POLITECNICO DI MILANO

Scuola di Ingegneria dei Sistemi

Corso di Laurea Specialistica in
Ingegneria Matematica

Bayesian Hierarchical Gaussian Process Model: an Application to Multi-Resolution Metrology Data

Relatore: Ch.ma Prof. Bianca Maria COLOSIMO

Tesi di Laurea Specialistica di:
Lucia DOLCI Matr. 675215

Anno Accademico 2009-2010

Ringraziamenti

Ringrazio la mia relatrice Prof.ssa Bianca Maria Colosimo, della Sezione Tecnologie e Produzione del Dipartimento di Meccanica, per la fiducia che ha riposto in me nell'assegnarmi questo stimolante progetto di ricerca, nonostante il mio "tormentato percorso accademico". Le sono grata inoltre per le appassionate discussioni in occasione dei nostri incontri e per avermi introdotta, con il suo corso di Gestione Industriale della Qualità, al mondo della statistica applicata ai problemi industriali.

Ringrazio l'Ing. Paolo Cicorella, studente della Scuola di Dottorato, insostituibile aiuto in questi questi mesi di lavoro presso la sezione Tecnologie e Produzione del Dipartimento di Meccanica. La sua disponibilità, la sua grinta e i suoi consigli sono stati indispensabili.
Un grazie sincero all'Ing. Luca Pagani, per la sua pazienza e il suo inesauribile interesse per qualunque problema sottopostogli.
Un affettuoso pensiero va anche a tutti gli studenti e ricercatori, conosciuti presso il laboratorio della Sezione, per la compagnia di questi mesi.

Dedico questo traguardo a mia mamma, a mio fratello e a mia zia, ai quali spetta il ringraziamento più grande: per essermi stati accanto in ogni momento, per le infinite possibilità che mi hanno riservato e per aver creduto in me.

Ringrazio infine i miei amici storici dell'arte, per avermi sopportata in questi ultimi "mesi bayesiani", nonostante non abbia fatto altro che parlare di cose a loro incompresibili.

Contents

List of Figures

List of Tables

Abstract

In the present work we discuss and extend an existing Bayesian Hierarchical Gaussian Process Model (BHGP) used to integrate data with different accuracies. The low-accuracy data are the deterministic output of a computer experiment and the high-accuracy data come from a more precise computer simulation or a physical experiment. A Gaussian process model is used to fit the low-accuracy data. Then the high-accuracy data are linked to the low-accuracy data using a flexible adjustment model where two further Gaussian processes perform scale and location adjustments. An empirical Bayesian approach is chosen and a Monte Carlo Markov Chain (MCMC) algorithm is used to approximate the predictive distribution at new input sites. The existing BHGP model is then extended in order to model the more general situation where also the low accuracy data come from a physical experiment. A measurement error term needs to be included in the model for the low-accuracy data and the MCMC prediction method is accordingly adjusted. The BHGP model is implemented in Matlab and a validation study is performed to verify the developed code and to evaluate the predictive performance of the model. The extended BHGP model is then applied to a set multi-sensor metrology data in order to model the surface of an object. The low-accuracy data are measured with an innovative optical-based Mobile Spatial Coordinate Measuring System II (MScMS-II), developed at Politecnico di Torino, Italy, and the high-resolution data are acquired with a Coordinate-Measuring Machine (CMM). Comparing the BHGP model with other existing methods allows us to conclude that significative improvements (by $11\% - 22\%$) in terms of prediction error are achieved when low-resolution and high-resolution data are combined using an appropriate adjustment model.

Sommario

Nel presente lavoro si analizza e si estende un modello bayesiano gerarchico che sfrutta i processi gaussiani (BHGP) con lo scopo di integrare dati con diversa accuratezza. I dati a bassa accuratezza provengono da un esperimento computazionale deterministico, mentre quelli ad elevata accuratezza da una simulazione numerica più precisa o da un esperimento fisico. Un processo gaussiano modella i dati a bassa accuratezza, mentre un modello flessibile di aggiustamento collega i dati molto accurati a quelli poco accurati, sfruttando due ulteriori processi gaussiani che svolgono la funzione di parametri di scala e di localizzazione. Si adotta un approccio bayesiano empirico per fare inferenza sui parametri del modello e si sfrutta un algoritmo Markov Chain Monte Carlo (MCMC) per approssimare la distribuzione a posteriori predittiva in corrispondenza di nuovi punti sperimentali. Il modello BHGP esistente viene esteso in modo da poter essere applicato al caso più generale in cui anche i dati a bassa accuratezza provengono da un esperimento fisico. Un temine di errore casuale è introdotto nel modello dei dati a bassa accuratezza e i passi dell'algoritmo MCMC devono essere corretti di conseguenza. Dopo aver implementato il modello in Matlab, si svolge uno studio di validazione per verificare la correttezza del codice e le prestazioni del modello in termini predittivi. Il modello BHGP viene infine applicato a dati di metrologia, provenienti da due distinti strumenti di misura a coordinate. I dati a bassa accuratezza sono misurati con un innovativo dispositivo portatile per la misura a coordinate su larga scala (MScMS-II) sviluppato presso il Politecnico di Torino, mentre quelli ad elevata accuratezza sono acquisiti con una macchina di misura a coordinate (CMM). Paragonando il modello BHGP ad altri modelli analizzati, si riscontra un significativo miglioramento delle prestazioni (dall'11% al 22%) in termini di errore di predizione, quando i dati multi-risoluzione sono combinati usando un opportuno modello di aggiustamento.

Introduction

In any scientific context researchers often have to deal with the analysis and synthesis of data from different types of experiments. Integrating data from distinct sources in an efficient way is a challenging topic.

Such data usually represent the same response of interest but they may be generated using different *mechanisms* (physical or computational) or different numerical methods. Qian and Wu [QW08] describe three situations that occur in general:

1. data sets generated from each mechanism have the same characteristics and share the same trend, so that it would be almost impossible to discern the sources;
2. data from each source share no similar patterns, have different magnitudes and appear to have very little in common;
3. each data set has different characteristics but shows similar trend and behavior.

In the first case the differences between each data set can be ignored and a single model could be used to fit data from all the available sources. Unfortunately this does not happen often in practice.

When we face data from the second category it is reasonable to infer that no efficient method could be adopted to integrate such data. Further investigation on the experiments is required. Researchers should try to consider again the underlying assumptions and better understand the differences in the mechanisms of data generation.

The last situation is the one that mostly occurs in practice and is the one discussed in the present work. The standard approach consists of analyzing data from each source separately. However, it has been acknowledged in a variety of situations that performing integrated analysis may lead to stronger conclusions than distinct analysis. The methods of data analysis that use all the available information from every data source is often referred to - in literature - as *combining information*, *borrowing strength* or *meta-analysis*.

As pointed out in the report from US National Science Council (1992) [US 92], *combining information* has quite a long history. It dates back to the XIX century, when Legendre and Gauss invented the Least Squares. While trying to estimate the orbit of comets and determine the meridian arcs in geodesy they used astronomical observations from different observatories.
In the XX century techniques for integrating information from separate studies were developed by researchers in many scientific fields, from agriculture to medicine, from physics to social sciences. These methodologies are very similar one another, sometimes even identical, but the terminology differs from field to field.
In the last few decades the problem of efficiently integrating multi-source data has become the subject of increasing interest in many different contexts. Advance in computer sciences and development of efficient numerical methods has recently allowed researchers to develop several ways of modeling such data.

Here we illustrate and extend the model developed by Qian and Wu [QW08] to integrate low-resolution and high-resolution data.
The authors treat the common situation in which two data sets coming from sources with different accuracy are available. One source provides data with high accuracy, but it is expensive to run and also time-consuming. It is the case of physical experiments or complex detailed computer simulations. The other source may be another computer experiment that is faster and cheaper to run but gives more approximate results.
[QW08] uses a Gaussian process model to smoothly fit the low-resolution data from the approximate computer experiment. Then, in a second step, the high-resolution data are linked to the low-resolution data using a flexible adjustment model where two Gaussian processes perform scale and location adjustments.
In order to predict the output of the high-resolution experiment at untried points the authors adopt a hierarchical Bayesian approach. This choice has the main advantage of incorporating the uncertainty on the unknown model parameters directly in the Bayesian formulation. In addition, it allows to compute a Bayesian predictive distribution for the high-resolution output at untried points given the

training data.

The purpose of the present work is to discuss and extend the model by [QW08] in order to deal with the the more general situation where the low-accuracy data also come from a physical experiment.
The model will be implemented in Matlab and a validation study of the developed code will be performed.
Finally the extended Bayesian Hierarchical Gaussian Process Model will be applied to a set of coordinate metrology data coming from measuring systems with different accuracies.

Chapter 1

Gaussian process models for multi-resolution data

In the present chapter we address the matter of integrating multi-resolution data using Gaussian process models.
This topic of study arises as a part of a research project (Progetto Integrato 2008) on Large Scale Metrology that involves Politecnico di Milano, Politecnico di Torino and Università degli Studi del Salento.
First of all, we introduce the Gaussian random Process and we show how it is used to model the output of computer experiments or, more in general, correlated data. Then, we introduce two Gaussian process models for integrating multi resolution: the data fusion model by [Qia+06] and the Bayesian Hierarchical Gaussian Process (BHGP) model in the version proposed by [QW08].

1.1 The role of experimentation in scientific research

The aim of experimentation is to answer specific research questions.
In order to study complex systems, the first step is to collect data to analyze. Statistics has provided the methodologies for designing and carrying out empirical studies.

Design of Experiments (DOE) is a discipline that dates back to the beginning of the XX century. Earliest techniques were developed by Fisher in the 1920s and the early 1930s, while he was working as a statistician at the Rothamsted Agricultural Experimental Station near London, England. He introduced the systematic use of Statistics in the design of experimental investigations and his pioneering work set the foundations for modern DOE [Mon01].

Later on, designed experiments were used in a wide variety of scientific contexts and new techniques were developed. For instance, the Response Surface Methodology (RSM), developed by Box and Wilson in chemical and process industry in the 1950s, or case-control clinical trials in medical research are still largely used.

Once experimental data are collected, appropriate techniques are required for data synthesis and analysis. For example, Fisher developed techniques to deal with physical experiments. Analysis of Variance (AnOVa) is a systematic technique to separate treatment effects from random error. Replication, randomization and blocking are used to *reduce* the effect of random error.

In any scientific or technological context, most of the systems studied are extremely complex. For this reason, the task of performing physical experiments to analyze complex processes is rarely achievable. This is due to the high costs or the long time required by the experimentation. Sometimes, for instance in the case of large environmental systems, such as weather models, it is even impossible to design and carry out the experiment procedures [Sac+89].

In the last few decades a new way of conducting experiments, i.e. via numerical computer-based simulations, has become increasingly important.

1.2 Literary review

A significative number of studies by different authors has proven that an integrated analysis of data with different scales and resolutions leads to better results than a separate analysis, combining strength across multiple sources.

The research topic addressed in the present work arises as a step of a PRIN (Progetto Integrato) research project titled "Large-scale coordinate metrology: study and realization of an innovative system based on a network of distributed and cooperative wireless sensors". This project is characterized by a tight collaboration between three Research Units. The first one, based in Politecniclo di Torino is mainly involved in the development/adjustment and metrological characterization of the wireless sensor system. The second, based in Politecnico di Milano, focuses on the metrological performance evaluation of the system and in the integration with a further optical system. The third, based in Università degli studi del Salento, is mainly involved in the study and development of mathematical models for integrating data obtained from systems with different resolutions.

The Bayesian approach for treating multi-resolution models that use Gaussian Processes to fit the observed data has raised increasing research interest in the last ten years.
Xia, Ding, and Mallick, in their recent work submitted to Technometrics [XDM08], provide a detailed review on methodologies developed to integrate deterministic computer simulations with different accuracies or computer simulations with data from physical experiments. They distinguish two main schools of thought on the matter.

They cite the work by Reese et al. [Ree+04] as an example of the first kind of approach identified in literature. Reese et al. analyze data sets observed from three distinct sources: computer experiment, physical experiment and expert opinion. First, they fit an appropriate model for data from each source. Then, they combine all the three sources of information, after using an appropriate flexible integration methodology that takes into account uncertainties and biases in the different data sources. They analyze all the data simultaneously using a Recursive Bayesian Hierarchical Model (RBHM).

In the second kind of approach identified by Xia et al., first a single-resolution model, typically for the low-resolution data, is developed, then a model for high-resolution data that uses low-resolution data as input variables is built. Such model is often called *linkage model* as it connects high resolution to low-resolution data by performing a scale transformation and a shift of location.
This is exactly the approach proposed by Qian and Wu in [Qia+06] and [QW08], that we decided to follow in the present work.
Kennedy and O'Hagan [KO01] build a Gaussian Process model to fit the data from a computer experiment then, they use the data from a physical experiment to adjust the model parameters in order to fit the model to the observed data. This process, known as *calibration*, is implemented in a Bayesian framework.
The work by Higdon et al. [Hig+04] is quite similar to [KO01]. Again a Bayesian approach combined with the use of Gaussian Process aims to calibrate the parameters of a computer simulator using field data (from a physical experiment). The authors mainly stress the role of uncertainty quantification in the whole Bayesian construction.
Finally, [XDM08] is a very interesting work itself. Xia, Ding, and Mallick provide a real case application of [QW08] in metrology, i.e. the same field of the application study we faced. They use high-resolution data measured with a highly precise

Coordinate Measurement Machine with a touch probe (CMM) and low-resolution data acquired with a less precise CMM with optical/laser sensor (OCMM). They develop a Gaussian-process model that is more suitable to fit data measured with a coordinate measurement systems compared to the usual universal kriging model [XDW07]. Like we do in the present work, they build a BHGP model that takes into account the measurement error for both the low-resolution and high-resolution data, but they mainly focus on the problem of the misalignment of the experimental points of the two measurements sets.

Though it does not use a Bayesian approach, the work by Qian and Wu, together with Seepersad, Joseph and Allen [Qia+06], deserves to be mentioned. The work [QW08] that followed, consists of a further development of this 2006 paper. First, a Gaussian Process model is used to approximate the low-resolution data. Then, a location-scale adjustment model that uses information from a small set of high-resolution data is used to improve the accuracy of the prediction. In [Qia+06] the scale change is modeled using a linear regression that can only account for linear changes in the scale parameter. In the case of [QW08], the use of a Gaussian process model for both the scale and location parameters allows to take into account more complex changes from low-accuracy and high-accuracy data.
In the present work we focus our attention on the model illustrated in [QW08] and we follow the Bayesian approach proposed by Qian and Wu in their 2008 work.

1.3 The Gaussian random process

Suppose X is a fixed subset of $\mathbb{R}^d$ having positive d-dimensional volume. We say that $Y(\mathbf{x})$, for $\mathbf{x} \in X$, is a Gaussian random Process (GP) provided that for any $k \geq 1$ and any choice of $\mathbf{x}_1, ..., \mathbf{x}_k$ in X, the vector $(Y(\mathbf{x}_1), ..., Y(\mathbf{x}_k))$ has a multivariate normal distribution.

As a direct consequence of this definition, GPs are completely determined by their first and second order moments, i.e. their mean function

$$\mu(\mathbf{x}) = E[Y(\mathbf{x})]$$

and covariance function

$$C(\mathbf{x}_1, \mathbf{x}_2) = Cov[Y(\mathbf{x}_1), Y(\mathbf{x}_2)]$$

for $\mathbf{x}_1, \mathbf{x}_2 \in X$.

In practice, GPs are required to be nonsingular, i.e. for any choice of input $\mathbf{x}$ the covariance matrix of the associated multivariate normal distribution is nonsingular. This property brings great advantages in calculating conditional distribution (of $Y(\mathbf{x}_i)|Y(\mathbf{x}_j)$).

To fulfil the objective of being good interpolators (predictors), Gaussian Process models must assure that their sample path exhibits certain regularity and smoothness properties. Smoothness is achieved with *separability* (Doob, 1953). Thus the GPs we use are chosen to be separable.

Another issue concerns the fact that the output of a computer experiment at training input points represent a single draw of a stochastic process. When predicting the output at a new point, the process must exhibit some regularity over X. Thus, in order to allow valid statistical inference about the process based on a single draw, ergodicity is a required property.

Therefore we restrict our attention to strongly stationary GPs.
The stochastic process $Y(\cdot)$ is *strongly stationary* if, for any $\mathbf{h} \in \mathbb{R}^d$, any $\mathbf{x}_1, ..., \mathbf{x}_k \in X$ with $\mathbf{x}_1 + \mathbf{h}, ..., \mathbf{x}_k + \mathbf{h} \in X$, then $(Y(\mathbf{x}_1), ..., Y(\mathbf{x}_k))$ and $Y(\mathbf{x}_1 + \mathbf{h}), ..., Y(\mathbf{x}_L + \mathbf{h})$ have the same distribution.
When applied to GPs the above definition is equivalent to requiring that $(Y(\mathbf{x}_1), ..., Y(\mathbf{x}_k))$ and $(Y(\mathbf{x}_1 + \mathbf{h}), ..., Y(\mathbf{x}_k + \mathbf{h}))$ always have the same mean and covariance, i.e. they have the same marginal distribution for every $\mathbf{x}$.
Moreover it is not difficult to show that the covariance function of a stationary GP must satisfy:

$$Cov(Y(\mathbf{x}_1), Y(\mathbf{x}_2)) = C(\mathbf{x}_1 - \mathbf{x}_2)$$

i.e. every couple of points with the same orientation and the same distance will have the same covariance.
An even stronger requirement is *isotropy*, which means that a GP is invariant under rotations. This property can be expressed as:

$$Cov(Y(\mathbf{x}_1), Y(\mathbf{x}_2)) = C(\parallel \mathbf{x}_1 - \mathbf{x}_2 \parallel)$$

where $\parallel \cdot \parallel$ is the Euclidean Distance, $\parallel \mathbf{h} \parallel = (\sum_i h_i^2)^{1/2}$.
Isotropic Gaussian processes imply that the associated multivariate normal vectors have the same covariance for any couple of equidistant input points.

A GP is completely defined by its mean and covariance function $C(\cdot)$. In many applications the covariance structure of the process is expressed in terms of both the process variance σ_Y^2 and the process *correlation function* defined as follows:

$$R(\mathbf{h}) = \frac{C(\mathbf{h})}{\sigma_Y^2} \quad \text{for} \quad \mathbf{h} \in \mathbb{R}^d.$$

Correlation functions of stationary GP must be symmetric about the origin and positive semidefinite.
A typical class of desirable correlation functions is the one that links the correlation between errors to the distance between the corresponding points. Euclidean distance is not adequate for this purpose because it equally weights all the variables. The following weighted distance is preferred:

$$d(\mathbf{x}_1, \mathbf{x}_2) = \sum_{j=1}^{k} \phi_j |\mathbf{x}_{1j} - \mathbf{x}_{2j}|^{p_j} \quad \phi_j > 0,\, p_j \in (0, 2].$$

Given this distance definition, a whole class of correlation functions is introduced under the name of *power exponential correlation functions*:

$$R(\mathbf{x}_1, \mathbf{x}_2) = \exp\left\{-d(\mathbf{x}_1, \mathbf{x}_2)\right\} = \exp\left\{-\sum_{j=1}^{k} \phi_j |\mathbf{x}_{1j} - \mathbf{x}_{2j}|^{p_j}\right\} \tag{1.1}$$

where $\phi_j > 0$ and $p_j \in (0, 2]$.
Parameters $\boldsymbol{\phi} = (\phi_1, ..., \phi_k)$, called scale correlation parameters, control how fast the correlation decays with distance, i.e. the activity of correlation along the coordinate directions as a function of distance, and $\mathbf{p} = (p_1, ..., p_L)$, called power parameters, control the smoothness of the sample path of the GP.
Every power-exponential function, for $p_h \in (0, 2]$, is continuous at the origin, though none, except the one with $p_h = 2$, is differentiable at the origin. When the correlation function (1.1) has $p_h = 2$ it is called *Gaussian correlation function*:

$$R(\mathbf{x}_1, \mathbf{x}_2) = \exp\left\{-\sum_{j=1}^{k} \phi_j |\mathbf{x}_{1j} - \mathbf{x}_{2j}|^2\right\} \tag{1.2}$$

Figures (1.1) and (1.2) from [SWN03] show the effect of varying the power ad scale parameters on the sample paths of a GP over $[0, 1]$ with the Gaussian correlation function (1.4).

For power parameters $p \in (0, 2)$ the sample paths are non-differentiable as seen in the panels (b) and (c) from figure (1.1). For $p = 2$ the sample paths, represented in the panel (a), are infinitely differentiable.

Figure (1.1) shows that, when the scale parameter θ decreases, the correlation decreases as well and the sample paths show a behavior closer to the one of random noise (panel (c)). As θ increases, the correlation increases (panel (b)). When the correlation parameters approaches 1 the sample path become closer to the process mean 0.

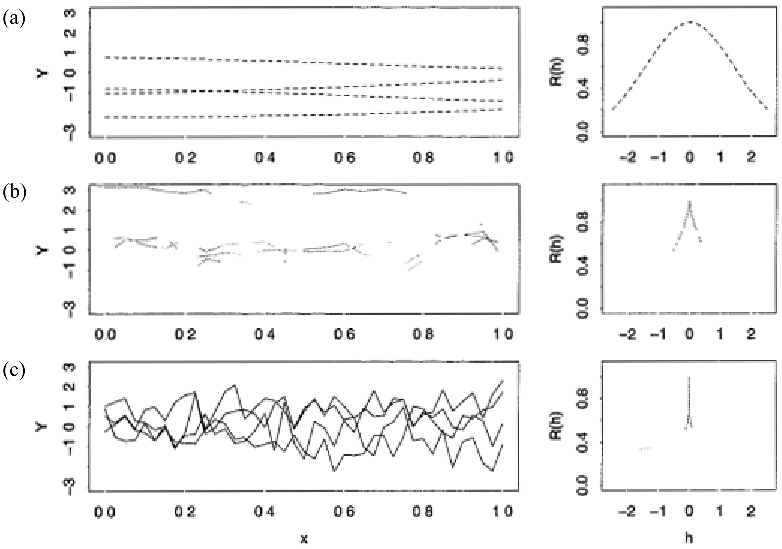

Figure 1.1: The effect of varying the power parameter on the sample paths of a GP with power exponential correlation function. Four draws from a GP$(\mu, \sigma^2, \theta, p)$, with $\mu = 0$, $\sigma^2 = 1.0$, $\theta = 1.0$ and respectively $p = 2.0$ (a), $p = 0.75$ (b) and $p = 0.2$ (c) [SWN03].

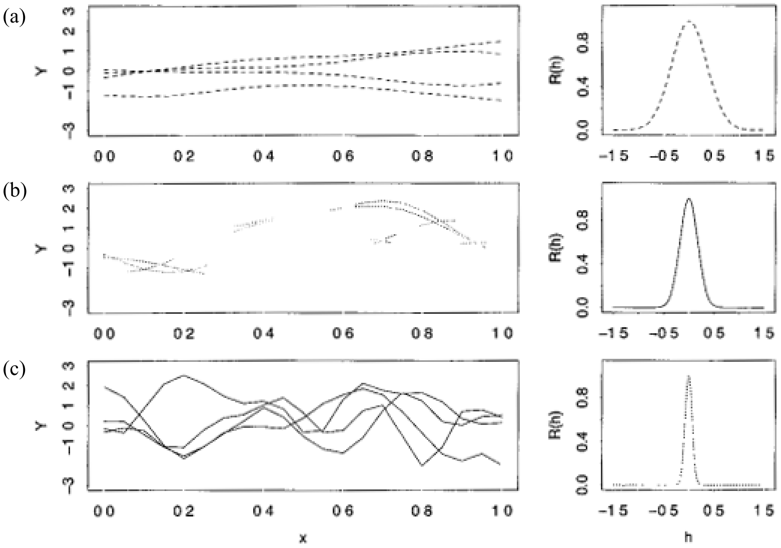

Figure 1.2: The effect of varying the scale parameter on the sample paths of a GP with power exponential correlation function. Four draws from a GP(μ, σ^2, θ, p), with $\mu = 0$, $\sigma^2 = 1.0$, $p = 2.0$ and respectively $\theta = 0.5$ (a), $\theta = 0.25$ (b) and $\theta = 0.1$ (c) [SWN03].

1.4 Computer experiments

Thanks to the advance in mathematical and computational modeling techniques, the technological progress and the enhancement of computational power, the use of computer experiments has become widespread in the last few decades. Mathematical modeling of complex systems and their implementation as computational codes has become common practice in any context of scientific research. Computer experiments allow one to obtain precise and reliable results, with significant savings in time and resources. In addition an important advantage is the possibility of running simulations with the desired level of accuracy.

1.4.1 Characterstics of computer experiments

Computer experiments are designed to have highly multidimensional input, that consists of scalar or functional variables. The output may be multivariate as well and usually represents the response of interest. Most of the cases treated in literature involve a small set of k input variables, usually denoted with $\mathbf{x}_i = (x_{i1}, ..., x_{ik})$, $i = 1, ..., n$, and a single scalar output variable $y(\mathbf{x}_i)$ [Sac+89].

A common feature of many computer experiments is that their output is deterministic. This means that the response variable is not affected by random measurement error and if such a computer code is run with the same input variables it gives the same response.

The lack of random error makes computer experiments different from physical ones and *ad hoc* techniques are required to analyze and model the output of computer experiments. As a matter of fact replication, randomization and blocking are of no use in the design and analysis of computer experiments and the adequacy of a model to fit the data depends only on the systematic bias.
Modeling computer experiments as realizations of random processes allows one to tackle the problem of quantifying the uncertainty associated with predictions using fitted models.

1.4.2 Modeling computer experiments outputs with a Gaussian Process model

Works by Sacks et al. [Sac+89], Santner, Williams, and Notz [SWN03] and Jones, Shonlau, and Welch [JSW98] illustrate a very popular statistical model for fitting the deterministic output of computer experiments. The goal is predicting the response at untried input and estimating prediction uncertainty.

The response of the computer experiment is modeled as the the realization of a random process. This approach is adopted from a branch of Statistics known as Spatial Statistics where it goes under the name of *kriging.* (see the work by Cressie [Cre93])
The stochastic process is described as the combination of a linear regression term that depends on the input variables $\mathbf{x}_i$ and a stochastic process $Z(\cdot)$:

$$Y(\mathbf{x}_i) = \sum_{j=0}^{k} f_j(\mathbf{x}_i)\beta_j + Z(\mathbf{x}_i) = \boldsymbol{f}(\mathbf{x}_i)^T\boldsymbol{\beta} + Z(\mathbf{x}_i) \quad i = 1, ..., n \tag{1.3}$$

where $\boldsymbol{f}(\mathbf{x}_i) = (f_0(\mathbf{x}_i), f_1(\mathbf{x}_i), ..., f_k(\mathbf{x}_i))^T$ is a vector of known linear or non-linear functions of the input variables and $\boldsymbol{\beta} = (\beta_0, \beta_1, ..., \beta_k)^T$ is a vector of unknown regression coefficients.
$Z(\cdot)$ is a zero mean stochastic process completely characterized by its first and second order moments.
In Spatial Statistic literature model (1.3) is known as *universal kriging.*

A very popular choice in literature is the Gaussian random process previously described. Gaussian Process models work as good interpolators when modeling the deterministic output of computer experiments. Moreover they are very flexible when representing complex nonlinear dependencies.

The GP model illustrated in equation (1.3) has the mean term that is a linear function of $\boldsymbol{f}(\mathbf{x}_i)$. This implies that $Y(\mathbf{x}_i)$ is a non-stationary process according to the definition provided in Section 1.3. This particular form of the GP model allows us to have more flexibility. Stationarity properties are though retained by the GP $Z(\cdot)$ that models the residual part.

Jones, Shonlau, and Welch [JSW98] give some intuitive justifications of the modeling choice of (1.3), in particular of the choice of assuming a correlation structure for the residual term $Z(\cdot)$.
For instance, assume that the residual term of the linear regression model (1.3) is a normally distributed i.i.d. error, $Z(\cdot) \sim N(0, \sigma^2)$. Suppose we have determined some suitable functional form for the regression terms. The assumption of $Z(\cdot)$ to be a random error does not stand when modeling the output of a computer code. As said above the output of computer experiment is not affected by random independent error due to measurement or noise. Since the output is deterministic, any lack of fit comes exclusively from modeling errors, i.e. incomplete set of regression terms $f_j(\mathbf{x}_i)$, $j = 1, ..., k$. This allows us to write the residual term as a function of the input, $Z(\mathbf{x}_i)$. Moreover if $Y(\mathbf{x}_i)$ is continuous, the error is also continuous as it is the difference between $Y(\mathbf{x}_i)$ and the continuous regression terms. So, if $\mathbf{x}_1$ and $\mathbf{x}_2$ are two close experimental points, then the errors $z(\mathbf{x}_1)$ and $z(\mathbf{x}_2)$ should also be close. Thus it is reasonable to assume that the residues are correlated.

A typical choice for the correlation structure is the Gaussian correlation function:

$$R(\mathbf{x}_1, \mathbf{x}_2) = \exp\left\{ -\sum_{j=1}^{k} \phi_i |x_{1j} - x_{2j}|^2 \right\}, \quad \phi_i > 0, \quad \forall j = 1, ..., k \tag{1.4}$$

where the scale correlation parameters $\boldsymbol{\phi} = (\phi_1, ..., \phi_k)$ control how fast the correlation decreases with distances, i.e. the "activity" of correlation along the coordinate directions as a function of the distance.
As stated in Section 1.3, the correlation function belonging to the this class are continuous and infinitely differential at the origin and determine the sample path of the corresponding GP to be smooth (infinitely differentiable).

1.5 Gaussian Process models for multi-resolution data

Here we illustrate the two Gaussian Process models developed by Qian and Wu [QW08] in both their works [Qia+06] and [QW08].

As previously outlined, the authors consider the situation in which two kinds of experiments provide data with different resolutions. There is a low-accuracy experiment (LE), fast to run, but approximate, and a high-accuracy experiment (HE), detailed but expensive.

LE and HE are supposed to have a common set of k factors $\mathbf{x} = (x_1, ..., x_k)$. The set of design variables for LE is denoted with $D_l = \{\mathbf{x}_1, ..., \mathbf{x}_n\}$. The corresponding low resolution data are indicated with $\mathbf{y}_l = (\mathrm{y}_l(\mathbf{x}_1), ..., \mathrm{y}_l(\mathbf{x}_n))^T$. The design set for HE is denoted with $D_h = \{\mathbf{x}_1, ..., \mathbf{x}_{n_1}\}$ and the corresponding high-resolution data are $\mathbf{y}_h = (\mathrm{y}_l(\mathbf{x}_1), ..., \mathrm{y}_l(\mathbf{x}_{n_1}))^T$. We assume that the number of available LE data is greater than the number of HE data ($n > n_1$), since HE data require longer times and more resources to be computed. Thus we assume that $D_h \subset D_l$ without loss of generality.

The main purpose of these model is prediction of the HE response at untried input points ($\mathbf{x}_0 \notin D_h$).

1.5.1 Low-accuracy experimental data

The authors treat the case in which LE data come from a computer experiment. Thus the modeling techniques described in Section 1.4.2 apply.

The model for LE response is:

$$\mathrm{y}_l(\mathbf{x}_i) = \boldsymbol{f}_l^T(\mathbf{x}_i)\boldsymbol{\beta}_l + \epsilon_l(\mathbf{x}_i) \quad i = 1, ..., n, \tag{1.5}$$

where $\boldsymbol{f}(\mathbf{x}_i) = (f_0(\mathbf{x}_i), f_1(\mathbf{x}_i), ..., f_k(\mathbf{x}_i))^T$ is a vector of known functions and $\boldsymbol{\beta}_l = (\beta_{l0}, \beta_{l1}, ..., \beta_{lk})^T$ is a vector of unknown regression coefficients.

$\epsilon_l(\cdot)$ is assumed to be a zero mean Gaussian Process. Its covariance function depends on the process variance σ_l^2 and $\boldsymbol{\phi}_l$, a vector of unknown correlation parameters. These parameters are the scale correlation parameters that appear in the Gaussian correlation function in the form (1.4):

$$R_l(\mathbf{x}_j, \mathbf{x}_m) = \exp\left\{-\sum_{i=1}^{k} \phi_{li}(x_{ji} - x_{mi})^2\right\}, \quad j, m = 1, ..., n \tag{1.6}$$

where $\phi_{li} > 0, \forall i = 1, ..., k$ and the power correlation parameter is set to 2. As previously explained this particular class of correlation functions produces sample paths of the corresponding GP that are infinitely differentiable. Given that $R(\cdot, \cdot)$ belongs to the Gaussian correlation functions class, the GP $\epsilon_l(\cdot)$ is completely defined by its mean μ (which is 0), its variance σ_l^2 and its correlation parameters $\boldsymbol{\phi}_l$. For simplicity we will refer to it as $GP(0, \sigma_l^2, \boldsymbol{\phi}_l)$. It directly follows that $\mathrm{y}_l(\mathbf{x}_i) \sim GP(\boldsymbol{f}_l^T(\mathbf{x}_i)\boldsymbol{\beta}_l, \sigma_l^2, \boldsymbol{\phi}_l)$.

[Qia+06] and [QW08] introduce the assumption that the factors considered in the experimentation have linear effect on the output, i.e. $f_l^T(\mathbf{x}_i) = (1, x_{i1}, ..., x_{ik})^T$. Moreover, they state that "[...] inclusion of *weak* main effects in the mean of a Gaussian Process can have additional numerical benefits for estimating the correlation parameters. [...] For a large number of observations [the likelihood function of $\mathbf{y}_l$] can be very small regardless the values of $\boldsymbol{\phi}_l$. As a result, $\boldsymbol{\phi}_l$ cannot be estimated accurately". This statement is confirmed by results of the numerical examples provided in the paper.

[SWN03] specifies the following prior distributions on the unknown parameters of the model:

$$\begin{aligned} p(\sigma_l^2) &\sim IG(\alpha_l, \gamma_l) \\ p(\boldsymbol{\beta}_l|\sigma_l^2) &\sim N(\mathbf{u}_l, \mathrm{v}_l\mathbf{I}_{(k+1)\times(k+1)}\sigma_l^2) \\ p(\phi_{li}) &\sim G(a_l, b_l) \quad \forall \quad i = 1, ..., k. \end{aligned} \tag{1.7}$$

with the following structure:

$$p(\boldsymbol{\beta}_l, \sigma_l^2, \boldsymbol{\phi}_l) = p(\boldsymbol{\beta}_l, \sigma_l^2)p(\boldsymbol{\phi}_l) = p(\boldsymbol{\beta}_l|\sigma_l^2)p(\sigma_l^2)p(\boldsymbol{\phi}_l) \tag{1.8}$$

assuming that $\boldsymbol{\beta}_l$ and σ_l^2 are both independent of $\boldsymbol{\phi}_l$.

If LE were the only source available, given the Gaussian process prior for the true realization of the computer experiment and the priors for the unknown model parameters, it would have been possible to compute the posterior predictive distribution of $y(\mathbf{x}_0)|\mathbf{y}_l, \boldsymbol{\phi}_l$ as the following noncentral t distribution [SWN03]:

$$y(\mathbf{x}_0)|\mathbf{y}_l, \boldsymbol{\phi}_l \sim T_1(\nu_1, \mu_1, \sigma_1^2), \tag{1.9}$$

where the correlation parameters $\boldsymbol{\phi}_l$ are known and ν_1, μ_1 and σ_1^2 are defined as:

$$\begin{aligned}
&\nu_1 = n + \nu_0, \quad \nu_0 = 2a_l, \quad c_0 = \sqrt{b_l/a_l},\\
&\mu_1 = \boldsymbol{f}_l(\mathbf{x}_0)\boldsymbol{\mu} + \mathbf{r}_{l\,0}\mathbf{R}_l^{-1}(\mathbf{y}_l - \mathbf{F}_l\boldsymbol{\mu}),\\
&\boldsymbol{\mu} = \left(\mathbf{F}_l^T\mathbf{R}_l^{-1}\mathbf{F} + \frac{1}{\mathrm{v}_l}\mathbf{I}_{n\times n}\right)^{-1}\left(\mathbf{F}_l^T\mathbf{R}_l^{-1}\mathbf{y}_l + \frac{\mathbf{u}_l}{\mathrm{v}_l}\mathbf{I}_{n\times n}\right),\\
&\hat{\boldsymbol{\beta}}_l = \left(\mathbf{F}_l^T\mathbf{R}_l^{-1}\mathbf{F}\right)^{-1}\left(\mathbf{F}_l^T\mathbf{R}_l^{-1}\mathbf{y}_l\right),\\
&\sigma_1^2 = \frac{Q_1^2}{\nu_1}\left(1 - [\ \boldsymbol{f}_l^T(\mathbf{x}_0) \quad \mathbf{r}_{l\,0}^T\ ]\begin{bmatrix} -\frac{1}{\mathrm{v}_l\mathbf{I}_{n\times n}} & \mathbf{F}_l^T, \\ \mathbf{F}_l^T & \mathbf{R}_l \end{bmatrix}^{-1}\begin{bmatrix} \boldsymbol{f}_l(\mathbf{x}_0) \\ \mathbf{r}_{l\,0} \end{bmatrix}\right), \qquad (1.10)\\
&Q_1 = c_0 + \mathbf{y}_l^T[\mathbf{R}_l^{-1} - \mathbf{R}_l^{-1}\mathbf{F}_l(\mathbf{F}_l^T\mathbf{R}_l^{-1}\mathbf{F}_l)^{-1}\mathbf{F}_l^T\mathbf{R}_l^{-1}]\mathbf{y}_l +\\
&\qquad (\mathbf{u}_l - \hat{\boldsymbol{\beta}}_l)^T[\mathrm{v}_l\mathbf{I}_{n\times n} + (\mathbf{F}_l^T\mathbf{R}_l^{-1}\mathbf{F}_l)^{-1}](\mathbf{u}_l - \hat{\boldsymbol{\beta}}_l),\\
&\mathbf{r}_{l\,0} = [\mathbf{R}_l(\mathbf{x}_0, \mathbf{x}_1), ..., \mathbf{R}_l(\mathbf{x}_0, \mathbf{x}_n)]^T.
\end{aligned}$$

The probability density function of (1.9) is:

$$p(z) = \frac{\Gamma((\nu_1 + 1)/2)}{\sigma_1(\nu_1\pi)^{1/2}\Gamma((\nu_1/2)}\left[1 + \frac{1}{\nu_1}\frac{(z - \mu_1)^2}{\sigma_1^2}\right]^{-(\nu_1+1)/2}. \qquad (1.11)$$

1.5.2 High-Accuracy experimental data

Since LE data are not very accurate and HE data are available, it is convenient to integrate the two data sets in order to improve the quality of the prediction model.

Qian and Wu propose two different adjustment models: Adjustment model I, described in [Qia+06] and Adjustment model II, described in [QW08].

1.5.3 Adjustment model I ("QW06")

In [Qia+06] the following adjustment model to link the high-resolution data to the low-resolution is proposed:

$$\mathrm{y}_h(\mathbf{x}_i) = \rho(\mathbf{x}_i)\mathrm{y}_{l1}(\mathbf{x}_i) + \delta(\mathbf{x}_i) \qquad (1.12)$$

where the scale parameter $\rho(\cdot)$ is a linear regression function:

$$\rho(\mathbf{x}_i) = \rho_0 + \sum_{j=1}^{k}\rho_j\mathrm{x}_{ij}, \quad j = 1, ..., n_1, \qquad (1.13)$$

and the location parameter $\delta(\cdot)$ is assumed to be a stationary $GP(\delta_0, \sigma_\delta^2, \boldsymbol{\phi}_\delta)$, with mean δ_0, variance σ_δ^2 and correlation parameters $\boldsymbol{\phi}_\delta$. The correlation structure is defined by the Gaussian correlation function:

$$R_\delta(\mathbf{x}_j, \mathbf{x}_m) = \exp\left\{-\sum_{i=1}^{k} \phi_{\delta i}(x_{ji} - x_{mi})^2\right\} \quad j, m = 1, ..., n_1. \tag{1.14}$$

We address Adjustment Model I as “QW06”.

1.5.4 Adjustment model II (“QW08”)

In the paper [QW08] the high-resolution data are connected to the low-resolution data using a flexible adjustment model in the form:

$$\mathrm{y}_h(\mathbf{x}_i) = \rho(\mathbf{x}_i)\mathrm{y}_l(\mathbf{x}_i) + \delta(\mathbf{x}_i) + \epsilon(\mathbf{x}_i) \quad i = 1, ..., n_1. \tag{1.15}$$

Here $\rho(\cdot) \sim GP(\rho_0, \sigma_\rho^2, \boldsymbol{\phi}_\rho)$ and $\delta(\cdot) \sim GP(\delta_0, \sigma_\delta^2, \boldsymbol{\phi}_\delta)$. They respectively work as scale and location parameters. Their Gaussian correlation functions are respectively:

$$R_\rho(\mathbf{x}_j, \mathbf{x}_m) = \exp\left\{-\sum_{i=1}^{k} \phi_{\rho i}(x_{ji} - x_{mi})^2\right\}, \tag{1.16}$$

$$R_\delta(\mathbf{x}_j, \mathbf{x}_m) = \exp\left\{-\sum_{i=1}^{k} \phi_{\delta i}(x_{ji} - x_{mi})^2\right\} \quad j, m = 1, ..., n_1. \tag{1.17}$$

If HE data come from a physical experiment, measurement error must be taken into account and it is modeled by $\epsilon(\cdot) \sim N(0, \sigma_\varepsilon^2)$, such that $\epsilon(\mathbf{x}_i) \perp \epsilon(\mathbf{x}_j)$, $\forall i \neq j$. $\mathrm{y}_l(\cdot)$, $\rho(\cdot)$, $\delta(\cdot)$ and $\varepsilon(\cdot)$ are assumed independent.
We address Adjustment Model I as “QW08”.

1.5.5 Specification of prior distributions of the unknown parameters

The model defined by equations (1.5) and (1.15) includes several unknown parameters $\boldsymbol{\theta}$. [QW08] addressed the inferential analysis on such parameters in a Bayesian framework and they call the model a Bayesian Hierarchical Gaussian Process Model (BHGP).
For an overview on Bayesian Inference refer to Appendix A.1.

The unknown parameters $\boldsymbol{\theta}$ are grouped into three sets:

$$\begin{aligned}\boldsymbol{\theta}_1 &= (\boldsymbol{\beta}_l, \rho_0, \delta_0)\\ \boldsymbol{\theta}_2 &= (\sigma_l^2, \sigma_\rho^2, \sigma_\delta^2, \sigma_\epsilon^2)\\ \boldsymbol{\theta}_3 &= (\boldsymbol{\phi}_l, \boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta)\end{aligned}$$

respectively mean, variance and correlation parameters.

If we make the assumption that the mean and the variance parameters are both independent of the correlation parameters, the prior distribution of $\boldsymbol{\theta}$ can be assumed to have the following structure:

$$p(\boldsymbol{\theta}) = p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3) = p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)p(\boldsymbol{\theta}_3) = p(\boldsymbol{\theta}_1|\boldsymbol{\theta}_2)p(\boldsymbol{\theta}_2)p(\boldsymbol{\theta}_3) \tag{1.18}$$

where the last equality is true because $p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) = p(\boldsymbol{\theta}_1|\boldsymbol{\theta}_2)p(\boldsymbol{\theta}_2)$ always holds. This choice simplifies a lot the definition of the parameter priors.

The problem of choosing an adequate set of prior distributions for the model parameters is not trivial at all. In the hierarchical framework, the choice of a set of adequate parameter values for the priors could be difficult. For this reason it is tempting to use non-informative priors. The problem with non-informative priors is that they often lead to improper posterior distributions, that are completely useless for inference purposes.

Given these observations, [QW08] selects the following proper priors:

$$\begin{aligned}p(\sigma_l^2) &\sim IG(\alpha_l, \gamma_l)\\ p(\sigma_\rho^2) &\sim IG(\alpha_\rho, \gamma_\rho)\\ p(\sigma_\delta^2) &\sim IG(\alpha_\delta, \gamma_\delta)\\ p(\sigma_\epsilon^2) &\sim IG(\alpha_\epsilon, \gamma_\epsilon)\\ p(\boldsymbol{\beta}_l|\sigma_l^2) &\sim N(\mathbf{u}_l, \mathrm{v}_l\mathbf{I}_{(k+1)\times(k+1)}\sigma_l^2)\\ p(\rho_0|\sigma_\rho^2) &\sim N(\mathrm{u}_\rho, \mathrm{v}_\rho\sigma_\rho^2)\\ p(\delta_0|\sigma_\delta^2) &\sim N(\mathrm{u}_\delta, \mathrm{v}_\delta\sigma_\delta^2)\\ p(\phi_{li}) &\sim G(a_l, b_l)\\ p(\phi_{\rho i}) &\sim G(a_\rho, b_\rho)\\ p(\phi_{\delta i}) &\sim G(a_\delta, b_\delta) \quad \forall \quad i = 1, ..., k.\end{aligned}$$

$IG(\alpha, \gamma)$ denotes the Inverse-Gamma distribution with density function:

$$p(z) = \frac{\gamma^{\alpha}}{\Gamma(\alpha)} z^{-(\alpha+1)} \exp\left\{-\frac{\gamma}{z}\right\} \quad z > 0.$$

$G(a, b)$ is the Gamma distribution with the density function:

$$p(z) = \frac{b^{a}}{\Gamma(a)} z^{(a-1)} \exp\{-bz\} \quad z > 0.$$

It can be noticed that the prior structure of the mean and variance parameters is the same of the well known Normal-Inverse Gamma conjugate model.
The specification of the prior of the correlation parameters $\boldsymbol{\phi}_l$, $\boldsymbol{\phi}_\rho$ and $\boldsymbol{\phi}_\delta$ depends on the choice of the correlation function. In the case of the Gaussian correlation function a common choice is a Gamma prior distribution [BCG04]

Now that the BHGP model is completely defined the next step is the prediction of y_h at untried point $\mathbf{x}_0$, given the training data $\mathbf{y}_h$ and $\mathbf{y}_l$.

Chapter 2

Bayesian prediction and MCMC sampling in multi-resolution data modeling

In the present chapter we focus our attention on the Bayesian Hierarchical Gaussian Process (BHGP) model by [QW08] introduced in Section 1.5.
Once the Gaussian Process model for LE data and the linkage model for HE data are defined and the prior distributions for the unknown parameters $\boldsymbol{\theta}$ are chosen, the BHGP model is complete. In order to predict $\mathrm{y}_h(\cdot)$ at untried point, the Bayesian posterior predictive density function $p(\mathrm{y}_l(\mathbf{x}_0)|\mathbf{y}_h, \mathbf{y}_l)$ needs to be computed.
Here we describe the method proposed in [QW08] to approximate the posterior predictive density function.
Then we extend the BHGP model proposed in [QW08], in order to model the more general situation where both the low-accuracy data and the high-accuracy data come from a physical experiment, i.e. they are affected by measurement error. We also modify the procedure for approximating the posterior predictive density function to take into account the introduced modifications.

At first we assume that the untried input point $\mathbf{x}_0$ belongs to D_l, but is not a point in D_h. Later on we will relax this assumption.

2.1 Bayesian predictive density function

The prediction of $\mathrm{y}_h(\mathbf{x}_0)$ given the LE and HE training data is computed with the Bayesian predictive density function $p(\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_h, \mathbf{y}_l)$ defined as follows:

$$\begin{aligned} p(\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_h, \mathbf{y}_l) &= \int_{\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\boldsymbol{\theta}_3} p(\mathrm{y}_h(\mathbf{x}_0), \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3|\mathbf{y}_h, \mathbf{y}_l)\, d\boldsymbol{\theta}_1 d\boldsymbol{\theta}_2 d\boldsymbol{\theta}_3 \\ &= \int_{\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\boldsymbol{\theta}_3} p(\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_h, \mathbf{y}_l, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3) \times \\ &\qquad \times p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3|\mathbf{y}_h, \mathbf{y}_l)\, d\boldsymbol{\theta}_1 d\boldsymbol{\theta}_2 d\boldsymbol{\theta}_3. \end{aligned} \tag{2.1}$$

The integral in $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3$ (2.1) of the joint posterior distribution of $(\mathrm{y}_h(\mathbf{x}_0), \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3)$ is quite complicated. It could even be impossible to compute analytically. A Markov Chain Monte Carlo (MCMC) algorithm is used to approximate such predictive distribution.

2.1.1 MCMC algorithm to approximate the Bayesian predictive density

Banerjee, Carlin, and Gelfand [BCG04] describe a two-step algorithm to estimate the predictive Bayesian density (2.1).

1. First M posterior samples $\left(\boldsymbol{\theta}_1^{(i)}, \boldsymbol{\theta}_2^{(i)}, \boldsymbol{\theta}_3^{(i)}\right)$, $i = 1, ..., M$, are drawn (after a properly chosen burn-in period) from the joint posterior distribution of the parameters $p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3|\mathbf{y}_h, \mathbf{y}_l)$.
2. Then the predictive Bayesian density $p(\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_h, \mathbf{y}_l)$ is computed as a Monte Carlo mixture of the form:

$$\hat{p}_m(\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h) = \frac{1}{M}\sum_{i=1}^{M} p\left(\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h, \boldsymbol{\theta}_1^{(i)}, \boldsymbol{\theta}_2^{(i)}, \boldsymbol{\theta}_3^{(i)}\right). \tag{2.2}$$

The posterior sampling in step one requires some care.

It would be preferable to marginalize the joint posterior distribution of the parameters $p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3|\mathbf{y}_h, \mathbf{y}_l)$ and compute independent estimates of $(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3)$ directly using their marginal posterior distributions. Unfortunately closed form marginalization of posteriors is rarely achievable in practice. As a result the posterior sampling of $(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3)$ needs to be carried out using an MCMC sampling method.

For an overview on MCMC sampling techniques applied to Bayesian Inference refer to Appendix A.2.

In this particular case, we have to sample $7+3k$ parameters (3 mean parameters, 4 variance parameters and $3k$ correlation parameters, where k is the number of regression variables).

2.1.2 Joint posterior distribution of the model parameters

The joint posterior distribution of the unknown parameters $(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3)$ is:

$$
\begin{aligned}
p(\boldsymbol{\theta}_1,&\boldsymbol{\theta}_2,\boldsymbol{\theta}_3|\mathbf{y}_l,\mathbf{y}_h) \propto \\
&\frac{1}{(\sigma_l^2)^{n/2}}\exp\left\{-\frac{1}{2\mathrm{v}_l\sigma_l^2}(\boldsymbol{\beta}_l-\mathbf{u}_l)^T(\boldsymbol{\beta}_l-\mathbf{u}_l)\right\}\times \\
&\times\frac{1}{(\sigma_\rho^2)^{1/2}}\exp\left\{-\frac{(\rho_0-\mathrm{u}_\rho)^2}{2\mathrm{v}_\rho\sigma_\rho^2}\right\}\cdot\frac{1}{(\sigma_\delta^2)^{1/2}}\exp\left\{-\frac{(\delta_0-\mathrm{u}_\delta)^2}{2\mathrm{v}_\rho\sigma_\delta^2}\right\}\times \\
&\times(\sigma_l^2)^{-(\alpha_l+1)}\exp\left\{-\frac{\gamma_l}{\sigma_l^2}\right\}\cdot(\sigma_\rho^2)^{-(\alpha_\rho+1)}\exp\left\{-\frac{\gamma_\rho}{\sigma_\rho^2}\right\}\times \\
&\times(\sigma_\delta^2)^{-(\alpha_\delta+1)}\exp\left\{-\frac{\gamma_\delta}{\sigma_\delta^2}\right\}\cdot(\sigma_\epsilon^2)^{-(\alpha_\epsilon+1)}\exp\left\{-\frac{\gamma_\epsilon}{\sigma_\epsilon^2}\right\}\times \\
&\times\prod_{i=1}^{k}\big(\phi_{li}^{(\alpha_l-1)}\exp\{-b_l\phi_{li}\}\cdot\phi_{\rho i}^{(\alpha_\rho-1)}\exp\{-b_\rho\phi_{\rho i}\}\cdot\phi_{\delta i}^{(\alpha_\delta-1)}\exp\{-b_\delta\phi_{\delta i}\}\big)\times \\
&\times\frac{1}{|\mathbf{Q}|^{1/2}}\exp\left\{-\frac{1}{2}(\mathbf{y}_h-\rho_0\mathbf{y}_{l_1}-\delta_0\mathbf{1}_{n_1})^T\mathbf{Q}^{-1}(\mathbf{y}_h-\rho_0\mathbf{y}_{l_1}-\delta_0\mathbf{1}_{n_1})\right\}\times \\
&\times\frac{1}{(\sigma_l^2)^{n/2}|\mathbf{R}_l|^{1/2}}\exp\left\{-\frac{1}{2\mathrm{v}_l\sigma_l^2}(\mathbf{y}_l-\mathbf{F}_l\boldsymbol{\beta}_l)^T\mathbf{R}_l^{-1}(\mathbf{y}_l-\mathbf{F}_l\boldsymbol{\beta}_l)\right\}
\end{aligned}
\tag{2.3}
$$

where $\mathbf{Q} = \sigma_\rho^2\mathbf{A}_1\mathbf{R}_\rho\mathbf{A}_1 + \sigma_\delta^2\mathbf{R}_\delta + \sigma_\epsilon^2\mathbf{I}_{(k+1)\times(k+1)}$, $A_1 = \mathrm{diag}\{\mathrm{y}_l(\mathbf{x}_1), ..., \mathrm{y}_l(\mathbf{x}_{n_1})\}$ and the correlation matrices $\mathbf{R}_l$, $\mathbf{R}_\rho$ and $\mathbf{R}_\delta$ depend respectively on the unknown correlation parameters $\boldsymbol{\phi}_l$, $\boldsymbol{\phi}_\rho$ and $\boldsymbol{\phi}_\delta$, as shown in Equations (1.6) and (1.16). Refer to Appendix C.1 for the intermediate computations.

The posterior distribution (2.3) has a very complicated form. Sampling of all the $7+3k$ unknown parameters from such a distribution, i.e. performing a fully Bayesian analysis, arises several computational issues, in particular for the set of $3k$ correlation parameters $\boldsymbol{\theta}_3 = (\boldsymbol{\phi}_l, \boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta)$. In fact, the correlation parameters all appear in (2.3) as elements of four complicated matrix determinants and inversions: $|\mathbf{Q}|$, $\mathbf{Q}^{-1}$, $|\mathbf{R}_l|$ and $\mathbf{R}_l^{-1}$. This makes the fully Bayesian analysis nearly impossible to carry out in reasonable computational times.

The full conditional distributions for the correlation parameters in the fully bayesian case are computed in Appendix C.4

In particular, what makes the sampling of the correlation parameters in the fully bayesian framework really awkward can be summarized as follows:

- the full conditional distributions of $\boldsymbol{\phi}_l$, $\boldsymbol{\phi}_\rho$ and $\boldsymbol{\phi}_\delta$ do not belong to any known family of distributions, so an MCMC algorithm should be used;
- in order to minimize the number of matrix determinants and inversions at each iteration of the MCMC algorithm, multivariate sampling of $\boldsymbol{\phi}_l$ first and then $(\boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta)$ may seem the most reasonable choice. On the contrary having to simultaneously draw several parameters with the Metropolis algorithm cause the acceptance rates to be too small, and consequently the convergence times are very slow.

To overcome this difficulty, [QW08] proposes to implement an empirical Bayesian analysis by previously computing posterior point estimates of the correlation parameters $\boldsymbol{\theta}_3$ for computational convenience.

2.2 The empirical Bayesian approach

The empirical Bayesian approach is quite popular in every context that uses (hierarchical) Gaussian process models for fitting functional responses (see for instance [SWN03], [KO01], [Bay+07]).
See Appendix A.4 for a brief explanation of empirical Bayesian approaches to Bayesian inference.
In [QW08] the analysis is carried out in two subsequent steps:

a. First the correlation parameters are estimated for computational convenience by setting them at the values of their posterior modes, that are computed by solving an optimization problem that will be discussed in subsection 2.2.1.

b. Then the estimates of the correlation parameters are *plugged* into the BHGP model and from now on $(\boldsymbol{\phi}_l, \boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta)$ will be considered as they were known.

The authors justify such approach quoting what [Bay+07] says on the matter:

> *Full justification of the use of the plug-in maximum likelihood estimates for the (correlation parameters) is an open theoretical issue. Intuitively, one expects modest variations in parameters to have little effect on*

> *the predictors because they are interpolators. In practice, "Studentized" cross-validation residuals (leave-one-out predictions of the data normalized by standard error) have been successfully used to gauge the "legitimacy" of such usage (...). Only recently, Nagy, Loeppky and Welch [NLW07] have reported simulations indicating reasonably close prediction accuracy of the plug-in MLE predictions to Bayes (Jeffreys priors) predictions in dimensions* $1-10$ *when the number of computer runs* $= 7\times$*dimension.*

This approach of course implies that the estimation uncertainty of the correlation parameters is not taken into account in the prediction process.

2.2.1 Estimation of the correlation parameters

The correlation parameters $\boldsymbol{\theta}_3 = (\boldsymbol{\phi}_l, \boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta)$ are estimated using the their posterior modes, i.e. the values that maximize their marginal posterior distribution (assuming it is unimodal). Such approach is chosen because it is the easiest way to compute a point estimate of unknown parameters in the Bayesian framework because it provides the answer to the point estimation problem by solving an optimization problem [BCG04], once the marginal posterior distribution is known.

The marginal posterior distribution of $\boldsymbol{\theta}_3$ is:

$$\begin{aligned} p(\boldsymbol{\theta}_3|\mathbf{y}_h,\mathbf{y}_l) &= \int_{\boldsymbol{\theta}_1,\boldsymbol{\theta}_2} p(\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\boldsymbol{\theta}_3|\mathbf{y}_h,\mathbf{y}_l)d\boldsymbol{\theta}_1 d\boldsymbol{\theta}_2 = \\ &= \int_{\boldsymbol{\theta}_1,\boldsymbol{\theta}_2} p(\boldsymbol{\theta}_1|\boldsymbol{\theta}_2)p(\boldsymbol{\theta}_2)p(\boldsymbol{\theta}_3)L(\mathbf{y}_h,\mathbf{y}_l|\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\boldsymbol{\theta}_3)d\boldsymbol{\theta}_1 d\boldsymbol{\theta}_2 = \\ &= p(\boldsymbol{\theta}_3)\int_{\boldsymbol{\theta}_1,\boldsymbol{\theta}_2} p(\boldsymbol{\theta}_1|\boldsymbol{\theta}_2)p(\boldsymbol{\theta}_2)L(\mathbf{y}_h,\mathbf{y}_l|\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\boldsymbol{\theta}_3)d\boldsymbol{\theta}_1 d\boldsymbol{\theta}_2 \end{aligned} \tag{2.4}$$

because $p(\boldsymbol{\theta}_3)$ is independent of $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$.

To ease the load of the MCMC computations a new parametrization for the variance parameters $\boldsymbol{\theta}_2 = (\sigma_l^2, \sigma_\rho^2, \tau_1, \tau_2)$ is introduced:

$$(\sigma_l^2, \sigma_\rho^2, \tau_1, \tau_2) = (\sigma_l^2, \sigma_\rho^2, \frac{\sigma_\delta^2}{\sigma_\rho^2}, \frac{\sigma_\epsilon^2}{\sigma_\rho^2}). \tag{2.5}$$

Such reparametrization eases the sampling of σ_ρ^2 from its full conditional distribution. In fact the Bayesian computations lead us to a set of full conditional distributions belonging to known distribution families for both σ_l^2 and σ_ρ^2. If we used the original parametrization for $\boldsymbol{\theta}_2$ we would get only one known-form full conditional distribution for σ_l^2.

As a consequence the integrand in (2.4) becomes:

$$\begin{aligned}
&p(\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\boldsymbol{\theta}_3|\mathbf{y}_l,\mathbf{y}_h) \propto \\
&\quad \frac{1}{(\mathrm{v}_l\sigma_l^2)^{n/2}} \exp\left\{-\frac{1}{2\mathrm{v}_l\sigma_l^2}(\boldsymbol{\beta}_l-\mathbf{u}_l)^T\mathbf{R}_l^{-1}(\boldsymbol{\beta}_l-\mathbf{u}_l)\right\}\times \\
&\quad \times\frac{1}{(\mathrm{v}_\rho\sigma_\rho^2)^{1/2}}\exp\left\{-\frac{(\rho_0-\mathrm{u}_\rho)^2}{2\mathrm{v}_\rho\sigma_\rho^2}\right\}\cdot\frac{1}{(\mathrm{v}_\delta\tau_1\sigma_\rho^2)^{1/2}}\exp\left\{-\frac{(\delta_0-\mathrm{u}_\delta)^2}{2\mathrm{v}_\rho\tau_1\sigma_\rho^2}\right\}\times \\
&\quad \times(\sigma_l^2)^{-(\alpha_l+1)}\exp\left\{-\frac{\gamma_l}{\sigma_l^2}\right\}\cdot(\sigma_\rho^2)^{-(\alpha_\rho+1)}\exp\left\{-\frac{\gamma_\rho}{\sigma_\rho^2}\right\}\times \\
&\quad \times(\tau_1\sigma_\rho^2)^{-(\alpha_\delta+1)}\exp\left\{-\frac{\gamma_\delta}{\tau_1\sigma_\rho^2}\right\}\cdot(\tau_2\sigma_\rho^2)^{-(\alpha_\epsilon+1)}\exp\left\{-\frac{\gamma_\epsilon}{\tau_2\sigma_\rho^2}\right\}\cdot\sigma_\rho^4\times \\
&\quad \times\frac{1}{(\sigma_\rho^2)^{n_1/2}|\mathbf{M}|^{1/2}}\times \\
&\quad \times\exp\left\{-\frac{1}{2\mathrm{v}_\rho\sigma_\rho^2}(\mathbf{y}_h-\rho_0\mathbf{y}_{l_1}-\delta_0\mathbf{1}_{n_1})^T\mathbf{M}^{-1}(\mathbf{y}_h-\rho_0\mathbf{y}_{l_1}-\delta_0\mathbf{1}_{n_1})\right\}\times \\
&\quad \times\frac{1}{(\sigma_l^2)^{n/2}|\mathbf{R}_l|^{1/2}}\exp\left\{-\frac{1}{2\mathrm{v}_l\sigma_l^2}(\mathbf{y}_l-\mathbf{F}\boldsymbol{\beta}_l)^T\mathbf{R}_l^{-1}(\mathbf{y}_l-\mathbf{F}\boldsymbol{\beta}_l)\right\}
\end{aligned} \tag{2.6}$$

where $\mathbf{M}=\mathbf{W}_\rho+\tau_1\mathbf{R}_\delta+\tau_2\mathbf{I}_{n_1\times n_1}$, $\mathbf{W}_\rho=\mathbf{A}_1\mathbf{R}_\rho\mathbf{A}_1$, $\mathbf{A}_1=\mathrm{diag}\{\mathrm{y}_l(\mathbf{x}_1),...,\mathrm{y}_l(\mathbf{x}_{n_1})\}$ and the correlation matrices $\mathbf{R}_l$, $\mathbf{R}_\rho$ and $\mathbf{R}_\delta$ depend respectively on the correlation parameters $\boldsymbol{\phi}_l$, $\boldsymbol{\phi}_\rho$ and $\boldsymbol{\phi}_\delta$ alone.

After integrating out $\boldsymbol{\beta}_l$, ρ_0, δ_0, σ_l^2, σ_ρ^2, (2.4) becomes proportional to an expression that depends on $\boldsymbol{\phi}_l$, $\boldsymbol{\phi}_\rho$, $\boldsymbol{\phi}_\delta$:

$$\begin{aligned}
&p(\boldsymbol{\theta}_3|\mathbf{y}_h,\mathbf{y}_l)\propto L_1(\boldsymbol{\phi}_l,\boldsymbol{\phi}_\rho,\boldsymbol{\phi}_\delta)= \\
&=p(\boldsymbol{\phi}_l)\mid\mathbf{R}_l\mid^{-1/2}\mid\mathbf{H}_1\mid^{-1/2}\left(\gamma_l+\frac{\mathbf{b}_1^T\mathbf{H}_1^{-1}\mathbf{b_1}-c_1}{2}\right)^{-\left(\alpha_l+\frac{n}{2}\right)}\cdot p(\boldsymbol{\phi}_\rho)p(\boldsymbol{\phi}_\delta)\times \\
&\quad\times\int_{\tau_1,\tau_2}(a_2a_3)^{-1/2}\mid\mathbf{M}\mid^{-1/2}\left(\frac{b_3^2-a_3c_3}{2a_3}+\gamma_\rho+\frac{\gamma_\delta}{\tau_1}+\frac{\gamma_\epsilon}{\tau_2}\right)^{-\left(\frac{n_1}{2}+\alpha_\rho+\alpha_\delta+\alpha_\epsilon+1\right)}\times \\
&\quad\times\tau_1^{-\left(\alpha_\delta+\frac{3}{2}\right)}\tau_2^{-(\alpha_\epsilon+1)}d\tau_1d\tau_2,
\end{aligned} \tag{2.7}$$

where $L_1(\boldsymbol{\phi}_l, \boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta)$ is the non-normalized posterior distribution of $\boldsymbol{\theta}_3$ and:

$$\begin{aligned}
\mathbf{H}_1 &= \frac{1}{\mathrm{v}_l}\mathbf{I}_{(k+1)\times(k+1)} + \mathbf{F}_l^T\mathbf{R}_l^{-1}\mathbf{F}_l,\\
\mathbf{b}_1 &= -2\frac{\mathbf{u}_l}{v_l} - 2\mathbf{F}_l^T\mathbf{R}_l^{-1}\mathbf{y}_l,\\
c_1 &= \frac{\mathbf{u}_l^T\mathbf{u}_l}{\mathrm{v}_l} + \mathbf{y}_l^T\mathbf{R}_l^{-1}\mathbf{y}_l,\\
a_2 &= \frac{1}{\mathrm{v}_\rho} + \mathbf{y}_{l_1}^T\mathbf{M}_l^{-1}\mathbf{y}_{l_1},\\
t_1 &= a_2\left(\mathbf{1}_{n_1}^T\mathbf{M}^{-1}\mathbf{1}_{n_1}\right) - \left(\mathbf{y}_{l_1}^T\mathbf{M}^{-1}\mathbf{1}_{n_1}\right)^2,\\
t_2 &= -2\left(a_2\left(\mathbf{y}_h^T\mathbf{M}^{-1}\mathbf{1}_{n_1}\right) - \left(\mathbf{y}_{l_1}^T\mathbf{M}^{-1}\mathbf{1}_{n_1}\right)\left(\frac{u_\rho}{v_\rho} - \mathbf{y}_{l_1}^T\mathbf{M}^{-1}\mathbf{y}_h\right)\right),\\
t_3 &= a_2\left(\frac{\mathrm{u}_\rho^2}{\mathrm{v}_\rho} - \mathbf{y}_{l_1}^T\mathbf{M}^{-1}\mathbf{y}_h\right) - \left(\frac{\mathrm{u}_\rho}{\mathrm{v}_\rho} - \mathbf{y}_{l_1}^T\mathbf{M}^{-1}\mathbf{y}_h\right)^2,\\
a_3 &= \frac{1}{\mathrm{v}_\delta\tau_1} + a_2^{-1}t_1,\\
b_3 &= -2\frac{\mathrm{u}_\delta}{\mathrm{v}_\delta\tau_1} - a_2^{-1}t_2,\\
c_3 &= \frac{\mathrm{u}_\delta^2}{\mathrm{v}_\delta\tau_1} + a_2^{-1}t_3.
\end{aligned}$$

For a detailed description of the integration steps see the Appendix in [QW08].

The posterior mode estimator of $\boldsymbol{\theta}_3$ is given by:

$$\begin{aligned}
\hat{\boldsymbol{\theta}}_3 = (\hat{\boldsymbol{\phi}}_l, \hat{\boldsymbol{\phi}}_\rho, \hat{\boldsymbol{\phi}}_\delta) = \arg\max_{\boldsymbol{\phi}_l,\boldsymbol{\phi}_\rho,\boldsymbol{\phi}_\delta} & L_1(\boldsymbol{\phi}_l, \boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta)\\
\text{s.t. } & (\boldsymbol{\phi}_l, \boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta) > \mathbf{0}
\end{aligned} \tag{2.8}$$

Given the particular formulation of the model, the LE parameters are independent from the HE parameters. This allows us to split Problem (2.8) into the following independent problems:

$$\begin{aligned}
\hat{\boldsymbol{\phi}}_l = & \arg\max_{\boldsymbol{\phi}_l} p(\boldsymbol{\phi}_l) \mid \mathbf{R}_l \mid^{-1/2} \mid \mathbf{H}_1 \mid^{-1/2} \left(\gamma_l + \frac{4c_1 - \mathbf{b}_1^T\mathbf{H}_1^{-1}\mathbf{b}_1}{8}\right)^{-\left(\alpha_l + \frac{n}{2}\right)}\\
& \text{s.t. } \boldsymbol{\phi}_l > \mathbf{0}
\end{aligned} \tag{2.9}$$

and:

$$\begin{aligned}(\hat{\boldsymbol{\phi}}_\rho, \hat{\boldsymbol{\phi}}_\delta) = &\arg\max_{\boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta} p(\boldsymbol{\phi}_\rho) p(\boldsymbol{\phi}_\delta) \int_{\tau_1,\tau_2} (a_2 a_3)^{-1/2} \times \\ &\times \mid \mathbf{M} \mid^{-1/2} \left(\frac{4a_3c_3 - b_3^2}{8a_3} + \gamma_\rho + \frac{\gamma_\delta}{\tau_1} + \frac{\gamma_\epsilon}{\tau_2} \right)^{-\left(\frac{n_1}{2} + \alpha_\rho + \alpha_\delta + \alpha_\epsilon + 1\right)} \times \\ &\times \tau_1^{-\left(\alpha_\delta + \frac{3}{2}\right)} \tau_2^{-(\alpha_\epsilon + 1)} d\tau_1 d\tau_2 \\ &\quad \text{s.t. } (\boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta) > \mathbf{0}. \end{aligned} \tag{2.10}$$

The optimization problem (2.9) is deterministic and can be solved using a standard nonlinear optimization algorithm (e.g. a Quasi-Newton method). The second optimization problem involves the computation of an integral in τ_1 and τ_2 and requires a more elaborate solution technique, called Sample Average Approximation (SAA).

2.2.2 Sample Average Approximation method

Problem (2.10) can be recast as:

$$\begin{aligned}(\hat{\boldsymbol{\phi}}_\rho, \hat{\boldsymbol{\phi}}_\delta) = &\arg\max_{\boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta} \int_{\tau_1,\tau_2} f(\tau_1, \tau_2) p(\tau_1, \tau_2) d\tau_1 d\tau_2 = \\ = &\arg\max_{\boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta} E_{\tau_1,\tau_2}[f(\tau_1, \tau_2)] \\ &\text{s.t. } (\boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta) > \mathbf{0}. \end{aligned} \tag{2.11}$$

where:

$$\begin{aligned} f(\tau_1, \tau_2) = &p(\boldsymbol{\phi}_\rho) p(\boldsymbol{\phi}_\delta) (a_2 a_3)^{-1/2} \mid \mathbf{M} \mid^{-1/2} \times \\ &\times \left(\frac{4a_3c_3 - b_3^2}{8a_3} + \gamma_\rho + \frac{\gamma_\delta}{\tau_1} + \frac{\gamma_\epsilon}{\tau_2} \right)^{-\left(\frac{n_1}{2} + \alpha_\rho + \alpha_\delta + \alpha_\epsilon + 1\right)} \times \\ &\times \exp\left\{ \frac{\gamma_1}{\tau_1} \right\} \exp\left\{ \frac{\gamma_2}{\tau_2} \right\}. \end{aligned} \tag{2.12}$$

$p(\tau_1, \tau_2)$ is the joint density function of τ_1 and τ_2, where:

$$\tau_1 \sim IG(\alpha_\delta + \frac{1}{2}, \gamma_1), \quad \tau_2 \sim IG(\alpha_\epsilon, \gamma_2). \tag{2.13}$$

and τ_1 and τ_2 are independent.

Problem (2.11) can be seen as a stochastic program. Qian and Wu propose to

solve it with the *Sample Average Approximation* (SAA) [RS03].
S independent Monte Carlo samples $(\tau_1^{(s)}, \tau_2^{(s)})$, $s = 1, ..., S$, are drawn from $p(\tau_1, \tau_2)$ and are used for approximating $E_{\tau_1,\tau_2}[f(\tau_1, \tau_2)]$:

$$E_{\tau_1,\tau_2}[f(\tau_1, \tau_2)] \simeq \frac{1}{S} \sum_{s=1}^{S} f(\tau_1^{(s)}, \tau_2^{(s)}). \tag{2.14}$$

The true stochastic optimization problem (2.11) then can be approximated with the corresponding deterministic problem:

$$\begin{gathered}(\hat{\boldsymbol{\phi}}_\rho, \hat{\boldsymbol{\phi}}_\delta) = \arg \max_{\boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta} \left\{ \frac{1}{S} \sum_{s=1}^{S} f(\tau_1^{(s)}, \tau_2^{(s)}) \right\} \\ \text{s.t. } (\boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta) > \mathbf{0}.\end{gathered} \tag{2.15}$$

This problem can be solved with a standard non-linear optimization algorithm.

2.2.3 Two step MCMC algorithm in the empirical Bayesian framework

Once the posterior modes of the correlation parameters are computed, we fix the values of $\boldsymbol{\phi}_l$, $\boldsymbol{\phi}_\rho$ and $\boldsymbol{\phi}_\delta$ to their estimates and proceed with the Bayesian computations as the values $\tilde{\boldsymbol{\theta}}_3$ of such parameters were given.
Therefore from now on the dependencies on the correlation parameters will be omitted.

The Bayesian predictive distribution (2.1) becomes:

$$\begin{aligned} p(\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_h, \mathbf{y}_l) &= \int_{\boldsymbol{\theta}_1,\boldsymbol{\theta}_2} p(\mathrm{y}_h(\mathbf{x}_0), \boldsymbol{\theta}_1, \boldsymbol{\theta}_2|\mathbf{y}_h, \mathbf{y}_l)\, d\boldsymbol{\theta}_1 d\boldsymbol{\theta}_2 \\ &= \int_{\boldsymbol{\theta}_1,\boldsymbol{\theta}_2} p(\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_h, \mathbf{y}_l, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2) \times \\ &\qquad \times p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2|\mathbf{y}_h, \mathbf{y}_l)\, d\boldsymbol{\theta}_1 d\boldsymbol{\theta}_2. \end{aligned} \tag{2.16}$$

The two-step MCMC algorithm illustrated in [BCG04] is still useful to carry out the integration in (2.16). The steps are identical to the ones described in section 2.1.1 with the only difference that now the correlation parameters are known. This time M samples $(\boldsymbol{\theta}_1^{(i)}, \boldsymbol{\theta}_2^{(i)})$, $i = 1, ..., M$, are taken from the posterior distribution $p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2|\mathbf{y}_h, \mathbf{y}_l)$ and the Bayesian predictive density described in step

2. is approximated by:

$$\hat{p}_m(\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_l,\mathbf{y}_h) = \frac{1}{M}\sum_{i=1}^{M} p\left(\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_l,\mathbf{y}_h,\boldsymbol{\theta}_1^{(i)},\boldsymbol{\theta}_2^{(i)}\right). \tag{2.17}$$

The joint posterior distribution for the unknown parameters $(\boldsymbol{\theta}_1,\boldsymbol{\theta}_2)$ is:

$$\begin{aligned} p(\boldsymbol{\theta}_1,\boldsymbol{\theta}_2|\mathbf{y}_l,\mathbf{y}_h) \propto & \\ & \frac{1}{(\sigma_l^2)^{n/2}}\exp\left\{-\frac{1}{2\mathrm{v}_l\sigma_l^2}(\boldsymbol{\beta}_l-\mathbf{u}_l)^T(\boldsymbol{\beta}_l-\mathbf{u}_l)\right\}\times \\ & \times\frac{1}{(\sigma_\rho^2)^{1/2}}\exp\left\{-\frac{(\rho_0-\mathrm{u}_\rho)^2}{2\mathrm{v}_\rho\sigma_\rho^2}\right\}\cdot\frac{1}{(\sigma_\delta^2)^{1/2}}\exp\left\{-\frac{(\delta_0-\mathrm{u}_\delta)^2}{2\mathrm{v}_\rho\sigma_\delta^2}\right\}\times \\ & \times(\sigma_l^2)^{-(\alpha_l+1)}\exp\left\{-\frac{\gamma_l}{\sigma_l^2}\right\}\cdot(\sigma_\rho^2)^{-(\alpha_\rho+1)}\exp\left\{-\frac{\gamma_\rho}{\sigma_\rho^2}\right\}\times \\ & \times(\sigma_\delta^2)^{-(\alpha_\delta+1)}\exp\left\{-\frac{\gamma_\delta}{\sigma_\delta^2}\right\}\cdot(\sigma_\epsilon^2)^{-(\alpha_\epsilon+1)}\exp\left\{-\frac{\gamma_\epsilon}{\sigma_\epsilon^2}\right\}\times \\ & \times\frac{1}{|\mathbf{Q}|^{1/2}}\exp\left\{-\frac{1}{2}(\mathbf{y}_h-\rho_0\mathbf{y}_{l_1}-\delta_0\mathbf{1}_{n_1})^T\mathbf{Q}^{-1}(\mathbf{y}_h-\rho_0\mathbf{y}_{l_1}-\delta_0\mathbf{1}_{n_1})\right\}\times \\ & \times\frac{1}{(\sigma_l^2)^{n/2}|\mathbf{R}_l|^{1/2}}\exp\left\{-\frac{1}{2\mathrm{v}_l\sigma_l^2}(\mathbf{y}_l-\mathbf{F}\boldsymbol{\beta}_l)^T\mathbf{R}_l^{-1}(\mathbf{y}_l-\mathbf{F}\boldsymbol{\beta}_l)\right\} \end{aligned} \tag{2.18}$$

where $\mathbf{Q}=\sigma_\rho^2\mathbf{A}_1\mathbf{R}_\rho\mathbf{A}_1+\sigma_\delta^2\mathbf{R}_\delta+\sigma_\epsilon^2\mathbf{I}_{(k+1)\times(k+1)}$ and $A_1=\mathrm{diag}\{\mathrm{y}_{l_1}(\mathbf{x}_1),...,\mathrm{y}_{l_1}(\mathbf{x}_{n_1})\}$. See Appendix C.2 for intermediate computations.

Again posterior parameter sampling is not easy but at least it requires reasonable computational times with the empirical approach.
The posterior sampling is carried out using a Monte Carlo Markov Chain algorithm.

2.3 Posterior sampling

In the MCMC framework the posterior distribution we want to sample from, is the target distribution.
Our target (2.18) has quite a complicated form and depends on seven unknown parameters $(\boldsymbol{\beta}_l,\rho_0,\delta_0,\sigma_l^2,\sigma_\rho^2,\sigma_\delta^2,\sigma_\epsilon^2)$. The BHGP model formulation combined with the optimal selection of prior distributions, makes the Gibbs sampling algorithm the most appealing choice. As a matter of fact, since the dependencies on the

correlation parameters $\boldsymbol{\theta}_3$ are omitted, splitting the high-dimensional posterior distribution into a sequence of full conditional distributions is quite an easy task. Such full conditional distributions are all univariate but one, and they are mostly easy to sample from .

Before proceeding with the computation of the full conditional distribution that will be used in the Gibbs sampler, [QW08] suggests to introduce a new parametrization of the variance parameters vector $\boldsymbol{\theta}_2 = (\sigma_l^2, \sigma_\rho^2, \sigma_\delta^2, \sigma_\epsilon^2)$:

$$(\sigma_l^2, \sigma_\rho^2, \tau_1, \tau_2) = (\sigma_l^2, \sigma_\rho^2, \frac{\sigma_\delta^2}{\sigma_\rho^2}, \frac{\sigma_\epsilon^2}{\sigma_\rho^2}) \tag{2.19}$$

We still refer to $(\sigma_l^2, \sigma_\rho^2, \tau_1, \tau_2)$ as $\boldsymbol{\theta}_2$.
This particular parametrization has the advantage of easing the sampling of σ_ρ^2 from its full conditional distribution. In fact the Bayesian computations lead us to a set of full conditional distributions belonging to known distribution families for both σ_l^2 and σ_ρ^2. If we used the original parametrization for $\boldsymbol{\theta}_2$ we would get only one known-form full conditional distribution for σ_l^2 (see appendix for a better explanation).

Using the change of variables rule described in Appendix B.2 we compute the joint posterior distribution with the new parametrization:

$$\begin{aligned}
&p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2|\mathbf{y}_l,\mathbf{y}_h) \propto \\
&\qquad \frac{1}{(\mathrm{v}_l\sigma_l^2)^{n/2}} \exp\left\{-\frac{1}{2\mathrm{v}_l\sigma_l^2}(\boldsymbol{\beta}_l - \mathbf{u}_l)^T \mathbf{R}_l^{-1}(\boldsymbol{\beta}_l - \mathbf{u}_l)\right\} \times \\
&\qquad \times \frac{1}{(\mathrm{v}_\rho\sigma_\rho^2)^{1/2}} \exp\left\{-\frac{(\rho_0 - \mathrm{u}_\rho)^2}{2\mathrm{v}_\rho\sigma_\rho^2}\right\} \cdot \frac{1}{(\mathrm{v}_\delta\tau_1\sigma_\rho^2)^{1/2}} \exp\left\{-\frac{(\delta_0 - \mathrm{u}_\delta)^2}{2\mathrm{v}_\rho\tau_1\sigma_\rho^2}\right\} \times \\
&\qquad \times (\sigma_l^2)^{-(\alpha_l+1)} \exp\left\{-\frac{\gamma_l}{\sigma_l^2}\right\} \cdot (\sigma_\rho^2)^{-(\alpha_\rho+1)} \exp\left\{-\frac{\gamma_\rho}{\sigma_\rho^2}\right\} \times \\
&\qquad \times (\tau_1\sigma_\rho^2)^{-(\alpha_\delta+1)} \exp\left\{-\frac{\gamma_\delta}{\tau_1\sigma_\rho^2}\right\} \cdot (\tau_2\sigma_\rho^2)^{-(\alpha_\epsilon+1)} \exp\left\{-\frac{\gamma_\epsilon}{\tau_2\sigma_\rho^2}\right\} \cdot \sigma_\rho^4 \times \\
&\qquad \times \frac{1}{(\sigma_\rho^2)^{n_1/2}|\mathbf{M}|^{1/2}} \times \\
&\qquad \times \exp\left\{-\frac{1}{2\mathrm{v}_\rho\sigma_\rho^2}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})^T \mathbf{M}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})\right\} \times \\
&\qquad \times \frac{1}{(\sigma_l^2)^{n/2}|\mathbf{R}_l|^{1/2}} \exp\left\{-\frac{1}{2\mathrm{v}_l\sigma_l^2}(\mathbf{y}_l - \mathbf{F}\boldsymbol{\beta}_l)^T \mathbf{R}_l^{-1}(\mathbf{y}_l - \mathbf{F}\boldsymbol{\beta}_l)\right\}
\end{aligned} \tag{2.20}$$

where $\mathbf{M} = \mathbf{W}_\rho + \tau_1\mathbf{R}_\delta + \tau_2\mathbf{I}_{n_1\times n_1}$, $\mathbf{W}_\rho = \mathbf{A}_1\mathbf{R}_\rho\mathbf{A}_1$, $\mathbf{A}_1 = \mathrm{diag}\{\mathrm{y}_l(\mathbf{x}_1), ..., \mathrm{y}_l(\mathbf{x}_{n_1})\}$ and the correlation matrices $\mathbf{R}_l$, $\mathbf{R}_\rho$ and $\mathbf{R}_\delta$ depend respectively on the previously estimated correlation parameters $\hat{\boldsymbol{\phi}}_l$, $\hat{\boldsymbol{\phi}}_\rho$ and $\hat{\boldsymbol{\phi}}_\delta$.
See Appendix C.2 for intermediate computations.

The full conditional distributions for the unknown parameters are:

$$\begin{aligned}\boldsymbol{\beta}_l|\mathbf{y}_h, \mathbf{y}_l, \overline{\boldsymbol{\beta}_l} \sim N\Bigg(&\left[\frac{1}{\mathrm{v}_l}\mathbf{I}_{(k+1)\times(k+1)} + \mathbf{F}_l^T\mathbf{R}_l^{-1}\mathbf{F}_l\right]^{-1}\left(\frac{\mathbf{u}_l}{\mathrm{v}_l} + \mathbf{F}_l^T\mathbf{R}_l^{-1}\mathbf{y}_l\right), \\ &\left[\frac{1}{\mathrm{v}_l}\mathbf{I}_{(k+1)\times(k+1)} + \mathbf{F}_l^T\mathbf{R}_l^{-1}\mathbf{F}_l\right]^{-1}\sigma_l^2\Bigg)\end{aligned} \tag{2.21}$$

$$\rho_0|\mathbf{y}_h, \mathbf{y}_l, \overline{\rho_0} \sim N\left(\frac{\frac{\mathrm{u}_\rho}{\mathrm{v}_\rho} + \mathbf{y}_{l_1}^T\mathbf{M}^{-1}(\mathbf{y}_h - \delta_0\mathbf{1}_{n_1})}{\frac{1}{\mathrm{v}_\rho} + \mathbf{y}_{l_1}^T\mathbf{M}^{-1}\mathbf{y}_{l_1}}, \frac{\sigma_\rho^2}{\frac{1}{\mathrm{v}_\rho} + \mathbf{y}_{l_1}^T\mathbf{M}^{-1}\mathbf{y}_{l_1}}\right) \tag{2.22}$$

$$\delta_0|\mathbf{y}_h, \mathbf{y}_l, \overline{\delta_0} \sim N\left(\frac{\frac{\mathrm{u}_\delta}{\mathrm{v}_\delta\tau_1} + \mathbf{1}_{n_1}^T\mathbf{M}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1})}{\frac{1}{\mathrm{v}_\delta\tau_1} + \mathbf{1}_{n_1}^T\mathbf{M}^{-1}\mathbf{1}_{n_1}}, \frac{\sigma_\rho^2}{\frac{1}{\mathrm{v}_\delta\tau_1} + \mathbf{1}_{n_1}^T\mathbf{M}^{-1}\mathbf{1}_{n_1}}\right) \tag{2.23}$$

$$\begin{aligned}\sigma_l^2|\mathbf{y}_h, \mathbf{y}_l, \overline{\sigma_l^2} \sim IG\Bigg(&\frac{n}{2} + \frac{k+1}{2} + \alpha_l, \\ &\frac{1}{2}\frac{(\boldsymbol{\beta}_l - \mathbf{u}_l)^T(\boldsymbol{\beta}_l - \mathbf{u}_l)}{\mathrm{v}_l} + \frac{1}{2}(\mathbf{y}_l - \mathbf{F}_l\boldsymbol{\beta}_l)'\mathbf{R}_l^{-1}(\mathbf{y}_l - \mathbf{F}_l\boldsymbol{\beta}_l) + \gamma_l\Bigg)\end{aligned} \tag{2.24}$$

$$\begin{aligned}\sigma_\rho^2|\mathbf{y}_h, \mathbf{y}_l, \overline{\sigma_\rho^2} \sim IG\Bigg(&\frac{n_1}{2} + 1 + \alpha_\rho + \alpha_\delta + \alpha_\epsilon, \frac{(\rho_0 - \mathrm{u}_\rho)^2}{2\mathrm{v}_\rho} + \frac{(\delta_0 - \mathrm{u}_\delta)^2}{2\mathrm{v}_\delta\tau_1} + \\ &+ \gamma_\rho + \frac{\gamma_\delta}{\tau_1} + \frac{\gamma_\epsilon}{\tau_2} + \frac{1}{2}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})'\mathbf{M}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})\Bigg)\end{aligned} \tag{2.25}$$

$$\begin{aligned}&p(\tau_1, \tau_2|\mathbf{y}_h, \mathbf{y}_l, \overline{\tau_1, \tau_2}) \\ &\quad\propto \tau_1^{-(\alpha_\delta+3/2)}\tau_2^{-(\alpha_\epsilon+1)}\exp\left(-\frac{1}{\tau_1}\left[\frac{\gamma_\delta}{\sigma_\rho^2} + \frac{(\delta_0 - \mathrm{u}_\delta)^2}{2v\delta\sigma_\rho^2}\right] - \frac{\gamma_\epsilon}{(\sigma_\rho^2\tau_2)}\right) \times \\ &\quad\times\frac{1}{|\mathbf{M}|^{1/2}}\exp\left(-\frac{1}{2\sigma_\rho^2}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})^T\mathbf{M}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})\right)\end{aligned} \tag{2.26}$$

We point out that $\overline{\omega}$ represents all of the components of $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ except for ω. The demonstrations are provided in Appendix C.3.

As above mentioned, the full conditional distributions from (2.21) to (2.25) have a known form and the sequential sampling required by the Gibbs sampler can be carried out easily. On the contrary we cannot directly draw samples of (τ_1, τ_2) from (2.26) because its form does not relate to any known distribution density function.

In order to sample from such a full conditional we need to use the Metropolis-within-Gibbs algorithm. This means that one draw of (τ_1, τ_2) using the Metropolis algorithm is included in sampling sequence of the Gibbs algorithm.

(τ_1, τ_2) are defined as the ratio of two variances, respectively $\frac{\sigma^2_\delta}{\sigma^2_\rho}$ and $\frac{\sigma^2_\epsilon}{\sigma^2_\rho}$, therefore they are positive-valued. The traditional random-walk Metropolis algorithm uses a normal proposal distribution with mean equal to the sampled valued of the previous iteration and covariance matrix Σ suitably chosen. In our case we have to sample two positive values at each iteration and a normal proposal distribution is not adequate, since it could return negative values for τ_1 and τ_2. To overcome this problem the traditional random-walk Metropolis algorithm needs to be modified in order to provide positive samples of (τ_1, τ_2) at each iteration. Further details on the modified random-walk Metropolis algorithm are provided in Appendix A.2.3.

2.4 Posterior predictive distribution when $\mathbf{x}_0 \in D_l \setminus D_h$

Once the posterior sampling of the unknown parameters $(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$ is carried out, the posterior predictive distribution (2.16) of $\mathrm{y}_h(\mathbf{x}_0)$ is approximated using the Monte Carlo mixture (2.17):

$$\hat{p}_m(\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h) = \frac{1}{M}\sum_{i=1}^{M} p\left(\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h, \boldsymbol{\theta}_1^{(i)}, \boldsymbol{\theta}_2^{(i)}\right). \tag{2.27}$$

The posterior distribution of $\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h, \boldsymbol{\theta}_1^{(i)}, \boldsymbol{\theta}_2^{(i)}$ can be easily computed if we observe that the joint posterior distribution of $(\mathrm{y}(\mathbf{x}_0), \mathbf{y}_h)$ is:

$$\mathrm{y}_h(\mathbf{x}_0), \mathbf{y}_h|\mathbf{y}_l, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2 \sim N(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*). \tag{2.28}$$

where its mean vector and covariance matrix are:

$$\boldsymbol{\mu}^* = \rho_0 \mathbf{y}_l^* + \delta_0 \mathbf{1}_{n_1+1} = \begin{bmatrix} \rho_0 \mathrm{y}_l^* + \delta_0 \\ \rho_0 \mathbf{y}_{l_1} + \delta_0 \mathbf{1}_{n_1} \end{bmatrix} = \begin{bmatrix} \mu_1^* \\ \boldsymbol{\mu}_2 \end{bmatrix} \begin{matrix} (1 \times 1) \\ (n_1 \times n_1) \end{matrix} \tag{2.29}$$

$$\begin{aligned}\boldsymbol{\Sigma}^* = \sigma_\rho^2\mathbf{M} = \sigma_\rho^2(\mathbf{A}_1^*\mathbf{R}_\rho^*\mathbf{A}_1^* + \tau_1\mathbf{R}_\delta^* + \tau_2\mathbf{I}_{(n_1+1)\times(n_1+1)}) = \\ = \begin{bmatrix} \sigma_\rho^2[(\mathrm{y}_l^*)^2 + \tau_1 + \tau_2] & \sigma_\rho^2(\mathrm{y}_l^*\mathbf{A}_1\mathbf{r}_\rho + \tau_1\mathbf{r}_\delta)^T \\ \sigma_\rho^2(\mathrm{y}_l^*\mathbf{A}_1\mathbf{r}_\rho + \tau_1\mathbf{r}_\delta) & \sigma_\rho^2\mathbf{M} \end{bmatrix} = \\ = \begin{bmatrix} \Sigma_{11} & \boldsymbol{\Sigma}_{21}^T \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{bmatrix} \begin{array}{cc} (1\times 1) & (1\times n_1) \\ (n_1\times 1) & (n_1\times 1) \end{array}\end{aligned} \tag{2.30}$$

with $\mathrm{y}_l^* = \mathrm{y}_l(\mathbf{x}_0)$, $\mathbf{y}_{l_1}^* = (\mathrm{y}_l(\mathbf{x}_0), \mathrm{y}_l(\mathbf{x}_1), ..., \mathrm{y}_l(\mathbf{x}_{n_1}))$, $\mathbf{r}_\cdot = (\mathbf{R}_\cdot(\mathbf{x}_0 - \mathbf{x}_1), ..., \mathbf{R}_\cdot(\mathbf{x}_0 - \mathbf{x}_{n_1}))^T$, $A_1^* = \mathrm{diag}\{\mathrm{y}_l^*, \mathrm{y}_{l_1}, ..., \mathrm{y}_{l_{n_1}}\}$ and matrices $\mathbf{R}_\rho$ and $\mathbf{R}_\delta$ are the correlation matrices computed at the input set expanded with the new point $\mathbf{x}_0$.
Using the multivariate normal conditional distribution (see Appendix B.1) we get the following distribution for $\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h$:

$$\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_h, \mathbf{y}_l, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2 \sim N(\mu_{pr}, \Sigma_{pr})$$

where:

$$\begin{aligned}\mu_{pr} =& \mu_1^* + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\mathbf{y}_h - \boldsymbol{\mu}_2) = \\ =& \rho_0\mathrm{y}_l^* + \delta_0 + (\mathrm{y}_l^*\mathbf{A}_1\mathbf{r}_\rho + \tau_1\mathbf{r}_\delta)^T\mathbf{M}^{-1}(\mathrm{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})\end{aligned}$$

$$\begin{aligned}\sigma_{pr}^2 =& \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21} = \\ =& \sigma_\rho^2\left\{[(\mathrm{y}_l^*)^2 + \tau_1 + \tau_2] - (\mathrm{y}_l^*\mathbf{A}_1\mathbf{r}_\rho + \sigma_\delta^2\mathbf{r}_\delta)^T\mathbf{M}^{-1}(\mathrm{y}_l^*\mathbf{A}_1\mathbf{r}_\rho + \sigma_\delta^2\mathbf{r}_\delta)\right\}\end{aligned}$$

The point predictor for the response of interest $\mathrm{y}_h(\cdot)$ evaluated in $\mathbf{x}_0$ is:

$$\hat{\mathrm{y}}_h(\mathbf{x}_0) = E[\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h] = \mu_{pr} \tag{2.31}$$

and it is approximated by the *mixture estimator*:

$$\begin{aligned}\hat{\mathrm{y}}_h(\mathbf{x}_0) \simeq& \frac{1}{M}\sum_{i=1}^{M}\mu_{pred}^{(i)} = \\ =& \frac{1}{M}\sum_{i=1}^{M}\left(\rho_0^{(i)}\mathrm{y}_l^* + \delta_0^{(i)} + (\mathrm{y}_l^*\mathbf{A}_1\mathbf{r}_\rho + \tau_1^{(i)}\mathbf{r}_\delta)^T\mathbf{M}^{-1}(\mathrm{y}_h - \rho_0^{(i)}\mathbf{y}_{l_1} - \delta_0^{(i)}\mathbf{1}_{n_1})\right).\end{aligned} \tag{2.32}$$

If we are interested in predicting $\mathrm{y}_h(\cdot)$ at a set of m new input points $\{\mathbf{x}_{01}, ..., \mathbf{x}_{0m}\}$ we could run m times the two-step MCMC algorithm previously described, but it would be computationally demanding. Alternatively we could also carry out the joint prediction of $\mathbf{y}_h^* = (\mathrm{y}_h(\mathbf{x}_{01}), ..., \mathrm{y}_h(\mathbf{x}_{0m}))^T$. The last procedure provides

predictions as realizations from the same estimated predictive density for the response variable $\mathrm{y}_h(\cdot)$ [BCG04].
In the last case the posterior predictive distribution:

$$\begin{aligned} p(\mathbf{y}_h^*|\mathbf{y}_h, \mathbf{y}_l) &= \int_{\boldsymbol{\theta}_1,\boldsymbol{\theta}_2} p(\mathbf{y}_h^*, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2|\mathbf{y}_h, \mathbf{y}_l)\, d\boldsymbol{\theta}_1 d\boldsymbol{\theta}_2 \\ &= \int_{\boldsymbol{\theta}_1,\boldsymbol{\theta}_2} p(\mathbf{y}_h^*|\mathbf{y}_h, \mathbf{y}_l, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2)\times \\ &\qquad \times p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2|\mathbf{y}_h, \mathbf{y}_l)\, d\boldsymbol{\theta}_1 d\boldsymbol{\theta}_2 \end{aligned} \tag{2.33}$$

is approximated with:

$$\hat{p}_m(\mathbf{y}_h^*|\mathbf{y}_l, \mathbf{y}_h) = \frac{1}{M}\sum_{i=1}^{M} p\left(\mathbf{y}_h^*|\mathbf{y}_l, \mathbf{y}_h, \boldsymbol{\theta}_1^{(i)}, \boldsymbol{\theta}_2^{(i)}\right). \tag{2.34}$$

Such posterior predictive density is computed using the very same procedure previously described in this section.

2.5 Relaxation of the assumption $\mathbf{x}_0 \in D_l \setminus D_h$

So far we have considered the situation in which we want to predict $\mathrm{y}_h(\cdot)$ at a new point $\mathbf{x}_0$ when the corresponding low resolution data is available, i.e. $\mathbf{x}_0 \notin D_l$. When this assumption is relaxed $\mathrm{y}_l(\mathbf{x}_0)$ is not observed. In this case two different approaches can be used to compute the predictive density.

We could use as a predictor for $\mathrm{y}_l(\mathbf{x}_0)$ the expected value of $\mathrm{y}_l(\mathbf{x}_0)|\mathbf{y}_l$, where $\mathrm{y}_l(\mathbf{x}_0)|\mathbf{y}_l$ follows the noncentral t predictive distribution (1.9) computed from the GP model for the LE data.
Once the point prediction $\hat{\mathrm{y}}_l(\mathbf{x}_0) = E[\mathrm{y}_l(\mathbf{x}_0)|\mathbf{y}_l]$ is computed, we add $\hat{\mathrm{y}}_l(\mathbf{x}_0)$ to the set of the $\mathbf{y}_l$ so that $\mathbf{x}_0 \in D_l \cup \{\mathbf{x}_0\}$.

Following another suggested approach, the predictive density $p(\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_h, \mathbf{y}_l)$ is approximated by:

$$\hat{p}_m(\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h) = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N} p\left(\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_l^{*(j)}, \mathbf{y}_l, \mathbf{y}_h, \boldsymbol{\theta}_1^{(i)}, \boldsymbol{\theta}_2^{(i)}\right) \tag{2.35}$$

where $\mathbf{y}_l^{*(j)} = (\mathrm{y}_l^{(j)}(\mathbf{x}_0), \mathrm{y}_l(\mathbf{x}_1), ..., \mathrm{y}_l(\mathbf{x}_{n_1}))^T$, $j = 1, ..., N$, and $\mathrm{y}_l^{(j)}(\mathbf{x}_0)$ are N independent draws from the distribution $p(\mathrm{y}_l(\mathbf{x}_0)|\mathbf{y}_l, \boldsymbol{\theta}_1^{(i)}, \boldsymbol{\theta}_2^{(i)})$, $i = 1, ..., M$.

In order to compute the conditional distribution $p(\mathrm{y}_l(\mathbf{x}_0)|\mathbf{y}_l, \boldsymbol{\theta}_1^{(i)}, \boldsymbol{\theta}_2^{(i)})$, $i = 1, ..., M$, we use the multivariate normal conditional distribution (Appendix B.1). We observe that:

$$p(\mathrm{y}_l(\mathbf{x}_0), \mathbf{y}_l|\boldsymbol{\theta}_1^{(i)}, \boldsymbol{\theta}_2^{(i)}) \sim N(\boldsymbol{\mu}_l^*, \boldsymbol{\Sigma}_l^*) \tag{2.36}$$

where the mean vector and the covariance matrix are respectively:

$$\boldsymbol{\mu}_l^* = \begin{pmatrix} \mu_{l1} \\ \boldsymbol{\mu}_{l2} \end{pmatrix} = \begin{pmatrix} \boldsymbol{f}_l^T(\mathbf{x}_0)\boldsymbol{\beta}_l^{(i)} \\ \mathbf{F}_l\boldsymbol{\beta}_l^{(i)} \end{pmatrix}$$

$$\boldsymbol{\Sigma}_l^* = \begin{bmatrix} \Sigma_{l11} & \Sigma_{l12} \\ \Sigma_{l21} & \Sigma_{l22} \end{bmatrix} = \begin{bmatrix} \sigma_l^2 & \sigma_l^2\mathbf{r}_l^T \\ \sigma_l^2\mathbf{r}_l & \sigma_l^2\mathbf{R}_l \end{bmatrix}$$

with $\mathbf{r}_l = (\mathbf{R}_l(\mathbf{x}_0 - \mathbf{x}_1), ..., \mathbf{R}_l(\mathbf{x}_0 - \mathbf{x}_{n_1}))^T$.
It follows that:

$$\begin{aligned} \mathrm{y}_l(\mathbf{x}_0)|\mathbf{y}_l, \boldsymbol{\theta}_1^{(i)}, \boldsymbol{\theta}_2^{(i)} &\sim N\left(\mu_1 + \Sigma_{21}^T\Sigma_{22}^{-1}(\mathbf{y}_l - \boldsymbol{\mu}_2), \Sigma_{11} - \Sigma_{21}^T\Sigma_{22}^{-1}\Sigma_{21}\right) \\ &= N\left(\mathbf{f}_l^T(\mathbf{x}_0)\boldsymbol{\beta}_l^{(i)} + \mathbf{r}_l^T\mathbf{R}_l^{-1}(\mathbf{y}_l - \mathbf{F}_l\boldsymbol{\beta}_l^{(i)}), \sigma_l^{2(i)}(1 - \mathbf{r}_l^T\mathbf{R}_l^{-1}\mathbf{r}_l)\right). \end{aligned} \tag{2.37}$$

2.6 Adaptation of the BHGP model to multi-resolution metrology data

The BHGP model previously introduced is meant to be used in case the low-accuracy data come from a computer experiment. Thus the model that describes the LE data is deterministic, i.e. it does not include a random error term.
When analyzing multi-resolution data coming from physical experiments, a more general form of the model described in [QW08] is needed. In particular, when the low-accuracy experiment is not deterministic, a measurement error term has to be introduced in the model for the LE data.

2.6.1 Introducing the measurement error in the LE response

When both the LE and HE responses come from actual measurement activities, both $\mathrm{y}_l(\cdot)$ and $\mathrm{y}_h(\cdot)$ are affected by measurement error. Such measurement error can be modeled with an i.i.d. random error. If we want to use the BHGP model, we need to correct it by introducing a random error term in the model for the low-resolution data. We call such error term $\eta(\cdot)$ and we assume that it is a white

noise.
Equation (1.5) previously introduced has to be modified as follows:

$$\mathrm{y}_l(\mathbf{x}_i) = \boldsymbol{f}_l^T(\mathbf{x}_i)\boldsymbol{\beta}_l + \epsilon_l(\mathbf{x}_i) + \eta(\mathbf{x}_i) \quad i = 1, ..., n, \tag{2.38}$$

where still $\boldsymbol{f}_l^T(\mathbf{x}_i) = (1, x_{i1}, ..., x_{ik})^T$, $\boldsymbol{\beta}_l = (\beta_{l0}, \beta_{l1}, ..., \beta_{lk})^T$ and $\epsilon_l(\cdot) \sim GP(0, \sigma_l^2, \boldsymbol{\phi}_l)$. We assume that the GP term is independent from the measurement error $\eta(\cdot)$ and that $\eta(\cdot) \sim N(0, \sigma_\eta^2)$ such that $\eta(\mathbf{x}_u) \perp \eta(\mathbf{x}_w), \forall u \neq w$.
The adjustment model that links the HE data to the LE data is the same as the one illustrated in Equation (1.15):

$$\mathrm{y}_h(\mathbf{x}_i) = \rho(\mathbf{x}_i)\mathrm{y}_{l_{n1}}(\mathbf{x}_i) + \delta(\mathbf{x}_i) + \epsilon(\mathbf{x}_i) \quad i = 1, ..., n_1, \tag{2.39}$$

where $\rho(\cdot) \sim GP(\rho_0, \sigma_\rho^2, \boldsymbol{\phi}_\rho)$, $\delta(\cdot) \sim GP(\delta_0, \sigma_\delta^2, \boldsymbol{\phi}_\delta)$ and the measurement error $\epsilon(\cdot) \sim N(0, \sigma_\varepsilon^2)$. $\mathrm{y}_{l_{n1}}(\cdot)$, $\rho(\cdot)$, $\delta(\cdot)$ and $\epsilon_l(\cdot)$ are assumed independent one another.

The model described in Equations (2.38) and (2.39) involves a new unknown parameter, i.e. the variance σ_η^2 of the measurement error $\eta(\cdot)$.
The new model is complete once the prior for σ_η^2 is selected. Consistently with the prior distributions chosen by [QW08] for the other variance parameters, we select the following prior for σ_η^2:

$$\sigma_\eta^2 \sim IG(\alpha_\eta, \gamma_\eta).$$

2.6.2 Estimation of the predictive density in the non-deterministic case

In order to estimate the predictive density in Equation (2.1) we still refer to the procedure illustrated in Section 2.2.3. We remind that this procedure approximates the predictive density for high-resolution data at untried points using a MCMC algorithm in an empirical bayesian framework.

The first step is the estimation of the correlation parameters $\boldsymbol{\phi}_l$, $\boldsymbol{\phi}_\rho$ and $\boldsymbol{\phi}_\delta$ of the three Gaussian processes involved in the model.
In order to take into account the measurement error in the LE data, optimization

Problem (2.9) must be adjusted as follows:

$$\begin{aligned} \hat{\boldsymbol{\phi}}_l = \arg\max_{\boldsymbol{\phi}_l} p(\boldsymbol{\phi}_l) \int_{\tau_3} (\tau_3)^{-(\alpha_\eta+1)} |\mathbf{M}_2|^{-1/2} |\mathbf{H}_1|^{-1/2} \\ (\gamma_l + \frac{\gamma_\eta}{\tau_3} + \frac{1}{2}(-\mathbf{b}_1^T \mathbf{H}_1 \mathbf{b}_1 + c_1))^{-(\alpha_l+\alpha_\eta+\frac{n}{2})} \, d\tau_3 \\ \text{s.t. } \boldsymbol{\phi}_l > \mathbf{0}, \end{aligned} \tag{2.40}$$

where:

$$\begin{aligned} \tau_3 &= \frac{\sigma_\eta^2}{\sigma_l^2}, \\ \mathbf{H}_1 &= \frac{1}{\mathrm{v}_l} + \mathbf{F}_l^T \mathbf{M}_2^{-1} \mathbf{F}_l, \\ \mathbf{b}_1 &= \frac{\mathbf{u}_l}{\mathrm{v}_l} + \mathbf{F}_l^T \mathbf{M}_2^{-1} \mathbf{y}_l, \\ c_1 &= \frac{\mathbf{u}_l^T \mathbf{u}_l}{\mathrm{v}_l} + \mathbf{y}_l^T \mathbf{M}_2^{-1} \mathbf{y}_l, \\ \mathbf{M}_2 &= \mathbf{R}_l + \frac{\sigma_\eta^2}{\sigma_l^2} \mathbf{I}_{n\times n}. \end{aligned} \tag{2.41}$$

Using the SAA approximation discussed above, Problem (2.40) ca be recast as:

$$\begin{aligned} \hat{\boldsymbol{\phi}}_l = \arg\max_{\boldsymbol{\phi}_l} \frac{1}{S} \sum_{s=1}^{S} f(\tau_3^{(s)}) \\ \text{s.t. } \boldsymbol{\phi}_l > \mathbf{0} \end{aligned} \tag{2.42}$$

where:

$$f(\tau_3) = |\mathbf{M}_2|^{-1/2} |\mathbf{H}_1|^{-1/2} (\gamma_l + \frac{\gamma_\eta}{\tau_3} + \frac{1}{2}(-\mathbf{b}_1^T \mathbf{H}_1 \mathbf{b}_1 + c_1))^{-(\alpha_l+\alpha_\eta+\frac{n}{2})} \exp\left\{\frac{\gamma_3}{\tau_3}\right\}, \tag{2.43}$$

and $\tau_3^{(s)}$, $s = 1, ..., S$, are samples drawn from an Inverse Gamma distribution $IG(\alpha_\eta, \gamma_3)$.

The steps to obtain the objective function in (2.40) are very similar to the ones that led to Equation (2.7) and they will be omitted.

The remaining correlation parameters $\boldsymbol{\phi}_\rho$ and $\boldsymbol{\phi}_\delta$ are estimated again by solving the optimization Problem (2.15) with no further adjustment.

Once the posterior estimates of the correlation parameters are available, we treat the correlation parameters as they were given, and the usual Gibbs sampler is used to draw samples of the unknown parameters $(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$ from their posterior distribution.

Again the method described in section 2.3 needs a few simple adjustments in order to include the new variance parameter σ_η^2 in the sampling procedure. The likelihood function for the data can be expressed again as:

$$L(\mathbf{y}_l, \mathbf{y}_h | \boldsymbol{\theta}_1, \boldsymbol{\theta}_2) \propto L(\mathbf{y}_l | \boldsymbol{\theta}_1, \boldsymbol{\theta}_2) L(\mathbf{y}_h | \mathbf{y}_l, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2), \tag{2.44}$$

where:

$$L(\mathbf{y}_l | \boldsymbol{\theta}_1, \boldsymbol{\theta}_2) \propto \frac{1}{|\sigma_l^2 \mathbf{M}_2|^{1/2}} \exp\left\{ -\frac{1}{2\sigma_l^2} (\mathbf{y}_l - \mathbf{F}_l \boldsymbol{\beta}_l)^T \mathbf{M}_2^{-1} (\mathbf{y}_l - \mathbf{F}_l \boldsymbol{\beta}_l) \right\} \tag{2.45}$$

$$\begin{aligned} L(\mathbf{y}_h | \mathbf{y}_l, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2) \propto & \frac{1}{|\sigma_\rho^2 \mathbf{M}_1|^{1/2}} \times \\ & \times \exp\left\{ -\frac{1}{2\sigma_\rho^2} (\mathbf{y}_h - \rho_0 \mathbf{y}_l + \delta_0 \mathbf{1}_{n1})^T \mathbf{M}_1^{-1} (\mathbf{y}_h - \rho_0 \mathbf{y}_l + \delta_0 \mathbf{1}_{n1}) \right\}, \end{aligned}$$

with $\mathbf{M}_2 = \mathbf{R}_l + \frac{\sigma_\eta^2}{\sigma_l^2} \mathbf{I}_{n \times n}$, $\mathbf{M}_1 = A_1 \mathbf{R}_l A_1 + \frac{\sigma_\delta^2}{\sigma_\rho^2} \mathbf{R}_\rho + \frac{\sigma_\epsilon^2}{\sigma_\rho^2} \mathbf{I}_{n_1 \times n_1}$ and $A_1 = \text{diag}\{\mathrm{y}_l(\mathbf{x}_1), ..., \mathrm{y}_l(\mathbf{x}_{n1})\}$.

We introduce again a new parametrization for the variance parameters:

$$(\sigma_l^2, \tau_3, \sigma_\rho^2, \tau_1, \tau_2) = (\sigma_l^2, \frac{\sigma_\eta^2}{\sigma_l^2}, \sigma_\rho^2, \frac{\sigma_\delta^2}{\sigma_\rho^2}, \frac{\sigma_\epsilon^2}{\sigma_\rho^2}).$$

This new parametrization makes the posterior sampling of the variance parameters easier as it allows us to obtain full conditional distributions all belonging to known distribution families except for τ_3, τ_1 and τ_2.

From now on we refer to $\left(\sigma_l^2, \tau_3, \sigma_\rho^2, \tau_1, \tau_3\right)$ as $\boldsymbol{\theta}_2$. Exploiting once again the change of variable rule illustrated in Appendix B.2, the following joint posterior distribution for $(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$ with the new parametrization is computed:

$$
\begin{aligned}
p(\boldsymbol{\theta}_1,&\boldsymbol{\theta}_2|\mathbf{y}_h, \mathbf{y}_l) = p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) L(\mathbf{y}_h, \mathbf{y}_l|\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) \\
&\propto (\sigma_l^2)^{-(\alpha_l+1)} \exp\left\{-\frac{\gamma_l}{\sigma_l^2}\right\} \cdot (\sigma_l^2\tau_3)^{-(\alpha_\eta+1)} \exp\left\{-\frac{\gamma_\eta}{\sigma_l^2\tau_3}\right\} \cdot \sigma_l^2 \times \\
&\quad \times (\sigma_\rho^2\tau_1)^{-(\alpha_\delta+1)} \exp\left\{-\frac{\gamma_\delta}{(\sigma_\rho^2\tau_1)}\right\} \cdot (\sigma_\rho^2\tau_2)^{-(\alpha_\epsilon+1)} \exp\left\{-\frac{\gamma_\epsilon}{(\sigma_\rho^2\tau_2)}\right\} \times \\
&\quad \times (\sigma_\rho^2)^{-(\alpha_\rho+1)} \exp\left\{-\frac{\gamma_\rho}{\sigma_\rho^2}\right\} \cdot \sigma_\rho^4 \times \\
&\quad \times (\sigma_l^2)^{-\frac{k+1}{2}} \exp\left\{-\frac{1}{2v_l\sigma_l^2}(\boldsymbol{\beta}_l - \mathbf{u}_l)^T(\boldsymbol{\beta}_l - \mathbf{u}_l)\right\} \times \\
&\quad \times (\sigma_\rho^2)^{-\frac{1}{2}} \exp\left\{\frac{1}{2v_\rho\sigma_\rho^2}(\rho_0 - u_\rho)^2\right\} \cdot (\sigma_\rho^2\tau_1)^{-\frac{1}{2}} \exp\left\{-\frac{1}{2v_\delta\sigma_\rho^2\tau_1}(\delta_0 - u_\delta)^2\right\} \times \\
&\quad \times \frac{1}{|\sigma_\rho^2\mathbf{M}_1|^{1/2}} \exp\left\{-\frac{1}{2\sigma_\rho^2}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})^T\mathbf{M}_1^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})\right\} \times \\
&\quad \times \frac{1}{|\sigma_l^2\mathbf{M}_2|^{1/2}} \exp\left\{-\frac{1}{2\sigma_l^2}(\mathbf{y}_l - \mathbf{F}_l\boldsymbol{\beta}_l)^T\mathbf{M}_2^{-1}(\mathbf{y}_l - \mathbf{F}_l\boldsymbol{\beta}_l)\right\}.
\end{aligned}
\tag{2.46}
$$

In order to implement the Gibbs algorithm for sampling from such joint posterior distribution we need to compute the full conditional distributions for all the unknown parameters involved:

$$
\begin{aligned}
\boldsymbol{\beta}_l|\mathbf{y}_l, \mathbf{y}_h, \overline{\boldsymbol{\beta}_l} \sim N\Bigg(&\left(\frac{1}{v_l}\mathbf{I}_{(k+1)\times(k+1)} + \mathbf{F}_l^T\mathbf{M}_2^{-1}\mathbf{F}_l\right)^{-1}\left(\frac{\mathbf{u}_l}{v_l} + \mathbf{F}_l^T\mathbf{M}_2^{-1}\mathbf{y}_l\right), \\
&\left(\frac{1}{v_l}\mathbf{I}_{(k+1)\times(k+1)} + \mathbf{F}_l^T\mathbf{M}_2^{-1}\mathbf{F}_l\right)^{-1}\sigma_l^2\Bigg)
\end{aligned}
\tag{2.47}
$$

$$
\rho_0|\mathbf{y}_h, \mathbf{y}_l, \overline{\rho_0} \sim N\left(\frac{\frac{u_\rho}{v_\rho} + \mathbf{y}_{l_1}^T\mathbf{M}_1^{-1}(\mathbf{y}_h - \delta_0\mathbf{1}_{n_1})}{\frac{1}{v_\rho} + \mathbf{y}_{l_1}^T\mathbf{M}_1^{-1}\mathbf{y}_{l_1}}, \frac{\sigma_\rho^2}{\frac{1}{v\rho} + \mathbf{y}_{l_1}^T\mathbf{M}_1^{-1}\mathbf{y}_{l_1}}\right)
\tag{2.48}
$$

$$
\delta_0|\mathbf{y}_h, \mathbf{y}_l, \overline{\delta_0} \sim N\left(\frac{\frac{u_\delta}{v_\delta\tau_1} + \mathbf{1}_{n_1}^T\mathbf{M}_1^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1})}{\frac{1}{v_\delta\tau_1} + \mathbf{1}_{n_1}^T\mathbf{M}_1^{-1}\mathbf{1}_{n_1}}, \frac{\sigma_\rho^2}{\frac{1}{v_\delta\tau_1} + \mathbf{1}_{n_1}^T\mathbf{M}^{-1}\mathbf{1}_{n_1}}\right)
\tag{2.49}
$$

$$
\begin{aligned}
\sigma_l^2|\mathbf{y}_l, \mathbf{y}_h, \overline{\sigma_l^2} \sim IG\Bigg(&\alpha_l + \alpha_\eta + \frac{k+1}{2} + \frac{n}{2}, \\
&\gamma_l + \frac{\gamma_\eta}{\tau_3} + \frac{1}{2v_l}(\boldsymbol{\beta}_l - \mathbf{u}_l)^T(\boldsymbol{\beta}_l - \mathbf{u}_l) + \frac{1}{2}(\mathbf{y}_l - \mathbf{F}_l\boldsymbol{\beta}_l)^T\mathbf{M}_2^{-1}(\mathbf{y}_l - \mathbf{F}_l\boldsymbol{\beta}_l)\Bigg)
\end{aligned}
\tag{2.50}
$$

$$p(\tau_3|\mathbf{y}_l, \mathbf{y}_h, \overline{\tau_3})$$
$$\propto \frac{1}{\tau_3^{(\alpha_\eta+1)}} \frac{1}{|\mathbf{M}_2|^{1/2}} \exp\left\{-\frac{1}{2\sigma_l^2}\left(\frac{\gamma_\eta}{\tau_3} + (\mathbf{y}_l - \mathbf{F}_l\boldsymbol{\beta}_l)^T \mathbf{M}_2^{-1} (\mathbf{y}_l - \mathbf{F}_l\boldsymbol{\beta}_l)\right)\right\} \tag{2.51}$$

$$\begin{aligned}\sigma_\rho^2|\mathbf{y}_h, \mathbf{y}_l, \overline{\sigma_\rho^2} \sim IG\Bigg(&\alpha_\rho + \alpha_\delta + \alpha_\epsilon + \frac{3}{2} + \frac{n_1}{2},\\ &\gamma_\rho + \frac{\gamma_\delta}{\tau_1} + \frac{\gamma_\epsilon}{\tau_2} + \frac{1}{2\sigma_\rho^2}\left(\frac{(\rho_0 - u_\rho)^2}{v_\rho} + \frac{(\delta_0 - u_\delta)^2}{v_\delta\tau_1}\right) + \\ &+ \frac{1}{2\sigma_\rho^2}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})^T\mathbf{M}_1^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})\Bigg)\end{aligned} \tag{2.52}$$

$$\begin{aligned}p(\tau_1, \tau_2|\mathbf{y}_h, \mathbf{y}_l, \overline{\tau_1, \tau_2})&\\ &\propto \tau_1^{-(\alpha_\delta+3/2)}\tau_2^{-(\alpha_\varepsilon+1)}\frac{1}{|\mathbf{M}_1|^{1/2}}\\ &\times \exp\left\{-\frac{\gamma_\delta}{\sigma_\rho^2\tau_1} - \frac{1}{\sigma_\rho^2\tau_2} - \frac{(\delta_0 - u_\delta)^2}{2v_\delta\sigma_\rho^2\tau_1}\right\}\\ &\times \exp\left\{-\frac{1}{2\sigma_\rho^2}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})^T\mathbf{M}_1^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})\right\}\end{aligned} \tag{2.53}$$

The particular form of the full conditional distributions from (2.47) to (2.53) allows us to easily sample all the parameters, except for the variance parameters τ_3 and (τ_1, τ_2). In order to draw samples from the full conditional distributions (2.51) and (2.53) we need to run two distinct Metropolis algorithms at every iteration of the Gibbs sampler. As τ_3 and (τ_1, τ_2) are all defined as ratios of variances, they are positive valued and as we did in Section 2.3, we use the modified version of the random-walk Metropolis algorithm (see Appendix 2.3).

Once a sufficient number of posterior samples of the unknown parameters is drawn from the joint posterior distribution (2.46) we can proceed with the approximation of the predictive distribution of the high resolution response at untried points $\mathbf{x}_0$, $p(\mathrm{y}_h(\mathbf{x}_0)|\mathrm{y}_h, \mathrm{y}_h)$. After applying the appropriate modifications in order to take into account the effect of the measurement error in the LE response, we are allowed to adopt the same procedures seen in Sections 2.4 and 2.5 in the following cases respectively:

- $\mathbf{x}_0 \in D_l \setminus D_h$, i.e. the prediction has to be computed at some input point where the LE response $\mathrm{y}_l(\cdot)$ is available, and the LE and HE responses are available at the same input points;
- $\mathbf{x}_0 \notin D_l \setminus D_h$ but again the LE and HE responses are available at the same

input points.

We do not report the expressions of the approximation of the predictive distributions in these cases because they are mere extensions of the formulas in Section 2.4 and 2.5 to the case where the LE response is affected by measurement error. We rather treat the more general situation in which D_l and D_h are disjoint sets.

2.7 Prediction when D_l and D_h are disjoint

When there is no perfect match between the experimental points of the low-resolution and the high-resolution data and the adjustment model for the HE response (2.39) cannot be directly employed. As a matter of fact Equation (2.39) implies that the set of n_1 low-resolution data $\mathrm{y}_l(\mathbf{x}_i)$ corresponding to the available high-resolution training set $\mathrm{y}_h(\mathbf{x}_i)$ is given.

Two different approaches can be taken to deal with this situation.
We call the first way to deal with the missing low-resolution data $\mathrm{y}_{l_1}(\mathbf{x}_i)$, $i = 1, ..., n_1$, *two-stage approach.* We simply compute a prediction of the LE response at the input points where the training HE data are available and we plug such prediction into the adjustment model (2.39).
We call the second method *data augmentation approach*, as it exploits the data augmentation technique introduced by [TW87]. We augment the set of unknown parameters, by treating the missing LE data corresponding to the high-resolution training set as they were unknown parameters.

2.7.1 Two-stage approach

As mentioned above, we compute a prediction of the LE response $\hat{\mathrm{y}}_{l_1}(\mathbf{x}_i)$ at the input points where the training HE data are available and we plug such prediction into the adjustment model (2.39), that can be rewritten as:

$$\mathrm{y}_h(\mathbf{x}_i) = \rho(\mathbf{x}_i)\hat{\mathrm{y}}_{l_1}(\mathbf{x}_i) + \delta(\mathbf{x}_i) + \epsilon(\mathbf{x}_i) \quad i = 1, ..., n_1. \tag{2.54}$$

To compute the prediction $\hat{\mathrm{y}}_{l_{n1}}(\mathbf{x}_i)$ we use the following posterior predictive distribution:

$$p(\mathrm{y}_{l_1}(\mathbf{x}_0)|\mathbf{y}_l) = \int_{\boldsymbol{\beta}_l, \sigma_l^2, \tau_3} p(\mathrm{y}_{l_1}(\mathbf{x}_0)|\mathbf{y}_l, \boldsymbol{\beta}_l, \sigma_l^2, \tau_3) p(\boldsymbol{\beta}_l, \sigma_l^2, \sigma_\eta^2|\mathbf{y}_l)\, d\boldsymbol{\beta}_l\, d\sigma_l^2\, d\sigma_\eta^2. \tag{2.55}$$

In accordance with the methodology adopted so far, we approximate such predictive distribution with the Monte Carlo mixture:

$$\hat{p}(\mathrm{y}_{l_1}(\mathbf{x}_0)|\mathbf{y}_l) = \frac{1}{M}\sum_{i=1}^{M} p(\mathrm{y}_{l_1}(\mathbf{x}_0)|\mathbf{y}_l, \boldsymbol{\beta}_l^{(i)}, \sigma_l^{2\,(i)}, \sigma_\eta^{2\,(i)}) \tag{2.56}$$

where $(\boldsymbol{\beta}_l^{(i)}, \sigma_l^{2\,(i)}, \sigma_\eta^{2\,(i)})$, $i = 1, ..., M$, are samples drawn from the joint posterior distribution of $(\boldsymbol{\beta}_l, \sigma_l^2, \sigma_\eta^2)$. The joint posterior distribution of such parameters is easily computed as the product of the likelihood function of the low resolution data (2.44) and the prior distributions $p(\boldsymbol{\beta}_l)$, $p(\sigma_l^2)$ and $p(\sigma_\eta^2)$ previously specified. Applying the usual change of variables for the second variance parameters:

$$\tau_3 = \frac{\sigma_l^2}{\sigma_\eta^2} \tag{2.57}$$

we get the following joint posterior distribution:

$$\begin{aligned} p(\boldsymbol{\beta}_l, \sigma_l^2, \sigma_\eta^2|\mathbf{y}_l) \propto\ & (\sigma_l^2)^{-(\alpha_l+1)} \exp\left\{-\frac{\gamma_l}{\sigma_l^2}\right\} \cdot (\sigma_l^2\tau_3)^{-(\alpha_\eta+1)} \exp\left\{-\frac{\gamma_\eta}{\sigma_l^2\tau_3}\right\} \cdot \sigma_l^2 \times \\ & \times (\sigma_l^2)^{-\frac{k+1}{2}} \exp\left\{-\frac{1}{2v_l\sigma_l^2}(\boldsymbol{\beta}_l - \mathbf{u}_l)^T(\boldsymbol{\beta}_l - \mathbf{u}_l)\right\} \times \\ & \times \frac{1}{|\sigma_l^2\mathbf{M}_2|^{1/2}} \exp\left\{-\frac{1}{2\sigma_l^2}(\mathbf{y}_l - \mathbf{F}_l\boldsymbol{\beta}_l)^T\mathbf{M}_2^{-1}(\mathbf{y}_l - \mathbf{F}_l\boldsymbol{\beta}_l)\right\}. \end{aligned} \tag{2.58}$$

In order to compute the approximation (2.56) of the predictive distribution of the LE data, we need to draw a sufficiently large number of samples from (2.58) using the Gibbs algorithm with the full conditional distributions (2.47), (2.50) and (2.51).

Once the prediction $\hat{\mathbf{y}}_l$ is computed, we can evaluate the predictions of $\mathrm{y}_h(\cdot)$ at untried points as done previously, after substituting $\mathbf{y}_{l1}$ with $\hat{\mathbf{y}}_{l1}$.

2.7.2 Data augmentation approach

As suggested in [QW08], if we treat the missing LE data corresponding to the high-resolution training set as they were unknown parameters, in accordance with the data augmentation technique introduced by [TW87], we augment the set of unknown parameters, that becomes $(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \mathbf{y}_{l_1\,mis})$.

The prediction of $\mathrm{y}_h(\cdot)$ at untried points is done through the predictive density

$p(\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h)$, that in this case has the following form:

$$p(\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h) = \int_{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \mathbf{y}_{l_1\,mis}} p(\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \mathbf{y}_{l_1\,mis}) \\ p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \mathbf{y}_{l_1\,mis}|\mathbf{y}_l, \mathbf{y}_h)\, d\boldsymbol{\theta}_1\, d\boldsymbol{\theta}_2\, d\mathbf{y}_{l_1\,mis}. \tag{2.59}$$

Thus the usual empirical Bayesian two-step MCMC algorithm needs to be modified in order to consider $\mathbf{y}_{l_1\,mis}$ as a further vector of unknown parameters.

1. First M posterior samples $\left(\boldsymbol{\theta}_1^{(i)}, \boldsymbol{\theta}_2^{(i)}, \mathbf{y}_{l_1\,mis}^{(i)}\right)$, $i = 1, ..., M$, are drawn (after a properly chosen burn-in period) from the joint posterior distribution of the unknown parameters $p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \mathbf{y}_{l_1\,mis}|\mathbf{y}_l, \mathbf{y}_h)$.

2. Then the predictive Bayesian density $p(\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_h, \mathbf{y}_l)$ is computed as the following Monte Carlo mixture:

$$\hat{p}_m(\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h) = \frac{1}{M}\sum_{i=1}^{M} p\left(\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h, \boldsymbol{\theta}_1^{(i)}, \boldsymbol{\theta}_2^{(i)}, \mathbf{y}_{l_1\,mis}^{(i)}\right). \tag{2.60}$$

The posterior sampling from the joint posterior density of the unknown parameters $p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \mathbf{y}_{l_1\,mis}|\mathbf{y}_l, \mathbf{y}_h)$ is carried out using the usual Gibbs algorithm with the full conditional distributions modified as follows:

$$\boldsymbol{\beta}_l|\mathbf{y}_l, \mathbf{y}_h, \overline{\boldsymbol{\beta}_l}, \mathbf{y}_{l_1\,mis} \sim N\Bigg(\left(\frac{1}{v_l}\mathbf{I}_{(k+1)\times(k+1)} + \mathbf{F}_l^T\mathbf{M}_2^{-1}\mathbf{F}_l\right)^{-1}\left(\frac{\mathbf{u}_l}{v_l} + \mathbf{F}_l^T\mathbf{M}_2^{-1}\mathbf{y}_l\right), \\ \left(\frac{1}{v_l}\mathbf{I}_{(k+1)\times(k+1)} + \mathbf{F}_l^T\mathbf{M}_2^{-1}\mathbf{F}_l\right)^{-1}\sigma_l^2\Bigg) \tag{2.61}$$

$$\rho_0|\mathbf{y}_h, \mathbf{y}_l, \overline{\rho_0}, \mathbf{y}_{l_1\,mis} \sim N\left(\frac{\frac{u_\rho}{v_\rho} + \mathbf{y}_{l_1\,mis}^T\mathbf{M}_1^{-1}(\mathbf{y}_h - \delta_0\mathbf{1}_{n_1})}{\frac{1}{v_\rho} + \mathbf{y}_{l_1\,mis}^T\mathbf{M}_1^{-1}\mathbf{y}_{l_1\,mis}}, \frac{\sigma_\rho^2}{\frac{1}{v\rho} + \mathbf{y}_{l_1\,mis}^T\mathbf{M}_1^{-1}\mathbf{y}_{l_1\,mis}}\right) \tag{2.62}$$

$$\delta_0|\mathbf{y}_h, \mathbf{y}_l, \overline{\delta_0}, \mathbf{y}_{l_1\,mis} \sim N\left(\frac{\frac{u_\delta}{v_\delta\tau_1} + \mathbf{1}_{n_1}^T\mathbf{M}_1^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1\,mis})}{\frac{1}{v_\delta\tau_1} + \mathbf{1}_{n_1}^T\mathbf{M}_1^{-1}\mathbf{1}_{n_1}}, \frac{\sigma_\rho^2}{\frac{1}{v_\delta\tau_1} + \mathbf{1}_{n_1}^T\mathbf{M}^{-1}\mathbf{1}_{n_1}}\right) \tag{2.63}$$

$$\sigma_l^2|\mathbf{y}_l, \mathbf{y}_h, \overline{\sigma_l^2}, \mathbf{y}_{l_1\,mis} \sim IG\Bigg(\alpha_l + \alpha_\eta + \frac{k+1}{2} + \frac{n}{2}, \\ \gamma_l + \frac{\gamma_\eta}{\tau_3} + \frac{1}{2v_l}(\boldsymbol{\beta}_l - \mathbf{u}_l)^T(\boldsymbol{\beta}_l - \mathbf{u}_l) + \frac{1}{2}(\mathbf{y}_l - \mathbf{F}_l\boldsymbol{\beta}_l)^T\mathbf{M}_2^{-1}(\mathbf{y}_l - \mathbf{F}_l\boldsymbol{\beta}_l)\Bigg) \tag{2.64}$$

$$p(\tau_3|\mathbf{y}_l,\mathbf{y}_h,\overline{\tau_3},\mathbf{y}_{l_1\,mis})$$
$$\propto\frac{1}{\tau_3^{(\alpha_\eta+1)}}\frac{1}{|\mathbf{M}_2|^{1/2}}\exp\left\{-\frac{1}{2\sigma_l^2}\left(\frac{\gamma_\eta}{\tau_3}+(\mathbf{y}_l-\mathbf{F}_l\boldsymbol{\beta}_l)^T\,\mathbf{M}_2^{-1}\,(\mathbf{y}_l-\mathbf{F}_l\boldsymbol{\beta}_l)\right)\right\} \tag{2.65}$$

$$\sigma_\rho^2|\mathbf{y}_h,\mathbf{y}_l,\overline{\sigma_\rho^2},\mathbf{y}_{l_1\,mis}\sim IG\Bigg(\alpha_\rho+\alpha_\delta+\alpha_\epsilon+\frac{3}{2}+\frac{n_1}{2},$$
$$\gamma_\rho+\frac{\gamma_\delta}{\tau_1}+\frac{\gamma_\epsilon}{\tau_2}+\frac{1}{2\sigma_\rho^2}\left(\frac{(\rho_0-u_\rho)^2}{v_\rho}+\frac{(\delta_0-u_\delta)^2}{v_\delta\tau_1}\right)+ \tag{2.66}$$
$$+\frac{1}{2\sigma_\rho^2}(\mathbf{y}_h-\rho_0\mathbf{y}_{l_1\,mis}-\delta_0\mathbf{1}_{n_1})^T\mathbf{M}_1^{-1}(\mathbf{y}_h-\rho_0\mathbf{y}_{l_1\,mis}-\delta_0\mathbf{1}_{n_1})\Bigg)$$

$$p(\tau_1,\tau_2|\mathbf{y}_h,\mathbf{y}_l,\overline{\tau_1,\tau_2},\mathbf{y}_{l_1\,mis})$$
$$\propto\tau_1^{-(\alpha_\delta+3/2)}\tau_2^{-(\alpha_\varepsilon+1)}\frac{1}{|\mathbf{M}_1|^{1/2}}$$
$$\times\exp\left\{-\frac{\gamma_\delta}{\sigma_\rho^2\tau_1}-\frac{1}{\sigma_\rho^2\tau_2}-\frac{(\delta_0-u_\delta)^2}{2v_\delta\sigma_\rho^2\tau_1}\right\}$$
$$\times\exp\left\{-\frac{1}{2\sigma_\rho^2}(\mathbf{y}_h-\rho_0\mathbf{y}_{l_1\,mis}-\delta_0\mathbf{1}_{n_1})^T\mathbf{M}_1^{-1}(\mathbf{y}_h-\rho_0\mathbf{y}_{l_1\,mis}-\delta_0\mathbf{1}_{n_1})\right\} \tag{2.67}$$

$$\mathbf{y}_{l_1\,mis}|\mathbf{y}_h,\mathbf{y}_l,\boldsymbol{\theta}_1,\boldsymbol{\theta}_2$$
$$\sim N\left(\mathbf{F}_{l_1\,mis}\beta_l+\mathbf{R}_{l_1\,12}\,\mathbf{M}_2^{-1}(\mathbf{y}_l-\mathbf{F}_l\boldsymbol{\beta}_l),\sigma_l^2(\mathbf{M}_{2\,mis}-\mathbf{R}_{l_1\,12}\,\mathbf{M}_2^{-1}\mathbf{R}_{l_1\,21})\right) \tag{2.68}$$

where $\mathbf{M}_2=\mathbf{R}_l+\frac{\sigma_\eta^2}{\sigma_l^2}\mathbf{I}_{n\times n}$, $\mathbf{M}_{2\,mis}=\mathbf{R}_{l\,mis}+\frac{\sigma_\eta^2}{\sigma_l^2}\mathbf{I}_{n_1\times n_1}$, $\mathbf{M}_1=A_1\mathbf{R}_lA_1+\frac{\sigma_\delta^2}{\sigma_\rho^2}\mathbf{R}_\rho+\frac{\sigma_\varepsilon^2}{\sigma_\rho^2}\mathbf{I}_{n_1\times n_1}$ and $A_1=\mathrm{diag}\,\{\mathbf{y}_{l\,mis}(\mathbf{x}_1),...,\mathbf{y}_{l\,mis}(\mathbf{x}_{n1})\}$.
The full conditional distribution (2.68) is obtained by observing that the joint posterior distribution of $(\mathbf{y}_{l_1\,mis},\mathbf{y}_l)$ is:

$$\left[\begin{array}{c}\mathbf{y}_{l_1\,mis}\\ \mathbf{y}_l\end{array}\middle|\mathbf{y}_h,\boldsymbol{\theta}_1,\boldsymbol{\theta}_2\right]\sim N\left(\left[\begin{array}{c}\mathbf{F}_{l_1\,mis}\\ \mathbf{F}_l\end{array}\right],\sigma_l^2\left[\begin{array}{cc}\mathbf{M}_{2\,mis}&\mathbf{R}_{l_1\,12}\\ \mathbf{R}_{l_1\,21}&\mathbf{M}_2\end{array}\right]\right), \tag{2.69}$$

and directly applying the usual result on the conditional multivariate normal distribution.

Finally, when a sufficiently large number M of posterior samples is drawn, we approximate the posterior predictive distribution (2.59) as:

$$\begin{aligned}\hat{\mathrm{y}}_h(\mathbf{x}_0) \simeq & \frac{1}{M}\sum_{i=1}^{M} E[\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h, \boldsymbol{\theta}_1^{(i)}, \boldsymbol{\theta}_2^{(i)}, \mathbf{y}_{l_1\,mis}^{(i)}] = \\ = & \frac{1}{M}\sum_{i=1}^{M}\Big(\rho_0^{(i)}\mathrm{y}_l(\mathbf{x}_0) + \delta_0^{(i)} + \\ & + (\mathrm{y}_l(\mathbf{x}_0)\mathbf{A}_1\mathbf{r}_\rho + \tau_1^{(i)}\mathbf{r}_\delta)^T\mathbf{M}^{-1}(\mathrm{y}_h - \rho_0^{(i)}\mathbf{y}_{l_1\,mis}^{(i)} + \delta_0^{(i)}\mathbf{1}_{n_1})\Big),\end{aligned} \tag{2.70}$$

which is nothing but an extension of Equation (2.32) when $\mathbf{y}_{l_1}$ is missing.

We want to point out, that the data augmentation approach presents a major advantage to the two-stage approach previously introduced. It allows us to incorporate in the predictive distribution $p(\mathrm{y}_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h)$, not only the uncertainties brought by the unknown parameters, but also the additional uncertainty due to the missing $\mathbf{y}_{l\,1}$.
On the other hand, the two-stage approach in many cases is much easier to implement and it requires less time to run.

***

Now that the Bayesian posterior predictive distribution has been constructed (even in the case when both the high-resolution and low-resolution experiments are subject to measurement error) and the MCMC simulation techniques required to complete the Bayesian computations have been described in detail, we can proceed with the application of the predictive method illustrated so far. First we need to implement the method as a computational code using Matlab and validate it using suitable data.

Chapter 3

Model validation

In the present chapter we present a validation study on the Matlab implementation of the previously described model.
First we run the implemented code on a simulated data set in order to verify that the inference procedure on the unknown parameters of the model works correctly.
Then we apply the model on a data set provided in [QW08].
Finally we test the predictive performance of the of the extension of the model presented in [QW08] (whose results are marked with the superscript "QW08") on another simulated data set and we compare the Mean Square Prediction Errors computed using:

- the GP model in the form 1.5 that uses only the low-accuracy data (superscript "GPlowres"),
- the GP model in the form 1.5 that uses both the low-accuracy data and the low-accuracy data as they indistinctly came from a unique data set (superscript "GPmerge"),
- the model in [Qia+06] that uses both the low-accuracy and the high-accuracy data (superscript "QW06").

3.1 Matlab Vs. WinBUGS

In their 2008 work Qian and Wu use the WinBUGS software to carry out the Bayesian computations using MCMC sampling techniques.

BUGS is a software developed to perform Bayesian analysis of complex models using MCMC techniques. In particular, WinBUGS is a version of BUGS that runs in Windows operating system. Its main advantage is that it is interactive and it can be easily run by users that are not familiar with the BUGS language. The WinBUGS user manual [Spi+03] describes in detail all the software features and functionalities. Moreover it provides detailed descriptions of how the software works, i.e. which MCMC methods are used, and it also features a comprehensive list of suggested bibliography.

We choose to implement the Bayesian computations in Matlab instead.
Writing our own code, testing it and perfecting it took a significant amount of time. The current version of the code is probably not at the best of its efficiency and it could undoubtedly use several improvements. Anyhow the use of our own code to perform the Bayesian computations allows to have better control and understanding of the results.
Moreover Matlab is probably the most efficient numerical computing environment when it comes to matrices manipulation. Since the computations required involve a significant number of matrix operations, such as inversions and determinants, and WinBUGS matrix standard routines are very slow [BCG04], Matlab sounded like the most natural choice as a programming environment over other softwares, like R for instance.

3.2 Model validation with simulated data - I

In order to validate the inference procedure on the unknown model parameters, we are going to apply the BHGP model to a set of suitably simulated data.

For simplicity we consider a deterministic model for both the LE and HE data, i.e.:

$$\mathrm{y}_l(\mathbf{x}_i) = \boldsymbol{f}_l^T(\mathbf{x}_i)\boldsymbol{\beta}_l + \epsilon_l(\mathbf{x}_i) \quad i = 1, ..., n, \tag{3.1}$$

$$\mathrm{y}_h(\mathbf{x}_i) = \rho(\mathbf{x}_i)\mathrm{y}_l(\mathbf{x}_i) + \delta(\mathbf{x}_i) \quad l = 1, ..., n_1. \tag{3.2}$$

where $\boldsymbol{f}(\mathbf{x}_i) = (1, \mathbf{x}_1, ..., \mathbf{x}_k))^T$ and $\boldsymbol{\beta}_l = (\beta_{l0}, \beta_{l1}, ..., \beta_{lk})^T$ is a vector of unknown regression coefficients, $\epsilon_l(\cdot) \sim GP(0, \sigma_l^2, \boldsymbol{\phi}_l)$. $\rho(\cdot) \sim GP(\rho_0, \sigma_\rho^2, \boldsymbol{\phi}_\rho)$ and $\delta(\cdot) \sim GP(\delta_0, \sigma_\delta^2, \boldsymbol{\phi}_\delta)$.

We assume that the responses depend on two input variables $\mathbf{x}_1$ and $\mathbf{x}_2$ and we need to select an appropriate experimental plan.

3.2.1 Experimental plan

In order to simulate the set of low-resolution data y_l, we build a complete factorial plan with two factors with 6 levels and one replicate. This means that we have a set of $n = 6^2$ experimental conditions. We use the same experimental plan to simulate a set of n high resolution data y_h. It is often the case the set of high resolution data is smaller than the set of low resolution data. Thus $n_1 = 24$ high resolution data are selected among the n that were simulated and they were used as a training set, $\mathbf{y}_{\text{training}}$. The remaining $n - n_1 = 12$ HE data ($\mathbf{y}_{\text{testing}}$) are used as a testing set for model validation.

3.2.2 Hyperparameter selection

First we select a set of values for the model parameters:

$$\begin{aligned}\boldsymbol{\theta}_1 &= (\boldsymbol{\beta}_l, \rho_0, \delta_0)\\ \boldsymbol{\theta}_2 &= (\sigma_l^2, \sigma_\rho^2, \sigma_\delta^2, \sigma_\epsilon^2)\\ \boldsymbol{\theta}_3 &= (\boldsymbol{\phi}_l, \boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta).\end{aligned}$$

Such values are sampled from the priors defined in Chapter 1, after setting the hyperparameters to the following values:

$$\begin{aligned}\alpha_l = \alpha_\rho = \alpha_\delta = 2\\ \gamma_l = \gamma_\rho = \gamma_\delta = 1\end{aligned} \tag{3.3}$$

$$\begin{aligned}\mathbf{u}_l = \mathbf{0},\ u_\rho = 1,\ u_\delta = 0\\ v_l = v_\rho = v_\delta = 1\end{aligned} \tag{3.4}$$

$$\begin{aligned}a_l = a_\rho = a_\delta = 0.1\\ b_l = 1,\ b_\rho = b_\delta = 0.1.\end{aligned} \tag{3.5}$$

Equations (3.3) tell that the variance parameters $\boldsymbol{\theta}_2 = (\sigma_l^2, \sigma_\rho^2, \sigma_\delta^2)$ follow a distribution $IG(2, 1)$, i.e. an Inverse-Gamma distribution with finite mean and infinite variance. This means that the priors of the variance parameters are non-informative.

Equations (3.4) imply that the distributions of the regression parameters $\boldsymbol{\beta}_l$ and the mean parameter δ_0 of the location GP are centered in zero, while the mean parameters ρ_0 of the scale GP has a distribution centered in 1. This means that we do not expect significative average changes of scale or location from the LE to

the HE experiment.
Equations (3.5) imply that the correlation parameters $\boldsymbol{\theta}_3 = (\boldsymbol{\phi}_l, \boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta)$ follow a Gamma distribution $G(.1, .1)$ with mean equal to 1 and variance equal to 10. This suggests that, since we expect the data to be significatively correlated, the correlation parameters need not to be too large.

3.2.3 Data simulation

Table 3.1 sums up the sampled values for the unknown parameters. From now on we will refer to them as the "true values" of the parameters.

Parameter	Sampled value
σ_l^2	0.8533
σ_ρ^2	0.2144
σ_δ^2	0.3327
$\boldsymbol{\beta}_l$	(0.1636, 1.2733, -0.3495)
ρ_0	1.0517
δ_0	-0.0021
$\boldsymbol{\phi}_l$	(0.6190, 0.1865)
$\boldsymbol{\phi}_\rho$	(0.0002, 0.0016)
$\boldsymbol{\phi}_\delta$	(0.0419, 0.0549)

Table 3.1: Sampled values for the parameters (Simulated data - I).

Since there is no built in Matlab function that samples from an Inverse Gamma distribution $IG(\alpha, \gamma)$ we generate the required random samples by taking the inverse of the samples from a Gamma distribution $G(\alpha, 1/\gamma)$, i.e:
`IG_sample = 1/gamrnd(alpha,1/gamma)`.

Once the sampling plan is determined and the parameters values are known, we can compute the correlation matrices of the gaussian processes involved in the model, $\mathbf{R}_l$, $\mathbf{R}_\rho$ and $\mathbf{R}_\delta$ using the function `regfunct`.
Given such matrices and the remaining true parameters we can simulate one realization of the Gaussian processes $\epsilon_l(\cdot)$, $\rho(\cdot)$ and $\delta(\cdot)$. This is done by sampling three n-dimentional vectors from the following multivariate normal distributions:

$$\begin{aligned}\boldsymbol{\epsilon}_l &\sim N(\mathbf{0}, \sigma_l^2\mathbf{R}_l)\\ \boldsymbol{\rho} &\sim N(\rho_0\mathbf{1}_{n_1}, \sigma_\rho^2\mathbf{R}_\rho)\\ \boldsymbol{\delta} &\sim N(\delta_0\mathbf{1}_{n_1}, \sigma_\delta^2\mathbf{R}_\delta)\end{aligned}$$

using the `mvnrnd` Matlab function.
Finally the simulated data $\mathbf{y}_l$ and $\mathbf{y}_h$ are computed as:

$$\begin{aligned} \mathrm{y}_{li} &= \boldsymbol{f}(\mathbf{x}_i)^T \boldsymbol{\beta}_l + \epsilon_{li} \\ \mathrm{y}_{hi} &= \rho_i \mathrm{y}_{li} + \delta_i, \quad \text{for } i = 1, .., 100. \end{aligned}$$

As previously stated we randomly pick $n_1 = 24$ high resolution data from $\mathbf{y}_h$ and we use them as training set. The remaining will be employed for model validation.

Now that we have suitably simulated both the low-resolution and high-resolution data we proceed with the inference procedure.

3.2.4 Estimation of the correlation parameters

First we compute the posterior estimates of the correlation parameters from the data, in agreement with the choice to adopt an empirical Bayesian approach. As in [QW08] the mode of the marginal posterior distribution of the correlation parameters is used as a point estimator for such quantities.

We previously illustrated in Section 2.2.1 that the marginal posterior distribution of $\boldsymbol{\theta}_3 = (\boldsymbol{\phi}_l, \boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta)$ is obtained by integrating out the mean and variance parameters, $\boldsymbol{\theta}_1 = (\boldsymbol{\beta}_l, \rho_0, \delta 0)$ and $\boldsymbol{\theta}_2 = (\sigma_l^2, \sigma_\rho^2, \tau_1)$ from the joint posterior distribution of all the unknown parameters $(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3)$. In the deterministic case of the model described in Equations 3.19 the marginal posterior distribution of $\boldsymbol{\theta}_3$ is:

$$\begin{aligned} & p(\boldsymbol{\theta}_3 | \mathbf{y}_h, \mathbf{y}_l) \propto L_1(\boldsymbol{\phi}_l, \boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta) = \\ & = p(\boldsymbol{\phi}_l) \mid \mathbf{R}_l \mid^{-1/2} \mid \mathbf{H}_1 \mid^{-1/2} \left(\gamma_l + \frac{\mathbf{b}_1^T \mathbf{H}_1^{-1} \mathbf{b_1} - c_1}{2} \right)^{-\left(\alpha_l + \frac{n}{2}\right)} \cdot p(\boldsymbol{\phi}_\rho) p(\boldsymbol{\phi}_\delta) \times \\ & \quad \times \int_{\tau_1} (a_2 a_3)^{-1/2} \mid \mathbf{M} \mid^{-1/2} \left(\frac{b_3^2 - a_3 c_3}{2 a_3} + \gamma_\rho + \frac{\gamma_\delta}{\tau_1} \right)^{-\left(\frac{n_1}{2} + \alpha_\rho + \alpha_\delta + 1\right)} \tau_1^{-\left(\alpha_\delta + \frac{3}{2}\right)} d\tau_1 \end{aligned} \tag{3.6}$$

where $\mathbf{M} = \mathbf{A}_1 \mathbf{R}_\rho \mathbf{A}_1 + \tau_1 \mathbf{R}_\delta$ and $\mathbf{H}_1$, $\mathbf{b}_1$, c_1, a_2, t_1, t_2, t_3, a_3, b_3, c_3 are the same defined in Section 2.2.1. The posterior mode estimates of $\boldsymbol{\phi}_l$, $\boldsymbol{\phi}_\rho$ and $\boldsymbol{\phi}_\delta$ are computed by separately solving the two following optimization problems:

$$\hat{\boldsymbol{\phi}}_l = \arg\max_{\boldsymbol{\phi}_l} \left\{ p(\boldsymbol{\phi}_l) \mid \mathbf{R}_l \mid^{-1/2} \mid \mathbf{H}_1 \mid^{-1/2} \left(\gamma_l + \frac{\mathbf{b}_1^T \mathbf{H}_1^{-1} \mathbf{b_1} - c_1}{2} \right)^{-\left(\alpha_l + \frac{n}{2}\right)} \right\} \quad \text{s.t. } \boldsymbol{\phi}_l > \mathbf{0} \tag{3.7}$$

$$\begin{aligned}(\hat{\boldsymbol{\phi}}_\rho, \hat{\boldsymbol{\phi}}_\delta) =& \arg\max_{\boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta} \Bigg\{ p(\boldsymbol{\phi}_\rho) p(\boldsymbol{\phi}_\delta) \int_{\tau_1} (a_2 a_3)^{-1/2} \mid \mathbf{M} \mid^{-1/2} \times \\ & \times \left(\frac{b_3^2 - a_3 c_3}{2a_3} + \gamma_\rho + \frac{\gamma_\delta}{\tau_1} \right)^{-\left(\frac{n_1}{2} + \alpha_\rho + \alpha_\delta + 1\right)} \tau_1^{-\left(\alpha_\delta + \frac{3}{2}\right)} d\tau_1 \Bigg\} \\ & \text{s.t. } (\boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta) > \mathbf{0}\end{aligned} \tag{3.8}$$

Problem (3.7) is exactly the same of Problem (2.9), while Problem (3.8) is Problem (2.10) adjusted to the deterministic case.

If we recast the optimization problem for $\boldsymbol{\phi}_\rho$ and $\boldsymbol{\phi}_\delta$ as in Equation (2.11) with:

$$\begin{aligned}f(\tau_1, \tau_2) =& p(\boldsymbol{\phi}_\rho) p(\boldsymbol{\phi}_\delta) (a_2 a_3)^{-1/2} \mid \mathbf{M} \mid^{-1/2} \times \\ & \times \left(\frac{b_3^2 - a_3 c_3}{2a_3} + \gamma_\rho + \frac{\gamma_\delta}{\tau_1} \right)^{-\left(\frac{n_1}{2} + \alpha_\rho + \alpha_\delta + 1\right)} \exp\left\{ \frac{\gamma_1}{\tau_1} \right\}\end{aligned} \tag{3.9}$$

where:

$$\tau_1 \sim IG(\alpha_\delta + \frac{1}{2}, \gamma_1), \tag{3.10}$$

we can approximate Problem (3.8) using the SAA method illustrated in Section 2.2.2 and solve the equivalent deterministic problem:

$$(\hat{\boldsymbol{\phi}}_\rho, \hat{\boldsymbol{\phi}}_\delta) = \arg\max_{\boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta} \left\{ \frac{1}{S} \sum_{s=1}^{S} f(\tau_1^{(s)}) \right\} \quad \text{s.t. } (\boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta) > \mathbf{0} \tag{3.11}$$

where $\tau_1^{(s)}$ are $S = 100$ samples from the distribution (3.10).

Both Problems (3.7) and (3.11) can be solved with a constrained non-linear optimization algorithm that uses a quasi-Newton method.
The Matlab function `fmincon` solves a minimization problem using this kind of algorithm. Thus we have to recast our problems as minimization problems. This is merely done by changing the signs of the objective functions.

Moreover, in order to avoid numerical issues, we perform the minimization of the natural logarithm of the objective functions. Minimizing the log of the objective function is equivalent to minimizing the objective function in its natural scale because the natural logarithm is a continuous monotonically increasing function. In conclusion, we need to solve the two following optimization problems to estimate the correlation parameters using their posterior mode:

$$\hat{\boldsymbol{\phi}}_l = \arg\min_{\boldsymbol{\phi}_l} \left\{ -\log \left(p(\boldsymbol{\phi}_l) \mid \mathbf{R}_l \mid^{-1/2} \mid \mathbf{H}_1 \mid^{-1/2} \left(\gamma_l + \frac{\mathbf{b_1^T} \mathbf{H}_1^{-1} \mathbf{b_1} - c_1}{2} \right)^{-\left(\alpha_l + \frac{n}{2}\right)} \right) \right\}$$
$$\text{s.t. } \boldsymbol{\phi}_l > \mathbf{0} \tag{3.12}$$

$$(\hat{\boldsymbol{\phi}}_\rho, \hat{\boldsymbol{\phi}}_\delta) = \arg \min_{\boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta} \left\{ -\log \left(\frac{1}{S} \sum_{s=1}^{S} f(\tau_1^{(s)}) \right) \right\} \quad \text{s.t. } (\boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta) > \mathbf{0} \tag{3.13}$$

An alternative implementation of the optimization can be used in order to increase its efficiency. This is accomplished by operating the following changes of variables in the objective functions:

$$(\boldsymbol{\phi}_l, \boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta) = \left(\exp\left\{\boldsymbol{\psi}_l\right\}, \exp\left\{\boldsymbol{\psi}_\rho\right\}, \exp\left\{\boldsymbol{\psi}_\delta\right\}\right). \tag{3.14}$$

This transformation makes the original constrained optimization unconstrained. In fact the solution of the equivalent unconstrained optimization in the variables $(\boldsymbol{\psi}_l, \boldsymbol{\psi}_\rho, \boldsymbol{\psi}_\delta)$ allows the correlation parameters $(\boldsymbol{\phi}_l, \boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta)$ to take only positive values.

After the change of variables (3.14), Problems (3.12) and (3.13) become:

$$\hat{\boldsymbol{\phi}}_l = \exp \arg\min_{\boldsymbol{\psi}_l} \Bigg\{ -\log \Bigg(p(\boldsymbol{\phi}_l) \mid \mathbf{R}_l \mid^{-1/2} \mid \mathbf{H}_1 \mid^{-1/2} \times \\ \times \left(\gamma_l + \frac{\mathbf{b_1^T} \mathbf{H}_1^{-1} \mathbf{b_1} - c_1}{2} \right)^{-\left(\alpha_l + \frac{n}{2}\right)} \exp\left\{ \sum_{j=1}^{k} \psi_{lj} \right\} \Bigg) \Bigg\} \tag{3.15}$$

$$(\hat{\boldsymbol{\phi}}_\rho, \hat{\boldsymbol{\phi}}_\delta) = \exp \arg \min_{\boldsymbol{\psi}_\rho, \boldsymbol{\psi}_\delta} \left\{ -\log \left(\frac{1}{S} \sum_{s=1}^{S} f(\tau_1^{(s)}) \exp\left\{ \sum_{j=1}^{k} (\psi_{\rho j} + \psi_{\delta j}) \right\} \right) \right\} \tag{3.16}$$

where the terms $\exp\left\{\sum_{j=1}^{k}\psi_{lj}\right\}$ and $\exp\left\{\sum_{j=1}^{k}(\psi_{\rho j}+\psi_{\delta j})\right\}$ are the determinants of the Jacobian matrices of the inverse of the transformations applied on $\boldsymbol{\phi}_l$ and $(\boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta)$ respectively (refer to Appendix B.2 for the change of variables rule). In our case $k = 2$.

Transformation (3.14) brings two major benefits to the optimization procedure.
First it allows to substitute constrained optimization with unconstrained optimization, which is generally easier to solve [XDW07]. We use the Matlab function `fminunc`, that implements a quasi-Newton algorithm for unconstrained non-linear optimization.
Secondly, transformation (3.14) improves the shape of the objective function. This argument finds an intuitive justification in the fact that after the transformation the optimization algorithm searches the minimum of the objective function in the space of the parameters $\boldsymbol{\psi}_l$, that corresponds to the space of the parameters $\boldsymbol{\phi}_l$ in a logarithmic scale.

The `fminunc` algorithm applied to the optimization problems (3.15) and (3.16) converges to the following solution:

$$\begin{aligned} \hat{\boldsymbol{\phi}}_l &= (0.3759,\ 0.1988) \\ \hat{\boldsymbol{\phi}}_\rho &= (0.0273 \times 10^{-3},\ 0.1388 \times 10^{-3}) \\ \hat{\boldsymbol{\phi}}_\delta &= (0.0267,\ 0.0477). \end{aligned} \tag{3.17}$$

We stick to such estimated values for the correlation parameters and we proceed with the implementation of the two-step algorithm described in Sections 2.1.1 and 2.2.3 to approximate the posterior predictive distribution.
Having chosen an empirical Bayesian implementation, we remind that we will not be able to incorporate the uncertainties due to the estimated correlation parameters in the predictive density.

3.2.5 MCMC sampling of the unknown parameters

Now that the correlation parameters are estimated, the correlation matrices of the three Gaussian Processes in model (3.19), $\mathbf{R}_l$, $\mathbf{R}_\rho$ and $\mathbf{R}_\delta$, are fully determined. We built such matrices according to the definition of the Gaussian correlation structure in (1.4) by using the Matlab function `regfunct` we implemented.

As mentioned in the previous chapter, the Gibbs sampling algorithm is used to draw samples of the unknown parameters from the joint posterior distribution of the parameters $(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$ in the empirical Bayesian framework. This task is accomplished by implementing the Gibbs sampler described in Section 2.3 in Matlab.

A few observations need to be done.
First, we point out that since both our LE and HE models are deterministic, the following simplification is allowed: all the terms involving the parameters τ_2 (i.e. σ_ϵ^2), α_ϵ and γ_ϵ can be dropped. This significantly eases the sampling Metropolis-within-Gibbs algorithm as it is used for univariate sampling of parameter τ_1 instead of both (τ_1, τ_2).
Secondly, the particular data we simulate satisfy the simplifying assumption made at the beginning of Section 1.5. The experimental plans of the low-resolution and the high-resolution data are such that $D_h \subset D_l$, i.e. $\mathrm{y}_h(\mathbf{x}_i)$ is available at the same input points $\mathbf{x}_i$, $i = 1, ..., n_1$, where the corresponding $\mathrm{y}_{l1}(\mathbf{x}_i)$ is available. Moreover, in order to compare the results of the prediction of y_h at new input points, we build a testing set of high resolution data that includes $\mathrm{y}_{h\text{test}}(\mathbf{x}_j)$ evaluated at untried points $\mathbf{x}_j$, $j = 1, ..., n - n_1$, such that $\mathbf{x}_j \in D_l \setminus D_h$. This means that the low-resolution response $\mathrm{y}_l(\cdot)$ is available at the testing points. In this case the mixture that approximates the posterior predictive distribution of high-resolution data at $\mathbf{x}_j \in D_l \setminus D_h$ is easier to compute and it solely depends on the sampled values of the unknown parameters ρ_0, σ_ρ^2, δ_0 and σ_δ^2, as in Equation (2.32). Thus, in order to compute the predictor $\hat{\mathrm{y}}(\mathbf{x}_j)$ with $\mathbf{x}_j \in D_l \setminus D_h$ sampling from the joint posterior distribution of the parameters $(\rho_0, \sigma_\rho^2, \delta_0, \sigma_\delta^2)$ would suffice in our case and the full conditional distributions of $\boldsymbol{\beta}_l$ and σ_l^2 are left out of the Gibbs algorithm.

Before launching the Gibbs sampler, it is important to suitably tune some parameters.
The starting points of the algorithm are set to the values of the Maximum Likelihood estimates of the unknown parameters ρ_0, δ_0, σ_ρ^2 and σ_δ^2. Such estimates are easily computed using the `fmincon` function. These starting points appear to be an appropriate initial guess and they allow to shorten the burn-in period, i.e. the number of iterations before convergence is achieved.
We remind that we need to use the Metropolis-within-Gibbs algorithm at each loop of the Gibbs algorithm because the form of the full conditional distribu-

tion of τ_1 does not belong to any known-form distribution family. Thus we also need to properly set the variance parameter of the proposal distribution of the Random-Walk Metropolis-within-Gibbs algorithm. This can be done by launching very few iterations of the Gibbs algorithm (say around 100) and by observing the mean and the standard deviation of the acceptance rates of the Metropolis-within-Gibbs algorithm computed at each iteration of the Gibbs algorithm. As suggested by [Gel+03] the variance of the proposal distribution should be tuned in order to have acceptance rates around 50% in the case of univariate sampling with Metropolis.

Finally we need to decide the number of iterations of the Metropolis-within-Gibbs algorithm. This is done by using two different diagnostics for assessing convergence of MCMC algorithms: the Geweke diagnostic, based on a test statistic that compares the average value of the chain at the first iterations after burn-in and at the latter iterations, and the Gelman-Rubin diagnostic, based on the comparison of multiple runs of the chain with overdispersed initial points by using within- and between-variances. For further details on these diagnostics see Appendix A.3.

We observed that if we repeatedly run $N = 5001$ iterations of Metropolis-within-Gibbs and we discard the first $k_0^{Metropolis} = 1667$ (about 1/3 of the total) the simulated chains pass the Geweke test most of the times. Furthermore, we ran 5 parallel chains with overdispersed starting points and we got a potential scale reduction $\hat{R} = 1.0005$ (refer to Equation (A.18)), which is sufficiently close to one. Given these results we can reasonably assess that if we draw $N = 5001$ samples and we discard the first $k_0^{Metropolis} = 5000$, the last sample, which is the only one retained, is representative of the target distribution that we are sampling from.

We run $K = 10\,000$ iterations of the Gibbs algorithm and we set the burn-in for the Gibbs sampler to $k_0^{Gibbs} = 3\,000$.

The `coda` function provides the p-values for the Chi-squared test of the Geweke convergence diagnostic:

$$\begin{aligned}
&\text{p-value}(\sigma_l^2) = 0.801951\\
&\text{p-value}(\rho_0) = 0.938348\\
&\text{p-value}(\sigma_\rho^2) = 0.059631\\
&\text{p-value}(\delta_0) = 0.485391\\
&\text{p-value}(\tau_1) = 0.067082.
\end{aligned} \tag{3.18}$$

Such p-values suggest that there is statistical evidence that the means of the first

10% and the last 50% of the chains are equal.

3.2.6 Inference on the unknown model parameters

Table 3.2 summarizes the results of the posterior sampling: the true value, the posterior means, the posterior medians and the lower and upper 95% HPD credibility limits of the marginal posterior distribution of the parameters $\boldsymbol{\beta}_l$, σ_l^2, ρ_0, δ_0, σ_ρ^2 and σ_δ^2.

	True value	ML estimate	Posterior mean	Posterior median	LCL HPD (2.5%)	UCL HPD (97.5%)
β_{l0}	0.1636	-0.0452	0.0325	0.0381	-1.2918	1.3404
β_{l1}	1.2733	1.2341	1.2071	1.2086	0.8947	1.5159
β_{l2}	-0.3495	-0.3679	-0.3632	-0.3633	-0.6295	-0.1037
σ_l^2	0.8533	1.1101	1.0161	0.9841	0.6458	1.5585
ρ_0	1.0517	1.0914	1.0574	1.0561	0.5392	1.5929
σ_ρ^2	0.2144	0.2688	0.4784	0.3847	0.1590	1.3783
δ_0	-0.0021	-0.1740	-0.1313	-0.1313	-0.2695	0.0057
σ_δ^2	0.3327	0.2016	0.2635	0.2496	0.1493	0.4580

Table 3.2: True values, posterior means and medians of the parameters $\boldsymbol{\beta}_l$, σ_l^2, ρ_0, δ_0, σ_ρ^2 and σ_δ^2 and their respective 95% credibility HPD limits (Simulated data - I).

Figures 3.1 and 3.2 represent the histograms of the values of the parameters sampled from their posterior distribution. Such histograms can be regarded also as the non-normalized empirical marginal posterior distributions of the parameters. We observe how the true values of the parameters always fall inside the 95% HPD credibility interval of their respective non-normalized empirical marginal posterior distributions. Only the posterior median of the parameter δ_0 appears to be smaller than its true value, but this result id confirmed by the corresponding ML estimate. We are allowed to conclude that the code Gibbs algorithm implemented in Matlab correctly samples the unknown model parameters from their joint posterior distribution.

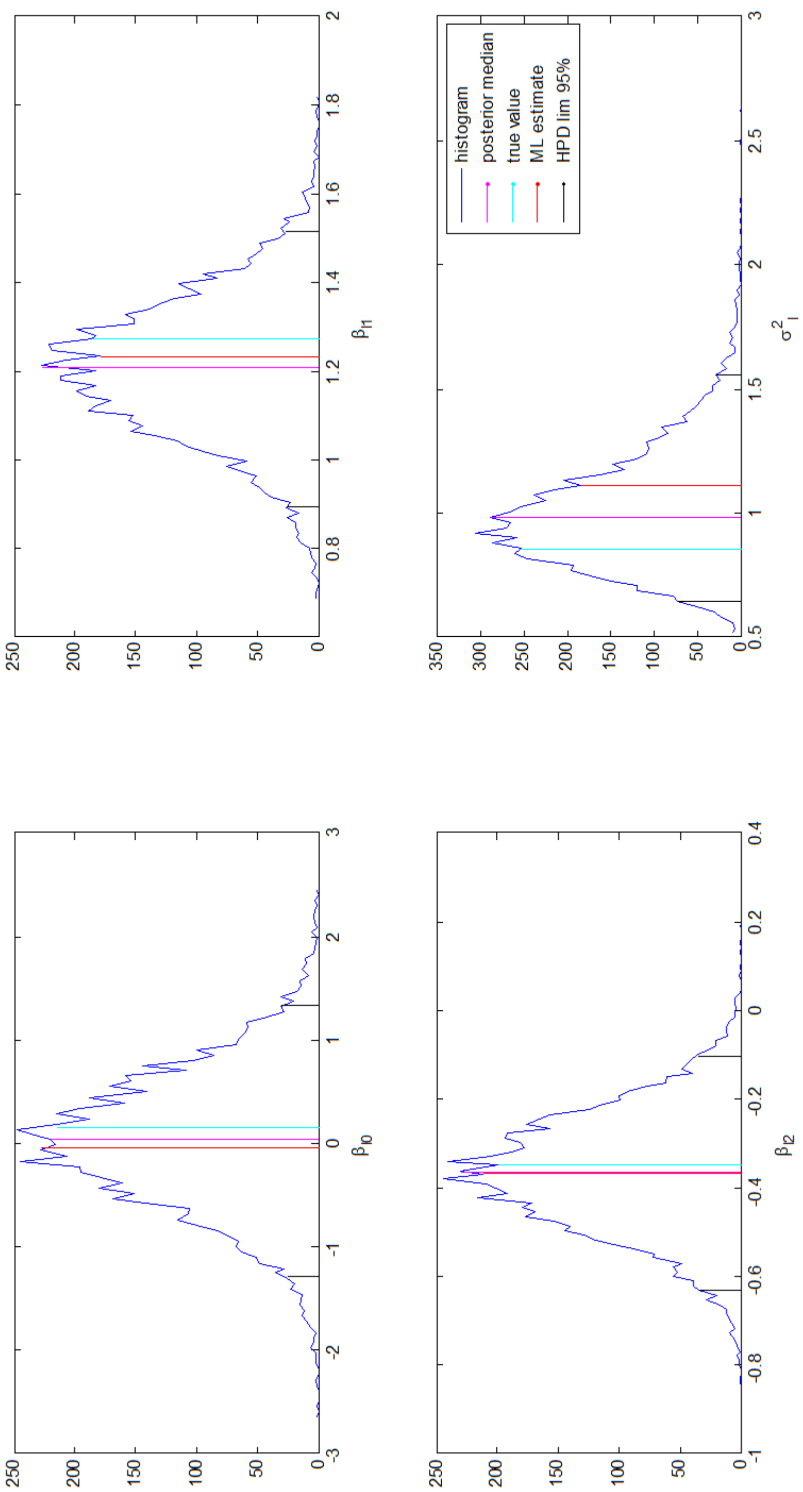

Figure 3.1: Histograms of β_l and σ_l^2. The purple lines represent the posterior medians, the cyan lines the true values of the parameters, the red lines their ML estimates, the black lines the 95% credibility HPD limits (Simulated data - I).

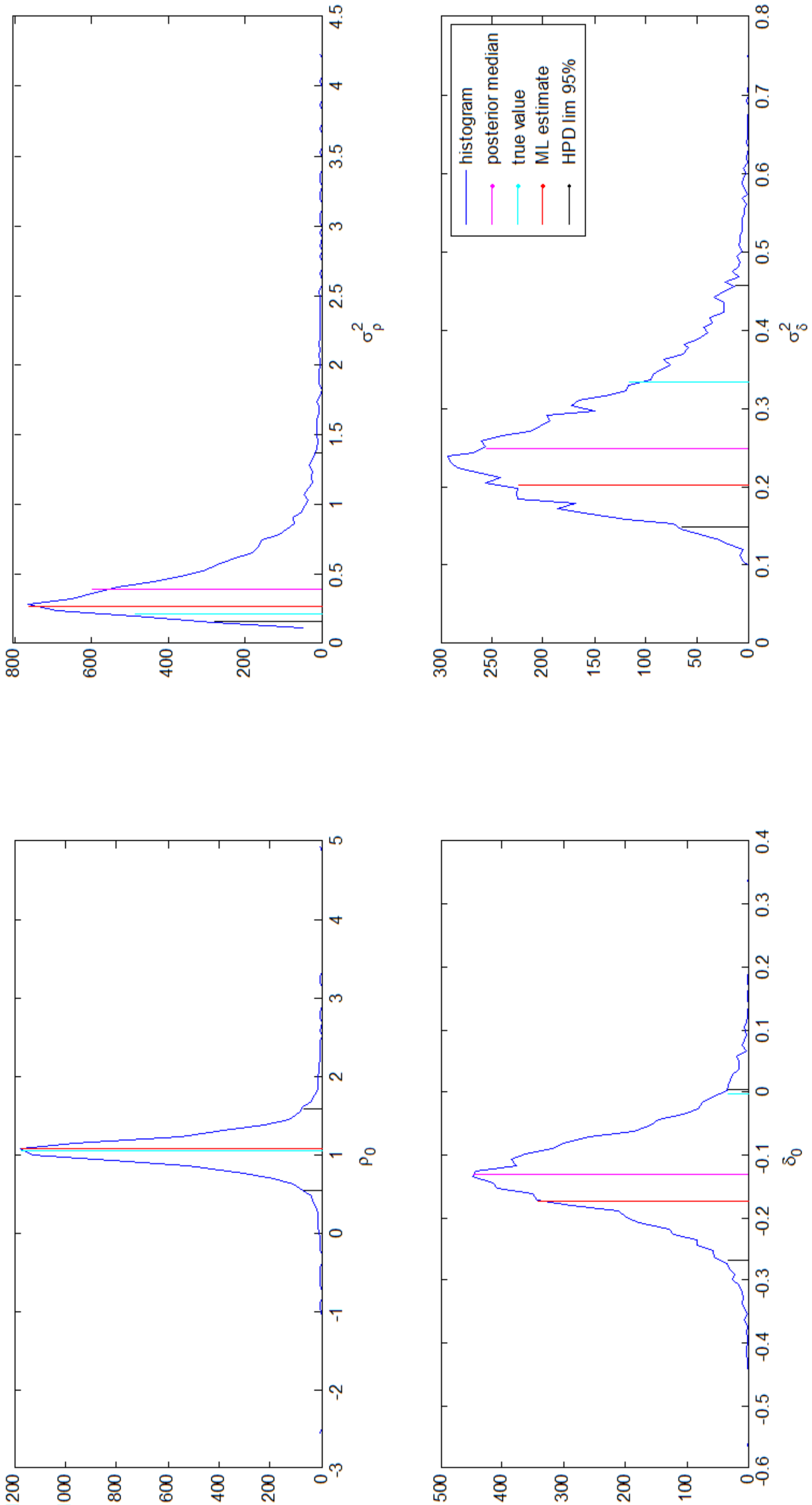

Figure 3.2: Histograms of ρ_0, δ_0, σ^2_ρ and σ^2_δ. The purple lines represent the posterior medians, the cyan lines the true values of the parameters, the red lines their ML estimates, the black lines the 95% credibility HPD limits (Simulated data - I).

3.3 Model validation with real data

In order to validate the code implemented in Matlab, we also use the data reported in Example 1 in [QW08] and we compare our results with the ones obtained in the paper.
Such data consist of the outputs of two different computer simulations for a heat exchanger in an electronic cooling application. We will provide just some essential information about the data themselves. For further details refer to [Qia+06] and [QW08], where a thorough description of the experiment is provided.

The response of interest y is the total rate of steady-state heat transfer of a device used to dissipate heat generated by a heat source. The response depends on the following $k = 4$ factors, $\mathbf{x}_1$, $\mathbf{x}_2$, $\mathbf{x}_3$ and $\mathbf{x}_4$ respectively:

- the mass flow rate of entry air $\dot{m}$,
- the temperature of entry air T_{in},
- the temperature of the heat source T_{wall},
- the solid material thermal conductivity k.

Once an appropriate experimental design is selected, two different computer simulations are run.
An approximate but fast simulation using finite difference method is used to produce the LE response and a detailed but slow simulation based on FLUENT finite element analysis is used to generate the HE response.
The low-resolution and high resolution responses are computed at $n = 36$ different experimental conditions. $\mathbf{y}_l$ denotes the LE data. We randomly select $n_1 = 24$ high resolution data among the n we simulated and we use such data as a training set. The $n - n_1 = 12$ conditions left out are used as a testing set for model validation.

It has to be pointed out that both the responses of the LE and HE experiments come from computer simulations. Thus, since no measurement error is involved, two deterministic models for both the LE and HE data are used, i.e.:

$$\mathrm{y}_l(\mathbf{x}_i) = \boldsymbol{f}_l^T(\mathbf{x}_i)\boldsymbol{\beta}_l + \epsilon_l(\mathbf{x}_i) \quad i = 1, ..., n, \tag{3.19}$$

$$\mathrm{y}_h(\mathbf{x}_l) = \rho(\mathbf{x}_l)\mathrm{y}_l(\mathbf{x}_l) + \delta(\mathbf{x}_l) \quad l = 1, ..., n_1. \tag{3.20}$$

where $\boldsymbol{f}(\mathbf{x}_i) = (1, \mathbf{x}_1, ..., \mathbf{x}_k)^T$ and $\boldsymbol{\beta}_l = (\beta_{l0}, \beta_{l1}, ..., \beta_{lk})^T$ is a vector of unknown regression coefficients, $\epsilon_l(\cdot) \sim GP(0, \sigma_l^2, \boldsymbol{\phi}_l)$, $\rho(\cdot) \sim GP(\rho_0, \sigma_\rho^2, \boldsymbol{\phi}_\rho)$ and $\delta(\cdot) \sim GP(\delta_0, \sigma_\delta^2, \boldsymbol{\phi}_\delta)$.
As a consequence we are allowed to simplify the procedure described in the previous chapter, by dropping all the terms involving the measurement error $\epsilon(\cdot)$.

We set the hyperparameters of the priors to the following values:

$$\begin{aligned}
&\alpha_l = \alpha_\rho = \alpha_\delta = 2\\
&\gamma_l = \gamma_\rho = \gamma_\delta = 1\\
&\mathbf{u}_l = \mathbf{0},\ u_\rho = 1,\ u_\delta = 0\\
&\mathrm{v}_l = \mathrm{v}_\rho = \mathrm{v}_\delta = 1\\
&a_l = a_\rho = a_\delta = 0.1\\
&b_l = b_\rho = b_\delta = 0.1.
\end{aligned}$$

The same observations we previously made about such choice of hyperparameters still stand.
Then we repeat the same steps we followed in the previous section and we run 10 000 iterations of the Gibbs algorithm. Using the Geweke convergence diagnostic implemented in the `coda` function we conclude that convergence is achieved after 3 000 burn-in iterations.

Table 3.3 summarizes the results of the posterior sampling: the posterior means, the posterior medians and the lower and upper 95% HPD credibility limits of the marginal posterior distribution of the parameters ρ_0, δ_0, σ_ρ^2 and σ_δ^2. As we expected the mean of the scale GP $\rho(\cdot)$ is about 1 and the mean of the location GP $\delta(\cdot)$ is small.

	Posterior mean	Posterior median	LCL HPD (2.5%)	UCL HPD (97.5%)
ρ_0	1.1128	1.1061	0.8687	1.4034
σ_ρ^2	0.3573	0.3174	0.1481	0.7826
δ_0	0.4040	0.3803	-0.3452	1.2885
σ_δ^2	0.5448	0.5057	0.2671	1.0489

Table 3.3: Posterior mean and median of the parameters ρ_0, δ_0, σ_ρ^2 and σ_δ^2 and their respective 95% credibility HPD limits (Example 1 from [QW08]).

Figure 3.3 represents the histograms of the values of the parameters sampled from their posterior distribution. Such histograms can be regarded also as the non-normalized empirical marginal posterior distributions of the parameters.

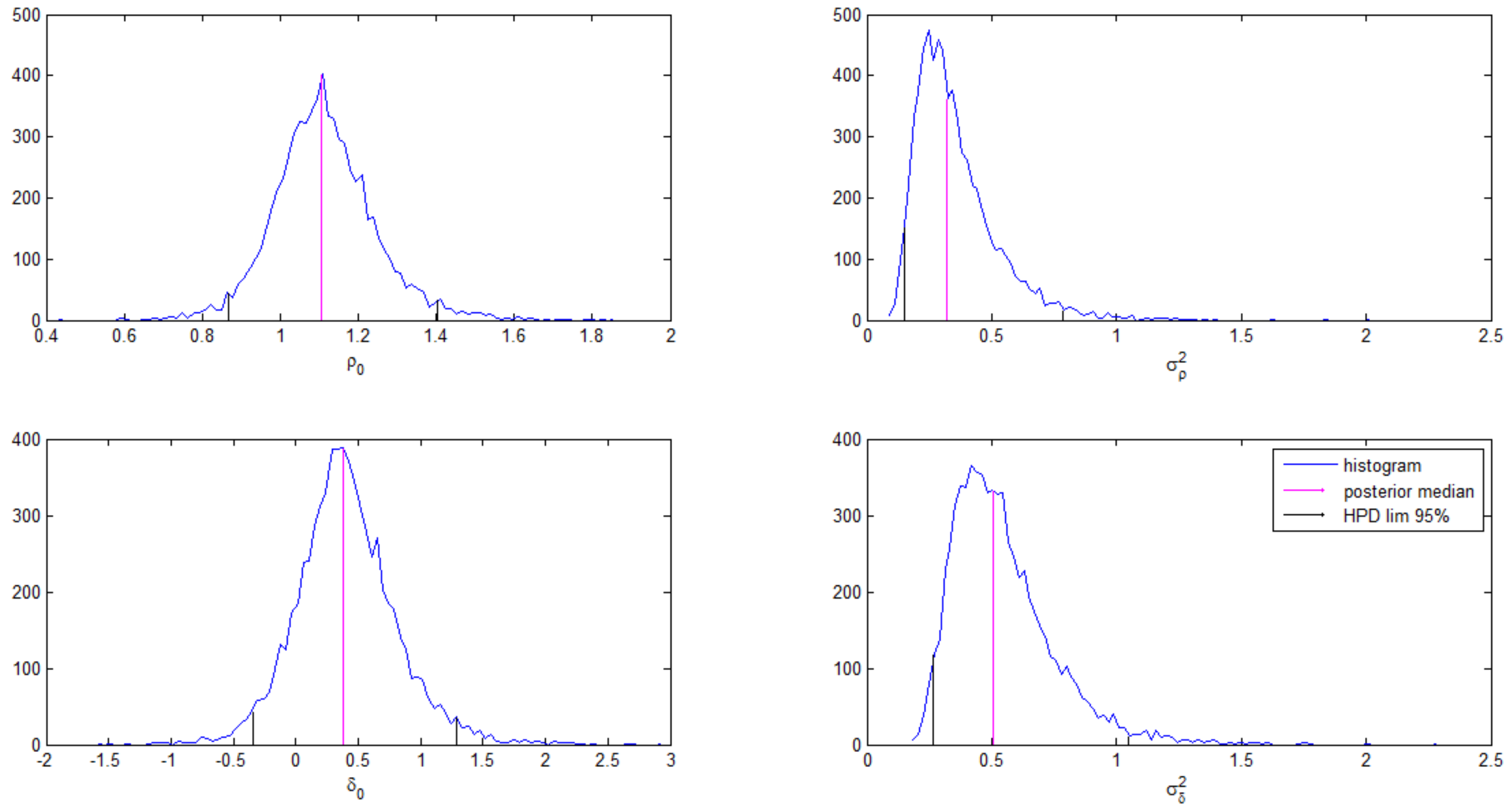

Figure 3.3: Histograms of ρ_0, δ_0, σ_ρ^2 and σ_δ^2. The purple lines represent the posterior medians, the black lines the 95% credibility HPD limits (Example 1 from [QW08]).

Although the data sets are the same, the numerical results we came to appear to be different from the ones presented in [QW08]. Our Matlab implementation led to different estimates of the correlation parameters and different marginal posterior distributions for the parameters.
In spite of this evident discrepancy, we proceed with the prediction procedure.

3.3.1 Approximation of the predictive distribution

Now that a large number of samples has been drawn from the joint posterior distribution (2.20), we can proceed with the second step of the MCMC algorithm described in Section 2.4. We predict the HE response at the $n_{test} = 12$ input points belonging to the testing set, for cross-validation
Table 3.4 reports the prediction results we obtained applying the BHGP model and the respective 95% empirical prediction limits. We refer to the results computed with such method using the superscript "QW08".
As pointed out in [QW08] the true response corresponding to the 7th test data (run nr. 18) appears to be significatively smaller than the responses corresponding to the other runs. The authors believe that the anomaly might be due to the failure of the finite element simulation for that particular run. The corresponding prediction is $\hat{\mathrm{y}}^{\mathrm{QW08}}_{h\,\mathrm{test}}(\mathbf{x}_{7_{\mathrm{th}}}) = 11.1491$ and it is probably close to the effective LE response. As a matter of fact we obtain a very high Mean Square Prediction Error:

$$\mathrm{MSPE}^{\mathrm{QW08}} = \frac{1}{n_{test}} \sum_{i=1}^{n_{test}} \Big(\mathrm{y}_{h\,\mathrm{test}}(\mathbf{x}_i) - \hat{\mathrm{y}}^{\mathrm{QW08}}_{h\,\mathrm{test}}(\mathbf{x}_i)\Big)^2 = 7.1039. \tag{3.21}$$

We follow the suggestion given in [QW08] and we exclude the 7th test data because it is an outlier. The new Mean Square Prediction Error is about half of the MSPE previously computed:

$$\mathrm{MSPE}^{\mathrm{QW08}}_{7\mathrm{th}} = 3.7900. \tag{3.22}$$

Finally in Table 3.5 the prediction $\hat{\mathrm{y}}^{\mathrm{QW08}}_{h\,\mathrm{test}}$ obtained with the integration of LE and HE data is compared with the predictions computed employing the GP universal kriging model that uses only the LE data. $\hat{\mathrm{y}}^{\mathrm{SWN}}_{\mathrm{test}}$ comes from the application of the predictive distribution (1.9) proposed by Santner, Williams, and Notz ([SWN03]) and $\hat{\mathrm{y}}^{\mathrm{GPlowres}}_{h\,\mathrm{test}}$ is the prediction computed using the 3.1 version of the GPML code by Rasmussen and Nickisch [RN11].

The respective Mean Square Prediction Errors are:

$$\begin{aligned} \mathrm{MSPE}^{\mathrm{QW08}}_{7_{\mathrm{th}}} &= 3.7900 \\ \mathrm{MSPE}^{\mathrm{SWN}}_{7_{\mathrm{th}}} &= 10.3742 \\ \mathrm{MSPE}^{\mathrm{GPlowres}}_{7_{\mathrm{th}}} &= 10.8325. \end{aligned} \tag{3.23}$$

Note that all the MSPEs were calculated after excluding the 18th run.

Run	$\mathrm{y}_{h\,\mathrm{test}}$	$\hat{\mathrm{y}}^{\mathrm{BHGP}}_{h\,\mathrm{test}}$	LPL (2.5%)	UPL (97.5%)
1	25.8200	23.9229	23.7105	24.1222
4	19.7700	23.7531	23.5366	24.0263
9	20.5200	20.2676	20.2423	20.3056
11	18.7800	17.2435	17.1863	17.2935
13	24.6800	25.4501	25.4104	25.4887
17	22.3000	22.2209	22.1485	22.3008
21	23.3300	23.0538	23.0015	23.1024
26	32.8500	37.0659	36.8758	37.2093
28	34.8000	34.0509	33.9398	34.1419
30	36.1100	35.7870	35.6297	36.0172
31	27.3600	26.5313	26.4439	26.6899

Table 3.4: Prediction results at the $n_{test} = 12$ testing points. $\mathrm{y}_{h\,\mathrm{test}}$ is the true HE response; $\hat{\mathrm{y}}^{\mathrm{BHGP}}_{h\,\mathrm{test}}$ is the response predicted using the BHGP model LPL and UPL are the 95% prediction limits (Example 1 from [QW08]).

Run	$\mathrm{y}_{h\,\mathrm{test}}$	$\hat{\mathrm{y}}^{\mathrm{BHGP}}_{h\,\mathrm{test}}$	$\hat{\mathrm{y}}^{\mathrm{SWN}}_{\mathrm{test}}$	$\hat{\mathrm{y}}^{\mathrm{Rasmussen}}_{\mathrm{test}}$
1	25.8200	23.9229	27.0525	26.9767
4	19.7700	23.7531	25.5577	25.6905
9	20.5200	20.2676	20.7199	20.7519
11	18.7800	17.2435	16.2783	16.2692
13	24.6800	25.4501	25.1349	25.3379
17	22.3000	22.2209	22.5746	22.3670
21	23.3300	23.0538	22.0156	22.0475
26	32.8500	37.0659	30.8178	30.9879
28	34.8000	34.0509	29.1543	28.9487
30	36.1100	35.7870	31.0714	30.9184
31	27.3600	26.5313	23.8499	23.8518

Table 3.5: Prediction results at the $n_{test} = 12$ testing points computed using the BHGP model are compared to the other two predictions $\hat{\mathrm{y}}^{\mathrm{SWN}}_{\mathrm{test}}$ and $\hat{\mathrm{y}}^{\mathrm{Rasmussen}}_{\mathrm{test}}$. Such responses are computed using only the LE data with the predictive density proposed by [SWN03] and the method proposed by [RW06] respectively (Example 1 from [QW08]).

Because of the discrepancies between our results on the marginal posteriors of the parameters and the ones in [QW08], we decide to proceed with further tests.

3.4 Performance comparison with simulated data - II

In order to evaluate the predictive performance of the BHGP model, we use the implemented Matlab code in a more complicated case. We consider now the situation in which both the LE and the HE data are affected by a random error, thus the data are described by the model in Equations (2.38) and (2.39). We simulate two sets of low-resolution and high-resolution data in the way described below.

Since we are going to apply the Gaussian process models seen so far to multi-resolution metrology data it is reasonable to assume that $k = 2$, i.e. two input variables $\mathbf{x}_1$ and $\mathbf{x}_2$ are available. This is in accordance to the fact that the data coming from coordinate measurements consist of "clouds" of points that represent the 3D coordinates of points on a surface. Thus the input variables are the x and y coordinates of the measured points and the response of interest is the height (the z coordinate) of the surface point corresponding to the given input variables. Furthermore, since we are going to work with surfaces, we expect that the data are highly correlated.

We suppose that our simulated surface is shaped as the central portion of the Matlab function `peaks`. Such function is a mixture of scaled and translated bivariate Gaussian probability density functions. Figure 3.4 gives a graphical representation of the surface of interest. The shape of such surface is quite complex due to the presence of peaks and valleys.

In order to simulate the low-resolution and high-resolution data sets, we add independent noise to the analytic form of the function. We evaluate the function `peaks` on a set of $n = 100$ randomly selected points for the LE data and on another disjoint set of $n_1 = 30$ points for the HE data. Because of the assumption that the high-resolution data $\mathbf{y}_{h_1}$ are highly accurate we simulate them by adding a very small noisy component to the exact value of the z coordinates. Similarly, in order to simulate the low-resolution data $\mathbf{y}_l$, we add a larger random error to the analytically computed z values. We also select a set of $n_{\text{test}} = 1\,000$ points for testing purposes. We are going to evaluate the predictive performance of the BHGP model at these points, with cross-validation. Moreover we are going to see if the use of the adjustment model significatively improves the performance of prediction when we use only the low resolution data.

Figure 3.5 provides a representation of the simulated LE and HE data with respect to the wireframe surface of the analytic function.

Note that in this case the experimental conditions corresponding to the LE and

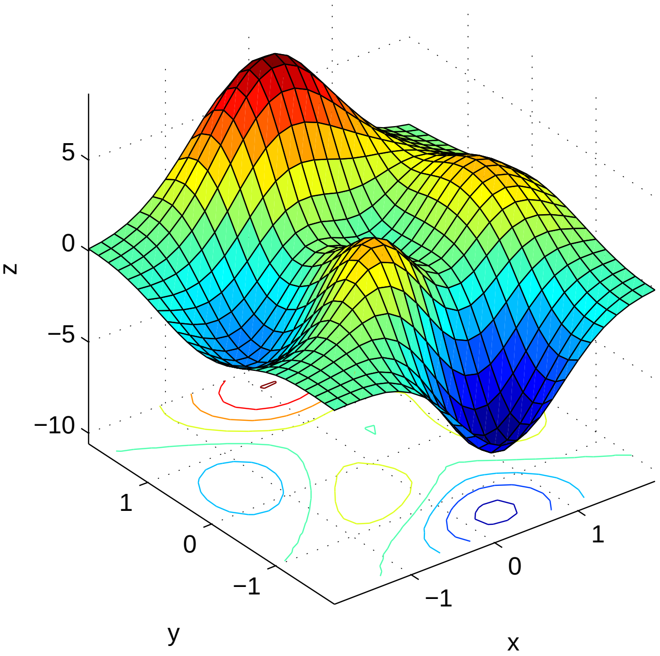

Figure 3.4: Graphical representation of the central portion of the Matlab function `peaks`.

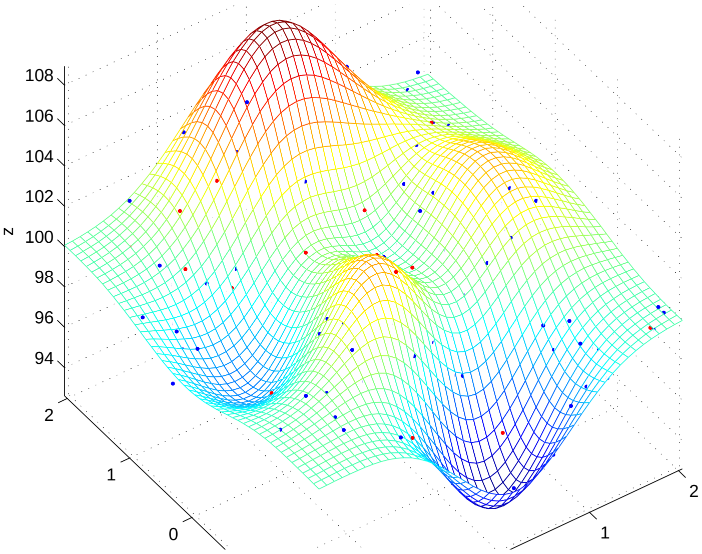

Figure 3.5: Representation of the simulated LE and HE data with respect to the wireframe surface of the analytic function.

HE, D_l and D_h respectively, are disjoint sets, thus we cannot apply the simplified predictive procedure illustrated in Section 2.4. We decide to adopt the two stage approach described in Section 2.7.1.

In order to obtain quick predictions of the missing low resolution $\mathbf{y}_{l_1}$ data we decide to predict them ($\hat{\mathbf{y}}_{l_1}$) using the universal kriging GP model (2.38) with the GPML Matlab code by [RN11] introduced in Appendix D, rather than applying the whole MCMC procedure illustrated in Section 2.7.1. The main drawback of this "naive" two-stage approach is that the resulting predictive density for the high resolution data does not take into account the prediction uncertainty of $\hat{\mathbf{y}}_{l_1}$, but only of the uncertainty due to the unknown adjustment model parameters.
Once the structure of our model is specified, we run the function `minimize` to compute the estimates of the unknown parameters of the model (2.38), by minimizing the negative marginal likelihood.
Then, given such estimated parameters and the low resolution data $\mathbf{y}_l$, the function `gp` is run. Such function evaluates the predictions of the missing LE data corresponding to the available n_1 HE data, i.e. $\hat{\mathbf{y}}_{l_1}^{\text{GPlowres}}$.
From now on, we treat the estimated $\hat{\mathbf{y}}_{l_1}$ as they were given. We point out again that this plug-in approach cannot count for the prediction uncertainty of $\hat{\mathbf{y}}_{l_1}$ in the further steps.

We proceed with the Bayesian empirical inferential analysis on the unknown parameters of the adjustment model (2.39).
In order to compute the values of the correlation parameters $\boldsymbol{\phi}_\rho$ and $\boldsymbol{\phi}_\delta$, we solve the optimization Problem (2.11), after applying some transformations analogous to the ones used in Section 3.2.4, and we get the following estimates:

$$\begin{aligned} \hat{\boldsymbol{\phi}}_\rho &= (0.0012,\, 0.0013), \\ \hat{\boldsymbol{\phi}}_\delta &= (0.3520,\, 0.0056). \end{aligned} \tag{3.24}$$

We sample from the joint posterior distribution of $(\rho_0, \sigma_\rho^2, \delta_0, \sigma_\delta^2, \sigma_\epsilon^2)$ using the usual Gibbs algorithm. We draw 10 000 posterior samples of the unknown parameters and discard the first 1 000.
The `coda` function provides the p-values for the Chi-squared test of the Geweke

convergence diagnostic:

$$\begin{aligned}
\text{p-value}(\rho_0) &= 0.387203\\
\text{p-value}(\sigma_\rho^2) &= 0.065893\\
\text{p-value}(\delta_0) &= 0.870070\\
\text{p-value}(\tau_1) &= 0.481181\\
\text{p-value}(\tau_2) &= 0.635662.
\end{aligned} \tag{3.25}$$

In spite of the auto-correlation of the simulated chains, according to the p-values of the Geweke diagnostic, convergence is achieved if we set the confidence level to 95%.

Table 3.6 summarizes the results of the posterior sampling: the posterior means, the posterior medians and the lower and upper 95% credibility HPD limits of the marginal posterior distribution of the parameters ρ_0, δ_0, σ_ρ^2, σ_δ^2 and σ_ϵ^2. Figure 3.6 represents the histograms of the values of the parameters sampled from their posterior distribution (non-normalized empirical marginal posterior distributions of the parameters).

	Posterior mean	Posterior median	LCL HPD (2.5%)	UCL HPD (97.5%)
ρ_0	0.8423	0.8513	0.5460	1.0912
σ_ρ^2	0.2933	0.2573	0.1111	0.6988
δ_0	-0.0533	-0.0445	-1.7638	1.4833
σ_δ^2	0.6994	0.5202	0.1748	2.2945
σ_ϵ^2	0.1218	0.1152	0.0700	0.2116

Table 3.6: Posterior means and medians of the parameters ρ_0, δ_0, σ_ρ^2 and σ_δ^2 and their respective 90% credibility HPD limits obtained with BHGP model (Simulated data - II).

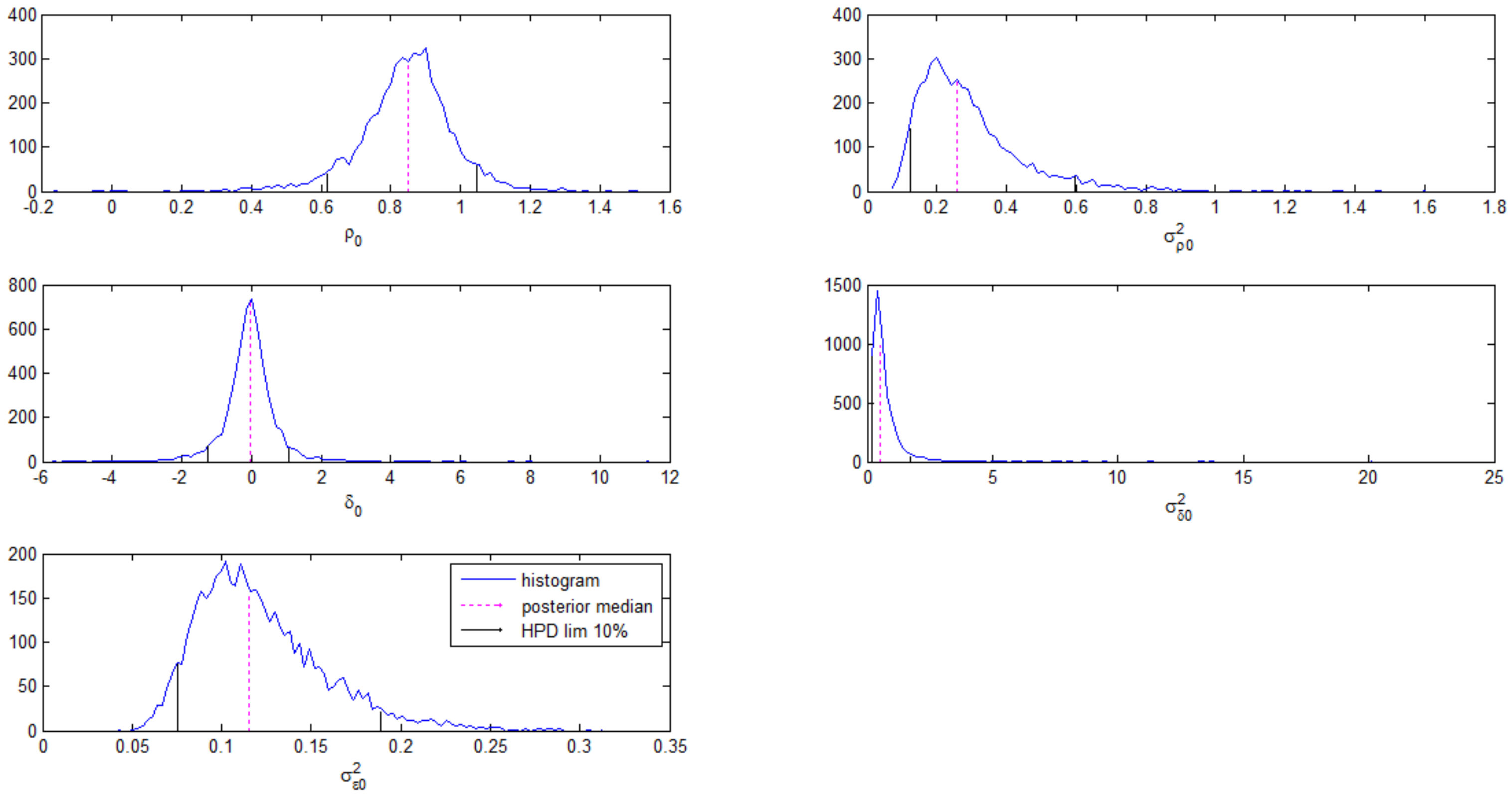

Figure 3.6: Histograms of ρ_0, δ_0, σ^2_ρ and σ^2_δ. The purple lines represents the posterior medians, the black lines the 90% credibility HPD limits (Simulated data - II).

Since we have a sufficiently large number of samples from the posterior distribution of the parameters, we can approximate the predictive density in the way described in the previous chapter and compute the predictions at the testing points and their respective 95% prediction limits. We will not report these results because $n_{\text{test}} = 1000$ predictions were computed. We provide instead a graphical representation of the cloud of the predicted points and the cloud of the corresponding points of the true function.

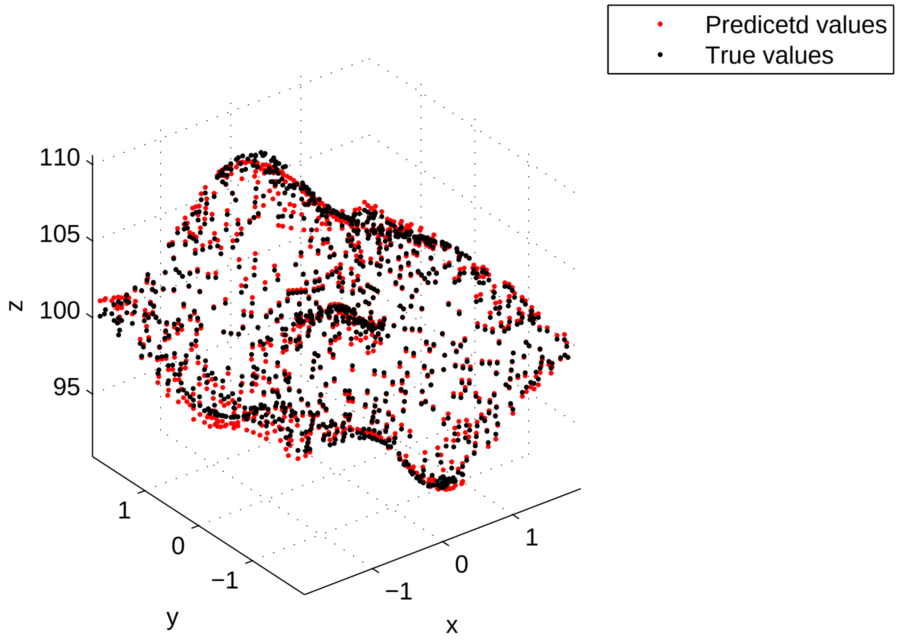

Figure 3.7: Representation of the clouds of the predicted points (red) computed using the BHGP model and the corresponding points of the true function (black) (Simulated data - II).

The Mean Square Prediction Error corresponding to such predictions is:

$$\text{MSPE}^{\text{QW08}} = 0.1155. \tag{3.26}$$

The superscript "QW08" indicates the results obtained with two-stage version of the BHGP model that integrates low-accuracy and high-accuracy data.

Another interesting graphical representation of the predictions is the one in

Figure 3.8. It displays the magnitude of the prediction error $y_{\text{test}} - \hat{y}_h^{\text{BHGP}}$ directly on the shape of the interpolated predicted surface.

As we expected, the prediction error is close to zero in those area of the surface where the high-accuracy data are placed. This makes us think that if we positioned the n_1 high-resolution data in the most critical areas of the surface, instead of randomly placing them, we would probably have had further improvement in the prediction with the BHGP model.

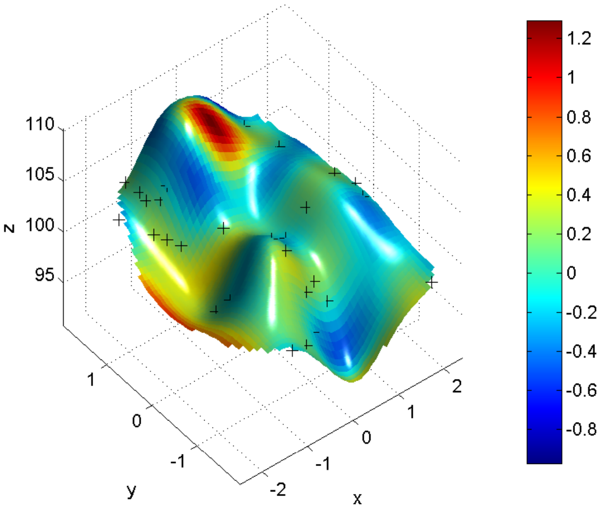

Figure 3.8: Interpolated error plot: it displays the magnitude of the prediction error $y_{\text{test}} - \hat{y}_h^{\text{BHGP}}$ on the shape of the interpolated predicted surface. Cold colors indicate negative prediction errors, i.e. the predicted points are overestimates of the testing measured points. Warm colors indicate positive prediction errors, i.e. the predicted points are "impossible" as they lie under the measured surface (model validation with simulated data - II).

3.4.1 Predictive performance comparison of different GP models

We finally carry out a comparison of the predictive performances of different models for the multi-resolution simulated data analyzed in the present section:

- "GPmodel": universal kriging GP model (2.38) that uses the low-accuracy data only;

- "GPmerge": universal kriging GP model (2.38) that uses both the low-accuracy data and the high-accuracy data as the indistinctly came from a unique source and had the same resolution;

- “QW06”: data fusion model [Qia+06] that integrates low-accuracy and high-accuracy data in the way described in Section 1.5.3;
- “QW08”: two-stage BHGP model [QW08] that integrates low-accuracy and high-accuracy data in the way described in Section 2.6.

The Mean Squared Prediction Errors computed with each one of these methods are respectively:

$$\begin{aligned} \mathrm{MSPE}^{\mathrm{GPlowres}} &= 0.1612 \\ \mathrm{MSPE}^{\mathrm{GPmerge}} &= 0.1522 \\ \mathrm{MSPE}^{\mathrm{QW06}} &= 0.1397 \\ \mathrm{MSPE}^{\mathrm{QW08}} &= 0.1155 \end{aligned} \tag{3.27}$$

The computed MSPEs suggest that models “QW06” and “QW08”, integrating multi-resolution data using two different GP adjustment models respectively, provide better prediction results.
To confirm this observation, we perform a series of statistical tests to verify that the prediction performances of the different methods are significatively different.

The Squared Prediction Errors (SPE) at every input point $\mathbf{x}_i^*$, $i = 1, ..., n_{test}$, of the testing set are computed as:

$$\mathrm{SPE}(\mathbf{x}_i^*) = (\mathrm{y}_{test}(\mathbf{x}_i^*) - \hat{\mathrm{y}}_h)^2, \quad i = 1, ..., n_{test} \tag{3.28}$$

where $\mathrm{y}_{test}(\cdot)$ is the measured high-resolution response used for cross-validation and $\hat{\mathrm{y}}_h$ is the estimated high-resolution response at the same input point using one of the above prediction methods.
We want to determine a suitable hypothesis testing procedure for evaluating the statistical significance of the difference of two different prediction methods according to some performance indicator, for instance a measure of centrality of the SPEs.
Assume we want to compare “Method 1” and “Method 2”. Then two populations of SPEs are available, $\mathrm{SPE}^{\mathrm{Method\,1}}$ and $\mathrm{SPE}^{\mathrm{Method\,2}}$. The test hypotheses to verify would be the following:

H_0: Measure of centrality of $\text{SPE}^{\text{Method 1}}$ is equal
to measure of centrality of $\text{SPE}^{\text{Method 2}}$.
Vs.
H_1: Otherwise.

Since we used the same testing set to compute the SPEs for each of the above methods, a test for paired observations would be appropriate. In this way we take into account the possible correlation between the SPEs corresponding to the same input point $\mathbf{x}_i^*$. As a matter of fact, if "Method 1" exhibits a high SPE at a given input point, we expect "Method 2" to exhibit a high SPE as well.
Paired tests can be seen as a kind of blocking technique, that allows to reduce the variance by comparing the SPEs "within" corresponding input points, rather than "across" different input points.

A paired t-test for testing the mean difference between paired observations would be our first choice, but such test needs the SPEs to be normally distributed. We perform the normality test on the pairwise differences of the SPEs computed with the different methods, but we cannot accept the hypothesis that such differences follow a normal distribution, not even after they were transformed using the Box-Cox power transformation.
Thus we have to rely on a non parametric test. We decide to use the One-Sample Sign Test on the differences of the SMPs. The hypotheses we want to test in our case are:

$$H_0\text{: median}(\text{SPE}^{\text{Method 1}} - \text{SPE}^{\text{Method 2}}) = 0.$$
Vs.
$$H_1\text{: median}(\text{SPE}^{\text{Method 1}} - \text{SPE}^{\text{Method 2}}) \neq 0.$$

Such test is implemented in Minitab and it is a valid non-parametric alternative to the paired t-test. Furthermore, it is robust to the lack of symmetry of the distribution of the differences.
We report the Minitab output corresponding to the following tests in the same order:

- H_0: median$(\text{SPE}^{\text{GPlowres}} - \text{SPE}^{\text{GPmerge}}) = 0$
- H_0: median$(\text{SPE}^{\text{GPlowres}} - \text{SPE}^{\text{QW06}}) = 0$
- H_0: median$(\text{SPE}^{\text{GPlowres}} - \text{SPE}^{\text{QW08}}) = 0$
- H_0: median$(\text{SPE}^{\text{GPmerge}} - \text{SPE}^{\text{QW06}}) = 0$

- H_0: median($\mathrm{SPE}^{\mathrm{GPmerge}} - \mathrm{SPE}^{\mathrm{QW08}}) = 0$
- H_0: median($\mathrm{SPE}^{\mathrm{QW06}} - \mathrm{SPE}^{\mathrm{QW08}}) = 0$.

```
Sign test of median =  0,00000 versus not = 0,00000

                            N  Below  Equal  Above       P   Median
diff(GPlowres_GPmerge)   1000    403      0    597  0,0000  0,00884
diff(GPlowres_QW06)      1000    400      0    600  0,0000  0,00942
diff(GPlowres_QW08)      1000    365      0    635  0,0000  0,01719
diff(GPmerge_QW06)       1000    449      0    551  0,0014  0,00052
diff(GPmerge_QW08)       1000    424      0    576  0,0000  0,00740
diff(QW06_QW08)          1000    433      0    567  0,0000  0,00670
```

The computed p-values allow us to conclude that there is no statistical evidence to support the hypothesis that the median of the pairwise differences is null. Furthermore the estimated median of the pairwise differences is always positive. This implies that, if we are testing the hypothesis H_0: median($\mathrm{SPE}^{\mathrm{Method\,1}} - \mathrm{SPE}^{\mathrm{Method\,2}}) = 0$, then the median of $\mathrm{SPE}^{\mathrm{Method\,2}}$ is significatively smaller than the median of $\mathrm{SPE}^{\mathrm{Method\,1}}$, i.e. "Method 2" determines better predictive performances than "Method 1".

Table 3.7 provides the ranking for the different prediction methods we considered according to the results of the paired Sign Tests we performed.

Pos.	Method	Median(SPE)
4	GPlowres	0.0461
3	GPmerge	0.0457
2	QW06	0.0456
1	QW08	0.0443

Table 3.7: Ranking of the compared prediction methods sorted by decreasing predictive performance according to the results of the Sign Tests performed on the differences of the SMPs (Simulated data - II).

As we expected methods "QW06" and "QW08", that combine low-accuracy data and high-accuracy data using an appropriate adjustment model, are the ones that lead to better prediction results in terms of Squared Prediction Errors.

Chapter 4

Large Scale Metrology: a real case study for multi-resolution data

In the present chapter we face the problem of integrating multi-sensor data in coordinate metrology. In particular we discuss an application of the Bayesian Hierarchical Gaussian Process Model in Large Scale Metrology.
In our case study the low resolution data are measured with an innovative optical-based metrological system (MScMS-II) and the high resolution data are measured with a traditional Coordinate-Measurement Machine (CMM).
First we introduce the matter of multiple sensor integration in coordinate metrology. Then we provide a detailed description of the architecture and working principles of the MScMS-II system for Large Scale Metrology applications, developed at the Industrial Quality and Metrology Laboratory of Politecnico di Torino, Italy. Finally we provide a synthetic description od the CMM measuring system.

4.1 Multi-sensor data integration in metrology

The capability to design and realize products with tight tolerances and satisfactory dimensional control has become a fundamental requisite in any industrial field. The fulfilment of such tasks is essential to meet the needs of a more and more competitive market, subject to continuous and rapid changes. For these reasons, high precision and rapid acquisition of coordinate measurements of parts and finite products with complex geometry has become an essential step in many industrial processes.

According to the definition given by [Wec+09] multisensor data fusion in dimensional metrology is:

> *(...) the process of combining data from several information sources (sensors) into a common represetational format in order that the metrological evaluation can benefit from all available sensor information and data.*[Wec+09]

The combination of multiple sensors mainly aims to improve the quality of measurement results and increase the amount of information carried by them.

[Wec+09] provides a review on some useful criteria to classify multisensor configurations.
Based on the characteristics of the information sources, sensors can be:

- homogeneous, if they are similar and are designed to capture the same (or comparable) physical measures;
- inhomogeneous, in the case the information acquired by distinct sensors is not directly comparable but requires pre-processing;

Based on the sensor configuration multisensor data combination can be:

- complementary, if the multiple sensor, being independent of each other, provides complete information on the measured object only after their respective measurement are combined;
- competitive, whether each sensor provides independent measurements of the same property, i.e. replicates of the same measurements acquired with different sensors are available;
- cooperative, when the measurements coming from independent sensors are used to derive information that would not be available from the sensors individually.

In dimensional metrology, data sets can be classified according to the their origin, source or physical characteristic they represent. Three significant examples are:

- intensity images;

- surface descriptions;
- volume data.

Finally the last classification criterion is related to the methodology for data merge:

- fusion across sensors, when a set of sensors measure the same property;
- fusion across attributes, when a set of sensors measure different properties associated with the same experimental environment;
- fusion across domains, when multiple sensors measure the same property over different domains of the working volume;
- fusion across time, in the case new measurements are merged with historical data.

Some other sensitive aspects of integrating multiresolution metrology data have to do with the two following topics:

- Pre-processing. Usually in metrology data fusion is applied at signal level, i.e. to the raw data (if the data are comparable and have consistent coordinate systems). In some specific cases mathematical manipulations of the data are needed.
- Registration. One critical aspect of multisensor data fusion is the problem of the alignment and transformation of the respective sensor coordinate systems into one common coordinate system. Typically the algorithms used in the registration process are based on Least Squares Criterion, i.e. on the minimization of the variance of distances of corresponding data points. In the process of registering 3D data set, a common practice to determine these corresponding data points is the application of markers.

[Wec+09] also provides an overview on different methods for geometric data acquisition. A simple classification of such methods is represented in Figure 4.1.

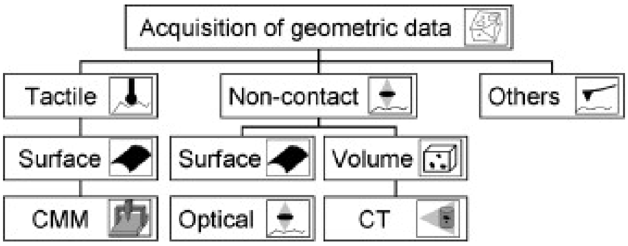

Figure 4.1: Methods of geometric data acquisition [Wec+09].

In the next sections we will focus our attention on two specific examples of coordinate measurement systems that are very different of each other: one, the MScMS-II, belongs to the optical sensor family, while the other, the CMM, uses a mechanical touching probe.

4.2 Definition and characteristics of Large Scale Metrology (LSM)

As previously stated, improving the accuracy of dimensional metrology is essential in many industrial applications.
Manufacturers in industrial sectors as automotive, aerospace, shipbuilding and railway industry have become increasingly aware of the necessity to meet the accuracy requirements for parts and final products. As a consequence, reliable and efficient large scale measuring systems for large-sized objects are needed as a support in many stages of the production process, for instance assembly, alignment and measurement inspection.

[Est+02] quotes the original definition of Large Scale Metrology (LMS) given by Puttock in 1978:

> *The field of large-scale metrology can be defined as the metrology of large machines and structures. The boundaries of this field are laboratory measurements at one end and surveying at the other. Neither boundary is well defined and [...] will generally be confined to the metrology of objects in which the linear dimensions range from tens to hundreds of meters.*

[GMP10a] also provides a concise but effective introduction on the matter of LSM. When measuring medium-size and large-size objects, traditional Coordinate-Measuring

Machines (CMMs) could be difficult to handle and have shown poor flexibility. For these reasons, the traditional CMM approach is overturned and rather than moving the object to the measuring machine, the measuring system is installed and arranged in the proximity of the object in order to suitably cover the working volume in which the object is positioned.
[Fra+09] lists a number of basic requirements of a LMS that are summarized in Table 4.1.

Requirement	**Description**
Portability	Capability of the system to be easily moved, easily assembled/disassembled, thus minimal weight and size of the system itself is desirable.
Flexibility	Capability of the system to perform different measurement tasks (i.e. determination of point coordinates, distances, curves, surfaces etc.) and be employed in different working environments.
Handiness	Simplicity of installation and rapid start-up and calibration times, before the system is ready to work; user-friendliness and intuitiveness of the software interface.
Metrological performance	Adequate metrological performances, in terms of stability, repeatability, reproducibility and accuracy [ISO 5725 1986].
Scalability	The capability of the system to cover different shaped and sized volumes with linear dimensions up to $30 - 60$ meters.
Low economic impact	It takes into account the product price plus the installation, training and maintenance costs.
Work indoor	The system should be able to work indoor (inside warehouses, workshops, or laboratories).

Table 4.1: Definition and description of LMS basic requirements [Fra+09], [GMP10a].

LSM is a challenging topic for Metrologysts and several measuring systems based on disparate technologies, such as optical, mechanical, electromagnetic and acoustic technologies, have been developed.
[Est+02] points out that optical-based systems have proven to be the most efficient and reliable in LMS applications, mostly thanks to the significant advances in optical and imaging technology and to the improvements in computational speed and precision.

[GMP09] describes the possible different classifications of such measuring systems according to:

- sensor layout (centralized or distributed);
- measurement operating conditions (contact or non-contact instruments);
- working principles (systems that use two angles and one length, systems that use multiple angles, i.e. *triangulation*, systems that use multiple lengths, i.e. *trilateration/multilateration*).

A review on a set of existing optical-based solutions for LMS, like laser tracker-based systems, theodolites and total stations, digital photogrammetry-based systems, indoor-GPS, is presented in [GMP10a]. For more details on such systems refer to the literature suggested by Galetto, Mastrogiacomo, and Pralio.

After this concise introduction to indoor Large Scale Measurement, we focus our attention on the MScMS-II.

4.3 The MScMS-II system

In some of their most recent publications ([GMP09], [GMP10a], [GMP10b]), Galetto, Mastrogiacomo, and Pralio describe an innovative InfraRed (IR) optical-based distributed measuring system "designed to perform low-cost, simple and rapid coordinate measurements of large-sized objects exploiting the principles of photogrammetry". Such system is also called MScMS-II, that stands for Mobile Spatial coordinate Measurement System - II.
MScMS-II has been developed at the Industrial Quality and Metrology Laboratory of Politecnico di Torino, Italy, and it was presented for the first time at the ASME International Manufacturing Science and Engineering Conference, West Lafayette, IN, in 2009.

An earlier prototype of MScMS that exploited UltraSound (US) technology was developed by the same research team. Anyway the poor characteristics of US devices led to unsatisfactory results and inaccurate measurements. In order to improve the measuring performances, an enhanced version of the system based on InfraRed technology was developed.
With respect to the existing systems the MScMS-II has some innovative characteristics, that are reported in Table 4.2.

Property	**Description**
Scalability	Capability to extend the measurement domain in order to cover large and geometrically complex working volumes by properly distributing the network sensors.
All-around visibility	Capability of the measuring probe to reach every part of the object from any side.
Wireless connection	The sensors are connected to the processing unit with a Bluetooth connection.
Layout optimization	Capability of the system to automatically suggest the optimal sensor positions in order to efficiently cover the working volume
Cooperation of blocks of sensors	It allows to optimize point acquisitions, system auto-diagnostics, and power consumption.
Sensor fusion	Capability to integrate the metrological system with other spatially distributed sensors (of temperature, humidity, vibrations, light intensity, etc.) in order to provide environmental mapping of the working volume and monitor the operating conditions of the system for auto-diagnostics or self-calibration.

Table 4.2: Innovative technical and operational characteristics of the MScMS-II [GMP10b].

According to the requirements previously described in Table 4.1, [GMP10a] presents a qualitative comparison of different optical-based distributed system for LMS (Table 4.3).

4.3.1 Architecture of the MScMS-II

The MScMS-II is composed of three basic units (Figure 4.2):

- a network ("constellation") of wireless sensors;
- a mobile wireless and armless probe;
- a data processing system.

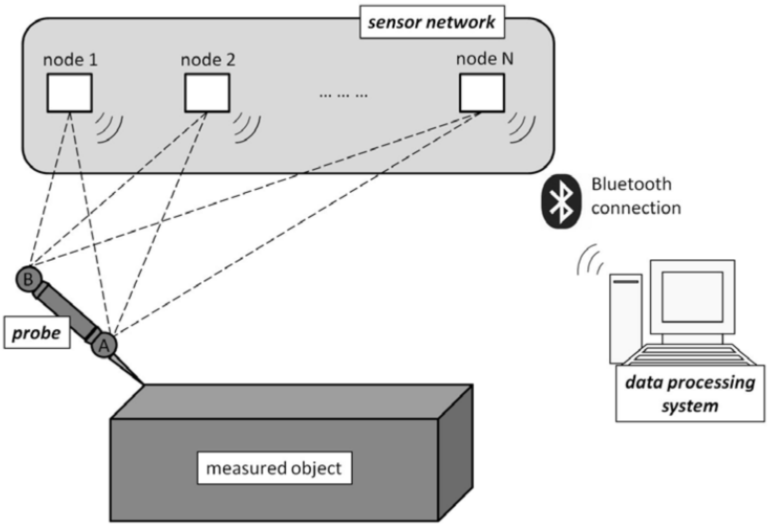

Figure 4.2: MScMS-II architecture. The dashed lines represent visual links between sensor nodes and retro-reflective markers (indicated as A and B) of the handheld probe. The Bluetooth connection is established between each node and the processing system [GMP10a].

The *network of sensors* is composed of at least three nodes. Such sensors are suitably distributed in order to adequately cover the measurement volume.
Each camera can track up to four IR sources (IR spots) in real time and it records the 2D position of the IR spots (in our case a couple of passive reflective markers at the probe extremities) in the view plane of the camera itself. Then a localization algorithm implemented on the processing unit is used to estimate the 3D coordinates of the measured point.

System	**Portability**	**Flexibility**	**Handiness**	**Scalability**	**Metrological performace**	**Purchasing (cost k€)**
Laser tracker	medium	medium	medium	low	high	80 - 150
Laser radar	medium	medium	medium	low	high	400 - 500
Digital photogrammetry	high	medium	high	medium	high	20 - 100
Indoor GPS	high	high	medium	high	high	$> 150^{(*)}$
Theodolite/ Total station	high	low	high	high/medium	low	1 - 4
MScMS II	high	high	high	high	medium	$> 3^{(*)}$

The last column reports a rough estimation of the economic impact (referred to the purchasing cost), expressed in k€. The wide range of variation of costs is related to the fact that different manufacturers offer metrology instruments with different performance levels and accessories.

(*) It has to be noted that the economic impact of the two distributed systems (indoor GPS and MScMS-II) is strongly related to the network sizing, i.e., the number of remote sensor devices. The reported values refer to the minimum number of sensors needed to perform network calibration (i.e., three sensing units).

Table 4.3: Qualitative comparison of optical-based distributed systems for large-scale metrology and quantitative estimate of their economic impact [GMP10a].

The prototype of the MScMS developed at Politecnico di Torino uses low-cost IR cameras. In order to work with passive markers, each camera is coupled with a near IR light source as represented in Figure 4.3.
Table 4.4 summarizes the technical specifications of the sensor components.

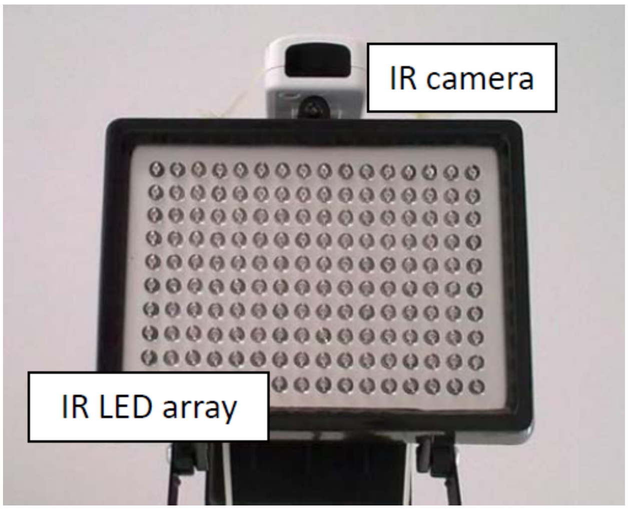

Figure 4.3: IR sensor: an IR camera is coupled with an IR LED array to locate passive retro-reflective targets [GMP10a].

Component	Characteristic	Specifications
IR camera	Interpolated resolution	1024 × 768 pixels
	Native resolution	128 × 96 pixels
	Max sample rate	100 Hz
	Field of view (FOV)	45° × 30°
IR light source	Nr. of LED per array	160
	Peak wavelength	940 nm

Table 4.4: Technical characteristics of the sensors in the MScMS-II [GMP10b].

The overall sensor set, composed of both the IR camera and the LED array, weights approximately 500 g and its sizes are 13 × 13 × 15 cm.
The IR sensor configuration has to be set according to the size and shape of the object to be measures and the characteristics of the working environment.
Figure 4.4 represents a virtual reconstruction of the working layout with a six-camera configuration.

The *mobile probe* is represented in Figure 4.5 and consists of a rod with two passive reflective markers. Their centers are identified with the letters A and B.

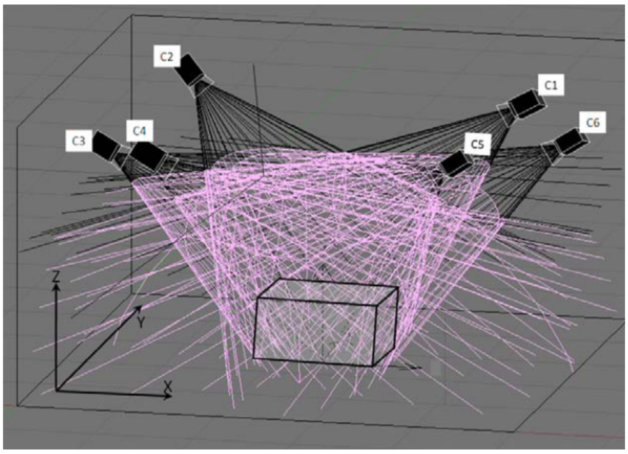

Figure 4.4: Virtual reconstruction of the working layout. The black lines represent the camera " field of sensing" and the pink lines identify the working volume that, according to the triangulation principles, is the volume of intersection of the field of sensing of at least two cameras.[GMP10a].

At one end there is a needle, whose tip, marked with V, is lined with A and B and physically touches the measured points.

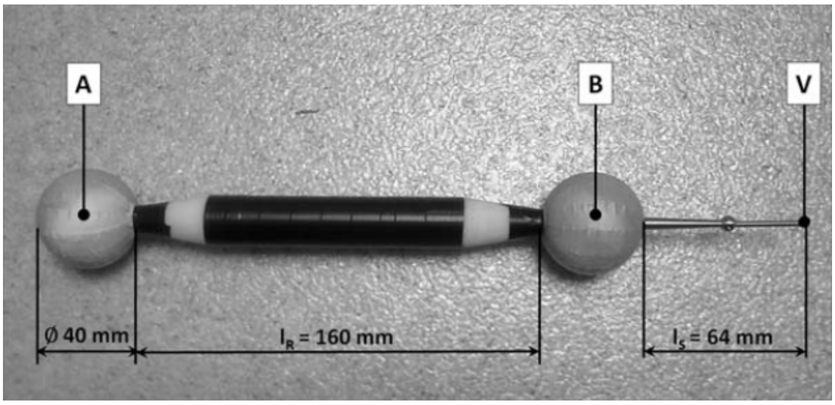

Figure 4.5: Mobile measuring probe. [GMP10a].

The passive markers are two polystyrene spheres wrapped with a retro-reflective silver transfer film. Their dimension depends on the working volume of the measuring systems and on the hardware specifications of the used instruments. For the prototype system with the characteristics described in Table 4.4, it has been demonstrated that two markers with diameters of 40 mm can be detected at a maximum distance of 6 m.

The probe has a very simple design and it is light and handy. It is usually handled

by an operator, who is free to move around the object and “touches” the points to be measured with the probe tip. It could also be fastened to another autonomous agent, such as ground or aerial robot. The probe is designed to allow all-around visibility, i.e. it makes possible to reach any side of the object and measure it without the limitations of a mechanical arm.
Referring to the notation of Figure 4.5, the spatial coordinates of the point $\mathbf{x}_V = (x_V, y_V, z_V)$, located on the probe tip, are univocally determined by the following linear equation:

$$\mathbf{x}_V = \mathbf{x}_A + \frac{(\mathbf{x}_B - \mathbf{x}_A)}{\parallel \mathbf{x}_B - \mathbf{x}_A \parallel} \cdot d_{V-A}, \tag{4.1}$$

where $\mathbf{x}_A = (x_A, y_A, z_A)$ and $\mathbf{x}_B = (x_B, y_B, z_B)$ are the coordinate of the reflective markers detected by the sensors. $d_{V-A} = \parallel \mathbf{x}_V - \mathbf{x}_A \parallel$ is known a priori as it depends on the geometry of the probe. It has to be pointed out that a further correction on the coordinates of V must be introduced because of the non-punctiform shape of the tip.

The *data processing system* allows to acquire and elaborate data sent by each network node. The sensor nodes and the processing unit are connected via Bluetooth.
The 2D coordinates of the IR spots in the view plane of each camera are transmitted form the sensor network to the processing unit. As previously stated, the IR sensors can track the IR spots in real time so the processing unit is spared the computational effort of performing the image analysis and identifying the coordinates of the IR-spot.
Since the connection is based on a Bluetooth link, the cameras are sequentially sampled and image synchronization is needed for 3D reconstruction. This is a critical issues as it affects the performances of the 3D reconstruction process due to acquisition delays. For the prototype system configuration a maximum number of six-cameras for processing unit is allowed. It was also proved that with an acquiring rate of 50 Hz the delay has a negligible influence on the measurement results.
The processing software implements:

- layout evaluation, designing and analyzing sensor network configurations;
- system calibration, providing position, orientation and technical parameters of sensors;
- 3D point localization;

- data elaboration procedures.

Figure 4.6 illustrates the data processing system.

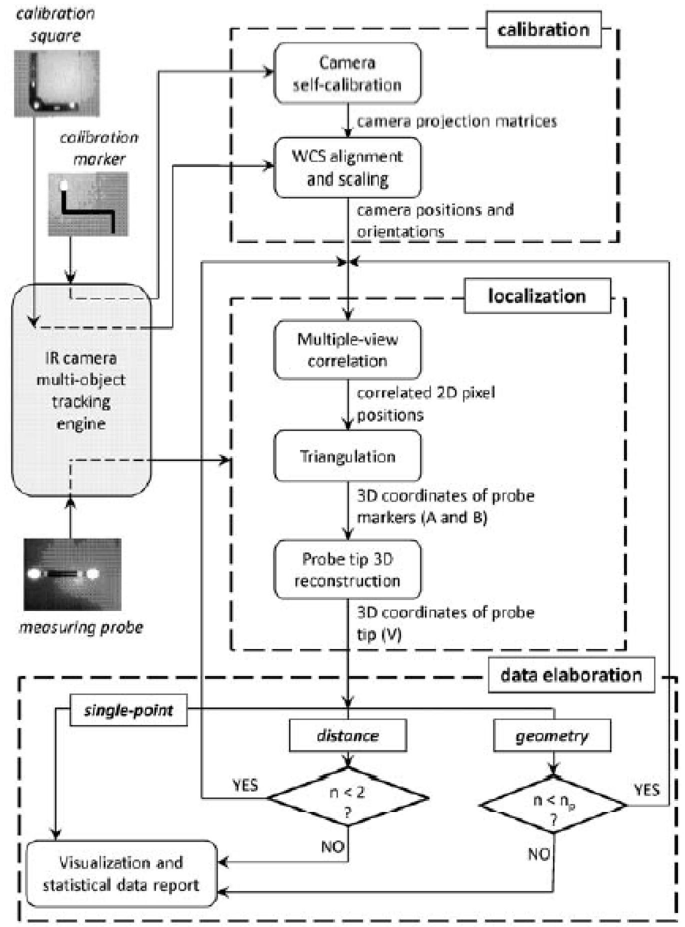
calibration square
calibration marker
IR camera multi-object tracking engine
measuring probe
calibration
Camera self-calibration
camera projection matrices
WCS alignment and scaling
camera positions and orientations
localization
Multiple-view correlation
correlated 2D pixel positions
Triangulation
3D coordinates of probe markers (A and B)
Probe tip 3D reconstruction
3D coordinates of probe tip (V)
data elaboration
single-point
distance
geometry
YES
n < 2 ?
NO
$n < n_p$?
YES
NO
Visualization and statistical data report

Figure 4.6: Scheme of the data processing system. The calibration procedure is responsible for determining positions and orientations of the IR sensors within the working environment. The localization procedure, implementing a triangulation method, reconstructs the 3D coordinates of the touched point by locating the passive markers on the measuring probe. A further step of data elaboration is implemented to coordinate the data processing operations (acquisition and elaboration), according to the purpose of the measurements (single-point, distance or geometry reconstruction). In this process, n is the number of measured points and n_p is the number of points needed to perform the processing. [GMP10a].

4.3.2 Localization algorithm and calibration

The *localization problem* can be stated as follows: given a camera layout, i.e. n_c cameras with known technical specifications, positions and orientations, focused on m markers, for each m-uple of 2D pixel coordinates $\boldsymbol{u}_{ij} = (u_{ij}, v_{ij})$, with $i = 1, ..., n_c$ and $j = 1, ..., m$, the 3D coordinates of the corresponding m markers are to be determined. Figure 4.7 gives a graphical representation of such problem where a 4-camera setup ($n_c = 4$) is used to reconstruct the 3D coordinates of two markers ($m = 2$).

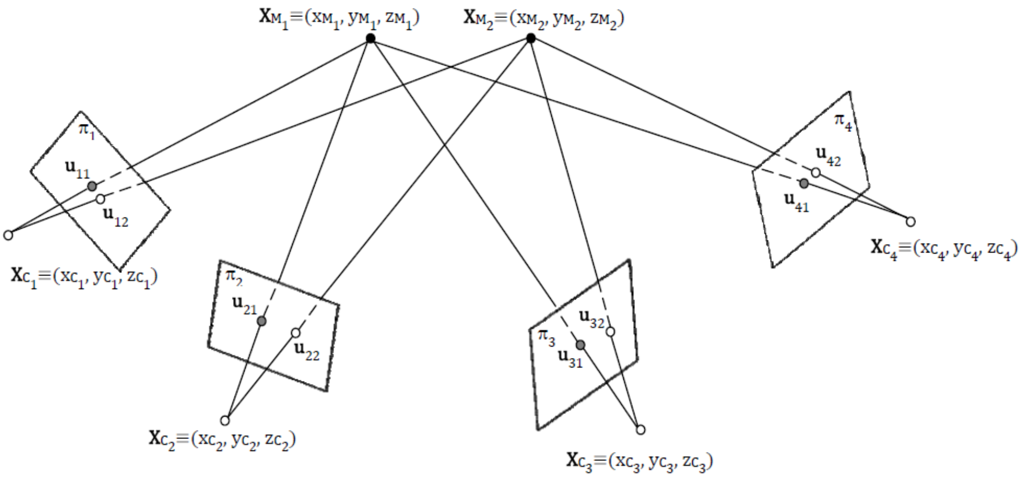
$\mathrm{X}_{M_1} \equiv (x_{M_1}, y_{M_1}, z_{M_1})$
$\mathrm{X}_{M_2} \equiv (x_{M_2}, y_{M_2}, z_{M_2})$
π_1
$\mathbf{u}_{11}$
$\mathbf{u}_{12}$
$\mathrm{X}_{C_1} \equiv (x_{C_1}, y_{C_1}, z_{C_1})$
π_2
$\mathbf{u}_{21}$
$\mathbf{u}_{22}$
$\mathrm{X}_{C_2} \equiv (x_{C_2}, y_{C_2}, z_{C_2})$
π_3
$\mathbf{u}_{32}$
$\mathbf{u}_{31}$
$\mathrm{X}_{C_3} \equiv (x_{C_3}, y_{C_3}, z_{C_3})$
π_4
$\mathbf{u}_{42}$
$\mathbf{u}_{41}$
$\mathrm{X}_{C_4} \equiv (x_{C_4}, y_{C_4}, z_{C_4})$

Figure 4.7: Graphical representation of the localization problem when a setup of four cameras ($n_c = 4$) is used to reconstruct the 3D position of two markers ($m = 2$). $\mathbf{x}_{ci}$ (with $i = 1, , 4$) and $\mathbf{x}_{Mj}$ (with $j = 1, 2$) refer to the 3D coordinates of the i-th camera center and the j-th marker, respectively. Point $\mathbf{u}_{ij}$ represents the 2D projection of $\mathbf{x}_{Mj}$ onto the image plane of the i-th camera. It corresponds to the intersection of the camera view plane π_i with the projection line of $\mathbf{x}_{Mj}$ (i.e., the line passing through the 3D point and the camera center) [GMP10a].

The localization algorithm follows the fundamentals of digital photogrammetry and can be summarized in the following two steps:

1. The correspondences among pixels in different image views are found using epipolar geometry.

2. The 2D information of different camera views are matched in order to recover the spatial coordinates for the 3D point using triangulation.

For what concerns the *multi-camera calibration problem*, a fully automatic self-point self-calibration technique is adopted.

A discussion of the localization and calibration problems goes beyond the purpose of the present work, hence for an extensive description of these two problems and the corresponding implemented algorithms refer to [GMP10b].

4.3.3 Uncertainty evaluation: preliminary tests results

Here we report the results of the preliminary uncertainty evaluation of 3D point coordinates, that have been performed in order to evaluate the metrological potentiality of the MScMS-II. The Multivariate Law of Propagation of Uncertainty (MLPU) has been used for this purpose.

The main contributions to overall uncertainty of 3D point coordinates can be summarized as follows:

- uncertainty of 2D point coordinates, which refers to the 2D pixel coordinates of point projection in the image plane;
- uncertainty of camera calibration parameters, which is associated with the internal and external camera parameters obtained in the calibration phase;
- camera synchronization error, which is considered negligible in static conditions (consideration would be necessary for a dynamic approach, i.e. in case of point tracking);
- uncertainty of 3D point coordinates, which can be traced back to the triangulation algorithm for 3D point reconstruction;
- uncertainty of probe tip coordinates, which actually determines the uncertainty of the point coordinates measured by the MScMS-II.

A set of preliminary tests has been performed in order to investigate the performance of the overall system, including the distributed sensor network, the measuring probe, and the data processing system. It has to be noted that the experimental results are strongly related to the network configuration, in terms of number of IR cameras, and their positions and orientations. The results reported hereafter have been obtained by using a set of six IR cameras, arranged in a working environment similar to the one shown in Figure 4.4. The resulting measurement volume was

about 2.0 × 2.0 × 2.0 m wide. A sampling frequency of 50 Hz has been used for data acquisition.
The system has been evaluated through stability, repeatability, and reproducibility tests and characterized by a preliminary estimation of the measurement accuracy according to the international standards (*VIM - International Vocabulary of basic and General Terms in Metrology*, International Organization for Standardization, Geneva, Switzerland, 2004).

4.4 Coordinate-Measuring Machine system

As previously mentioned, the high-resolution data are acquired with a Coordinate-Measuring Machine, the *CMM DEA Iota 0101* represented in Figure 4.8.

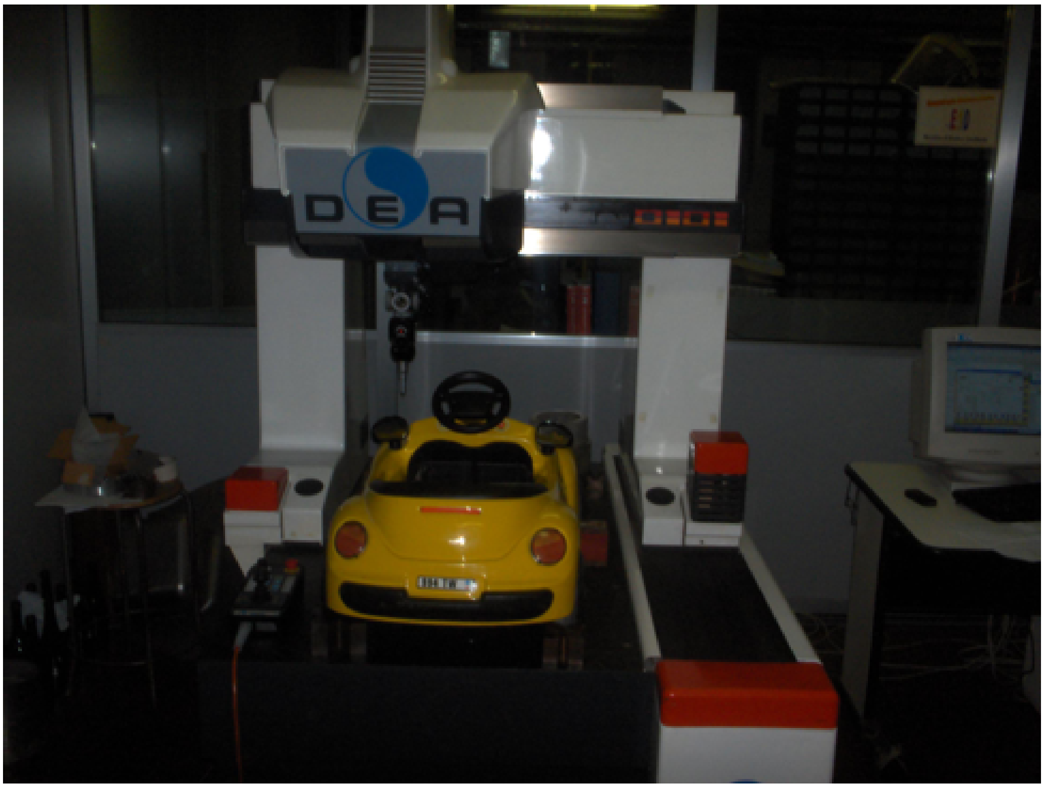

Figure 4.8: Coordinate-Measuring Machine *CMM DEA Iota 0101* used at the metrology laboratory of DISPEA, Torino. [Fra+10].

Such device is a more *traditional* and better known measuring system than the MScMS-II system, so here we provide only a brief introduction to its main features and working principles.

A Coordinate-Measuring Machine (CMM) is a device for measuring the geometrical characteristics of an object. The measurements are the spatial coordinates of points on the surface of the measured object.
A CMM is typically composed by the following parts:

- a main structure, that includes a gantry type superstructure (often called a bridge) and a working deck,
- a measuring probe, that can be tactile (i.e. mechanical probe) or non-contact (i.e. optical or laser probe),
- an electronic system for control and data collection.

The *CMM DEA Iota 0101* has overall dimensions of 1 500 mm × 1 200 mm × 2 800 mm and weight of 3 t ca.
Its bridge structure allows the probe to move along three axes of motion that delimit a measuring volume of 590 mm × 590 mm × 440 mm.
The CMM is equipped with a motorized mechanical probe head *Renishaw PH10M* that holds a steel needle, called stylus. The small ball at the end of the stylus is made of synthetic ruby and it is characterized by high hardness. The probe moves towards the surface of the measured object on an orthogonal direction to the surface itself and the approaching speed needs to be slow enough to avoid mechanical deformation of the surface. As the ruby ball touches the surface, the stylus deflects and the data processing system simultaneously elaborates the coordinates of the measured spot.
Though the *CMM DEA Iota 0101* was originally manually controlled by an operator, it was later equipped with a numerical control system.
Table 4.5 summarizes the technical specifications of the *CMM DEA Iota 0101*.

Property	**Description**

Measuring volume	X = 590 mm
	Y = 590 mm
	Z = 440 mm

Length measuring uncertainty according to VDI/VDE 2617 (2,1) guidelines in μm (including probe head). Repeatability U96 ($\pm 2 = 95\%$).	U1 = 4 + 4L/1000, U3 = 4 + 5L/1000
Probe head	Renishaw PH10M
Overall dimensions	1 500 mm × 1 200 mm × 2 800 mm
Weight	ca. 3 t

Table 4.5: Technical characteristics of the *CMM DEA Iota 0101* used at the metrology laboratory of DISPEA, Torino.

The CMM is a very precise measuring system, as mechanical probes can have resolutions with order of magnitude of $0.5\,\mu$m and they are very robust [SHM00]. Furthermore, it is a numerically controlled device, so it significatively reduces the occurrence of measurement errors caused by the operator.

Speaking in terms of Large Scale Metrology, the CMM has some main drawbacks when compared to the MScMS-II:

- *Non Portability.* The CMM is extremely heavy and its large dimensions make transportation uneasy. Usually, once it is installed at one site, it is rarely moved.
- *Limited measuring volume.* The measuring volume of the CMM is limited by the dimensions of the axes of motion and by the mechanism that handles the probe.
- *Slow acquisition speed.* The CMM has long acquisition times due to the required low approaching speed of the probe to the object and the slow digitization speed .
- *High cost.* Industrial CMM with large working volumes, manufactured to measure large-sized objects, are extremely expensive.

Chapter 5

Gaussian process models applied to multi-resolution Large Scale Metrology data

In the present chapter we show an application of the Bayesian Hierarchical Gaussian Process (BHGP) model to a set of metrology data with the purpose of modeling the surface of an object.

The same portion of the surface of a relatively small toy car was acquired with both the Mobile Spatial Coordinate Measuring System II (MScMS-II) and the Coordinate Measuring Machine (CMM) described in the previous chapter. Thus two sets of coordinate data, or point clouds, are available.

Given the notation introduced in Chapter 2 and the different characteristics and technological features of the two coordinate measuring systems, we are allowed to label the data acquired with the MScMS-II as low-resolution data and the data acquired with the CMM as high-resolution data. We point out that the combination of data from the MScMS-II and from the CMM does not reflect a real existing need in manufacturing industry, but it rather provides a good case study to prove how the BHGP works.

Furthermore, we are going to compare the predictive performances of different GP models applied to the available multi resolution metrology data.

5.1 Measurement activities

The measurement tests with both the MScMS-II and the CMM measuring systems were performed at the metrology laboratory of DISPEA (Dipartimento Sistemi di Produzione) at Politecnico di Torino.

Here we report only some information about the measured object and the adopted procedures. All the details about the measurement activities are provided in the technical report [Fra+10].
The measured object is the small toy car represented in Figure 5.1. Its maximum size is 507 × 350 × 912 mm including the tyres but not the steering wheel.

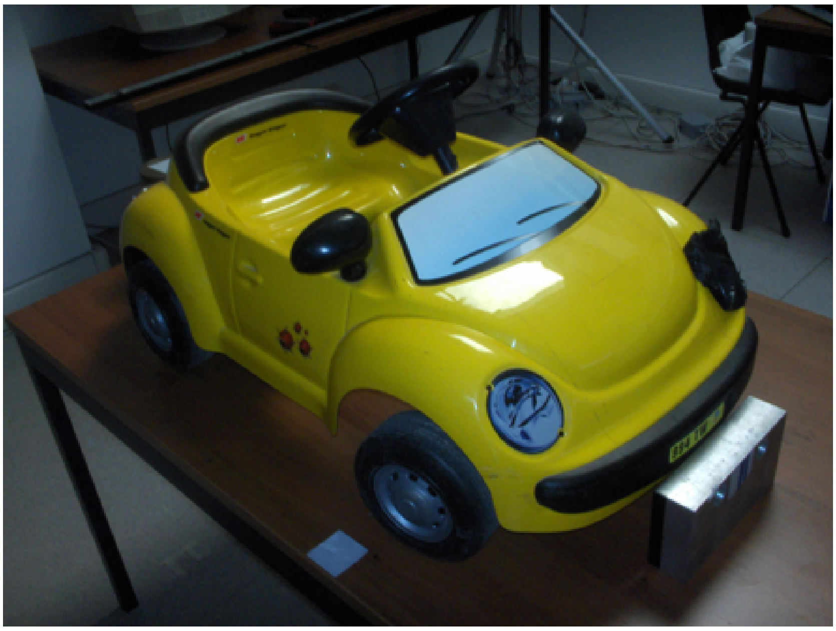

Figure 5.1: The measured object [Fra+10].

Measurements of the surface of the car front were acquired, according to the scheme shown in Figure 5.2. The measured portion of the hood is delimited by the red lines and includes the windshield, the cental part of the hood and the central part of the front bumper. The points corresponding to the spots labeled with numerical tags were used as markers in the registration process of the point clouds acquired with the two measuring systems.

We point out that the surface of interest is quite complex and presents several critical spots. The central part of the hood is smooth and quite regular, but the area where the hood and the windshield join introduces a discontinuity in the

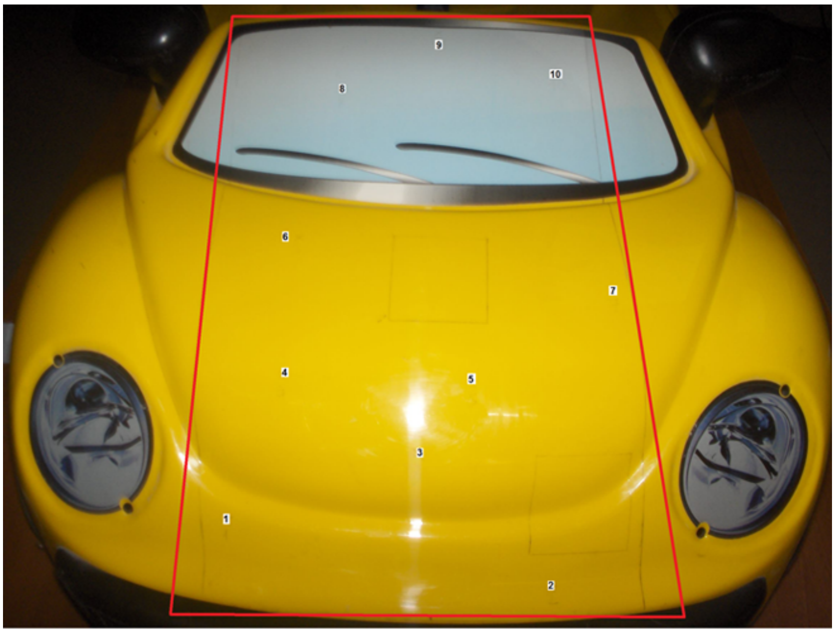

Figure 5.2: The detail of the measured surface of the hood and the front bumper [Fra+10].

curvature of the surface. Furthermore the lower front of the object, where the front bumper is, presents several changes in shape and curvature.

After the MScMS-II system was suitably calibrated, the coordinate system of the instrument was aligned to a user-defined room coordinate system, with the help of a reference calibrated artifact. Then 910 measurements of randomly selected points on the surface delimited by the red lines in Figure 5.2, were acquired with the MScMS-II system.
A coordinate-measuring machine *CMM DEA Iota 0101* was used to acquire the high resolution data. After the calibration process was completed, 1 243 points of the surface of the hood were measured.
Ten further reference points (markers) were measured at the same positions with both the measuring systems. Such markers will be used to register the point clouds acquired in a unique common coordinate system.
Figure 5.3 represents the non-registered point clouds and their respective markers.

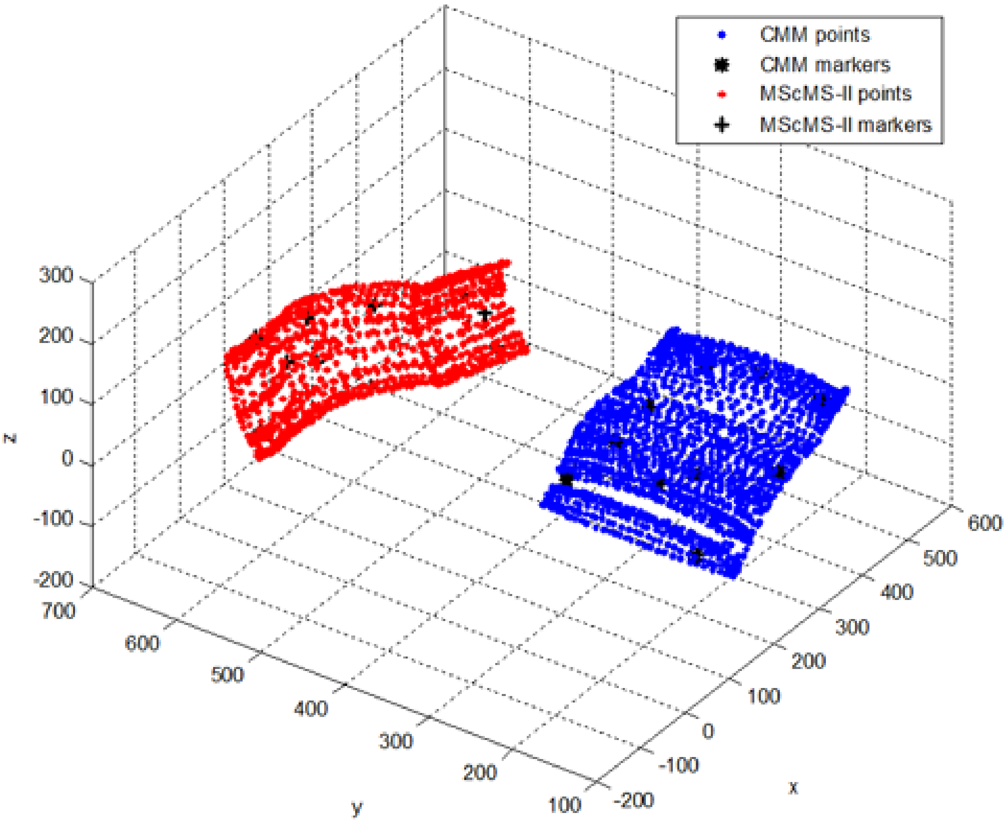

Figure 5.3: Representation of the point clouds of the measurements acquired with the MScMS-II system (red) and the CMM (blue). Their respective markers are represented as black Xs and stars.

5.2 Registration

In order to register the two sets of measurements, we use the Matlab function `procrustes` on the ten reference points. Such function takes two matrices as inputs and it determines a linear transformation that, when applied to the points in the second matrix, conforms them to the points in the first matrix. The goodness-of-fit criterion is the sum of squared errors.
We use `procrustes` in the following form:

```
[a, b, tr] = procrustes(rifCMM,rifIR,'Scaling',false,'Reflection',false);
```

i.e we compute the linear transformation without scaling and reflection, in order to align the points acquired with the MScMS-II to the ones acquired with the CMM. Figure 5.4 represents the registered point clouds.

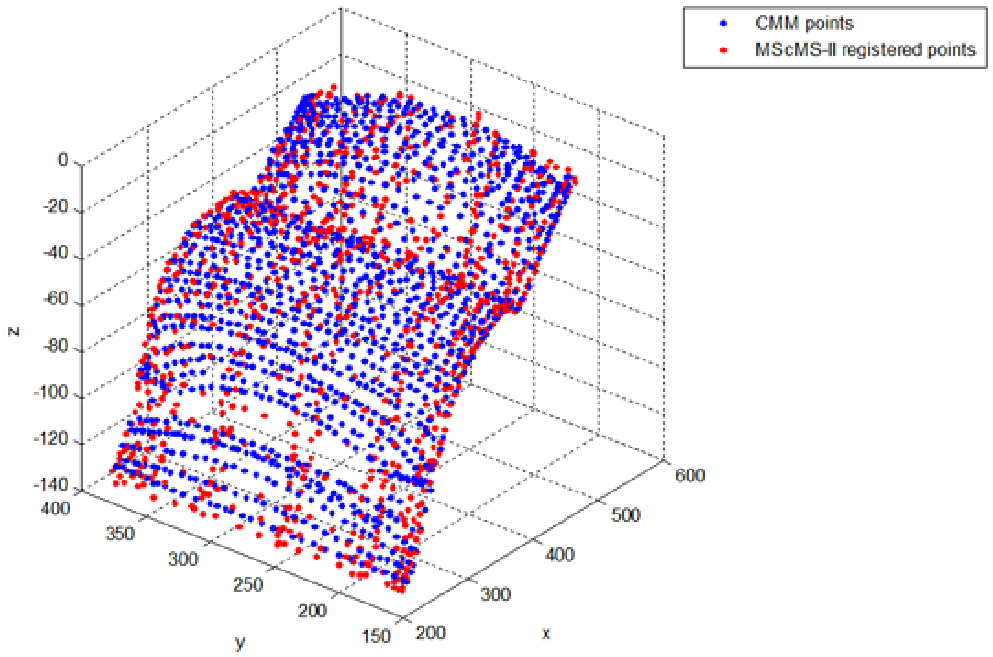

Figure 5.4: Representation of the point clouds of the measurements acquired with the MScMS-II system (red) and the CMM (blue) after the registration procedure with the Matlab function `procrustes`.

Now that the low-resolution and high-resolution data are represented in a common coordinate system, we are ready to apply the BHGP model described in the previous chapters.

5.3 Justification of the application of BHGP on metrology data

The BHGP model appears to be an adequate tool to describe metrology data for a few reasons.

The application of the BHGP model to multi-resolution metrology data implies that $k = 2$, i.e. two input variables $\mathbf{x}_1$ and $\mathbf{x}_2$ are available. As a matter of fact our data consist of "clouds" of points that represent the 3D coordinates of the surface of interest. Thus the input variables are the x and y coordinates of the measured point, while the response of interest is the height (the z coordinate) of the surface point corresponding to the given input variables.

More importantly, Gaussian process models are very popular in spatial Statistics when dealing with intrinsically highly correlated data. When magnified, the surface of an object reminds of a geographical surface. This analogy provides an intuitive justification for the use of a spatial Statistics model to represent the surface of a manufactured object [XDW07].
In Section 1.5.1 we introduced the Gaussian correlation function (1.4) for the GP $\epsilon_l(\cdot)$:

$$R_{\boldsymbol{\phi}}(\mathbf{x}_j, \mathbf{x}_m) = \exp\left\{ -\sum_{i=1}^{k} \phi_{li}(x_{ji} - x_{mi})^2 \right\}, \quad j, m = 1, ..., n,$$

where the power exponential is set to two (because it makes the correlation function continuous and infinitely differentiable at the origin) and the scale correlation parameters control the activity of the correlation as a decreasing function of the distance of two points. This assumption on the correlation structure seems reasonable in our case study.

Finally, we point out that we are in the case $D_l \cap D_h = \varnothing$, i.e. there is no perfect correspondence between the input points of the low-resolution data and the input-points of the high-resolution data. As a matter of fact, the high-resolution data were measured with a CMM, where the movements of the mechanical arm to which the touching probe is fastened, are numerically controlled. The low-resolution data were instead acquired using a measurement system where the probe is manually handled by an operator. This means that there is no perfect match between the position of the low-resolution and the high-resolution data and the adjustment model for the HE response (2.39) cannot be directly employed. So we rely on the two approaches illustrated in Section 2.7.

5.4 Downsampling

A very large amount of data is available (910 MScMS-II data and 1 243 CMM data), thus we decide to thin the data set in order reflect a more realistic situation. Furthermore, as we explained in the previous chapter, low-resolution MScMS-II are usually easier and quicker to acquire than the high-resolution CMM data.
In order to compare our results to the ones obtained in the paper in progress [CP11], we work with the following data percentage configurations:

- Data configuration 1 - 50% MScMS-II data and 2% CMM data ($n = 455$, $n_1 = 25$),
- Data configuration 2 - 50% MScMS-II data and 5% CMM data ($n = 455$, $n_1 = 62$),
- Data configuration 3 - 50% MScMS-II data and 10% CMM data ($n = 455$, $n_1 = 124$).

We point out that we reduce the size of the data using the following deterministic downsample procedure: starting from the first data, we keep one data every $1/r$, where r is the percentage of data we want to keep. In this way we are fairly sure that our thinned data sets uniformly cover the whole surface of the hood.
The remaining CMM data ($n_{test} = 1\,243 - n_1$) in all the three data configurations are used as testing sets for cross-validation of the prediction results computed with the different models we are going to analyze.
Figures 5.5 represents the 3D scatter plots of the three data configuration described above.

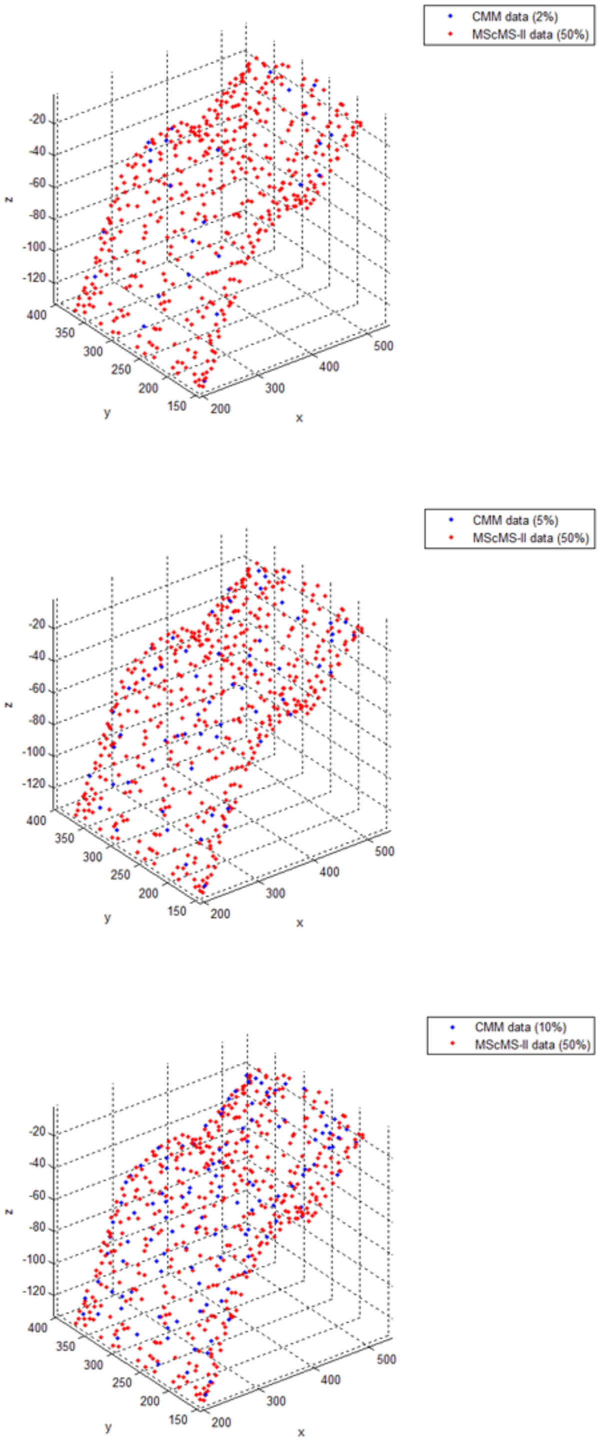

Figure 5.5: Representation of the point clouds acquired with the MScMS-II (red) and the CMM (blue) in the three data configurations.

5.5 Application of the Bayesian Hierarchical Gaussian Process model

In order to apply the BHGP model proposed in [QW08] (see Section 1.5.4), we follow the same steps we thoroughly described in Chapter 3 for all the three data configurations defined above.

We set the hyperparameters of the hyperpriors to the same values we used in 3, because they previously proved to work well.

We start with the estimation of the correlation parameters $\boldsymbol{\phi}_l$ by solving the optimization Problem (2.42).
Unfortunately some major numerical issues arise in this phase. In order to estimate the correlation parameters of the Gaussian process ϵ_l, we have to maximize an objective function that is the joint posterior distribution of such parameters. Such objective function depends explicitly on the number of the available low-resolution data n. As a matter of fact, n appears as a negative exponent in one of the terms of the objective function. When n is large, as in our case, such term drops to zero (because of the limited numerical precision of Matlab) and so does the entire objective function.
For this reason, we cannot approximate the correlation parameters $\boldsymbol{\phi}_l$ using the posterior modes in our case study.

We overcome this issue by exploiting some of the results in [CP11].
[CP11] implements the GP model proposed in [Qia+06] (see Section 1.5.3) exploiting the GPML toolbox by [RN10]. The following GP model is used:

$$z_l(\mathbf{v}) \sim GP\left(\beta_0, \sigma_z^2 \exp\left\{-\frac{|\mathbf{v}-\mathbf{v}'|^2}{2\ell^2} + \sigma_n^2 \Delta\right\}\right), \tag{5.1}$$

where $\mathbf{v} = (x, y)$, β_0 is the unknown mean of the GP, σ_z^2 is the unknown variance of the GP, ℓ is the correlation parameter (according to the notation in [RW06]), σ_n^2 is the unknown variance of the measurement error and Δ is the Kronecker delta.
If we express model (5.1) accordingly to the notation used so far, it is equivalent to the following low resolution model:

$$\mathrm{y}_l(\mathbf{x}_i) = \beta_{l_0} + \epsilon_l(\mathbf{x}_i) + \eta(\mathbf{x}_i), \tag{5.2}$$

where $\epsilon_l(\cdot) \sim GP(0, \sigma_l^2, \phi_l)$, $\eta(\cdot) \sim N(0, \sigma_\eta^2)$ and:

$$\begin{aligned}
&\mathrm{y}_l(\cdot) = z_l(\cdot),\\
&\mathbf{x}_i = \mathbf{v} = (x, y),\\
&\beta_{l_0} = \beta_0,\\
&\sigma_l^2 = \sigma_z^2,\\
&\phi_l = \frac{1}{2\ell^2},\\
&\sigma_\eta^2 = \sigma_n^2.
\end{aligned}$$

We point out that, the mean of $\mathrm{y}_l(\cdot)$ is a perfect plan parallel to the $x-y$ plan and the discrepancy between this plan and the data is modeled as a Gaussian process with an isotropic Gaussian correlation function and an i.i.d. noise. Thus, unlike the model we previously defined, the mean is a constant and the correlation is assumed to be isotropic, i.e. the activity of the correlation function along both the directions of the x and y axes is the same.

In order to proceed, we decide to apply the BHGP with the two-stage approach for prediction, described in Section 2.7.1. This appears to be a convenient choice at this point, because the missing low resolution data $\hat{\mathrm{y}}_{l_1}$ that we need in order to implement the adjustment model (2.54), are readily obtainable with the function gp from the GPML Toolbox by [RN11].
We remind again that this plug-in approach does not take into account the prediction uncertainty of $\hat{\mathbf{y}}_{l_1}$ in the approximation of the posterior predictive distribution, but only the uncertainty of the unknown parameters of the GP adjustment model.

We approximate the correlation parameters $\boldsymbol{\phi}_\rho$ and $\boldsymbol{\phi}_\delta$ using their posterior mode and we notice how small the estimated $\boldsymbol{\phi}_\rho$ are:

$$\begin{aligned}
&\text{Data configuration 1: } \boldsymbol{\phi}_\rho = (0.3329 \cdot 10^{-8},\ 0.0941 \cdot 10^{-8}),\\
&\text{Data configuration 2: } \boldsymbol{\phi}_\rho = (0.9107 \cdot 10^{-10},\ 0.1935 \cdot 10^{-10}),\\
&\text{Data configuration 3: } \boldsymbol{\phi}_\rho = (0.6377 \cdot 10^{-6},\ 0.0002 \cdot 10^{-6}).
\end{aligned}$$

This implies that the Gaussian process $\rho(\cdot)$ is extremely highly correlated. We suspect that the realization of scale GP $\rho(\cdot)$ will be very close to 1 on the whole surface of interest.

We run the MCMC algorithm and we approximate the predictive distribution as we previously did in Chapter 3.

Figure 5.6 represents the 3D scatter plots of the clouds of the predicted points and the clouds of the corresponding CMM testing points for all the three analyzed data configurations.

Figures 5.7-5.9 display the magnitude of the prediction error $\mathrm{y}_{\mathrm{test}} - \hat{\mathrm{y}}_h^{\mathrm{BHGP}}$ on the shape of the interpolated predicted surface.
As we expected, the prediction error is larger where discontinuities in the curvature of the surface are more evident.

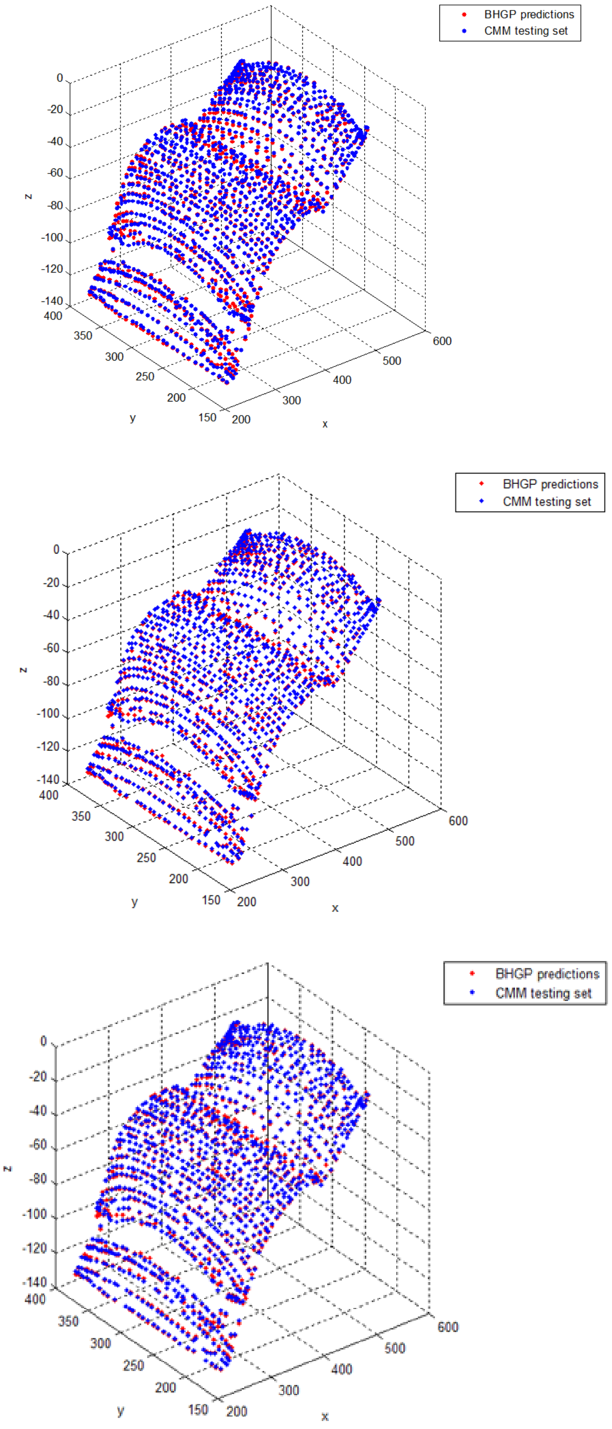

Figure 5.6: Representation of the clouds of the predicted points (red) and the corresponding testing set of CMM points (blue) for all the three analyzed data configurations.

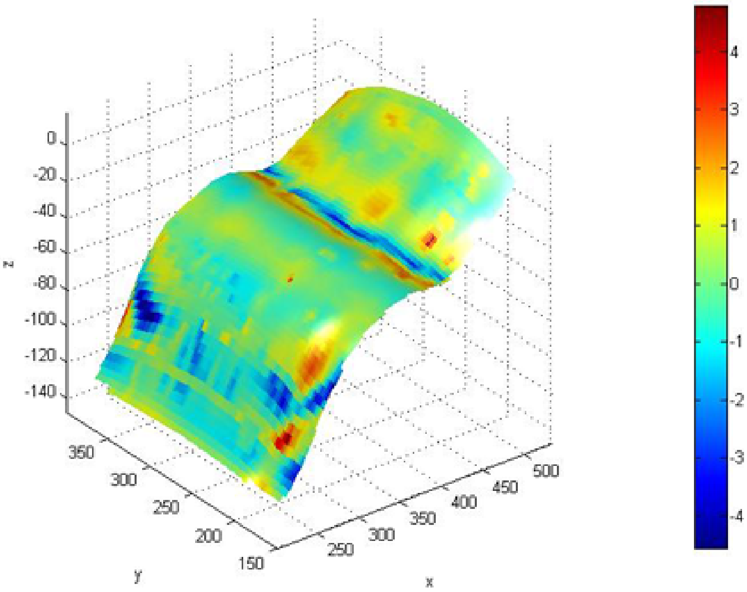

Figure 5.7: Interpolated error plot: it displays the magnitude of the prediction error $\mathrm{y}_{\mathrm{test}} - \hat{\mathrm{y}}_h^{\mathrm{BHGP}}$ on the shape of the interpolated predicted surface. Cold colors indicate negative prediction errors, i.e. the predicted points are overestimates of the testing measured points. Warm colors indicate positive prediction errors, i.e. the predicted points are "impossible" as they lie under the measured surface (Data configuration 1).

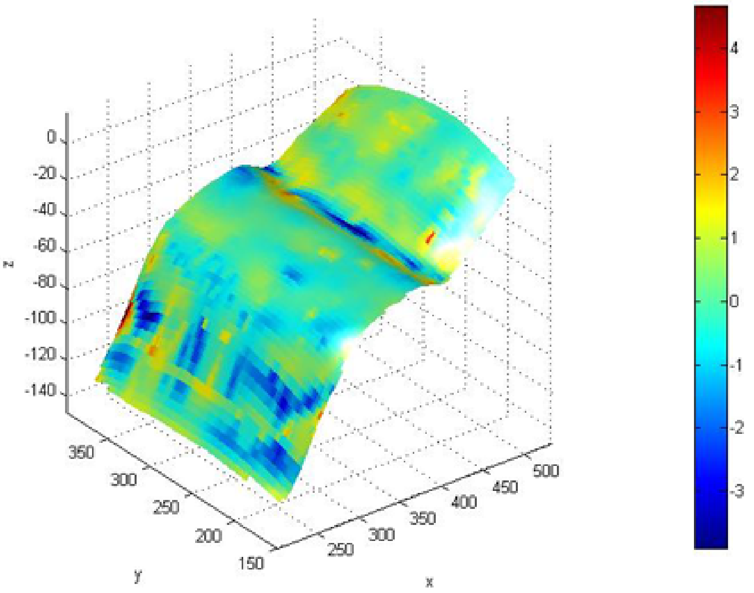

Figure 5.8: Interpolated error plot: it displays the magnitude of the prediction error $\mathrm{y}_{\text{test}} - \hat{\mathrm{y}}_h^{\text{BHGP}}$ on the shape of the interpolated predicted surface. Cold colors indicate negative prediction errors, i.e. the predicted points are overestimates of the testing measured points. Warm colors indicate positive prediction errors, i.e. the predicted points are "impossible" as they lie under the measured surface (Data configuration 2).

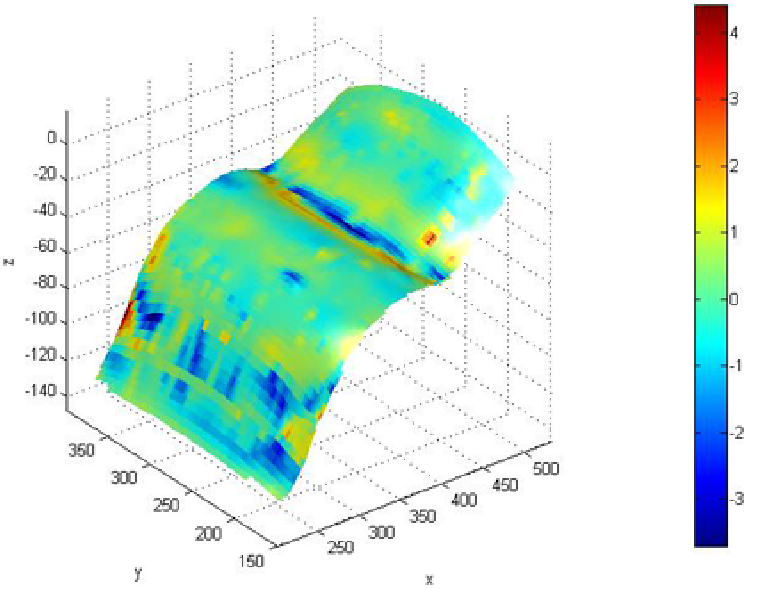

Figure 5.9: Interpolated error plot: it displays the magnitude of the prediction error $\mathrm{y}_{\mathrm{test}} - \hat{\mathrm{y}}_h^{\mathrm{BHGP}}$ on the shape of the interpolated predicted surface. Cold colors indicate negative prediction errors, i.e. the predicted points are overestimates of the testing measured points. Warm colors indicate positive prediction errors, i.e. the predicted points are "impossible" as they lie under the measured surface (Data configuration 3).

5.6 Comparison of different GP models on multi-accuracy metrology data

As we did in Section 3.4.1, we compare the predictive performances of different GP models applied to the available multi-resolution metrology data.
We remind the notation we are going to use:

- "GPmodel": universal kriging GP model (2.38) that uses the low-accuracy data only;
- "GPmerge": universal kriging GP model (2.38) that uses both the low-accuracy data and the high-accuracy data as the indistinctly came from a unique source and had the same resolution;
- "QW06": data fusion model [Qia+06] that integrates low-accuracy and high-accuracy data in the way described in Section 1.5.3;
- "QW08": two-stage BHGP model [QW08] that integrates low-accuracy and high-accuracy data in the way described in Section 2.6.

Table 5.1 summarizes the prediction results in terms of Square Root Mean Squared Prediction Error, defined as:

$$\mathrm{SRMSPE}(\mathbf{x}_i^*) = \sqrt{\frac{1}{n_{test}} \sum_{i=1}^{n_test} (\mathrm{y}_{test}(\mathbf{x}_i^*) - \hat{\mathrm{y}}_h)^2}. \tag{5.3}$$

This indicator is more appropriate in the present case study because it allows to quantify the prediction error as a quantity that is dimensionally comparable to the physical distance between the true high-accuracy measure and the corresponding predicted measure.

SRMSPE	**Data configuration 1** 455 MScMS-II data 25 CMM data	**Data configuration 3** 455 MScMS-II data 62 CMM data	**Data configuration 3** 455 MScMS-II data 124 CMM data
GPlowres	1.1382	1.1382	1.1382
GPmerge	1.1330	1.0879	1.0798
QW06	1.1519	1.0014	0.9973
QW08	1.1302	0.9769	0.9204

Table 5.1: Comparison of the prediction results in terms of Square Root Mean Squared Prediction Error.

The computed SRMSPEs suggest again that models “QW06” and “QW08” provide lower MSPEs. Furthermore, the best prediction results appear to be the ones corresponding to the third data configuration, where $n_1 = 124$ high-accuracy data were used to adjust the predictions computed with the low-accuracy data.
In order to draw conclusions on the matter of prediction performances, we need to carry out some appropriate statistical tests to verify that the prediction performances of the different methods are significatively different.

We proceed as we did in Section 3.4.1. This time we choose as a performance indicator the Square Root Squared Prediction Error (SRSPE) at every input point $\mathbf{x}_i^*$, $i = 1, ..., n_{test}$, of the testing set (for the reason explained above):

$$\text{SRSPE}(\mathbf{x}_i^*) = \sqrt{(\mathrm{y}_{test}(\mathbf{x}_i^*) - \hat{\mathrm{y}}_h)^2}, \quad i = 1, ..., n_{test} \tag{5.4}$$

where $\mathrm{y}_{test}(\cdot)$ is the measured high-resolution response used for cross-validation and $\hat{\mathrm{y}}_h$ is the estimated high-resolution response at the same input point using one of the above prediction methods.

We want to determine a suitable hypothesis testing procedure for evaluating the statistical significance of the difference of two different prediction methods according to some performance indicator.
Since we used the same testing set to compute the SRSPEs for each of the above methods, a test for paired observations would be appropriate because, as we already explained, it allows to reduce the variance by comparing the SRSPEs “within” corresponding input points, rather than “across” different input points. We cannot use the paired-t test because the SRSPEs are not normally distributed. Thus we perform the non parametric One-Sample Sign Test on the pairwise differences of the SRSMPs, because it is robust to the lack of symmetry of the distribution of the differences.. Assume we want to compare “Method 1” and “Method 2”. Then two populations of SRSPEs are available, $\text{SPE}^{\text{Method 1}}$ and $\text{SPE}^{\text{Method 2}}$. The hypotheses we need to test are:

$$H_0\text{: median}(\text{SPE}^{\text{Method 1}} - \text{SPE}^{\text{Method 2}}) = 0.$$

Vs.

$$H_1\text{: median}(\text{SPE}^{\text{Method 1}} - \text{SPE}^{\text{Method 2}}) \neq 0.$$

We report the Minitab output corresponding to the following tests for data configuration 2 and data configuration 3, because they provide the most significative results:

- H_0: $\text{median}(\text{SPE}^{\text{GPlowres}} - \text{SPE}^{\text{GPmerge}}) = 0$
- H_0: $\text{median}(\text{SPE}^{\text{GPlowres}} - \text{SPE}^{\text{QW06}}) = 0$
- H_0: $\text{median}(\text{SPE}^{\text{GPlowres}} - \text{SPE}^{\text{QW08}}) = 0$
- H_0: $\text{median}(\text{SPE}^{\text{GPmerge}} - \text{SPE}^{\text{QW06}}) = 0$
- H_0: $\text{median}(\text{SPE}^{\text{GPmerge}} - \text{SPE}^{\text{QW08}}) = 0$
- H_0: $\text{median}(\text{SPE}^{\text{QW06}} - \text{SPE}^{\text{QW08}}) = 0$.

For data configuration 2 we obtain the following results:

```
Sign test of median =  0,00000 versus not = 0,00000

                      N  Below  Equal  Above       P   Median
diff_lowres_merge  1181    442      0    739  0,0000  0,02799
diff_lowres_QW06   1181    531      0    650  0,0006  0,06414
diff_lowres_QW08   1181    495      0    686  0,0000  0,08552
diff_merge_QW06    1181    522      0    659  0,0001  0,06226
diff_merge_QW08    1181    532      0    649  0,0007  0,06792
diff_QW06_QW08     1181    573      0    608  0,3225  0,01180
```

The computed p-values allow us to conclude that there is no statistical evidence to support the hypothesis that the median of all the pairwise differences is null, except for the one corresponding to the pairwise comparison of the SRSPEs computed with models “QW06” and “QW08”.
Table 5.2 provides the ranking for the different prediction methods we considered according to the results of the paired Sign Tests we performed.

Pos.	Method	Median(SRSPE)	Improvement rate
3	GPlowres	0.6025	-
2	GPmerge	0.6020	$0,1\%$
1*	QW06	0.5341	$11,3\%$
1*	QW08	0.5174	$3,1\%$

Table 5.2: Ranking of the compared prediction methods sorted by decreasing predictive performance according to the results of the Sign Tests performed on the differences of the SMPs (Data configuration - II).

The rates reported in the last column confirm that a significative improvement in the prediction results is accomplished when using GP models that integrate the low-resolution and the high-resolution data.

The Minitab output for Sign Test on the pairwise differences of the SRSMPs computed using data configuration is:

```
Sign test of median =  0,00000 versus not = 0,00000

                     N  Below  Equal  Above       P   Median
diff_lowres_merge  1119    506      0    613  0,0015  0,02169
diff_lowres_QW06   1119    454      0    665  0,0000  0,08976
diff_lowres_QW08   1119    402      0    717  0,0000   0,1046
diff_merge_QW06    1119    450      0    669  0,0000  0,08292
diff_merge_QW08    1119    385      0    734  0,0000   0,1248
diff_QW06_QW08     1119    531      0    588  0,0941  0,02661
```

The p-values allow us to conclude that, assuming a confidence level of 90%, the median of the pairwise differences of the SRSPEs are statistically significative.

Table 5.2 shows the ranking of the different prediction methods according to the results of the paired Sign Tests.

Pos.	Method	Median(SRSPE)	Improvement rate
4	GPlowres	0.6033	-
3	GPmerge	0.5998	$0,6\%$
2	QW06	0.4664	$22,2\%$
1	QW08	0.4346	$6,8\%$

Table 5.3: Ranking of the compared prediction methods sorted by decreasing predictive performance according to the results of the Sign Tests performed on the differences of the SMPs (Data configuration - II).

The box-plots in Figure 5.10 provide a graphical representation of the results and allow to appreciate the reduction of the prediction uncertainty when models “QW06” and “QW08” are used.

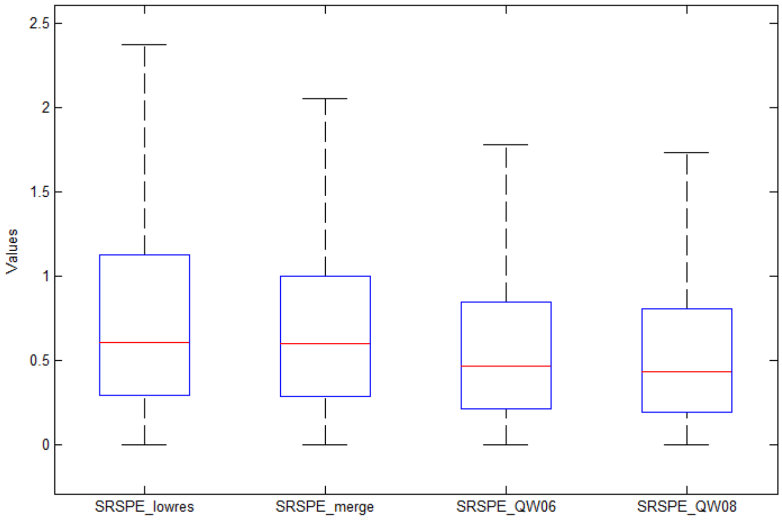

Figure 5.10: Box-plots of the Square Root Squared prediction Errors computed using different GP models (Data configuration 3).

As one would expect, we observe that data configuration 3 (455 MScMS-II data and 124 CMM data) leads to better prediction results compared to the other data configurations.

However data configuration 3 may not be the optimal one because of the high computational effort required for running the MCMC sampling algorithm, due to the high dimensions of the correlation matrices of the Gaussian processes. As a consequence data configuration 2 appears to be the best compromise between good prediction results and reasonable computational times.

These observations make us wonder whether the MCMC approach to Bayesian inference is really worth it in this case study, where such a large amount of data is available. We witnessed that the method by [CP11], implemented using the GMPL toolbox, provides good prediction results in instant computational times if compared to the hours required by the BHGP model. The trade-off between reasonable computational times and good predictive performance is not an easy decision.

5.7 Final remarks

In the present application study, we proved that the structured integration of multiple sensor coordinate measurements using GP models “QW06” and “QW08” significatively improves the prediction results, in comparison with the classical universal kriging GP model by a rate that ranges from 11% to 20% in the current case study.

We point out that the application of the BHGP model “QW08” has the major drawback of long computational times due to the choice of using an MCMC approach.

The implemented BHGP model also exhibits some numerical issues when dealing with a large amount of data. It has to be evaluated whether the use of Cholesky factorization and matrix inversion lemma could help to deal with this situation.

Finally we noticed that the predictive performance of the analyzed GP model is very sensible to the choice of the positions for the high-accuracy point. It would be appropriate to develop a sampling technique in order to optimally select the positions where the high accuracy points should be placed, in order to maximize the predictive power of the model.

Conclusion

In the preceding chapters we discussed the Bayesian Hierarchical Gaussian Process Model developed by [QW08]. Such model is used to integrate - in a structured way - data coming from two different sources with different accuracy. The low-accuracy data are the output of a computer experiment and the high-accuracy data come from a more precise computer simulation or a physical experiment. In a first stage, a Gaussian Process Model is used to fit the low-accuracy data. Then the high resolution data are linked to the low resolution data using a flexible adjustment model where two Gaussian processes perform scale and location adjustments. The definition of the Bayesian model is completed once a set of prior distributions on the unknown parameters is suitably selected. An empirical Bayesian approach is chosen in order to ease the computational load, and the correlation parameters of the Gaussian processes involved in the model are estimated using their posterior modes. Then a two-stage Monte Carlo Markov Chain algorithm is used to approximate the posterior predictive distribution at new input sites. The choice of a Bayesian framework has the main advantage of incorporating the uncertainty of the unknown model parameters in the posterior predictive distribution.

The Bayesian Hierarchical Gaussian Process Model was then extended in order to model the more general situation where also the low-accuracy data come from a physical experiment. A measurement error term was included in the model for the low-accuracy data and a series of adjustments had to be applied to the prediction method, as a consequence of this modification.
The Bayesian Hierarchical Gaussian Process Model was then implemented in Matlab and a validation study was performed in order to verify the Matlab code and evaluate the predictive performance of the model. Cross-validation on the prediction results was performed on three distinct data sets: one data set was provided in [QW08], the other two data sets were simulated using two different methodologies. Positive results were achieved, in terms of Mean Square Prediction Error, in

all the above situations and the model proved to work well as a predictive tool.

The extended Bayesian Hierarchical Gaussian Process Model was then applied to a set of metrology data acquired with two instruments with different accuracy. The data are point cloud measures of a portion of the surface of a toy car used for test purposes. The low-resolution data are acquired with the innovative optical-based Mobile Spatial Coordinate Measuring System II, developed at Politecnico di Torino, Italy, and the high-resolution data are acquired with a Coordinate-Measuring Machine.
In our case study, the combination of a large amount of low-resolution data with a few high-resolution data using the GP models proposed in [QW08] and [Qia+06], proved to be more effective for modeling purposes than using simpler GP models that use low-resolution data alone or treat low-resolution and high-resolution data as they indistinctly came from a unique source. Improvements rates, in terms of prediction error, range from 11% to 22%.

The use of spatial Statistics models - in our case Gaussian Process models - in multi-sensor metrology is relatively new. Working on such a broad and complicated subject was a very compelling challenge. Although it allowed us to face many different topics and appreciate the complexity and the magnitude of the matter, many of the problems we dealt with are left for future research.
The first among all the possible future developments is the use of alternative correlation functions. We assumed that the correlation structure of the Gaussian processes involved in the Bayesian Hierarchical Gaussian Process Model is modeled as a Gaussian correlation function. This choice seemed reasonable for our case study, but it definitely is not the only possibility. There are many other form of correlation that could work better than ours. The use of alternative correlation functions needs to be investigated, better if using the available tools from spatial Statistics.
The most tough topic we faced was the estimation of the correlation parameters. In the Bayesian framework the use of the posterior mode as a point estimate is widespread because it is usually the easiest to compute because it does not need any integration, unlike the posterior mean or the posterior median. Posterior mode, mean and median are equivalent for symmetric unimodal posterior distributions, but, when the posterior is multimodal, estimating the parameters with the posterior mode, could be deceiving. We relied on the posterior mode estimates of the correlation parameters and this choice worked for us. Anyway we do not

exclude the possibility that different estimation procedures for such parameters could be more efficient or provide better results.

In order to overcome the problem of estimating the correlation parameters and avoid the implications of the Bayesian empirical approach, we considered the idea of adopting a fully Bayesian approach, but we couldn't find an efficient implementation for our case. Putting further effort in this direction would allow our model to take into account the uncertainty of all the unknown parameters.

Furthermore, we are not sure of how plug-in estimates of the correlation parameters affect prediction. We think that a sensitivity analysis, to study the effect of small perturbations of the correlation parameters on the prediction efficiency, could be appropriate.

The Bayesian Hierarchical Gaussian Process model is very "rich". Besides using a Gaussian process to fit the low-accuracy data, it includes two further Gaussian processes in the adjustment model. It is likely that this structure could be exceedingly articulate for some applications. Thus it would be appropriate to verify whether the use of a Gaussian Process as a scale parameter in the adjustment model is really necessary, or a simpler constant or linear scale parameter is enough for the application purposes.

We noticed that the predictive performance of the Bayesian Hierarchical Gaussian Process model is very sensible to the choice of the positions for the high-accuracy point. It would be appropriate to develop some sampling technique in order to optimally select the positions where the high accuracy points should be placed, in order to maximize the predictive power of the model.

Monte Carlo Markov Chain Methods, although they are a powerful approach to Bayesian inference, usually require significative computational efforts and long times to run. In the case of complicated inference problems, it is licit to wonder if Markov Chain Monte Carlo Methos are really worth it. [RM07] proposes an approach to approximate marginal posterior densities using numerical deterministic schemes, that works for many Hierarchical Gaussian Random Field models with significative time saving.

Finally, further research efforts will be put in the development of Gaussian Process models for applications in multi-sensor data fusion in the Metrology field, for instance for the combination of data coming from the Mobile Spatial Coordinate Measuring System II and other optical based system, as Structured Light.

Appendix A

Bayesian Inference and Markov Chain Monte Carlo Sampling

In the present chapter we introduce the basics of Bayesian inference and Markov Chain Monte Carlo techniques to carry out the Bayesian computations.
We also provide a quick overview on Hierachical Bayesian Models.

A.1 Bayesian Inference

Bayesian Inference is a branch of Statistical Inference that, in opposition to the "frequentist" approach, combines prior information on the unknown parameters of a population or a model to the evidence carried by observed data, using Bayes' theorem. For our illustrative purposes, we consider the univariate case for simplicity. We point out that all the following results and observations can be easily extended to the multivariate case.
θ indicates the unknown parameter and $\mathbf{y}$ the observed data, described by their likelihood function $L(\mathbf{y}|\theta) = p(\mathbf{y}|\theta)$, i.e. a probabilistic model $p(\cdot)$ conditioned on the unknown θ.
The two basic elements to perform Bayesian inference are the likelihood function of the data $L(\mathbf{y}|\theta)$ and a probabilistic distribution on the unknown parameter $p(\theta)$, known as prior distribution, that describes prior beliefs or knowledge on θ.
The aim of Bayesian inference is to compute the probabilistic law that describes the behavior of the unknown parameter given the information carried by the observed data, in order to draw conclusions (or make decisions based) on θ. In other words, the goal is to compute a conditional probabilistic distribution $p(\theta|\mathbf{y})$, called

posterior distribution, that summarizes the available prior knowledge on θ and the information carried by the data $\mathbf{y}$, through the use of Bayes' theorem.

A.1.1 Bayes' theorem for densities

For the definition of conditional probability, the two following equalities hold:

$$\begin{aligned} L(\mathbf{y}|\theta) &= \frac{p(\mathbf{y},\theta)}{p(\theta)} \\ p(\theta|\mathbf{y}) &= \frac{p(\mathbf{y},\theta)}{p(\mathbf{y})}. \end{aligned} \tag{A.1}$$

From (A.1) it directly follows that the joint density function of $(\theta, \mathbf{y})$ is:

$$p(\mathbf{y},\theta) = L(\mathbf{y}|\theta)p(\theta) = L(\theta|\mathbf{y})p(\mathbf{y}),$$

which implies:

$$p(\theta|\mathbf{y}) = \frac{L(\mathbf{y}|\theta)p(\theta)}{p(\mathbf{y})}. \tag{A.2}$$

The density at the denominator is defined as:

$$p(\mathbf{y}) = \int_\theta L(\mathbf{y}|\theta)p(\theta)d\theta. \tag{A.3}$$

Equation (A.2) is known as Bayes theorem for density functions and provides the density function $p(\theta|\mathbf{y})$ of the posterior distribution of the unknown parameter θ given the oserved data $\mathbf{y}$.

Since the denominator of (A.2) does not depend on θ, it can be considered as a constant. Thus the expression:

$$p(\theta|\mathbf{y}) \propto L(\mathbf{y}|\theta)p(\theta) \tag{A.4}$$

is equivalent to (A.2) and it is known as the non-normalized posterior density function of the parameter θ.

This set of rules represents the core of Bayesian inference.

A.1.2 Inferential analysis on the unknown parameters

The posterior distribution $p(\theta|\mathbf{y})$ is all we need to perform inferential analysis on the unknown parameter θ.

Point estimates of the parameter θ can be computed from $p(\theta|\mathbf{y})$. The usual choice is some measure of centrality. Three familiar choices are the posterior mean:

$$\hat{\theta} = \int_{\theta} p(\theta|\mathbf{y})\, d\theta, \tag{A.5}$$

the posterior median:

$$\hat{\theta} \quad \text{s.t.} \quad \int_{-\infty}^{\hat{\theta}} p(\theta|\mathbf{y})\, d\theta = 0.5, \tag{A.6}$$

and the posterior mode:

$$\hat{\theta} = \arg\sup_{\theta} p(\theta|\mathbf{y}). \tag{A.7}$$

The posterior mode is the easiest to compute, because it does not need any integration (if $p(\theta|\mathbf{y})$ is replaced by its non-standardized form, $L(\mathbf{y}|\theta)p(\theta)$), unlike the posterior mean or the posterior median.
Posterior mode, mean and median are equivalent for symmetric unimodal posterior distributions, but, when the posterior is multimodal, estimating the parameters with the posterior mode, could be deceiving.

The posterior distribution also allows to compute interval estimations or Bayesian credibility intervals. Suppose we can find the $\alpha/2$ and $(1-\alpha/2)$ quantiles of $p(\theta|\mathbf{y})$, i.e. $q_{\alpha/2}$ and $q_{1-\alpha/2}$ such that:

$$\begin{aligned} \int_{-\infty}^{q_{\alpha/2}} p(\theta|\mathbf{y})\, d\theta &= \frac{\alpha}{2} \\ \int_{q_{1-\alpha/2}}^{\infty} p(\theta|\mathbf{y})\, d\theta &= 1 - \frac{\alpha}{2}. \end{aligned} \tag{A.8}$$

The interval $(q_{\alpha/2}, q_{1-\alpha/2})$ is the $100 \times (1-\alpha)\%$ Bayesian credibility interval for θ. Unlike the frequentist confidence interval, the Bayesian credibility interval allows the following interpretation: the probability that θ lies in $(q_{\alpha/2}, q_{1-\alpha/2})$ is $1-\alpha$. Sometimes, particularly in the case of unimodal posterior distribution, it is desirable to compute the Highest Posterior Density (HPD) interval, i.e. the interval $(a,\, b)$ computed by solving:

$$\min b - a \quad \text{s.t} \quad \int_{a}^{b} p(\theta|\mathbf{y})\, d\theta = 1 - \alpha. \tag{A.9}$$

For multimodal posterior distribution the HPD interval is more complicated to compute.

A.1.3 Bayesian prediction

In order to perform inference on a future observation y^* the posterior predictive distribution is computed:

$$\begin{aligned} p(\mathrm{y}^*|\mathbf{y}) &= \int_\theta p(\mathrm{y}^*, \theta|\mathbf{y})\, d\theta = \\ &= \int_\theta p(\mathrm{y}^*|\mathbf{y}, \theta) p(\theta|\mathbf{y})\, d\theta = \\ &= \int_\theta p(\mathrm{y}^*|\mathbf{y}, \theta) \frac{p(\theta)L(\mathbf{y}|\theta)}{\int p(\theta)L(\mathbf{y}|\theta) d\theta}\, d\theta. \end{aligned} \tag{A.10}$$

In the first line the marginalization of the joint posterior distribution $p(\mathrm{y}^*, \theta|\mathbf{y})$ is carried out by integrating in $d\theta$. The equality in the second line always holds for the definition of conditional probability. The last step is a direct application of Bayes' theorem described in Section A.1.1.
Like we did in the previous section for the parameter θ, an inferential analysis can be carried out for the prediction y^* too. Point estimates and interval estimates can be computed in the same way from the predictive distribution $p(\mathrm{y}^*|\mathbf{y})$.

The predictive distribution (A.10) points out a very important feature that depends on the choice of adopting a Bayesian approach. The ability to attain prediction through the definition of a probability distribution allows one to incorporate the uncertainty on the unknown parameters, directly in the predictive distribution itself.

A.1.4 Choice of prior distribution

A sensible topic in Bayesian inference is the choice of an appropriate prior distribution $p(\theta)$, i.e. the distribution that describes our prior knowledge or beliefs on the unknown parameter.
The first possible choice is the use of conjugate priors, i.e. the prior and the posterior distributions have the same parametric form. The main advantage of conjugate priors is that they ease the computations, as they allow to compute the integral in the posterior distribution analytically and to obtain a posterior that has a closed known from. Unfortunately, the selection of an appropriate conjugate prior is not always simple, as a conjugate prior does not always suitably reflect the

prior knowledge on the unknown parameter.
The use of non-conjugate priors cause the posterior to have a different parametric form than the prior. The advantage is that they can be chosen in order to better reflect the prior knowledge on the parameter θ.
A particular class of non-conjugate priors is the one that includes non-informative priors. The main feature of such prior distributions is that they express a complete lack of (or very little) prior knowledge on the parameter. In some cases, non-informative priors are improper, i.e. their density functions do not have a finite integral. Apart from some particular cases, improper priors could lead to improper posterior distributions, that are completely useless for inference purposes.

The main difficulty in Bayesian inference, apart from the conceptual challenge of choosing an adequate prior, is due to the computations needed to obtain the posterior distribution.
Once the posterior is computed, inferential analysis is carried out by solving other integrals that involve the posterior distribution, that are usually in the form:

$$J = E(f(\theta)|\mathbf{y}) = \int_{\theta} f(\theta)p(\theta|\mathbf{y})\, d\theta, \tag{A.11}$$

where $f(\theta)$ is a generic function of the unknown parameter.
As a consequence, Bayesian inference implies the necessity to solve complicated integrals, especially when dealing with the multivariate case, that often cannot be computed in closed form. In the last decades, the development of simulation-based approaches to Bayesian analysis allowed to overcome this situation.
A very important family of simulation-based approaches is Markov Chain Monte Carlo methods.

A.2 Markov Chain Monte Carlo sampling

Markov Chain Monte Carlo (MCMC) methods are used to approximate integrals like the one in Equation (A.11) by drawing dependent samples generated using Markov chains.
Markov Chains are an important class of random processes where the next state of the chain depends only on the current state and not on the previous ones. Markov chains used in MCMC algorithm should be ergodic, i.e. irreducible (any state of the chain can be reached from any other state in a finite number of steps) and aperiodic (no cyclic patterns should emerge).

The stationary distribution of the samples is the target distribution, i.e. the distribution we want to sample from. In order to carry out correct inference, we have to be sure that the MCMC algorithm has reached convergence, i.e. the chain used to generate the samples has reached its steady state.
Once the chains of dependent samples have been generated, integral (A.11) is approximated using the following Monte Carlo integration:

$$J = E(f(\theta|\mathbf{y})) \approx \frac{1}{K-k_0} \sum_{k=k_0+1}^{K} f(\theta^{(k)}) = \hat{J}. \tag{A.12}$$

$\theta^{(k)}$ are the samples generated using a Markov chain with stationary distribution given by the posterior distribution $p(\theta|\mathbf{y})$ (target distribution). K is the total number of drawn samples and k_0 is the number of samples that need to be discarded before the chain has reached its stationary distribution (burn-in period).

Important examples of MCMC methods are the Gibbs algorithm, the Metropolis-Hastings algorithm and the Random-Walk Metropolis algorithm. In the following sections we will provide concise descriptions of such algorithms, besides we will illustrate a modified version of the Random-Walk Metropolis algorithm for sampling positive-valued chains.

A.2.1 Gibbs sampler

Suppose we want to sample D parameters $\boldsymbol{\theta} = (\theta_1, ..., \theta_D)$ from the multivariate posterior target distribution $p(\boldsymbol{\theta}|\mathbf{y}) = p(\theta_1, ..., \theta_D|\mathbf{y})$ given the observed data $\mathbf{y}$. The Gibbs sampler is the most powerful MCMC sampling method, as its convergence is almost independent of the number of parameters D.
We assume that all the full conditional distributions defined as:

$$p(\theta_d|\mathbf{y}, \overline{\theta_d}) = p(\theta_d|\mathbf{y}, \theta_1, \theta_2, ..., \theta_{d-1}, \theta_{d+1}, ..., \theta_D), \quad \forall \quad d = 1, .., D, \tag{A.13}$$

are easily obtainable from the target distribution.
The Gibbs sampling algorithm is summarized in the following steps:

- Initialize $\boldsymbol{\theta}$ to the value $\boldsymbol{\theta}^{(0)}$.
- For $k = 1, ..., K$ perform the following sequential draws of $\boldsymbol{\theta}^{(k)} = (\theta_1^{(k)}, ..., \theta_D^{(k)})$ from their respective D full conditional distributions defined in (A.13):

  1. Draw one sample of $\theta_1^{(i)}$ from $p(\theta_1|\mathbf{y}, \theta_2^{(i-1)}, ..., \theta_D^{(i-1)})$.

2. Draw one sample of $\theta_2^{(i)}$ from $p(\theta_2|\mathbf{y}, \theta_1^{(i)}, \theta_3^{(i-1)}, ..., \theta_D^{(i-1)})$.

$\vdots$

D. Draw one sample of $\theta_D^{(i)}$ from $p(\theta_D|\mathbf{y}, \theta_1^{(i)}, \theta_2^{(i)}, \theta_3^{(i)}, ..., \theta_{D-1}^{(i)})$.

It has been proven that as $K \to \infty$, after excluding a proper number of burn-in iterations k_0, samples $\boldsymbol{\theta}^{(k)} = (\theta_1^{(k)}, ..., \theta_D^{(k)})$, $k = k_0, ..., K$ can be viewed as they were sampled form the target distribution $p(\theta_1, ..., \theta_D|\mathbf{y})$ [CG92].
For further details on the Gibbs sampler refer to [CG92], [CC07] and [Gel+03].

A.2.2 Metropolis-Hastings algorithm

Suppose we want to draw samples of θ from the posterior distribution $p(\theta|\mathbf{y})$, our target distribution. In order to simplify the notation we will refer to $p(\theta)$ as the target distribution.
The Metropolis-Hastings (MH) algorithm is summarized in the following steps:

- Initialize $\theta^{(0)}$.
- For $k = 1, ..., K$:
  1. Draw a proposal sample θ^* from the transition kernel $T(\theta^*|\theta^{(k-1)})$.
  2. Accept sample θ^*, i.e. set $\theta^{(k)} := \theta^*$, if $u < \alpha(\theta^{(k-1)}, \theta^*)$, where:

$$\alpha(\theta^{(k-1)}, \theta^*) = \min\left\{\frac{T(\theta^{(k-1)}|\theta^*)p(\theta^*)}{T(\theta^*|\theta^{(k-1)})p(\theta^{(k-1)})}, 1\right\}$$

u is sampled form the distribution $U(0, 1)$,

else: $\theta^{(k)} := \theta^{(k-1)}$.

$T(a|b)$ usually is the transition kernel of the Markov chain, i.e. the probabilistic law that controls the transition from state a to state b. However the exact transition kernel of the chain is not know, thus $T(\cdot|\cdot)$ indicates an arbitrarily selected transition kernel, also called proposal distribution. $\alpha(a, b)$ plays the role of acceptance ratio and quantifies the probability that a transition from a to b is accepted. We point out that the non-normalized posterior distribution can be used here in place of the target distribution, because the normalization factor appears both in the numerator and denominator of the acceptance ratio.
If we want the MH algorithm to work well, the support of the proposal density has to be included in the support of the target density, because this condition usually

ensures that the chain is irreducible and aperiodic. Thus the proposal distribution needs to be suitably selected in order to have an efficient algorithm.
For further details on the Metropolis-Hastings algorithm see [CG95], [CC07] and [Gel+03].

A.2.3 Random-Walk Metropolis algorithm

A popular choice for the transition kernel in the MH algorithm (also the one proposed in the original Metropolis algorithm) is the symmetric transition kernel, such that $T(a|b) = T(b|a)$. Random Walk Metropolis (RWM) algorithm uses as a transition kernel $T(a|b) = f(|b-a|)$ where $f(\cdot)$ is a symmetric density, for instance a (multivariate) normal distribution centered in 0.
Suppose we want to draw multivariate samples of $\boldsymbol{\theta}$ from the target distribution $p(\boldsymbol{\theta}) = p(\boldsymbol{\theta}|\mathbf{y})$. In such case the RWM algorithm can be summarized as follows:

- Initialize $\boldsymbol{\theta}^{(0)}$.
- For $k = 1, ..., K$:
  1. Sample $\boldsymbol{\theta}^*$ from the symmetric proposal distribution $N(\boldsymbol{\theta}^{(k-1)}, \Sigma)$.
  2. Accept the sample $\boldsymbol{\theta}^*$, i.e. $\boldsymbol{\theta}^{(k)} := \boldsymbol{\theta}^*$, if $u < \alpha(\boldsymbol{\theta}^{(k-1)}, \boldsymbol{\theta}^*)$, where:

  $$\alpha(\boldsymbol{\theta}^{(k-1)}, \boldsymbol{\theta}^*) = \min\left\{\frac{p(\theta^*)}{p(\theta^{(k-1)})}, 1\right\}$$

  u is sampled form the distribution $U(0, 1)$,

  else: $\theta^{(k)} := \theta^{(k-1)}$.

We notice how the acceptance ratio simplifies to the ratio of the target distribution at the proposal sample and at the previous sample. Furthermore the generated chain is a random walk, i.e. a random process that at the next state is equal to the current state plus a random error.
In our case the transition kernel is a multivariate normal distribution. As suggested in [Gel+03] the covariance of such distribution Σ (i.e. the parameter that controls the magnitude of the jump of the chain) should be tuned in order to have acceptance rates between 20% and 40%. This choice guarantees an appropriate convergence speed for the algorithm.
For further details on the RWM algorithm see [CG95] and [RS03].

A.2.4 Modified random-walk Metropolis algorithm

Suppose we want to sample a positive-valued multivariate D-dimensional parameter $\boldsymbol{\psi}$. In this case a normal transition kernel is no longer adequate, because proposal samples could be negative. To overcome this issue we need to modify the traditional Random-Walk Metropolis algorithm described in the previous section so that the sampled values are always positive.

We sample the proposal value $\boldsymbol{\theta}^*$ from a normal distribution centered in the natural logarithm of the chain at the previous iteration:

$$\boldsymbol{\theta}^* \sim N\left(\ln(\boldsymbol{\psi}^{(k-1)}), s^2\right).$$

To switch back to the original scale we anti-transform the sample $\boldsymbol{\theta}^*$:

$$\boldsymbol{\psi}^* = \exp\left\{\boldsymbol{\theta}^*\right\}.$$

The distribution of $\boldsymbol{\psi}^*$, i.e the proposal distribution, is not a normal anymore. We compute the density function of $\boldsymbol{\psi}^*$ using the the change of variable rule (see Appendix B.2):

$$\begin{aligned}
&\mathbf{X} = \boldsymbol{\theta}^* \text{ with density function } f(\mathbf{X})\\
&\mathbf{Y} = H(\mathbf{X}) = \exp\left\{\boldsymbol{\theta}^*\right\} = \boldsymbol{\psi}^*\\
&g(\mathbf{Y}) = f(\mathbf{X}) \det\left(\frac{d\mathbf{X}}{d\mathbf{Y}}\right) = f(\mathbf{X})\left(\prod_{i=1}^{D} \psi_i^*\right)^{-1}
\end{aligned}$$

Therefore the ratio of the proposal distribution that appears in the acceptance ratio $\alpha(\boldsymbol{\psi}_1^{(k-1)}, \boldsymbol{\psi}^*)$ evaluated at each step of the Random-Walk Metropolis algorithm is:

$$\frac{T(\boldsymbol{\psi}_1^{(k-1)}|\boldsymbol{\psi}^*)}{T(\boldsymbol{\psi}^*|\boldsymbol{\psi}_1^{(k-1)})} = \frac{\left(\prod_{i=1}^{D} \psi_i^{(k-1)}\right)^{-1}}{\left(\prod_{i=1}^{D} \psi_i^*\right)^{-1}} = \frac{\prod_{i=1}^{D} \psi_i^*}{\prod_{i=1}^{D} \psi_i^{(k-1)}}.$$

To avoid numerical overflow issues [Ntz09] suggests to switch to the log-scale to compute the acceptance ratio $\alpha(\boldsymbol{\psi}^{(k-1)}, \boldsymbol{\psi}^*)$.

In conclusion, the modified Random-Walk Metropolis algorithm for sampling positive valued parameters is summarized in the following steps:

- Initialize $\boldsymbol{\psi}^{(0)}$.

- For $k = 1, ..., K$:

  1. Define $\boldsymbol{\theta}^{(k-1)} = \ln(\boldsymbol{\psi}^{(k-1)})$.
  2. Sample $\boldsymbol{\theta}^*$ from the distribution $N(\boldsymbol{\theta}^{(k-1)}, \Sigma)$.
  3. Anti-transform: $\boldsymbol{\psi}^* = \exp\{\boldsymbol{\theta}^*\}$.
  4. Accept sample $\boldsymbol{\psi}^*$, i.e. set $\boldsymbol{\psi}^{(k)} := \boldsymbol{\psi}^*$, if $u < \alpha(\boldsymbol{\psi}^{(k-1)}, \boldsymbol{\psi}^*)$, where:

     $$\ln(\alpha(\psi_1^{(k-1)})) = \min\left\{0, \ln\left(\frac{p(\boldsymbol{\psi}^*)}{p(\boldsymbol{\psi}^{(k-1)})}\frac{\prod_{i=1}^{D}\psi_i^*}{\prod_{i=1}^{D}\psi_i^{(k-1)}}\right)\right\}$$

     u is sampled form the distribution $U(0, 1)$,

     else: $\boldsymbol{\psi}^{(k)} := \boldsymbol{\psi}^{(k-1)}$.

We finally point out that the Metropolis algorithm usually comes handy when we need to sample from target distributions with no known form.
For instance, the Metropolis algorithm can be used in the sampling steps of the Gibbs algorithm when some of the full conditional distributions do not have a known form. In this situation we address it as the Metropolis-within-Gibbs algorithm.

A.3 Convergence Diagnostics

When using iterative MCMC simulation algorithms such as the Gibbs sampler or the Metropolis algorithm, we have to be reasonably sure that the number of iterations is large enough to guarantee convergence of the sampled distribution to the target distribution. If the number of iterations is too small the simulations risk to be very little representative of the target distribution. Furthermore we have to decide the number of samples to discard before the sampled chain reaches its stationary distribution, i.e. the burn-in period.

A.3.1 Geweke convergence diagnostic

The convergence diagnostic proposed by Geweke is based on a test statistic that uses a spectral estimate of the variance of the chain and compares the means of two subsets of a single chain. If we cannot reject the assumption that the means of the two subsequences of samples are equal, than the two subsequence of samples are very likely to come from the same stationary distribution and convergence has been achieved.

After discarding the burn-in period we select among the remaining samples the first n_A and the last n_B, usually the first 10% and the last 50% of the chain respectively.
Then the following statistic is computed:

$$Z = \frac{\overline{\theta}_A - \overline{\theta}_B}{\sqrt{\frac{1}{n_A}\hat{S}_\theta^A(0) + \frac{1}{n_B}\hat{S}_\theta^B(0)}}, \tag{A.14}$$

where $\overline{\theta}_A$ and $\overline{\theta}_B$ are the sample means and $\hat{S}_\theta^A(0)$ and $\hat{S}_\theta^B(0)$ are estimates of the spectral density at zero computed using the first n_A and the last n_B samples respectively.
Geweke proved that Z is asymptotically distributed as a standard normal.
As a consequence Z^2 is asymptotically distributed as a Chi-squared distribution with one degree of freedom.

In order to perform the Geweke diagnostic of convergence in Matlab we used the `apm` function from a version of the famous `coda` package for Matlab [LeS99]. CODA (Convergence Diagnostics and Output Analysis) is a suite of S functions that provide the tools for analyzing the output of MCMC simulations.
The function `apm` performs the Geweke Z-test of the hypothesis of equality of the means of the two portions of the simulated chain and the p-value of the Chi-squared test is reported.

A.3.2 Gelman-Rubin convergence diagnostic

[Gel+03] suggests to use a multiple sequence technique to assess convergence of iterative simulation. The convergence diagnostic developed by Gelman and Rubin, described in [Gel+03], uses m independent parallel sequences, with $m \geq 2$ and starting points sampled from overdispersed distributions. Such method applies to univariate MCMC sampling, so it will allow us to assess convergence for all the sampled parameters independently to their marginal posterior distribution.

Suppose we want to make inference on a random variable using the scalar summary θ (which is a random variable itself). In our case θ is distributed according to a target distribution (our marginal posterior distribution $p(\theta|\mathbf{y})$) with mean μ and variance σ^2.
We first simulate M parallel chains of length N. In order to reduce the effect of the starting distribution we discard the first half of the simulations, so that we are left

with m sequences of length $N_0 = \frac{N}{2}$. We use θ_{ij}, $i = 1, ..., N_0$ and $j = 1, ..., M$, to indicate the $i-$th sample from the $j-$th chain. We compute the between-sequence variance B and within-sequence variance W as follows:

$$B = \frac{N_0}{M-1} \sum_{j=1}^{M} \left(\overline{\theta}_{\cdot j} - \overline{\theta}_{\cdot\cdot}\right)^2, \tag{A.15}$$

$$W = \frac{1}{M} \sum_{j=1}^{M} s_j^2, \tag{A.16}$$

where:

$$\overline{\theta}_{\cdot j} = \frac{1}{N_0} \sum_{i=1}^{N_0} \theta_{ij},$$

$$\overline{\theta}_{\cdot\cdot} = \frac{1}{M} \sum_{j=1}^{M} \overline{\theta}_{\cdot j},$$

$$s_j^2 = \frac{1}{N_0 - 1} \sum_{i=1}^{N_0} \left(\theta_{ij} - \overline{\theta}_{\cdot j}\right)^2.$$

The marginal posterior variance of θ is computed as a weighted average of B and W:

$$\hat{\sigma}_+^2 = \frac{N_0 - 1}{N_0} W + \frac{B}{N_0}. \tag{A.17}$$

This quantity is an unbiased estimate of the true variance σ^2, if the starting points of the sequences were drawn from the target distribution, but it's an overestimate if the starting distribution is overdispersed.

The within-sequence variance for a finite N_0 can be interpreted as an underestimate of σ^2.

Then we compute a quantity called the potential scale reduction:

$$\hat{R} = \sqrt{\frac{\hat{\sigma}_+^2}{W}}. \tag{A.18}$$

It represents the factor by which the scale of the current distribution for θ might be reduced if $N_0 \to \infty$. In order to assess convergence, $\hat{R}$ has to be close to 1, otherwise it is likely that further simulations may improve the inference on θ.

If we are making inference on the simulation of a multivariate posterior distribution using an iterative algorithm as the Gibbs sampler, we have to compute the potential scale reduction $\hat{R}$ for all the sampled chains for each parameter.

A.3.3 Brooks-Gelman convergence diagnostic

In order to assess convergence for multivariate chains an extension of the method by Gelman and Rubin is required. Brooks and Gelman developed an extension to the multivariate case of the previously described methodology [BG98].
We compute the within-sequence covariance matrix $\boldsymbol{W}$ and the between-sequence covariance matrix $\boldsymbol{B}$:

$$\boldsymbol{W} = \frac{1}{M(N_0-1)}\sum_{j=1}^{M}\sum_{t=1}^{N_0}(\boldsymbol{\theta}_{jt}-\overline{\boldsymbol{\theta}}_{j\cdot})(\boldsymbol{\theta}_{jt}-\overline{\boldsymbol{\theta}}_{j\cdot})^T, \tag{A.19}$$

$$\frac{\boldsymbol{B}}{N_0} = \frac{1}{M-1}\sum_{j=1}^{M}\left(\overline{\boldsymbol{\theta}}_{j\cdot}-\overline{\boldsymbol{\theta}}_{\cdot\cdot}\right)\left(\overline{\boldsymbol{\theta}}_{j\cdot}-\overline{\boldsymbol{\theta}}_{\cdot\cdot}\right)^T, \tag{A.20}$$

The posterior covariance matrix is estimated by:

$$\hat{\boldsymbol{V}} = \frac{N_0-1}{N_0}\boldsymbol{W} + \left(1+\frac{1}{M}\right)\frac{\boldsymbol{B}}{N_0} \tag{A.21}$$

The potential scale reduction factor in the multivariate case is computed as:

$$\hat{R} = \max_{\mathbf{a}} \frac{\mathbf{a}^T\hat{\boldsymbol{V}}\mathbf{a}}{\mathbf{a}^T\boldsymbol{W}\mathbf{a}}. \tag{A.22}$$

Brooks and Gelman prove that (A.22) is equivalent to the following:

$$\hat{R} = \frac{N_0-1}{N_0} + \frac{M+1}{M}\lambda \tag{A.23}$$

where λ is the largest eigenvalue of the positive definite matrix $\boldsymbol{W}^{-1}\frac{\boldsymbol{B}}{N_0}$.

A.4 Hierarchical Bayesian models

In a Hierarchical bayesian model we assume that the unknown parameter θ is a random quantity sampled from a prior distribution $p(\theta|\lambda)$ where λ is called a hyperparameter. The inferential analysis on θ is then based on the posterior distribution:

$$p(\theta|\mathbf{y},\lambda) = \frac{p(\mathbf{y},\theta|\lambda)}{\int_\theta p(\mathbf{y},\theta|\lambda)\,d\theta} = \frac{L(\mathbf{y}|\theta)p(\theta|\lambda)}{\int_\theta L(\mathbf{y}|\theta)p(\theta|\lambda)\,d\theta}. \tag{A.24}$$

In practice λ is not known. So a second stage distribution called a hyperprior $p(\lambda)$ is specified. It follows that:

$$p(\theta|\mathbf{y}) = \frac{p(\mathbf{y}, \theta)}{p(\mathbf{y})} = \frac{\int_\lambda L(\mathbf{y}|\theta)p(\theta|\lambda)p(\lambda)\, d\lambda}{\int_\lambda \int_\theta L(\mathbf{y}|\theta)p(\theta|\lambda)\, d\theta\, d\lambda} \tag{A.25}$$

An alternative way to proceed is to estimate λ and plug its estimate $\hat{\boldsymbol{\lambda}}$ into (A.24) and make inference on θ using the approximated posterior $p(\theta|\mathbf{y}, \hat{\boldsymbol{\lambda}})$. This kind of approach, that integrates previously computed parameter estimates in the Bayesian machinery, is called empirical Bayesian analysis, in opposition to the fully Bayesian analysis required by (A.25) [BCG04].

Hierarchical modeling in the Bayesian framework allows one to completely specify complex models, and enhance their flexibility, by specifying hyperparameters and hyperpriors.

Appendix B

Useful results

B.1 Multivariate Normal conditional distribution

Let $\mathbf{X} = (\mathbf{X}_1, \mathbf{X}_2)^T$ be a random vector distributed as a multivariate normal $N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with:

$$\boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{bmatrix}, \quad \boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{bmatrix} \tag{B.1}$$

and $|\boldsymbol{\Sigma}_{22}| > 0$. Then the conditional distribution of $\mathbf{X}_1 | \mathbf{X}_2 = \mathbf{x}_2$ follows a normal distribution $N(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$ with mean [JW02]:

$$\boldsymbol{\mu}^* = \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\mathbf{x}_2 - \boldsymbol{\mu}_2) \tag{B.2}$$

and covariance matrix:

$$\boldsymbol{\Sigma}^* = \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}. \tag{B.3}$$

B.2 Change of variables in density functions

Let $\mathbf{X}$ be an n-dimensional random variable with joint density function given by f. If $\mathbf{Y} = H(\mathbf{X})$ where H is a differentiable bijective function then $\mathbf{Y}$ has a joint density function given by:

$$g(\mathbf{Y}) = f(\mathbf{X}) \left| \det \left(\frac{d\mathbf{X}}{d\mathbf{Y}} \right) \right| \tag{B.4}$$

where $\left|\det\left(\frac{d\mathbf{X}}{d\mathbf{Y}}\right)\right|$ is the determinant of the Jacobian of the inverse of $\mathbf{H}$ evaluated in $\mathbf{Y}$.

Appendix C

Posterior Calculations

C.1 Joint posterior parameters distribution (full Bayesian approach)

The likelihood function of the HE and LE training data $(\mathbf{y}_l, \mathbf{y}_h)$ is:

$$L(\mathrm{y}_h, \mathrm{y}_l|\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3) = L(\mathrm{y}_h|\mathrm{y}_l, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3)L(\mathrm{y}_l|\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3) \tag{C.1}$$

Since:

$$\begin{aligned} &L(\mathrm{y}_h|\mathrm{y}_l, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3) \propto \\ &\qquad \frac{1}{|\mathbf{Q}|^{1/2}} \exp\Big(-\frac{1}{2}(\mathrm{y}_h - \rho_0 \mathrm{y}_{l_1} - \delta_0 \mathbf{1}_{n_1})^T \mathbf{Q}^{-1} (\mathrm{y}_h - \rho_0 \mathrm{y}_{l_1} - \delta_0 \mathbf{1}_{n_1})\Big) \end{aligned} \tag{C.2}$$

$$L(\mathrm{y}_l|\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3) \propto \frac{1}{|\sigma_l^2 \mathbf{R}_l|^{1/2}} \exp\left(-\frac{1}{2\sigma_l^2}(\mathrm{y}_l - \mathbf{F}_l\boldsymbol{\beta}_l)^T \mathbf{R}_l^{-1}(\mathrm{y}_l - \mathbf{F}_l\boldsymbol{\beta}_l)\right) \tag{C.3}$$

then:

$$\begin{aligned} &L(\mathbf{y}_l, \mathbf{y}_h|\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3) \propto \\ &\qquad \frac{1}{|\mathbf{Q}|^{1/2}} \exp\left\{-\frac{1}{2\mathrm{v}_\rho\sigma_\rho^2}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})^T \mathbf{Q}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})\right\} \times \\ &\qquad \times \frac{1}{(\sigma_l^2)^{n/2}|\mathbf{R}_l|^{1/2}} \exp\left\{-\frac{1}{2\mathrm{v}_l\sigma_l^2}(\mathbf{y}_l - \mathbf{F}\boldsymbol{\beta}_l)^T \mathbf{R}_l^{-1}(\mathbf{y}_l - \mathbf{F}\boldsymbol{\beta}_l)\right\} \end{aligned} \tag{C.4}$$

where $\mathbf{Q} = \sigma_\rho^2\mathbf{W}_\rho + \sigma_\delta^2\mathbf{R}_\delta + \sigma_\epsilon^2\mathbf{I}_{n_1\times n_1}$, $\mathbf{W}_\rho = \mathbf{A}_1\mathbf{R}_\rho\mathbf{A}_1$, $\mathbf{A}_1 = \mathrm{diag}\{\mathrm{y}_l(\mathbf{x}_1), ..., \mathrm{y}_l(\mathbf{x}_{n_1})\}$ and the correlation matrices $\mathbf{R}_l$, $\mathbf{R}_\rho$ and $\mathbf{R}_\delta$ depend respectively on the unknown correlation parameters $\boldsymbol{\phi}_l$, $\boldsymbol{\phi}_\rho$ and $\boldsymbol{\phi}_\delta$.

The joint prior distribution for the unknown parameters $(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3)$ has the following form:

$$p(\boldsymbol{\theta}) = p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)p(\boldsymbol{\theta}_3) = p(\boldsymbol{\theta}_1|\boldsymbol{\theta}_2)p(\boldsymbol{\theta}_2)p(\boldsymbol{\theta}_3) \tag{C.5}$$

where:

$$\begin{aligned} p(\sigma_l^2) &\sim IG(\alpha_l, \gamma_l) \\ p(\sigma_\rho^2) &\sim IG(\alpha_\rho, \gamma_\rho) \\ p(\sigma_\delta^2) &\sim IG(\alpha_\delta, \gamma_\delta) \\ p(\sigma_\epsilon^2) &\sim IG(\alpha_\epsilon, \gamma_\epsilon) \end{aligned} \tag{C.6}$$

$$\begin{aligned} p(\boldsymbol{\beta}_l|\sigma_l^2) &\sim N(\mathbf{u}_l, \mathrm{v}_l\mathbf{I}_{(k+1)\times}(k+1)\sigma_l^2) \\ p(\rho_0|\sigma_\rho^2) &\sim N(\mathrm{u}_\rho, \mathrm{v}_\rho\sigma_\rho^2) \\ p(\delta_0|\sigma_\delta^2) &\sim N(\mathrm{u}_\delta, \mathrm{v}_\delta\sigma_\delta^2) \end{aligned} \tag{C.7}$$

$$\begin{aligned} p(\phi_{li}) &\sim G(a_l, b_l) \\ p(\phi_{\rho i}) &\sim G(a_\rho, b_\rho) \\ p(\phi_{\delta i}) &\sim G(a_\delta, b_\delta) \\ \forall \quad & i = 1, ..., k. \end{aligned} \tag{C.8}$$

It follows that the joint posterior distribution of $(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3)$ is:

$$
\begin{aligned}
p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2|\mathbf{y}_l, \mathbf{y}_h) = p(\boldsymbol{\theta})L(\mathbf{y}_l, \mathbf{y}_h|\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3) = p(\boldsymbol{\theta}_1|\boldsymbol{\theta}_2)p(\boldsymbol{\theta}_2)p(\boldsymbol{\theta}_3)L(\mathbf{y}_h|\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3) \propto \\
&\times \frac{1}{(\sigma_l^2)^{n/2}} \exp\left\{-\frac{1}{2\mathrm{v}_l\sigma_l^2}(\boldsymbol{\beta}_l - \mathbf{u}_l)^T(\boldsymbol{\beta}_l - \mathbf{u}_l)\right\} \times \\
&\times \frac{1}{(\sigma_\rho^2)^{1/2}} \exp\left\{-\frac{(\rho_0 - \mathrm{u}_\rho)^2}{2\mathrm{v}_\rho\sigma_\rho^2}\right\} \cdot \frac{1}{(\sigma_\delta^2)^{1/2}} \exp\left\{-\frac{(\delta_0 - \mathrm{u}_\delta)^2}{2\mathrm{v}_\rho\sigma_\delta^2}\right\} \times \\
&\times (\sigma_l^2)^{-(\alpha_l+1)} \exp\left\{-\frac{\gamma_l}{\sigma_l^2}\right\} \cdot (\sigma_\rho^2)^{-(\alpha_\rho+1)} \exp\left\{-\frac{\gamma_\rho}{\sigma_\rho^2}\right\} \times \\
&\times (\sigma_\delta^2)^{-(\alpha_\delta+1)} \exp\left\{-\frac{\gamma_\delta}{\sigma_\delta^2}\right\} \cdot (\sigma_\epsilon^2)^{-(\alpha_\epsilon+1)} \exp\left\{-\frac{\gamma_\epsilon}{\sigma_\epsilon^2}\right\} \times \\
&\times \prod_{i=1}^{k}\left(\phi_{li}^{(a_l-1)} \exp\{-b_l\phi_{li}\} \cdot \phi_{\rho i}^{(a_\rho-1)} \exp\{-b_\rho\phi_{\rho i}\} \cdot \phi_{\delta i}^{(a_\delta-1)} \exp\{-b_\delta\phi_{\delta i}\}\right) \times \\
&\times \frac{1}{|\mathbf{Q}|^{1/2}} \times \\
&\times \exp\left\{-\frac{1}{2}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})^T\mathbf{Q}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})\right\} \times \\
&\times \frac{1}{(\sigma_l^2)^{n/2}|\mathbf{R}_l|^{1/2}} \exp\left\{-\frac{1}{2\mathrm{v}_l\sigma_l^2}(\mathbf{y}_l - \mathbf{F}\boldsymbol{\beta}_l)^T\mathbf{R}_l^{-1}(\mathbf{y}_l - \mathbf{F}\boldsymbol{\beta}_l)\right\}.
\end{aligned}
\tag{C.9}
$$

C.2 Joint posterior parameters distribution (empirical Bayesian approach)

The likelihood function for the observed data is the same as (C.4) with the only difference that we omit the dependence on the correlation parameters $\boldsymbol{\theta}_3$

$$
\begin{aligned}
L(\mathbf{y}_l, \mathbf{y}_h|\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) \propto \\
&\frac{1}{|\mathbf{Q}|^{1/2}}\exp\left\{-\frac{1}{2\mathrm{v}_\rho\sigma_\rho^2}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})^T\mathbf{Q}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})\right\} \times \\
&\times \frac{1}{(\sigma_l^2)^{n/2}|\mathbf{R}_l|^{1/2}} \exp\left\{-\frac{1}{2\mathrm{v}_l\sigma_l^2}(\mathbf{y}_l - \mathbf{F}\boldsymbol{\beta}_l)^T\mathbf{R}_l^{-1}(\mathbf{y}_l - \mathbf{F}\boldsymbol{\beta}_l)\right\}
\end{aligned}
\tag{C.10}
$$

where $\mathbf{Q} = \sigma_\rho^2\mathbf{W}_\rho + \sigma_\delta^2\mathbf{R}_\delta + \sigma_\epsilon^2\mathbf{I}_{n_1\times n_1}$, $\mathbf{W}_\rho = \mathbf{A}_1\mathbf{R}_\rho\mathbf{A}_1$, $\mathbf{A}_1 = \mathrm{diag}\{\mathrm{y}_l(\mathbf{x}_1), ..., \mathrm{y}_l(\mathbf{x}_{n_1})\}$ and the correlation matrices $\mathbf{R}_l$, $\mathbf{R}_\rho$ and $\mathbf{R}_\delta$ depend respectively on the estimated correlation parameters $\hat{\boldsymbol{\phi}}_l$, $\hat{\boldsymbol{\phi}}_\rho$ and $\hat{\boldsymbol{\phi}}_\delta$.

The joint prior distribution for the unknown parameters $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2$ has the form:

$$p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) = p(\boldsymbol{\theta}_1|\boldsymbol{\theta}_2)p(\boldsymbol{\theta}_2) \tag{C.11}$$

where $p(\boldsymbol{\theta}_1|\boldsymbol{\theta}_2)$ and $p(\boldsymbol{\theta}_2)$ are defined as (C.7) and (C.6).

It follows that the joint posterior distribution of $(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$ is:

$$\begin{aligned}
p(\boldsymbol{\theta}_1,&\boldsymbol{\theta}_2|\mathbf{y}_l, \mathbf{y}_h) = p(\boldsymbol{\theta})L(\mathbf{y}_l, \mathbf{y}_h|\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) = p(\boldsymbol{\theta}_1|\boldsymbol{\theta}_2)p(\boldsymbol{\theta}_2)L(\mathbf{y}_h|\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) \propto \\
&\times \frac{1}{(\sigma_l^2)^{n/2}} \exp\left\{-\frac{1}{2\mathrm{v}_l\sigma_l^2}(\boldsymbol{\beta}_l - \mathbf{u}_l)^T(\boldsymbol{\beta}_l - \mathbf{u}_l)\right\} \times \\
&\times \frac{1}{(\sigma_\rho^2)^{1/2}} \exp\left\{-\frac{(\rho_0 - \mathrm{u}_\rho)^2}{2\mathrm{v}_\rho\sigma_\rho^2}\right\} \cdot \frac{1}{(\sigma_\delta^2)^{1/2}} \exp\left\{-\frac{(\delta_0 - \mathrm{u}_\delta)^2}{2\mathrm{v}_\rho\sigma_\delta^2}\right\} \times \\
&\times (\sigma_l^2)^{-(\alpha_l+1)} \exp\left\{-\frac{\gamma_l}{\sigma_l^2}\right\} \cdot (\sigma_\rho^2)^{-(\alpha_\rho+1)} \exp\left\{-\frac{\gamma_\rho}{\sigma_\rho^2}\right\} \times \\
&\times (\sigma_\delta^2)^{-(\alpha_\delta+1)} \exp\left\{-\frac{\gamma_\delta}{\sigma_\delta^2}\right\} \cdot (\sigma_\epsilon^2)^{-(\alpha_\epsilon+1)} \exp\left\{-\frac{\gamma_\epsilon}{\sigma_\epsilon^2}\right\} \times \\
&\times \frac{1}{|\mathbf{Q}|^{1/2}} \times \\
&\times \exp\left\{-\frac{1}{2}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})^T\mathbf{Q}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})\right\} \times \\
&\times \frac{1}{(\sigma_l^2)^{n/2}|\mathbf{R}_l|^{1/2}} \exp\left\{-\frac{1}{2\mathrm{v}_l\sigma_l^2}(\mathbf{y}_l - \mathbf{F}\boldsymbol{\beta}_l)^T\mathbf{R}_l^{-1}(\mathbf{y}_l - \mathbf{F}\boldsymbol{\beta}_l)\right\}.
\end{aligned} \tag{C.12}$$

Following the suggestion given in [QW08] a new parametrization for the variances is introduced:

$$(\sigma_l^2, \sigma_\rho^2, \tau_1, \tau_2) = \left(\sigma_l^2, \sigma_\rho^2, \frac{\sigma_\delta^2}{\sigma_\rho^2}, \frac{\sigma_\epsilon^2}{\sigma_\rho^2}\right). \tag{C.13}$$

We still refer to $(\sigma_l^2, \sigma_\rho^2, \tau_1, \tau_2)$ as $\boldsymbol{\theta}_2$.

This new parametrization is very convenient in the perspective of the computation of the full conditional distributions, that will be used in the Gibbs sampler.

The new joint posterior distribution for all the unknown parameters with the new parametrization is computed with the change of variables rule described in Appendix B.2.

Thus, if $\mathbf{X} = (\sigma_l^2, \sigma_\rho^2, \sigma_\delta^2, \sigma_\epsilon^2)$ and $\mathbf{H}(\mathbf{X}) = \mathbf{Y} = \left(\sigma_l^2, \sigma_\rho^2, \frac{\sigma_\delta^2}{\sigma_\rho^2}, \frac{\sigma_\epsilon^2}{\sigma_\rho^2}\right)$, then $\left|\det\left(\frac{d\mathbf{X}}{d\mathbf{Y}}\right)\right| = \sigma_\rho^4$ and the posterior distribution (C.9) with the new parametrization (C.13) becomes:

$$
\begin{aligned}
&p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2 | \mathbf{y}_l,\mathbf{y}_h) \propto \\
&\qquad \frac{1}{(\mathrm{v}_l\sigma_l^2)^{n/2}} \exp\left\{-\frac{1}{2\mathrm{v}_l\sigma_l^2}(\boldsymbol{\beta}_l - \mathbf{u}_l)^T\mathbf{R}_l^{-1}(\boldsymbol{\beta}_l - \mathbf{u}_l)\right\} \times \\
&\qquad \times \frac{1}{(\mathrm{v}_\rho\sigma_\rho^2)^{1/2}} \exp\left\{-\frac{(\rho_0 - \mathrm{u}_\rho)^2}{2\mathrm{v}_\rho\sigma_\rho^2}\right\} \cdot \frac{1}{(\mathrm{v}_\delta\tau_1\sigma_\rho^2)^{1/2}} \exp\left\{-\frac{(\delta_0 - \mathrm{u}_\delta)^2}{2\mathrm{v}_\rho\tau_1\sigma_\rho^2}\right\} \times \\
&\qquad \times (\sigma_l^2)^{-(\alpha_l+1)} \exp\left\{-\frac{\gamma_l}{\sigma_l^2}\right\} \cdot (\sigma_\rho^2)^{-(\alpha_\rho+1)} \exp\left\{-\frac{\gamma_\rho}{\sigma_\rho^2}\right\} \times \\
&\qquad \times (\tau_1\sigma_\rho^2)^{-(\alpha_\delta+1)} \exp\left\{-\frac{\gamma_\delta}{\tau_1\sigma_\rho^2}\right\} \cdot (\tau_2\sigma_\rho^2)^{-(\alpha_\epsilon+1)} \exp\left\{-\frac{\gamma_\epsilon}{\tau_2\sigma_\rho^2}\right\} \cdot \sigma_\rho^4 \times \\
&\qquad \times \prod_{i=1}^{k} \big(\phi_{li}^{(a_l-1)} \exp\{-b_l\phi_{li}\} \cdot \phi_{\rho i}^{(a_\rho-1)} \exp\{-b_\rho\phi_{\rho i}\} \cdot \phi_{\delta i}^{(a_\delta-1)} \exp\{-b_\delta\phi_{\delta i}\}\big) \times \\
&\qquad \times \frac{1}{(\sigma_\rho^2)^{n_1/2}|\mathbf{M}|^{1/2}} \times \\
&\qquad \times \exp\left\{-\frac{1}{2\mathrm{v}_\rho\sigma_\rho^2}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})^T\mathbf{M}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})\right\} \times \\
&\qquad \times \frac{1}{(\sigma_l^2)^{n/2}|\mathbf{R}_l|^{1/2}} \exp\left\{-\frac{1}{2\mathrm{v}_l\sigma_l^2}(\mathbf{y}_l - \mathbf{F}\boldsymbol{\beta}_l)^T\mathbf{R}_l^{-1}(\mathbf{y}_l - \mathbf{F}\boldsymbol{\beta}_l)\right\}
\end{aligned}
\tag{C.14}
$$

where $\mathbf{M} = \mathbf{W}_\rho + \tau_1\mathbf{R}_\delta + \tau_2\mathbf{I}_{n_1\times n_1}$.

C.3 Full conditional distributions for the mean and variance parameters

From the joint posterior distribution (C.14) we compute the full conditional distributions for all the mean and variance parameters.

C.3.1 Full conditional distribution for $\boldsymbol{\beta}_l$

$$\begin{aligned}
&p(\boldsymbol{\beta}_l|\mathbf{y}_h,\mathbf{y}_l,\overline{\boldsymbol{\beta}_l})\\
&\qquad\propto \exp\left(-\frac{1}{2\sigma_l^2}\left[\frac{(\boldsymbol{\beta}_l-\mathbf{u}_l)^T(\boldsymbol{\beta}_l-\mathbf{u}_l)}{\mathrm{v}_l}+(\mathbf{y}_l-\mathbf{F}_l\boldsymbol{\beta}_l)^T\mathbf{R}_l^{-1}(\mathbf{y}_l-\mathbf{F}_l\boldsymbol{\beta}_l)\right]\right)=\\
&\qquad=\exp\Bigg(-\frac{1}{2\sigma_l^2}\Bigg[\frac{\boldsymbol{\beta}_l^T\boldsymbol{\beta}_l}{\mathrm{v}_l}-2\frac{\boldsymbol{\beta}_l^T\mathbf{u}_l}{\mathrm{v}_l}+\frac{\mathbf{u}_l^T\mathbf{u}_l}{\mathrm{v}_l}+\\
&\qquad\qquad\qquad+\mathbf{y}_l^T\mathbf{R}_l^{-1}\mathbf{y}_l-2\boldsymbol{\beta}_l^T\mathbf{F}_l^T\mathbf{R}_l^{-1}\mathbf{y}_l+\boldsymbol{\beta}_l^T\mathbf{F}_l^T\mathbf{R}_l^{-1}\mathbf{F}_l\boldsymbol{\beta}_l\Bigg]\Bigg)\\
&\qquad\propto \exp\left(-\frac{1}{2\sigma_l^2}\left[\frac{\boldsymbol{\beta}_l^T\boldsymbol{\beta}_l}{\mathrm{v}_l}-2\frac{\boldsymbol{\beta}_l^T\mathbf{u}_l}{\mathrm{v}_l}-2\boldsymbol{\beta}_l^T\mathbf{F}_l^T\mathbf{R}_l^{-1}\mathbf{y}_l+\boldsymbol{\beta}_l^T\mathbf{F}_l^T\mathbf{R}_l^{-1}\mathbf{F}_l\boldsymbol{\beta}_l\right]\right)=\\
&\qquad=\exp\left(-\frac{1}{2\sigma_l^2}\left[\boldsymbol{\beta}_l^T\mathbf{A}\boldsymbol{\beta}_l-2\boldsymbol{\beta}_l^T\mathbf{B}\right]\right)\\
&\qquad\propto \exp\left(-\frac{1}{2\sigma_l^2}\left[\boldsymbol{\beta}_l^T\mathbf{A}\boldsymbol{\beta}_l-2\boldsymbol{\beta}_l^T\mathbf{A}\mathbf{A}^{-1}\mathbf{B}+\mathbf{c}-\mathbf{c}\right]\right)\\
&\qquad=\exp\left(-\frac{1}{2\sigma_l^2}\left[(\boldsymbol{\beta}_l-\mathbf{A}^{-1}\mathbf{B})^T\mathbf{A}(\boldsymbol{\beta}_l-\mathbf{A}^{-1}\mathbf{B})-\mathbf{c}\right]\right)
\end{aligned}$$

where $\mathbf{A}=\frac{1}{\mathrm{v}_l}\mathbf{I}_{(k+1)\times(k+1)}+\mathbf{F}_l^T\mathbf{R}_l^{-1}\mathbf{F}_l$, $\boldsymbol{B}=\frac{\mathbf{u}_l}{\mathrm{v}_l}+\mathbf{F}_l^T\mathbf{R}_l^{-1}\mathbf{y}_l$ and $\mathbf{c}$ is an arbitrary constant vector.

$$\begin{aligned}
\boldsymbol{\beta}_l|\mathbf{y}_h,\mathbf{y}_l,\overline{\boldsymbol{\beta}_l},\boldsymbol{\theta}_2\sim N\Bigg(&\left[\frac{1}{\mathrm{v}_l}\mathbf{I}_{(k+1)\times(k+1)}+\mathbf{F}_l^T\mathbf{R}_l^{-1}\mathbf{F}_l\right]^{-1}\left(\frac{\mathbf{u}_l}{\mathrm{v}_l}+\mathbf{F}_l^T\mathbf{R}_l^{-1}\mathbf{y}_l\right),\\
&\left[\frac{1}{\mathrm{v}_l}\mathbf{I}_{(k+1)\times(k+1)}+\mathbf{F}_l^T\mathbf{R}_l^{-1}\mathbf{F}_l\right]^{-1}\sigma_l^2\Bigg)
\end{aligned}\tag{C.15}$$

C.3.2 Full conditional distribution for ρ_0

$$
\begin{aligned}
&p(\rho_0|\mathbf{y}_h, \mathbf{y}_l, \overline{\rho_0})\\
&\quad\propto \exp\left(-\frac{1}{2\sigma_\rho^2}\left[\frac{(\rho_0-\mathrm{u}_\rho)^2}{\mathrm{v}_\rho} + (\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})^T\mathbf{M}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})\right]\right)\\
&\quad\propto \exp\left(-\frac{1}{2\sigma_\rho^2}\left[\frac{(\rho_0^2 - 2\mathrm{u}_\rho\rho_0)}{\mathrm{v}_\rho} + \mathbf{y}_{l_1}^T\mathbf{M}^{-1}\mathbf{y}_{l_1}\rho_0^2 - 2\mathbf{y}_{l_1}^T\mathbf{M}^{-1}(\mathbf{y}_h - \delta_0\mathbf{1}_{n_1})\rho_0\right]\right) =\\
&\quad= \exp\left(-\frac{1}{2\sigma_\rho^2}\left[a\rho_0^2 - 2b\rho_0\right]\right)\\
&\quad\propto \exp\left(-\frac{1}{2\sigma_\rho^2 a^{-1}}\left[\rho_0^2 - 2a^{-1}b\rho_0 + c - c\right]\right) =\\
&\quad= \exp\left(-\frac{1}{2\sigma_\rho^2 a^{-1}}\left[(\rho_0 - a^{-1}b)^2 - c\right]\right)
\end{aligned}
$$

where $a = \left(\frac{1}{\mathrm{v}_\rho} + \mathbf{y}_{l_1}^T\mathbf{M}^{-1}\mathbf{y}_{l_1}\right)$, $b = \left(\frac{\mathrm{u}_\rho}{\mathrm{v}_\rho} + \mathbf{y}_{l_1}^T\mathbf{M}^{-1}(\mathbf{y}_h - \delta_0\mathbf{1}_{n_1})\right)$ and c is an arbitrary constant.

$$
\rho_0|\mathbf{y}_h, \mathbf{y}_l, \overline{\rho_0}, \boldsymbol{\theta}_2 \sim N\left(\frac{\frac{\mathrm{u}_\rho}{\mathrm{v}_\rho} + \mathbf{y}_{l_1}^T\mathbf{M}^{-1}(\mathbf{y}_h - \delta_0\mathbf{1}_{n_1})}{\frac{1}{\mathrm{v}_\rho} + \mathbf{y}_{l_1}^T\mathbf{M}^{-1}\mathbf{y}_{l_1}}, \frac{\sigma_\rho^2}{\frac{1}{\mathrm{v}_\rho} + \mathbf{y}_{l_1}^T\mathbf{M}^{-1}\mathbf{y}_{l_1}}\right) \tag{C.16}
$$

C.3.3 Full conditional distribution for δ_0

$$
\begin{aligned}
&p(\delta_0|\mathbf{y}_h, \mathbf{y}_l, \overline{\delta_0})\\
&\quad\propto \exp\left(-\frac{1}{2\sigma_\rho^2}\left[\frac{(\delta_0-\mathrm{u}_\delta)^2}{\mathrm{v}_\delta\tau_1} + (\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})^T\mathbf{M}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})\right]\right)\\
&\quad\propto \exp\left(-\frac{1}{2\sigma_\rho^2}\left[\frac{(\delta_0^2 - 2\mathrm{u}_\delta\delta_0)}{\mathrm{v}_\delta\tau_1} + \mathbf{1}_{n_1}^T\mathbf{M}^{-1}\mathbf{1}_{n_1}\delta_0^2 - 2\cdot\mathbf{1}_{n_1}^T\mathbf{M}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1})\delta_0\right]\right) =\\
&\quad= \exp\left(-\frac{1}{2\sigma_\rho^2}\left[a\delta_0^2 - 2b\delta_0\right]\right)\\
&\quad\propto \exp\left(-\frac{1}{2\sigma_\rho^2 a^{-1}}\left[\delta_0^2 - 2a^{-1}b\delta_0 + c - c\right]\right) =\\
&\quad= \exp\left(-\frac{1}{2\sigma_\rho^2 a^{-1}}\left[(\delta_0 - a^{-1}b)^2 - c\right]\right)
\end{aligned}
$$

where $a = \left(\frac{1}{\mathrm{v}_\delta\tau_1} + \mathbf{1}_{n_1}^T\mathbf{M}^{-1}\mathbf{1}_{n_1}\right)$, $b = \left(\frac{\mathrm{u}_\delta}{\mathrm{v}_\delta\tau_1} + \mathbf{1}_{n_1}^T\mathbf{M}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1})\right)$ and c is an arbitrary constant.

$$\delta_0|\mathbf{y}_h,\mathbf{y}_l,\overline{\delta_0},\boldsymbol{\theta}_2 \sim N\left(\frac{\frac{\mathrm{u}_\delta}{\mathrm{v}_\delta\tau_1}+\mathbf{1}_{n_1}^T\mathbf{M}^{-1}(\mathbf{y}_h-\rho_0\mathbf{y}_{l_1})}{\frac{1}{\mathrm{v}_\delta\tau_1}+\mathbf{1}_{n_1}^T\mathbf{M}^{-1}\mathbf{1}_{n_1}},\frac{\sigma_\rho^2}{\frac{1}{\mathrm{v}_\delta\tau_1}+\mathbf{1}_{n_1}^T\mathbf{M}^{-1}\mathbf{1}_{n_1}}\right) \quad \text{(C.17)}$$

C.3.4 Full conditional distribution for σ_l^2

$$\begin{aligned}
&p(\sigma_l^2|\mathbf{y}_h,\mathbf{y}_l,\overline{\sigma_l^2})\\
&\quad\propto(\sigma_l^2)^{-(\alpha_l+1)}\exp\left(\frac{-\gamma_l}{\sigma_l^2}\right)\cdot(\sigma_l^2)^{-\frac{k+1}{2}}\exp\left(-\frac{1}{2\mathrm{v}_l\sigma_l^2}(\boldsymbol{\beta}_l-\mathbf{u}_l)^T(\boldsymbol{\beta}_l-\mathbf{u}_l)\right)\times\\
&\qquad\times(\sigma_l^2)^{-\frac{n}{2}}\exp\left(-\frac{1}{2\sigma_l^2}(\mathbf{y}_l-\mathbf{F}_l\boldsymbol{\beta}_l)^T\mathbf{R}_l^{-1}(\mathbf{y}_l-\mathbf{F}_l\boldsymbol{\beta}_l)\right)\\
&\quad\propto(\sigma_l^2)^{-(\frac{n}{2}+\frac{k+1}{2}+\alpha_l+1)}\exp\Bigg(-\frac{1}{2\sigma_l^2}\Bigg[\frac{(\boldsymbol{\beta}_l-\mathbf{u}_l)^T(\boldsymbol{\beta}_l-\mathbf{u}_l)}{2\mathrm{v}_l}+\\
&\qquad\qquad+\frac{(\mathbf{y}_l-\mathbf{F}_l\boldsymbol{\beta}_l)^T\mathbf{R}_l^{-1}(\mathbf{y}_l-\mathbf{F}_l\boldsymbol{\beta}_l)}{2}+\gamma_l\Bigg]\Bigg)
\end{aligned}$$

$$\begin{aligned}
\sigma_l^2|\mathbf{y}_h,\mathbf{y}_l,\boldsymbol{\theta}_1,\overline{\sigma_l^2}\sim IG\Bigg(&\frac{n}{2}+\frac{k+1}{2}+\alpha_l,\\
&\frac{1}{2}\frac{(\boldsymbol{\beta}_l-\mathbf{u}_l)^T(\boldsymbol{\beta}_l-\mathbf{u}_l)}{\mathrm{v}_l}+\frac{1}{2}(\mathbf{y}_l-\mathbf{F}_l\boldsymbol{\beta}_l)'\mathbf{R}_l^{-1}(\mathbf{y}_l-\mathbf{F}_l\boldsymbol{\beta}_l)+\gamma_l\Bigg)
\end{aligned} \quad \text{(C.18)}$$

C.3.5 Full conditional distribution for σ^2_ρ

$$
\begin{aligned}
&p(\sigma^2_\rho|\mathbf{y}_h, \mathbf{y}_l, \overline{\sigma^2_\rho}) \\
&\quad\propto (\sigma^2_\rho)^{-(\alpha_\rho+1)} \exp\left(-\frac{\gamma_\rho}{\sigma^2_\rho}\right) \cdot (\sigma^2_\rho)^{-(\alpha_\delta+1)} \exp\left(-\frac{\gamma_\delta}{(\sigma^2_\rho\tau_1)}\right) \times \\
&\qquad \times (\sigma^2_\rho)^{-(\alpha_\epsilon+1)} \exp\left(-\frac{\gamma_\epsilon}{(\sigma^2_\rho\tau_2)}\right) (\sigma^2_\rho)^2 \times \\
&\qquad \times (\sigma^2_\rho)^{-\frac{1}{2}} \exp\left(-\frac{(\rho_0 - \mathrm{u}_\rho)^2}{2\mathrm{v}_\rho\sigma^2_\rho}\right) \cdot (\sigma^2_\rho)^{-\frac{1}{2}} \exp\left(-\frac{(\delta_0 - \mathrm{u}_\delta)^2}{2\mathrm{v}_\delta\sigma^2_\rho\tau_1}\right) \times \\
&\qquad \times (\sigma^2_\rho)^{-\frac{n_1}{2}} \exp\left(-\frac{1}{2\sigma^2_\rho}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})^T\mathbf{M}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})\right) = \\
&\quad = (\sigma^2_\rho)^{-(\alpha_\rho+1+\alpha_\delta+1+\alpha_\epsilon+1-2+\frac{1}{2}+\frac{1}{2}+\frac{n_1}{2})} \times \\
&\qquad \times \exp\Bigg(-\frac{1}{\sigma^2_\rho}\Bigg[\gamma_\rho + \frac{\gamma_\delta}{\tau_1} + \frac{\gamma_\epsilon}{\tau_2} + \frac{(\rho_0 - \mathrm{u}_\rho)^2}{2\mathrm{v}_\rho} + \frac{(\delta_0 - \mathrm{u}_\delta)^2}{2\mathrm{v}_\delta\tau_1} + \\
&\qquad\qquad + \frac{1}{2}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})^T\mathbf{M}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})\Bigg]\Bigg)
\end{aligned}
$$

$$
\begin{aligned}
\sigma^2_\rho|\mathbf{y}_h, \mathbf{y}_l, \overline{\sigma^2_\rho} \sim IG\Bigg(&\frac{n_1}{2} + 1 + \alpha_\rho + \alpha_\delta + \alpha_\epsilon, \frac{(\rho_0 - \mathrm{u}_\rho)^2}{2\mathrm{v}_\rho} + \frac{(\delta_0 - \mathrm{u}_\delta)^2}{2\mathrm{v}_\delta\tau_1} + \\
&+ \gamma_\rho + \frac{\gamma_\delta}{\tau_1} + \frac{\gamma_\epsilon}{\tau_2} + \frac{1}{2}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})'\mathbf{M}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})\Bigg)
\end{aligned}
\tag{C.19}
$$

C.3.6 Full conditional distribution for (τ_1, τ_2)

$$
\begin{aligned}
&p(\tau_1,\tau_2|\mathbf{y}_h, \mathbf{y}_l, \overline{\tau_1, \tau_2}) \\
&\quad\propto (\tau_1)^{-(\alpha_\delta+1)} \exp\left(-\frac{\gamma_\delta}{(\sigma_\rho^2\tau_1)}\right) \cdot (\tau_2)^{-(\alpha_\epsilon+1)} \exp\left(-\frac{\gamma_\epsilon}{(\sigma_\rho^2\tau_2)}\right) \times \\
&\qquad \times (\tau_1)^{-\frac{1}{2}} \exp\left(-\frac{1}{2\mathrm{v}_\delta\sigma_\rho^2\tau_1}(\delta_0 - \mathrm{u}_\delta)^2\right) \times \\
&\qquad \times \frac{1}{|\mathbf{M}|^{1/2}} \exp\left(-\frac{1}{2\sigma_\rho^2}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})^T\mathbf{M}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})\right) \\
&\quad\propto \tau_1^{-(\alpha_\delta+\frac{3}{2})}\tau_2^{-(\alpha_\epsilon+1)} \exp\left(-\frac{1}{\tau_1}\left[\frac{\gamma_\delta}{\sigma_\rho^2} + \frac{(\delta_0 - \mathrm{u}_\delta)^2}{2\mathrm{v}_\delta\sigma_\rho^2}\right] - \frac{\gamma_\epsilon}{(\sigma_\rho^2\tau_2)}\right) \times \\
&\qquad \times \frac{1}{|\mathbf{M}|^{1/2}} \exp\left(-\frac{1}{2\sigma_\rho^2}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})^T\mathbf{M}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})\right)
\end{aligned}
\tag{C.20}
$$

C.4 Full conditional distributions for the correlation parameters

Consider the join posterior distribution of the parameters $(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3)$ of equation (C.9). If we apply the re-parametrization (C.13) to the variance parameters, the joint posterior distribution in the fully Bayesian case becomes:

$$
\begin{aligned}
&p(\boldsymbol{\theta}|\mathbf{y}_l,\mathbf{y}_h) \propto \\
&\quad \frac{1}{(\mathrm{v}_l\sigma_l^2)^{n/2}} \exp\left\{-\frac{1}{2\mathrm{v}_l\sigma_l^2}(\boldsymbol{\beta}_l-\mathbf{u}_l)^T\mathbf{R}_l^{-1}(\boldsymbol{\beta}_l-\mathbf{u}_l)\right\}\times \\
&\quad \times \frac{1}{(\mathrm{v}_\rho\sigma_\rho^2)^{1/2}} \exp\left\{-\frac{(\rho_0-\mathrm{u}_\rho)^2}{2\mathrm{v}_\rho\sigma_\rho^2}\right\}\cdot\frac{1}{(\mathrm{v}_\delta\tau_1\sigma_\rho^2)^{1/2}} \exp\left\{-\frac{(\delta_0-\mathrm{u}_\delta)^2}{2\mathrm{v}_\rho\tau_1\sigma_\rho^2}\right\}\times \\
&\quad \times (\sigma_l^2)^{-(\alpha_l+1)}\exp\left\{-\frac{\gamma_l}{\sigma_l^2}\right\}\cdot(\sigma_\rho^2)^{-(\alpha_\rho+1)}\exp\left\{-\frac{\gamma_\rho}{\sigma_\rho^2}\right\}\times \\
&\quad \times (\tau_1\sigma_\rho^2)^{-(\alpha_\delta+1)}\exp\left\{-\frac{\gamma_\delta}{\tau_1\sigma_\rho^2}\right\}\cdot(\tau_2\sigma_\rho^2)^{-(\alpha_\epsilon+1)}\exp\left\{-\frac{\gamma_\epsilon}{\tau_2\sigma_\rho^2}\right\}\cdot\sigma_\rho^4\times \\
&\quad \times \prod_{i=1}^{k}\big(\phi_{li}^{(a_l-1)}\exp\{-b_l\phi_{li}\}\cdot\phi_{\rho i}^{(a_\rho-1)}\exp\{-b_\rho\phi_{\rho i}\}\cdot\phi_{\delta i}^{(a_\delta-1)}\exp\{-b_\delta\phi_{\delta i}\}\big)\times \\
&\quad \times \frac{1}{(\sigma_\rho^2)^{n_1/2}|\mathbf{M}|^{1/2}}\times \\
&\quad \times \exp\left\{-\frac{1}{2\mathrm{v}_\rho\sigma_\rho^2}(\mathbf{y}_h-\rho_0\mathbf{y}_{l_1}-\delta_0\mathbf{1}_{n_1})^T\mathbf{M}^{-1}(\mathbf{y}_h-\rho_0\mathbf{y}_{l_1}-\delta_0\mathbf{1}_{n_1})\right\}\times \\
&\quad \times \frac{1}{(\sigma_l^2)^{n/2}|\mathbf{R}_l|^{1/2}}\exp\left\{-\frac{1}{2\mathrm{v}_l\sigma_l^2}(\mathbf{y}_l-\mathbf{F}\boldsymbol{\beta}_l)^T\mathbf{R}_l^{-1}(\mathbf{y}_l-\mathbf{F}\boldsymbol{\beta}_l)\right\}
\end{aligned}
$$

where $\mathbf{M}=\mathbf{W}_\rho+\tau_1\mathbf{R}_\delta+\tau_2\mathbf{I}_{n_1\times n_1}$, $\mathbf{W}_\rho=\mathbf{A}_1\mathbf{R}_\rho\mathbf{A}_1$, $\mathbf{A}_1=\mathrm{diag}\{\mathrm{y}_l(\mathbf{x}_1),...,\mathrm{y}_l(\mathbf{x}_{n_1})\}$ and the correlation matrices $\mathbf{R}_l$, $\mathbf{R}_\rho$ and $\mathbf{R}_\delta$ depend respectively on the unknown correlation parameters $\boldsymbol{\phi}_l$, $\boldsymbol{\phi}_\rho$ and $\boldsymbol{\phi}_\delta$.

In order to apply the two-step MCMC algorithm illustrated in Section 2.1 to approximate the predictive posterior distribution, we need to generate $\left(\boldsymbol{\theta}_1^{(i)},\boldsymbol{\theta}_2^{(i)},\boldsymbol{\theta}_3^{(i)}\right)$ with $i=1,...,K$ from the joint posterior distribution of the unknown parameters. We still sample the mean and variance parameters from the same full conditional distributions computed in Appendix C.3. Plus we have to sample the correlation parameters form the following posterior distribution:

$$p(\boldsymbol{\theta}_3|\mathbf{y}_l,\mathbf{y}_h,\boldsymbol{\theta}_1,\boldsymbol{\theta}_2)\propto p(\boldsymbol{\theta}_3)L(\mathbf{y}_l,\mathbf{y}_h|\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\boldsymbol{\theta}_3)$$

that can be split in $3k$ full conditional distributions (where k is the number of the input variables) since all the correlation parameters are independent from each other and from $(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$:

$$p(\phi_{li}|\mathbf{y}_l, \mathbf{y}_h, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \overline{\boldsymbol{\theta}_3}) \propto \\ \phi_{li}^{(a_l-1)} \frac{1}{|\mathbf{R}_l|^{1/2}} \exp\left\{-\left[b_l\phi_{li} + \frac{1}{2\mathrm{v}_l\sigma_l^2}(\mathbf{y}_l - \mathbf{F}_l\boldsymbol{\beta}_l)^T\mathbf{R}_l^{-1}(\mathbf{y}_l - \mathbf{F}_l\boldsymbol{\beta}_l)\right]\right\} \tag{C.21}$$

$$p(\phi_{\rho i}|\mathbf{y}_l, \mathbf{y}_h, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \overline{\boldsymbol{\theta}_3}) \propto \phi_{\rho i}^{(a_\rho-1)} \frac{1}{|\mathbf{M}|^{1/2}} \times \\ \times \exp\left\{-\left[b_\rho\phi_{\rho i} + \frac{1}{2\mathrm{v}_\rho\sigma_\rho^2}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})^T\mathbf{M}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})\right]\right\} \tag{C.22}$$

$$p(\phi_{\delta i}|\mathbf{y}_l, \mathbf{y}_h, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \overline{\boldsymbol{\theta}_3}) \propto \phi_{\delta i}^{(a_\delta-1)} \frac{1}{|\mathbf{M}|^{1/2}} \times \\ \times \exp\left\{-\left[b_\delta\phi_{\delta i} + \frac{1}{2\mathrm{v}_\rho\sigma_\delta^2}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})^T\mathbf{M}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})\right]\right\} \tag{C.23}$$

with $i = 1, ..., k$.
Such full conditional distributions do not allow direct sample draws, so we have to use Metropolis-within-Gibbs algorithm. For computational convenience, to minimize the number of matrix inversions and determinants at each Metropolis-within-Gibbs iteration, it may seem reasonable to draw samples from the following two multivariate distributions:

$$p(\boldsymbol{\phi}_l|\mathbf{y}_l, \mathbf{y}_h, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \overline{\boldsymbol{\phi}_l}) \propto \\ \prod_{i=1}^{k} \phi_{li}^{(a_l-1)} \frac{1}{|\mathbf{R}_l|^{1/2}} \exp\left\{-b_l\sum_{i=1}^{k}\phi_{li} + \frac{1}{2\mathrm{v}_l\sigma_l^2}(\mathbf{y}_l - \mathbf{F}_l\boldsymbol{\beta}_l)^T\mathbf{R}_l^{-1}(\mathbf{y}_l - \mathbf{F}_l\boldsymbol{\beta}_l)\right\} \tag{C.24}$$

$$p(\boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta|\mathbf{y}_l, \mathbf{y}_h, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \overline{\boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta}) \propto \\ \prod_{i=1}^{k}\left(\phi_{\rho i}^{(a_\rho-1)}\phi_{\delta i}^{(a_\delta-1)}\right) \exp\left\{-b_\rho\sum_{i=1}^{k}\phi_{\rho i} - b_\delta\sum_{i=1}^{k}\phi_{\delta i}\right\} \times \\ \times \frac{1}{|\mathbf{M}|^{1/2}} \exp\left\{-\frac{1}{2\mathrm{v}_\rho\sigma_\rho^2}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})^T\mathbf{M}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})\right\}. \tag{C.25}$$

Appendix D

The GPML Toolbox

The Gaussian Process for Machine Learning (GPML) Toolbox is a Matlab 7.x (and Octave 3.2.x) implementation of a variety of Gaussian Process models for inference and prediction, developed by Rasmussen and Nickisch. It implements the algorithms and techniques illustrated in [RW06]. It is a very powerful and efficient tool that allows the user to specify a large variety of model structures and inference procedures, thanks to its modular structure.
The GPML Toolbox will come in handy at a certain point for our purposes, so here we provide a concise description of its main functionalities and working principles. For a complete discussion on its features refer to [RN10] and [RN11].

Gaussian processes are used to specify distributions over function without having to stick to a specific functional form. Thus a GP prior distribution on an unknown latent function:

$$f \sim GP(m_{\phi}(\mathbf{x}), k_{\psi}(\mathbf{x}, \mathbf{x}')), \tag{D.1}$$

is fully determined by its mean function:

$$m_{\phi}(\mathbf{x}) = E[f(\mathbf{x})], \tag{D.2}$$

and covariance function:

$$k_{\psi}(\mathbf{x}, \mathbf{x}') = E[(f(\mathbf{x}) - m_{\phi}(\mathbf{x}))(f(\mathbf{x}') - m_{\phi}(\mathbf{x}'))]. \tag{D.3}$$

Both (D.2) and (D.3) depend on some unknown parameters (ϕ, ψ) that need to be fit using the information carried by the data $\mathbf{y}$ through a specific likelihood function, $p(\mathbf{y}|f)$. This is done by maximizing the log marginal likelihood as explained in Chapter 5 of [RW06].

The inference techniques adopted in this phase significatively depend on the form of the likelihood function. As shown in the previous sections, in our case we deal only with Gaussian likelihood functions. As explained in detail in [RW06] this eases a lot the computations because it is the only case in which it is possible to carry out exact inference, as the analytic expressions of the marginal likelihood and its gradient with respect to the unknown parameters are readily obtainable.

The function `minimize` by Rasmussen and Nickisch implements a very efficient optimizer based on conjugate gradients and it is used to find the values of the unknown parameters that minimize the negative log marginal likelihood.

Once the unknown parameters are suitably fit, the prior distribution (D.1) is fully specified and the next step is the computation of predictive distributions at new input points $\mathbf{x}_0$.

The function `gp`, updated with the estimates of the unknown parameters, computes the predictions at the new inputs as the approximate marginal predictive mean $E[p(\mathrm{y}^*|\mathrm{y})]$ and also the predictive variance $Var[p(\mathrm{y}^*|\mathrm{y})]$.

It has to be pointed out that the approach implemented in the GPML toolbox significatively differs from the empirical Bayesian approach illustrated in the previous sections.

So far we adopted MCMC techniques to treat all the integrals involved in the Bayesian inference computations. In particular we used a MCMC sampling technique in order to make inference on the distributions of the of unknown parameters $(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$ given the prior information and the information carried by the observations.

On the contrary, Rasmussen and Nickisch rely on a so-called type II Maximum Likelihood approximation.

Bibliography

[Bay+07] J. O. Bayarri BM. J.and Berger et al. “Computer model validation with functional output.” In: *Annals of Statistics* 35.No. 5 (2007), pp. 1874–1906.

[BCG04] Sudipto Banerjee, P. Carlin Carlin, and Alan E. Gelfand. *Hierarchical Modelinh and Analysis for Spatial Data.* Boca Raton,FL: Chapman & Hall / CRC, 2004.

[BG98] Stephen P. Brooks and Andrew Gelman. “General Methods for Monitoring Convergence of Iterative Simulations.” In: *Journal of Computational and Graphical Statistics* 7.No. 4 (1998), pp. 435–455.

[CC07] Bianca M. Colosimo and Enrique del Castillo. *Bayesian Process Monitoring, Control and Optimization.* Boca Raton,FL: Chapman & Hall/CRC, 2007.

[CG92] George Casella and Edward I. George. “Explaining the Gibbs Sampler.” In: *The American Statistician* 46.No. 3 (1992), pp. 167–174.

[CG95] Siddhartha Chib and Edward Greenberg. “Understanding the Metropolis-Hastings Algorithm.” In: *The American Statistician* 49.No. 4 (1995), pp. 327–335.

[CP11] Bianca Maria Colosimo and Massimo Pacella. “Gaussian process for spatial modeling of profiles and surfaces.” Paper in progress. Feb. 2011.

[Cre93] Noel A.C. Cressie. *Statistics for Spatial Data. Revised Edition.* New York, NY: Wiley & Sons, 1993.

[Est+02] W. T. Estler et al. “Large-Scale Metrology - An Update.” In: *CIRP Annals - Manufacturing Technology* 51.2 (2002), pp. 587–609.

[Fra+09] Fiorenzo Franceschini et al. “Mobile Spatial coordinate Measuring System (MScMS). Introduction to the system.” In: *International Journal of Production Research* 47.14 (2009), 38673889.

[Fra+10] Fiorenzo Franceschini et al. *PRIN - D4.1 Rapporto tecnico relativo alle misurazioni effettuate per “Studio, realizzazione e caratterizzazione metrologica del sistema Mobile Spatial coordinate Measuring System (MScMS), Versione 1.0.* Tech. rep. Politecnico di Torino, 2010.

[Gel+03] Andrew Gelman et al. *Bayesian Data Analysis.* 2nd ed. Boca Raton,FL: Chapman & Hall/CRC, 2003.

[GMP09] M. Galetto, L. Mastrogiacomo, and B. Pralio. “An innovative indoor coordinate measuring system for large-scale metrology based on a distributed IR sensor network.” In: *Proceedings of the ASME 2009 International Manufacturing Science and Engineering Conference MSEC2009.* Ed. by ASME. 2009.

[GMP10a] Maurizio Galetto, Luca Mastrogiacomo, and Barbara Pralio. “MScMS-II: an innovative IR-based indoor coordinate measuring system for large-scale metrology applications.” In: *The INternational Journal of Advanced Manufacturing Technology* 52.1-4 (2010), pp. 291–302.

[GMP10b] Maurizio Galetto, Luca Mastrogiacomo, and Barbara Pralio. “The Mobile Spatial coordinate Measuring System II (MScMS-II): system description and preliminary assessment of the measurement uncertainty.” In: *The International Journal of Metrology and Quality Engineering* 1.2 (2010), pp. 111–119.

[GR92] Andrew Gelman and Donald B. Rubin. “Inference from Iterative Simulation Using Multiple Sequences.” In: *Statistical Science* 7.No. 4 (1992), pp. 457–472.

[Hig+04] Dave Higdon et al. “Combining Field Data and Computer Simulations for Calibration and Prediction.” In: *SIAM Journal of Scientific Computing* 26.No. 2 (2004), pp. 448–466.

[JSW98] Donald R. Jones, Matthias Shonlau, and William J. Welch. “Efficient Global Optimization of Expensive Black-Box Functions.” In: *Journal of Global Optimization* No. 13 (1998), pp. 455–492.

[JW02] Richard A. Johnson and Dean W. Wichern. *Applied Multivariate Statistical Analysis.* Fifth Edition. XXXXXXX: Pearson Education, 2002.

[KO01] M. C. Kennedy and A. O'Hagan. "Bayesian Calibration of Computer Models." In: *Journal of the Royal Statistics Society* xxx.xxx (2001), pp. 425–464.

[LeS99] James LeSage. *Applied Econometrics using Matlab.* 1999. URL: `http://www.spatial-econometrics.com/`.

[Mon01] Douglas C. Montgomery. *Design and Analysis of Experiments.* Fifth Edition. New York, NY: Wiley & Sons, 2001?

[NLW07] Bla Nagy, Jason L. Loeppky, and William J. Welch. *Fast Bayesian Inference for Gaussian Process Models.* Tech. rep. Department of Statistics - The University of British Columbia, 2007.

[Ntz09] Ioannis Ntzoufras. *Bayesian Modeling Using WinBUGS.* Hoboken, NJ: John Wiley & Sons, Inc., 2009.

[Qia+06] Zhiguang Qian et al. "Building Surrogate Models Based on Detailed and Approximate Simulations." In: *ASME Journal of Mechanical Design* Vol. 128 (2006), pp. 668–677.

[QW08] Peter Z.G. Qian and Jeff C.F. Wu. "Bayesian Hierarchical Modeling for Integrating Low-Accuracy and High-Accuracy Experiments." In: *Technometrics* Vol. 50.No. 2 (2008), pp. 192–204.

[Ree+04] C.Shane Reese et al. "Integrated Analysis of Computer and Physical Experiments." In: *Technometrics* 46.No. 2 (2004), pp. 153–164.

[RM07] Havard Rue and Sara Martino. "Approximate Bayesian inference for hierarchical Gaussian Markov random field models." In: *Journal of Statistical Planning and Inference* 137 (2007), 3177 3192.

[RN10] Carl Edward Rasmussen and Hannes Nickisch. "Gaussian Process for Machine Learning (GPML) Toolbox." In: *Journal of Machine Learning Research* 11 (2010), pp. 3011–3015.

[RN11] Carl Edward Rasmussen and Hannes Nickisch. *The GPML Toolbox - Version 3.1.* 2011. URL: `www.GaussianProcess.org/gpml`.

[RS03] A. Ruszczynski and A. Shapiro. *Stochastic Programming. Handbooks in Operations Research and Management Science.* Vol. 10. Amsterdam, NE: Elsevier Science B.V., 2003.

[RW06] C.E. Rasmussen and C.K.I. Williams. *Gaussian Processes for Machine Learning.* Cambridge, MA: the MIT Press, 2006.

[Sac+89] Jerome Sacks et al. "Design and Analysis of Computer Experiments." In: *Statistical Science* Vol. 4.No. 4 (1989), pp. 409–435.

[SHM00] Tzung-Sz Shen, Jianbing Huang, and Chia-Hsiang Menq. "Multiple-sensor integration for rapid and high-precision coordinate metrology." In: *IEEE/ASME Transactions on Mechatronics* 5.2 (2000), pp. 110–121.

[Spi+03] David Spiegelhalter et al. *WinBUGS User Manual - Version 1.4.* 2003. URL: `http://www.mrc-bsu.cam.ac.uk/bugs`.

[SWN03] Thomas J. Santner, Brian J. Williams, and William I. Notz. *The Design and Analysis of Computer Experiments.* New York, NY: Springer, 2003.

[TW87] Martin A. Tanner and Wing Hung Wong. "The Calculation of Posterior Distributions by Data Augmentation." In: *Journal of the American Statistical Association* 82.No. 389 (1987), pp. 528–540.

[US 92] US National Science Council. *Combining Information. Statistical Issues and Opportunities for Research.* Washington, D.C.: National Academy Press, 1992.

[Wec+09] A. Weckenmann et al. "Multisensor data fusion in dimensional metrology." In: *CIRP Annals - Manufacturing Technology* 58.2 (2009), pp. 701–721.

[XDM08] Haifeng Xia, Yu Ding, and Bani K. Mallick. "Bayesian Hierarchical Model for Integrating Multi-Resolution Metrology Data." In: *submitted to Technometrics* (2008).

[XDW07] Haifeng Xia, Yu Ding, and Jyhwen Wang. "Gaussian Process Model for Form Error Assessment Using Coordinate Measurement." In: *submitted to IEEE Transactions* (2007).