Article:

Astfalck, LC, Cripps, EJ, Gosling, JP orcid.org/0000-0002-4072-3022 et al. (1 more author) (2019) Emulation of vessel motion simulators for computationally efficient uncertainty quantification. Ocean Engineering, 172. pp. 726-736. ISSN 0029-8018

https://doi.org/10.1016/j.oceaneng.2018.11.059

Emulation of vessel motion simulators for computationally efficient uncertainty quantification

L.C. Astfalck[a], E.J. Cripps[b], J.P. Gosling[c], I.A. Milne[d,*]

[a]*School of Mechanical and Chemical Engineering, The University of Western Australia, Perth, Australia*

[b]*School of Mathematics and Statistics, The University of Western Australia, Perth, Australia*

[c]*School of Mathematics, University of Leeds, Leeds, UK*

[d]*Centre for Offshore Foundation Systems, The University of Western Australia, Perth, Australia*

Abstract

The development and use of numerical simulators to predict vessel motions is essential to design and operational decision making in offshore engineering. Increasingly, probabilistic analyses of these simulators are being used to quantify prediction uncertainty. In practice, obtaining the required number of model evaluations may be prohibited by time and computational constraints. Emulation reduces the computational burden by forming a statistical surrogate of the model. The method is Bayesian and treats the numerical simulator as an unknown function modelled by a Gaussian process prior, with covariances of the model outputs constructed as a function of the covariances of the inputs. In offshore engineering, simulator inputs include directional quantities and we describe a way to build this information into the covariance structure. The methodology is discussed with reference to a numerical simulator which computes the mean turret offset amplitude of a FPSO in response to environmental forcing. It is demonstrated through statistical diagnostics that the emulator is well designed, with evaluations executed around 60,000 times faster than the numeric simulator. The method is generalisable to many offshore engineering numerical simulators that require directional inputs and is widely applicable to industry.

Keywords: Gaussian process emulation, FPSO vessel motions, directional inputs, uncertainty quantification, Bayesian statistics

1. Introduction

Using numerical simulators to represent the behaviour of complex physical systems is central to design and operational decision making in offshore engineering. For example, simulating the hydrodynamic responses of floating facilities is used to assess the motions of spread-moored vessels during squalls (Legerstee et al., 2006), predict the heading and motions of vessels given steady metocean conditions (Milne et al., 2016; Milne and Zed, 2018), model ship stability due to roll motions (Surendran and Reddy, 2003), and understand the hydrodynamics of side-by-side offloading (Zhao et al., 2018). While capable of providing valuable point predictions such simulators would also benefit from a common statistical framework to quantify the simulator's uncertainty to produce statistical predictions. By propagating the distributions of the input parameters through a simulator, Monte Carlo (MC) techniques are readily available (Saltelli et al., 2000, 2004) but at the cost of many thousands of simulator runs. This becomes challenging when each simulator

[*] Corresponding author
Email address: ian.milne@uwa.edu.au (I.A. Milne)

run is expensive and/or results are needed quickly (O'Hagan, 2006), as in operational phases where decisions must regularly be made reactively and promptly to changing conditions. This article describes a method to quantify vessel motion simulators' predictive uncertainties at significantly reduced computational cost by approximating the simulator with a statistical surrogate.

The process of approximating a computationally intensive simulator with a cheaper model is known in the literature as emulation (Oakley and O'Hagan, 2002). The idea is to model comparatively few simulator outputs from a data analytic perspective and use the results to predict the behaviour of the simulator at unobserved locations. Many techniques are available, including neural networks (Schiller and Doerffer, 1999; Knutti et al., 2006; Sanderson et al., 2008), regression models (Rougier et al., 2009; Sexton et al., 2011; Williamson et al., 2013) and Gaussian processes (Sacks et al., 1989; Oakley and O'Hagan, 2002; O'Hagan, 2006; Conti et al., 2009). Within the statistics community the principal technique used for emulation involves Gaussian processes (GPs) [1] (Bastos and O'Hagan, 2009), a stochastic process similar to kriging in spatial statistics. Marrel et al. (2008) compare GPs to other emulation techniques and conclude that it is a good and judicious alternative to both simple linear regression models, and more complex methods such as neural networks and boosting trees. We adopt a Bayesian approach by treating the simulator as an unknown function whose prior distribution is a GP. Updating the prior with simulator outputs yields posterior predictive densities at unobserved input locations. The attractions of Bayesian GP emulation are their flexibility, analytic tractability, proved model efficiency, and probabilistic quantification of uncertainty (see O'Hagan (2006) for a tutorial).

When appropriately designed, the cheaper model, herein referred to as the emulator, requires orders of magnitude less runs than what is typically needed by a MC based analysis (Kennedy and O'Hagan, 2001). Emulation has been successfully employed across many disciplines, including contributions to research in climate science (Johnson et al., 2015), cosmology (Vernon et al., 2014), complex fluid dynamics modelling (Moonen and Allegrini, 2015), epidemiology (Andrianakis et al., 2015), hydrology (Rajabi et al., 2015), water resources (Razavi et al., 2012), petroleum engineering (Craig et al., 2001), and environmental engineering (Petropoulos et al., 2013). In contrast, to the best of our knowledge, emulation in the offshore engineering literature has only been addressed in Green et al. (2016), who emulate a univariate linear finite element simulator of an offshore platform as part of a Bayesian calibration exercise. We extend this work and offer a comprehensive overview of the topic, methodological advances to allow for multi-dimensional circular and linear inputs, and provide a suite of diagnostic tools.

Under a GP framework, the simulator's outputs are modelled as a (infinite dimensional) multivariate Gaussian distribution, of which any finite subset also has a multivariate Gaussian distribution. In practice, this implies that (given the parameter estimates) the posterior predictive densities are also multivariate Gaussian, updated conditionally on the simulator outputs. The covariance of the GP is typically constructed such that simulator outputs that are close together have higher correlation than outputs that are further apart. An important characteristic of GP emulation is that "close together" and "further apart" are specified by distances in the simulator's input space. This introduces an additional layer of complexity for many offshore engineering processes where simulators require not only inputs defined on the real number line but also directional inputs, known in the statistical literature as "linear" and "circular" inputs, respectively, (Fisher, 1995). The inclusion of both circular and linear inputs in emulation is novel both in theory and application and we demonstrate how they are incorporated into the covariance structure.

We develop a GP emulator to predict the response for a turret-moored Floating Production Storage Offloading (FPSO) vessel, as simulated by a function of environmental forcing. The FPSO of interest is currently mooring in the Australian North–West Shelf, one of Australia's most economically significant maritime regions. This problem is selected due to both the recent industry interest in mooring integrity monitoring (Prislin et al., 2017), and the computational cost of the simulator. The emulation methodology presented

[1] Here the term "Gaussian process" refers to the statistical process, and not the wave description as it has been used in other offshore engineering literature.

can theoretically be used to predict any continuous univariate output from the simulator, of which we analyse the mean offset amplitude from the simulator's stochastic time-domain output. The output stochasticity necessitates an additional noise term within the emulator and we do so following methodologies provided in previous research (Andrianakis and Challenor, 2012; Andrianakis et al., 2015; Johnson et al., 2011). The model input space is defined by the metocean parameters describing the wind, wind-wave, swell and surface currents that result in a ten dimensional input space comprising four circular and six linear variables over which the emulator is constructed. The covariance matrix is constructed as a product of squared exponential functions for linear inputs and C^4–Wendland functions for circular inputs (Gneiting, 2013).

In total 3,000 time-domain simulation runs are used, of which 2,500 are used to train the emulator and 500 are withheld for validation. To capture the joint probability structure of the wind, wind-wave, and swell-wave input parameters we sample them jointly from a 34 year hindcast dataset over the region of interest. Unfortunately, surface current hindcasts were not available over this time span. Instead, we augment the hindcast dataset with samples from the distributional results of a recent expert elicitation workshop on surface currents on the North–West Shelf's Exmouth Plateau as described in Astfalck et al. (2018). It is vital to confirm the emulator is a reasonable surrogate of the simulator and the statistical diagnostics presented in Bastos and O'Hagan (2009) are used to assess the emulator performance. The diagnostics indicate satisfactory performance. Furthermore, the emulator offers a run-time decrease of ∼60,000 times, with a single emulator run executing in ∼0.5 ms as compared to a single simulation run of ∼30 s. Furthermore, the predictive variances are sufficiently small to be meaningful in application. The method presented herein is generalisable and may be applied to many numerical models with univariate and continuous outputs in offshore engineering.

The paper proceeds as follows. Section 2 describes the underlying numerical simulator and the required inputs. Section 3 reviews the concepts behind GP emulation, including the additional considerations required by this application. Section 4 presents the results of the emulation together with the validation diagnostics. Section 5 discusses the significance of the results and outlines some further uses of emulation when analysing simulator performance, and Section 6 summarises the key conclusions of the study. Appendix A provides a more detailed derivation of Bayesian GP emulation for the interested reader. Code used within this paper, and non-confidential synthetic data with which it may be used, have been uploaded to the first author's GitHub repository `https://github.com/astfalckl/emulation`.

2. Description of the numerical simulations

2.1. Simulation methodology

The dynamic response of a turret-moored FPSO to environmental forcing was simulated in the time domain using Ariane7 developed by Bureau Veritas (Bureau Veritas, 2007). The commercial multi-body hydrodynamic software was utilised to compute the turret-offset from a 6-DOF coupled analysis of the vessel and its moorings in response to wind, waves and currents. The analyses accounted for the inherent non-linearities and inertias associated with the combined low and high wave frequency dynamic responses of the vessel. The simulations were three hours in duration, which excluded a 3,000 s initialisation period and used a time step of one second. The mean turret-offset amplitude was computed from the time histories through the relation $|\frac{1}{T}\sum_{t=1}^{T}\mathbf{r}_t|$, where $\mathbf{r}_t$ is a vector position in terms of Easting and Northing, referenced to a fixed global reference, for each time measurement, t. A screen shot of a single Ariane7 output time-series is shown in Figure 1; the left section of the graph denotes the initialisation period where data was not used in training the emulator. Figure 2 shows the frequency domain response of the FPSO's mean offset amplitudes. It is seen that the FPSO mooring response is dominated by the low frequency responses, with the higher wave frequency components having a comparatively smaller influence — indicative of a relatively compliant mooring system. Nevertheless, the Ariane time-domain simulation is set-up to ensure that both constituents are accounted for in the calculation of mean offset amplitude.

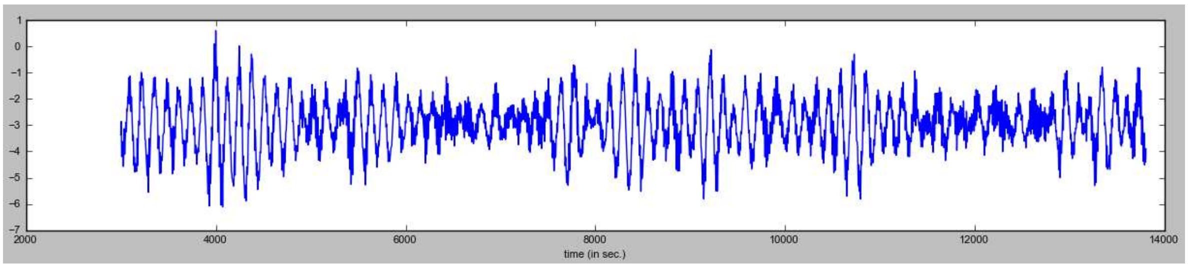


Figure 1: A single Ariane7 output time-series of mean offset amplitude, generated with a randomly sampled metocean state from the hindcast data.

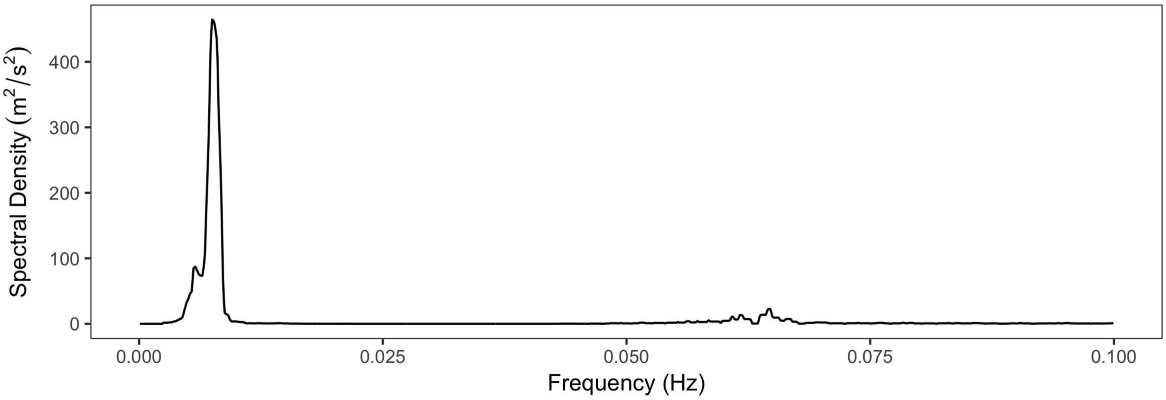


Figure 2: Frequency domain responses of the FPSO's mean offset amplitudes.

2.2. Vessel properties

The FPSO considered in the study has a length of approximately 350 m, a beam of 60 m and is moored in a water depth of approximately 350 m in the North–West Shelf of Australia. The internal turret mooring system comprises steel chain lines arranged in a three-by-three configuration which are fixed to the seabed using drag anchors, as seen in Figure 3. The simulations were performed for typical (non-extreme) operating conditions corresponding to a mean draft of 12 m and displacement of 180,000 Te.

2.3. Site Properties

The environmental conditions at the facility include temporally variable surface currents, wind, wind-waves, and swell. The dominant swell primarily originates from the South–West, and wind, surface currents, and wind-waves exhibit large directional variation. The wind and wave parameters which describe the environment at the site were acquired from a numerical hindcast dataset based on the NCEP Climate Forecast System Reanalysis (CFSRv2) (Saha et al., 2010) and the third generation WaveWatch III spectral wave model (Tolman, 2014), which had previously been verified against field measurements. The wind data comprised the mean wind speed and direction at 10 m elevation. The wind-waves and swell are individually described in terms of JONSWAP parameters (i.e. significant wave height, peak spectral period, mean direction and the peak enhancement factor) (Hasselmann et al., 1973). Hindcast data of surface currents were however deemed to not be reliable, owing to their particularly high spatial and temporal variability. In lieu of field measurement of the currents, results of a recent expert elicitation of the surface currents at the site were utilised (Astfalck et al., 2018). Severe tropical cyclones are expected during the Wet season, during which the vessel is expected to disconnect. As such, simulating the vessel response in cyclonic conditions is not considered.

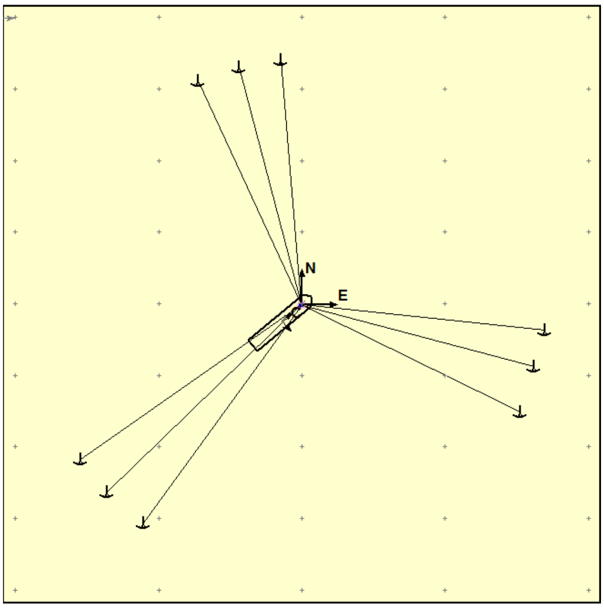


Figure 3: Mooring configuration of the turret-moored FPSO.

2.4. Input parameters

Ten input parameters represent the variable environmental input space of Ariane7. Each of these inputs is listed in Table 1. Directional and magnitude components of wind and surface current are used. Each wave state is described by a direction, significant wave height, and peak crossing period and the other JONSWAP parameters are assumed temporally constant. The circular inputs, $x_{1:4}$, are normalised on $[0, 2\pi)$, and the linear inputs, $x_{5:10}$, are normalised on $(0, 1]$. Normalisation aids both the numerical stability of the emulator and serves to confidentialise the data for publication. A-priori, all parameters listed in Table 1 are considered to affect the FPSO mean offset amplitude and are therefore included for consideration in modelling.

Table 1: Model input parameters. The symbol ° denotes angular measurements in degrees relative to compass North.

	Parameter	Unit
x_1	Surface current direction	°
x_2	Wind direction	°
x_3	Swell direction	°
x_4	Wind–wave direction	°
x_5	Surface current magnitude	m s^{-1}
x_6	Wind magnitude	m s^{-1}
x_7	Swell significant wave height	m
x_8	Swell peak period	s
x_9	Wind-wave significant wave height	m
x_{10}	Wind-wave peak period	s

3. Gaussian process emulation

GP emulation considers the simulator to be an unknown function f(·), modelled by a GP prior, where f(·) returns univariate output, y, from a set of p–dimensional inputs, $\mathbf{x} \in \mathscr{X}$. Herein the mean offset amplitude is represented by y and the inputs in Table 1 are represented by $\mathbf{x}$. The GP prior represents the belief that the predictive distribution of any output y, conditioned on its input $\mathbf{x}$, is Gaussian. Furthermore, the joint distribution of the outputs of any finite collection of elements in the input space follows a multivariate Gaussian distribution. If $\mathscr{X}$ is continuous, the set of all possible inputs in $\mathscr{X}$ results in a infinite-dimensional Gaussian distribution from which a realisation of the function may be sampled. In practice, we only consider the finite collection of training and prediction locations. Mathematically, this prior belief is expressed as

$$\mathrm{f}(\cdot)|\Theta \sim \mathcal{GP}(m_0(\cdot), \lambda_0(\cdot,\cdot))$$

where $m_0(\cdot)$ is the mean function, $\lambda_0(\cdot,\cdot)$ is the covariance function, and $\Theta = [\beta, \sigma^2, \psi]$ is a vector of all unknown hyperparameters. The mean function models the global trends in the data, and is most commonly represented as the linear regression model $m_0(\mathbf{x}) = \mathbf{h}(\mathbf{x})^T\beta$. Here, $\mathbf{h}(\cdot)$ is a vector of m known regression functions, $h_1(\cdot), \ldots, h_m(\cdot)$, and β is an unknown vector of coefficients. More advanced mean structures have been used (for instance in Vernon et al. (2010)); however, due to the satisfactory results seen in Section 4 these structures are deemed not necessary herein. The covariance function is defined as $\lambda_0(\mathbf{x}, \mathbf{x}') = \sigma^2 c(\mathbf{x}, \mathbf{x}'; \psi)$, where σ^2 is an unknown amplitude parameter, and $c(\cdot,\cdot)$ is a known correlation function dependent on unknown parameters ψ. The correlation function controls important aspects of the emulator, such as smoothness and periodicity; further details pertinent to our application are discussed in Section 3.1.1.

Given a set of training data, where the simulator has been run n times producing outputs $\mathbf{y} = \{y_1, \ldots, y_n\}$ from corresponding inputs $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathscr{X}$, the likelihood of the data can be expressed as

$$\mathbf{y}|\beta, \sigma^2, \psi \sim \mathcal{MN}(H\beta, \sigma^2\Sigma), \tag{1}$$

where $H = [\mathbf{h}(\mathbf{x}_1), \ldots, \mathbf{h}(\mathbf{x}_n)]^T$, and Σ is a matrix with i,jth elements $\Sigma_{i,j} = c(\mathbf{x}_i, \mathbf{x}_j; \psi), i, j \in \{1, \ldots, n\}$. As detailed in Appendix A, by setting a prior over the hyperparameters and invoking Bayes' theorem we can first make inference about the posterior distribution of the hyperparameters (Equation A.6). Subsequently the posterior predictive distribution can be calculated using Equations A.4 and A.5 to predict simulator outputs for any input in $\mathscr{X}$. It is the posterior predictive distribution that forms the emulator. Because sampling from the posterior predictive distribution involves straightforward linear algebra, emulator predictions are orders of magnitude faster than executing the simulator.

Despite its simplicity, the GP emulator is highly flexible and capable of capturing very non-linear model behaviour (see O'Hagan (1978) for early work on flexible curve fitting using GPs). The emulation literature has shown that these assumptions are generally appropriate. In any case, a collection of statistical diagnostics are employed in Section 4 to validate the emulator. For the sake of clarity and brevity we have omitted much of the mathematics from the main body of this text and provide a more thorough mathematical description of GP emulation in Appendix A.

3.1. Adapting emulation for application to vessel motion modelling

The above methodology can be applied to the emulation of vessel motion simulations with two important extensions. The first is the specification of a correlation function to incorporate knowledge about both the circular and linear inputs. The second is to incorporate stochasticity in the model outputs—this is to reflect the knowledge that repeated executions of Ariane7 at the same inputs result in randomly varying outputs due to random wave-state sampling in the time-domain.

3.1.1. Specification of the correlation function

By selecting an appropriate correlation function, $c(\cdot, \cdot; \psi)$, we can incorporate our knowledge about the emulator properties, such as output smoothness and periodicity. When multiplied by the amplitude parameter, σ^2, this yields a non-negative definite variance-covariance matrix for any set of inputs in $\mathscr{X}$. Generally, in emulation, the correlation function reflects that “nearness” in the input space implies high correlation in the outputs and the converse. Figure 4 shows the evident change in model behaviour from different correlation functions. Rasmussen (2004) provides a comprehensive review of correlation functions and their applications.

In the present application, the input space is defined by four circular and six linear parameters; hence, we define $c(\cdot,\cdot;\psi)$ for both types of inputs. This is achieved by specifying the global correlation function as a product of correlation functions for each dimension in $\mathscr{X}$. We define $c(\cdot,\cdot;\psi)$ as

$$c(\mathbf{x},\mathbf{x}';\psi) = \prod_{m=1}^{4} c_{C^4}(x_m, x'_m;\psi) \prod_{n=5}^{10} c_{SE}(x_n, x'_n;\psi).$$

Here, $c_{SE}(\cdot,\cdot;\psi)$ and $c_{C^4}(\cdot,\cdot;\psi)$ denote the families of squared exponential and C^4–Wendland correlation functions respectively. An implication of the squared exponential function is that samples from the GP are infinitely differentiable. As we wish to model the belief that a small change in inputs imparts a small change in the outputs, this assumption of smoothness is appropriate. The squared exponential correlation function is defined as

$$c_{SE}(x_n, x'_n;\rho_n) = \exp\left(-\frac{|x_n - x'_n|^2}{2\rho_n}\right), \qquad \rho_n > 0,$$

where ρ_n is a length-scale parameter governing output smoothness for each linear input n. Larger values of ρ_n reflect a smoother output surface. This effect is demonstrated in Figure 4 with a simple toy simulator containing one input parameter and emulated using the squared exponential correlation function. Figures 4a and 4b show the GP emulator results when $\rho_1 = 1$ and $\rho_1 = 0.05$. The underlying true function is denoted by the black dashed line and the red dots are the training points for the emulator. For each case a single sample from the GP is shown by the blue solid line, and 0.95 predictive intervals are shaded in grey. Smoother estimates of the underlying function and tighter predictive densities are apparent for $\rho_1 = 1$ than $\rho_1 = 0.05$. Note also in Figure 4 that at the training points the emulator's predictive uncertainties are zero. This is because the toy simulator is deterministic so the underlying function is known exactly at these points. In this sense the mathematical foundation of the simulator is encoded into the emulator, which in turn uses standard results of multivariate Gaussian random variables to quantify posterior predictions at unobserved locations.

The C^4–Wendland function is a class of circular correlation function, defined in Gneiting (1999) as

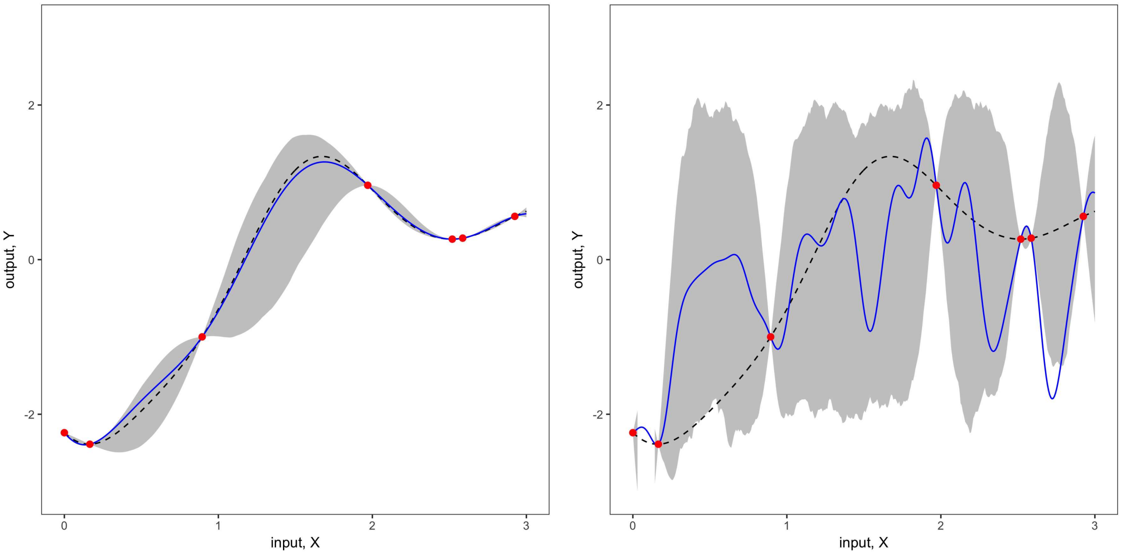
$$c_{C4}(x_m, x'_m;\tau_m) = \left(1 + \tau_m \frac{d(x_m, x'_m)}{\pi} + \frac{(\tau_m^2 - 1)}{3}\frac{d(x_m, x'_m)^2}{\pi^2}\right)\left(1 - \frac{d(x_m, x'_m)}{\pi}\right)_+^{\tau_m}, \qquad \tau_m \geq 6,$$

where $(t)_+ = \max(t, 0)$, and $d(x_m, x'_m)$ is the geodesic distance, $d(x_m, x'_m) = \arccos\left(\cos\left(x_j - x'_j\right)\right)$, between two angles on the unit circle. Similar to the squared exponential function, τ_m governs the output smoothness for each circular input m, with larger values of τ_m reflecting a rougher output surface.

By defining the correlation function as such, we assume that the underlying function is homogeneous along each dimension, but as a higher dimensional surface is anisotropic. This means that differing degrees of smoothness can be used to describe response change between dimensions, but this smoothness does not change within dimensions. Methods for building GPs over heterogeneous models is an ongoing topic of discussion within the statistics community (for example see Pope et al. (2018)), but we do not require such methodologies herein.

3.1.2. Allowance for output stochasticity

The majority of the emulation literature deals with functions that are deterministic, such that repeated evaluations of the model with identical inputs results in identical outputs. However, for time-domain simulations such as Ariane7, repeated executions of the same inputs result in slight variations in output due to



(a) $\rho = 1$

(b) $\rho = .05$

Figure 4: Effect of varying length-scale, ρ, on the GP predictions. The true function (black dashed line), the 95% prediction intervals (grey boundary), and a single sample (blue solid line) are given for both GPs. The red points were provided to both GPs as training points. The approximation in (a), where the appropriate values have been used is clearly more suitable than that shown in (b).

different wave train histories with varying seeds. Thus, an allowance should be made for the fact that we can not observe the true output value. There are many approaches to the emulation of stochastic simulators (Andrianakis and Challenor, 2012; Andrianakis et al., 2015; Johnson et al., 2011). Here, we add a term to the diagonal of the prior distribution's variance–covariance matrix, such that $\tilde{\Sigma} = \Sigma + \nu I$, as in Rasmussen (2004). This addition changes the likelihood in Equation 1 to be

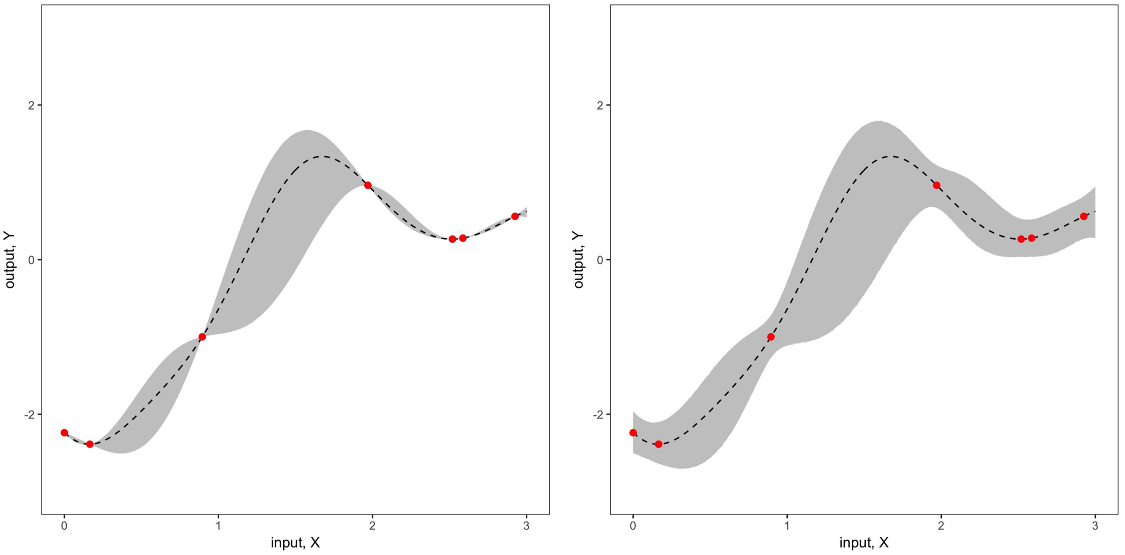
$$\mathbf{y}|\beta, \sigma^2, \psi \sim \mathcal{MN}(H\beta, \sigma^2(\Sigma + \nu I)).$$

The addition of ν in the prior distribution accounts for noise in the observed data but without further augmentations it would not account for noise in the predictions. As emulation is concerned with predicting simulation outputs, rather than modelling the underlying process, we must include prediction noise. To do this we substitute $\tilde{\Sigma}_{**} = \Sigma_{**} + \nu I$ in the posterior variance, where Σ_{**} is a matrix with i, jth elements $\Sigma_{**_{i,j}} = c(\mathbf{x}_i^*, \mathbf{x}_j^*; \psi), i, j \in \{1, \ldots, n^*\}$. Here, we use $*$ to denote predictive quantities, such that the $\mathbf{x}_i^*$'s are the locations of the prediction inputs, and n^* is the number of predictions. The effects that the addition of ν has on the emulator are demonstrated in Figure 5, which shows the same toy example as in Figure 4 with and without the effects of including ν in prediction. The values of ρ_1 are constant in both Figures. By including ν, rather than interpolating the points (Figure 5a), the emulator will approximate the points with uncertainty (Figure 5b). In emulation, this addition is only suitable when the simulator outputs are stochastic, as the false inclusion of a noise term may lead to unsuitable estimates of the other hyperparameters.

De Oliveira (2007) shows that after the addition of the noise term in $\tilde{\Sigma}$ and $\tilde{\Sigma}_{**}$, the non-informative prior $p(\beta, \sigma^2) \propto \sigma^{-2}$ remains conjugate. Following the procedure in Appendix A, similar results are produced to that of Equations A.4 and A.5, differing only in that Σ and Σ_{**} are replaced by $\tilde{\Sigma}$ and $\tilde{\Sigma}_{**}$. The value that ν takes may be estimated along with ψ. Equation $A.6$ is adapted to incorporate ν such that

$$p(\psi, \nu|\mathbf{y}) \propto p(\psi, \nu)|\tilde{\Sigma}|^{-1/2}|H^T\tilde{\Sigma}^{-1}H|^{-1/2}(\hat{\sigma}^2)^{-(n-q)/2}.$$

A maximum-a-posteriori estimate is then made on the parameters ν and ψ.



(a) $\nu = 0$

(b) $\nu = 0.05$

Figure 5: The effects of including the noise term, ν, in the GPs. The true function (black dashed line), and the 95% prediction intervals (grey boundary) are shown for both GPs. The red points were provided to both GPs as training points. Effects of ν can be seen where in (a) when it is zero, training points are interpolated exactly. Conversely, in (b) where ν is non-zero, the training points have an associated noise.

4. Emulation results and diagnostics

4.1. Model specification and results

Calculations of mean offset amplitude resulting from a series of numerical simulations of FPSO trajectories in response to varying environmental conditions are emulated using the methodology presented in Section 3. Table 1 lists the considered inputs; this results in a ten dimensional input space with both circular and linear variables. A numerical hindcast dataset is used to inform wind, wind-wave, and swell parameters, and the results of an expert elicitation workshop are used to inform surface current parameters (Astfalck et al., 2018).

As with many other offshore engineering applications, the input space is highly correlated and complex. Fortunately, a 34 year dataset of hourly hindcast observations for wind, wind-wave and swell is available. We note that whilst the hindcast data are not real observations, it is widely understood that these data form good approximations to the real metocean processes (Chawla et al., 2013). To prevent designing the emulator in regions of negligible probability, we select our training data by uniformly randomly sampling time-measurements from the hindcast dataset. This exploits the inherent correlated structure present in the inputs, allowing us to obtain random draws from the joint distribution of wind, wind-wave, and swell. The elicitation results in Astfalck et al. (2018) present seasonal distributions for surface current magnitude and direction, at the location of the FPSO. We augment the training samples from the hindcast dataset with seasonally matched samples from the elicited distributions of surface current. This method of design offers a pragmatic approach for designing the emulator, without the requirement to specify *a priori* the complicated, dependent probability distributions of the input parameters. Furthermore, this reduces the effective size of the input space over which the emulator is to be constructed. A consequence of this design method is the loss of space filling afforded by other popular methods such as Latin hyper-cube sampling. However, given the highly correlated and complex structure of the input space, this is deemed as acceptable. We iteratively generated training points and monitored the root mean squared error, and the average prediction variance — both as a function of the number of training points. These results are shown in Figure 6. Relatively little increases in accuracy and precision are seen after 2500 training points, and thus we deem this as an acceptable sample size for our training data. In total we use 2,500 Ariane7 executions for training and 500 for validation. The design process of the emulator is illustrated in Figure 7.

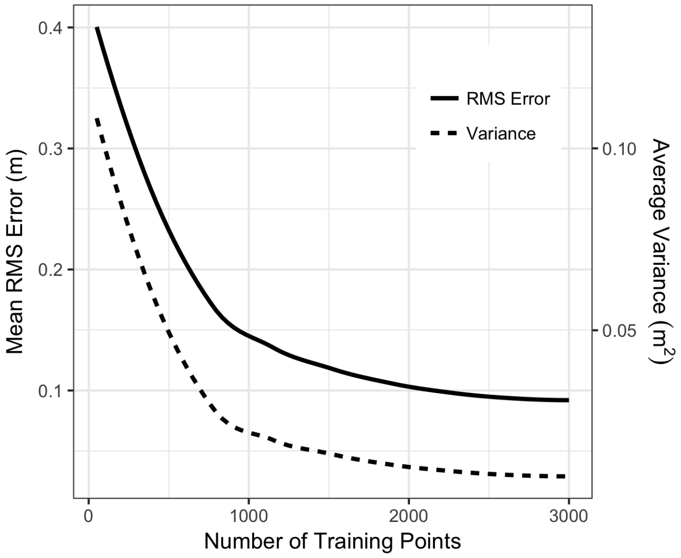
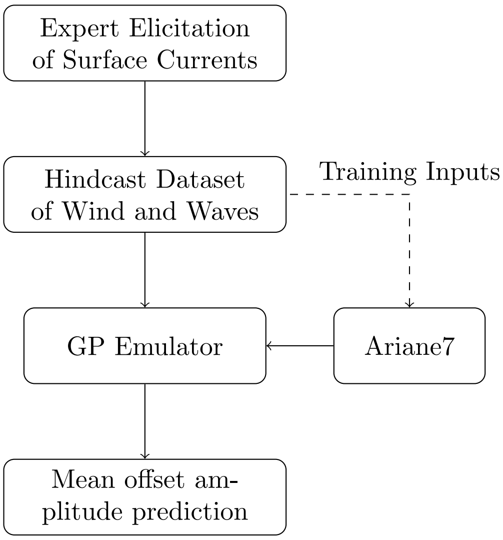


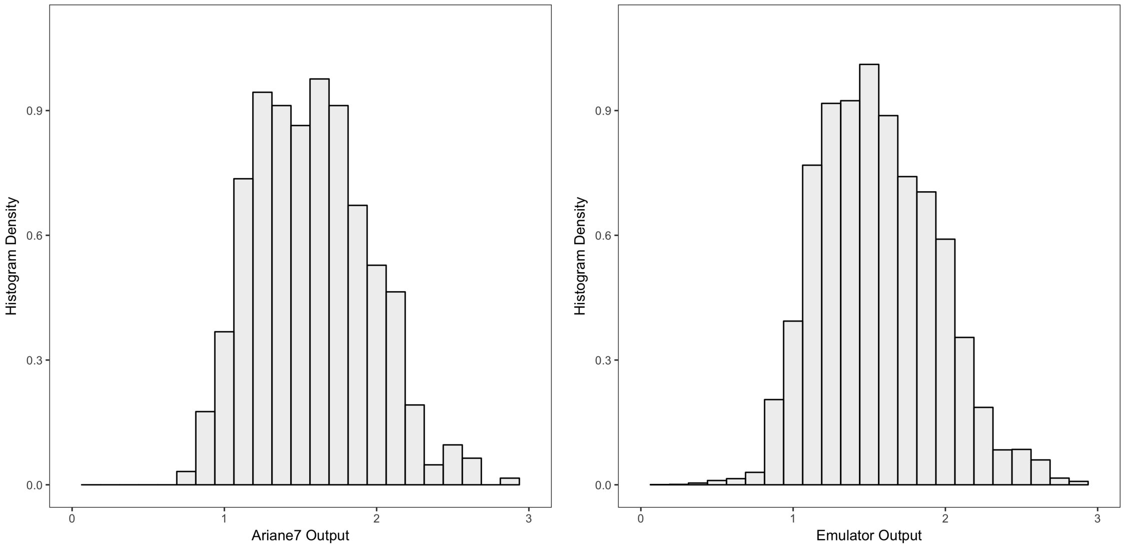
Figure 6: The average root mean squared error between the emulator and the simulator, and the average prediction variance of the emulator.



Expert Elicitation of Surface Currents
Hindcast Dataset of Wind and Waves
Training Inputs
GP Emulator
Ariane7
Mean offset amplitude prediction

Figure 7: Flowchart depicting the emulation methodology. The results of an expert elicitation are used to augment the the hindcast dataset. Training inputs are then run through Ariane7, and the resulting outputs are used to train the emulator. Once the emulator is trained predictions of mean offset amplitude, given a metocean condition, are made.

With regard to the primary motivation of reducing simulation time, the trained emulator offers a run-time decrease of ∼60,000 times, where a single emulator run requires ∼0.5 ms compared to a single Ariane7 run of ∼30 s. As an initial indicator of performance we conduct an uncertainty analysis on the mean offset amplitude and compare validation (unobserved) data obtained from the simulator with outputs of the emulator run at the same locations. Figure 8 provides histograms of the distribution of the mean offset amplitude (measured in metres) from the Ariane7 training runs (Figure 8a), and the emulator (Figure 8b). In general, the agreement is favourable, with relatively small differences in the tails which is attributable to emulator uncertainty. However, the total simulation run times required to produce these histograms are 14,700 s for Ariane7, and 0.27 s for the emulator. This reinforces the advantages that emulation holds, as estimated predictive values, and their uncertainty, are able to be obtained in near-real time. For instance, this would be useful in informing predictions of vessel motions during operational phases.



(a)

(b)

Figure 8: Histograms to show the distribution of the mean offset amplitude (measured in metres) from (a) the Ariane7 training runs, and (b) the emulator.

4.2. Validation diagnostics

Due to the input dimensionality, validation of the emulator by simple inspection is challenging, as visualisation of high-dimensional surfaces is difficult and often impossible. Rather, we utilise a collection of statistical diagnostics presented in Bastos and O'Hagan (2009), to analyse the emulator performance. We examine the prediction variances, individual prediction errors, coverage, and prediction intervals. Visual presentations of these diagnostics are shown in Figure 9. These measures are selected as they diagnose common issues in emulation and are easily interpretable. A more comprehensive overview of emulator diagnostics is available in Bastos and O'Hagan (2009).

The diagnostics presented in Bastos and O'Hagan (2009) assume that the simulator is deterministic, meaning that the true simulator outputs can be observed. As discussed in Section 3.1.2 this is not the case with Ariane7. Its inherent stochasticity is however quantified by ν — calculated to be $\nu = 5 \times 10^{-3}$. As this value is several orders of magnitude less than the simulation outputs, for the purpose of diagnostic calculations, we consider the simulator stochasticity to be negligible.

4.2.1. Prediction variances

Figure 9a plots the empirical density of the variances, $\mathrm{Var}[\mathbf{y}^*|\mathbf{x}_i^*]$, of the predicted outputs $\mathbf{y}^*$ with corresponding inputs $\mathbf{x}^*$. This is calculated by Equation A.5. For emulator predictions to be useful, the predictive variances must be smaller than a stipulated threshold, often determined as a percentage of the total output variance of the simulator, $\mathrm{Var}[\mathbf{y}]$. The requisite precision is application specific, as more critical applications may necessitate more precise predictions. We use the threshold 0.1Var[$\mathbf{y}$], and Figure 9a shows that 91.4% of the data lies underneath this bound, which is indicated by the dotted line. This suggests that the majority of emulator predictions are precise enough to be meaningful, at this threshold. Large prediction variances are symptomatic of validation points lying far from the training points. The implemented random sampling design of the training and validation points makes covering the entire input space with the training data difficult. Despite this, we find by using a random sampling design the emulator is in general acceptably meaningful in its predictions.

4.2.2. Individual prediction errors

The standardised individual prediction errors are calculated as the difference between the simulator's outputs and the emulator's mean predictions, standardised by the predictive standard deviations. Mathematically this is defined as

$$D_i^I(\mathbf{y}^*) = \frac{y_i^* - M_2(\mathbf{x}_i^*|\psi)}{\sqrt{\Lambda_2(\mathbf{x}, \mathbf{x}_i^*; \psi)}},$$

for each validation point, i, where $M_2(\mathbf{x}_i^*|\psi)$ and $\Lambda_2(\mathbf{x}, \mathbf{x}_i^*; \psi)$ are defined in Equations A.4 and A.5 as

$$M_2(\mathbf{x}^*; \psi) = H^*\hat{\beta} + \Sigma_*^T \Sigma_{**}^{-1}(\mathbf{y} - H\beta), \text{ and}$$
$$\Lambda_2(\mathbf{x}, \mathbf{x}^*; \psi) = \hat{\sigma}^2[\Sigma_{**} - \Sigma_*^T \Sigma^{-1} \Sigma_* + (H - \Sigma_*^T \Sigma^{-1} H)(\Sigma_*^T \Sigma^{-1} \Sigma_*)^{-1}(H - \Sigma_*^T \Sigma^{-1} H)^T].$$

Given the emulator's Gaussian assumption, the $D_i^I(\mathbf{y}^*)$ have standard Student-t distributions with $n-1$ degrees of freedom, where n is the number of training points. As n is large in this case, we can approximate the errors to be normally distributed. Thus, values of $D_i^I(\mathbf{y}^*)$ located far outside regions of reasonable probability for a standard normal distribution indicate disagreement between the emulator and the simulator. This is useful to diagnose issues in emulation, possibly indicating an inappropriate choice of mean function, poor estimation of hyperparameters, false assumption of model homogeneity, or non-Gaussian distributed outputs.

Figure 9b compares the empirical density of the $D_i^I(\mathbf{y}^*)$ (solid black line) to the standard normal distribution (grey dashed line). The small excess mass in the tails of the empirical distribution suggests minor disagreement between the emulator and the simulator. We calculate 97% of the validation data to be bounded by $[-3, 3]$, where 99% is expected. Investigation into the locations of the points not bounded by $[-3, 3]$ show them to be isolated and random, suggesting no significant structure in the error. It is noted that this does not necessarily diagnose a fault in the emulator, as complex simulators can be highly non-linear and may behave erroneously, thus breaking the assumption of simulator homogeneity. This point is further discussed in Section 4.2.4 in the context of the prediction intervals of the emulator.

4.2.3. Coverage

Emulator coverage, also known as the D^α diagnostic, calculates the proportion of the validation points lying within the emulator's marginal prediction intervals of size α. For a given nominal coverage, α, the diagnostic is evaluated as

$$D^\alpha(\mathbf{y}^*) = \frac{1}{m}\sum_{i=1}^{m} \mathbb{1}(\mathbf{y}_i^* \in \mathrm{PI}_i(\alpha)),$$

where $\mathbb{1}(\cdot)$ is an indicator function, $\mathrm{PI}_i(\alpha)$ denotes the predictive interval bounding α probability at input i, and m is the total number of validation points. Should the emulator accurately approximate the simulator we expect the value of α and D^α to be close. When $D^\alpha > \alpha$, this indicates accurate prediction of the true values, however with excessive uncertainty. Conversely, should $D^\alpha < \alpha$ the emulator does not predict the true values; this may either be due to too-small prediction variance or a more fundamental disagreement between the emulator and the simulator.

Figure 9c plots the nominal coverage, α, against the empirical coverage, D^{α}. It can be seen that the emulator slightly over-predicts the simulator outputs in the interval $\alpha \in [0.25, 0.75]$, and slightly under-predicts towards the tail values. Due to random sampling variation, slight deviation from the diagonal is expected—especially towards higher values of α. The discrepancy between α and D^{α} is minor, and we conclude that the emulator predicts with reasonable accuracy throughout its entire predictive distribution.

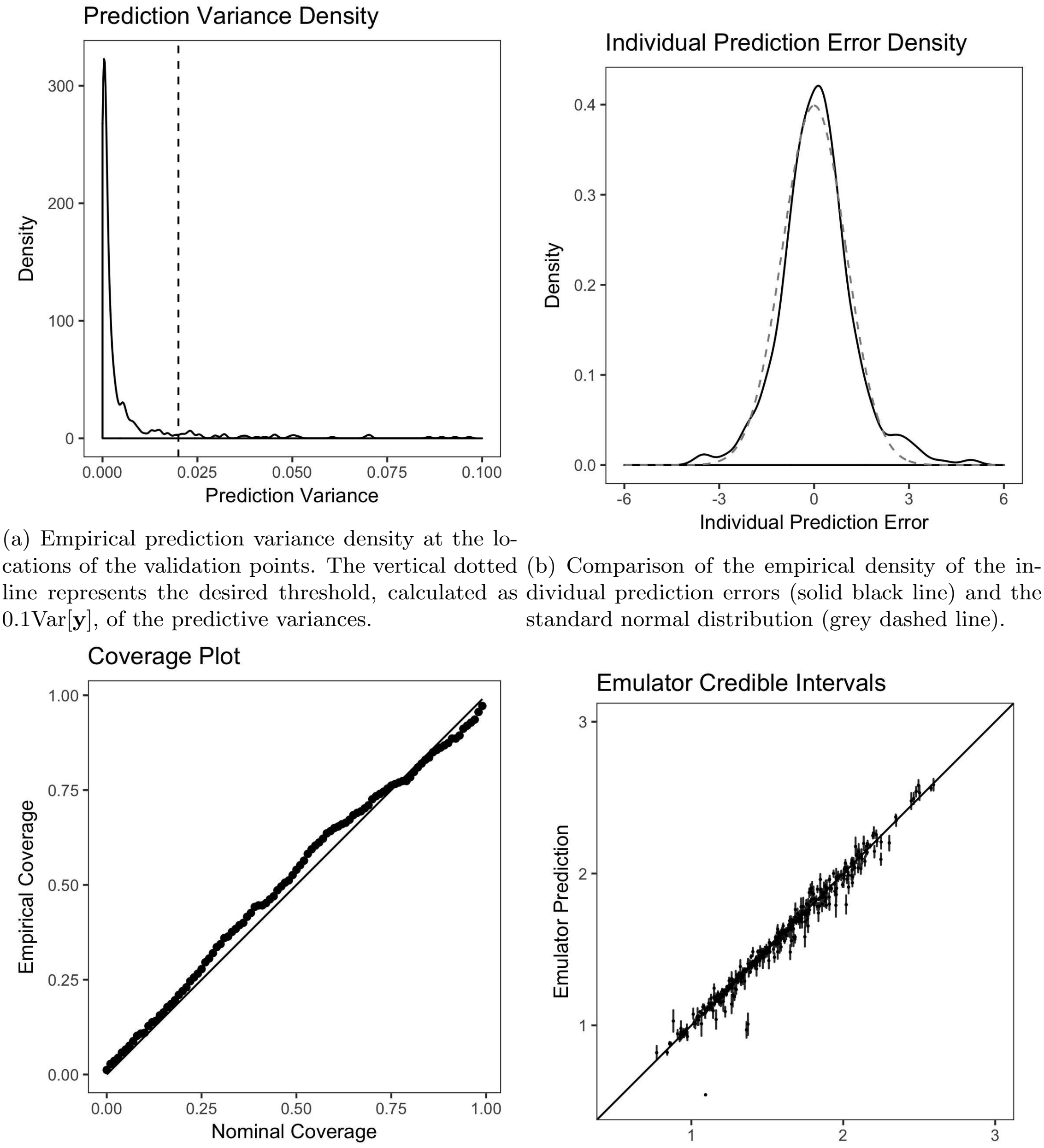
4.2.4. Prediction intervals

Finally, in Figure 9d, we plot the Ariane7 outputs against the emulator's 50% prediction intervals. This may help diagnose structure in the error of the emulator, as a function of the output value. An example of where this is useful would be to diagnose an emulator that regularly under-predicts large values. As we analyse the 50% prediction intervals, under the assumption of output independence, we expect that half of the plotted lines intersect the diagonal, with all lines being close. However, there are three locations where there is a serious discrepancy between the emulator and the simulator. These points are all located towards low Ariane7 output values. Again, this does not necessarily diagnose a fault in the emulator, as many computationally expensive models are complicated, and often unexpected and erratic behaviour may occur. In such situations, it is unreasonable for the emulator to predict this behaviour. When the emulator is expected to perform well, and there are large disagreements, a simulation developer may be notified to aid with further exploration. This is especially salient for simulators with high-dimensional input spaces, as dimensionality may prevent assiduous exploration of the simulator. Thus, further to efficient computation we may also use emulation to help diagnose erroneous simulator outputs.

5. Discussion

With the goal of increasing the computational efficiency of predicting mean offset amplitude from Ariane7, we demonstrate the application of a Gaussian process emulator. We achieve a time increase of ~60,000 times, and diagnostic results indicate that the emulator predicts Ariane7 accurately and precisely. The mean offset amplitude is of particular interest to the ocean engineering community for assessing mooring integrity. The theory presented is the same for any continuous univariate output, and thus may be adapted to a wide range of applications. We note that the emulation methodology presented within is not capable of simulating full time-series outputs, but rather predicts univariate statistical summaries of the simulation. Though information from the simulation time-series output may be lost (for instance the turret-position rate of change), many statistical summaries remain that are important to decision making and to capturing the hydrodynamic non-linearities. Should more complicated output structures be required, extensions to to allow for multivariate (Conti and O'Hagan, 2010), functional (Bayarri et al., 2009), and time-series outputs (Conti et al., 2009) are available. Whilst the run time of Ariane7 is not prohibitive to some statistical analyses, for instance the global uncertainty analysis shown in Figure 8a, emulation enables a much richer class of analyses; this may include short-term uncertainty analysis for operational decisions based on simulator predictions, sensitivity analysis, and Bayesian model calibration—all of which are of increasing importance to the offshore engineering industry, as discussed below.

A natural extension of this theory is to use the emulator for operational predictions with near-real time uncertainty quantification. For instance, given an uncertain and impending weather forecast, samples may be propagated through the emulator to provide an uncertain output prediction. Such a Monte Carlo based uncertainty analysis may require thousands of runs, which the emulator can perform in a matter of seconds. Further, this may be computed on-board the vessel using fairly standard desktop computers, mitigating data security concerns associated with advanced computing techniques if data must be transmitted to shore for analysis. The current state-of-the-art for vessel motion prediction relies either on heavily simplified models, more advanced simulations that require substantial computational resources—typically external to the ship, or the vessel-captain's subjective experience. Emulation can allow for the information embedded in complex numerical simulators to be used locally, at low cost and efficiently.



(a) Empirical prediction variance density at the locations of the validation points. The vertical dotted line represents the desired threshold, calculated as 0.1Var[$\mathbf{y}$], of the predictive variances.

(b) Comparison of the empirical density of the individual prediction errors (solid black line) and the standard normal distribution (grey dashed line).

(c) The D^{α} coverage plot showing the proportion of validation points lying within the emulator's marginal predictive intervals.

(d) Ariane7 outputs plotted against the emulator's 50% prediction intervals.

Figure 9: Emulation diagnostics adapted from Bastos and O'Hagan (2009). All diagnostics have been calculated using the validation points.

Sensitivity analysis calculates the relative importance of each simulator input via examination of their influence on the simulator output. Examples of sensitivity analysis being implemented in the offshore engineering community may be found in Huang et al. (2017); Eldin and Kim (2016); Xu et al. (2015). Many techniques are available, however we provide discussion with respect to variance-based sensitivity analysis (Saltelli et al., 2000; Chan et al., 1997; Oakley and O'Hagan, 2004): the most widely implemented method for sensitivity analysis amongst the statistics community (see Saltelli and Annoni (2010) for a tutorial). To ensure this is done efficaciously, Saltelli and Annoni (2010) advocate for over 10,000k model runs to be used, where k is the input dimensionality of the model. This estimation is under the assumption of independent inputs; note that for dependent inputs—as demanded by many metocean processes—this number may be orders of magnitude larger as the assumptions made in Saltelli et al. (2010), for the efficient sampling of model runs, are violated. Computationally less intensive methodologies for sensitivity analysis are available, however many of these are not as meaningful in their results as their variance-based brethren. Furthermore, to the best of the authors knowledge, variance-based methods have been the only methods proposed that may allow for dependent inputs—as necessitated by offshore modelling. Thus, the use of emulation to facilitate variance-based methods for sensitivity analysis is desirable.

Model calibration has always been of importance to numerical simulation developers. For the offshore engineering community this plays a particularly important role when model-scale or field data are incorporated in the simulations (for example in Jiao et al. (2018) and Hifi and Barltrop (2015)). The model uncertainty due to uncertain input parameter estimates is often challenging to capture. Kennedy and O'Hagan (2001) present a Bayesian method to calibrate numerical simulators, whereby all sources of uncertainty are included in the calibration. This has since become a standard practice amongst the statistics community (Santner et al., 2003). The method demonstrated in Kennedy and O'Hagan (2001) however requires Markov-chain-Monte-Carlo (MCMC) sampling of the simulator, with changing input parameter values. MCMC sampling frequently involves hundreds of thousands of iterations—where a larger input space often necessitates more samples—and the simulator must be re-evaluated each time. Bayesian calibration is thus generally not feasible for use with expensive simulators. Combining emulation with Bayesian calibration offers an attractive means to overcome this hurdle in offshore engineering applications such vessel motion prediction.

By increasing the breadth of application for Ariane7, and other numerical simulators, more detailed insights and predictions may be garnered. Complex numerical simulators are predominantly used in design phases, and not in operations, primarily due to their computational expense. Whilst many design analyses are enabled by emulation, so to are a range of operational applications. Emulation builds upon the information contained inside of complex simulators, and embeds this knowledge into a efficient modelling framework to probabilistically predict at unobserved locations. Thus, the use of emulation does not supplant the need for numerical simulators; instead, emulation may ultimately lead to the wider uptake of both existing and new simulators.

6. Conclusion

Motivated by the need to overcome the burden of prohibitive simulation run times, emulation has been applied to model the turret offset of a FPSO in response to environmental forcing. The requirements for incorporating directional inputs and stochastic outputs are identified. Both the theoretical development of a covariance structure to account for input directionality, and the application of emulation, are particularly novel to offshore engineering. The trained emulator offers a run-time increase of ∼60,000 times, where a single emulator run requires ∼0.5 ms compared to a single Ariane7 run of ∼30 s. Four diagnostics are used to assess the appropriateness of the emulator design, and they indicate satisfactory performance. Reduction of the computational burden induced by numerical simulators may enable a collection of different analyses, including probabilistic analyses such as near-real time uncertainty quantification, sensitivity analysis, and Bayesian model calibration. It is demonstrated that emulation offers a new and robust technique to reduce computational expense and is generally applicable to a wide range of offshore engineering problems.

7. Acknowledgements

This research was supported by the ARC Industrial Transformation Research Hub for Offshore Floating Facilities which is funded by the Australian Research Council, Woodside Energy, Shell, Bureau Veritas and Lloyds Register (Grant No. IH140100012)

References

Andrianakis, I., Challenor, P.G., 2012. The effect of the nugget on Gaussian process emulators of computer models. Computational Statistics & Data Analysis 56, 4215–4228.

Andrianakis, I., Vernon, I.R., McCreesh, N., McKinley, T.J., Oakley, J.E., Nsubuga, R.N., Goldstein, M., White, R.G., 2015. Bayesian history matching of complex infectious disease models using emulation: a tutorial and a case study on HIV in Uganda. PLoS computational biology 11, e1003968.

Astfalck, L.C., Cripps, E.J., Gosling, J.P., Hodkiewicz, M.R., Milne, I.A., 2018. Expert elicitation of directional metocean parameters. Ocean Engineering 161C, 268–276.

Bastos, L.S., O'Hagan, A., 2009. Diagnostics for Gaussian process emulators. Technometrics 51, 425–438.

Bayarri, M.J., Berger, J.O., Kennedy, M.C., Kottas, A., Paulo, R., Sacks, J., Cafeo, J.A., Lin, C.H., Tu, J., 2009. Predicting vehicle crashworthiness: Validation of computer models for functional and hierarchical data. Journal of the American Statistical Association 104, 929–943.

Bureau Veritas, 2007. Ariane7 Theoretical Manual.

Chan, K., Saltelli, A., Tarantola, S., 1997. Sensitivity analysis of model output: variance-based methods make the difference, in: Proceedings of the 29th conference on Winter simulation, IEEE Computer Society. pp. 261–268.

Chawla, A., Spindler, D.M., Tolman, H.L., 2013. Validation of a thirty year wave hindcast using the climate forecast system reanalysis winds. Ocean Modelling 70, 189–206.

Conti, S., Gosling, J.P., Oakley, J.E., O'Hagan, A., 2009. Gaussian process emulation of dynamic computer codes. Biometrika 96, 663–676.

Conti, S., O'Hagan, A., 2010. Bayesian emulation of complex multi-output and dynamic computer models. Journal of statistical planning and inference 140, 640–651.

Craig, P.S., Goldstein, M., Rougier, J.C., Seheult, A.H., 2001. Bayesian forecasting for complex systems using computer simulators. Journal of the American Statistical Association 96, 717–729.

De Oliveira, V., 2007. Objective Bayesian analysis of spatial data with measurement error. Canadian Journal of Statistics 35, 283–301.

Eldin, M.N., Kim, J., 2016. Sensitivity analysis on seismic life-cycle cost of a fixed-steel offshore platform structure. Ocean Engineering 121, 323–340.

Fisher, N.I., 1995. Statistical analysis of circular data. Cambridge University Press.

Gneiting, T., 1999. Correlation functions for atmospheric data analysis. Quarterly Journal of the Royal Meteorological Society 125, 2449–2464.

Gneiting, T., 2013. Strictly and non-strictly positive definite functions on spheres. Bernoulli 19, 1327–1349.

Green, P., Tygesen, U., Stevanovic, N., 2016. Bayesian modelling of offshore platforms, in: Model Validation and Uncertainty Quantification, Volume 3. Springer, pp. 15–22.

Hasselmann, K., Barnett, T.P., Bouws, E., Carlson, H., Cartwright, D.E., Enke, K., Ewing, J., Gienapp, H., Hasselmann, D.E., Kruseman, P., et al., 1973. Measurements of wind-wave growth and swell decay during the Joint North Sea Wave Project (JONSWAP). Ergänzungsheft 8-12 .

Hifi, N., Barltrop, N., 2015. Correction of prediction model output for structural design and risk-based inspection and maintenance planning. Ocean Engineering 97, 114–125.

Huang, X., Ni, Z., Zhang, Z., Hua, H., 2017. Stiffness optimization of marine propulsion shafting system by FRF-based substructuring method and sensitivity analysis. Ocean Engineering 144, 243–256.

Jiao, J., Sun, S., Li, J., Adenya, C.A., Ren, H., Chen, C., Wang, D., 2018. A comprehensive study on the seakeeping performance of high speed hybrid ships by 2.5 d theoretical calculation and different scaled model experiments. Ocean Engineering 160, 197–223.

Johnson, J.S., Cui, Z., Lee, L.A., Gosling, J.P., Blyth, A.M., Carslaw, K., 2015. Evaluating uncertainty in convective cloud microphysics using statistical emulation. Journal of Advances in Modeling Earth Systems 7, 162–187.

Johnson, J.S., Gosling, J.P., Kennedy, M.C., 2011. Gaussian process emulation for second-order Monte Carlo simulations. Journal of Statistical Planning and Inference 141, 1838–1848.

Kennedy, M.C., O'Hagan, A., 2001. Bayesian calibration of computer models. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 63, 425–464.

Knutti, R., Meehl, G., Allen, M., Stainforth, D., 2006. Constraining Climate Sensitivity from the Seasonal Cycle in Surface Temperature . Journal of Climate 19, 109–135.

Legerstee, F., Francois, M., Morandini, C., Le-Guennec, S., 2006. Squall: Nightmare for designers of deepwater West African mooring systems, in: 25th International Conference on Offshore Mechanics and Arctic Engineering, American Society of Mechanical Engineers. pp. 171–176.

Marrel, A., Iooss, B., Van Dorpe, F., Volkova, E., 2008. An efficient methodology for modeling complex computer codes with gaussian processes. Computational Statistics & Data Analysis 52, 4731–4744.

Milne, I., Zed, M., 2018. Full-scale validation of the hydrodynamic motions of a ship derived from a numerical hindcast. Ocean Engineering 168, 83–94.

Milne, I.A., Delaux, S., McComb, P., 2016. Validation of a predictive tool for the heading of turret-moored vessels. Ocean Engineering 128, 22–40.

Moonen, P., Allegrini, J., 2015. Employing statistical model emulation as a surrogate for CFD. Environmental Modelling & Software 72, 77–91.

Oakley, J.E., O'Hagan, A., 2002. Bayesian inference for the uncertainty distribution of computer model outputs. Biometrika 89, 769–784.

Oakley, J.E., O'Hagan, A., 2004. Probabilistic sensitivity analysis of complex models: a Bayesian approach. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 66, 751–769.

O'Hagan, A., 1978. Curve fitting and optimal design for prediction. Journal of the Royal Statistical Society. Series B (Methodological) , 1–42.

O'Hagan, A., 2006. Bayesian analysis of computer code outputs: a tutorial. Reliability Engineering & System Safety 91, 1290–1300.

Petropoulos, G.P., Griffiths, H.M., Tarantola, S., 2013. A sensitivity analysis of the SimSphere SVAT model in the context of EO-based operational products development. Environmental modelling and software 49, 166–179.

Pope, C.A., Gosling, J.P., Barber, S., Johnson, J., Yamaguchi, T., Feingold, G., Blackwell, P., 2018. Modelling spatial heterogeneity and discontinuities using Voronoi tessellations .

Prislin, I., Maroju, S., et al., 2017. Mooring Integrity and Machine Learning, in: Offshore Technology Conference, Offshore Technology Conference.

Rajabi, M.M., Ataie-Ashtiani, B., Simmons, C.T., 2015. Polynomial chaos expansions for uncertainty propagation and moment independent sensitivity analysis of seawater intrusion simulations. Journal of Hydrology 520, 101–122.

Rasmussen, C.E., 2004. Gaussian processes in machine learning, in: Advanced lectures on machine learning. Springer, pp. 63–71.

Razavi, S., Tolson, B.A., Burn, D.H., 2012. Review of surrogate modeling in water resources. Water Resources Research 48.

Rougier, J., Sexton, D., Murphy, J., Stainforth, D., 2009. Analyzing the Climate Sensitivity of the HadSM3 Climate Model Using Ensembles from Different but Related Experiments. Journal of Climate 22(13), 3540–3557.

Sacks, J., Welch, W.J., Mitchell, T.J., Wynn, H.P., 1989. Design and analysis of computer experiments. Statistical science 4, 409–423.

Saha, S., Moorthi, S., Pan, H.L., Wu, X., Wang, J., Nadiga, S., Tripp, P., Kistler, R., Woollen, J., Behringer, D., Liu, H., Stokes, D., Grumbine, R., Gayno, G., Wang, J., Hou, Y.T., Chuang, H.Y., Juang, H.M.H., Sela, J., Iredell, M., Treadon, R., Kleist, D., Delst, P.V., Keyser, D., John Derber, M.E., Meng, J., Wei, H., Yang, R., Lord, S., Dool, H.V.D., Wang, A.K.W., Long, C., Chellia, M., Xue, Y., Huang, B., Schemm, J.K., Ebisuzaki, W., Lin, R., Xie, P., Chen, M., Zhou, S., Higgins, W., Zou, C.Z., Liu, Q., Chen, Y., Han, Y., Cucurull, L., Reynolds, R.W., Rutledge, G., Goldberg, M., 2010. The NCEP climate forecast system reanalysis. Bull. Amer. Meteor. Soc. , 10151057.

Saltelli, A., Annoni, P., 2010. How to avoid a perfunctory sensitivity analysis. Environmental Modelling & Software 25, 1508–1517.

Saltelli, A., Annoni, P., Azzini, I., Campolongo, F., Ratto, M., Tarantola, S., 2010. Variance based sensitivity analysis of model output. design and estimator for the total sensitivity index. Computer Physics Communications 181, 259–270.

Saltelli, A., Chan, K., Scott, E.M., et al., 2000. Sensitivity analysis. volume 1. Wiley New York.

Saltelli, A., Tarantola, S., Campolongo, F., Ratto, M., 2004. Sensitivity analysis in practice: a guide to assessing scientific models. John Wiley & Sons.

Sanderson, B., Knutti, R., Aina, T., Christensen, C., Faull, N., Frame, D.J., Ingram, W.J., Piani, C., Stainforth, D.A., Stone, D.A., Allen, M.R., 2008. Constraints on model response to greenhouse gas forcing and the role of subgrid-scale processes. Computer Physics Communications 21, 2384–2400.

Santner, T.J., Williams, B.J., Notz, W.I., 2003. The design and analysis of computer experiments. Springer Science & Business Media.

Schiller, H., Doerffer, R., 1999. Neural network for emulation of an inverse model operational derivation of Case II water properties from MERIS data. International Journal of Remote Sensing 20, 1735–1746.

Sexton, D., Murphy, J., Collins, M., 2011. Multivariate probabilistic projections using imperfect climate models part 1: outline of methodology. Climate Dynamics 38, 2513–2542.

Surendran, S., Reddy, J.V.R., 2003. Numerical simulation of ship stability for dynamic environment. Ocean Engineering 30, 1305–1317.

Tolman, H.L., 2014. User manual and system documentation of WAVEWATCH III version 4.18.

Vernon, I., Goldstein, M., Bower, R., 2014. Galaxy formation: Bayesian history matching for the observable universe. Statistical science , 81–90.

Vernon, I., Goldstein, M., Bower, R.G., et al., 2010. Galaxy formation: a Bayesian uncertainty analysis. Bayesian Analysis 5, 619–669.

Williamson, D., Goldstein, M., Allison, L., Blaker, A., Challenor, P., Jackson, L., Yamazakil, K., 2013. History matching for exploring and reducing climate model parameter space using observations and a large perturbed physics ensemble. Climate Dynamics 41, 1703–1729.

Xu, S., Wang, X., Wang, L., Meng, S., Li, B., 2015. A thrust sensitivity analysis based on a synthesized positioning capability criterion in DPCap/DynCap analysis for marine vessels. Ocean Engineering 108, 164–172.

Zhao, W., Milne, I.A., Efthymiou, M., Wolgamot, H.A., Draper, S., Taylor, P.H., Taylor, R.E., 2018. Current practice and research directions in hydrodynamics for flng-side-by-side offloading. Ocean Engineering 158, 99–110.

Appendix A. Gaussian Process emulation

Consider a function, $\mathrm{f}(\cdot)$, returning univariate outputs, y, from a set of p–dimensional inputs, $\mathbf{x} \in \mathscr{X} \subseteq \mathbb{R}^p$. Despite the function outputs, y, being theoretically known for any $\mathbf{x} \in \mathscr{X}$, in practice, we must execute software to evaluate $\mathrm{f}(\cdot)$; this is often computationally intensive. From a Bayesian perspective, we thus consider the function $\mathrm{f}(\cdot)$ to be unknown, and assign it a prior distribution. When observations from $\mathrm{f}(\cdot)$ are available, the prior belief of the function is updated to form a posterior distribution, where the uncertainty is zero at the observed locations. Output predictions, y^*, are obtained from the posterior predictive distribution, $p(y^*|y)$, for inputs $\mathbf{x}^* \in \mathscr{X}$. Following the recommendations of Oakley and O'Hagan (2002) we implement a GP prior over $\mathrm{f}(\cdot)$.

The GP prior is a generalisation of the multivariate Gaussian distribution to infinitely many variables, where any finite collection of variables is distributed as a multivariate Gaussian distribution. When representing a function, each variable in the multivariate Gaussian distribution represents a single element in the input space. Mathematically, we represent our prior belief about $\mathrm{f}(\cdot)$ as

$$\mathrm{f}(\cdot)|\beta, \sigma^2, \psi \sim \mathcal{GP}(m_0(\cdot;\beta), \lambda_0(\cdot,\cdot;\Phi)),$$

where the mean function $m_0(\cdot)$ is given by $m_0(\mathbf{x}) = \mathbf{h}(\mathbf{x})^T\beta$ and $\lambda_0(\cdot,\cdot;\Phi)$ denotes the covariance function with hyperparameters $\Phi = [\sigma^2, \psi]$. Here, $\mathbf{h}(\cdot)$ is a vector of m known regression functions, $h_1(\cdot), \dots, h_m(\cdot)$, and β is an unknown vector of coefficients. The non–zero mean function is designed to represent the global trends of the model output over the input space, and is commonly defined as $\mathbf{h} = (\mathbf{1}, \mathbf{x})^T$. The covariance function $\lambda_0(\cdot,\cdot;\Phi)$ is defined as $\lambda_0(\mathbf{x}, \mathbf{x}'; \Phi) = \sigma^2 c(\mathbf{x}, \mathbf{x}'; \psi)$, where σ^2 is an unknown scale parameter and $c(\cdot,\cdot;\psi)$ is a known correlation function with a vector of unknown parameters ψ. The choice of $c(\cdot,\cdot;\psi)$ determines properties of the emulator's structure, for instance, smoothness and periodicity.

Given n realisations from $\mathrm{f}(\cdot)$, $\mathbf{y} = [y_1, \dots, y_n]$, corresponding to inputs $\mathbf{x}_1, \dots, \mathbf{x}_n \in \mathscr{X}$, the likelihood of the observed data is given by the multivariate Gaussian distribution

$$\mathbf{y}|\beta, \sigma^2, \psi \sim \mathcal{MN}(H\beta, \sigma^2\Sigma), \tag{A.1}$$

where $H = [\mathbf{h}(\mathbf{x}_1), \dots, \mathbf{h}(\mathbf{x}_n)]^T$, and Σ is a matrix with i,jth elements $\Sigma_{i,j} = c(\mathbf{x}_i, \mathbf{x}_j; \psi), i, j \in \{1, \dots, n\}$. Given n^* inputs $[\mathbf{x}_1^*, \dots, \mathbf{x}_{n^*}^*]$ where the function has not been observed, with corresponding unknown outputs $\mathbf{y}^*$, the joint distribution of $\mathbf{y}$ and $\mathbf{y}^*$ is the multivariate Gaussian distribution

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{y}^* \end{bmatrix} \sim \mathcal{MN}\left(\begin{bmatrix} H\beta \\ H^*\beta \end{bmatrix}, \sigma^2 \begin{bmatrix} \Sigma & \Sigma_* \\ \Sigma_*^T & \Sigma_{**} \end{bmatrix} \right), \tag{A.2}$$

where $H^* = [\mathbf{h}(\mathbf{x}_1^*), \dots, \mathbf{h}(\mathbf{x}_{n^*}^*)]^T$, Σ_* is the $n \times n^*$ matrix with elements $\Sigma_{i,j} = c(\mathbf{x}_i, \mathbf{x}_j^*; \psi), i \in \{1, \dots, n\}, j \in \{1, \dots, n^*\}$, and Σ_{**} is the $n^* \times n^*$ matrix with elements $\Sigma_{i,j} = c(\mathbf{x}_i^*, \mathbf{x}_j^*; \psi), i, j \in \{1, \dots, n^*\}$. By conditioning on $\mathbf{y}$ the full conditional distribution is given by

$$\mathbf{y}^*|\mathbf{y}, \beta, \sigma^2, \psi \sim \mathcal{MN}(M_1(\mathbf{x}^*; \beta, \Phi), \Lambda_1(\mathbf{x}, \mathbf{x}^*; \Phi)), \tag{A.3}$$

where

$$M_1(\mathbf{x}^*; \beta, \Phi) = H^*\beta + \Sigma_*^T\Sigma^{-1}(\mathbf{y} - H\beta), \text{ and}$$
$$\Lambda_1(\mathbf{x}, \mathbf{x}^*; \Phi) = \sigma^2(\Sigma_{**} - \Sigma_*^T\Sigma^{-1}\Sigma_*).$$

To make inference about the unobserved outputs $\mathbf{y}^*$, we must first account for the unknown hyperparameters β, σ^2 and ψ. Following Kennedy and O'Hagan (2001) and Oakley and O'Hagan (2002), we specify the prior distribution of β and σ^2 as $p(\beta, \sigma^2) \propto \sigma^{-2}$. Combining this with Equation 1 using Bayes' theorem, the posterior distribution is a normal inverse-gamma distribution, given by

$$\beta|\mathbf{y}, \sigma^2, \psi \sim \mathcal{MN}(\hat{\beta}, \sigma^2(H^T\Sigma^{-1}H)), \text{ and}$$

$$\sigma^2|\mathbf{y}, \psi \sim \mathcal{IG}\left(\frac{n-q}{2}, \frac{(n-q-2)\hat{\sigma}^2}{2}\right),$$

where $\hat{\beta} = (H^T\Sigma^{-1}H)^{-1}H^T\Sigma^{-1}\mathbf{y}$ and $\hat{\sigma}^2 = \frac{\mathbf{y}^T(\Sigma^{-1}-\Sigma^{-1}\Sigma_*(\Sigma_*^T\Sigma^{-1}\Sigma_*)^{-1}\Sigma_*^T\Sigma^{-1})\mathbf{y}}{n-q-2}$. Equation A.3 may now be expressed independent of β and σ^2 by integrating them out, where

$$p(\mathbf{y}^*|\mathbf{y}, \psi) = \int\int p(\mathbf{y}^*|\mathbf{y}, \beta, \sigma^2, \psi)p(\beta|\mathbf{y}, \sigma^2, \psi), p(\sigma^2|\mathbf{y}, \psi)d\beta d\sigma^2.$$

This yields the multivariate t distribution

$$\mathbf{y}^*|\mathbf{y}, \psi \sim \mathcal{MT}_{n-q}(M_2(\mathbf{x}^*; \psi), \Lambda_2(\mathbf{x}, \mathbf{x}^*; \psi)),$$

where

$$M_2(\mathbf{x}^*; \psi) = H^*\hat{\beta} + \Sigma_*^T\Sigma^{-1}(\mathbf{y} - H\hat{\beta}), \text{ and} \tag{A.4}$$

$$\Lambda_2(\mathbf{x}, \mathbf{x}^*; \psi) = \hat{\sigma}^2[\Sigma_{**} - \Sigma_*^T\Sigma^{-1}\Sigma_* + (H - \Sigma_*^T\Sigma^{-1}H)(\Sigma_*^T\Sigma^{-1}\Sigma_*)^{-1}(H - \Sigma_*^T\Sigma^{-1}H)^T]. \tag{A.5}$$

Finally, we must estimate the unknown parameters, ψ, in the correlation function, $c(\cdot, \cdot; \psi)$. Due to the analytically intractable structure of the correlation function, there exist no analytical solutions for ψ. A fully Bayesian analysis would estimate ψ probabilistically using computational techniques such as Markov-chain-Monte-Carlo sampling. Alternatively, an estimate of ψ is made, and then assumed to be the truth. For any prior belief $p(\psi)$, it can be shown that

$$\begin{aligned} p(\psi|\mathbf{y}) &\propto p(\psi)\int\int p(\mathbf{y}|\beta, \sigma^2, \psi)p(\beta, \sigma^2)d\beta d\sigma^2 \\ &\propto p(\psi)|\Sigma|^{-1/2}|H^T\Sigma^{-1}H|^{-1/2}(\hat{\sigma}^2)^{-(n-q)/2}, \end{aligned} \tag{A.6}$$

where Σ and $\hat{\sigma}^2$ are functions of ψ. To estimate ψ we take a maximum-a-posteriori estimate by finding the value of ψ for which Equation A.6 is maximised. This does not effect Equations A.4 and A.5 as ψ is only required to calculate values of Σ, Σ_* and Σ_{**}.