University of London
Imperial College of Science, Technology and Medicine
Department of Civil and Environmental Engineering

Data-driven uncertainty quantification and predictive digital twin for offshore piles

Ningxin Yang

Submitted on December 10, 2023

Abstract

Proper estimation of offshore piles is vital to the life and permanence of the foundation in question. However, due to the uncertainties of soil parameters in the field, the offshore piles are greatly affected. The source of soil uncertainties may come from various reasons, such as lack of uniformity between in-situ test and laboratory experiment; spatial variability of soil profile and rationality of the constitutive model, etc. Traditional statistical analysis which is based on Monte Carlo, is time-consuming and laborious. Though Bayesian theorem provides ways to understand and update the uncertainties, the amount of the inference analysis is still computationally heavy, thus bringing big challenges for the pile design. Additionally, to mimic the geotechnical structures and bearing behaviors as accurate as possible, digital twin is seeping into all kinds of engineering problems, enabling to evolve over time to persistently represent a unique physical asset and achieving data-driven decision making process. However, state-of-the-art digital twins are still relying on considerable expertise and deployment resources, leading to an only one-off implementation and remaining limitation on providing adaptive digital models on unique offshore piles.

In this thesis, we hope to introduce probabilistic graphical model (PGM) involving in Bayesian inverse analysis to speed up the calculation and provide reasonable posterior results for the soil parameters. Besides, as a mathematical and rigorous foundation, partially observed PGM is proposed to support the transition from custom defined model towards accessible digital twins at scale. Based on such flexible asset-specific models, the entire loading life-cycle can be incorporated into a digital twin forming a unified and accessible foundation for a wide range of offshore piles. Combined with monitored data, the proposed dynamic updated digital twin provides rapid analysis results for reliable soil parameters and enables intelligent decision making on the pile bearing behaviors.

Contents

List of Tables

List of Figures

Nomenclature

$\boldsymbol{x}$	Vector of input parameter
$\boldsymbol{\varepsilon}$	Gaussian discrepancy
$\boldsymbol{x}^{\mathrm{MAP}}$	Maximum a posterior estimate
$\boldsymbol{x}^{\mathrm{mean}}$	posterior mean
$\boldsymbol{x}^{\mathrm{ML}}$	Maximum likelihood estimate
$\boldsymbol{x}^{s}_{t\_N_{chain}}$	s_{th} samples from N_{th} chain at t_{th} stage
$\boldsymbol{x}_t^{(\star)}$	Candidate samples at t_{th} stage
$\boldsymbol{x}_t^s$	s_{th} samples from input space at t_{th} stage
$\boldsymbol{y}$	Vector of particular evaluation/observation
$\hat{S}_{eff}$	Effective sample size
$\mathcal{D}_{\boldsymbol{X}}$	Input parameters space
$\mathcal{L}(\boldsymbol{x};\mathcal{Y})$	Likelihood function
$\mathcal{M}$	Computational model
$\mathcal{R}$	Residual
$\mathcal{X}_{t\_N_{chain}}$	Samples from N_{th} chain at t_{th} stage
$\mathcal{X}_t$	Samples at t_{th} stage
$\mathcal{Y}$	Set of evaluations/observations across all time stages

$\mathcal{Y}_{\mathrm{t}}$ Evaluation/observation at t_{th} stage

$\pi(\boldsymbol{x})$ Prior

$\pi(\cdot \mid \cdot)$ Conditional Probability density

$\pi(\cdot)$ Probability density

$\pi(\mathcal{Y})$ Evidence

$\tilde{\mathcal{M}}$ Surrogate model

$q(\boldsymbol{x})$ Proposal distribution

S_{min} Threshold effective sample size

Acronyms

AIES Affine invariant ensemble sampler. 20, 21, 26

CDF Cumulative Density Function. 11, 12, 15

CI Confidence Interval. 12

COV Coefficient of variance. xi, 12

MAP Maximum a Posterior. 10, 11

MCMC Markov Chain Monte Carlo. 13, 17, 18, 20, 22, 26, 27

MH Metropolish-Hasting. xi, 18–20

ML Maximum Likelihood. 11

PDF Probability Density Function. 8, 10, 11, 18, 21

QoI Quantities of interest. 28

SIS Sequential importance sampling. xi, 24–26

SISR Sequential importance sampling and resampling. xi, 25

SMC Sequential Monte Carlo. 13, 22, 26

Chapter 1

Introduction

1.1 Background

Offshore monopiles are increasingly favored in wind farm installations due to their advantages in clean energy generation and easy deployment. These cylindrical steel structures are driven into the seabed to provide a stable foundation for wind turbines. While offshore wind energy presents a promising source of clean and sustainable power, the utilization of offshore monopiles has introduced certain engineering challenges. One of the key concerns associated with offshore monopiles is the need to address the potential issues related to excessive pile displacements induced during their installation and operation(Byrne & Houlsby, 2003; Randolph et al., 2005). Excessive pile movements can lead to significant displacements and rotations in supporting structures, which, in turn, may result in damage or structural instability. Consequently, it becomes crucial to accurately predict and manage pile deformations when designing and analyzing support systems for offshore monopile installations.

When soil properties, design load and pile dimensions are acquired from a technical report, estimating the pile response is typically done through empirical solutions in guidelines or numerical simulations. Several design methods have been provided in the design codes (API, 2011; Bhattacharya, 2019) to predict offshore pile $p-y$ curve. However, it is challenging to incorporate all influential factors, such as pile length, soil layer, soil prop-

erties and loading conditions into a simplified empirical model for predicting monopile displacement. More recently, the rapid advancement of computational techniques has led to the increased application of numerical models (Randolph & Gourvenec, 2017; Taborda et al., 2020; Zdravković et al., 2020; Royston et al., 2022). While numerical modeling serves as a potent analytical tool, it demands a substantial number of simulation runs based on observed data. This presents a significant challenge in the context of monopile analysis and prediction, especially when attempting manual back-calculation of soil properties. Because the observed data is acquired incrementally during construction stages, as opposed to simultaneous data collection.

In recent research endeavors, Bayesian probability frameworks have garnered increasing recognition as an efficacious approach for inverse parameter estimation and response prediction (Finno & Calvello, 2005; Nakamura et al., 2011; Hsein Juang et al., 2013; Nguyen & Nestorović, 2016; Wagner et al., 2020; Jin et al., 2021; Tao et al., 2021; Buckley et al., 2023; Tang et al., 2023). In contrast to conventional back-analysis methodologies, which primarily focus on the determination of fixed input variable values, a probabilistic framework takes an approach where the parameters of interest are considered stochastic variables. Subsequently, the updated parameters are expressed in terms of posterior distributions. In such circumstances, the Bayesian framework emerges as a powerful tool within the probabilistic context, facilitating parameter learning and informed decision-making. Based on this, digital twin (DT) can be constructed and enable the real time data exchange between the digital and physical twins. In offshore engineering, in particular, it has shown substantial potential in various domains, including health monitoring, pile penetration and long-term bearing capacities (Wang et al., 2021; Zhao et al., 2023; Stuyts et al., 2023).

In practice, through adaptive Bayesian updating soil parameters and constructing digital twin on offshore piles, a field engineer would benefit from: (1) properly accounting the uncertainties of input variables (2) real-time monitoring and adaptively predicting the pile response in probabilistic setting (3) providing an efficient tool for data-driven decision making on pile operation and design.

1.2 Problem statement

Modern pile installation and proper estimation is becoming increasingly complex and vital to the reliability and permanence of the foundation in question. However, in the construction process, uncertainties and insufficient information about the soil parameters lead to inaccurate predictions of pile-soil response and bearing capacities. The source of soil uncertainties may come from various reasons, such as lack of uniformity between in-situ test and laboratory experiment; spatial variability of soil profile and rationality of the constitutive model, etc. Dealing with different uncertainties sources is a challenging task. One typical soil profile can be illustrated in Figure 1.1, which shows uncertainties sources:

- Fluctuating curve indicates the spatial variability
- Non-uniformity exists between in-situ test and laboratory experiment

Furthermore, geotechnical engineering problems inherently belong to the high-dimensional realm with substantial uncertainties. The quantity of unknown distribution parameters may become excessively large to be inferred accurately from the limited sample size within the available dataset, resulting in an underdetermined problem. Although some well-established approaches for fitting pile deformations are proposed to infer the underlying soil parameters and reduce the uncertainties, this task becomes nontrivial when the number of input variables is large (i.e., $\mathcal{O}(10^2 - 10^4)$) (Lataniotis, 2019). Even if an adequate probabilistic input model can be obtained, performing the inference analysis through Monte Carlo simulation is still expensive. This poses challenges in understanding uncertainties and providing timely predictions for pile design. In such cases, the underlying model is substituted by a surrogate. In high dimension, however, the performance of surrogate models decreases, while the cost of computing and storing them increases. This is a well-known issue known as the curse of dimensionality (Verleysen & François, 2005). The surrogate computation may even be intractable when the number of input parameters is large (Lataniotis, 2019).

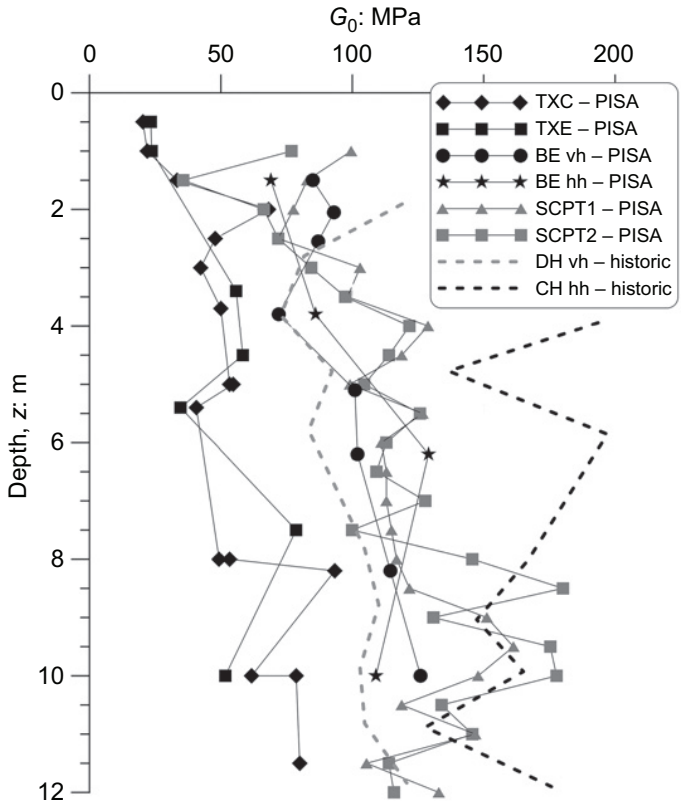


Figure 1.1: Stiffness characteristics at Cowden. Source: Zdravković et al. (2020)

To address these challenges in real time, digital twin (DT) has gained popularity in handling abundant data and predict the pile's response in a more organized and accurate manner (Wang et al., 2021). DT makes full use of data such as physical models, sensor updates and operating history, and integrates simulation processes to real-time reproduce the dynamics of a physical system in the virtual space. More importantly, the DT model cannot only describe the current state of the physical entity, but also predict the future state. However, state-of-the-art digital twins are still relying on considerable expertise and deployment resources (Kapteyn et al., 2021), leading to an only one-off implementation and remaining limitation on providing adaptive digital models on unique offshore piles. Thus, unified and scalable models should be developed and incorporated into digital twin to enable a intelligent decision making. Details about the need for unified and scalable digital twin for piles are outlined next.

1.3 Urgent need for a unified and scalable digital twins for piles

The demand for efficiency, reliability, and safety continues to grow in offshore wind foundation constructions. Computational models are an invaluable tool for understanding complex pile behaviors for new designs, operating conditions and control strategies. This reduces the need for costly experiments or field tests. However, insights gained from a computational model are contingent on the model being an accurate reflection of the underlying soil parameters. Moreover, real-world pile foundations are constantly changing and evolving throughout their lifecycle. Using a single static computational model that ignores these differences fundamentally limits the specificity, and thus the accuracy, of the model and any insights gained through its use.

While the value proposition of digital twin has become widely appreciated in geotechnical engineering, the pile design process remains in a custom production phase. Current digital twin for offshore piles are still bespoken, relying on highly specialized implementations and thus requiring considerable resources and expertise to deploy and maintain. Therefore, it is necessary to move toward digital twins at scale by developing a rigorous and unified mathematical foundation. Based on a robust computational approach and probabilistic graphical model (Kapteyn et al., 2021), this unified mathematical foundation enables a promising application at scale in the offshore pile bearing response.

1.4 Objectives and outline

The underlying objective of this thesis is to develop a robust and scalable digital twin model for offshore piles. This endeavor will leverage cutting-edge methodologies in surrogate modeling and uncertainty quantification, all geared towards facilitating extensive predictive digital twin capabilities. In particular, the specific goals of this thesis are:

- Develop a surrogate model suitable for offshore piles characterized by high input dimensions.

- Accelerate Bayesian inversion calculations for soil parameters to reduce the uncertainties, and providing real-time pile response predictions through adaptive enrichment of observed monitoring data.

- Develop a unifying mathematical foundation for predictive digital twins for offshore piles in the form of a probabilistic graphical model.

Chapter 2

Bayesian probabilistic theory

In probabilistic theory, two main interpretations prevail: frequentist and Bayesian. The frequentist perspective views probabilities as the long-term frequencies observed in infinite trials. For example, in this context, the statement implies that, over many coin flips, heads are expected roughly half the time.

On the other hand, the Bayesian interpretation associates probability with uncertainty and information, rather than repeated trials. From the Bayesian viewpoint, the statement suggests an equal likelihood of the coin landing heads or tails in the next toss.

Depending on the amount of available data, which may range from zero to infinite, various techniques may be used:

- when no data is available to characterize the input parameters, a probabilistic model may be prescribed purely by expert judgment;
- when a large amount of data is available, the tools of statistical inference may be fully applied, like the method of moments (Wagner et al., 2020);
- when both expert judgment and very limited observations are available, Bayesian inference may be resorted to.

One big advantage of the Bayesian interpretation is that it can be used to model our events that do not have long term frequencies. Take, for example, the assessment

of the probability of structural damage to a high-rise building, the collapse of a tunnel, or the occurrence of irreversible deformation in bridge piers.This event is anticipated to occur only a limited number of times over the structure's lifetime and is not expected to happen repeatedly. Nevertheless, we ought to be able to quantify our uncertainty about this event and take appropriate actions (see chapter 3 and chapter 4).

Since data collection is inherently constrained during the progression of most engineering projects, Bayesian theory stands out as a highly effective method. Therefore, this thesis exclusively explores Bayesian methods next, while detailed information on frequentist approaches can be found in Murphy (2012).

2.1 Bayesian inference

When dealing with a limited number of data points, direct statistical estimation becomes unreliable due to substantial statistical uncertainty in the sample estimates. In this context, *Bayesian inference* provides a solution by integrating prior knowledge on parameters with a small set of observed data points. Operating in this fully probabilistic setting, all unknowns are treated as random vectors. Distribution parameters can be denoted by $\boldsymbol{x}$ as realisations of the random vector $\boldsymbol{X} : \Omega \to \mathcal{D}_{\boldsymbol{X}}$. *Quantities of interest* gathered from output are gathered in a vector $\boldsymbol{y} \in \mathbb{R}^{N_{\text{out}}}$. The joint probability distribution of the combined random vector $(\boldsymbol{X}, \boldsymbol{Y}) : \Omega \to \mathcal{D}_{\boldsymbol{X}} \times D_{\boldsymbol{Y}}$ is represented by $\pi(\boldsymbol{x}; \boldsymbol{y})$. Leveraging the fundamental *sum rule* and *product rule* in probabilistic theory, the Probability Density Function (PDF) of the parameters and the data can be expressed as

$$\pi(\boldsymbol{x}|\boldsymbol{y}) = \frac{\mathcal{L}(\boldsymbol{x}; \boldsymbol{y}) \cdot \pi(\boldsymbol{x})}{\pi(\boldsymbol{y})} \tag{2.1}$$

which is also known as *Bayes' theorem* or *Bayes' rule*. In Bayesian terminology, this distribution $\pi(\boldsymbol{x}|\boldsymbol{y})$ is called the posterior distribution and it is calculated by prior $\pi(\boldsymbol{x})$, likelihood $\mathcal{L}(\boldsymbol{x}; \boldsymbol{y}) \overset{\text{def}}{=} \pi(\boldsymbol{y}|\boldsymbol{x})$ and the evidence $\pi(\boldsymbol{y})$. These definitions of the likelihood function and evidence strictly hold only for a single data point $\mathcal{Y} = \{\boldsymbol{y}\}$, but can be generalised to multiple data points easily $\mathcal{Y} \overset{\text{def}}{=} \{\boldsymbol{y}^{(1)}, \cdots, \boldsymbol{y}^{(N)}\}$. These terms in Equation (2.1) have practical significance that we will briefly summarise next.

- Prior $\pi(\boldsymbol{x})$: In the Bayesian paradigm, before considering the data the parameters $\boldsymbol{x}$ are treated as realisations from a random vector $\boldsymbol{X}$ which is assumed to follow the so-called prior distribution.

- Likelihood function $\mathcal{L}(\boldsymbol{x};\mathcal{Y})$: The likelihood function is a measure of how well the prescribed parametric distribution $\pi(\mathcal{Y}|\boldsymbol{x})$ describes the data. In most engineering cases, input parameters $\boldsymbol{x}$ are not measurable directly. To evaluate the likelihood $\mathcal{L}(\boldsymbol{x};\mathcal{Y})$, some ingredients are needed: a computational forward model $\mathcal{M}$, a set of input parameters $\boldsymbol{x} \in \mathcal{D}_{\boldsymbol{X}}$ that need to be inferred, and a set of experimental data $\mathcal{Y}$. The forward model $\boldsymbol{x} \rightarrow \boldsymbol{M}(\boldsymbol{x})$ is a mathematical representation of the system under consideration. All models are always simplifications of the real world. Thus, to connect model predictions to the observations $\mathcal{Y}$, a *discrepancy term* $\boldsymbol{\varepsilon}$ shall be introduced. We consider the following well-established format:

$$\boldsymbol{y} = \mathcal{M}(\boldsymbol{x}) + \boldsymbol{\varepsilon} \tag{2.2}$$

where $\boldsymbol{\varepsilon} \in \mathbb{R}^{N_{\text{out}}}$ is the term that describes the discrepancy between an experimental observation $\mathcal{Y}$ and the model prediction. For the sake of simplicity, we consider it as an additive *Gaussian discrepancy* with zero mean and a covariance matrix $\boldsymbol{\Sigma}$ in this introduction:

$$\boldsymbol{\varepsilon} \in \mathcal{N}(\varepsilon|\boldsymbol{0}, \boldsymbol{\Sigma}) \tag{2.3}$$

It is noted that simple Gaussian discrepancy assumption is only one out of many possible models. In a more general setting, other distributions for the discrepancy are used as well (Wagner et al., 2022). Due to the widespread used of the additive Gaussian models in engineering disciplines, the thesis is limited to Gaussian type. If N independent measurement $\boldsymbol{y_i}$ are available and gathered in the data set $\mathcal{Y} \overset{\text{def}}{=} \{\boldsymbol{y}^{(1)}, \cdots, \boldsymbol{y}^{(N)}\}$, the likelihood can thus be written as:

$$\begin{aligned} \mathcal{L}(\boldsymbol{x};\mathcal{Y}) &= \prod_{i=1}^{N} N(\boldsymbol{y_i}|\mathcal{M}(\boldsymbol{x}), \boldsymbol{\Sigma}) \\ &= \prod_{i=1}^{N} \frac{1}{\sqrt{(2\pi)^{N_{\text{out}}} \det(\boldsymbol{\Sigma})}} \exp\left(-\frac{1}{2}\left(\boldsymbol{y_i} - \mathcal{M}(\boldsymbol{x})\right)^{\mathsf{T}} \boldsymbol{\Sigma}^{-1} \left(\boldsymbol{y_i} - \mathcal{M}(\boldsymbol{x})\right)\right) \end{aligned} \tag{2.4}$$

- Evidence $\pi(\mathcal{Y})$: In Bayesian inference, $\pi(\mathcal{Y})$ is often seen as a normalizing factor that ensures that posterior PDF integrates to one:

$$\pi(\mathcal{Y}) \overset{\text{def}}{=} \int_{\mathcal{D}_{\boldsymbol{X}}} \mathcal{L}(\boldsymbol{x};\mathcal{Y})\pi(\boldsymbol{x})\mathrm{d}\boldsymbol{x} \tag{2.5}$$

A schematic Bayesian inference in two dimensional space is displayed in Figure 2.1. The plots show the various elements of the Bayesian inference procedure in the parameter and data spaces. In the parameter space, with new experimental data comes in, the posterior is more concentrated than the prior distribution.

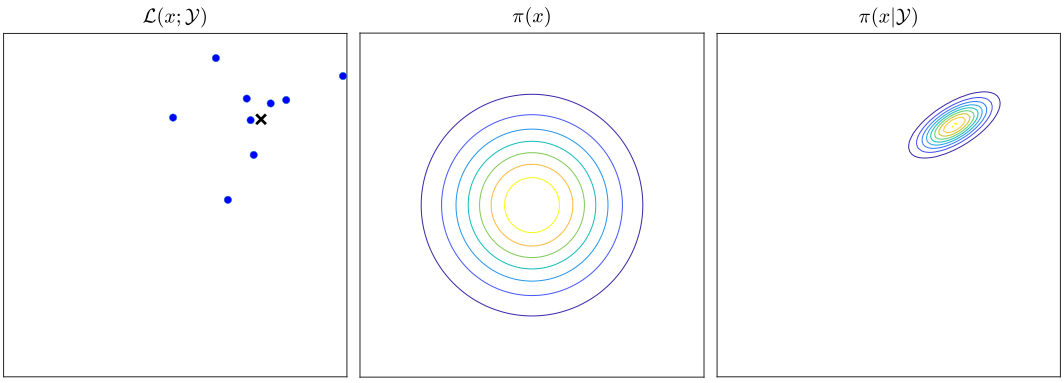


Figure 2.1: Bayesian inference in 2D space

2.2 Posterior quantities of interest

Under the Bayesian paradigm, the posterior distribution $\pi(\boldsymbol{x}|\mathcal{Y})$ is the solution of the inverse problem. However, in practice it can also serve as an intermediate result that is further processed for interpretation or prediction purpose. Furthermore, the full distribution can contain too much information to allow statements about the inferred parameters. Therefore, it is common to process the posterior and extract certain quantities of interest that summarize the inversion results more concisely.

In many applications, one is only interested in a single parameter, i.e., the one that characterise the inversion most suitably. The two most common *point estimation* methods are the *posterior mean* $\boldsymbol{x}^{\text{mean}}$ and Maximum a Posterior (MAP). The $\boldsymbol{x}^{\text{mean}}$ is given as:

$$x^{\text{mean}} = \mathbb{E}[\boldsymbol{X}|\mathcal{Y}] = \int_{\mathcal{D}_{\boldsymbol{X}|\mathcal{Y}}} \boldsymbol{x}\pi(\boldsymbol{x}|\mathcal{Y})\mathrm{d}\boldsymbol{x} \tag{2.6}$$

It reflects what we expect the parameter value to be after the inference. The MAP parameter, as the mode of the posterior distribution on the other hand, is the one maximises the posterior:

$$\begin{aligned} \boldsymbol{x}^{\mathrm{MAP}} &= \arg\max_{\boldsymbol{x}\in\mathcal{D}_{\boldsymbol{X}}} \pi(\boldsymbol{x}|\mathcal{Y}) \\ &= \arg\max_{\boldsymbol{x}\in\mathcal{D}_{\boldsymbol{X}}} \mathcal{L}(\boldsymbol{x};\mathcal{Y})\pi(\boldsymbol{x}) \end{aligned} \tag{2.7}$$

where the evidence constant $\pi(\mathcal{Y})$ was omitted. The MAP point corresponds to the most likely value of the input parameters. It is closely related to the Maximum Likelihood (ML) point that is defined as

$$\boldsymbol{x}^{\mathrm{ML}} = \arg\max_{\boldsymbol{x}\in\mathcal{D}_{\boldsymbol{X}}} \mathcal{L}(\boldsymbol{x};\mathcal{Y}) \tag{2.8}$$

for which the forward model $\mathcal{M}$ produces the best agreement with the available data. Unlike the ML, the MAP point considers the prior information imposed by the prior distribution. The difference is typically larger in the case of little data, where the regularisation effect of the prior distribution is stronger. In case of uniform priors, the two are equal, provided that the ML point does not lie outside the prior support.

Choices for the point estimation above disregards the estimation uncertainty in the parameters. Therefore, to more comprehensively characterise the posterior distribution and investigate the calibration, it is useful to compute the second *posterior moments* and the *covariance*. They are summarised in the *posterior covariance matrix* $\boldsymbol{C} \in \mathbb{R}^{M\times M}$ with entries:

$$\begin{aligned} \boldsymbol{C} &= \mathrm{Cov}[\boldsymbol{X}|\mathcal{Y}] \\ &= \int_{\mathcal{D}_{\boldsymbol{X}}} (\boldsymbol{x} - \mathbb{E}[\boldsymbol{X}|\mathcal{Y}])(\boldsymbol{x} - \mathbb{E}[\boldsymbol{X}|\mathcal{Y}])^{\mathsf{T}} \pi(\boldsymbol{x}|\mathcal{Y})\mathrm{d}\boldsymbol{x} \end{aligned} \tag{2.9}$$

As the full posterior distribution is an M-dimensional object, which is inherently difficult to comprehend, one is typically also interested in the *posterior marginals*. The distribution of the marginalised random variable $X_i|\mathcal{Y}$ is given by (see Equation (2.10)):

$$\pi(x_i|\mathcal{Y}) = \int_{\mathcal{D}_{\boldsymbol{X}_{\mathbf{v}}}} \pi(\boldsymbol{x}|\mathcal{Y})\mathrm{d}\boldsymbol{x}_{\mathbf{v}}, \text{with } \mathbf{v} = \{1,\cdots,M\}\backslash i \tag{2.10}$$

This univariate PDF can be integrated to obtain the corresponding Cumulative Den-

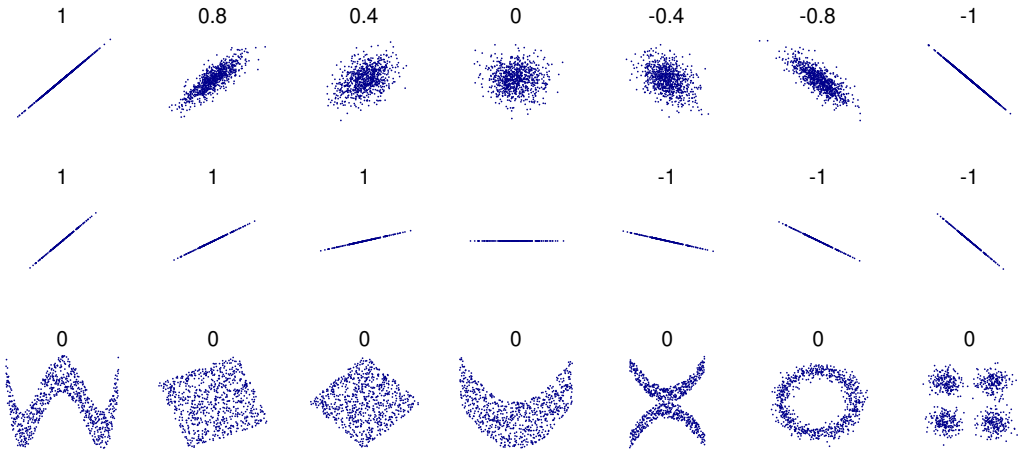


Figure 2.2: Example sets of (x, y) points with different COV. Source: *Wikipedia*

sity Function (CDF) which may then be used to define Confidence Interval (CI) on the calibrated parameters by means of quantiles.

To assess the predictive capabilities of a computational model, the Bayesian inference framework offers the possibility to compute *predictive distributions*. Using previously defined discrepancy model from Equation (2.3) and Equation (2.2), the *prior predictive* distribution can be written as:

$$\pi(\boldsymbol{y}) = \int_{\mathcal{D}_{\boldsymbol{X}}} \pi(\boldsymbol{x})\pi(\boldsymbol{y}|\boldsymbol{x})\mathrm{d}\boldsymbol{x} \tag{2.11}$$

It summarises the uncertainty about the model output considering also the discrepancy model before calibration. It should in practice be used to determine whether the measured data can be reproduced and consequently rule out severely ill-posed inverse problems that should be re-evaluated before proceeding with expensive calibration procedures. The *posterior predictive* distribution (see Equation (2.12)) can be similarly written as:

$$\pi(\boldsymbol{y}|\mathcal{Y}) = \int_{\mathcal{D}_{\boldsymbol{X}|\mathcal{Y}}} \pi(\boldsymbol{x}|\mathcal{Y})\pi(\boldsymbol{y}|\boldsymbol{x})\mathrm{d}\boldsymbol{x} \tag{2.12}$$

2.3 Computational methods for Bayesian inference

The practical computation of posterior distributions $\pi(\boldsymbol{x}|\mathcal{Y})$ is not trivial. Because computing evidence $\pi(\mathcal{Y})$ is usually not a tractable problem, analytical solutions thus

require more restrictions on the model. It can be only calculated analytically if it is given in a closed form. A common strategy we usually choose is a *conjugate prior* (Gelman et al., 1995) to the likelihood, so the integral can be represented analytically. For example, in a static Bayesian network, choices such as *Variant elimination* and *Belief propagation* (Murphy, 2012) can be seen. In the realm of a dynamic sequential model, *kalman filtering* (Nguyen & Nestorović, 2016) gives an closed form for the parameter identification. However, in the general cases, the solution to the posterior can be rarely analytical due to a complex model or high computation for the evidence $\pi(\mathcal{Y})$. Thus, we need to resort to approximation method.

There are usually two categories of approximation methods: *optimisation based approximation* and *Monte Carlo sampling* methods.

- *Optimisation based approximation*: This method usually refers to variational inference. The basic idea of optimized based approximation is to use an analytical form of a distribution to approximate the posterior based on some loss functions. Then we can see the similarity (e.g., *Kullback-Leibler divergence*) to measure in information contained within two distributions.

  The advantages of optimised approximation methods are: (1) Computationally efficient and work well on large models. (2) It has absolute converging criteria which makes easy to determine when to stop the modelling. (3) It scales better and are more amenable to parallelization. However, there are some problems itself: (1) Unlike *sampling-based methods*, variational approaches will almost never find the globally optimal solution. (2) their accuracy is often limited by the form of the approximation.

- *Monte Carlo sampling*: Sampling method is another way to approximate the posterior distribution. Examples include *inverse probability transform*, *rejection sampling*, *importance sampling*, Markov Chain Monte Carlo (MCMC) and Sequential Monte Carlo (SMC). These methods generate random samples from a *proposal distribution* and use them to estimate the posterior distribution and the derived statics.

  Monte Carlo sampling has the advantages that: (1) It is more straightforward and

flexible. (2) It is guaranteed to find the globally optimal solution given enough time and samples. However, In order to quickly reach a good solution, *Monte Carlo sampling* take more time and require choosing an appropriate sampling technique.

In short mentioned above, *Monte Carlo sampling* is asymptotically exact, accurately approximating the target distribution with increasing samples. In contrast, *Optimisation based approximation* lacks guarantees but yields faster results. Both *Optimisation based approximation* and *Monte Carlo sampling* are significant topics. Current studies on these are still very active with numerous techniques proposed in the past few years. More details can be found in Murphy (2012) and Blei et al. (2017).

It is worthy noted that it is not always the case *Monte Carlo sampling* is better than *Optimisation based approximation* or vice versa. The choice for approximate Bayesian method is totally problem-specified. Faster posterior approximation requires trading off additional accuracy. In this thesis, however, we expect to find the global optimal values and hope the solution with guarantee. Therefore, our thesis is only focused on *Monte Carlo sampling* as detailed in the next section.

2.4 Monte Carlo sampling

In this section, we will discuss the class of algorithms based on the idea of *Monte Carlo approximation*. The idea is straightforward: generate s samples from the t_{th} step posterior, $\boldsymbol{x}_t^s \sim \pi(\boldsymbol{x}_t|\mathcal{Y}_t)$, in which t represents the time in a dynamic model (state-space model) and $\mathcal{Y}_t$ denotes the observation at t_{th} step. Because most engineering projects are carried out in stages as shown in Figure 2.3, a recursive updating and sampling scheme is required across the multiple stages in the dynamic model. The introduction of subscript t aims to elucidate the sampling process at different stages. Subsequently, these samples are used to aggregate any quantity of interest $\mathbb{E}[f|\mathcal{Y}_t] \approx \frac{1}{S}\sum_{s=1}^{S} f(\boldsymbol{x}_t^s)$ given suitable function f. By generating enough samples, we can achieve any desired level of accuracy we like.

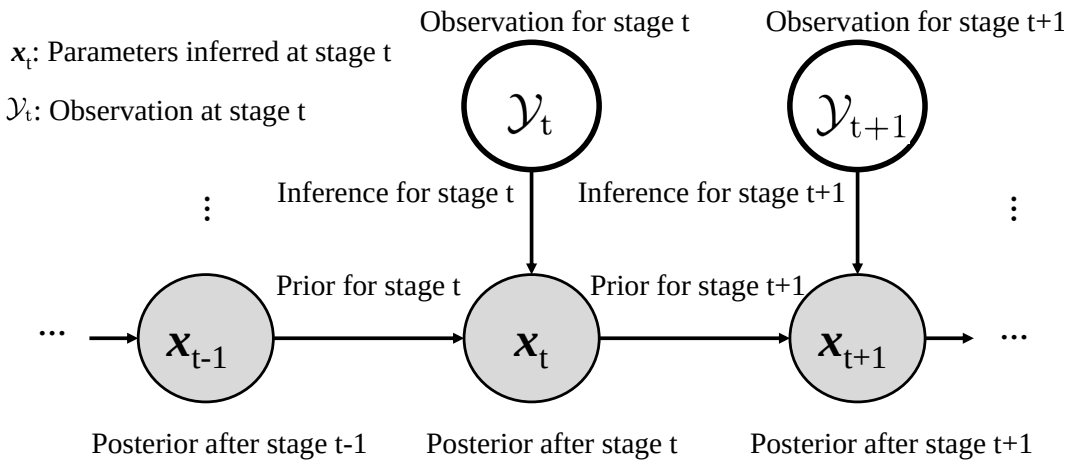


Figure 2.3: Sequential Bayesian updating

2.4.1 Inverse probability transform

The simplest method for sampling from a univariate distribution is based on the *inverse probability transform*. As shown in Figure 2.4, if we can get the Cumulative Density Function (CDF), then we can easily generate samples by computing $\boldsymbol{x}_t^s = \text{CDF}(\mathcal{U})$. $\mathcal{U}$ follows uniform distribution $\mathcal{U} \sim U(0,1)$ using a *pseudo random number generator*.

2.4.2 Rejection sampling

When the inverse CDF method cannot be used, one simple alternative is to use *rejection sampling*. In *rejection sampling*, we create a proposal distribution $q(\boldsymbol{x})$ which satisfies $Mq(\boldsymbol{x}) \geq \tilde{\pi}(\boldsymbol{x})$, for some constant M, where $\tilde{\pi}(\boldsymbol{x})$ is an unnormalized version of $\pi(\boldsymbol{x})$ (i.e., $\pi(\boldsymbol{x}) = \tilde{\pi}(\boldsymbol{x})/Z$ for some unknown constant Z). The function $Mq(\boldsymbol{x})$ provides an upper envelope for $\tilde{\pi}$. We then sample at time stage t: $\boldsymbol{x}_t^s \sim q(\boldsymbol{x}_t)$, which corresponds to picking a random $\boldsymbol{x}_t^s$ location, and then we sample $u \sim U(0,1)$ which corresponds to picking a random height (y location) under the envelope. If $u \geq \frac{\tilde{\pi}(\boldsymbol{x}_t^s)}{Mq(\boldsymbol{x}_t^s)}$, we reject the sample, otherwise we accept it. See Figure 2.5, where acceptance region is shown shaded, and the rejection region is the white region between the shaded zone and the upper envelope. But, large-dimensional spaces tend to be very empty, and the chances that this method accepts a point may be dramatically low when working with multidimensional spaces.

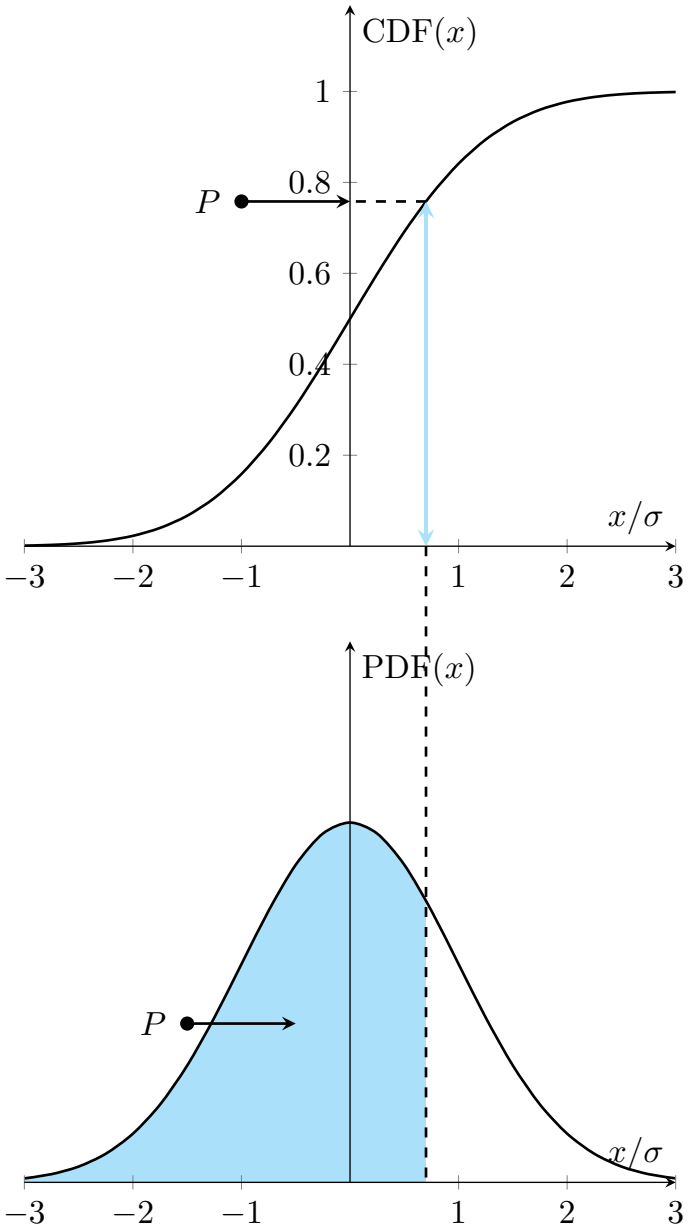


Figure 2.4: Sampling using an inverse CDF

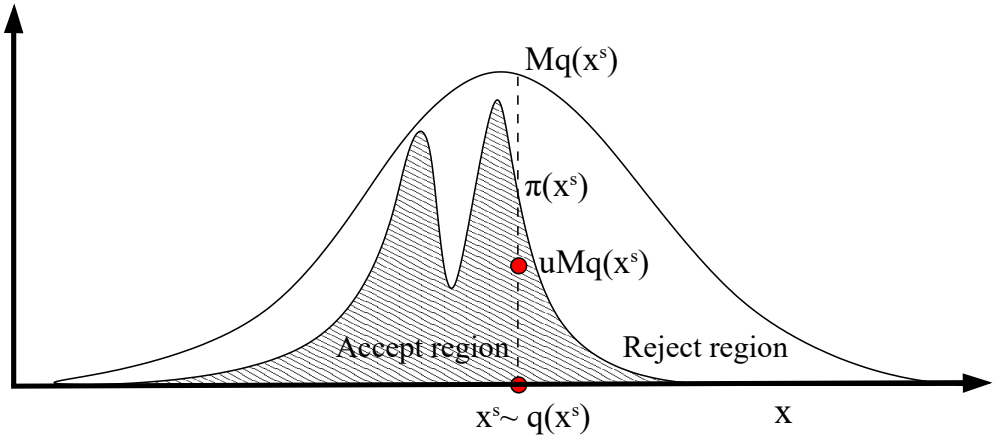


Figure 2.5: Schematic illustration of *rejection sampling* from Andrieu et al. (2003)

2.4.3 Importance sampling

We now describe a Monte Carlo method known as *importance sampling* for approximating integrals of the form:

$$E[f(\boldsymbol{x}_t)] = \int f(\boldsymbol{x}_t)\pi(\boldsymbol{x}_t)d\boldsymbol{x}_t \approx \frac{1}{S}\sum_{s=1}^{S} f(\boldsymbol{x}_t^s) \tag{2.13}$$

where $\boldsymbol{x}_t \sim \pi(\boldsymbol{x}_t)$ The idea of *Monte Carlo sampling* is to simply sample $\boldsymbol{x}_t$ from the distribution $\pi(\boldsymbol{x}_t)$. However, when it comes to tricky target $\pi(\boldsymbol{x}_t)$, the above estimation can be less efficient, meaning it requires huge amounts of samples to ensure the accuracy. Then it is better to sample from a proposal form $q(\boldsymbol{x}_t) \propto f(\boldsymbol{x}_t)\pi(\boldsymbol{x}_t)$ than to sample from $\pi(\boldsymbol{x}_t)$ itself.

The idea of *importance sampling* is: draw the t_{th} step samples from a proposal distribution $q(\boldsymbol{x}_t)$ and re-weight the integral using importance weights so that the correct distribution is targeted. It then uses these samples to estimate the integral as follows:

$$E[f(\boldsymbol{x}_t)] = \int f(\boldsymbol{x}_t)\pi(\boldsymbol{x}_t)d\boldsymbol{x}_t = \int f(\boldsymbol{x}_t)\frac{\pi(\boldsymbol{x}_t)}{q(\boldsymbol{x}_t)}q(\boldsymbol{x}_t)d\boldsymbol{x}_t \approx \frac{1}{S}\sum_{s=1}^{S} f(\boldsymbol{x}_t^s)\frac{\pi(\boldsymbol{x}_t^s)}{q(\boldsymbol{x}_t^s)} \tag{2.14}$$

where $\frac{\pi(\boldsymbol{x}_t^s)}{q(\boldsymbol{x}_t^s)}$ is the importance weight. With a proper proposal distribution $q(\boldsymbol{x}_t)$, the integral estimation can be close as possible to the true value (improving the approximation means decreasing the variance of estimation $\text{Var}(\boldsymbol{x}_t) = \mathbb{E}[\boldsymbol{x}_t{}^2] - \mathbb{E}[\boldsymbol{x}_t]^2$). However, in practice, the choice for the $q(\boldsymbol{x}_t^s)$ is not satisfying. This is a general difficult task, especially in high dimensions.

2.4.4 Markov Chain Monte Carlo

Closed-form expressions for the posterior distribution in Equation (2.1) cannot be derived in general. The exceptions are some linear models with conjugate distributions which were mentioned in section 2.3. Compared with non-iterative methods above, Markov Chain Monte Carlo (MCMC) methods are a class of algorithms that allow one to sample from probability distributions based on the construction of a Markov chain,

which asymptotically behaves to the PDF which is to be sampled.

MCMC methods were a breakthrough for solving Bayesian inverse problems and since then it has gained tremendous popularity in engineering areas. In a survey by *SIAM News*, MCMC was placed in the top 10 most important algorithms of the 20th century. Originally developed by statistical mechanics, it is used to construct *Markov chains* to sample a selected *target distribution*. In the context of Bayesian updating, this *target distribution* is the posterior in our analysis. Based on the *Markov chains*, every stage depends only on the last previously attained stage. We consider here one discrete *Markov chain* of the type $\mathcal{X}_t = \{\boldsymbol{x}_t^1, \cdots, \boldsymbol{x}_t^{N_\mathcal{X}}\}$, in which $N_\mathcal{X}$ is the number of chain iterations, $\mathcal{X}_t$ are the samples from sampling process and t is stage number. They start from an arbitrary state $\boldsymbol{x}_t^1 \in \mathcal{D}_{\boldsymbol{X}}$ that is updated sequentially based on a *transition kernel*. This kernel is to ensure that *Markov chains* fulfil *detailed balance* condition and are reversible, i.e., that the dynamics of the chains is not affected by the direction of iterative sampling. We omit an in-depth discussion of those properties and refer interest readers to exhaustive resources on the matter in Murphy (2012).

Nowadays, practitioners can choose from a vast of MCMC algorithms, ranging from the classical Metropolish-Hasting (MH), to advanced *Hamiltonian mechanics-inspired samplers*, *transitional MCMC algorithm* and *ensemble algorithms*. In the following, we will present two popular standard MCMC algorithms that are used in the context of Bayesian inference, but other alternative more effective sampling methods can be also easily implemented.

Metropolish-Hasting (MH)

The basic idea in MH is to define a proposal distribution $q(\boldsymbol{x}_t^{s+1}|\boldsymbol{x}_t^s), s = 1, \cdots, N_\mathcal{X}$, between two adjacent samples from the current sample point $\boldsymbol{x}_t^s$ to the new point $\boldsymbol{x}_t^{s+1}$. If we have the first sample $\boldsymbol{x}_t^1$, we can generate the later samples for any number $N_\mathcal{X} > 1$. These samples can be summarized to describe the distribution of inversed parameters $\boldsymbol{x}_t$. For any given distribution $\pi(\boldsymbol{x}_t)$, we define the acceptance function, which is used to

decide whether to accept this move, as follows:

$$f(\boldsymbol{x}_t^{s+1}|\boldsymbol{x}_t^s) = \min(1, \alpha) \tag{2.15}$$

$$\alpha = \min(1, \frac{q(\boldsymbol{x}_t^s|\boldsymbol{x}_t^{s+1})\pi(\boldsymbol{x}_t^{s+1}|\mathcal{Y}_t)}{q(\boldsymbol{x}_t^{s+1}|\boldsymbol{x}_t^s)\pi(\boldsymbol{x}_t^s|\mathcal{Y}_t)}) \tag{2.16}$$

In practice, in order to accept and reject the proposed candidates with probability in Equation (2.15) and Equation (2.16), one usually samples a random variate $u \in (0, 1)$ from the standard uniform distribution, i.e., $U \sim \mathcal{U}(0, 1)$, and compares it to the probability α. If $\alpha > 1$, which implies $q(\boldsymbol{x}_t^s|\boldsymbol{x}_t^{s+1})\pi(\boldsymbol{x}_t^{s+1}|\mathcal{Y}_t) > q(\boldsymbol{x}_t^{s+1}|\boldsymbol{x}_t^s)\pi(\boldsymbol{x}_t^s|\mathcal{Y}_t)$, we accept the candidate sampled point $\boldsymbol{x}_t^{(\star)}$ from $q(\boldsymbol{x}_t^s|\boldsymbol{x}_t^{s+1})$. If $\alpha < 1$, we accept the candidate point $\boldsymbol{x}_t^{(\star)}$ with probability α. If the candidate is accepted, the new sample point $\boldsymbol{x}_t^{(\star)}$ is $\boldsymbol{x}_t^{s+1}$, otherwise stays the same as $\boldsymbol{x}_t^s$. A commonly used proposal distribution is Gaussian distribution centered on the current point $\boldsymbol{x}_t^s$. The acceptance function can be modified as:

$$f(\boldsymbol{x}_t^{s+1}|\boldsymbol{x}_t^s) = \min(1, \alpha) = \min(1, \frac{q(\boldsymbol{x}_t^{s+1})}{q(\boldsymbol{x}_t^s)}) \tag{2.17}$$

The process of MH is shown in Algorithm 1. It is worth mentioning that although the

Algorithm 1: Metropolish-Hasting algorithm at t_{th} step

Data: $q(\boldsymbol{x}_t)$: Proposal distribution; $\pi(\boldsymbol{x}_t|\mathcal{Y})$: Target posterior.
Result: MCMC samples at t_{th} stage: $\mathcal{X}_t = \{\boldsymbol{x}_t^1, \cdots, \boldsymbol{x}_t^{N_\mathcal{X}}\}$

```
Initialization x_t^1 ∈ D_X;
for s ← 2 to N_X do
    Sample x_t^{s+1} ~ q(x_t^{s+1}|x_t^s);
    Compute acceptance probability α;
    Compute f(x_t^{s+1}|x_t^s) = min(1, α);
    Sample u ~ U(0, 1);
    Set candidate sample x_t^(⋆) to x_t^{s+1} with probability α;
end for
```

proposal distribution $q(\boldsymbol{x}_t)$ can be any distribution, the closer it is to the actual target distribution $\pi(\boldsymbol{x}_t|\mathcal{Y})$, the more efficient the chain mixture would be. Also, we need to assign an initial position $\boldsymbol{x}_t^1 \in \mathcal{D}_{\boldsymbol{X}}$ that is not zero probability. Although the initial position does not affect the convergence of the sampling, a good initial guess would help to accelerate mixing for a Markov chain. With more paralleled N_{chain} chains and iterative

steps used ($\mathcal{X}_t = \{\mathcal{X}_{t\_1}, \cdots, \mathcal{X}_{t\_N_{chain}}\}$), where each chain $\mathcal{X}_{t\_i}$ contains $\{\boldsymbol{x}_{t\_i}^1, \cdots, \boldsymbol{x}_{t\_i}^{N_\mathcal{X}}\}$, the convergence will be improved accordingly. One schematic figure of the MH algorithm for sampling from a mixture of two 1D Gaussian can be seen in Figure 2.6.

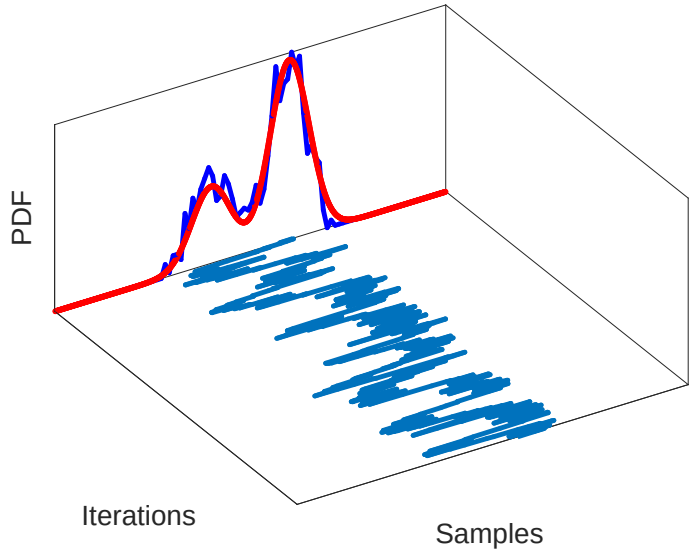


Figure 2.6: An schematic figure of the MH algorithm for sampling from a mixture 1D Gaussian at t_{th} stage

Affine invariant ensemble sampler (AIES)

Many MCMC algorithms perfom poorly when the target (i.e., posterior) distribution $\pi(\boldsymbol{x}_t|\mathcal{Y})$ exhibits strong, priorly unknown, inter-parameter correlation. For the MH, for example, it is the challenging to construct a proposal distribution that propose states in the high-probability regions of the parameter space. This results in low acceptance rates which can only be improved by considerable tuning efforts. The AIES originally presented in (Goodman & Weare, 2010) alleviates this problem. Its invariance to affine transformations means that if there exists an affine transformation of the *difficult-to-sample* (by standard MCMC methods) target distbution to an *easier-to-sample* target distribution, AIES samples both distributions equally well without explicitly requiring this affine transformation.

The AIES algorithm presented in Algorithm 2, simultaneously generates an *ensemble* of N_{chain} *Markov chains* at the t_{th} stage ($\mathcal{X}_t = \{\mathcal{X}_{t\_1}, \cdots, \mathcal{X}_{t\_N_{chain}}\}$). Each chain is called a *walker*. At the $\{s_{th}\}_{s\in\{1,\cdots,N_\mathcal{X}\}}$ iteration, all *Markov chain states* $\{\boldsymbol{x}_{t\_i}^s\}_{i\in\{1,\cdots,N_{chain}\}}$ are updated *walker by walker*. To update the i_{th} *walker*, the algorithm randomly picks a

conjugate walker from $j \in \{1, \cdots, N_{chain}\} \backslash i$, i.e., excluding the current i_{th} walker. The affine invariance property is then achieved by generating proposals according to a so-called *stretch move*. This refers to proposing a new candidate along a straight line between the current walker and the *conjugate walker* with

$$\boldsymbol{x}_t^{(\star)} = \boldsymbol{x}_{t\_i}^{(s)} + z \cdot (\boldsymbol{x}_{t\_j}^{(\tilde{s})} - \boldsymbol{x}_{t\_i}^{(s)}) \tag{2.18}$$

where $\tilde{s} = s + 1$ if $j < i$ and $tilde(t) = s$ otherwise, i.e., it denotes tha latest available stage of the j_{th} *walker*. The stretch factor z is randomly drawn from the PDF

$$p(z|a) = \begin{cases} \frac{1}{\sqrt{z}(2\sqrt{a} - \frac{2}{\sqrt{a}})} & \text{if } z \in [1/a, a] \\ 0 & \text{otherwise} \end{cases} \tag{2.19}$$

which depends on the tuning parameter $a > 1$. The candidate sample $x_t^{(\star)} \in \mathcal{D}_{\boldsymbol{X}} \subseteq \mathbb{R}^M$ is then accepted as the new location of the i_{th} walker with probability:

$$\alpha = \min(1, z^{M-1} \frac{\pi(x_t^{(\star)}|\mathcal{Y})}{\pi(x_{t\_i}^{(s)}|\mathcal{Y})}) \tag{2.20}$$

This is repeated for all N_{chain} walkers in the ensemble. The resulting chains fulfil the *detailed balanced condition*. A practical advantage of the AIES algorithm is that it only has a single scalar tuning parameter a, which is often set to $a = 2$ (Wagner et al., 2022).

Algorithm 2: Affine invariant ensemble sampler algorithm at t_{th} step

Data: $\pi(\boldsymbol{x}_t|\mathcal{Y})$: Target posterior; tuning parameter a
Result: MCMC samples at t_{th} stage: $\mathcal{X}_t = \{\mathcal{X}_{t\_1}, \cdots, \mathcal{X}_{t\_N_{chain}}\})$, with $\mathcal{X}_{t\_i} = \{\boldsymbol{x}_{t\_i}^1, \cdots, \boldsymbol{x}_{t\_i}^{N_{\mathcal{X}}}\}$

```
Initialization N_chain samples {x^1_{t_1}, ..., x^1_{t_N_chain}}, with x^1_{t_i} ∈ D_X
for s ← 2 to N_X do
    for i ∈ {1, ..., N_chain} do
        Pick random j from {1, ..., N_chain}\i;
        Propose x_t^(⋆) with Equation (2.18);
        Set x^s_{t_i} = x_t^(⋆) with probability α (see Equation (2.20));
    end for
end for
```

2.4.5 Sequential Monte Carlo

Like MCMC, Sequential Monte Carlo (SMC), is another popular sampling method for a dynamic model. The parameters $\boldsymbol{x_t}$ which we are interested can be linked with the observations $\mathcal{Y}_t$ in a time series way:

$$\begin{aligned} \boldsymbol{x_t} &= g(\boldsymbol{x_{t-1}}) + \boldsymbol{v} \qquad \text{(state equation)} \\ \mathcal{Y}_t &= m(\boldsymbol{x_t}) + \boldsymbol{w} \qquad \text{(observation equation)} \end{aligned} \tag{2.21}$$

which stands for the prediction step and correction step, respectively. $g(\cdot)$ and $m(\cdot)$ denote *transition equation* and *emission equation*, respectively. They can be either linear or non-linear functions. $\boldsymbol{v}$ and $\boldsymbol{w}$ are independent random variables (or vectors) that represent process noise and observation noise, respectively.

In this setting, we only focus on nonlinear non-Gaussian state-space model which is also called *particle filter*. The feature of the non-linearity is used to relax the constraints at some simplified models (i.e., Kalman filter (KF) or Unscented Kalman filter (UKF), etc.). SMC is a sequential Bayesian technique by generating a population of independent particles (each particle corresponding to a possible parameter set). It will filter these particles in a way that the final distribution of particles approximates the posterior distribution $\pi(x_t|\mathcal{Y})$. This filtering typically consists of several steps: initial samples, reweighting and resampling, which are sequentially applied while new observations $\mathcal{Y}_t$ are added. There are different flavours of *particle filter*. For book-length treatment, see Murphy (2012). Due to space limitation, we will only present two standard SMC algorithms that are used in the context of Bayesian inference, but other effective *particle filter* methods can be also easily implemented.

Sequential importance sampling

The basic idea is to approximate the belief state (of the entire stage trajectory) using a weighted set of particles:

$$\pi(\boldsymbol{x}_{1:t}|\mathcal{Y}_{1:t}) \approx \sum_{s=1}^{S} \tilde{w}_t^s \delta_{\boldsymbol{x}_{1:t}^s}(\boldsymbol{x}_{1:t}) \tag{2.22}$$

$$\tilde{w}_t^s = \frac{w_t^s}{\sum_{s=1}^{S}(w_t^s)} \tag{2.23}$$

where $\tilde{w}_t^s$ is the normalised weight of sample $s = \{1, \cdots, N\}$ at time t and δ is *Dirac delta function*. From this representation, we can easily compute the marginal distribution over the most recent state,$\pi(\boldsymbol{x}_t|\mathcal{Y}_{1:t})$, by simply ignoring the previous parts of the trajectory $\boldsymbol{x}_{1:t}$.

We update this belief state using importance sampling. If the proposal has the form $q(\boldsymbol{x}_{1:t}|\mathcal{Y}_{1:t})$, we can rewrite the numerator recursively as follows:

$$\pi(\boldsymbol{x}_{1:t}|\mathcal{Y}_{1:t}) \propto \pi(\mathcal{Y}_t|\boldsymbol{x}_t)\pi(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})\pi(\boldsymbol{x}_{t-1}|\mathcal{Y}_{t-1}) \tag{2.24}$$

where we have made the usual Markov assumptions. We will restrict attention to proposal densities of the following form:

$$\pi(\boldsymbol{x}_{1:t}|\mathcal{Y}_{1:t}) = \pi(\boldsymbol{x}_t|\boldsymbol{x}_{1:t-1}, \mathcal{Y}_{1:t})\pi(\boldsymbol{x}_{1:t-1}|\mathcal{Y}_{1:t-1}) \tag{2.25}$$

Then adding new state $\boldsymbol{x}_t$ to the end and assume we only need to keep the most recent part of the trajectory and observation sequence, rather than the whole history $1:t$ for simplicity. In this case, the weight becomes:

$$w_t^s \propto w_{t-1}^s \frac{\pi(\mathcal{Y}_t|\boldsymbol{x}_t^s)\pi(\boldsymbol{x}_t^s|\boldsymbol{x}_{t-1}^s)}{q(\boldsymbol{x}_t^s|\boldsymbol{x}_{t-1}^s, \mathcal{Y}_t)} \tag{2.26}$$

Hence we can approximate the posterior $\pi(\boldsymbol{x}_t|\mathcal{Y}_{1:t})$ filtered density using Equation (2.27)

$$\pi(\boldsymbol{x}_t|\mathcal{Y}_{1:t}) \approx \sum_{s=1}^{S} \tilde{w}_t^s \delta_{\boldsymbol{x}_t^s}(\boldsymbol{x}_t) \tag{2.27}$$

As $S \to \infty$, one can show that this approaches the true posterior. All initial particles have the same values of weight and can be sampled uniformly using *Latin hypercube sampling* in feasible ranges of parameter values. The whole SIS without resampling process is shown in Figure 2.7 and SIS algorithm is presented in Algorithm 3.

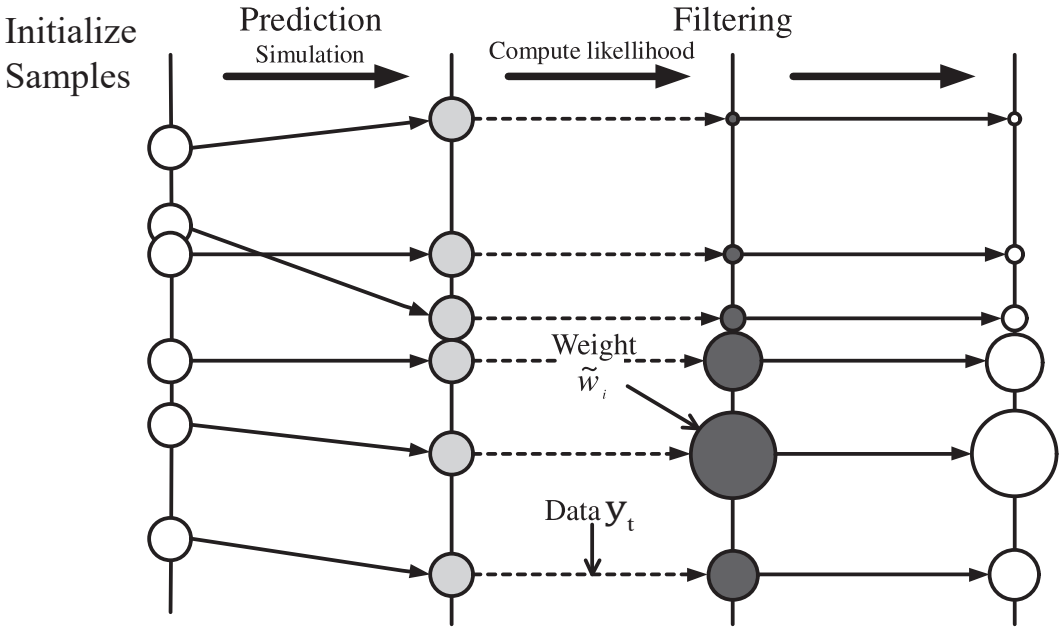


Figure 2.7: Particle filtering using SIS adapted from Nguyen & Nestorović (2016)

Algorithm 3: Sequential importance sampling algorithm at t_{th} step

Data: Samples $\boldsymbol{x}_{t-1}^s$ with weights w_{t-1}^s, $s = \{1, \cdots, N\}$; observation $\mathcal{Y}_t$ at t_{th} stage
Result: SMC samples with normalized weights $\tilde{w}_t^s$ at t_{th} stage:
$\boldsymbol{x}_t^{(\star)} = \mathcal{X}_t = \{\boldsymbol{x}_t^1, \cdots, \boldsymbol{x}_t^N\}$)

```
for s ← 1 to N do
    Sample from proposal distribution x_t^s ~ q(x_t^s | x_{t-1}^s, Y_t);
    Compute weight using Equation (2.26);
end for
Normalized weights;
```

Sequential importance sampling and resampling

The basic SIS algorithm fails after a few steps because most of the particles will have negligible weight. This is called the *degeneracy problem*, and occurs especially when

we are sampling in a high-dimensional space. We can quantify the degree of degeneracy using the *effective sample size*, defined by:

$$\hat{S}_{eff} = \frac{1}{\sum_{s=1}^{S}(w_t^s)^2} \tag{2.28}$$

If the variance of the weights is large, then we are wasting our resources updating particles with low weight, which do not contribute much to our posterior estimate. One effective way to the degeneracy problem is adding a resampling step. This improvement to the basic SIS algorithm can monitor the *effective sample size*, and whenever it drops below a threshold S_{min}, to eliminate particles with low weight, nad the nto create replicates of the surviving particles. This scheme is illustrated in Figure 2.8 and the overall SISR algorithm is summarized in Algorithm 4. In this case, $\boldsymbol{x}_t^{(\star)} \neq \mathcal{X}_t$ comes from fact that the resampling process discards low weighted particles.

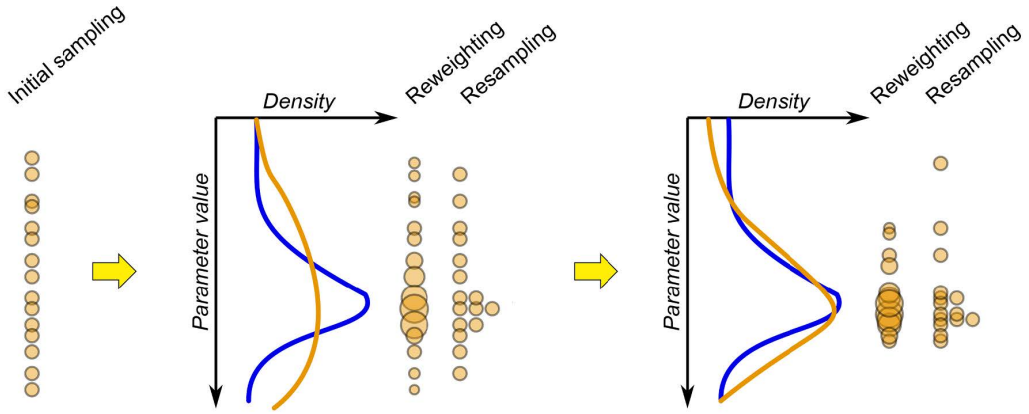
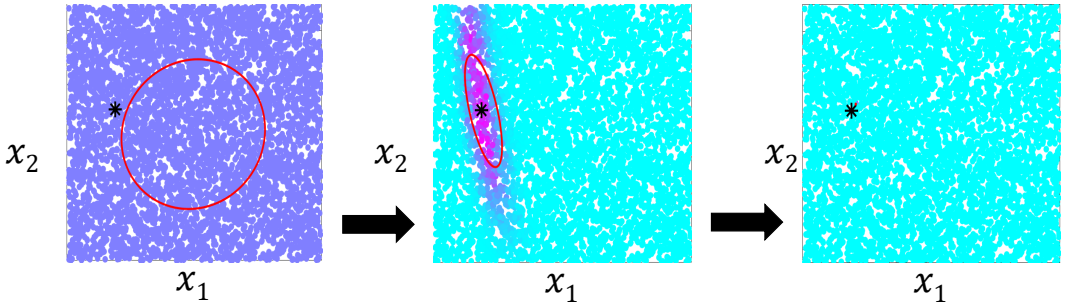


Figure 2.8: Principle of SISR algorithm adapted from Speich et al. (2021)

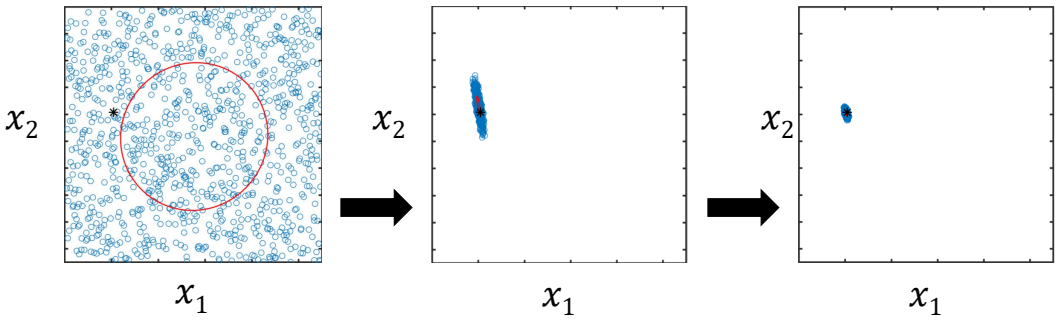
Algorithm 4: Sequential importance sampling and resampling algorithm at t_{th} step

Data: Samples $\boldsymbol{x}_{t-1}^s$ with weights w_{t-1}^s, $s = \{1, \cdots, N\}$; observation $\mathcal{Y}_t$ at t_{th} stage
Result: SMC samples with normalized weights $\tilde{w}_t^s$ at t_{th} stage:
$\boldsymbol{x}_t^{(\star)} = \{\boldsymbol{x}_t^1, \cdots, \boldsymbol{x}_t^N\}$)

```
for s ← 1 to N do
    Sample from proposal distribution x_t^s ~ q(x_t^s | x_{t-1}^s, Y_t);
    Compute weight using Equation (2.26);
end for
Normalized weights;
Calculate degeneracy measure using Equation (2.28);
if Ŝ_eff < S then
    Resample;
end if
```



(a) Particle filter with Sequential importance sampling



(b) Markov Chain Monte Carlo using Affine invariant ensemble sampler

Figure 2.9: SMC and MCMC in a two-dimensional example

2.5 Choices for sampling methods

In a continuous Bayesian calibration process, MCMC and SMC are two prevalent sampling methods. They are differed based on different sampling principles. MCMC-derived uncertain variables are iteratively constructed based on *random walk* and outperforms in high dimensions, while SMC-derived uncertain variables are represented by updating weights on the particles and yields faster inference results. As shown in Figures 2.9a and 2.9b, both methods demonstrates their effectiveness in updating parameters in a two dimensional space example.

SMC, with its non-iterative nature, offered the advantage of parallel processing, significantly enhancing computational efficiency. This attribute makes SMC a favorable option when quick results are imperative. However, *particle filter* doesnot work well in high dimensional spaces (Murphy, 2012). Instead, MCMC, with its random walk approach, showcased its proficiency in exploring high-dimensional parameter spaces, making it a suitable choice for complex and multifaceted problems. Therefore, the choice between these methods should be guided by the specific needs and priorities of the analysis,

whether precision or computational efficiency takes precedence. Since most engineering are naturally in high dimensional space, this thesis adopts MCMC during the sequential Bayesian calibration for the next sampling analysis.

Chapter 3

Uncertainty quantification in parameter identification

Uncertainty quantification (UQ) aims at taking into account the uncertainties in the parameters of the model of a physical system and at studying their impact onto the system response. A well-established representation of an uncertainty quantification problem is presented next. The rationale is that any such problem can be represented by a combination of these elements, as sketched in Figure 3.1:

- Definition of the computational model $\mathcal{M}$ of the physical system. This is rather broad step that may refer to an analytical function in its simplest form, or an black box containing different levels of partial differential equations (e.g., finite element packages or finite difference packages). In general, the computational model $\mathcal{M}$ maps a set of input parameters $\boldsymbol{x}$) to one or more Quantities of interest (QoI). often referred to as *model responses*.

- The sources of uncertainties in input space. This steps entails the identification of the input parameters that are uncertain and their description within a probabilistic context.

- Uncertainty propagation from input parameters $\boldsymbol{x}$ to $\boldsymbol{y}$. This step refers to the quantification of the uncertainty in the QoI by propagating the uncertainty of the

input space through the computational model $\mathcal{M}$.

- Iterative updating of the source of uncertainty. This step may refer to several techniques used to updated the information available on the sources of uncertainty identified above. Examples includes *sensitivity analysis* or *Bayesian inference*. If we are using *sensitivity analysis* to update the uncertainties, it can be explicitly called as *forward problems*. If we are using *Bayesian inference* to update the uncertainties, it can be called as *inverse problems*.

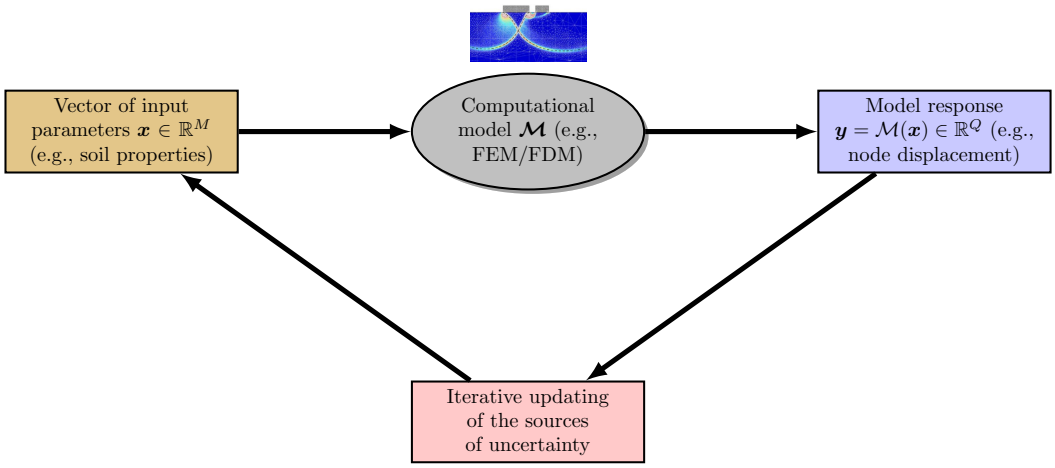


Figure 3.1: Global framework for uncertainty quantification

3.1 Problem statement in UQ

Modern engineering applications simulate UQ depending on a large number of input parameters. And the computational model is often treated as black box, i.e., only the input parameters $\boldsymbol{x}$ and model response $\boldsymbol{y}$ are available. Dealing with uncertainties using *Monte Carlo* in such systems poses a major challenge to computational resources. In such cases, the underlying model can be substituted by a surrogate $\tilde{\mathcal{M}}$ as shown in Figure 3.2. The solid line denotes the current working flow. Surrogate models enable many forward and inverse UQ analyses on high-fidelity computational models, by approximating the original model with a cheap-to-evaluate replacement model. In high dimensions, however, the performance of surrogate models decreases, while the cost of computing and storing them increases. This is well-known issue as the *curse of dimensionality*. Therefore, there is never enough data for constructing a perfect surrogate in high dimensions. To alleviate this, we can perform *experiment of design* sequentially using *active learning* techniques

or choose an appropriate type of surrogates. Next, we will introduce these in the next sections.

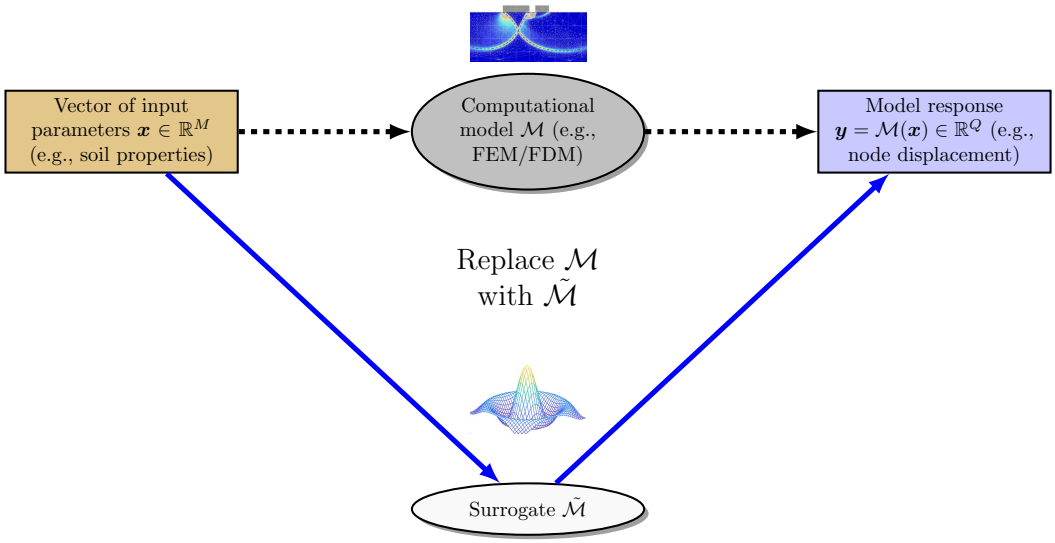


Figure 3.2: Using a surrogate to obtain the model response

3.2 Surrogate model

Computer modelling is used in nearly every field of science and engineering. Often, these computer codes model complex phenomena, have many input parameters, and are expensive to evaluate. In order to explore the behavior of the model under uncertainty (e.g., uncertainty propagation, parameter calibration from data or sensitivity analysis), many model runs are required. However, if the model is costly, only a few model evaluations can be afforded, which often do not suffice for thorough uncertainty quantification. In engineering and applied sciences, a popular work-around in this situation is to construct a reduced-order surrogate model. A reduced-order surrogate model is a cheap-to-evaluate proxy of the original model, which typically can be constructed from a relatively small number of model evaluations and approximates the input-output relation of the original model well. Since the surrogate model is cheap to evaluate, uncertainty quantification can be performed at a low cost by using the surrogate model instead of the original model. Therefore, surrogate modelling aims at constructing a metamodel that provides an accurate approximation to the original model while requiring as few model evaluations

as possible for its construction. A surrogate model $\tilde{\mathcal{M}}$ can be expressed as:

$$\tilde{\mathcal{M}}(\boldsymbol{X}) \stackrel{\text{def}}{=} \mathcal{M}(\boldsymbol{X}) - \mathcal{R}(\boldsymbol{X}) \tag{3.1}$$

$$\tilde{\mathcal{M}}(\boldsymbol{X}) \stackrel{\text{def}}{=} \mathcal{M}(\boldsymbol{X}) - \mathcal{R}(\boldsymbol{X}) \tag{3.2}$$

where $\mathcal{R}$ is the residual between the original model and the surrogate.

The general concept of Equation (3.2) is straightforward: given a finite set of input realisations and their corresponding model output, known as *experiment of design*. A variety of methods is available in the surrogate modelling, in which some of them are listed in the table 3.1.

Table 3.1: Surrogate model choices

Name	Shape	Parameters
Polynomial chaos expansions	$\tilde{\mathcal{M}}(\boldsymbol{x}) = \sum_{\boldsymbol{\alpha}\in\mathcal{A}} c_\alpha \Psi_\alpha(\boldsymbol{x})$	c_α
Low-rank tensor approximations	$\tilde{\mathcal{M}}(\boldsymbol{x}) = \sum_{l=1}^{R} b_l \left(\prod_{i=1}^{M} v_l^i x_i\right)$	$b_l,\ z_{k,l}^i$
Kriging (a.k.a Gaussian processs)	$\tilde{\mathcal{M}}(\boldsymbol{x}) = \boldsymbol{\beta}^T \cdot \boldsymbol{f}(\boldsymbol{x}) + Z(\boldsymbol{x}, \omega)$	$\boldsymbol{\beta},\ \sigma_Z^2,\ \boldsymbol{\theta}$
Support vector machines	$\tilde{\mathcal{M}}(\boldsymbol{x}) = \sum_{i=1}^{m} a_i K(\boldsymbol{x}_i, \boldsymbol{x}) + b$	$\boldsymbol{a},\ b$
Neural networks	$\tilde{\mathcal{M}}(\boldsymbol{x}) = f_n\left(\cdots f_2(b_2 + f_1(b_1 + \boldsymbol{w}_1 \cdot \boldsymbol{x}) \cdot \boldsymbol{w}_2)\right)$	$\boldsymbol{w},\ \boldsymbol{b}$

Why not neural networks?

Model reduction lets us create approximate models that are fast to solve, and — importantly — it provides us a rigorous mathematical basis on which to establish strong guarantees of accuracy of the low-dimensional model. This is in contrast with black-box machine learning methods (ANN), where we just have to hope that our training data was rich enough to yield a sufficiently accurate surrogate model. This is especially problematic for engineering applications where we often need to issue extrapolatory predictions.

3.2.1 Polynomial chaos expansion

Metamodelling (or surrogate modelling) attempts to offset the increased costs of stochastic modelling by substituting the expensive-to-evaluate computational models (e.g. finite element models, FEM) with inexpensive-to-evaluate surrogates. Polynomial chaos expansions (PCE) are a powerful metamodelling technique that aims at providing a func-

tional approxmation of a computational model through its spectral representation on a suitably built basis of polynomial functions.

Different from other surrogate modelling approaches such as support vector machine, Gaussian process regression, or neural networks, the mathematical theory underlying reduced-order models can lead to more reliable and robust predictive capability (Frangos et al., 2010; Kapteyn et al., 2021).

Different from Monte Carlo simulation (MCS) which is based on point-to-point exploring the output space, PCE assumes a generic structure, which better exploits the available runs of the FE realizations.

3.3 Data-driven inference

3.3.1 Inference of the marginal distributions

3.3.2 Inference of the copula

3.4 Dimensionality reduction

Engineering problems inherently involve high dimensionality, posing challenges for learning methods like surrogate modeling. Technical constraints impact the storage and processing of such huge amount of data. Furthermore, as input and output data expand, independent scalar surrogate models show inadequate in accurately capturing the covariance matrix of the original data, leading to less reliable predictions. Consequently, in high dimensional space, the need for dimensionality reduction technique (DR) becomes more critical.

3.4.1 Linear DR technique

3.4.2 Nonlinear DR technique

3.5 DR-based surrogate modelling

3.6 Inverse problems in UQ

When faced with large-scale forward models characteristic of many engineering and science applications, high computational cost arises from: (1) In the large-scale setting, performing thousands or millions of forward simulations is often computationally intractable; (2) the dimension of the input space are complex; (3) sampling may be complicated by the large dimensionality of the input space. Thus, to reduce the computational cost of solving of a statistical inverse problem, methods can be broadly in three groups: (1) Surrogate models to accelerate a forward simulation; (2) Reduce the dimension of the input space,i.e., sensitive analysis; (3) Efficient sampling method to posterior,i.e., MCMC.

3.6.1 Inverse problems

3.6.2 Bayesian inference

3.6.3 Bayesian calibration

3.6.4 Sampling methods

Using the CDF

Rejection sampling

Importance sampling

Often, in practical Bayesian models, it is not possible to obtain samples directly from p.x j y1WT / due to its complicated functional form.

Sequential Monte Carlo

Markov chain Monte Carlo

3.6.5 Choice of sampling method

3.6.6 Seqential Bayesian inference

3.7 Sequential enrichment for surrogate model

Traditional large-scale physics-based models are intractable to solve real-time, many-query context problem.

Instead of sampling the whole experimental design at once, it has been proposed to use sequential enrichment. Starting with a small experimental design, additional points are chosen based on the last computed sparse solution. In the context of machine learning, sequential sampling is also known as active learning. In all cases, numerical examples show that the sequential strategy generally leads to solutions with a smaller validation error compared to non-sequential strategies

3.8 Sensitivity analysis

3.9 Uncertainty quantification in high dimensions

3.9.1 Forward problem

Some Monte Carlo methods, including rejection sampling, importance sampling and particle filtering. The trouble with these methods is that they do not work well in high dimensional spaces. The most popular method for sampling from high-dimensional distributions is Markov chain Monte Carlo or MCMC.

3.9.2 High dimensional problem

Some Monte Carlo methods, including rejection sampling, importance sampling and particle filtering. The trouble with these methods is that they do not work well in

high dimensional spaces. The most popular method for sampling from high-dimensional distributions is Markov chain Monte Carlo or MCMC.

3.9.3 Choice for sampling method in high dimensions

Chapter 4

Predictive digital twins at scale for piles

This chapter develops a mathematical and computational foundation for digital twins of piles.

While the value proposition of digital twins has become widely appreciated, the technology itself remains in a custom production phase.

4.1 State space model

4.2 Probabilistic graphical model: Control theory

4.3 Partially observable Markov decision process

As shown in Figure 4.1, based on Markov Chain, with introducing Rewards and Actions, it can form the basis of Partially observed Markov decision process.

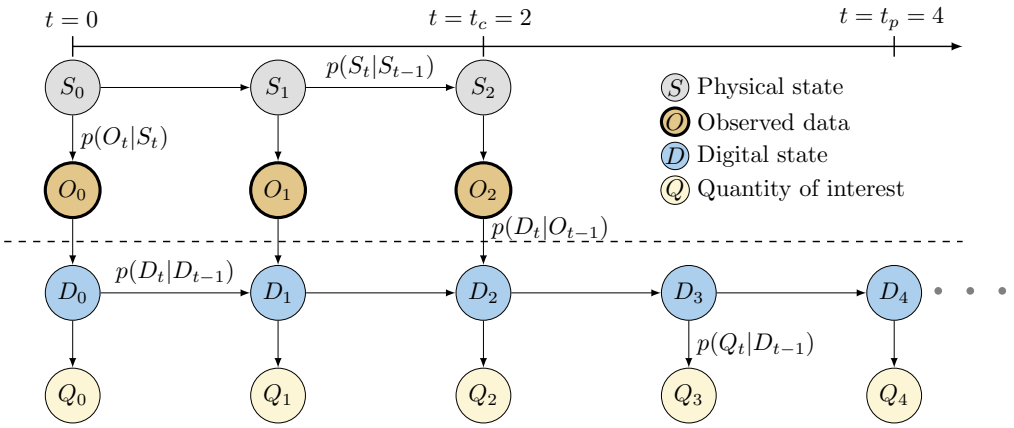


Figure 4.1: Digital twin

Generally, Digital Twin can be divided into two main parts, including (1) calibration and assimilation (2) Prediction, as shown in Equation (4.1) and Equation (4.2).

$$\begin{aligned} &p(D_0, ..., D_{t_c}, Q_0, ..., Q_{t_c}, R_0, ..., R_{t_c} | o_0, ..., o_{t_c}, u_0, ..., u_{t_c}) \\ &= \prod_{t=0}^{t_c} [\phi_t^{update} \phi_t^{QoI} \phi_t^{evaluation}] \end{aligned} \tag{4.1}$$

$$\begin{aligned} &p(D_0, ..., D_{t_p}, Q_0, ..., Q_{t_p}, R_0, ..., R_{t_p}, U_{t_c+1}, ..., U_{t_p} | o_0, ..., o_{t_c}, u_0, ..., u_{t_c}) \\ &\propto \prod_{t=0}^{t_p} [\phi_t^{dynamics} \phi_t^{QoI} \phi_t^{evaluation}] \prod_{t=0}^{t_c} \phi_t^{assimilation} \prod_{t=t_c+1}^{t_p} \phi_t^{control} \end{aligned} \tag{4.2}$$

4.4 Computational model-ICFEP

4.5 Planning and prediction via digital twin

Chapter 5

Work Plan

5.1 Stage 1

Calibrate the models in Figure 5.1 with partially observed Markov decision process.

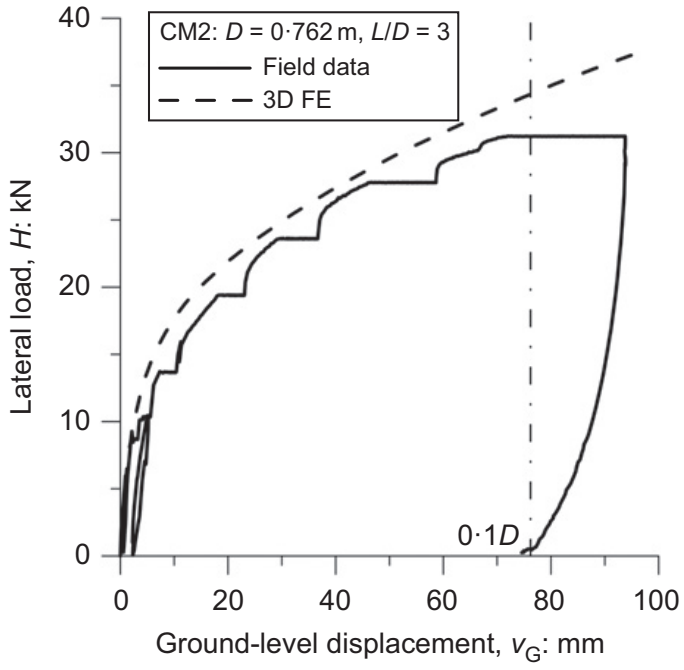


Figure 5.1: CM2 pile load displacement from Zdravković et al. (2020)

- Software: ICFEP-Likelihoods and observed data.
- Constitutive model: clay in Zdravković et al. (2020) and sand in Taborda et al. (2020).

- Consider the soil profile variance-Create the random field (scale of fluctuation in ICFEP).

Objective: Ensure the soil parameters in digital model can reveal unique characteristics of piles.

5.2 Stage 2

In operational Phase, Based on Partially observed Markov decision process method, continue the assimilation process: extend the digital twin capability to capture the piles response during loading.

5.3 Stage 3

Extension to Prediction

As shown in table 5.1

5.4 Time plan

Table 5.1: PhD timeline

month	0	3	6	9	12	15	18	21	24	27	30	33	36	39	42	45	48
Literature review	✓	✓	✓														
Numerical modelling (Data collection)		✓	✓	✓	✓	✓	✓										
Statistics Methods learning		✓	✓	✓	✓	✓	✓	✓	✓								
Statistics analysis calibration			✓	✓	✓												
Statistics analysis assimilation						✓	✓	✓	✓	✓	✓						
Statistics analysis prediction												✓	✓	✓			
Thesis writing															✓	✓	✓
Journal/Conference								✓				✓					✓

References

Andrieu, C., De Freitas, N., Doucet, A., & Jordan, M. I. (2003). An introduction to mcmc for machine learning. *Machine learning*, *50*, 5–43.

API. (2011). *Geotechnical and foundation design considerations.* API Washington, DC.

Bhattacharya, S. (2019). *Design of foundations for offshore wind turbines.* John Wiley & Sons.

Blei, D. M., Kucukelbir, A., & McAuliffe, J. D. (2017). Variational inference: A review for statisticians. *Journal of the American statistical Association*, *112*(518), 859–877.

Buckley, R., Chen, Y. M., Sheil, B., Suryasentana, S., Xu, D., Doherty, J., & Randolph, M. (2023). Bayesian optimization for cpt-based prediction of impact pile drivability. *Journal of Geotechnical and Geoenvironmental Engineering*, *149*(11), 04023100.

Byrne, B. W., & Houlsby, G. T. (2003). Foundations for offshore wind turbines. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, *361*(1813), 2909–2930.

Finno, R. J., & Calvello, M. (2005). Supported excavations: observational method and inverse modeling. *Journal of geotechnical and geoenvironmental engineering*, *131*(7), 826–836.

Frangos, M., Marzouk, Y., Willcox, K., & van Bloemen Waanders, B. (2010). Surrogate and reduced-order modeling: a comparison of approaches for large-scale statistical inverse problems. *Large-Scale Inverse Problems and Quantification of Uncertainty*, 123–149.

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (1995). *Bayesian data analysis*. Chapman and Hall/CRC.

Goodman, J., & Weare, J. (2010). Ensemble samplers with affine invariance. *Communications in applied mathematics and computational science*, *5*(1), 65–80.

Hsein Juang, C., Luo, Z., Atamturktur, S., & Huang, H. (2013). Bayesian updating of soil parameters for braced excavations using field observations. *Journal of Geotechnical and Geoenvironmental Engineering*, *139*(3), 395–406.

Jin, Y., Biscontin, G., & Gardoni, P. (2021). Adaptive prediction of wall movement during excavation using bayesian inference. *Computers and Geotechnics*, *137*, 104249.

Kapteyn, M. G., Pretorius, J. V., & Willcox, K. E. (2021). A probabilistic graphical model foundation for enabling predictive digital twins at scale. *Nature Computational Science*, *1*(5), 337–347.

Lataniotis, C. (2019). *Data-driven uncertainty quantification for high-dimensional engineering problems* (Unpublished doctoral dissertation). ETH Zurich.

Murphy, K. P. (2012). *Machine learning: a probabilistic perspective*. MIT press.

Nakamura, K., Yamamoto, S., & Honda, M. (2011). Sequential data assimilation in geotechnical engineering and its application to seepage analysis. In *14th international conference on information fusion* (pp. 1–6).

Nguyen, L. T., & Nestorović, T. (2016). Nonlinear kalman filters for model calibration of soil parameters for geomechanical modeling in mechanized tunneling. *Journal of Computing in Civil Engineering*, *30*(2), 04015025.

Randolph, M., Cassidy, M., Gourvenec, S., & Erbrich, C. (2005). Challenges of offshore geotechnical engineering. In *Proceedings of the international conference on soil mechanics and geotechnical engineering* (Vol. 16, p. 123).

Randolph, M., & Gourvenec, S. (2017). *Offshore geotechnical engineering*. CRC press.

Royston, R., Sheil, B. B., & Byrne, B. W. (2022). Undrained bearing capacity of the cutting face for an open caisson. *Géotechnique*, *72*(7), 632–641.

Speich, M., Dormann, C. F., & Hartig, F. (2021). Sequential monte-carlo algorithms for bayesian model calibration–a review and method comparison. *Ecological Modelling*, *455*, 109608.

Stuyts, B., Weijtjens, W., & Devriendt, C. (2023). Development of a semi-structured database for back-analysis of the foundation stiffness of offshore wind monopiles. *Acta Geotechnica*, *18*(1), 379–393.

Taborda, D. M., Zdravković, L., Potts, D. M., Burd, H. J., Byrne, B. W., Gavin, K. G., ... others (2020). Finite-element modelling of laterally loaded piles in a dense marine sand at dunkirk. *Géotechnique*, *70*(11), 1014–1029.

Tang, C., Cao, Z.-J., Hong, Y., & Li, W. (2023). State space model of undrained triaxial test data for bayesian identification of constitutive model parameters. *Géotechnique*, 1–15.

Tao, Y.-q., Sun, H.-l., & Cai, Y.-q. (2021). Bayesian inference of spatially varying parameters in soil constitutive models by using deformation observation data. *International Journal for Numerical and Analytical Methods in Geomechanics*, *45*(11), 1647–1663.

Verleysen, M., & François, D. (2005). The curse of dimensionality in data mining and time series prediction. In *International work-conference on artificial neural networks* (pp. 758–770).

Wagner, P.-R., Fahrni, R., Klippel, M., Frangi, A., & Sudret, B. (2020). Bayesian calibration and sensitivity analysis of heat transfer models for fire insulation panels. *Engineering structures*, *205*, 110063.

Wagner, P.-R., Nagel, J., Marelli, S., & Sudret, B. (2022). *UQLab user manual – Bayesian inversion for model calibration and validation* (Tech. Rep.). Chair of Risk, Safety and Uncertainty Quantification, ETH Zurich, Switzerland. (Report UQLab-V2.0-113)

Wang, M., Wang, C., Hnydiuk-Stefan, A., Feng, S., Atilla, I., & Li, Z. (2021). Recent progress on reliability analysis of offshore wind turbine support structures considering digital twin solutions. *Ocean Engineering*, *232*, 109168.

Zdravković, L., Jardine, R. J., Taborda, D. M., Abadias, D., Burd, H. J., Byrne, B. W., ... others (2020). Ground characterisation for pisa pile testing and analysis. *Géotechnique*, *70*(11), 945–960.

Zhao, X., Dao, M. H., & Le, Q. T. (2023). Digital twining of an offshore wind turbine on a monopile using reduced-order modelling approach. *Renewable Energy*, *206*, 531–551.