\Lambda}_{41} = [1,\, \mu_k,\, \mu_k^2 + \sigma_k^2,\, \mu_k^3 + 3\sigma_k^2\mu_k,\, \mu_k^4 + 6\sigma_k^2\mu_k^2 + 3\sigma_k^4]^\top$;
- $\mathbf{\Lambda}_{42} = [0,\, 1,\, \mu_k + x_1,\, \mu_k^2 + 2\sigma_k^2 + x_1^2 + \mu_k x_1,\, \mu_k^3 + x_1^3 + x_1\mu_k^2 + \mu_k x_1^2 + 3\sigma_k^2 x_1 + 5\sigma_k^2\mu_k]^\top$;
- $\mathbf{\Lambda}_{43} = [0,\, 1,\, \mu_k + x_2,\, \mu_k^2 + 2\sigma_k^2 + x_2^2 + \mu_k x_2,\, \mu_k^3 + x_2^3 + x_2\mu_k^2 + \mu_k x_2^2 + 3\sigma_k^2 x_2 + 5\sigma_k^2\mu_k]^\top$;
- $\mathbf{\Lambda}_{51} = [1,\, -\mu_D,\, \mu_D^2 + \sigma_k^2,\, -\mu_D^3 - 3\sigma_k^2\mu_D,\, \mu_D^4 + 6\sigma_k^2\mu_D^2 + 3\sigma_k^4]^\top$;
- $\mathbf{\Lambda}_{52} = [0,\, 1,\, -\mu_D - x_1,\, \mu_D^2 + 2\sigma_k^2 + x_1^2 + \mu_D x_1,\, -\mu_D^3 - x_1^3 - x_1\mu_D^2 - \mu_D x_1^2 - 3\sigma_k^2 x_1 - 5\sigma_k^2\mu_D]^\top$;
- $\mathbf{\Lambda}_{61} = [\mu_A,\, \mu_A^2 + \sigma_k^2,\, \mu_A^3 + 3\sigma_k^2\mu_A]^\top$;
- $\mathbf{\Lambda}_{62} = [1,\, \mu_A + w_{jk}^{\mathcal{T}},\, \mu_A^2 + 2\sigma_k^2 + \left(w_{jk}^{\mathcal{T}}\right)^2 + \mu_A w_{jk}^{\mathcal{T}}]^\top$;
- $\mathbf{\Lambda}_{71} = [-\mu_B,\, \mu_B^2 + \sigma_k^2,\, -\mu_B^3 - 3\sigma_k^2\mu_B]^\top$;
- $\mathbf{\Lambda}_{72} = [1,\, -\mu_B - w_{jk}^{\mathcal{T}},\, \mu_B^2 + 2\sigma_k^2 + \left(w_{jk}^{\mathcal{T}}\right)^2 + \mu_B w_{jk}^{\mathcal{T}}]^\top$;
- $\mathbf{E}_1 = \left[1 - \frac{\sqrt{5}w_{ik}^{\mathcal{T}}}{\gamma_k} + \frac{5\left(w_{ik}^{\mathcal{T}}\right)^2}{3\gamma_k^2},\, \frac{\sqrt{5}}{\gamma_k} - \frac{10w_{ik}^{\mathcal{T}}}{3\gamma_k^2},\, \frac{5}{3\gamma_k^2}\right]^\top$;
- $\mathbf{E}_2 = \left[1 + \frac{\sqrt{5}w_{ik}^{\mathcal{T}}}{\gamma_k} + \frac{5\left(w_{ik}^{\mathcal{T}}\right)^2}{3\gamma_k^2},\, \frac{\sqrt{5}}{\gamma_k} + \frac{10w_{ik}^{\mathcal{T}}}{3\gamma_k^2},\, \frac{5}{3\gamma_k^2}\right]^\top$;
- $\mathbf{E}_3 = [E_{30},\, E_{31},\, E_{32},\, E_{33},\, E_{34}]^\top$;
- $\mathbf{E}_4 = [E_{40},\, E_{41},\, E_{42},\, E_{43},\, E_{44}]^\top$;
- $\mathbf{E}_5 = [E_{50},\, E_{51},\, E_{52},\, E_{53},\, E_{54}]^\top$;

- $E_{30} = 1 + \dfrac{25x_1^2x_2^2 - 3\sqrt{5}\left(3\gamma_k^3 + 5\gamma_k x_1x_2\right)(x_1+x_2) + 15\gamma_k^2\left(x_1^2+x_2^2+3x_1x_2\right)}{9\gamma_k^4}$,

  $E_{31} = \dfrac{18\sqrt{5}\gamma_k^3 + 15\sqrt{5}\gamma_k\left(x_1^2+x_2^2\right) - (75\gamma_k^2 + 50x_1x_2)(x_1+x_2) + 60\sqrt{5}\gamma_k x_1x_2}{9\gamma_k^4}$,

  $E_{32} = \dfrac{5\left[5x_1^2+5x_2^2+15\gamma_k^2 - 9\sqrt{5}\gamma_k(x_1+x_2) + 20x_1x_2\right]}{9\gamma_k^4}$,

  $E_{33} = \dfrac{10\left(3\sqrt{5}\gamma_k - 5x_1 - 5x_2\right)}{9\gamma_k^4}$, and $E_{34} = \dfrac{25}{9\gamma_k^4}$;
- $E_{40} = 1 + \dfrac{25x_1^2x_2^2 + 3\sqrt{5}\left(3\gamma_k^3 - 5\gamma_k x_1x_2\right)(x_2-x_1) + 15\gamma_k^2\left(x_1^2+x_2^2-3x_1x_2\right)}{9\gamma_k^4}$,

  $E_{41} = \dfrac{5\left[3\sqrt{5}\gamma_k\left(x_2^2-x_1^2\right) + 3\gamma_k^2(x_1+x_2) - 10x_1x_2(x_1+x_2)\right]}{9\gamma_k^4}$,

  $E_{42} = \dfrac{5\left[5x_1^2+5x_2^2-3\gamma_k^2 - 3\sqrt{5}\gamma_k(x_2-x_1) + 20x_1x_2\right]}{9\gamma_k^4}$,

  $E_{43} = -\dfrac{50(x_1+x_2)}{9\gamma_k^4}$, and $E_{44} = \dfrac{25}{9\gamma_k^4}$;
- $E_{50} = 1 + \dfrac{25x_1^2x_2^2 + 3\sqrt{5}\left(3\gamma_k^3 + 5\gamma_k x_1x_2\right)(x_1+x_2) + 15\gamma_k^2\left(x_1^2+x_2^2+3x_1x_2\right)}{9\gamma_k^4}$,

  $E_{51} = \dfrac{18\sqrt{5}\gamma_k^3 + 15\sqrt{5}\gamma_k\left(x_1^2+x_2^2\right) + (75\gamma_k^2 + 50x_1x_2)(x_1+x_2) + 60\sqrt{5}\gamma_k x_1x_2}{9\gamma_k^4}$,

  $E_{52} = \dfrac{5\left[5x_1^2+5x_2^2+15\gamma_k^2 + 9\sqrt{5}\gamma_k(x_1+x_2) + 20x_1x_2\right]}{9\gamma_k^4}$,

  $E_{53} = \dfrac{10\left(3\sqrt{5}\gamma_k + 5x_1 + 5x_2\right)}{9\gamma_k^4}$, and $E_{54} = \dfrac{25}{9\gamma_k^4}$;
- $\mu_A = \mu_k - \dfrac{\sqrt{5}\sigma_k^2}{\gamma_k}$, $\mu_B = \mu_k + \dfrac{\sqrt{5}\sigma_k^2}{\gamma_k}$, $\mu_C = \mu_k - \dfrac{2\sqrt{5}\sigma_k^2}{\gamma_k}$, $\mu_D = \mu_k + \dfrac{2\sqrt{5}\sigma_k^2}{\gamma_k}$.

REFERENCES

[1] Y. Andrianakis and P. G. Challenor, *Parameter Estimation and Prediction Using Gaussian Processes*, Tech. report, University of Southampton, Southampton, UK, 2009.

[2] R. Baptista, Y. Marzouk, K. Willcox, and B. Peherstorfer, *Optimal approximations of coupling in multidisciplinary models*, AIAA J., 56 (2018), pp. 2412–2428.

[3] J. Beck and S. Guillas, *Sequential design with mutual information for computer experiments (MICE): emulation of a tsunami model*, SIAM/ASA J. Uncertain. Quantif., 4 (2016), pp. 739–766, https://doi.org/10.1137/140989613.

[4] A. Chaudhuri, R. Lam, and K. Willcox, *Multifidelity uncertainty propagation via adaptive surrogates in coupled multidisciplinary systems*, AIAA J., 56 (2018), pp. 235–249.

[5] K. R. Dalbey, *Efficient and Robust Gradient Enhanced Kriging Emulators*, Tech. Report SAND2013–7022, Sandia National Laboratories, Albuquerque, NM, 2013.

[6] P. Embrechts, F. Lindskog, and A. Mcneil, *Chapter 8 - Modelling dependence with copulas and applications to risk management*, in Handbook of Heavy Tailed Distributions in Finance, S. T. Rachev,

ed., Vol. 1, North-Holland, Amsterdam, 2003, pp. 329–384.
[7] H. Fazeley, H. Taei, H. Naseh, and M. Mirshams, *A multi-objective, multidisciplinary design optimization methodology for the conceptual design of a spacecraft bi-propellant propulsion system*, Struct. Multidiscip. Optim., 53 (2016), pp. 145–160.
[8] T. E. Fricker, J. E. Oakley, and N. M. Urban, *Multivariate Gaussian process emulators with nonseparable covariance structures*, Technometrics, 55 (2013), pp. 47–56.
[9] R. B. Gramacy, *Surrogates: Gaussian Process Modeling, Design, and Optimization for the Applied Sciences*, CRC Press, Boca Raton, FL, 2020.
[10] M. Gu and J. O. Berger, *Parallel partial Gaussian process emulation for computer models with massive output*, Ann. Appl. Stat., 10 (2016), pp. 1317–1347.
[11] M. Gu, X. Wang, and J. O. Berger, *Robust Gaussian stochastic process emulation*, Ann. Statist., 46 (2018), pp. 3038–3066.
[12] E. Hawkins, R. S. Smith, J. M. Gregory, and D. A. Stainforth, *Irreducible uncertainty in near-term climate projections*, Clim. Dynam., 46 (2016), pp. 3807–3819.
[13] R. Jandarov, M. Haran, O. Bjørnstad, and B. Grenfell, *Emulating a gravity model to infer the spatiotemporal dynamics of an infectious disease*, J. R. Stat. Soc. Ser. C. Appl. Stat., 63 (2014), pp. 423–444.
[14] R. H. Johnstone, E. T. Chang, R. Bardenet, T. P. De Boer, D. J. Gavaghan, P. Pathmanathan, R. H. Clayton, and G. R. Mirams, *Uncertainty and variability in models of the cardiac action potential: Can we build trustworthy models?*, J. Mol. Cell. Cardiol., 96 (2016), pp. 49–62.
[15] J. E. Kay, C. Deser, A. Phillips, A. Mai, C. Hannay, G. Strand, J. M. Arblaster, S. Bates, G. Danabasoglu, J. Edwards, M. Holland, P. Kushner, J.-F. Lamarque, D. Lawrence, K. Lindsay, A. Middleton, E. Munoz, R. Neale, K. Oleson, L. Polvani, and M. Vertenstein, *The community earth system model (CESM) large ensemble project: A community resource for studying climate change in the presence of internal climate variability*, Bull. Am. Meteorol. Soc, 96 (2015), pp. 1333–1349.
[16] S. Kodiyalam, R. Yang, L. Gu, and C.-H. Tho, *Multidisciplinary design optimization of a vehicle system in a scalable, high performance computing environment*, Struct. Multidiscip. Optim., 26 (2004), pp. 256–263.
[17] K. N. Kyzyurova, J. O. Berger, and R. L. Wolpert, *Coupling computer models through linking their statistical emulators*, SIAM/ASA J. Uncertain. Quantif., 6 (2018), pp. 1151–1171, https://doi.org/10.1137/17M1157702.
[18] S. Marque-Pucheu, G. Perrin, and J. Garnier, *Efficient sequential experimental design for surrogate modeling of nested codes*, ESAIM Probab. Stat., 23 (2019), pp. 245–270.
[19] T. P. Minka, *Expectation Propagation for Approximate Bayesian Inference*, preprint, https://arxiv.org/abs/1301.2294, 2013.
[20] T. Rainforth, R. Cornish, H. Yang, A. Warrington, and F. Wood, *On nesting Monte Carlo estimators*, Proc. Mach. Learn. Res., 80 (2018), pp. 4267–4276.
[21] C. E. Rasmussen and C. K. Williams, *Gaussian Processes for Machine Learning*, MIT Press, Cambridge, 2006.
[22] J. Rougier, S. Guillas, A. Maute, and A. D. Richmond, *Expert knowledge and multivariate emulation: The thermosphere–ionosphere electrodynamics general circulation model (TIE-GCM)*, Technometrics, 51 (2009), pp. 414–424.
[23] D. Salmanidou, S. Guillas, A. Georgiopoulou, and F. Dias, *Statistical emulation of landslide-induced tsunamis at the Rockall Bank, NE Atlantic*, Proc. A., 473 (2017), 20170026.
[24] S. Sankararaman and S. Mahadevan, *Likelihood-based approach to multidisciplinary analysis under uncertainty*, J. Mech. Design, 134 (2012), 031008.
[25] F. Sanson, O. Le Maitre, and P. M. Congedo, *Systems of Gaussian process models for directed chains of solvers*, Comput. Methods Appl. Mech. Eng., 352 (2019), pp. 32–55.
[26] A. Santiago, J. Aguado-Sierra, M. Zavala-Aké, R. Doste-Beltran, S. Gómez, R. Arís, J. C. Cajas, E. Casoni, and M. Vázquez, *Fully coupled fluid-electro-mechanical model of the human heart for supercomputers*, Int. J. Numer. Meth. Bio., 34 (2018), e3140.
[27] T. J. Santner, B. J. Williams, W. Notz, and B. J. Williams, *The Design and Analysis of Computer Experiments*, Springer, New York, 2003.

[28] T. W. Simpson, T. M. Mauery, J. J. Korte, and F. Mistree, *Kriging models for global approximation in simulation-based multidisciplinary design optimization*, AIAA J., 39 (2001), pp. 2233–2241.
[29] M. L. Stein, *Interpolation of Spatial Data: Some Theory for Kriging*, Springer, New York, 1999.
[30] P. M. Tagade, B.-M. Jeong, and H.-L. Choi, *A Gaussian process emulator approach for rapid contaminant characterization with an integrated multizone-CFD model*, Build. Environ., 70 (2013), pp. 232–244.
[31] W. Thuiller, M. Guéguen, J. Renaud, D. N. Karger, and N. E. Zimmermann, *Uncertainty in ensembles of global biodiversity scenarios*, Nat. Commun., 10 (2019), 1446.
[32] T. Ulrich, S. Vater, E. H. Madden, J. Behrens, Y. van Dinther, I. van Zelst, E. J. Fielding, C. Liang, and A.-A. Gabriel, *Coupled, physics-based modeling reveals earthquake displacements are critical to the* 2018 *Palu, Sulawesi Tsunami*, Pure Appl. Geophys., 176 (2019), pp. 4069–4109.
[33] B. Zhang, B. A. Konomi, H. Sang, G. Karagiannis, and G. Lin, *Full scale multi-output Gaussian process emulator with nonseparable auto-covariance functions*, J. Comput. Phys., 300 (2015), pp. 623–642.
[34] W. Zhao, Y. Wang, and C. Wang, *Multidisciplinary optimization of electric-wheel vehicle integrated chassis system based on steady endurance performance*, J. Clean Prod., 186 (2018), pp. 640–651.