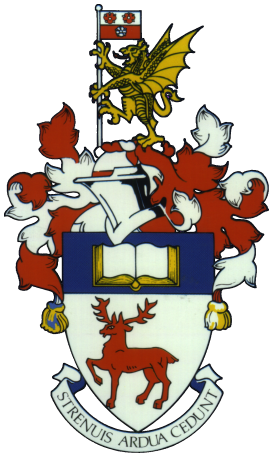
Uncertainty in Numerical Wind-wave Models



Ben Timmermans

School of Ocean and Earth Science
University of Southampton

Thesis for the degree of

Doctor of Philosophy

May 2015

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING, SCIENCE AND MATHEMATICS
SCHOOL OF OCEAN AND EARTH SCIENCE

Doctor of Philosophy

UNCERTAINTY IN NUMERICAL WIND-WAVE MODELS

by Ben W. Timmermans

The modelling of ocean waves is now carried out routinely at meteorological centres around the world. However, little is know about the source of the uncertainty in the predictions of waves produced, and sources can be numerous depending on the specific application. Historically it was felt that the dominant source of uncertainty originated from incomplete knowledge and expression of forcing winds. However more recent studies have focused on the underlying physical processes and their representations, with some authors questioning whether the limitation of the current modelling approach has been reached. Recently, methods for the statistical analysis of complex computer models, including models such as those used for wave prediction, have been developed. In this thesis these methods are applied to perform the first ever uncertainty analysis of a wave model. These new methods are applied to the state of the art wave model Wavewatch III®.

This thesis principally explores the effect of tuning parameter uncertainty relating to the "Tolman and Chalikov" input and dissipation parameterisation, the discrete interaction approximation scheme for nonlinear wave-wave interactions and uncertainty about wind forcing, on wave simulation output, in a range of idealised cases, and realistically on Lake Michigan. The effectiveness of the statistical methods is first demonstrated in simple cases, before analysis is performed for progressively more complex simulations. In each case, uncertainty measures are computed with respect to simulation output in terms of summary wave statistics, typically including significant wave height and peak period. The analysis reveals nonlinear response and the relative importance of the various input, which in turn shows the active physical processes, and where the greatest sources of uncertainty lie. Both uncertainty about wind forcing and the process of nonlinear wave-wave interactions are found to be dominant in all cases, although energy dissipation is important in growing sea states. Finally, observational wave height data is used to perform a parameter calibration for simulations of stormy conditions on Lake Michigan, leading to improved performance.

Contents

List of Figures

List of Tables

Declaration of Authorship

I, **Ben Timmermans**, declare that the thesis entitled **Uncertainty in numerical wind-wave models** and the work presented in the thesis are both my own, and have been generated by me as the result of my own original research. I confirm that:

- this work was done wholly or mainly while in candidature for a research degree at this University;
- where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
- where I have consulted the published work of others, this is always clearly attributed;
- where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
- I have acknowledged all main sources of help;
- where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
- parts of this work have been published as, "Uncertainty in wave power from a wave model" in the Proceedings of the Ninth European Wave and Tidal Energy Conference 2011, Southampton, UK

Signed:.....

Date:.....

Acknowledgements

I would like to convey my sincere thanks to my supervisor Peter Challenor, for his continued support, input and confidence in my capability. I am also very grateful to my co-supervisors, Christine Gommenginger, for her considered input and positive spirit, and Peter Jan van Leeuwen, for his support, feedback and assistance with funding. I am grateful to have had the opportunity to undertake this research, which was funded by the UK National Centre for Earth Observation. Thank you also to a number of others including; Jeff Blundell for computing support at the National Oceanography Centre, Chris Bunney at the UK Met Office for support with Wavewatch III, W. Eric Rogers at Naval Research Laboratoty, Mississippi (US) for providing code and assistance with Lake Michigan and Gregory Lang at the Great Lakes Environmental Research Laboratory (US) for providing forcing winds for simulations of Lake Michigan. Karl Ropkins (University of Leeds) provided valuable help and modification to the "openair" package in R. Thanks also to Richard Humphreys for proof reading, and in particular to Chris Hughes for numerous helpful chats about statistics. A special thanks to my friends who were there during difficult times, and finally, to my courageous Mum for her unfailing support, and great strength in the face of extreme adversity.

In loving memory of my Dad.

1 Introduction and motivation

Wind driven waves on the surface of the ocean, or wind-waves, have been a focus of study for many decades. Prediction of wind-waves is relevant to the undertaking of many ocean-based activities. The pioneering oceanographer, Walter Munk, developed methods to predict sea conditions in order to help the planning of the "D-Day" landings during World War 2. Since then, the theory of ocean waves has been developed into modern day global numerical wind-wave models that can make good predictions of short term wave conditions and future wave climate. Such models have been developed by numerous institutions for a variety of applications, that involve simulations ranging from large lakes, to oceanic settings on coastal, regional and global scales. Applications are diverse - surfers demand accurate swell forecasts, and in the UK the value of the surfing industry to Cornwall is reportedly in excess of £50 million annually. This figure is small compared to the financial risk globally of dangerous sea conditions to shipping and coastal operations, where human life is also a potential cost of poor understanding of waves. Perhaps of greater value still is the global wave energy resource, as society looks to the sea for low carbon and renewable sources of power. Our ability to accurately quantify this resource is very important to this rapidly growing industry. Furthermore, there has been much recent interest in the role of waves in moderating the ocean-atmosphere boundary exchange (Hanley and Belcher, 2009; Huang et al., 2011) and researchers increasingly employ coupled models to better understand the effects of these interactions (Wolf, 2008; Cavaleri et al., 2012; Janssen et al., 2013). The obvious benefits of understanding and improving wind-wave predictions provides compelling motivation for on-going research.

Modern numerical wind-wave models are based upon the principle of specifying waves in a *spectral* form. That is, given it is clearly impossible to resolve waves everywhere, ocean waves, or rather their amplitudes, directions and frequencies throughout the geographic domain, are necessarily regarded in a statistical sense and represented by the variance of the sea surface height distributed over a direction and frequency domain. Spectral wave models do not resolve individual waves and so cannot predict "freak" waves[1] nor do they incorporate necessary physics to investigate Tsunamis.

Imagine then that on the sea surface there are waves of all frequencies travelling in all directions, and that we can represent this approximately by defining a "directional frequency spectrum" at any location that specifies the variance in sea

[1]For a discussion of freak waves and their relationship to spectral wave models, see Janssen (2002) and Babanin and Rogers. (2014)

surface height attributable to waves at each frequency band and travelling in each direction. For the purposes of numerical modelling it is necessary to discretise frequency and direction. The time evolution of this directional spectrum can then be computed according to governing physical laws. Numerical wave models operate by computing the time evolution of the directional frequency spectrum at each grid point throughout a geographic domain. The computation of the spectrum (at each location) is based upon equations governing the propagation of wave energy, and various physical processes, or *source terms*, that govern energy input from the wind, subsequent dissipation and the process of energy transfer between waves. For most practical purposes such as forecasting for shipping, wave conditions are summarised by average heights, periods and directions that are calculated from the frequency spectrum. Modern wave models such as Wavewatch III® (Tolman, 2009a) have been in development for more than 30 years. In that time some important developments have been made (for an overview, see e.g. Janssen, 2008) but as he notes there remains considerable uncertainty in a number of areas and the debate as to the depth of our understanding can be heated. Areas of recent progress, together with other underlying principles are discussed in chapter 2.

While the performance of forecasts and hindcasts produced from modern day wave models is generally good (Tolman, 2002c; Bidlot et al., 2002; Ardhuin et al., 2010) the same studies, and others, often find areas of weakness (Feng et al., 2006; Cavaleri, 2009). This is unsurprising given the complexities of the physical processes, the challenges of numerical representation, the difficulties in obtaining accurate measurements of forcing winds, and the wide range of wave model applications. Significant parts of the numerical computation are based upon empirical parameterisations of the physical processes (e.g. wind input) that have been tuned to match observation in given conditions. Empiricism in complex numerical models is commonplace and the investigation of the physical processes that govern wind-waves presents a significant challenge to modellers and experimentalists alike. From a theoretical position for example, explaining the dissipation of energy due to turbulent breaking waves is as challenging as measuring the dissipation directly. Rogers et al. (2012) describes an example of testing a theoretical formulation and Leckler et al. (2013) and Gemmrich et al. (2013) describe the use of different types of validation data, none of which are easily obtained. Moreover, sources of uncertainty are not limited to the theoretical representation of wave physics. Traditionally, wind speed errors have been regarded as the dominant source of uncertainty in predictions of wave height (Janssen et al., 1997). Studies such as (Feng et al., 2006) show that differences between forcing winds (obtained from different high quality sources) can

result in predicted wave conditions that differ by up to fifty percent (in terms of wave height) in some cases. In conditions that could potentially be dangerous this is clearly very significant. There is also the question of *numerical* uncertainty, that is errors introduced due to the approximate solution of the underlying physical equations. Global wave models are typically run at a longitude / latitude grid resolution of 1 degree, but this is inadequate to resolve smaller islands which can be problematic (Tolman, 2003; de Leon and Soares, 2005). In addition to the discretization of the geographic domain, both the spectral frequency and directional domains are discretized. Anomalies arising from this have also been documented (Tolman, 2002a). Coarse resolution is necessitated by the computational expense of the model, particularly for global operation. Another implication of computational cost, that will be discussed in more detail shortly, is the resulting restriction on the number of model runs that can be performed for the purpose of model analysis and validation. Global or regional models require tens to hundreds of CPU cores for just a few hours of fore- or hindcast simulation. Thus, the potential to assess uncertainty through ensemble runs is limited.

Although a number of studies have explored the issue of uncertainty, or at least sources of error, in one way or another, none has so far performed a rigorous, comprehensive and quantitative assessment. This is not a criticism of the authors given the challenge and possible extent of the task. Rogers (2002), Rogers et al. (2002) and Rogers et al. (2005) have tackled the question of sources of errors in perhaps the most comprehensive manner. Rogers (2002) separates errors in wave modelling into three broad categories: numerics and resolution, physical formulations and forcing. The scope of the study is limited to the modelling of low frequency waves (which is itself a broad area), but each source of error is addressed systematically. Rogers et al. (2005) approach the issue of model structural error (bias) directly, posing and answering qualitative questions about the relative magnitudes of errors.

So it is unsurprising that, to date, quantification of uncertainty due to the various aspects of wave modelling has not been addressed formally in the literature. Yet, there is a need to answer some fundamental questions about the capacity of wave models to produce reliable prediction. The following are examples of the type of scientific questions that have been posed.

1. How influential is the uncertainty in forcing winds in wave model predictions? (Bidlot et al., 2002)
2. What choice of model input parameter values gives the best performance when hindcasting wave conditions? (Ardhuin et al., 2010)
3. Has wave modelling reached its limit? (Liu et al., 2002).

None of these authors had access to a rigorous methodology to explore the origin of uncertainties in a wave model. Notably, in the context of wave model performance in more extreme conditions, Cavaleri (2009) emphasises the need for a further reaching investigation of wave model physics, and advocates an extensive sensitivity analysis in order to better focus development work. He goes further to suggest that certain statistical methods (which will be introduced shortly), could possibly be employed, though indicates some skepticism that they may be effective given the complexities. Of course, the complexity and lack of suitable methods is a primary reason for the absence (to date) of such analysis. Owing to this absence, this thesis brings to bear rigorous statistical methods on modern wave models as advocated by Cavaleri (2009). The objective of this study, therefore, is for the first time, *to establish and apply rigorous methods of quantitative uncertainty analysis to wave models, and obtain uncertainty measures.* Questions such as those posed above are addressed formally, which thus far has not been possible.

1.1 Tackling uncertainty in complex models

Numerical models are abundant in the all fields of scientific research. It is often desirable to build a model in order to conduct *computer experiments* that would otherwise be impossible in the real world system, and answer "*What if...*" type questions. Here, a complex model is defined as one that expresses our detailed understanding of a nonlinear real world process in terms of a mathematical formulation, that typically comprises numerous dimensions of input and output data, physical and empirical quantities or parameters, control features, and so on. Having constructed a model we are usually uncertain about various aspects of both the real world process and our representation, be they specific values of physical constants, the "best" values for empirical parameters, the choice of parameterisation, etc. The model is also likely to be computationally expensive, thus limiting the number of times we can run the model and explore the effect of uncertainty about its operation.

The increasing availability of high performance computing platforms and the growth of computational capacity has permitted the implementation of more and more sophisticated models at higher and higher resolutions[2]. This has allowed us to investigate in unprecedented detail and certainly appears a sensible approach, supported by leading scientists (e.g. Palmer, 2011). There is however debate as to whether our lust for detail and increased complexity has overtaken our ability to

[2]Indeed, it often seems unclear whether it is the rapid advance in our understanding and formulation of scientific problems that drives computational demand, or whether, knowing that more computational power will be available in the future, we relentlessly plan to utilise it with ever more complex models!

obtain improved scientific insight. Anderson (2010) and Rougier and Cruifix (2012) caution against the increase in sophistication for a number of reasons. Notably, the difficulty in fully comprehending any complex model and the implications of its predictions, given the wealth of uncertainty we have about its constituent processes, their mathematical representation, and the lack of observational data we can bring to bear in validating the output. Particularly in high profile examples, others are calling for uncertainties to be addressed more rigorously and consistently (Katz et al. 2013). It may even be the case that output from models of large and complex natural systems will always be underdetermined by possible observations and they therefore cannot ever be validated with confidence, as asserted by Oreskes et al. (1994).

A recent example that elucidates these points (though not related specifically to wave modelling) is that of the difference between successive revisions of earth system models developed by the UK Met Office Hadley Centre[3]. Under a future climate scenario, the "HADCM3LC" model predicted an extensive "dieback" of the Amazon rainforest, whereas its successor, the "HADGEM2-ES" model predicted very little change to the forest under the same climatic forcing. This stark contrast is investigated by Good et al. (2013). The authors attribute the difference primarily to a change in dry season length (which is itself due to other factors) but acknowledge a lack of understanding about the physical processes leading to this. Moreover, with regard to the question of what the conflicting models tell us about the overall risk of Amazon dieback, the authors state that the risk was, and remains, hard to quantify. With respect to the range of improvements made in the HADGEM2-ES model, the authors further acknowledge that *"... it is not currently possible to say how significant these improvements might be for the specific case of Amazon forest projections"*. Although only a preliminary study, Good et al. (2013) do not employ any formal methods of uncertainty or sensitivity analysis and it is perhaps telling of the complexity and uncertainty involved that this particular (and surprising?) finding, presumably one of many possible differences in behaviour between the two models, warrants extensive and onerous follow-up work. For models with potential and far reaching implications for global policy makers, *a priori* one might expect somewhat more certainty, or at least a more robust quantification of it.

Wave models are demonstrably better understood than earth-system models - indeed, they are often a sub-component of such models - and there is ample opportunity to validate their performance. However, they exhibit their own shortcomings. For example, Cavaleri (2009) draws attention to the on-going problem of simulating waves in more extreme conditions. He cites some examples of where wave models

[3]http://www.metoffice.gov.uk/climate-guide/science/science-behind-climate-change/hadley

fail to capture "peak" events and explores why this remains the case, in spite of improved modern measurements of wind (that were traditionally held as responsible for most error in prediction). He suggests that, given good wind data, if substantive improvements are to be made, then sensitivity analysis is required.

All but the most parsimonious of numerical models require considerable interrogation in order to be understood and improved upon, so a complex model presents a number of problems with respect to uncertainty analysis. A number of ingredients are required to proceed. Before these are expounded upon it is necessary to clarify what is meant by *uncertainty*. Kennedy and O'Hagan (2001) designate terms for specific aspects of uncertainty with respect to complex numerical models. This is discussed in more detail in chapter 3 but for now three terms are introduced. *Parameter uncertainty* refers to our uncertainty about the "true" value(s) for model input parameters. *Model inadequacy* refers to the model's fundamental inability to make a perfect prediction, even given perfect[4] input. Model inadequacy is also commonly referred to as structural uncertainty, and often associated with systematic model error. Thirdly, *observation error* refers to uncertainty about observations of the real world system in question that are used for validation. Without accounting for this source of uncertainty, we are clearly limited in what we can determine about how well the model represents the real world. In this thesis, chapters 4, 5, 6 focus on parameter uncertainty, with respect to wave model input and dissipation, and nonlinear wave-wave interaction parameterisations. Uncertainty and sensitivity analysis is used to evaluate wave model response to uncertain input parameters, and obtain probability distributions for model output. Observation error is discussed in chapters 5 and 7. Section 1.2 describes the approach taken in this thesis to addressing these sources of uncertainty in wave models.

In terms of performing quantitative analysis of these sources of uncertainty, we require a range of tools with which to understand and evaluate the effects of the uncertainty (which are discussed in chapter 3). The terms "uncertainty analysis", "sensitivity analysis" and "uncertainty quantification" are ubiquitous in this context. They may occasionally be preceded by the word "global" to imply an analysis over the whole domain of uncertainty (in a given context). Saltelli et al. (2000) discusses these methods at length and in a variety of contexts, which are reviewed in more detail in chapter 3. The details of sensitivity analysis for simulators differs from sensitivity analysis in software engineering, for example, but the general principle is the exploration of how information is processed through a model. As mentioned, complex models require expensive computational resource which there-

[4]Kennedy and O'Hagan (2001) discuss what is meant by "perfect" input.

fore necessitates efficient methods of uncertainty analysis. Methods described by Saltelli et al. (2000) generally require the evaluation of integrals for which no closed form solutions exist (due to the complexity of the model). Numerical solutions are computationally demanding and thus some sophistication is required to overcome this difficulty. One possible approach is the use of "surrogate" models. A surrogate model is an accurate representation of the complex model, formulated in some way based upon observations of output from the complex model, usually referred to as "training data". The principle is that of regression. A simple linear regression model attempts to explain the relationship between some observed quantity (or dependent variable) and an independent variable, given the assumption that there is a linear relationship between the two. If the regression model is good, then it is essentially capable of *predicting* the dependent variable for any given value of the independent variable, with an estimate of uncertainty about the prediction. Imagine, then, that the observations are output from a complex model and the independent variable is some input to the complex model. The relationship between the two is unlikely to be so trivial that a simple linear regression suffices, although this has been used to good effect (see e.g. Sexton et al., 2003) but it may be a reasonable approximation, or a more sophisticated regression model could be formulated to do a better job. If a suitable regression model can be found, then in principle it can predict the output of the complex model at any given value(s) of the independent variable(s). Moreover, if the regression model is efficient it can be fitted using few observations of the complex model, and once fitted, it runs very quickly, much faster than the complex model. Having obtained a fast surrogate of the complex model that is a statistically accurate and robust representation, computation of various uncertainty measures, using the methods of Saltelli et al. (2000) for example, is now possible. It turns out that certain classes of regression models can in fact perform very well in this role. Haylock and O'Hagan (1996) outlines the use of this method as applied to radiological modelling. They cover various aspects of the approach. The regression model described is commonly referred to as a statistical *emulator*. The background, theory and motivation to use statistical emulators is discussed in detail in chapter 3.

Having acquired the appropriate tools, the second aspect of the problem of uncertainty analysis is that our input uncertainty must be specified. In some cases this can be relatively straight forward. For example, if the input to a complex model is the value of some physical quantity that is measured, then the measurement will have some uncertainty associated with the measuring equipment or method. However, many physical quantities are difficult, or impossible, to measure directly. In

these cases a judgement has to be made which typically requires the scientist face a rather unpleasant issue: an honest appraisal of their knowledge. Furthermore, the representation of the uncertainty estimate is not always simple, and the expert may not be familiar with how to characterise this. The process of uncertainty *elicitation* has many aspects to it, and may involve the interrogation of one or more experts by a trained statistician, with the objective of obtaining uncertainty information in terms of probability and probability distributions. O'Hagan et al. (2006) discusses expert elicitation in depth and this will be considered further in chapters 3 and 4.

With the above two ingredients - tools to do the job, and an accurate estimate of our uncertainty - uncertainty analysis can be performed. This may yield, for example, an estimate of total uncertainty about a specific prediction, or could yield a measurement of the numerical model's sensitivity to one or more inputs, given uncertainty.

1.2 Approach and thesis layout

Noting the objective, the approach in this thesis is to analyse quantitatively the effects of wave model input uncertainty on model output. Specifically this is with respect to the tuning parameters of two important physical parameterisations, and (parametric representation of) wind input. This yields measures of output uncertainty, typically in terms of wave summary statistics, which reveals the strength of influence of specific input, any nonlinear response and highlights more generally (and quantitatively) which physical processes are dominant. In turn, this information can be used to formulate a strategy that guides further research effort to maximum effect. Importantly, the proposed analysis captures the effects of uncertainty on simulations as a whole, including all interactions, which may reveal different results than if individual sources were considered in isolation.

Analysis is performed for a range of physical situations, beginning with simple cases. The approach is then to build understanding by increasing simulation complexity incrementally, thus obviating the effect of additional sources of uncertainty at each stage. Research therefore begins by considering simulations that make use of a single grid cell in chapter 4 and progresses to a realistic case in chapter 7. The layout of the thesis is therefore as follows.

Chapter 2 discusses relevant wave modelling history and theory, the current state-of-the-art and areas of uncertainty. Details of the parameterisations investigated in this thesis are given, together with a description of relevant physical situations. Chapter 3 introduces methods of uncertainty and sensitivity analysis that will be applied in this thesis. This includes background to the principles of

computer experiment design, building statistical emulators and how these can be used to perform various statistical analysis for computationally expensive numerical models. Analysis begins in chapter 4, by considering the simple case of a single grid cell which permits the investigation of duration-limited wave growth. The effects of uncertainty about input parameters associated with important physical parameterisations are investigated with respect to simulated wave summary statistics. Relevant uncertainty information is elicited from the literature and used as input for the analysis. Preliminary findings are discussed, which is followed by an investigation of how uncertainty about the wave energy dissipation parameterisation would lead to a probabilistic estimate of wave power extraction from a wave energy converter (WEC).

In chapter 5 complexity is increased by allowing energy propagation (advection) in one dimension. This permits the simulation and investigation of uncertain input on fetch-limited wave growth. The same sources of input uncertainty are investigated, with respect to both wave summary statistics at specific points along the fetch, and a parametric representation of the growth profile. Probabilistic sensitivity analysis is also introduced, so that the effects of specific input can be determined. This reveals how and where physical processes (or rather their parameterisations) are active.

In chapter 6 complexity is again increased by allowing energy propagation in both spatial dimensions. This permits the simulation and investigation of directional waves. It also allows for the expression of more complex forcing winds, and some aspects of what is meant by "wind uncertainty" are discussed. Making use of the same input parameters, designed experiments therefore analyse the effects of uncertainty on wave direction and directional spreading in the case of a "turning wind". In 2-D, this idea is extended by implementing a parametric representation of the wind, in order to assess the effect of uncertainty about specific aspects of the wind dynamics (in an epistemic sense). Probabilistic output for wave direction is obtained, and sensitivity analysis is performed to identify the effects of specific input. This reveals the relative importance of the accuracy of wind input compared to uncertainty about the other physical parameterisations.

The experimental work in this thesis concludes with chapter 7, where realistic hindcasts of waves on Lake Michigan are investigated. Designed experiments are performed, conditional on historic winds, and the effects of uncertainty about input parameters in different cases, are identified. The results, in terms of parameter sensitivity, and how these identify active physical processes, are compared to results found in the idealised cases considered in previous chapters. In the final experi-

ment, poor performance of the wave model is identified in stormy conditions, and observational data from data buoys is formally incorporate to perform a parameter calibration for the wave model. This leads to an improved parameterisation in such conditions.

Conclusions and possible areas of further study are given in chapters 8 and 9 respectively.

2 Introduction to wind-wave modelling

This chapter presents a background to wind-wave modelling and identifies key aspects of uncertainty in terms of model formulation. In this context, these key aspects are investigated specifically in later chapters. Section 2.1 broadly describes the history and evolution of the scientific theory underlying spectral wave models. This includes a description of the state-of-the-art model Wavewatch III® that is studied throughout this thesis. Section 2.2 explores uncertainty in wave models by considering components of a modern wave model in more detail. It also sets out more specifically terminology associated with uncertainty and how types of uncertainty manifest. Section 2.3 summarises literature that describes previous work on uncertainty in wave modelling. The chapter concludes with section 2.4 which describes important physical situations that are investigated in this thesis.

2.1 Spectral wave models

2.1.1 Background

The modelling of ocean waves is generally regarded as beginning with Sverdrup and Munk (1947). This work arose from the need to predict wave conditions in order to make the best preparations for beach landings in Africa and Europe during World War 2. In the intervening years the theory of the generation and propagation of wind waves, and the field of wave modelling, has grown enormously. The literature is extensive, and a full review is not attempted here. For a comprehensive review of the physical principles the reader is directed to Komen et al. (1994) and Holthuijsen (2007). Cavaleri et al. (2007) reviews active research areas, and shorter reviews with some historic perspective are given by Mitsuyasu (2002) and Janssen (2008). Tolman (2013a) also summarised knowledge gaps and recent areas of progress. This thesis is concerned with how uncertainty about input affects a modern wave model and so important relevant aspects of the theory are summarised.

For the purpose of modelling (and by necessity), ocean waves are represented statistically. Sverdrup and Munk (1947) introduced the idea of "significant waves" which arose from both practical and statistical considerations. Subsequently, the study of the statistics of random processes, based on original work by Rice (1944), contributed significantly to the development of the statistical theory of waves. Longuet-Higgins (1952) and Pierson (1952) for example, made important contributions to the analysis of waves in terms of their statistical representation, the latter providing an in-depth treatment of the properties of the statistics of waves arising in realistic conditions. It turns out that to a good approximation, the distribution of the sea

surface height is Gaussian. This implies that the sea surface can be represented as a linear superposition of waves of random phase, hence why spectral wave models are often referred to as "phase-averaged". Given stationarity in time and homogeneity in space, a Fourier transform of the auto-covariance function with respect to these temporal and spatial variables yields the spectral "variance density spectrum". This quantity summarises the variance of sea surface height associated with a particular frequency of wave. With some further manipulation, the variance density spectrum can be presented as the "directional frequency spectrum", $E(f, \theta)$, which is a continuous function of frequency f and direction θ. In a sense this is the "currency" of a modern wave model and a very important concept. For a more thorough discussion and derivation see Holthuijsen (2007) (section 3.5). Section 2.1.2 describes how a wave model evolves the directional frequency spectrum in time through an energy balance equation.

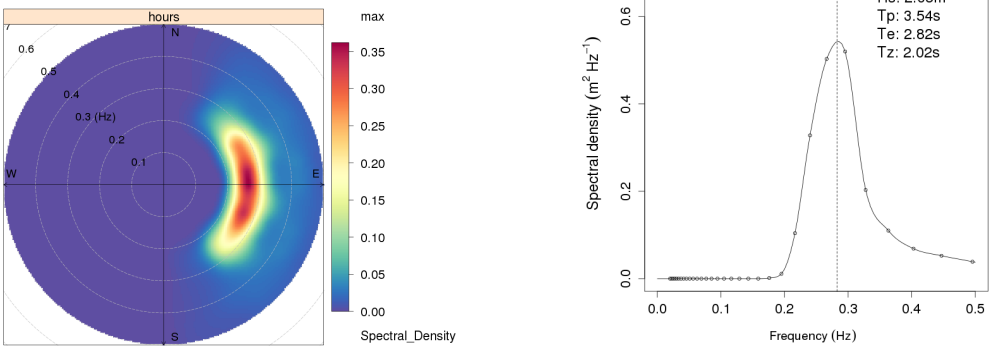
Knowledge of the directional frequency spectrum is of little practical use. It cannot be easily measured, nor is it helpful to mariners since it is not readily interpretable. For practical purposes however, various summary statistics can be obtained from the spectrum. In the real world, for safety or engineering purposes for example, a measure of wave height or period is typically required. These are obtained from the moments of the spectrum. For a spectrum, $E(f, \theta)$, the nth spectral moment m_n is given by,

$$m_n = \int_0^\infty \int_{-\pi}^{\pi} f^n E(f, \theta) \, \mathrm{d}\theta \, \mathrm{d}f \tag{2.1}$$

Of particular importance is the significant wave height, H_s,

$$H_s \approx 4\sqrt{m_0} \tag{2.2}$$

H_s is defined as the average height of the largest 1/3 of waves. A full discussion of this important quantity is provided in Holthuijsen (2007) (section 4.2.2) but summarised briefly here. Given the Gaussian distribution of sea surface height, individual frequency bands, Δf, turn out to be Rayleigh distributed, and characterised by m_0, the first moment of the frequency spectrum. If the sea state is "narrow banded", that is most of the variance (or energy) lies in a narrow frequency band, then the probability of wave height H can be reasonably approximated by a Rayleigh distribution. Given this distribution, using the wave crest height H^* that satisfies $\int_{H^*}^{\infty} p(H) \mathrm{d}H = 1/3$, the resulting calculation for the expectation of the height of the



(a) An example of a directional frequency spectrum formed from a steady unidirectional wind.

(b) Direction integrated "1-D" spectrum obtained from (a). Note statistic T_z corresponds to zero-crossing period.

Figure 2.1: Example output spectra from a simulation of a developed sea in deep water.

highest third of waves is,

$$H_s = H_{m_0} = E\{H : H \geq H^*\} = \frac{\int_{H^*}^{\infty} Hp(H)\mathrm{d}H}{\int_{H^*}^{\infty} p(H)\mathrm{d}H} = 4.004\ldots\sqrt{m_0} \tag{2.3}$$

because the Rayleigh distribution happens to be characterised by the first moment of the frequency spectrum. Similar results can be obtained for other quantities of interest. The energy period, T_e, and zero-crossing period, T_z, are given by,

$$T_e = m_{-1}/m_0 \tag{2.4}$$

$$T_z = \sqrt{m_2/m_4} \tag{2.5}$$

Peak wave period, T_p, is the value of the modal period (or the reciprocal of the modal frequency). Other often used averages are the mean period $T_{m_{01}} = 1/f_{mean}$ and the zero crossing period, T_z. Both Holthuijsen (2007) (chapter 4) and Tucker and Pitt (2001) provide further details.

To get a better feel for these concepts and quantities, figure 2.1 (a) shows a visualisation of a spectrum, output from a simulation of a steady (unidirectional) wind over deep water. Note that although there is a clear mode (f_p), there is also a good degree of directional spreading. The direction integrated spectrum, or "1-D spectrum" seen in panel (b), is more easily interpretable and can be measured directly by data buoys on the sea surface. The 1-D spectrum can be described by summary statistics (H_s, f_p, etc.) as shown in the top of the figure. Owing to the

complexity of measuring and utilising the directional frequency spectrum, modern wave models are routinely validated with respect to observations of H_s, T_p and T_e. *Mean direction* and *directional spread* are also important summary statistics and are discussed further in section 2.4.3. The bulk of the analysis in this thesis is with respect to these summary statistics, which are used extensively.

Given that the form of the 1-D spectrum, seen in figure 2.1, is typical of a wide range of unidirectional sea states, it seems apparent that the spectral shape could be represented more completely, rather than having to reduce it to summary wave statistics (and thus lose information). This is explored further in appendix A.

2.1.2 Wave modelling and the energy balance equation

With the development of directional frequency spectrum and the advent of computers, wave models for the purpose of forecasting were possible. Early numerical models arrived in the 1950's and 60's but their performance was limited because they were forced to impose simple parametric forms on the frequency spectrum. To obtain realistic simulations, the directional frequency spectrum must be evolved in a more sophisticated way. The problem can be formulated as an energy balance equation,

$$\frac{dE(f,\theta;x,y,t)}{dt} = S(f,\theta;x,y,t) \tag{2.6}$$

where $S(f,\theta;x,y,t)$ is a source term that accounts for energy loss and gain. Exactly how this problem is solved depends on whether a Lagrangian or Eulerian formulation is adopted. The former, which is not discussed here, could be solved by integrating the source term along all wave paths leading to the point in question. However, the source term is a function of location and therefore must be computed everywhere[5]. For wind-wave modelling this is not a practical approach. An Eulerian approach dictates that either a Cartesian or longitude-latitude grid be specified, and then the energy balance for each grid cell be computed. The energy balance in a single grid cell for each time step is then the sum of energy propagated into the cell and energy transferred due to the source function, minus the sum of the energy propagated out of the cell and any losses associated with the source function (e.g. dissipation by white-capping). See Holthuijsen (2007) section 6.4 for further discussion.

The source function $S(f,\theta;x,y,t)$ is of particular importance in this thesis and so is described further. The work of Pierson and Moskowitz (1964) and Hasselmann et al. (1973), among others, were significant experimental campaigns designed to

[5]The "ray-tracing" approach is very effective, for example, for light rays where there are a small number of sources. It has also been implemented for modelling swell propagation, see Hsu et al. (2004).

evaluate the source function and determine how it relates to evolving sea states. They were able to specify the source function in terms of the frequency spectrum. In particular the forms of the frequency spectrum that evolve under certain conditions (discussed shortly) have become ubiquitous in wave modelling - the "Pierson-Moskovitz" spectrum (usually abbreviated to PM) and the "JONSWAP" spectrum (after the name of the famous study[6]). In order to better understand the source function let us rewrite (2.6) based upon an Eulerian approach, which gives the energy balance equation in deep water,

$$\frac{\partial E(f,\theta)}{\partial t} + \mathbf{c_g} \cdot \nabla_{\mathbf{x},\mathbf{y}} E(f,\theta) = S_{\text{wind}} + S_{\text{nl}} + S_{\text{ds}} \tag{2.7}$$

where $\mathbf{c_g}$ is the group velocity of a particular spectral band, and the spatial derivative is with respect to the $\mathbf{x}$ and $\mathbf{y}$ coordinates. A formulation can be easily obtained for a longitue-latitude coordinate system. The source function has now been separated into terms which are respectively *wind input*, *nonlinear wave-wave interactions* (that transfer energy between spectral components), and *energy dissipation*. These three terms describe the case for deep water, where energy dissipation from "white-capping", that is the breaking of waves, is the only significant dissipative process. However, in specific situations additional source terms associated with other physical processes, such as shoaling and bottom friction, may be included. It is important to note that our understanding of these terms is neither complete nor consistent, and these are significant sources of uncertainty in a modern wave model. These terms are discussed below, and further in the context of the uncertainty analysis proposed in this thesis in section 2.2.2. Before doing so, consider the growth of the spectrum shown in figure 2.2. There is no advection so these spectra are the time integrated source function, shown at intervals of 4 hours after starting from calm conditions. It is clear from the earliest times that wind energy is initially imparted to waves at high frequency (approximately 0.4Hz). As energy builds and the waves grow, energy is input at lower frequencies. It is not clear from these figures how energy dissipation is acting (it is often regarded as quasi-linear, see e.g. Komen et al., 1984), however, as time evolves we can clearly see energy "piling up" at the low frequency end of the spectrum. The spectrum becomes more and more narrow banded, corresponding to more monochromatic characteristics. The cause of this is the nonlinear wave-wave interactions, S_{nl}, that transfer energy from the intermediate region of the spectrum to the higher and lower frequency regions. This process is discussed shortly.

As the spectrum continues to develop and wave energy is moved to lower and

[6]The JOint North Sea Wave Project, Hasselmann et al. (1973).

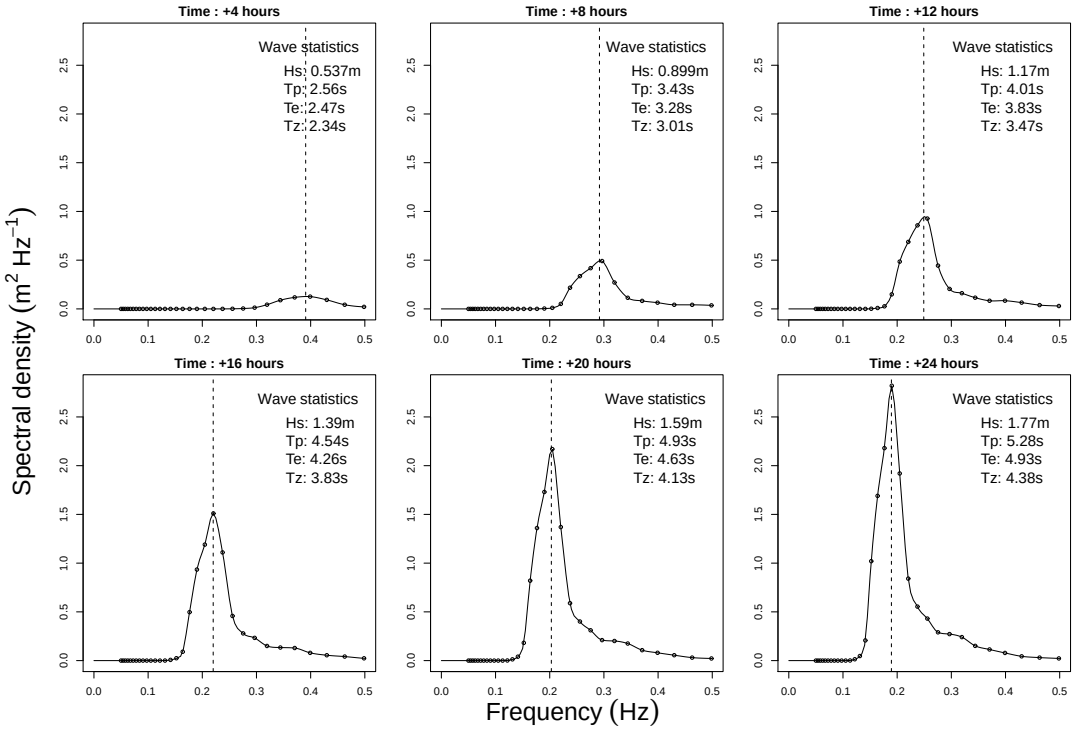


Figure 2.2: Evolution of the sea state in a homogeneous infinite sea.

lower frequencies, very low frequency (or long wavelength) waves are formed. These eventually become what we know as ocean swell. They evolve from storms at sea and may propagate very long distances around the globe showing very little dissipation (see e.g. Ardhuin et al., 2009, Young et al., 2013). Note that deep water waves have a nonlinear dispersion relationship and so are "dispersive". Physically this means that waves of low frequency travel faster than those of high frequency. There is a tendency therefore for wave packets of mixed frequency to disperse over time (and distance). The mechanisms behind the stability of ocean swell remain an open question (see e.g. Henderson and Segur, 2013).

2.1.3 Input and dissipation source terms

Returning to the specific source terms, S_{wind} and S_{ds} are discussed here. There is considerable literature associated with each source term alone and so only a summary is provided. Cavaleri et al. (2007) discuss the state-of-the-art as was, in respect of these source terms, although this has not changed hugely since. S_{wind} and S_{ds} are typically formulated in concert to better cope with maintaining an energy balance.

Wind input is believed to be fairly well understood and the formulation of S_{wind} in wave models has changed little in the past two decades. Measurements and anal-

ysis of wind input date back nearly 100 years although notable early work on the generation of waves is that of Phillips (1957) and Miles (1957). Phillips' work essentially explained the early linear growth of waves resulting from resonant turbulent fluctuations. Miles' theory, variations of which are often employed in modern wave models (Komen et al., 1994), explains the subsequent exponential growth of waves due to a shear wind blowing over a wavy surface. Mechanistically, wind blowing across a wave creates a positive pressure difference over the water moving up at the rear of the crest, thus "pulling" it up further, and a negative pressure difference at the front of the crest, thus "pushing" it down. This leads to a positive feedback mechanism that causes exponential wave growth. In order to characterise the Miles component of the wind input Snyder et al. (1981) took detailed measurements of atmospheric pressure fluctuations which they used to determine an empirical form for S_{wind}. However, this was found to underestimate input at higher frequencies. In a wave model the exponential input regime is typically expressed in the form $S_{\text{wind,exp}} = \beta(u^*)E(f,\theta)$, where β is a function of the square of wind speed u^*. Janssen (1991) modified Miles' theory to include the effects of turbulent airflow and argued that owing to the complex interaction of wind and waves that a coupled system of both wave and atmospheric models was required in order to simulate the true behaviour. Others have taken a slightly different approach. Tolman and Chalikov (1996) proposed a different formulation based on a fairly intricate specification of β as a function of frequency. They also incorporate the effect of wind blowing against (or at a large angle to) the direction of the waves, thus resulting in a loss of wave energy. An important parameterisation developed and implemented by Ardhuin et al. (2010)[7] also utilises the input scheme of Miles as modified by Janssen (1991). However, Rogers et al. (2012) highlights that it does not include some behaviour of the interaction between wind and waves seen in more energetic conditions. They present an input scheme, based upon earlier work by Tsagareli et al. (2009) and Babanin et al. (2010), that accounts for the separation of air flow at high wind speed, and the resulting nonlinearity of energy input. It is clear that much work remains to be done in this area alone, and it is interesting to note that the extensive work of Ardhuin et al. (2010) took many years to go from published research to operational implementation. Janssen (2008) notes that wind input at high wind speeds (and turbulent ocean) is very hard to express theoretically and investigate experimentally, and as such it remains an active area of research.

Energy dissipation due to white-capping, represented by S_{ds}, is regarded as the

[7] As of April 2014 this has become the default input and dissipation parameterisation in the release version (4.18) of Wavewatch III.

least well understood aspect of wave modelling (Cavaleri et al., 2007; Babanin and van der Westhuysen, 2007; Janssen, 2008). While there has been some focus on wind input over recent years, as just described, many of the resulting parameterisations incorporate a revised energy dissipation scheme, which has been the main focus of research. Cavaleri et al. (2007) pose a number of questions that remain to be answered in relation to this problem such as; *How much energy is lost due to white-capping and where in the spectrum? What causes waves to break and what causes them to stop breaking? What does the breaking severity depends on?* The breaking of waves is a highly nonlinear process and extremely difficult to explain theoretically, or investigate experimentally. The lack of rigorous physical theory has led to largely empirical parameterisations which lack detailed physical interpretation. The early work of Hasselmann (1974) made some progress by regarding the breaking of waves as a statistical process. He represents breaking as random pulses and goes on to derive a source function on a physical basis. Tolman and Chalikov (1996) proposed a scheme that separated the breaking of low and high frequency waves. Recent work has focused on a more physical representation of wave breaking. Observations of wave breaking probabilities such as Banner et al. (2000) and Babanin et al. (2001) have led to the idea of a threshold, characterised by conditions such as wave steepness, below which wave breaking does not occur. This has opened up the possibility of parameterisations that explicitly provide estimates of breaking which can be directly tested from observations (see e.g. Leckler et al., 2013). The recent dissipation schemes of Babanin et al. (2010), Ardhuin et al. (2010) and Rogers et al. (2012) are based much more strongly on these physical processes. Other examples of input and dissipation schemes that have been proposed over recent years are van der Westhuysen et al. (2007), Bidlot et al. (2007) and Filipot et al. (2010). Wave energy dissipation remains an active area of research.

2.1.4 Nonlinear wave-wave interactions source term

Of the three source terms described above, nonlinear wave-wave interactions, represented by S_{nl}, are in fact the best understood. Through a perturbation analysis of the free equations of motion for the ocean surface Hasselmann (1962a,b) established rigorously how energy is transferred between waves. It is somewhat counter intuitive, because one might imagine that energy "diffuses". That is, in the case of a spectrum with a well defined peak (such as a PM or JONSWAP), the finer structure of the wave spectrum would degrade with time. However, as Webb (1978) notes, order near the spectral peak is increased at the cost of increased disorder at higher frequencies. The process of transfer can be thought of as taking place between two

pairs of waves. Each pair of waves creates an interference pattern, which in turn are able to interact causing a scattering effect. As the second pair interacts with the first, energy is passed between the two waves in the second pair, depending on their particular wave numbers[8]. In general this means that the spectral peak is amplified, and narrows over time, while additional energy (from the wind) in the intermediate region of the spectrum is also passed to higher frequencies. Webb (1978) generates the S_{nl} source term arising from a PM type spectrum (see his figure 1). It is highly nonlinear in both frequency and direction, and so S_{nl} is important to both frequency and directional development of the sea state.

The chief issue in terms of implementation of this source term is that the exact solution is computationally prohibitive for any realistic simulation of waves. An exact computation method is typically available in modern wave models (the Webb, Resio, Tracy, or WRT scheme is near exact, based upon the work of Webb, 1978, Tracy and Resio, 1982 and Resio and Perrie, 1992), but it is up to 10,000 times slower than all other parts of the wave computation. Unsurprisingly cheaper approximations have been sought, which includes the "Discrete Interaction Approximation" (DIA) scheme, described by Hasselmann and Hasselmann (1985) and Hasselmann et al. (1985). The DIA scheme is particularly important as it has been the backbone of modern wave models for decades allowing for the efficient computation of S_{nl} to good accuracy. As Tolman (2013b) notes, virtually every wave model still uses the DIA scheme, and new parameterisations for input and dissipation are still tested in conjunction with it (e.g. Ardhuin et al., 2010). The basic principle of its operation is that, for a given spectrum, rather than compute the interaction of all wave components in all 4-wave combinations that satisfy the resonance, it considers only a single representative (or *discrete*) interaction. In spite of its great effectiveness it has well known shortcomings, some identified during the original formulation by Hasselmann et al. (1985). In particular it gives rise to excessive directional spreading in energetic seas and thus is believed to contribute to the underestimation of waves in stormy conditions. Various researchers have therefore attempted improvements. Tamura et al. (2010) for example perform testing of an alternative parameterisation that exhibits improvements over the DIA scheme. Perrie et al. (2010) propose a formulation for S_{nl} based on an approach different to the DIA. Tolman et al. (2005) suggests the use of a replacement "surrogate" parameterisation based upon an artificial neural network (this is briefly discussed further in section 9.2). More recently other efforts have proved fruitful. Tolman (2013a) and Tolman and Grumbine (2013) describe the Generalised Multiple DIA which is an extension to the DIA that addresses the

[8]Wavenumber k and wavelength λ are related as $k = 2\pi/\lambda$.

previous shortcomings. While maintaining computational efficiency he makes use of two additional representative interactions (hence *multiple* DIA). Excellent performance is demonstrated in a number of test cases, including in stormy conditions, by comparison with the exact solution.

Nonetheless, this source term is still subject to debate and its importance is continually emphasised. Zakharov and Badulin (2011) for example argue that S_{nl} plays a leading role in the evolution of the wave spectrum, being an order of magnitude larger than S_{wind} and S_{ds}, and as such it should be regarded with more importance than it is currently receiving (particularly noting the recent focus on energy dissipation).

2.1.5 Wavewatch III®: A third generation global wave model

Early wave models, referred to as first and second generation, typically imposed spectral forms parametrically and so were limited in their capability to produce realistic sea states. Examples of such models are described in SWAMP (1985). Third generation models were marked by the way they allowed the sea state to develop realistically through the computation of nonlinear wave-wave interactions. Examples of third generation models include Wavewatch III Tolman, 2002c,b, 2009a, the WAM model (WAMDI, 1988; Komen et al., 1994) and the SWAN[9] model (Booij et al., 1999). Wavewatch III (herein referred to as WW3), developed by NCEP[10], is a third generation global wave model utilised by numerous meteorological agencies worldwide, including the UK Met Office. It is parallelised and typically requires a HPC platform to run global simulations. In practice, the WW3 software comprises a number of auxiliary programs that perform various functions such as pre-processing input (e.g. wind) fields, configuring the spatial grid and post-processing output. Once appropriate input has been specified for each (in a set of input files), the programs are run sequentially.

The software comes as source code, and thus an important part of the compiling process is that the user must specify which model physics is used. This is done via a *switch* file which, as the name suggests, contains a list of abbreviations that correspond to switches that turn various model features on and off. Different parameterisations are available for energy input and dissipation, nonlinear wave-wave interactions, propagation scheme, bottom friction, and so on. By default, S_{ds} is according to Tolman and Chalikov (1996) (henceforth referred to as TC96), S_{nl} is computed via the DIA scheme, and a third order finite differencing numerical

[9] Simulating WAves Nearshore.

[10] US National Centers for Environmental Prediction, http://www.ncep.noaa.gov/

scheme is employed to integrate the action balance equation. An important feature of the code is that typically a "namelist" is provided for each of the various physical parameterisations and numerical schemes (see Tolman, 2009a, section 4.4.2). These essentially allow user-defined input to be specified[11] which, of importance to this research, allows input parameter values to be specified conveniently.

Given compilation following choices of switches and appropriate input in terms of wind forcing and bathymetry, the model can be executed. Output can be obtained for specific points or fields over a spatial domain. For point output, requests can be made for the full 2-D frequency spectra, a direction integrated 1-D spectra or simply a table of wave summary statistics (including H_s, f_p etc). Note that in this thesis, most experiments make use of summary statistics for point output, although this is supplemented by full 2-D spectra where details are examined more closely. Given that in this thesis the spectral and directional resolution is often 36×36, very large amounts of data can be generated. Computation time is approximately 1-2 minutes per 4 days of simulation time, run with the aforementioned spectral resolution on a 100×100 spatial grid using 64 CPU cores. Single grid cell models can however be run very rapidly on a desktop machine.

See appendix B for a description of how to run designed experiments with WW3, or Tolman (2009a) for further details of the model.

2.2 Wave model components and uncertainty

In this section wave model source terms and numerical schemes are considered in more detail in the context of learning about the effects of uncertainty and conducting analysis. Before commencing the discussion, there is an important point to bear in mind. Although the following sections describe approximations associated with individual source terms and aspects of the numerical wave model, it is not clear *a priori* how these sources of uncertainty combine and manifest when running actual simulations of waves. If the model is nonlinear, this might be quite unpredictable. The situation is like the problem of measuring a single source term (e.g. dissipation) on the open ocean. It is extremely difficult because other processes (e.g. nonlinear interactions) are happening concurrently. The same essentially applies to the simulation of waves. When uncertainty is attributed to some process in a complex model, via a specific input parameter or grid resolution for example, then *a priori* it is not clear what the effect of that uncertainty will be. In simple simulations, such as those described in sections 2.4.1 and investigated in chapters 4 and 5, the effects

[11] The flexibility of the parameterisation (i.e. number of free input parameters) depends largely upon the authors of the code.

may be more traceable, but even then that is not necessarily so.

2.2.1 Categorising uncertainty

Before considering sources of uncertainty in more detail let us briefly step away from the modelling of waves to specify clearly what we mean by uncertainty. Recall from chapter 1 that Kennedy and O'Hagan (2001) (their section 2) set out terminology for different types of uncertainty in the context of computer experiments (the term "code" used here generally refers to a numerical model). Their terminology is reviewed here, and generally adopted throughout this thesis.

Parameter uncertainty refers to code input parameters, that we regard as unknown. This source of uncertainty is predominantly studied in this thesis. Note however that terminology varies across the literature. For example, Williamson et al. (2012) distinguishes between "model parameters" and "decision parameters", where decision parameters are associated with input that reflects a specific policy decision. In their case decision scenarios were related to real situations, such as carbon dioxide content of atmospheric forcing (in a climate research context).

Model inadequacy refers to the difference between code output at the true input values and true value of the process being modelled. The term *structural error* is also commonly used to describe the same kind of uncertainty. Kennedy and O'Hagan (2001) caution that we need to be clear about the meaning of "true" values. For example, if the input parameters are empirical, "true values" are not at all clear. A possible interpretation could be the difference between the observed mean value of the real process, and the code output at inputs determined through a calibration exercise (see section 3.4.5). That is, where the input has been found to minimise (in some sense) the difference between code output and the observed mean value of the process. In this thesis the term *structural error* is generally used in place of model inadequacy.

Observation error is self explanatory. This is typically only relevant to the calibration problem but observation error may be significant. In this thesis (see chapter 7) observations are typically of wave summary statistics from data buoys, although estimates of observation error are not readily available. In addition, observation error can apply to wind forcing for a wave model, in the sense that the winds are subject to such errors. However, in this thesis, wind uncertainty is regarded in the sense of parameter uncertainty, and is described in more detail in section 2.2.3.

The above paragraphs detail the types of uncertainty analysed in this thesis. Kennedy and O'Hagan (2001) do list additional types of uncertainty that are not explicitly relevant to this work. However, the term *numerical error* is also identified

and used in section 2.2.4. This refers specifically to error associated with the approximate numerical solution produced by the finite differencing scheme. Otherwise this would be incorporated into structural error.

2.2.2 Uncertainty about source terms

Details of uncertainty about source terms, and how it is specified is discussed in this section. In practice, uncertainty about source terms can be expressed and analysed by means of tuning parameters. In principle, (we would hope) the formulation of a parameterisation captures the physics to at least a good approximation, and then input parameters can be adjusted to tune the performance of the parameterisation to give optimal results (given a comparison with some set of observations). That is, there is uncertainty about the details of the physical processes, and tuning parameters provide a means of expressing this. Different choices of parameter values lead to different parameterisations that may give better performance in different physical situations. Larger uncertainty often leads to more tuning parameters but remember that the input space grows geometrically with the number of parameters so uncertainty can grow rapidly, and it is often not at all clear *a priori* how parameters interact, or whether they give rise to nonlinear model response. While for wave models a discrete number of parameter choices is often prescribed, say two, for stable and unstable wind stratification (see TC96), in fact a continuous input space exists, any point in which might be best suited to some given conditions. Our uncertainty is then essentially what input is the "best" and this can be expressed though some choice of probabilistic specification (i.e. a probability distribution) for the input. Typically the probability distribution would be determined through expert elicitation, or interrogation of the relevant literature.

Bearing this in mind, let us return to consideration of the source terms. Although the source terms are often written and discussed separately (for good reasons, such as to specifically identify physical processes), as Ardhuin et al. (2010) note, the separation is somewhat arbitrary. For example, the input and dissipation parameterisations are typically treated together and used as a kind of "tuning knob". The two provide a way of balancing energy input and output in equation (2.7). However, the whole right hand side of (2.7) is nonlinear and so separating the terms while perhaps convenient, is not necessarily going to lead to the best model. Furthermore, from a more general modelling point of view, given the possible interactions between the source terms it is not clear how adjustment of the formulation of individual terms will result in overall changes. In this sense, the behaviour of the wave model "as a whole" seems to have been largely overlooked. For example, TC96 performed tun-

ing on the basis of using the DIA scheme for S_{nl} with specific settings. The same approach was adopted by Ardhuin et al. (2010), and indeed most other researchers. On the other hand, analysis and development of S_{nl} tends to be performed on the basis of a fixed choice of S_{ds} This is even the case for very recent (and important) developments in the computation of S_{nl}, such as Perrie et al. (2013), Tolman (2013a) and Tolman and Grumbine (2013) (based upon earlier work by Tolman (2004a)). All are based on use of the default TC96 scheme, which is itself known to have a number of shortfalls. In fact, Tolman (2004a), who explicitly analyses the effect of adjusting the tuning of the DIA scheme with respect to the exact nonlinear computation, represents possibly the only work where in-depth analysis of the effect of parameter adjustment has been explicitly published. With respect to S_{ds}, important studies such as Bidlot et al. (2007), Ardhuin et al. (2010) and Rogers et al. (2012) all perform some kind of parameter tuning but say little or nothing about how it was undertaken or how uncertainty about the approach might affect performance. In no cases have parameterisations been adjusted simultaneously, nor have beliefs about choice of parameterisation been formally incorporated into an analysis via a probabilistic specification.

The approach to uncertainty analysis taken in this thesis is to consider variation in both S_{ds} and S_{nl} simultaneously (together with variation in other input in some cases). This therefore represents a much more comprehensive approach than has been done before. Consideration of S_{ds} and S_{nl} is limited to TC96 and the DIA scheme respectively. Detailed analysis of the spectrum, such as that performed by Tolman (2004a) is not generally undertaken, but the effects of tuning uncertainty are analysed with respect to wave summary statistics in a variety of cases.

On a practical note, WW3 provides direct user control of input parameters via namelist input, and the namelists for TC96 and the DIA are designated, SIN2 and SNL1 respectively. SIN2 provides control over the six parameters associated with energy dissipation and SNL1 provides control over two key parameters for the DIA scheme. These parameters, and elicitation and specification of uncertainty about them, are described in chapter 4.

2.2.3 Forcing winds

Historic studies have shown that improvement in forcing wind accuracy gives a corresponding rise in wave model performance. Janssen et al. (1997) cites improved forcing wind fields as the primary reason for wave forecast improvement at the ECMWF[12]. Bidlot et al. (2002) provide a model inter-comparison study and note

[12]European Centre for Medium-Range Weather Forecasting, http://www.ecmwf.int/

that the (forecasting) centre with the lowest wind speed errors also has the lowest wave forecast errors. Significant improvements in wind accuracy have been made over more recent years, however Cavaleri et al. (2007) note that winds still lack short term variability. Given, say, a 6 hour time step between wind updates in a global model, it is easy to imagine that there is still considerable room for improvement. Short term variability associated with storms for example, is often blamed for poor performance in such conditions. There is no doubt that the improvement in wind accuracy has alleviated many early problems. Cavaleri (2009) acknowledges such improvements and suggests that attention should shift back towards wave model physics if further improvements are to be made. Nonetheless, wind is so fundamental that there will continue to be examples of where wind accuracy is a major source of uncertainty. Bricheno et al. (2013) report notable improvements in simulated wind accuracy (from an atmospheric model) when reducing resolution from 12km to 4km, and this had knock-on improvement for simulated wave conditions.

A distinction should be made between types of wind forcing that are used in wave simulations. For example in operational forecast settings a wave model is forced by winds generated from a forecasting atmospheric model. Studies such as Bidlot et al. (2002) report and compare performance statistics for such predictions. However wave models are also used extensively to produce hindcasts from observed data. Observational wind data may be obtained from land or sea based observational platforms (such as that used in chapter 7), or as is more common now from satellite based radar. Chu et al. (2004) use winds observed using the Nasa QuickSCAT scatterometer to force a wave model for the South China sea. Commonly, simulations and observations are combined through the use of statistical techniques such as data assimilation to produce a "re-analysis" product. Uppala et al. (2005) describe the ERA-40 re-analysis product. While such products result from efforts to achieve high quality, there are often considerable discrepancies between products. Ponce de Leon and Guedes Soares (2008) and Feng et al. (2006) use a number of such products to force wave simulations for comparative purposes and find stark differences in wave simulation performance. Both conclude that the differences in performance are due to the differing winds.

In order to investigate the effect of wind uncertainty using quantitative methods we need to be able to characterise and quantify it in an appropriate way. However, the complexity of the wind makes this somewhat challenging. Wind quality or accuracy is routinely quantified, and often discussed in terms of summary statistics, for example Bidlot et al. (2002) uses RMSE, bias and scatter index when comparing observations with winds simulated by various forecasting centres. Studies such as

Vogelzang et al. (2011) assess, in considerable detail, the accuracy of global winds obtained from space-borne scatterometer. They describe the (complex) process of obtaining wind measurements in this way, and broadly conclude that the accuracy of winds measured at a resolution of 12.5km is to within $1ms^{-1}$ (but do not provide distributional data). However, the use of single measures to characterise such a massive amount of complex data is an enormous simplification. Various post-processing is often required which could involve, for example, truncation error (Bidlot et al., 2002). Wave models typically utilise winds measured at a height of 10m and so wind data may require scaling to this height based upon certain assumptions (such as stable atmospheric flow) which may or may not be fully justified. Additionally, all of the above studies are with respect to winds observed or simulated globally. Characterisation of accuracy at the global level will not in general apply to a specific region and uncertainty may have spatial or temporal dependence. This is particularly important to remember noting that poor performance of a global wave model is often attributable to very specific geographic regions (see e.g. Ardhuin et al., 2010). Does this imply therefore that the wind is poorly measured or simulated in these regions, or that some aspect of the wave model physics is inappropriate? Clearly that question cannot be answered by performing only a global analysis, and detailed studies of both forcing wind and wave modelling in the specific region would be required.

In this thesis, wind uncertainty is addressed in two ways. Firstly, in chapters 4 and 5 the representation of forcing winds is simple. Uncertainty about the wind is expressed in much the same way as it is described in the above studies, that is by simply giving it a measure of variance for a fixed speed. In chapter 6 the idea of wind uncertainty is discussed and developed further by attempting to represent spatial and temporal uncertainty. To do this, a parameterised approach is used to create synthetic winds, for the purpose of analysis. The details of each approach are described further in the relevant experimental description. Although a relatively simple approach, probabilistic sensitivity analysis yields a quantitative measure of the influence of the wind speed (or other characterising parameter) including any nonlinear response induced, which in turn provides detailed insight into the workings of the wave model.

2.2.4 Numerical schemes and uncertainty

Error associated with the numerical solution to the action balance equation (2.7) is an important source of uncertainty to be considered. To implement a numerical solver the spatial and spectral domains have to be discretised which leads to number

of different types of error and these have been the subject of various investigations (Booij and Holthuijsen, 1987; Tolman, 1992, 1995, 2002a; Rogers, 2002). Cavaleri et al. (2007), section 8, discusses and summarises the important issues, which are also mentioned in Tolman (2009b).

In this thesis numerical error, or uncertainty, is not specifically investigated for reasons set out at the end of this section. The justification is essentially that, for the type of experiments conducted, numerical error does not manifest itself in ways that typically make a big contribution to overall error, given the use of sufficiently high resolution. However, this issue is discussed further in the following paragraphs. WW3, like other third generation models (e.g. WAMDI, 1988; Booij et al., 1999) uses a finite-differencing scheme to solve the action balance equation. It is possible to implement other schemes, such as finite-element or ray-tracing methods but each has their own disadvantages. Tolman (1995) and Cavaleri et al. (2007) provide discussion on this matter. In general, finite-differencing schemes are easier to implement for wave models. WW3 employs an explicit scheme, which means that future states are explicitly calculated on the basis of the current one. Explicit schemes unfortunately give rise to errors that are proportional to the time step and in some cases can require very small time steps to avoid solution instability. That is, solutions are not stable unless the size of the time step satisfies the Courant-Friedrichs-Lewy (CFL) condition described by Courant et al. (1967). This dictates a minimum time step that scales geometrically with geographic grid and spectral resolution. Note that the SWAN wave model for coastal waters (Booij et al., 1999) makes use of an implicit method which does not suffer from this problem. However implicit schemes are generally not practical for global simulations. WW3 in fact allows the specification of four different time step criteria in order to obtain optimal performance. In addition to the global time step, the spatial, intra-spectral and minimum source term intregration time steps can be specified. In general if the global time step is sufficiently small the others need not be different. However, WW3 determines these steps dynamically and therefore performance can be improved with appropriate choices.

Numerical schemes give rise to numerical *diffusion* and *dispersion*, which can affect simulations in various ways. Numerical diffusion arises from even order truncation errors and is an effect that tends to "spread" energy around, as the name suggests. That is, when looking at time series of wave heights or periods, maxima may be slightly underpredicted and minima slightly overpredicted. This would typically affect highly non-uniform conditions and could be thought of as a loss of fidelity. First order finite-differencing schemes are computationally cheap and also

offer numerical stability but suffer considerably from numerical diffusion. Rogers (2002) provides an illustration of the effect of diffusion when conducting a propagation test (see his Fig. 1). Numerical dispersion arises due to odd order truncation errors and tends to cause separation of spectral components (but note that this is not the same as natural dispersion). This can manifest itself as nonphysical features or "wiggles" that are essentially oscillations around a solution as it converges. Other specific effects include the "garden sprinkler effect" (GSE) which is due to coarse spectral resolution (see e.g. Tolman, 2002a; Janssen, 2008). Due to a discretised spectral grid, the propagation of swell over a long distance results in the "disintegration" of the swell into discrete geographic features. The continuous spread of energy is then lost and the resulting arrangement of features resembles the pattern of a garden sprinkler. An image given in Rogers (2002) provides a nice illustration (his Fig. 2). Obviously coarse spatial resolution will also result in errors where geographic features like islands or coastlines are not adequately resolved. All of these errors are functions of spatial, temporal and spectral resolution but the choice of numerical scheme is also important. Some first order schemes result in a strong preference for propagation in a specific direction, whereas higher order schemes can exhibit little or no numerical diffusion which exacerbates certain problems such as the GSE.

Tolman (1992) explicitly examines the effects of numerics on a number of idealised and realistic cases for a third generation wave model. He concludes that errors resulting from numerical schemes can be considerable and must be either removed or accounted for, and proposes a number of improvements to the numerics in order to alleviate some issues. Tolman (1995) goes on to make recommendations taking into account developments. He sets out four requirements for a good numerical scheme for a third generation wave model based upon experience, these being: 1. *The scheme has to result in accurate propagation of poorly resolved swell fields over large distances at an angle with the spatial grid.* 2. *The scheme has to be able to deal with extremely poor resolutions, in particular, with respect to spectral directions (see Booij et al., 1999).* 3. *The scheme should result in negligible spurious oscillations and/or negative wave energy.* 4. *When combined with source terms, the scheme should result in stable fetch-limited growth.* His choice of the "ULTIMATE QUICKEST"[13] third order accurate scheme remains the default scheme in the current release version of WW3, and is found to have good properties by Rogers (2002), for example. In the discussion of this subject, Cavaleri et al. (2007) points

[13]The name derives from the combination of the "QUICKEST" ("QUICK" with Estimated Streaming Terms) convective scheme (Leonard, 1979) and the "ULTIMATE" total variance diminshing scheme (Leonard, 1991).

out that the importance of numerics as a source of error depends very much on the scale of simulation, the type of simulation (e.g. is it dominated by swell?) and the type of scheme used. For example, Rogers et al. (2002) look specifically at reducing numerical diffusion at the coastal scale. However, it is important to remember that the size of these errors are resolution dependent, and they diminish with increasing resolution, although the rate of improvement may not be monotonic.

The discretisation of the spectral domain gives rise to another potentially important source of numerical uncertainty, which has not yet been mentioned. The distribution is generally discretised into between 20 and 40 frequency bins that span a frequency range from 0 to (typically) about 0.60Hz. The tail of the distribution, lying beyond the maximum value in this resolved range, is expressed parametrically. Based upon an assumption that high frequencies are dominated by wave breaking (see e.g. Holthuijsen, 2007), Phillips (1958) proposed that the tail of the distribution decayed as f^{-5}. Such a relationship has been shown to be appropriate for a range of sea states, including growing seas, rather than just JONSWAP type spectra. However, there is a continuing debate as to whether an f^{-4} relationship is more appropriate, and it has been shown (e.g. Alves et al., 2003) that the original JONSWAP data agree better with an f^{-4} tail, rather than an f^{-5}. In terms of a contribution to uncertainty this raises at least two important points. Firstly, the choice of -4 or -5 (or perhaps some value in between) may be influential. For example, the wave zero-crossing period given in equation (2.5) is sensitive to the kurtosis (the fourth moment) of the spectrum. The behaviour of the parametric tail clearly influences the kurtosis, and therefore any investigation concerned with the evaluating the zero-crossing period might need to be cognisant of its potential impact. However, as Holthuijsen (2007) notes, for engineering purposes the representation of the tail as f^{-5} remains effective and commonplace. This is in fact the case for the TC96 input and dissipation scheme, where the tail decays as f^{-5}. This cannot be adjusted without direct modification of the WW3 source code.

Secondly, the magnitude of the influence is dictated by the value of the high frequency cut off. Frequencies can be resolved up to higher values, thus reducing the length of the parameterised approximation. In this thesis the frequency is typically resolved up to 0.67Hz, thus minimising the influence of the parametric tail. Although not considered in this thesis, it would potentially be straightforward to include the high frequency cut off as an uncertain parameter and evaluate its effect on simulation output.

As outlined above, numerical uncertainty is fairly well studied and understood, and the situations most susceptible to numerical uncertainty typically involve the

propagation of swell over long distances, use of certain numerical schemes and coarse spatial or spectral resolution. Since this is not characteristic of simulations investigated here, and high resolution is generally used, numerical uncertainty is judged to be small and not explicitly investigated in this thesis. However, it is recognised that for more complex large scale simulations, where there is typically a trade-off between numerical accuracy and performance, investigation of the effects of numerical error would be very important.

2.2.5 Boundary conditions

Boundary conditions, that is, the wave conditions or geographic specification at the boundary of the spatial ocean domain, are potentially an important source of uncertainty. This is not a concern when performing a global simulation but in many cases the focus is on a regional or coastal area where a higher resolution simulation is conducted. Where there is a boundary to the open ocean (when simulating the North Atlantic for example), the wave conditions at the oceanic boundary need to be specified. These may be time-dependent wave conditions, expressed in spectral form or as summary statistics, generated by a larger scale simulation (e.g. global) at lower resolution (Millar et al., 2007; Rusu and Soares, 2009).

In this thesis, the complications associated with time-dependent boundary conditions (such as an open sea boundary) are avoided by choice of simulation. In chapters 4, 5 and 6, simulations are idealised. It is assumed and that the waves generated are entirely due to the specified winds and that no wave systems enter at the boundaries. Further, there is no reflection at the boundaries and in effect, the boundaries "absorb" all wave energy. In chapter 7, lake Michigan, by definition, is entirely enclosed by land and so wave systems cannot be generated remotely. Reflection at the coast could potentially be an issue in such a case, perhaps if waves encountered a sudden cliff face in deep water. However, this is not the case, and more generally is unlikely to be of interest unless very small scale coastal effects are important. In any case, an alternate wave model would be required for such situations because WW3 (version 3.14) does not compute reflection at the boundaries.

2.3 Uncertainty analysis of wave models

As previously discussed there are few examples in the literature of uncertainty and sensitivity analysis with respect to a wave model. Those that do exist tend to

describe the problem in quite broad strokes and often consider only global prediction. Some examples of relevant and related studies are given as follows.

The analysis of Tolman (2004a) is possibly the only published study of a detailed exploration of a (2-D) parameter space. It is a (non-probabilistic) sensitivity study that details the effects of parameter adjustment of two important parameters relating to the DIA scheme. The analysis is, however, focussed on the effect of perturbation on specific wave spectra rather than hindcasting or prediction scenarios. The objective is also to develop a new parameterisation, rather than to explicitly examine the effect of uncertainty.

Examples of a more qualitative uncertainty analysis for a wave model are given by Rogers (2002) and Rogers et al. (2005). Rogers (2002) examines the sources of error in the modelling of low frequency (<0.08Hz) wave energy. He specifically identifies the wind specification and model physics as important sources of error (numerical error being only a lower order source of error). However, the approach taken to the investigation is a kind of traditional "one-of-at-time" analysis. That is, comparison is made by using different forcing winds (from difference sources) and different model physics (WW3 vs WAM) but each is simply alternated in turn, without controlled adjustment of any specific elements. Furthermore, a detailed perturbation of the physics, or winds, is not considered. He concludes that although the wind forcing appears to be the dominant source of error, model physics is clearly important too, but that it is, "*... difficult to make a judgement regarding the skill of WAM4 physics vs WW3 physics. This is especially true with regard to the [wave] generation stage ...*". While this work provides great insight into the issue of uncertainty, it lacks a quantitative methodology and therefore is unable to clearly "disentangle" the sources of error. Rogers et al. (2005) consider the error in global simulations of waves. They pose questions of interest such as, "*If we can identify a scenario where a wave model's representation of physics (generation, dissipation, and nonlinear interactions) is likely to be the primary source of error, is the wave model bias positive or negative? How does the answer depend on the frequency-wavenumber range considered, or perhaps the geographic location?*". In answering the questions they do provide quantitative insight in terms of, for example, the sign of the bias in certain cases, and while making the recommendation that WW3 would benefit from additional tuning or physics improvement, they do not provide details as to how this might best be approached. Other studies in the literature present similar kinds of analysis, but no in-depth quantitative analysis has yet been identified.

An approach to extreme value analysis that makes use of surrogate models is

given by Gouldby et al. (2014), based upon work by Camus et al. (2011). They generate large numbers of samples of nearshore conditions by forcing the SWAN model with boundary conditions from a global model (WW3). Owing to the computational cost of generating the number of samples typically required for such an analysis, they construct a statistical "meta-model" based upon the use of radial basis functions (RBFs). In fact, the form of the meta-model is similar to a regression model with a mean function based upon the RBFs. Once the model is fitted (or trained) to the observations, it is used to generate a large sample of nearshore conditions which can be used for subsequent risk analysis. This is a kind of statistical downscaling and closely related to statistical emulation which is explained in chapter 3.

Some other interesting work involves the application of so called "soft computing" methods to predicting wave conditions (usually in terms of summary statistics). Soft computing techniques include Artificial Neural Networks (ANN), Fuzzy Inference Systems (FIS) and other learning machines. Mahjoobi et al. (2008) provides a brief review. Typically the motivation for utilising such methods is the prohibitive computational expense of running a numerical model, particularly when resolving complex and localised geographical features in the coastal zone (see e.g. Makarynskyy and Makarynska, 2007). Mahjoobi et al. (2008) present a study into the effectiveness of a number of soft computing methods, specifically ANNs and FISs, as a means of wave parameter prediction for hindcasts of wave conditions over lake Ontario. Although these studies are not concerned with assessing the performance of a wave model and quantifying uncertainty, the methods share commonality Gaussian process emulators. A neural network, for example, explains variability in terms of the input training data (much like regression) and therefore can be used for uncertainty analysis. In the aforementioned study, sensitivity analysis showed, perhaps unsurprisingly, that the relative importance of wind speed was significantly higher than the other three input parameters. They conclude that of the various soft methods ANNs gave the best performance though mean square relative error for H_s was not better than 20% for any system.

2.4 Physical situations

In the final section of this chapter consideration is given to some "idealised" wave conditions and how they are generated. The motivation for discussing these is that they are typically used as benchmark cases for studying waves. These situations are generally brought about by winds blowing continuously, in the case of duration- or fetch-limited growth, or changing in a predictable way such as the response of waves to a turning wind. Aspects of wave model skill are assessed and calibrated by

comparison to such cases (see e.g. TC96, Ardhuin et al., 2010). These cases have been used extensively for experimentation in this thesis, and their physical basis is briefly discussed in the following subsections.

2.4.1 Duration-limited wave growth

"Duration-limited" wave growth describes the case where the sea state is governed by the length of time for which the wind has been blowing. Imagine an infinite sea, such that energy lost from waves propagating away from the area is exactly balanced by energy arriving into the area. In the absence of propagation the energy balance equation (2.7) can be written as,

$$\frac{\partial E(f,\theta)}{\partial t} = S_{\text{wind}} + S_{\text{nl}} + S_{\text{ds}} \tag{2.8}$$

The evolution of this with time is actually shown in figure 2.2. Such simulation output can be compared with observations or spectra derived from theoretical considerations such as given by Pierson and Moskowitz (1964). In this situation the wind speed can be regarded as known (and constant), and therefore variation in the resulting spectra can be directly attributed to adjustment of specific source terms. Although in this thesis $E(f,\theta)$ is typically reduced to H_s or T_p, the approach is the same. The effect of parameter perturbation on the fully developed sea state (i.e. where the left hand side of (2.8) is set to zero) is performed in chapter 4.

2.4.2 Fetch-limited wave growth

A well studied phenomenon is that of "fetch-limited" wave growth. The idealised arrangement for this situation is described in many authoritative texts (see e.g. Holthuijsen, 2007) but is summarised here. As the name suggests under fetch-limited conditions wave growth is limited by the length of fetch over which the wind can impart energy (as opposed to the duration of time). Formally, consider an infinitely long straight coastline, meeting a very deep ocean. If a steady wind blows perpendicular to the coastline waves will grow steadily with time. Waves at a given location will be due to both the sum of energy from direct forcing, and energy propagated from the upwind direction. At locations close to the coastline the upwind fetch is short thus limiting the amount of energy propagated to that point. At locations further and further away the fetch increases, thus increasing the propagated energy budget. Sufficiently far from the coastline a point is reached that approximates fully developed conditions, because the incoming energy is balanced by the out-going energy. The region from the coastline to this point, where the wave

spectrum remains a function of fetch, is thus called fetch-limited.

Numerous experiments have studied this phenomenon, a famous example is the JONSWAP campaign (Hasselmann et al., 1973). The experiments were conducted over a period of ten weeks during 1968 and 1969 with the aim of determining the source function. An ambitious project, it required the use of thirteen measuring stations aligned over a stretch of the North Sea, such that when steady winds blew in the direction of alignment, both wind and waves could be measured accurately. Direction integrated 1-D spectra were recorded, allowing for the study of the evolution of the spectrum along the fetch. From those observations the authors were able to define the following empirical function that fitted the shape of the spectra well,

$$E(f) = 0.076\,\tilde{x}^{-0.22} g^2 (2\pi)^{-4} f^{-5} \exp\left\{ -\frac{5}{4}\left(\frac{f}{f_p}\right)^{-4} \right\} \gamma^{\exp\left(-\frac{(f-f_p)^2}{2\sigma^2 f_p^2}\right)} \tag{2.9}$$

$$\sigma = \begin{cases} \sigma_a, & \text{for } f \leq f_p \\ \sigma_b, & \text{for } f > f_p \end{cases} \tag{2.10}$$

where $\tilde{x}$ is the dimensionless fetch (see below), f_p is the peak frequency, g is acceleration due to gravity and γ, σ_a and σ_b are free parameters to fit the function. Note also that the multiplicative constant α often seen in front of (2.9), and first introduced by Pierson and Moskowitz (1964), has been replaced by,

$$\alpha = 0.076\,\tilde{x}^{-0.22} \tag{2.11}$$

Hasselmann et al. (1973) suggest this relationship based upon scaling considerations. A visualisation of this is given in figure 5.2. Kahma and Calkoen (1992) discuss the discrepancies between various observations of fetch-limited growth.

Equation (2.9) was written as a function of "dimensionless" fetch. It turns out that summary quantities, such as wave height, period and fetch can be written in a dimensionless form. That is, behaviour of waves is found to scale with wind speed and so observations in mild conditions can be used learn about waves in more energetic conditions. In practice this means quantities can be expressed in a kind of scale invariant way as follows. The dimensionless fetch, $\tilde{F}$ is computed by,

$$\tilde{F} = \frac{gF}{U_{10}^2} \tag{2.12}$$

where U_{10} is the wind speed at a height of 10 metres. Dimensionless wave height

and period are converted in a similar way.

$$\tilde{H}_s = \frac{gH_s}{U_{10}^2} \tag{2.13}$$

$$\tilde{T} = \frac{gT_p}{U_{10}} \tag{2.14}$$

We make of these quantities in experiment 5.A when considering fetch-limited growth.

2.4.3 Wave response to a turning wind

Another idealised case often used as a benchmark in wave model testing is that of a "turning wind". The physical idea is akin to duration-limited growth described earlier, but that at some point during the growth phase, or even under fully-developed conditions, the wind changes direction rapidly by 90°. Compared with the previous two cases the situation of a turning wind is somewhat more dynamic in nature. During the SWAMP (1985) wave model intercomparison study, a turning wind was used as a benchmark test, as it was by Komen et al. (1994), who also use it as a test case during wave model development. A number of researchers have studied the case of a turning wind and made observations of the phenomenon on the ocean. Holthuijsen et al. (1987) proposed a "relaxation model" based on theoretical considerations and obtained supporting observational data. The model, which describes the response of the wave direction to a change in wind direction, is as follows,

$$\frac{\partial \theta_0}{\partial t} = \frac{1}{\epsilon}\frac{\partial \epsilon}{\partial t} sin(\theta_w - \theta_0) \tag{2.15}$$

where θ_0 is the wave direction, θ_w is the wind direction and $\frac{1}{\epsilon}\frac{\partial \epsilon}{\partial t}$ is essentially a dimensionless timescale that can be approximated as a constant.

Perrie and Toulany (1995) propose an alternative model, and Masson (1990) and van Vledder and Holthuijsen (1993) provide further considerations on this subject. Of particular note, although numerous observational data have been obtained on this subject, owing to the difficulty in obtaining accurate data and in well characterised conditions, there is still considerable uncertainty surrounding this physical process.

Like wave height and period, wave direction is often derived and summarised from the directional frequency spectrum (figure 2.1). "Mean wave direction" is self explanatory but it is accompanied by the "directional spread". Although not examined extensively in this thesis (see chapter 6, note that wave directional spread is a measure of variance obtained by integrating the frequency spectrum over direction.

Holthuijsen (2007) provides the following,

$$\sigma_\theta^2 = \int_{-\pi}^{+\pi} \left[2sin\left(\frac{1}{2}\theta\right)\right]^2 D(\theta) d\theta \tag{2.16}$$

where $D(\theta)$ is $E(f, \theta)$ integrated with respect to frequency. The expression in the squared parentheses is mathematically more tractable than simply θ.

3 Uncertainty analysis using Gaussian process emulators

This chapter discusses uncertainty with respect to numerical models more generally, and related statistical analysis, including statistical models called *emulators*. Firstly, in section 3.1, motivation for such analysis is presented. This is followed by sections 3.2 and 3.3 that describe methods of uncertainty and sensitivity analysis that will be used in this thesis. Section 3.4 describes a Gaussian process emulator, which is followed by more details of extended applications in section 3.4.5. The chapter concludes with section 3.5 which provides some examples of the application of emulators to simple wave models.

Before proceeding, a note is made on terminology. Where statistical analysis is performed for numerical models, as is the case in this thesis, the word "model" becomes overloaded and so to clearly distinguish between a statistical model and a numerical model, it is customary to refer to the latter as a *simulator*. This more clearly emphasises the fact that we are conducting computer simulations of, in this case, ocean waves. If, on occasion, the word "model" is used in a different context, it will be preceded by a qualifying word for clarification.

3.1 Uncertainty in computer simulations

Before describing specific methods of analysis in more detail, some comments are noted on how, in the context of computer simulations, uncertainty is managed and reported in the literature. Its treatment ranges from the rigorous, where robust statistical methods are brought to bear on a problem, to less rigorous, where uncertainty is discussed rather qualitatively. For example, Rougier et al. (2009a) apply robust statistical methods, including the formal incorporation of expert knowledge, to perform inference for a simulator of the electrodynamics of the upper atmosphere. This contrasts with the work of Bidlot et al. (2007) which discusses improvements made to the ECMWF[14] operational wave forecasting system. Although the two cases differ in terms of specific objectives, the level of success is very much related to understanding simulator uncertainty and sensitivity. In the latter case, there is an overarching question of uncertainty in wave simulator operation due to unknown physical and tuning parameters. In relation to the determination of three such parameters, the authors state that, "*A tuning exercise was performed in such a way that the duration limited growth curve for significant wave height and the time*

[14] http://www.ecmwf.int/

evolution of the Charnock parameter[15] *resembled as much as possible the corresponding results of the reference model...*". (Footnote added for clarity.) They do not however provide any information on the certainty of the results nor any detailed information about how they went about the tuning process (how was the sensitivity analysis carried out?). They go on to demonstrate good performance of the updated simulator but this leaves questions about their confidence in the final choice of parameters, and whether further improvement could have been made.

The lack of application of rigorous statistical methods in such cases does not always imply criticism of the authors since the problems are challenging and hampered by computational cost and complexity. Where computational cost is a barrier, an (less robust) approach to uncertainty analysis involves running a small *ensemble* of simulations. This is commonly performed for large physical simulations, such as global ocean circulation or climate. Owing to the computational expense of running the simulation, an ensemble of circa 20 to 50 members is created based upon perturbations of the physics, boundary conditions, etc. The ensemble members are generated by drawing perturbed inputs from a multivariate distribution that represents our uncertainty about the initial conditions. The resulting sets of simulation output (be it sea surface temperature or wave height) provides a measure of uncertainty but it could be regarded as rather crude since the ensemble size is small. Clayton et al. (2013) describes a recent implementation of an (23 member) ensemble based system at the UK Met Office. Multimodel ensembles are also commonly used, perhaps most notably in the Climate Model Intercomparison Project (CMIP)[16]. The objective of this project is to generate large datasets from numerous different simulators in a coordinated way, in order to obtain a complete assessment of our uncertainty, from which statements can be made about future climate prediction. A summary of the projects including details of the experiment design is given in Meehl et al. (2007) and Taylor et al. (2012). Enormous amounts of research relating to the simulation of the earth system has arisen from this project. One example relevant to the simulation of waves is presented by Hemer et al. (2013) who analysed the future wave climate derived from a subsample of five simulations from the CMIP3 ensemble to determine expected future wave climate in different regions of the oceans. However they point out fairly severe limitations in their work in terms of the small sample size and the range of sources of uncertainty, which include differences in simulators, atmospheric forcing and the methodology that was used in each case to determine the future wave climate. They suggest that the uncertainty associated with the re-

[15] The surface roughness length which relates the sea-surface roughness to wind stress, see e.g. Komen et al. (1994).

[16] http://cmip-pcmdi.llnl.gov/cmip5/

spective methodologies was the dominant source of uncertainty but that the sample size was too small to determine total uncertainty, and that simulations from the larger CMIP5 ensemble would be better suited to this task[17]. Hemer et al. (2013) do not attempt to explore and quantify in detail uncertainty due to the tuning and structure of a wave simulator as is performed in this thesis. Multimodel ensembles capture such uncertainties in a broad sense but cannot be used for a detailed analysis. Rougier and Goldstein (2014) discuss some issues with respect to the analysis of uncertainty in the context of running multimodel ensembles and coupled models.

So we can see from this brief discussion there is a range of rigour in the methods with which uncertainty is analysed and addressed in the literature. It is clear that computational cost is a key issue preventing detailed uncertainty analysis in many different cases. In the next section some methods of uncertainty analysis are reviewed, which reveals why complex and computationally expensive simulators present such a challenge.

3.2 Uncertainty analysis

The purpose of an uncertainty analysis is to evaluate the uncertainty in an output induced by uncertainty on the inputs. In a sense there are at least two steps involved, being elicitation of the input uncertainty, and subsequent calculation. Before any computation can take place our uncertainty about the input must be determined and appropriately specified. In this thesis uncertainty will be discussed in terms of probability and probability distributions. This may be determined from observational data or more formally in an expert elicitation exercise. One or more experts with knowledge of the simulator and its input may be questioned in such a way that their uncertainty can be elicited in terms of probability distributions. An important aspect of this process is that an expert's *subjective* uncertainty be captured and accurately expressed, although this can be a subject they might rather avoid given that it requires honesty about the limits of their knowledge. Rougier et al. (2009a), Brito et al. (2012b) and Brito et al. (2012a) all give examples of how expert knowledge was elicited and incorporated into an uncertainty analysis. Brito et al. (2012a) provides an appendix giving details of how the elicitation process was conducted, which highlights its importance. Multiple experts were selected from around the globe, and were brought together on more than one occasion to provide information. The information was consolidated and used as the basis of the risk assessment. O'Hagan et al. (2006) goes into considerable detail regarding the elicitation process.

[17] Results from the CMIP5 ensemble were yet to be produced.

Given that uncertainty is expressed as a probability distribution, the shape of the distribution will be dictated by the form of the uncertainty. A normal or Gaussian distribution is symmetric and so represents the probability decaying at the same rate for values above and below the mean. In contrast, a *Beta*-distribution can be specified by choosing appropriate shape parameters[18] which results in a "hard" limit with a minimum or maximum value, implying that there is zero probability of values beyond this limit (an example of the use of a *Beta*-distribution is given in chapter 6). Where very little is known about the value other than it is constrained by some physical limits, a uniform distribution may be appropriate. Some caution must be adopted during the elicitation process, however it is conducted, in order that beliefs are expressed and / or captured accurately. For example, people have a tendency to give information in a rather rhetorical way - "*We've always done it like that...*" is a common response to enquiry about why a process is done in a certain way. By comparison, an expert may feel most comfortable putting firm limits on the possible values of certain simulator input for similar reasons, but that might not be their true considered judgement if they think about the situation in a broader context. Indeed, they may not have given it much prior thought at all. In fact, when they consider the enquiry in more detail they might concede that hard limits are unrealistic. It is probably fair to say that in most examples, placing hard limits on the range of possible values of a parameter that can be quantified, whether it is simulator input, values of physical constants, consequences of an action, etc. is unrealistic.

Once the uncertainty distribution for the simulator input has been established, the next step is to perform statistical inference. This can be described quantitatively as follows. If the computer simulation is a deterministic function $f(\cdot)$ with inputs $\mathbf{X}$ distributed as $G(\mathbf{X})$, then the purpose of the uncertainty analysis is to determine the distribution induced in $f(\mathbf{X})$. Specifically, the expectation is computed by,

$$E\{f(\mathbf{X})\} = \int_{\chi} f(\mathbf{X}) \mathrm{d}G(\mathbf{X}) \tag{3.1}$$

The form of $f(\cdot)$ dictates how difficult this calculation is to perform. In a simple case, it may be possible to closely approximate $f(\cdot)$ by a simple linear form such that for a specific input x,

$$f(x) \simeq h(x)^{\mathrm{T}}\beta \tag{3.2}$$

where $h(\cdot)$ is a vector of basis functions and β is a vector of coefficients. If the basis

[18] In a similar way that a normal distribution is specified by a mean and variance, a *Beta*-distribution is specified by two *shape* parameters typically labelled as α and β.

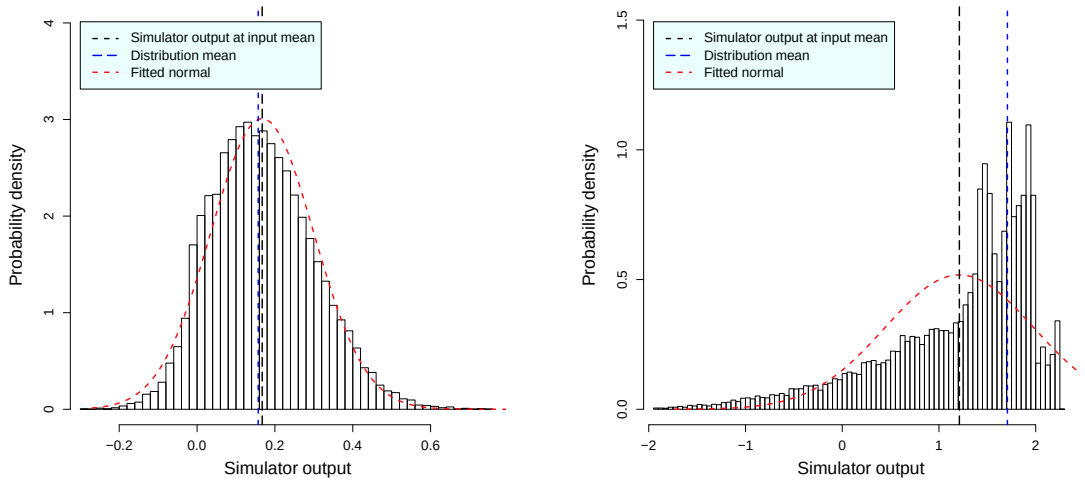
functions are sufficiently tractable, such as $h(\cdot) = [1, \cdot]^{\mathrm{T}}$, then,

$$E\{f(\mathbf{X})\} \simeq \int_{\chi} h(\mathbf{X})^{\mathrm{T}} \beta \, \mathrm{d}G(\mathbf{X}) \tag{3.3}$$

which is trivially solved where $G(\mathbf{X})$ is Gaussian. However if $f(\cdot)$ is some complex nonlinear operator then the approximation (3.2) will be poor and the integral (3.1) will therefore need to be solved by some kind of numerical method. A Monte Carlo approach for example requires a large sample $\mathbf{x} = [\mathbf{x}_1 \,..\, \mathbf{x}_n]$ to be drawn from $G(\mathbf{X})$ (where typically $n = \mathcal{O}(10^4)$ to obtain precise probabilistic information), and $f(\mathbf{x}_i)$ to be computed at each sample point $\mathbf{x}_i$. Often it is prohibitively expensive to run a complex simulator a number of times sufficient for a Monte Carlo approach. This provides us strong motivation to seek a more efficient method to address this problem.

One means of addressing this is through the use of surrogate models. A number of alternative terms, such as meta-models, are used in the literature to describe what is an approximate version of the simulator. In broad terms, a surrogate model can be constructed from output from the simulator in such a way that it accurately captures the simulator's behaviour. If the surrogate model executes rapidly, then uncertainty analysis can be performed using the surrogate model rather than the simulator. One particular type of surrogate model is the Gaussian process emulator, which is described in section 3.4.

Given that it is possible to perform the uncertainty analysis using a Monte Carlo approach, results from such an analysis can tell us different information about the simulator output and its operation. Specifically, nonlinearity will be evident if the output distribution is significantly different from the input distribution. Two example output distributions generated by a simple simulator with a single dimension of input are shown in figure 3.1. Panel (a) shows a distribution with a slight skew but which can be seen to be close to normal, when compared with the normal distribution fitted to the output mean and variance. We can therefore deduce that the simulator response is approximately linear. In contrast, panel (b) shows a highly skewed distribution which also exhibits multiple modes, revealing a highly nonlinear response in the simulator. Note also that in panel (a) the mean of the output distribution is very nearly equal to the simulator output at the mean of the input. However, the nonlinear operation exhibited in panel (b) has resulted in an appreciable difference between the output mean and the output run at the mean input. In such a case, if the uncertainty were not considered and only a single run (or small sample) were evaluated near the mean input (essentially on the basis of a linear response), any



(a) Uncertainty distribution from a near linear response in the simulator to its input.

(b) Uncertainty distribution from a highly non-linear response in the simulator to its input.

Figure 3.1: Two contrasting uncertainty distributions induced in simulator output for a simple one dimensional simulator.

subsequent calculation or action based on the result could be misguided.

While an uncertainty analysis tells us about uncertainty induced in the output, and reveals nonlinearity in the simulator, in the case of multiple dimensions of input, it does not tell us details about any individual dimension of input such as how each contributes to the output uncertainty. A sensitivity analysis can give us this information, which is discussed in the next section.

3.3 Sensitivity analysis

Sensitivity analysis (SA) is a companion to uncertainty analysis in the sense that it reveals information about how the specific sources of input uncertainty induce output uncertainty. It provides a means of obtaining quantitative measures of influence for the various sources of uncertainty within the simulator. In a sense this allows us to "unpick" the inner workings of the model, which we might want to do for various reasons. French (2003) provides an extensive discussion of the motivations for performing SA. In the context of this work for example, SA could tell us, quantitatively, how simulated wave period responds to change in a given free parameter. Moreover, we can determine the combined effects of two or more dimensions of input - that is whether the inputs significantly *interact*. Specification of our uncertainty is important in order to obtain meaningful results, but SA is often conducted, if a little crudely, by giving only a passing thought to the formulation of our beliefs.

Nonetheless, it may be instructive to determine the change in the output for an arbitrary change in input or we may simply be curious about what happens when we push input to more extreme values.

Many different methods of performing sensitivity analysis are well established. Saltelli et al. (2000) provide a comprehensive and in depth discussion of different methods and relevant issues. An important distinction to make is the difference between a simple one-at-a-time (OAT) analysis and a simultaneous analysis that captures interactions between inputs, and any resulting nonlinearity. An OAT analysis will only be effective if the simulator has a linear response to all input. A common approach is to use a variance based method, which involves decomposing the total variance with respect to each of the input sources. That is, to determine what proportion of the total variance is attributable to each input. In this way, sensitivity or *importance* measures can be defined for each input, which provides a standardised way of comparing the influence of different input. As we saw in the previous section for uncertainty analysis, when analysing a complex simulator typically there are no closed form solutions to the resulting equations for SA, and so numerical methods are generally required. However, the resulting integrations usually require at least an order of magnitude more computation power than for a simpler uncertainty analysis. Moreover, it is usually very difficult to re-use the same data for further analysis. If the simulator is very fast to execute then these issues are not important but this is not the case with many modern complex simulators. Again this provides strong motivation to find a more efficient approach.

Saltelli et al. (2005) provide some examples of sensitivity analysis in practice, and cover variance based methods. Jacques et al. (2006) propose using a similar approach in the context of correlated inputs and simulator variation. Oakley and O'Hagan (2004) also advocate this approach and Lee et al. (2011) gives an example of its application (utilising an emulator) in the context of the analysis of a numerical model for atmospheric aerosols. Some of the mathematical formulation for a variance decomposition approach is provided in the remainder of this section. For some input $\mathbf{x}$ with elements $x_1 \dots x_d$ that are indepdendent and that correspond to the d dimensions of input to the simulator, where $\mathbf{x}$ is a realisation of the random vector $\mathbf{X}$ we can write the value of some deterministic function $y = f(\mathbf{x})$ as the following decomposition,

$$f(\mathbf{x}) = E(Y) + \sum_{i=1}^{d} z_i(x_i) + \sum_{i<j} z_{i,j}(x_{i,j}) + \sum_{i<j<k} z_{i,j,k}(x_{i,j,k}) + \cdots + z_{1,2,\dots,d}(x) \quad (3.4)$$

where z are referred to as the effects and interactions of the input x_i. The *main*

effect, z_i, is given by,

$$z_i(x_i) = E(Y|x_i) - E(Y) \tag{3.5}$$

the *first order interactions*, $z_{i,j}$, are given by,

$$z_{i,j}(\mathbf{x}_{i,j}) = E(Y|\mathbf{x}_{i,j}) - z_i(x_i) - z_j(x_j) - E(Y) \tag{3.6}$$

and the higher order interaction terms follow the same pattern.

The main effect is a function of x_i and it is calculated by marginalising over all other inputs[19], x_{-i}, and running the simulator at a subset of values of x_i consistent with our uncertainty. The main effect z_i is therefore an expectation and accounts for the uncertainty about x_{-i}. It can be seen that for a linear function, where there is no interaction between elements of $\mathbf{x}$, only the main effect term will be of interest. However, interactions may be particularly important in nonlinear systems. Main effects and first order interactions are functions of only one or two variables and can therefore be visualised easily giving a clear graphical indication of the sensitivity of the model to each input parameter, or pair of parameters. It is possible to extend the method to higher order interactions but they are difficult to interpret and so it is common practice to focus on lower order terms.

A decomposition for the variance of the effects can be obtained in a way similar to the above, as follows,

$$\text{var(Y)} = \sum_{i=1}^{d} W_i + \sum_{i<j} W_{i,j} + \sum_{i<j<k} W_{i,j,k} + \cdots + W_{1,2,\ldots,d} \tag{3.7}$$

where $W_i = \text{var}\{z_i(x_i)\} = \text{var}\{E(Y|x_i)\}$, $W_{i,j} = \text{var}\{z_{i,j}(x_{i,j})\}$, etc. It can be seen that W_i is the amount by which $\text{var}(Y)$ is reduced if we learned the true value of x_i. In other words, W_i is the proportion of variance attributable to (the input) x_i. Dividing by the total variance $\text{var}(Y)$ gives rise to a scale invariant importance measure referred to as the *main effect index*.

Clearly knowledge of an influential input will lead to a greater reduction in the variance of Y, than knowledge of an inactive input. Somewhat counter intuitively, where interactions are present, the input with the largest main effect is not necessarily the most interesting. Consider a function with three inputs, linear in x_1 but where x_2 and x_3 interact in some way. The response of the function to x_1 may be quite strong, resulting in a large main effect. However, even if the response to variation in x_2 or x_3 is considerably smaller, the variance arising from their interaction

[19]The notation $-i$ denotes all dimensions other than i, adopted from Oakley and O'Hagan (2004)

might be large, resulting in the total variance explained being greater than that for either x_1 and x_2, or x_1 and x_3. There are many aspects of SA that can be brought to bear on the study of simulator response to input, including the identification of interesting inputs, screening methods, and so on. The reader is directed to Saltelli et al. (2000) and Oakley and O'Hagan (2004) and references therein for further information.

As previously noted, many 10,000s of runs of the simulator are required to undertake these kinds of analyses which make it prohibitive for all but the most simplistic of simulators. A robust and efficient alternative approach is therefore desirable, and one such method that has proved very effective over recent years is the Gaussian process emulator. In this thesis the methods outlined above will be used extensively, in conjunction with emulators, to perform analysis for various configurations of a wave simulator. A detailed description of emulators is provided in the next section of this chapter.

3.4 Introduction to Gaussian process emulators

As discussed in chapter 1 and section 3.2, uncertainty analysis is important in the use and application of numerical simulators. However, owing to the prohibitive computational cost of traditional Monte Carlo methods, alternatives have been developed. A Gaussian process (GP) emulator is a type of statistical regression model, formulated using a Bayesian approach, that completely represents a simulator, including the nonlinearity in its operation. It is a robust reproduction of the simulator and so can be used for the accurate prediction of simulator output at input values for which the simulator has not been run. It can do this very rapidly and so it becomes possible to generate the data necessary for a Monte Carlo based uncertainty or sensitivity analysis. In fact, because the principle of emulation is based upon creating a robust statistical model of the simulator, it is possible to perform statistical inference analytically for certain choices of emulator construction. Even where this is not the case, uncertainty analysis can be performed efficiently by utilising components of the emulator model. This is a very powerful tool and is described in more detail in section 3.4.4. Another attractive benefit of using an emulator is that once it has been constructed, typically from less than a few hundred runs of the simulator, it can be used repeatedly to perform various analyses which avoids having to rerun the simulator. In effect the simulator output data can be re-used many times over.

A GP emulator is a nonlinear regression model, fitted to output from a simulator, where the explanatory variables are the various inputs to the simulator. In statistical terms, it is the complete distribution of the simulator output conditional upon each

of the input parameters. It draws its name from the use of the Gaussian process model that is used to represent uncertainty in the simulator. It is closely linked to the Bayesian linear regression model, but utilises a spatial Gaussian process to describe the error structure. In the same way that a Gaussian distribution describes a random variable, the Gaussian process describes a (infinite) collection of variables, any subset of which is jointly Gaussian. In this way it can be thought of as a distribution of functions, that arises due to uncertainty about the functional response to the uncertain input. The emulator model scales well to high dimension input, just as a multiple regression model is used where there is more than one explanatory variables. Without making the emulator model too complex and mathematically intractable, the Gaussian process can be implemented with a mean function formed of linearly independent basis functions of the input parameters. This is very useful when the simulator is nonlinear and the form of its output can be approximated by the use of a few simple basis functions. The choice of the emulator formulation is reliant on some prior beliefs about the form of the simulator, however in practice this can be established through the use of simple regression models. The use of an appropriate mean function and suitable choices for other model components makes the emulator very efficient and accurate at interpolating the simulator output.

Once choices about the emulator structure have been made the model is conditioned on simulator output - that is, observations from the simulator. The choice of observations is not arbitrary, but like any good experiment is subject to a design. Experimental design for building emulators is discussed in detail in section 3.4.2. Conditioned on observations at design points, the GP model, thinking about a distribution of functions, becomes constrained to those that most closely represent the form of the simulator output. In practice this constrains the mean of the conditioned GP model to be equal to the simulator output at each design point. The mean of the emulator is therefore a point predictor of the simulator, and this forms the basis for interpolation of the simulator output. Being a statistical model, the emulator also provides an estimate of the uncertainty about each point prediction (which is zero at the design points). Use of the GP model does not require that the simulator output is in some way Gaussian distributed, only that this is the case for the uncertainty within the approximate model. Sometimes however the application of a transformation to the output data may be required to satisfy this condition[20]. Another important aspect of the use of the GP is that the simulator output is as-

[20] A log-transformation could be used for example, however this can complicate subsequent use of the emulator because the emulator then computes log-transformed output, which requires a reverse transformation. However, closed form solutions are then with respect to the log-transformed output and require careful interpretation if they even obtainable.

sumed to be quite smooth, in that a data point provides information about other points in its proximity. This is typically the case for computer simulators although care must be taken to ensure this is an appropriate assumption, for example if the simulator included a bifurcation or "tipping point".

Emulators are a very powerful tool for performing uncertainty analysis for complex numerical simulators. They can be applied and extended in a number of ways, some of which are discussed further in sections 3.4.4 and 3.4.5. Details of the construction of an emulator are now reviewed, although the reader is directed to the references in this section for further information, many of which deal with this subject is considerable depth. In particular, the formulation of the emulator is very similar to the Bayesian linear regression model, a detailed treatment of which is provided in O'Hagan and Forster (2004). Rasmussen and Williams (2006) provide an extensive discussion of the GP and its properties in the context of machine learning[21]. The MUCM Toolkit[22] is an extensive online resource that covers many aspects of emulation and provides worked examples and case studies. A useful tutorial is provided by O'Hagan (2006) and Santner et al. (2003) discusses various aspects of designing and running computer experiments from both frequentist and Bayesian perspectives, including the undertaking of uncertainty and sensitivity analysis. Finally the similarity between emulation for simulators and the method of spatial "Kriging" is noted. In the field of geostatistics Kriging is a statistical method for modelling and predicting spatial data. It dates back to the 1960's where it evolved as a means of predicting grades of rock ore by means of spatial weighted average. Cressie (1990) describes the early history of Kriging. Initially it was based upon finding the best unbiased linear predictor for spatial data but has now developed into a large field of research. Sophisticated models for spatio-temporal prediction are now widely used for a variety of spatial-temporal problems. For example, Sahu et al. (2011) developed a model for forecasting regional ozone levels based upon real-time data. Emulators are sometimes described as "Kriging for computer models" since mathematically the two fields share many commonalities. In oceanography and meteorology these methods are often known as "objective analysis" and "optimal interpolation". An obvious difference is that geospatial processes are limited to two spatial dimensions whereas emulators are typically aimed at exploring up to tens of input dimensions. However, and with some differences in terminology and notation, the following mathematical discussion can be found widely in the geostatistics literature. The reader is directed to Cressie (1993), Stein (1999) and Banerjee et al.

[21] Notation differs considerably between the literatures of statistics and machine learning.

[22] http://mucm.aston.ac.uk/MUCM/MUCMToolkit/index.php?page=MetaHomePage.html

(2004) for in depth coverage of the field.

3.4.1 Statistical theory

A GP is defined by its mean, μ, and covariance function, often written as the product of a variance and a correlation function, $\sigma^2 c(\cdot, \cdot)$, that is,

$$\eta(\cdot) \mid \mu, \sigma \sim \mathcal{GP}(\mu, \sigma^2 c(\cdot, \cdot)) \tag{3.8}$$

The mean function of the GP can be zero but that assumption would require that the variation in the data was entirely explained by the covariance function. An effective mean function is a linear combination of basis functions, which often explains a good deal of the variability. For basis functions with only linear terms x, where x is in p dimensions, this is written in vector form $\mathbf{h}(\cdot)^T\beta$, where $\mathbf{h}(\cdot) = [1, x_1, \ldots, x_p]^T$ and β is a vector of unknown coefficients. The mean function could, however, be much more complex where appropriate. A periodic mean function, for example, may be well suited to certain dynamical simulators (see e.g. Rougier et al. 2009a). The covariance function explains the relationship between points in the process. Consider then, that we wish to interpolate, or predict, some quantity at a location in some space. Our prior beliefs are that the value of the quantity at locations close to each other will be closely related, and locations far apart will not inform much about each other. That is, we expect $f(\mathbf{x})$ and $f(\mathbf{x}')$ to be closely correlated when $|\mathbf{x} - \mathbf{x}'|$ is small. Beliefs of stationarity can be expressed by ensuring the covariance depends only on $\mathbf{x} - \mathbf{x}'$, for example. This idea is incorporated into an emulator via the prior specification for the covariance function. Two covariance functions are described below, but the reader is directed to Rasmussen and Williams (2006) for a detailed discussion of covariance functions in general.

If the covariance function is expressed in the form,

$$c(\cdot, \cdot) = \sigma^2 r(\mathbf{x} - \mathbf{x}') \tag{3.9}$$

then one possible, and commonly used form of the correlation function referred to as the Gaussian, or "squared exponential"[23], is expressed as,

$$r(\mathbf{x} - \mathbf{x}') = \exp\left\{ -\sum_{k=1}^{p} \frac{(x_k - x_k')^2}{\delta_k} \right\} \tag{3.10}$$

[23] Although commonly used, strictly speaking this terminology is incorrect since it is the exponential of a negative squared distance.

Here, the sum is over the k dimensions of input and the hyperparameters[24] δ_k are often referred to as the correlation length scales which are related to the sensitivity of the output to change in any given dimension of input. In fact, the sensitivity and degree of nonlinear response of the simulator output to each input can be readily interpreted in terms of the values of the correlation length scales. Sensitivity increases with decreasing value of δ. Where x is scaled to $\{0, 1\}$, values of δ greater than unity suggest linearity of response and could indicate insensitivity to an input. Values of δ less than about 0.05 (i.e. 1/20 of the input range of the given input) indicate very high output sensitivity and may need specific investigation to ensure validity of the correlation function. Equation 3.10 implies belief that the simulator output is smooth, in that the correlation function is infinitely differentiable. This offers mathematical tractability although the smoothness assumption has proven to be inappropriate for many simulators. Alternatives advocated by Stein (1999), Cornford et al. (2002) and Rougier et al. (2009a), are functions from the Matérn class, which, where x is a scalar, is written as,

$$r(x - x') = \frac{1}{\Gamma(\nu)2^{\nu-1}} \left(\sqrt{2\nu} \frac{x - x'}{\delta} \right)^{\nu} K_{\nu} \left(\sqrt{2\nu} \frac{x - x'}{\delta} \right) \tag{3.11}$$

$\Gamma(\cdot)$ is the Gamma function, K_{ν} is a modified Bessel function of the second kind and δ is a length scale parameter. The Matérn equation, is parameterised by ν, providing control over the smoothness. A Gaussian process with Matérn covariance has sample paths that are $\nu - 1$ times differentiable. Where $\nu = 1/2$ the equation reduces to an exponential form. Rasmussen and Williams (2006) suggest that choices of $\nu = 3/2$ and $\nu = 5/2$ give the most interesting functions, providing more roughness than the "squared exponential" form (and removing the complication of the Bessel function). They provide figures of the functional behavior in (their) section 4.2. The Matérn function for $\nu = 3/2$ (where x is a scalar) is given by,

$$r_{\nu=3/2}(x - x') = \left(1 + \frac{\sqrt{3}(x - x')}{\delta} \right) \exp \left(-\frac{\sqrt{3}(x - x')}{\delta} \right) \tag{3.12}$$

where δ is a correlation length scale. At this point the "nugget"[25] term is discussed. The nugget term is commonly added to the covariance function as a way of accounting for uncertainty that is not explained by the regression variables. It appears as

[24] In order to distinguish between input parameters of the simulator and parameters of the statistical model (i.e. the emulator) the term "hyperparameters" is employed. The hyperparameters govern the behaviour of the emulator.

[25] The name originates from geostatistical Kriging used in the mining industry where *nuggets* of ore were particularly important.

a small quantity ϵ added to the diagonal of the observation correlation matrix, or equivalently subtracted from all the non-diagonal entries. For a covariance matrix based upon (3.10) with elements $\sigma^2 r_{i,j}$, where subscripts i, j are over the design points $\mathbf{x}$, and $x_{i,k}$ is the kth dimension of design point $\mathbf{x}_i$, the correlation function with nugget, ϵ, is computed as,

$$r_{i,j}(\mathbf{x}_i - \mathbf{x}_j) = I_{i=j}\epsilon + (1-\epsilon)\exp\left\{-\sum_{k=1}^{p}\frac{(x_{ik}-x_{jk})^2}{\delta_k}\right\} \tag{3.13}$$

where $I_{i=j}$ is the indicator function. The nugget term is discussed, for example, in Banerjee et al. (2004) and Rasmussen and Williams (2006), where the latter present a derivation on the basis of noisy observations. Typically the value of ϵ is small although the specific value is determined on a case-by-case basis and must be proportionate to the total variance. It can also be estimated as an additional hyper-parameter. It may seem unnecessary to make use of a nugget term when conducting a computer experiment, since for a deterministic simulation the output is known precisely. However, Gramacy and Lee (2010) assert that there are a number of good reasons to use one. One practical advantage is that mathematical stability tends to be improved as a result of better conditioning of the covariance matrix. They argue that beyond improved mathematical properties, adding noise to the output reflects that the simulator is fundamentally an approximation, and point out that many modern simulators are stochastic anyway and so the use of a nugget should be automatic. Moreover, the nugget term can also account for inappropriate modelling assumptions such as non-stationarity which again are almost always present to some degree. However, Andrianakis and Challenor (2012b) investigate the properties of the nugget and show that the automatic use of a nugget can produce unexpected results. Specifically, they show that the inclusion of the nugget term can lead to two distinct modes in the Gaussian process likelihood. In effect this leads to two types of behaviour. The first is an emulator that smooths but has a posterior mean function that passes close to the design points, more consistent with an emulator without nugget term. The second is an emulator with a constant posterior variance, and a mean function that is less constrained by the design points. They show that in some cases, and contrary to the belief that the effect of a small nugget may be subtle, even for values as low as 10^{-12}, either behaviour can be induced. This highlights that some caution should be used where use of a small nugget term is automatic. Examples of how a nugget can be used are given in section 3.5.

Given the choice of GP (here linear mean and Gaussian correlation functions can be assumed) the formulation of the emulator can proceed. Description of the working

is given in more detail in Haylock and O'Hagan (1996) and Oakley (1999) but here some important steps and results are summarised. Note also that in the following, the dependence of $c(\cdot,\cdot)$ and the output covariance matrix $\mathbf{A}$ on hyperparameter δ is omitted for clarity. Firstly, $\mathbf{y}^{\mathrm{T}} = [f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n)]$ is the simulator output at input points $\mathbf{X}$ (a $d \times n$ matrix) which yields,

$$\mathbf{y} \mid \beta, \sigma^2 \sim \mathcal{GP}(\mathbf{H}\beta, \sigma^2\mathbf{A}) \tag{3.14}$$

where $\mathbf{H} = [1, \mathbf{X}]$ and $\mathbf{A}$ is the $n \times n$ correlation matrix between the outputs; see below. A likelihood function for the model is generated by conditioning the prior GP model (3.8) on the simulator output resulting in,

$$\eta(\cdot)|\mathbf{y}, \beta, \sigma^2, \delta \sim \mathcal{GP}(m^*(\cdot), \sigma^2 c^*(\cdot,\cdot)) \tag{3.15}$$

where,

$$m^*(\mathbf{x}) = \mathbf{h}(\mathbf{x})^{\mathrm{T}}\beta + \mathbf{t}(\mathbf{x})^{\mathrm{T}}\mathbf{A}^{-1}(\mathbf{y} - \mathbf{H}\beta) \tag{3.16}$$

$$c^*(\mathbf{x}, \mathbf{x}') = c(\mathbf{x}, \mathbf{x}') - \mathbf{t}(\mathbf{x})^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{t}(\mathbf{x}') \tag{3.17}$$

$$\mathbf{t}(\mathbf{x})^{\mathrm{T}} = (c(\mathbf{x}, \mathbf{x}_1), \ldots, c(\mathbf{x}, \mathbf{x}_n)) \tag{3.18}$$

$$\mathbf{A} = \begin{pmatrix} 1 & c(\mathbf{x}_1, \mathbf{x}_2) & \ldots & c(\mathbf{x}_1, \mathbf{x}_n) \\ c(\mathbf{x}_2, \mathbf{x}_1) & 1 & & \vdots \\ \vdots & & \ddots & \\ c(\mathbf{x}_n, \mathbf{x}_1) & \ldots & & 1 \end{pmatrix} \tag{3.19}$$

However, we wish to remove the conditioning on the model hyperparameters, β, σ^2 and δ. In order to do so we first apply Bayes' rule and multiply (3.14) by prior distributions for the hyperparameters. Haylock and O'Hagan (1996) assert that prior information about the hyperparameters when modelling computer simulations is typically weak, and so assign the (improper) uninformative prior,

$$p(\beta, \sigma) \sim 1/\sigma^2. \tag{3.20}$$

O'Hagan and Forster (2004) derive this improper prior from the *normal-inverse-gamma* prior (conjugate to the Gaussian distribution), which can also be used in this analysis. Omitting some steps here, re-arranging (3.14) and multiplying by (3.20), it is found that,

$$\beta \mid \mathbf{y}, \sigma^2 \sim N(\hat{\beta}, \sigma^2(\mathbf{H}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{H})^{-1}) \tag{3.21}$$

where,

$$\hat{\beta} = (\mathbf{H}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{y} \tag{3.22}$$

is the classical weighted least squares estimator[26], and,

$$\sigma^2 \mid \mathbf{y} \sim (n-q-2)\hat{\sigma}^2\chi^{-2}_{n-q} \tag{3.23}$$

where,

$$\hat{\sigma}^2 = \frac{\mathbf{y}^{\mathrm{T}}(\mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{H}(\mathbf{H}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}}\mathbf{A}^{-1})\mathbf{y}}{n-q-2} \tag{3.24}$$

Multiplying (3.21) and (3.15) and integrating with respect to β (which can be done analytically in this case) yields,

$$\eta(\cdot)|\mathbf{y}, \sigma^2 \sim \mathcal{GP}(m^{**}(\cdot), \sigma^2 c^{**}(\cdot,\cdot)) \tag{3.25}$$

where,

$$m^{**}(\mathbf{x}) = \mathbf{h}(\mathbf{x})^{\mathrm{T}}\hat{\beta} + \mathbf{t}(\mathbf{x})^{\mathrm{T}}\mathbf{A}^{-1}(\mathbf{y} - \mathbf{H}\hat{\beta}) \tag{3.26}$$

$$c^{**}(\mathbf{x},\mathbf{x}') = c^{*}(\mathbf{x},\mathbf{x}') + (\mathbf{h}(\mathbf{x})^{\mathrm{T}} - \mathbf{t}(\mathbf{x})^{\mathrm{T}}A^{-1}H)(H^{\mathrm{T}}A^{-1}H)^{-1}(\mathbf{h}(\mathbf{x}')^{\mathrm{T}} - \mathbf{t}(\mathbf{x}')^{\mathrm{T}}A^{-1}H)^{\mathrm{T}} \tag{3.27}$$

The final step of multiplying (3.23) and (3.25) and integrating with respect to σ^2 yields a t-distribution, with $n-q$ degrees of freedom, for the posterior, written as,

$$\frac{\eta(\mathbf{x}) - m^{**}(\mathbf{x})}{\sqrt{\frac{\hat{\sigma}^2 c^{**}(\mathbf{x},\mathbf{x}')}{n-q-2}}} \sim t_{n-q}. \tag{3.28}$$

Recall however that the covariance function $c(\cdot,\cdot)$ contains the hyperparameter δ, but owing to the intractable form of (3.28) it cannot be integrated analytically. A number of options are available, and are discussed by Rasmussen and Williams (2006). One option is to integrate numerically using a Markov Chain Monte Carlo (MCMC) scheme (see e.g. Gilks et al., 1996). This process involves sampling from the full conditional distribution for the posterior of the length scale parameter, requiring the specification of its prior, which may not be forthcoming. MCMC is computationally demanding and can also require some experience. Initial settings, steps sizes, acceptance and rejection criteria all require careful consideration in order for the method to be effectively and efficiently sample from the posterior distribu-

[26] The uninformative prior essentially allows the data to dictate β, however use of an informative prior with respect to β would not recover the classical result. Sacks et al. (1989) discuss the linear predictor from a frequentist perspective.

tion. Banerjee et al. (2004) discusses the use of such techniques in Bayesian inference at length. Another possibility is to make a point estimate of δ and simply plug it into (3.28). This can be very effective but, as noted by Bayarri et al. (2007) in so doing variability is underestimated because account is not taken of uncertainty about δ. When making the point estimate some care has to be taken. Commonly the posterior distribution for δ is maximised, that is its modal value is found using an optimisation algorithm which is the equivalent of maximum likelihood estimation (MLE). However, the shape of the posterior can be very flat and also be multimodal, making it taxing for the optimisation algorithm. Often the log of the posterior is maximised, which can alleviate this problem to some degree. The posterior distribution for δ is found by multiplying the likelihood function for the hyperparameters (3.14) by their prior disitrubtions (3.20), and assigning a uniform prior to δ. The calculation is covered by Oakley (1999) and the result is given here,

$$f(\delta \mid \mathbf{y}) \propto (\hat{\sigma}^2)^{-\frac{n-q}{2}} |\mathbf{A}|^{-\frac{1}{2}} |\mathbf{H}^{\mathrm{T}} \mathbf{A}^{-1} \mathbf{H}|^{-\frac{1}{2}} \tag{3.29}$$

An alternative to the MLE method is a "leave-one-out" method. The principle is that candidate values of δ are chosen through a minimisation algorithm, and a set of n emulators are trained using the simulator output, but with each of the n members removed. Each emulator is then used to make a prediction of simulator output at the point which was omitted from the training set. The sum of the differences between the predictions and the observed values is then minimised by the algorithm, in order to determine the optimal value of δ.

Once an estimate of δ has been made using the preferred method, the emulator is ready for testing. Like all models, statistical or otherwise, the emulator must be validated before it can be put to use. The validation process must assess how closely the emulator represents the simulator and identify any significant conflicts. There is no single definitive measure for this but a range of tests and graphical diagnostics are available. These are reviewed in section 3.4.3. Before proceeding to that section however, we note that little has been said so far about the design used to generate the emulator training data. This is an important subject and is discussed in the next section.

3.4.2 Experiment design

In order to build an emulator observations of the simulator output must be made according to some kind of design. McKay et al. (1979) were among the first to consider experimental design for deterministic computer simulators. Sacks et al.

(1989) also discuss a number of issues relating to design. The design of any scientific experiment should naturally be such as to obtain information in the most efficient way thus when conducting a computer experiment we adopt a design in order to maximise our knowledge of the simulator output from a minimum number of points. Prior knowledge of the simulator may lead to the choice of a specific strategy however generally we wish to ensure even and complete coverage of the input space. We also wish to minimise correlation between variables, that is, ensure as far as possible orthogonality in the design. McKay et al. (1979) examined properties of a random, a stratified and a *Latin Hypercube* design and found that Latin Hypercube sampling (LHS) had these desirable properties. A LHS design is formulated by dividing the design space into regions, and then sampling points randomly from within each region. In a simple two dimensional case for example, 100 design points can be obtained by dividing each dimension into ten equal regions, and then chossing a single point at a random position within each of the 100 resulting regions. LHS based designs have very good space filling qualities, which can be optimised by maximising the minimum pairwise distance between points, giving rise to the so called "maximin" LHS design. For example Williamson et al. (2013) achieved this by employing an optimisation that maximised the harmonic mean of the pairwise inter-point distances. Other advantages are that designs are cheap to produce and they scale up to large numbers of dimensions without issue. The properties of an LHS based design is such that a fairly small numbers of points are required per dimension and this scales linearly, rather than geometrically, with the number of dimensions. Loeppky et al. (2009) determine that the rule-of-thumb guidance of 10 points per dimension is well supported by analysis. Craig et al. (2001), Kennedy et al. (2008) and Lee et al. (2011) are examples where an LHS based design was employed. An example of a two dimensional LHS design with 100 points is shown in the centre panel of figure 3.2. The space filling property of the method is readily apparent. Also shown in panels to the left and right of the centre are the histograms of the distributions of the design points for each input, which can be seen to be uniform illustrating how the input space has been divided equally.

A possible alternative is a design based upon a Sobol sequence (Sobol, 1967). Based upon a quasi-random number sequence, it was originally identified as a means of performing numerical integration more efficiently than simply relying on random numbers. Being sequentially generated, it differs from an LHS approach in that the design can easily be expanded if necessary. This is an attractive feature when conducting computer experiments since the simulator runs are often expensive, and if a design was found to be too small, it is wasteful to have to discard those runs.

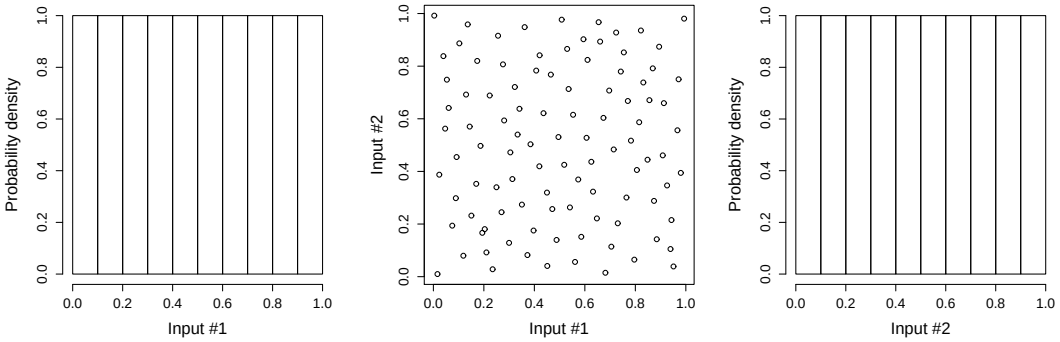


Figure 3.2: An example 100 point maximin LHS based design for two input variables. Panels 1 and 3 show a histogram of the distribution of the sampled points in each dimension. The centre panel shows the points plotted in the design space. The space filling and orthogonal properties are clear.

However Challenor (2013) notes that for small numbers of points Sobol sequences lack the space filling properties of LHS based designs, which is also highlighted by Santner et al. (2003), who provide a comparison of these two approaches to design.

There is a large and growing literature on computer experiment design. Some examples include Mease and Bingham (2006) who propose an approach for designs with unequal cell sizes and Dette and Pepelyshev (2010) who investigate nonuniform designs that place a high proportion of points near the design boundary. Qian (2009) proposes a methodology for nested designs, that can be used in experimental situations where multilevel emulators are employed. This application of emulators is discussed further in section 3.4.5. Nested designs for multilevel experiments bears similarity with the approach proposed by Challenor (2011) who investigates design for simulators that include switches as input. Many examples of modern simulators incorporate switches that turn on and off various features. WW3 is a good examples of this, where a number of different physical parameterisations are available, the selection of which are governed by a number of switches[27]. In general, whether running multilevel experiments or investigating switches, the design must incorporate sub-designs spanning the switched sub-space, but that also retain qualities such as orthogonality.

A final note on this topic is that use of a formal design may be a luxury in some cases. The simulation output may already have been produced by others thus complicating the analysis. Bastos and O'Hagan (2009) faced this kind of problem and sub-divided the data set in a randomised way. In this thesis experimental design

[27]In practice the switched modules are compiled separately leading to a number of different executables but the selection of these could easily be implemented via an external switching system.

is entirely based upon a maximin LHS approach, generated using the "lhs" package[28] for R.

3.4.3 Emulator validation

Section 3.4 concluded by emphasising that an emulator must be validated for use. This is of course a key step in the development of any mathematical model whether numerical or statistical. The literature on validation of emulators is fairly sparse although since an emulator is a statistical model, principles and methods that apply to these are generally applicable to emulators. The approach to validation generally involves making predictions with the model and assessing, using various means, the quality of the prediction. When building an emulator we therefore need to gain confidence that both its predictive capability is good and its statistical properties are a close match to those of the simulator output.

The validation of any model typically proceeds initially by obtaining some kind of validation data according to a methodology or design and then using the model to predict the same data. Performance measures are often based upon the mean square prediction error. Such an approach to model selection is advocated in Santner et al. (2003) for example. However the emulator provides a lot of information beyond a simple point estimate. The posterior covariance can also be analysed in different ways to identify conflicts between the emulator and simulator. Bastos and O'Hagan (2009) propose a number of diagnostic tools based on the analysis of this data, and their work has become something of a "go to" reference for emulator validation. Their approach is discussed in more detail shortly but first a few comments are made about assessing the validity of an emulator. Part of the formulation process of an emulator, as described in section 3.4, is to determine appropriate mean and covariance functions for the prior GP model. A useful exploratory technique is simply to fit a multiple linear regression to the simulator output. Visual diagnostics such as plots of the residuals against input can provide clear insight into the structure of the output data well before any attempt is made to formulate an emulator. It is often clear from such plots, seen for example in the bottom panels of figure 3.4, whether the training data might present difficulties. It should be clear, for example, if the data is exhibiting heteroscedasticity, or whether the multiple regression gives rise to extreme outlying data. This knowledge is not only valuable when choosing mean and correlation functions but also for diagnosing performance later.

Assuming an emulator has been formulated, the approach commonly used, and that is used throughout this thesis, is to run both the simulator and emulator at

[28] http://cran.r-project.org/web/packages/lhs/index.html

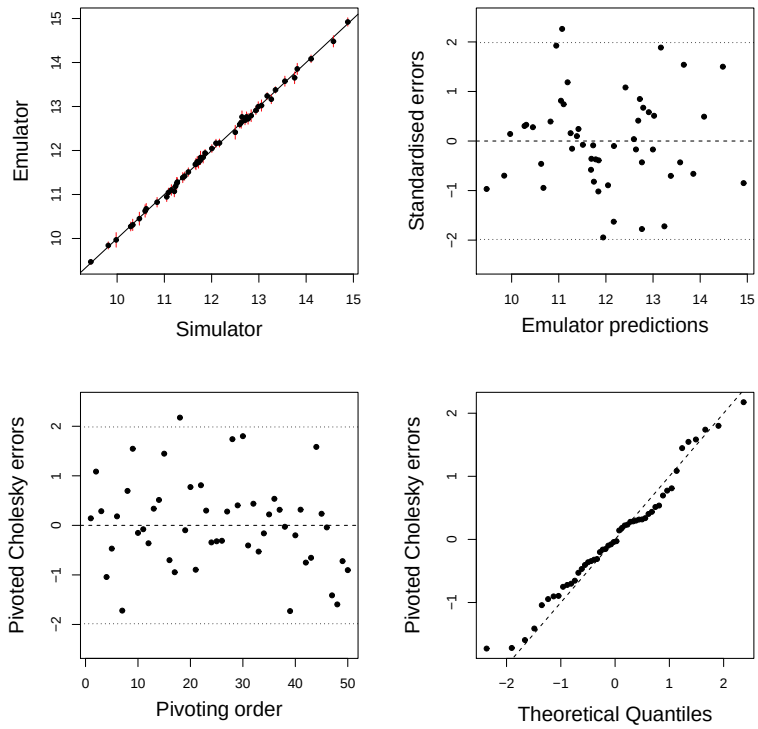


Figure 3.3: Example of graphical tools to assess consistency of the emulator with the simulator. Top left: direct comparison of point predictions. Top right: Standardised errors by point. Bottom left: Standardised errors ordered by size of error. Bottom right: Q-Q plot for errors against reference t-distribution.

a number of validation points. If available, the validation data is usually an LHS design of size n_{val}, where n_{val} is typically at least as large as the original design. The validation process then proceeds by computing a vector of residuals R_{val} (of length n_{val}), generated from the emulator posterior mean prediction and the actual output from simulator. Adopting the notation of Bastos and O'Hagan (2009), $R_{val} = \mathbf{y}^* - E[\eta(\mathbf{X}^*)|\mathbf{y}]$. The error covariance matrix $V[\eta(\mathbf{X}^*)|\mathbf{y}]$ is also computed, by obtaining the posterior covariances for each pair of validation points.

Bastos and O'Hagan (2009) propose a number of methods based upon the analysis of the prediction errors and error covariance, both graphical and numeric, that can reveal conflicts between the emulator and simulator. An example of the output from the graphical diagnostics is shown in figure 3.3. The top left panel is a plot of the emulator posterior mean prediction against the simulator output for each validation point. This gives a very quick indication of the point prediction accuracy. Taking note of the range of values on the axes, it also shows how much variance there is in the output from the validation data. The top right panel shows a plot of standardised error against the prediction. A good emulator will show errors that are uncorrelated to the emulator prediction, and not reveal any extreme errors.

Typically we would not expect to see any more than 5% of validation points lying beyond 2 standard deviations. The bottom left panel is related to the top right, but is based on a decomposition of the error covariance matrix. This is discussed further shortly, but in terms of the graphical diagnosis, again for a good emulator we would expect to see little or no correlation with design point, and no extreme outliers. The final panel is a Q-Q plot comparing the quantiles of the error distribution with the reference t-distribution (see equation (3.28)). Here we need to look for significant deviation from the reference distribution.

Bastos and O'Hagan (2009) also advocate calculation of the Mahalanobis distance. The Mahalanobis distance (Mahalanobis, 1936) is sometimes referred to as a generalised interpoint distance. It is similar to the Euclidean distance D_{Euc},

$$\mathrm{D}_{Euc} = \sqrt{R_{val}^T R_{val}} \tag{3.30}$$

but accounts for the correlation between points via the covariance matrix,

$$\mathrm{D}_{Mahal} = \sqrt{R_{val}^T V^{-1} R_{val}} \tag{3.31}$$

The mean value of the reference F-distribution for D_{Mahal} is equal to the number of validation points. Clearly large residuals (i.e. large values in R_{val}) that do not correspond with appropriate uncertainties in V will increase the value of D_{Mahal}. Conversely if the posterior variance estimates are large with respect to the residuals then D_{Mahal} will be reduced. In each of these two cases the emulator can be regarded as overconfident or underconfident respectively. Note that D_{Mahal} is quite sensitive to extreme single residuals and so is a good test for outliers.

Bastos and O'Hagan (2009) discuss ways of examining the error covariance matrix to identify conflicts. That is, by writing $V = GG^{\mathrm{T}}$, a vector of transformed (standardised) errors is obtained from $G^{-1}R$. In fact, it is clear to see that this is equivalent to decomposing the Mahalanobis distance. There are different ways of decomposing V, two of which are discussed by Bastos and O'Hagan (2009).

One important issue is the interpretation of any decomposition. An eigenvalue decomposition is often used to identify dominant modes, which might for example suggest large variances associated with individual points. However, the eigenvalue decomposition projects the information onto a new basis creating linear combinations of variances, which loses the correspondence with individual validation points. In contrast, the Cholesky decomposition, which computes the lower triangular matrix G, retains variances that are still linked to individual validation points. In this case, the variance in each row is conditioned on the variance for the previous entries.

However, the decomposition is sensitive to the ordering of the validation points, and given that the ordering is (initially) typically random, this confuses any subsequent interpretation. The authors therefore advocate a re-ordering, or pivoting, of the rows (that correspond to the validation points) such that the first row is the one with the largest variance. The decomposition is therefore applied to $P^T V P$, where P is the permutation matrix, and the resulting "pivoted Cholesky errors" can be plotted. An example is displayed in the bottom left panel of figure 3.3. This allows for a more obvious interpretation - any resulting grouping in the validation points can be linked directly to the size of variance, and individual points. The "pivoted Cholesky errors" are what is displayed in the bottom left panel of figure 3.3. In practice, outliers or trends identified by this approach usually correspond closely with the plot of standardised error against the prediction.

The overarching point of the validation is of course to identify conflicts. If outliers or trends that suggest a conflict are identified this could be for a number of reasons which are not always easy to diagnose. Typically, one or more underlying assumptions are being violated, such as non-stationarity or homoscedasticity of the data. Problems are most commonly encountered when the emulator training data set is too small or does not adequately sample the design space. This can lead to poor estimates of the correlation length scale or localised fitting problems. This kind of problem tends to show up very clearly in the diagnostics described above. Poor choices of emulator prior mean and correlation function will often lead to noticeable conflicts, as is demonstrated in section 3.5.

In summary, the tools described are very useful and used widely in this thesis where the general approach is to generate a number of validation data sets according to the same design as used for the training data. However since there is no definitive test for the "perfect" emulator the results from these kinds of tools are inevitably subject to interpretation. Experience is also a valuable asset when formulating and validating an emulator. Once confidence has been gained that an emulator is in good agreement with the simulator it can then be put to use in performing analysis. In the next section we look in more detail at how uncertainty and sensitivity analysis can be performed analytically using an emulator.

3.4.4 Analytical uncertainty and sensitivity analysis using an emulator

In this section an analytical approach to statistical inference with emulators is discussed. An emulator executes very rapidly, even for designs of a 1000 points or more, and so statistical inference can be conducted numerically. Monte Carlo methods can easily be implemented by either using the emulator posterior mean function in place

of the simulator[29] or making draws from the emulator posterior distribution. However, recall that the emulator is a complete statistical representation of the simulator so it is possible, given certain assumptions, to infer statistical information analytically from the emulator posterior mean and covariance functions. An example of how this can be done is given by Oakley and O'Hagan (2004). The inference gives rise to a considerable number of integral expressions, all of which, together with derivations, are given in the MUCM Toolkit[30], with further discussion provided by O'Hagan (2011). Here, a summary is given of the important results, with relevant discussion where appropriate.

Firstly, a few important points are highlighted. In particular, if the approach taken is in accordance with Oakley and O'Hagan (2004), in order to keep the mathematics tractable the emulator has to be formulated with a Gaussian correlation function. The mean function need not be limited to $\mathbf{h}(\mathbf{x}) = [1, \mathbf{x}]$ as is commonly advocated, but some additional calculations are required if more complexity is to be used. Furthermore, weak prior information (as described in section 3.4.1) has to be imposed and the prior uncertainty distributions for the input variables must be multivariate normal.

O'Hagan (2011) derives fourteen integrals, and uses these to express another eight. Many of the details are not duplicated here but some consideration is given to integrals relevant to uncertainty analysis. Recall equation (3.1), and let us proceed to determine the expectation of the distribution induced by the uncertain input $g(\mathbf{x})$. If $f(\cdot)$ is closely approximated by the emulator posterior mean function $m^{**}(\cdot)$ in (3.26) then the expectation of the induced output distribution with respect to the emulator is (using the notation of O'Hagan, 2011, where E^* denotes the expectation of the emulator) given by,

$$E^*\{E(f(\mathbf{x}))\} = R_h^{\mathrm{T}}\hat{\beta} + R_t^{\mathrm{T}}\mathbf{e}$$

where,

$$R_h = \int_{\chi} \mathbf{h}(\mathbf{x})\,\mathrm{d}g(\mathbf{x}) \tag{3.32}$$

$$R_t = \int_{\chi} \mathbf{t}(\mathbf{x})\,\mathrm{d}g(\mathbf{x}) \tag{3.33}$$

$$\mathbf{e} = A^{-1}(\mathbf{y} - H\hat{\beta}) \tag{3.34}$$

[29]The term *simulation* is often used, in the context of Monte Carlo analysis, to mean drawing samples (see e.g. the MUCM Toolkit.) It is avoided here for clarity.

[30]http://mucm.aston.ac.uk/MUCM/MUCMToolkit/index.php?page=MetaHomePage.html

It is also of practical interest to consider,

$$E^*\{Var(f(\mathbf{x}))\} = (I_1 - Var^*[M]) + (I_2 - E^*[M]^2)$$

where (using the same notation) $M = E(f(\mathbf{x}))$ and, with reference to the emulator posterior covariance function (3.27),

$$\begin{aligned}
Var^*[M] &= \int_\chi \int_\chi c^{**}(\mathbf{x}, \mathbf{x}')\, \mathrm{d}g(\mathbf{x})\, \mathrm{d}g(\mathbf{x}') = \hat{\sigma}^2[U - R_t^{\mathrm{T}} A^{-1} R_t + (R_h - G^{\mathrm{T}} R_t)^{\mathrm{T}} W (R_h - G^{\mathrm{T}} R_t)] \\
I_1 &= \int_\chi c^{**}(\mathbf{x}, \mathbf{x})\, \mathrm{d}g(\mathbf{x}) = \hat{\sigma}^2[\tilde{U} - \mathrm{tr}\mathbf{A}^{-1} R_{tt} + \mathrm{tr} W (R_{hh} - 2R_{ht}G + G^{\mathrm{T}} R_{tt} G)] \\
I_2 &= \int_\chi m^{**}(\mathbf{x})^2\, \mathrm{d}g(\mathbf{x}) = \hat{\beta}^{\mathrm{T}} R_{hh} \hat{\beta} + 2\hat{\beta}^{\mathrm{T}} R_{ht}\, \mathbf{e} + \mathbf{e}^{\mathrm{T}} R_{tt} \mathbf{e}
\end{aligned}$$

These expressions introduce the integrals $R_h, R_t, R_{hh}, R_{ht}, R_{tt}, U, \tilde{U}$ which are defined as,

$$\begin{aligned}
R_h &= \int_\chi h(\mathbf{x})\, \mathrm{d}g(\mathbf{x}) & R_t &= \int_\chi t(\mathbf{x})\, \mathrm{d}g(\mathbf{x}) \\
R_{hh} &= \int_\chi h(\mathbf{x})^{\mathrm{T}} h(\mathbf{x}) \mathrm{d}g(\mathbf{x}) & R_{ht} &= \int_\chi h(\mathbf{x}) t(\mathbf{x})^{\mathrm{T}} \mathrm{d}g(\mathbf{x}) \\
R_{tt} &= \int_\chi t(\mathbf{x}) t(\mathbf{x})^{\mathrm{T}} \mathrm{d}g(\mathbf{x}) & U &= \int_\chi \int_\chi c(\mathbf{x}, \mathbf{x}')\, \mathrm{d}g(\mathbf{x})\, \mathrm{d}g(\mathbf{x}') \\
\tilde{U} &= \int_\chi c(\mathbf{x}, \mathbf{x})\, \mathrm{d}g(\mathbf{x})
\end{aligned}$$

Inspection of these integral forms reveals that for suitable choices of mean and correlation functions, and uncertainty distribution $g(\mathbf{x})$, analytical solutions can be obtained. O'Hagan (2011) proceeds to derive such solutions on the basis of choosing linear mean, Gaussian correlation function and a joint normal distribution $g(\mathbf{x})$. It is clear that in such a case the mathematics are tractable. Where a Gaussian correlation function is used and the integrations are with respect to multivariate normal distributions, expressions can be obtained in a form that permits the use of the definite integral,

$$\int_{-\infty}^{\infty} exp\left\{-(ax^2 + bx + c)\right\} \mathrm{d}x = \sqrt{\frac{\pi}{a}}\, exp\left\{\frac{b^2 - 4ac}{4a}\right\}$$

The resulting analytical expressions are extremely fast to compute, and a very powerful tool.

Furthermore the integrals are extended for the purpose of sensitivity analysis as

described in section 3.3. The main effects for input variables w is as follows starting with,

$$E^*\{E(f(\mathbf{x}_{-w}|\mathbf{x}_w))\} = \int_{\chi_{-w}} E^*(f(\mathbf{x}_{-w}|\mathbf{x}_w))\,\mathrm{d}G_{-w|w}(\mathbf{x}_{-w}|\mathbf{x}_w)$$

for some distribution G conditional on $\mathbf{x}_w$. This can be rewritten in terms of two integrals,

$$E^*\{E(f(\mathbf{x}_{-w}|\mathbf{x}_w))\} = R_w(\mathbf{x}_w)\hat{\beta} + T_w(\mathbf{x}_w)\mathbf{e}$$

where

$$R_w(\mathbf{x}_w) = \int_{\chi_{-w}} \mathbf{h}(\mathbf{x})^{\mathrm{T}}\mathrm{d}G_{-w|w}(\mathbf{x}_{-w}|\mathbf{x}_w) \tag{3.35}$$

$$T_w(\mathbf{x}_w) = \int_{\chi_{-w}} \mathbf{t}(\mathbf{x})^{\mathrm{T}}\mathrm{d}G_{-w|w}(\mathbf{x}_{-w}|\mathbf{x}_w) \tag{3.36}$$

If we note that $E^*\{M\} = R_h\hat{\beta} + R_t\mathbf{e}$ where R_h and R_t are unconditional on any input parameters and integrals over the whole parameter space as previously defined, we can use (3.5) to determine the posterior mean for a main effect i,

$$E^*\{z_i(x_i)\} = \{R_i(x_i) - R_h\}\hat{\beta} + \{T_i(x_i) - R_t\}\mathbf{e}$$

Note that this expression is a function of x_i and so can be plotted to visualise the main effect. These expressions can be further developed in order to find the variance attributable to each input, and also extended to $w = [i, j]$ to find the interactions between the two inputs i and j. See the MUCM Toolkit and O'Hagan (2011) for further details. This approach to uncertainty and sensitivity analysis is extremely efficient but unfortunately has some drawbacks.

In practice many authors have noted that the Gaussian correlation function is inappropriate, including Trucano et al. (2006) who reported that for their GP model based upon a Gaussian correlation function the covariance matrix of observations became ill-conditioned very quickly. Unfortunately, a lot of work is required to obtain closed form solutions if an alternative is chosen such as a Matérn function, which exhibits much more stable mathematical properties. A derivation of an analytical solution for the R_t integral, given the choice of a Matérn ($\nu = 3/2$) correlation function and $g(\mathbf{x})$ normally distributed (in one dimension), is shown in appendix C. Although potentially useful, the mathematics is extremely laborious and has to be performed for every variation of correlation function, and is further complicated if the choice of uncertainty distribution is something other than Gaussian.

Nonetheless, by forming a set of integrals from the emulator mean and covariance functions as shown in O'Hagan (2011) we can progress by integrating numerically. It

turns out to be efficient to utilise the built-in numerical integration schemes of software packages, such as R. This allows for fairly arbitrary choices of emulator mean function, correlation function and input uncertainty distributions. Code to perform various inference including the computation of main effects, first order interactions and their resulting sensitivity measures has been developed as part of this research and is given in appendix D, and is utilised throughout this thesis.

3.4.5 Further applications of emulators

Many extensions and modifications of the basic principles of emulation have been reported in the literature. For completeness some of these are mentioned briefly in this section. The reader is also directed to the MUCM Toolkit for extensive discussion of many of these issues.

Complex simulators almost always provide more than one output and in more than one form, and it is desirable to build an emulator that can represent this. Fricker and Oakley (2013) discuss the issues surrounding this problem and provide a comparison of approaches. They refer to simulators producing "field output" where, as the name suggests, output is in some sense on a referenceable grid structure and can be expressed by some kind of covariance structure. Conversely, "multiple-type output" simulators produce various output that includes summary or discrete parameters such as global average surface temperature. Where output is low dimension and (approximately) uncorrelated it can be most straightforward to build separate emulators for each output of interest. However, this is often not the case. Where output is high dimension, it may be desirable to perform dimension reduction. Higdon et al. (2008) describe the application of emulation to an implosion physics simulation where output dimension was of order 10^4. They utilised principal component decomposition that captured approximately 99% of the uncertainty with only the first five components. They do, however, caution that a more chaotic and less forced system may not be so easily represented by so few components. Rougier (2009) and Conti and O'Hagan (2010) give further examples of the development of emulators to tackle complex multivariate simulators.

There are many cases where a simulator can be run at different levels of accuracy. For dynamic geospatial type simulations this is typically a difference in spatial (or temporal) resolution. Low resolution simulations are cheap to run but do not capture the small scale dynamics that are available at higher resolutions. However it has been pointed out by Williamson et al. (2012) out that low resolution simulations often capture underlying trends, much the same as their high resolution counterparts. Here the term "multilevel" emulator is used to designate an emulator conditioned

on information from the same simulation but at different precisions. Often higher precision results may be obtained at a higher computational cost by, for example, running at higher resolution in space or time. It can often be observed that the faster (lower resolution) run approximately captures much of the average or trend in the slower higher resolution output. Kennedy and O'Hagan (2000) employed this property to use fast approximate runs to build emulators of slower more expensive simulators at higher resolution. An accurate emulator is built from many cheap simulator runs that captures the average behaviour and then from much fewer runs of the expensive simulator, a further emulator is built to predict the difference. The large potential gain in computational efficiency makes this approach particularly attractive. A possible design strategy for such an experiment is described by Qian (2009).

Calibration of a simulator is the process of identifying the parameter values that, in some sense, give the best performance. The parameters may be physical quantities that are poorly known or estimated, or could be entirely empirical. The task of calibration is then to find the "best" set of values of the parameters, which is usually determined on the basis of finding the most accurate parameterisation with respect to some kind of observed (real world) data. Trucano et al. (2006) suggests that "*... the purpose of calibration is to adjust a set of code input parameters associated with one or more calculations so that the resulting agreement of the code calculations with a chosen and fixed set of experimental data is maximized.*" They contrast this with a definition of validation: "*... to quantify our confidence in the predictive capability of a code for a given application through comparison of calculations with a set of experimental data.*". The two are closely linked and both require the comparison of simulator output with real world observations. However simulator validation introduces some philosophical considerations and so any direct comparison of simulation output with observations must be treated carefully. Discussion of the many complex issues surrounding the validation and calibration of simulators is beyond the scope of this thesis but Trucano et al. (2006) provide lengthy discussion and relevant references.

Since the approach to emulation involves building a complete statistical model of a simulator, it also lends itself to the incorporation of real world observations, and treating them in a probabilistic way. Kennedy and O'Hagan (2001) present an approach to calibration based upon statistically modelling the (numerical) model discrepancy - the difference between observations and the simulator output. This approach is similar to that of multilevel emulation but there the difference was between simulator output at differing precision. By modelling the difference between

observations and simulation output as an independent GP (the model discrepancy, also referred to as an "inadequacy function") the uncertainty about the tuning parameters can be characterised. This information can then be used to choose optimal values. The model proposed by Kennedy and O'Hagan (2001) is given by,

$$z_i = \zeta(\mathbf{x}_i) + e_i = \rho\eta(\mathbf{x}_i, \boldsymbol{\theta}) + \delta(\mathbf{x}_i) + e_i \tag{3.37}$$

although they note that other choices could easily be argued. Here, z_i is the ith observation of the true process $\zeta(\cdot)$. ρ is a regression term, $\eta(\cdot, \boldsymbol{\theta})$ is the simulator output (i.e. the emulator) where $\boldsymbol{\theta}$ represents the parameters to be calibrated, $\delta(\cdot)$ is the so called inadequacy function and e_i represents any residual uncertainty, assumed Gaussian. We can see that once the observational errors have been accounted for by e_i, the inadequacy function $\delta(\cdot)$ represents the difference between the (emulated) simulator and reality. Appropriate choices of $\boldsymbol{\theta}$ will therefore minimise the sum of $\eta(\cdot, \boldsymbol{\theta})$ and $\delta(\cdot)$, and can be regarded as optimal in some sense. The mathematics involved for a full Bayesian calculation of the various posterior distributions is in general not practical and simplified in various ways in order to find solutions. The authors perform a comparison between the Bayesian approach to calibration (and subsequent prediction) and more traditional methods such as minimising the least squared errors between prediction and data, noting a substantial improvement. Trucano et al. (2006) provides further discussion of both the general concepts of calibration (in the context of code verification and validation) and the approach proposed by Kennedy and O'Hagan (2001). Simulator calibration can be achieved by using an emulator in conjunction with other more traditional methods such as minimisation of a cost function or a genetic algorithm. In spite of the obvious efficiency, examples of calibration using such approaches are not yet abundant in the literature, particularly with respect to complex geospatial simulators. Gregoire et al. (2010) for example make use of a LHS based design for an ensemble of runs in an effort to better calibrate a climate simulator. They do not formulate an emulator but more simply specify a cost function, based upon a variety of important output from the climate simulator, that can be computed for each ensemble. This is undoubtedly an improvement on previous studies cited in the paper, where a sequential one-at-a-time tuning approach was employed but use of an emulator in such a case would be very effective, and allow a much more detailed analysis of the parameter space. Sexton et al. (2012) utilise the emulation methodology of Rougier et al. (2009b) (which is similar to that described in this chapter) and follow a methodology similar to Kennedy and O'Hagan (2001) by estimating the inadequacy function by using output from an ensemble of simulators. By incorpo-

rating observational data Williamson et al. (2013) employ another variation on this approach to explore and characterise the parameter space for a climate simulator, leading to a parameter subspace that gives rise to improved operation. In chapter 7 this type of problem is discussed further, and a calibration of WW3 is performed in a somewhat less complex way, by using fast emulators to minimise a cost function with respect to observations of waves on Lake Michigan.

3.5 A simple example (0-D wave model)

Having covered the relevant theory underlying emulators, an example of an application to a wave simulator is given in this section. This elucidates how an emulator works in practice and highlights some of the steps involved. The analysis performed for more complex wave simulations throughout this thesis is conducted in much the same way as shown here. Firstly the analysis is motivated, and the physical situation is summarised, together with relevant aspects of the simulator configuration. The construction of the emulator then proceeds and once complete, an uncertainty analysis is performed. Following the example, a simple "toy model" is examined to explore issues identified in the example.

The focus in the example is not on the physical simulation but on how the emulator is implemented and utilised. Details of the simulator are therefore limited to those necessary for this purpose. Similar simulator configurations are looked at more formally in chapter 4.

3.5.1 Outline of simulator configuration and experiments

The efficacy of a wave simulator is tested by its ability to generate the correct sea state in various conditions. In "fully developed" wave conditions in deep water (see e.g. Holthuijsen, 2007), this is equivalent to setting the left-hand-side of (2.7) to zero, giving,

$$S_{\text{wind}} + S_{\text{nl}} + S_{\text{ds}} = 0 \tag{3.38}$$

This states that energy input from the wind is balanced locally by nonlinear interactions and dissipation. The sea state is described fully by the directional spectra however in this simple case the wind is constant in time and direction the spectrum is more conveniently described by wave summary statistics alone. In this experiment the peak period (T_p) of the spectrum is examined. Recall that the terms on the left-hand-side of (3.38) allow some control over the energy balance and shape of the spectrum through tuning parameters. These were discussed in chapter 2 and are explored further in chapter 4. There is uncertainty about the "best" choices

Table 3.1: Sources of uncertainty (0-D example simulation).

Source term	**Sources of uncertainty**	**Category**	**Distribution**
S_{wind}	Wind speed: May take any constant value. Stable or unstable flow.	Observation	Wind speed $\sim N(10.0, 0.5^2)$
S_{nl}	Tuning of DIA source function.	Parameter	N/A
S_{ds}	TC96 dissipation source function Various empirical tuning parameters. (*sdsa0* and *sdsa2* selected)	Parameter	*sdsa0* $\sim N(4.8, 0.2^2)$ *sdsa2* $\sim N(2.0, 0.2^2)$

of these parameters and so they can be regarded as random variables that can be investigated using an uncertainty analysis. By representing our uncertainty about them we can generate probabilistic output which tells us about the distribution of output, and how this is affected by any nonlinearity. It may also be possible to reconcile findings with understanding of the simulator and how we expect the relevant physics to operate (i.e. do we expect to see any nonlinearity?).

The experimental approach here is that the wind speed, together with two tuning parameters from the TC96 parameterisation are regarded as sources of uncertainty. Uncertainty about these inputs is expressed in the form of probability distributions. The distribution of wind speed is assumed to be normal with a mean value of 10ms^{-1} and standard deviation of 0.5ms^{-1}. Two dissipation tuning parameters, *sdsa0* and *sdsa2*, that govern the behaviour of S_{ds} are studied and these are also regarded as normally distributed. Uncertainty ranges for these were determined on the basis of model tuning as discussed in Tolman and Chalikov (1996). The source of input uncertainty and their specification are summarised in table 3.1. (These are discussed in more detail in chapter 4). This experiment does not require energy propagation in the simulator, which in effect is treating the geographic domain as an infinite sea. As a result only a single grid cell is required.

Following section 3.4.2 a maximin LHS design covering the 3-dimensional parameter space is adopted. Six 50 point designs were generated although considering the analysis of Loeppky et al. (2009) who suggest designs should be at least 10 points per dimension, this may be excessive. Here we are interested in the steady state of the simulator, according to (3.38), so it was run until the sea state reached equilibrium. In practice this requires approximately 30 days of simulation time. Details of the

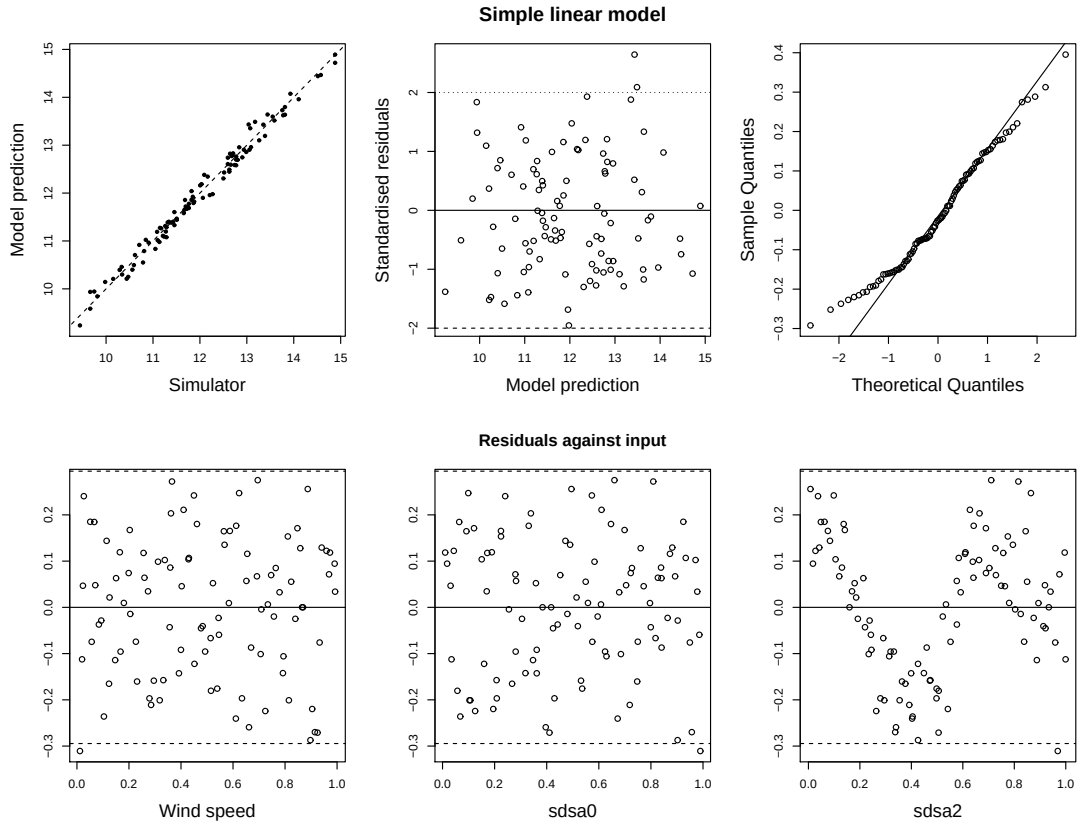


Figure 3.4: Diagnostics for the simple linear regression model.

work flow for generating and running an ensemble for WW3 are given in appendix B. Once run, we proceed by exploring the relationship between the output (T_p) and the uncertain input parameters using simple regression modelling.

3.5.2 Linear regression

To emphasise the principle of emulation, a simple regression model can be implemented with ease, and from which much can be learned about the data. Note that a linear regression model differs from an emulator in that it assumes that the residuals from the linear relationship are independent and it ignores correlation between the design points. A linear model is fitted to the first three designs (150 points) and we see some general diagnostics in figure 3.4. The top panels show that a simple linear model appears to do a reasonable job of explaining the trend in the data. However, inspection of the residuals plotted against each of the inputs (bottom panels) shows very high correlation between the output and *sdsa2*. A model with a cubic relationship to that input would capture more of the trend. In fact, terms up to fifth order were found to explain the variance best. A regression function of the form $T_p \sim 1 + x_{[1,2,3]} + x_3^2 + x_3^3 + x_3^4 + x_3^5$ was fitted for comparison and the same graphical

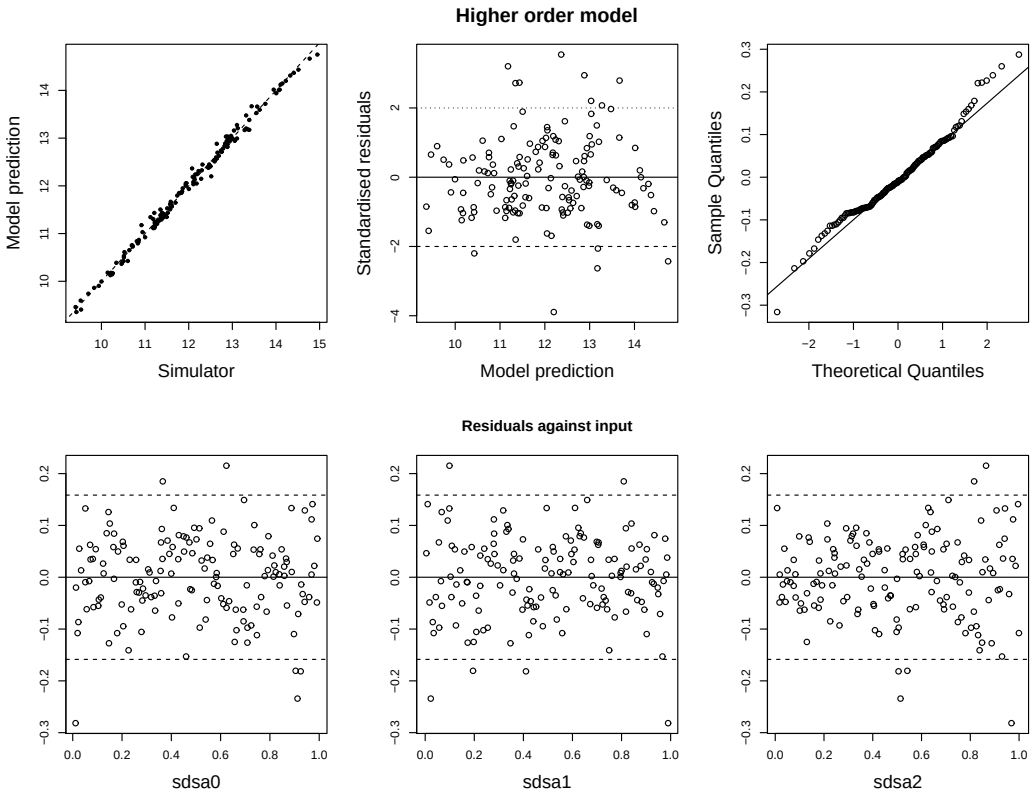


Figure 3.5: Diagnostics for the cubic regression model.

diagnostics are shown in figure 3.5. The higher order model reduces the residual standard error from 0.141 to 0.082. Initially however, for the purpose of comparison, we proceed to construct an emulator with a simple linear mean function of the form $T_p \sim 1 + x_{[1,2,3]}$. In the absence of any prior information about the smoothness of the simulator the Gaussian correlation function is adopted.

3.5.3 Emulator construction and validation

Following section 3.4 and using the output from the first designed experiment we proceed to train the emulator and determine the length scale parameters. This is done by taking the maximum from the posterior distribution for δ, and plugging it into expressions (3.26) and (3.27). Formulation of the posterior requires the specification of prior information for δ but in the absence of strong prior information a uniform non-informative improper probability distribution can be assumed. The uniform prior is a constant term and so does not appear in (3.29) and does not affect the optimisation. The logarithm of (3.29) is thus,

$$\log(f(\delta \mid \mathbf{y})) \propto -\frac{1}{2}\left\{(n-q)\log(\hat{\sigma}^2) + \log|\mathbf{A}| + \log|\mathbf{H}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{H}|\right\} \tag{3.39}$$

The maximisation routine found a number of modes, the maximum of which corresponded to values of δ of (0.587,1.011,0.0275) Following section 3.4.3 performance of the emulator was assessed by computing the Mahalanobis distance and use of a range of graphical diagnostic tools. The second design set of output data was used for validation, and the results are shown in figure 3.6. The observed value of the Mahalanobis distance was 440, which is clearly many standard deviations from the reference distribution with mean and s.d. of 50 and 14.6 respectively. Reviewing the panels from top left to bottom right in figure 3.6 we see that the absolute predictions of the simulator are quite good (noticeably better than the simple linear regression) and therefore have confidence we are on the right track. Secondly we see that a number of points are predicted poorly, but also have very low variances estimates and appear as outliers in the top right and bottom left panels. Thirdly, the pivoted Cholesky errors suggest that the variance estimate is reasonable but there may be some local fitting problems. Finally, the mismatch in the Q-Q plot shows that the emulator is underestimating the variance in the "positive" tail. There is however, no clear evidence of error correlation, or obvious signs of clustering or patterns that might suggest some fundamental problem with our underlying assumptions. Given that we already saw the correlation between T_p and *sdsa2* in the linear regression modelling, it is likely that the GP is struggling to capture that behaviour. The

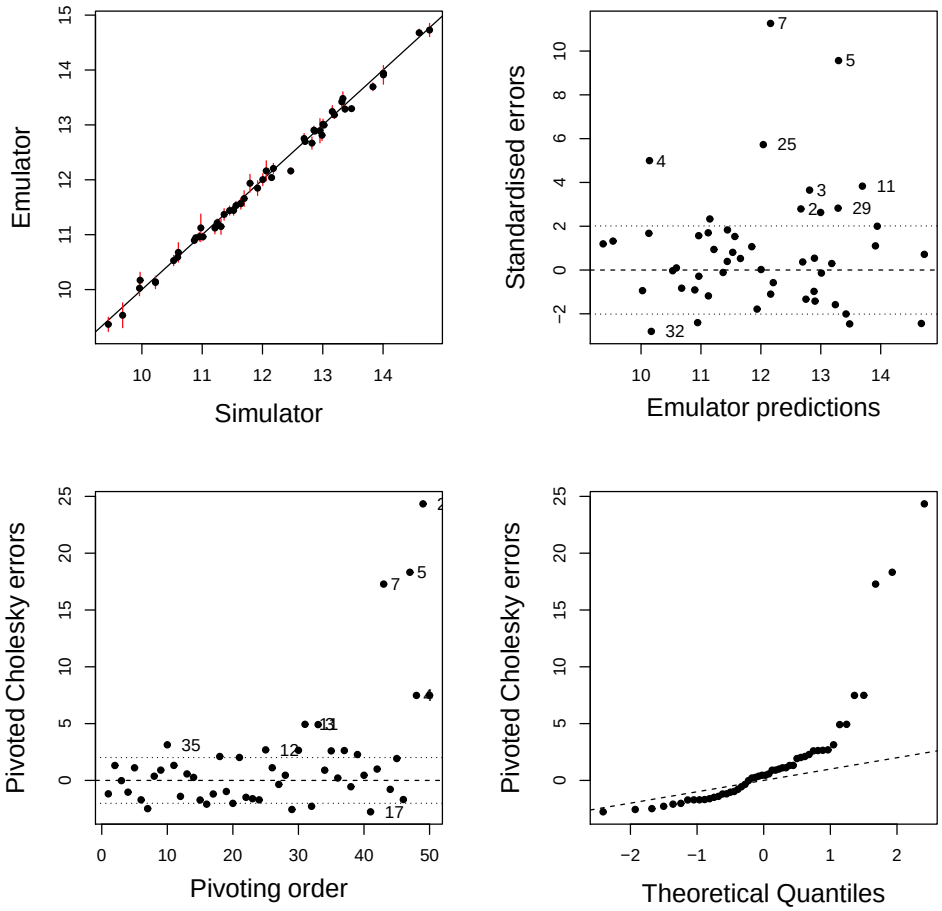


Figure 3.6: Graphical tools to assess consistency of the emulator with the simulator. Top left: direct comparison of point predictions. Top right: Standardised errors by point. Bottom left: Standardised errors ordered by size of error. Bottom right: Q-Q plot for errors against reference t-distribution.

estimate of the correlation lengths may be inappropriate at this point.

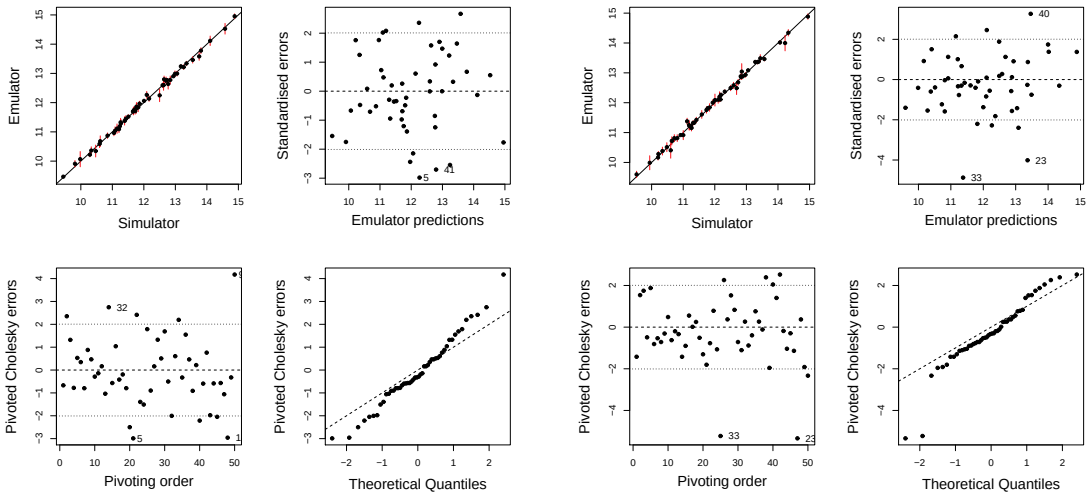
So we proceed on the basis that our beliefs about the structure of the simulator are reasonable and that additional data points will correct possible localised fitting issues and the emulator's overconfidence. Half the validation data set was thus added to the emulator training data and the MLE of δ repeated. δ is now determined to be (0.0609, 1.9590, 0.1083), which differs substantially from the previous estimate.

Note that, depending on the structure of the posterior distribution, and the type of optimisation algorithm used to determine the length scale parameters, it is quite common to find multiple modes. This can be particularly problematic where design points are sparse. In some cases the posterior might be very flat, thus making it difficult for the optimisation algorithm to find a consistent maxima. There is no single way to mitigate this problem so at the very least it is advisable to run the optimisation algorithm using a number of randomised initialisation points. The commonly used "optim" command in R[31] provides different algorithms and it may be advantageous to try more than one. Also, for lower dimensional cases, it is possible to visualise the posterior to explore any problems. If inconsistent results persist, it may be that one or more modelling assumptions are inappropriate.

In the context of this issue, bear in mind that the optimisation approach ignores uncertainty about the correlation length scales. Recall that a uniform prior is specified for the length scale (see section 3.4.1), whereas if we had stronger prior information it would be possible to incorporate this, which may alleviate the issue of selecting inappropriate choices of values due to poor optimisation performance. The added complexity would, however, necessitate the use of an MCMC method to fully characterise the emulator posterior distribution.

Figure 3.7 (a) shows that the new emulator is much improved however, figure 3.7 (b) reveals a small number of "extreme" outliers (points 23 and 33) that are not consistent with normally distributed errors. Incorporating more design points at this stage could lead to an improved estimate of the correlation length scales, however with presence of a small number of extreme outlying points suggests both a poor length scale estimation and some localised fitting problems. Another important point here is that we have not used a nugget term in the training data covariance matrix. This would probably be useful here, but we also have other avenues to explore, specifically that we already suspect a prior mean function with higher order terms will explain more variance. Also, we note that the Gaussian correlation function is not always appropriate so rather than persist with this approach it may be more advantageous to modify the model. As noted previously (e.g. Rougier et al. 2009a)

[31] http://www.r-project.org/



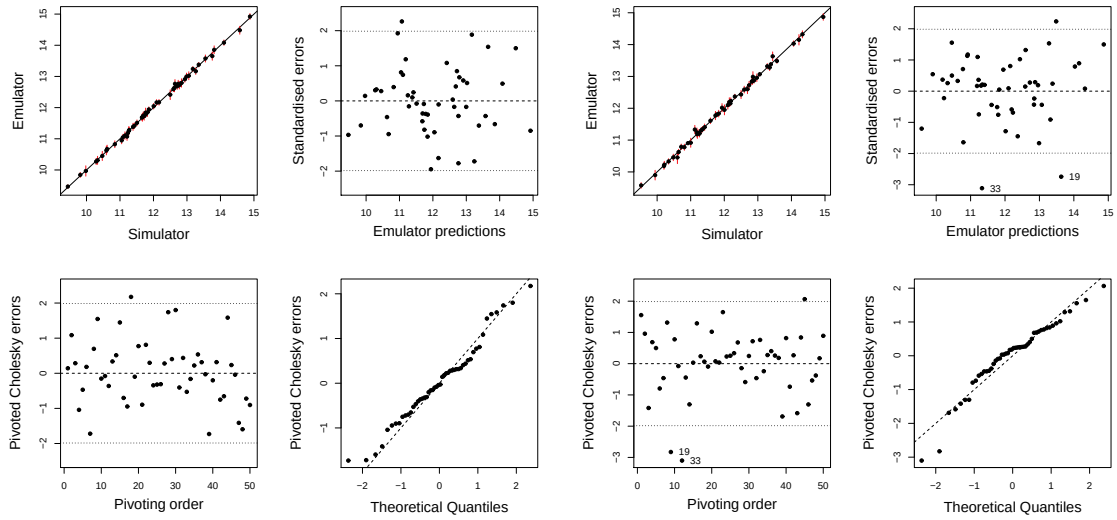
(a) Validation set #3 **(b)** Validation set #4

Figure 3.7: Further emulator diagnostics.

the Gaussian correlation function may be too smooth for many simulators and thus an alternative such as a function from the Matérn family may be better. Therefore the model is modified by utilising the Matérn correlation function (ν=3/2), and incorporating the higher order mean function in the prior model. Designs 1 and 2 were retained for the training data and results of the validation studies are shown in figure 3.8. The Mahalanobis distances for validation against set 3 and 4 were 39.8 and 51.6 respectively, within 1 s.d. of the mean of the reference distribution, suggesting good agreement between emulator and simulator. The diagnostics also show superior statistical properties to the previous emulator. No nugget term was used in this particular case and there does not seem any advantage in doing so. At this point we judge the emulator to be robust and fit-for-purpose. In the next section an uncertainty analysis using the emulator is performed.

3.5.4 Uncertainty Analysis

Once satisfied that the emulator is a robust approximation of the simulator we can proceed to perform analysis. A powerful feature of the emulator is that once built and validated it can be used for numerous statistical analyses. This contrasts with a standard Monte Carlo approach whereby it is not so easy to reuse the model runs. Using this emulator an uncertainty analysis is performed. Recall that the expectation of the simulator output given the uncertain input was given by (3.1). One way of computing this integral using the emulator is, in fact, a Monte Carlo approach. This can be done since the emulator mean function can be run extremely



(a) Validation set #3 **(b)** Validation set #4

Figure 3.8: Emulator diagnostics. Top left: direct comparison of point predictions. Top right: Standardised errors by point. Bottom left: Standardised errors ordered by size of error. Bottom right: Q-Q plot for errors against reference t-distribution.

quickly. So 10,000 samples are drawn from the (joint) distribution for the simulator inputs (described at the beginning of section 3.5) and run on the emulator. The output, in the form of a histogram is shown in figure 3.9. So utilising the emulator we have been able to obtain probabilistic output from the wave simulator, WW3. This output shows the prediction of T_p from WW3 in the case of a "fully developed" sea state in deep water when accounting for uncertainty about two tuning parameters *sdsa0*, *sdsa2* and forcing wind speed. There are a number of features of the histogram to be discussed. Firstly the total variance is important. We can see that there is no appreciable probability of values of T_p below 10.5s or above 13.5s. However, between ± 2 s.d. there is approximately 2 seconds of variation. In reality this kind of difference would unlikely be dramatic in terms of safety at sea or at the coast, however it is nonetheless an appreciable percentage of the mean prediction and shows that our uncertainty is not trivial. The distribution itself is approximately normal, but notice that near the peak there is clearly a heavier weighting of probability towards lower values. In spite of this, the distribution mean lies slightly higher than the simulated value at the mean input. The discrepancy is a measure of nonlinearity in the simulator. In highly nonlinear systems the distribution mean might lie far from the output generated from the input mean, which could be problematic if linearity was being assumed. Here the discrepancy is small, and the approximate normal shape of the distribution suggests the response is close to linear. There is however clearly a degree of nonlinearity giving rise to skewed shape near the peak.

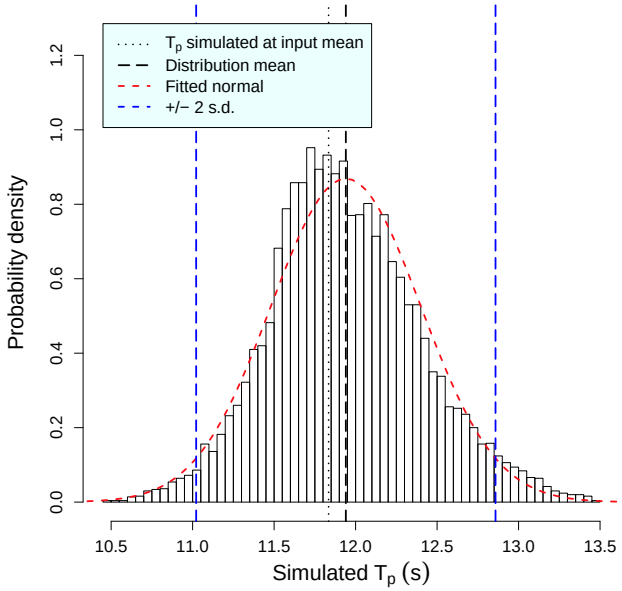


Figure 3.9: Results from uncertainty analysis.

The impact of these findings would depend upon the specific application. In terms of the performance of a wave simulator, it is perhaps unsurprising that behaviour is quite linear in this simple case. It would be of interest to determine the relative importance of the different input here, particularly whether the wind input is more dominant. This is explored further in chapter 4 and later in this thesis.

Before we leave this chapter and move on to perform uncertainty analysis for more complex configurations of WW3, a brief digression is made to consider the issue of poor emulator fitting seen in figure 3.7. This is done by means of "toy model", or in this case a simple deterministic function with a single input.

3.5.5 Comparison with toy model

It is often instructive to use a much simpler experiment to investigate specific phenomenon. The emulator diagnostic tools shown in figure 3.7 (b) revealed two points for which the posterior variance estimates were very small, such that the mean predictions lay beyond 4 s.d. from the validation data. A possible explanation for this is non-stationarity in the data and a simplistic "toy model" is now used to show how it could give rise to such fitting problems. An example of a deterministic function with a single input is shown in figure 3.10. The output, given by $F(x) = 5.5x + 3.15\cos(4.5x) + 0.4\exp\left(-(\frac{x-0.5}{0.2})^2\right)\sin(50x)$, exhibits non-stationarity in the sense that there is a region of shorter scale variation in an otherwise smoothly

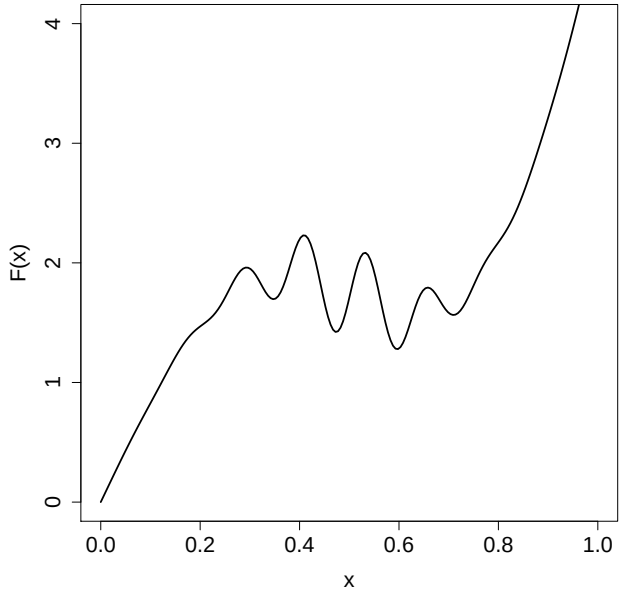
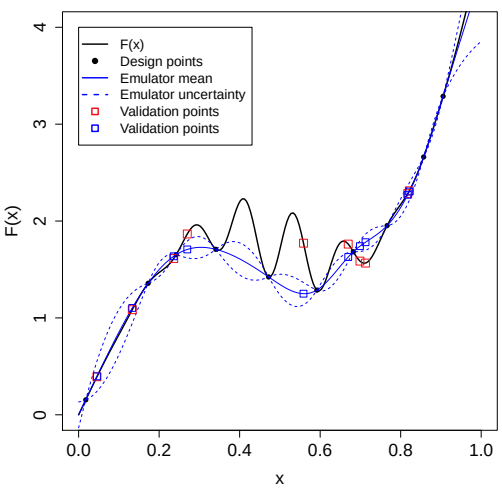
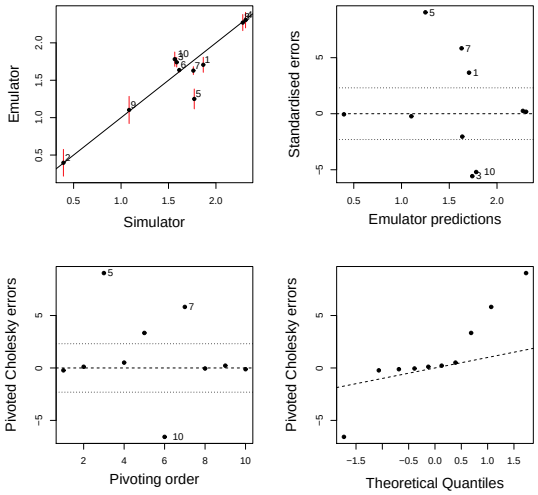


Figure 3.10: Output from a 1-dimensional deterministic function that exhibits localised "non-stationarity".

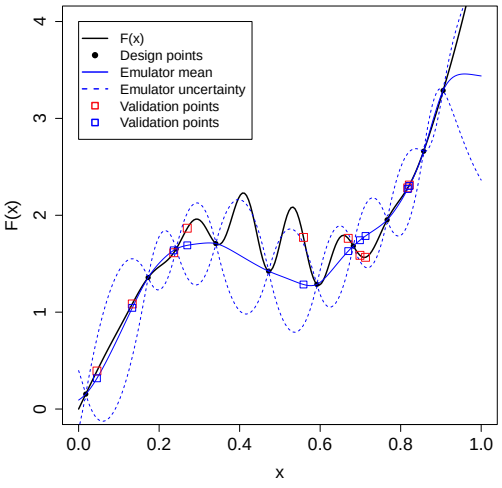
varying function. Another way of describing it, in terms of an emulator covariance function, is that the correlation length scale is a function of the input rather than constant. A series of emulators are built using the output from this function, and consideration is given to the specifics of the design and how it can affect the emulator's performance. A simple linear prior mean function is adopted, with a matern ($\nu = 3/2$) covariance function. The design for the first emulator comprising 10 points, shown in figure 3.11 (a) is space filling, but evidently leads to a poor emulator since the design points capture only the general trend, missing the shorter scale variation. The MLE of the correlation length scale is 1.9, which is clearly inappropriate. The graphical validation diagnostics for this design, shown in figure 3.11 (b), highlight the poor resulting variance estimates. However, if we set the value of δ to 0.10 and re-validate, we obtain a much better emulator, as indicated by figures 3.11 (c) & (d). Furthermore, the Mahalanobis distance has been reduced from 170 in the first instance to 10.8. Noting that for half of the validation points the emulator mean function prediction is actually very close to the simulator but with large variance, we might seek to improve the design to obtain a better localised fit, and determine a more accurate correlation length scale. At this stage increasing the number of design points will improve the emulator mean prediction, and is likely to lead to a better estimation of correlation length scale. An important point to emphasise however is that non-stationary data exhibits a non-constant correla-



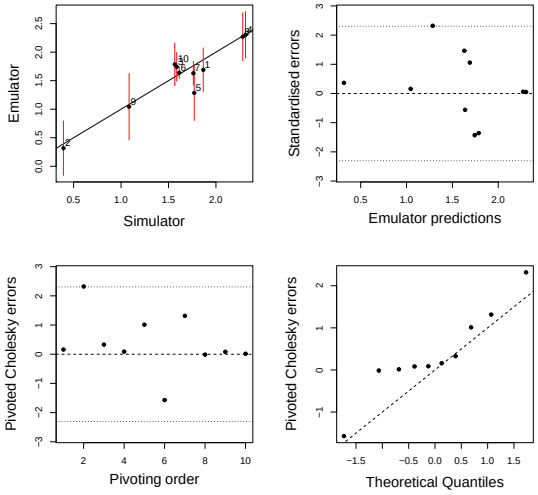
(a) Emulator 1a: (n=10), $\delta = 1.9$



(b) Emulator 1a diagnostics



(c) Emulator 1b: $\delta = 0.10$

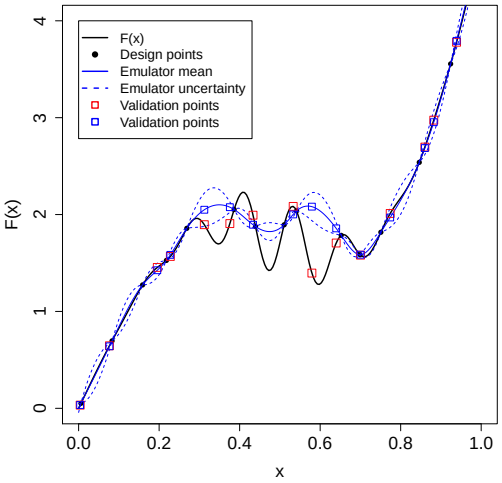


(d) Emulator 1b diagnostics

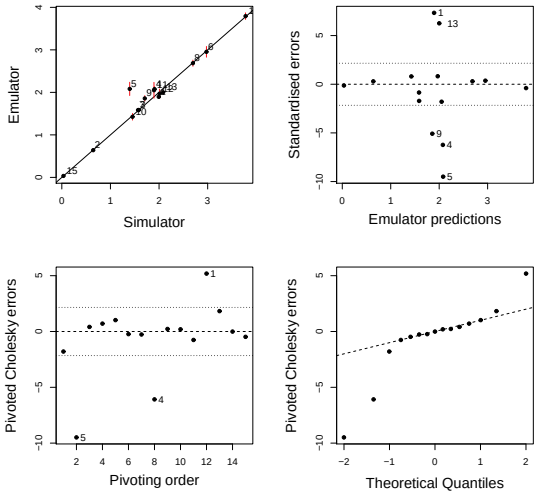
Figure 3.11: Evaluation of different emulators for a toy model.

tion length scale and so there is no "right" answer. Additional design points may lead to a *better* length scale estimate, but it can be highly dependent upon where the design points lie. The new design points may capture the shorter length scale which may bring improvement, but for very short length scales could also cause the posterior variance to inflate drastically away from design points, which is generally inappropriate for the majority of the training data. The emulator would then become underconfident. Conversely, the additional design points may not lie in the shorter length scale region and the estimate may remain unaffected. In this case, the posterior uncertainty will not inflate enough away from design points and the consequence for the emulator is that the posterior variance for predicted points in the shorter length scale region will be massively underestimated, yielding "extreme" outliers. Figures 3.12 (a) & (b) illustrate this. Here 15 design points are used (an increase of 50%) but the MLE of the correlation length was still 0.80, considerably larger than expected given complete knowledge of the simulator. Just as with the example in section 3.5.3 (figure 3.7) the validation diagnostics reveal extreme outliers, indicating the emulator's overconfidence. The point here is that simply adding more design points (where possible, and it may not be if the simulator is unavailable) should be done with some consideration and may not lead to improvement. Indeed, it could be at some performance costs, as very large designs will lead to large covariance matrices that can become ill-conditioned (although this is not an issue in such a simple example).

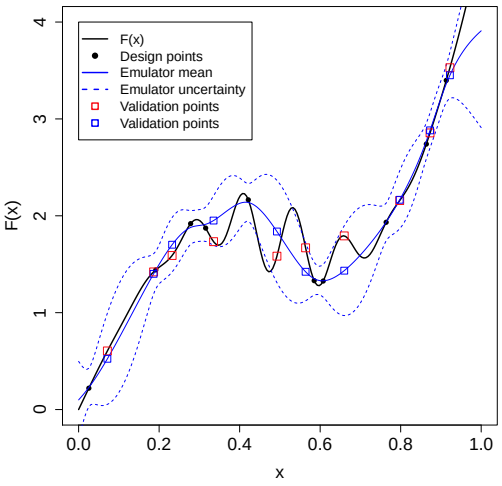
To illustrate the point further, a final emulator design shown in figure 3.12 (c) has only 10 design points but uses a nugget term and has the value of δ set to 0.19. It does not quite capture the fine structure of the nonstationary region but has more robust statistical properties, as shown by the graphical diagnostics in figure 3.12 (d). The observed Mahalanobis distance is 8.6, which lies very close to the reference value of 10.0. In effect, the nugget term has accounted for the uncertainty in correlation length scale and the invalidity of the non-stationarity assumption. A final observation here on this subject is that, in a sense, where the training data do not quite meet the underlying assumptions the number of design points and nugget term have a kind of opposing relationship. That is, a sparsity of design points leads to larger variance in many regions since nothing has been learned about it. As design points are added, the variance is typically reduced because we have better knowledge of the whole region. However, adding a nugget term and relaxing the requirement for the prediction to pass through each design point, i.e. we have uncertainty about the *exact* location or value, is akin to asserting that although we have a design point in a certain location, we do not have such a strong belief in it - which is somewhat



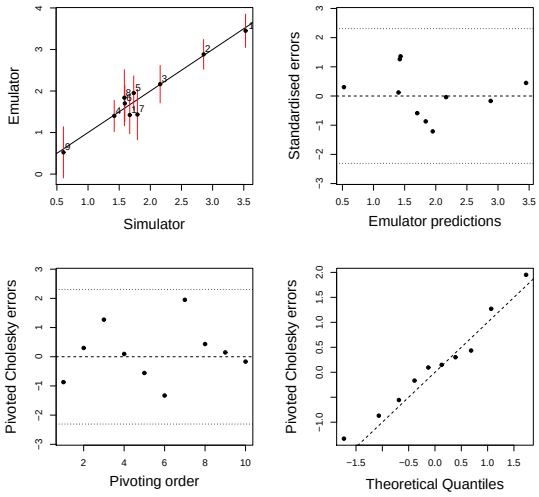
(a) Emulator 2: $\delta = 0.80$



(b) Emulator 2 diagnostics



(c) Emulator 3: $\delta = 0.19$



(d) Emulator 3 diagnostics

Figure 3.12: Evaluation of different emulators for a toy model. Emulator 3 includes a nugget term.

like saying we have a lower density of design points.

Other emulators based upon different statistical models can deal with this type of problem in difference ways, for example a nonstationary covariance function could be employed. Wang and Neal (2012) and Yin et al. (2011) both propose modified kriging type models to appropriately handle inhomogeneous data. Further discussion of this is not made here but this example highlights a type of issue that can be encountered when building a statistical emulator.

3.6 Chapter summary

In this chapter the details of a range of aspects of the creation and application of Gaussian process emulators has been reviewed. The methods and approach described are used extensively throughout this thesis in order to investigate the effect of uncertainty in the tuning of a wave simulator and in its wind input. The approach used generally follows that described in section 3.5. In the next chapter these methods are applied to a range of simple configurations of WW3, with a focus on duration limited wave conditions in deep water, with consideration of how uncertainty in wave conditions could affect predicted wave power extraction from a device.

4 Uncertainty analysis for duration-limited waves in a single grid cell (0-D)

4.1 Introduction

The investigation of uncertainty in WW3 begins in this chapter by considering simple simulations of waves with the purpose of analysing the effects of input parameter and forcing wind uncertainty. During development wave simulators are validated and tuned against simple physical cases for which observational data is available. Consideration of such cases is therefore a sensible starting point and is also a means of gaining confidence in the analytical methods without introducing too much complexity that might unnecessarily obfuscate findings and understanding. Waves generated in duration-limited conditions are an example of such a case, which is fundamental in assessing whether the source function of a simulator can correctly reproduce realistic sea states. The SWAMP (1985) study (see their chapter 7), ran simulations to generate both duration- and fetch-limited wave growth in order to compare the performance of a range of first and second generation spectral simulators. Komen et al. (1994) performed investigations in a similar way for the purpose of validating the WAM simulator. In this chapter a similar experimental approach is taken in that computer experiments are conducted for simulations of waves that recreate duration-limited conditions, but with the addition of performing quantitative uncertainty analysis.

Spectral wave simulators operate on the basis of computing the local wave directional frequency spectrum as governed by the energy balance equation (2.7). Recall from section 2.4.1 that in the case of duration-limited conditions in deep water the energy balance equation reduces to,

$$\frac{\partial E(f,\theta)}{\partial t} = S_{\text{wind}} + S_{\text{nl}} + S_{\text{ds}}$$

In terms of the physics, the situation is equivalent to an infinite homogeneous sea surface such that energy propagating from a region is equalled by the energy propagated into the region. In terms of the geographic arrangement, a single grid cell be can used, and wind is simply specified by a speed in a constant direction. In this chapter fully developed conditions are considered, which is expressed by setting the left hand side of (4.1) to zero. In order that this is satisfied, the simulation is run until the rate of change of the resulting spectrum, in terms of H_s and T_p, is less than 0.1% per time step.

There is uncertainty about the form of each of the source terms in (4.1). The

specific details of each, and the elicitation of the uncertainty are discussed in section 4.2. Table 3.1 identifies the categories of uncertainty that are investigated in this chapter and these are summarised again here; S_{wind} requires a wind specification, in terms of a magnitude and direction, that is typically subject to observational uncertainty; S_{nl} is typically an approximation to the well understood physical process of nonlinear wave-wave interactions but requires tuning via parameters; S_{ds} is typically an empirical formulation of the poorly understood physical process of wave energy dissipation. This chapter does not address structural uncertainty, and so direct comparison with real world observations is not made. Structural uncertainty is discussed further in chapter 7.

The investigation of uncertainty in this chapter is limited to consideration of tuning parameters associated with the TC96 dissipation scheme and the DIA nonlinear wave-wave interaction scheme. In the final experiment uncertainty about wind speed is also introduced. The effect of this uncertainty is analysed with respect to wave summary statistics for wave height and period. The structure of the chapter is as follows. In section 4.2 the relevant sources of uncertainty are discussed in detail and elicitation is conducted. Experiments are summarised and described in sections 4.3, 4.4, 4.5 and 4.6. Although the experiments in this chapter are fairly simple in their configuration, they can be extended to situations of relevance to the real world, which has been done in section 4.5 where a simple uncertainty analysis is performed for the case of wave power extraction by a wave energy converter (WEC). Section 4.7 briefly examines how emulators can be used for the purpose of visualisation before the chapter concludes with a summary of important results in section 4.8.

4.2 Sources of uncertainty and elicitation

The experiments conducted in this chapter investigate parameter uncertainty associated with the parameterisations for energy dissipation (TC96) and the nonlinear wave-wave interaction (DIA) terms. Additionally, in experiment 4.B, forcing wind speed is also regarded as uncertain. The uncertainty associated with parameters and its elicitation is discussed in more detail in the following subsections.

As discussed in chapter 2 most source term parameterisations include a number of uncertain tuning parameters. However, elicitation of their associated uncertainty is made more complicated since often the literature (and indeed the authors themselves!) reveals little about the extent of the sensitivity analysis conducted or the method employed to obtain the tuning. Anecdotally, the methods used are informal and often carried out by hand. The problem is exacerbated by the fact that parameterisations often involve numerical parameters that have little direct relevance to real

world physics and as such can neither be measured directly nor inferred from physical theory. Their uncertainty distribution must typically be elicited by considering the fit to observational data. As has been mentioned, this situation is not limited to the field of wave modelling. Smith et al. (2008) made updates and improvements to the FAMOUS climate simulator alleviating some systematic errors, but evidently used only an ad-hoc approach taking no account of the effects of interaction of their modifications.

4.2.1 Elicitation for S_{ds}

The widely used TC96 input and dissipation scheme (Tolman and Chalikov, 1996; Tolman, 2009a) is implemented in WW3. At the time of its development, much of the physics of energy dissipation was (as it largely remains today), poorly understood. However, it is something of a testament to its effectiveness that it has until very recently been the "default" scheme for the release version of WW3. Tolman and Chalikov (1996) took the approach of defining governing equations for energy dissipation in low and high frequency regimes of the wave spectrum. The dissipation processes are represented empirically, and are tuned to fit observational data. The tuning process is carried out by the authors, though the implementation of the scheme within the WW3 code permits the user to enter their own choice of values for a number of tuning parameters. These are considered in more detail.

The authors express S_{ds} as the sum of $\mathcal{S}_{ds,l}$ and $\mathcal{S}_{ds,h}$ thus,

$$S_{ds} = \mathcal{S}_{ds,l}(f, \theta; b_0, b_1, \phi_{min}) + \mathcal{S}_{ds,h}(f, \theta; a_0, a_1, a_2) \tag{4.1}$$

The detailed formulation of the terms $\mathcal{S}_{ds,l}$ and $\mathcal{S}_{ds,h}$, which is quite convoluted, is given in Tolman and Chalikov (1996) and Tolman (2009a). It is important to note however that the terms are parameterised as functions of six parameters a_0, a_1, a_2, b_0, b_1 and ϕ_{min}, which govern how the scheme dissipates energy. Specific values of these parameters were determined by fitting the empirical formulation to observed data for fetch-limited wave growth under conditions of stable and unstable wind stratification. The tuning process, as described in Tolman and Chalikov (1996), is quite lengthy and not reproduced here in its entirety for the sake of clarity. Essentially it proceeds in a sequential manner, beginning with treatment of the parameters associated with high frequency dissipation. For each parameter the choice of value is made on the basis of some criteria, but the process does not account for the way the parameters might interact together. Figure 4.1 (a) and (b) show figures 4 and 5 from the original work to give some insight into the process. The

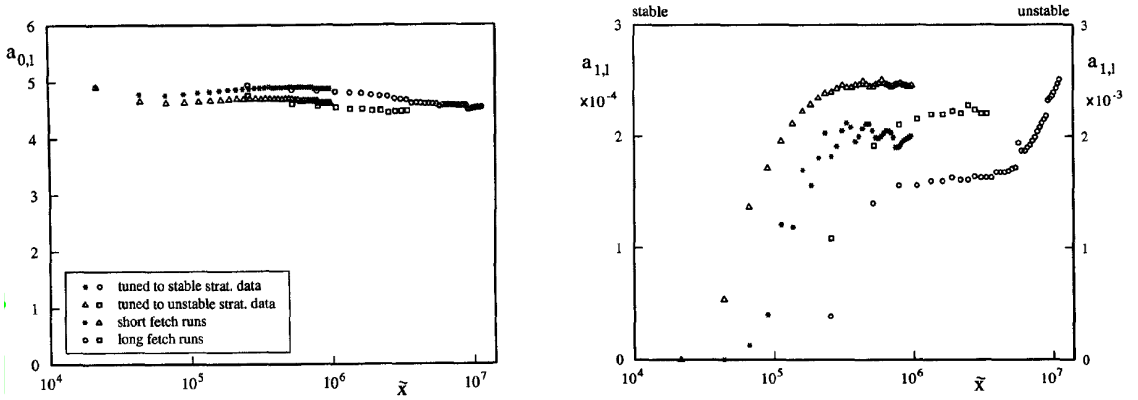


Figure 4.1: Reproduction of figures 4 and 5 from TC96, showing local estimates of parameters a_0 and a_1 at different (dimensionless) fetches and in differing atmospheric conditions.

observational data used for tuning (see Kahma and Calkoen, 1992) was taken in atmospheric conditions of both stable and unstable air flow, as the effect of gustiness tends to allow larger sea states to build (see e.g. Abdalla and Cavaleri, 2002). So it can be seen that parameters a_0 and a_1 are estimated locally, that is, at a given fetch and in given wind conditions. A number of points are raised here. Firstly, a global, or even regional, wave model clearly has to be able to predict waves in all wind conditions simultaneously, but the TC96 parameterisation is therefore approximating by using only a single value for parameters that show variability across condition types. Therefore by selecting only a single value, uncertainty about the resulting conditions is being introduced. Secondly, while the wind conditions may result from either stable or unstable atmospheric conditions, there is not an on / off transition between the two. Thus the process of fitting neglects the possibility of intermediate states and associated parameter values. Again uncertainty is introduced, this time by virtue of the lack of data. Thirdly, as mentioned no account is taken of the way the parameters might interact together. While a sequential approach might give the feeling that the "best" values are being chosen at each step, it completely overlooks very large regions of the parameter space. Finally, a very important aspect relating to uncertainty introduced by the tuning process is that no explicit account has been taken of observational error.

In order to conduct an uncertainty analysis, a probability distribution for these parameters must be elicited in some way. The most straightforward approach is to make use of figures 4.1 (a) and (b), and similar analysis for other parameters provided in Tolman and Chalikov (1996). Considering figure 4.1 (a) in more detail, it can

	a_0	a_1	a_2	b_0	b_1
Eq. (39)	4.8	1.7×10^{-4}	2.0	0.3×10^{-3}	0.47
Eq. (40)	4.5	2.3×10^{-3}	1.5	-5.8×10^{-3}	0.60

Figure 4.2: Reproduction of table 2 from TC96, showing optimal parameter values for the tuning of the dissipation source term.

be seen that the suggested value of a_0 appears to be relatively consistent across all fetches and wind conditions. Tolman and Chalikov (1996) provide estimates from these figures, given in their table 2 (reproduced in figure 4.2), of $a_0 = 4.8$ and $a_0 = 4.5$, for stable and unstable conditions respectively. However, variation is clearly present and it is suggested that where both conditions might be encountered, a_0 might best be represented as $N(4.8, 0.2^2)$. This is justified on the basis that there is no reason to impose hard limits on the possible upper and lower values of a_0 (as could be implied by the use of beta- or uniform-distributions for example). Furthermore, the data do appear to constrain the value of a_0 quite well, implying a distribution with a distinct mode and rapidly diminishing tails, consistent with a normal distribution. Considering figure 4.1 (b), a good deal more spread in the observed (local) estimates of parameter a_1 is evident. In fact, when compared to the optimal values shown in 4.2, it is evident how much uncertainty has been introduced. Of particular note is that there is an order of magnitude difference for the values of a_1 depending on whether it is tuned to stable or unstable stratification. It might be inferred from this that simulator response is not very sensitive to this parameter. Noting again that for a global wave simulation where both stable, unstable and some intermediate states of atmospheric stability need to be represented simultaneously, it is not immediately apparent what probability distribution might best characterise the uncertainty about the possible value of a_1. In the absence of any strong reason to choose a specific alternative, a log-normal distribution is chosen. Elicitation of probability distributions for the remaining tuning parameters a_2, b_0, b_1 and ϕ_{min} followed in a similar fashion and the resulting distributions are shown in table 4.1. For the remainder of this thesis the dissipation parameters will, in accordance with their WW3 input namelist label, be referred to as *sdsa0, sdsa1, sdsa2, sdsb0, sdsb1* and *phimin*.

Finally, note that S_{diss} is very much a closure term in (3.38), as noted by Tolman and Chalikov (1996), and the determination of it's exact form is largely established through a process of tuning, owing to the lack of detailed knowledge of the real dissipation process. In light of this some caution is taken in performing experiments because, in the absence of clear information about the uncertainty associated with

the parameteristion, it is quite possible to explore areas of the parameter space that yield unrealistic output.

4.2.2 Elicitation for S_{nl}

The DIA scheme is a fast approximation to the exact non-linear interaction calculation. Given it is an approximation, it does not necessarily follow that the best performance in any given situation will be attained with a single tuning, and this is indeed not the case. While there have been a few studies (almost exclusively by Tolman) that explore the effects of tuning in terms of accurate spectral reproduction, there is very little information on how this affects realistic simulations. Details of the error introduced by the DIA as compared with the exact computation are documented in, for example, Hasselmann et al. (1985) and Tolman and Chalikov (1996). Note the two principle free parameters are designated λ and C, and relate to properties of the integration. Henceforth these will be referred to as λ_{DIA} and C_{DIA}. Hasselmann et al. (1985) suggest values of the tuning parameters, $\lambda_{DIA} = 0.25$ and $C_{DIA} = 3\times10^7$, without discussing any sensitivity analysis in detail. The authors use language such as, "*After some experimentation ...*" and "*Satisfactory agreement with the exact computations was obtained with the configuration ...* $\lambda = 0.25$.", suggesting some kind of sensitivity analysis was undertaken. However, Tolman and Chalikov (1996) when formulating their source term parameterisation, undertook some sensitivity analysis with respect to the DIA tuning. Not much detail is provided about how the adjustment was made but they found that a reduction in the S_{nl} term was required by means of setting $C_{DIA} = 1\times10^7$. This change was made on the basis that it spread the errors associated with the DIA more evenly across the spectral range, rather than weighting them more toward higher frequencies. Since this earlier work little seems to have been done to determine simulation sensitivity to the tuning of the DIA. Tolman (2004b) does however present detailed information about how tuning, via parameters λ_{DIA} and C_{DIA}, affects the accuracy of the approximation. He in fact maps the resulting error as a function of both λ_{DIA} and C_{DIA}, using the ranges, $[0.12, 0.30]$ and $[9\times10^6, 4\times10^7]$ respectively. During a calibration process conducted with respect to spectra generated by the exact nonlinear computation for duration- and fetch-limited growth, he identifies values for C_{DIA} and λ_{DIA} of approximately 1.5×10^7 and 0.16 respectively ($\lambda_{DIA} \approx 0.20$ in developed conditions). These values are somewhat lower than those recommended by Hasselmann et al. (1985). Nonetheless this information suggests credible ranges within which the integration should at least be stable, and where the results are not unphysical. For C_{DIA} then, this information suggests some kind of range about an approximate mean of 10^7 and

so a log-normal distribution is chosen such that $C_{DIA} \sim logN(7, 0.12^2)$. In effect this sets $\mathrm{P}\{C_{DIA} < 4.3 \times 10^6\} = \mathrm{P}\{C_{DIA} \geq 2.3 \times 10^7\} = 0.001$. Initially, retaining the historical default value as the mean, $\lambda_{DIA} \sim N(0.25, 0.03125^2)$ is adopted, setting $\mathrm{P}\{\lambda_{DIA} < 0.15\} = \mathrm{P}\{\lambda_{DIA} \geq 0.35\} = 0.001$. These distributions are essentially centralised around the value specified in the literature with a variance commensurate with any available information. If observational data were being incorporated into this analysis in order to constrain the parameter space (see e.g. Williamson et al., 2013), broader more speculative prior distributions could be specified.

4.2.3 Wind uncertainty

The wind specification for the experiments in this chapter is simply that of a constant speed and so uncertainty about that is easily interpretable. If the wind speed was derived from a measurement, or a simulation (from an atmospheric numerical model), then precise uncertainty information would be available. The natural variability associated with the wind is not incorporated into the uncertainty specification here. The effect of wind gustiness is incorporated into the input and dissipation parameterisaton, by using different parameter values. Moreover, the wave simulator cannot resolve wind variability on very short time scales, even if the data was available (e.g. from an anemometer). In reality the wind direction may also be subject to variability on some short time scale but again this kind of aleatory uncertainty is subsumed by the parametric specification.

For the experiments in this chapter, the wind speed was regarded as $N(10.0\mathrm{ms}^{-1}, 0.5^2)$.

4.2.4 Numerical uncertainty

As discussed in section 2.2.4, numerical error is minimised by using high spatial and spectral resolution. Here, in the absence of the advection terms in (3.38), there is no spatial grid so we are only concerned with spectral resolution. The number of frequency bins is set to 36, somewhat higher than the default number of 24. The number of directional bins is set to 24, but given energy is not being propagated and the simulation output is direction integrated, this is judged to be of little consequence.

4.2.5 Other sources of uncertainty

No other sources of uncertainty were analysed.

4.2.6 Design summary

Following the elicitation and discussion of uncertainty in the previous sections, table 4.1 lists the parameters that are regarded as uncertain and the probability distributions assigned to them.

4.3 Experiments and analysis

Experiments conducted to investigate the effect of uncertainty on input parameters are presented in the following subsections and listed in table 4.2.

4.3.1 Simulation configuration

WW3 was configured with the TC96 and DIA source term package. A list of all the switches used for the build is given in appendix E, and see Tolman (2009a) for details. Wind is specified as a homogeneous field with a uniform direction using the *ww3_shel* input file. It is possible to specify time and strength variation within the input file, thus avoiding the need to process more complex files with the field pre-processor.

Note that in this configuration that uses only a single grid cell, several days of simulation time in WW3 can be run in seconds on a modern desktop computer. LHS designs of 60 points take approximately ten minutes to execute.

The reader is directed to appendix B for more detailed coverage of running WW3.

4.3.2 Notes on experiment design

In general, for each of the experiments presented here, 60 point maximin LHS designs were generated, as described in section 3.4.2. Design ranges are typically dictated by the choice of input distribution. For example, given a Gaussian input distribution, the design range would be ± 4 s.d. about the mean, ensuring that the input region of interest is well covered by the resulting emulator. Some care has to be taken however, because if the design is too large, it is possible that the emulator will not be well validated in the (small) central region of interest. On the other hand, overly constraining the input region could limit the capability of the emulator, if, for example, at a later date it became desirable to perform an analysis over a broader input space. Additionally, as described in section 3.4.2, it is often desirable to generate at least two LHS designs. The first is used to generate the training data. The second, is typically used to validate the emulator. Depending upon the size of the design it is sometimes necessary to include additional design points. These may be taken from the second design, with a further design generated for subsequent validation.

Table 4.1: WW3 input parameters and sampling distributions for experiments 4.A (a), (b) and 4.B

Switch	**Description**	**Tuning parameter** [**Variable Assignment in WW3 manual**]	**Default value**	**Sampling distribution**
ST2	Dissipation (TC96)			
	Low frequency dissipation	SDSA0 [a_0]	4.8	$\mathrm{N}(4.8, 0.2^2)$
	Low frequency dissipation	SDSA1 [a_1]	1.7×10^{-4}	$log_{10}\mathrm{N}(-3.293, 0.177^2)$
	Low frequency dissipation	SDSA2 [a_2]	2.0	$\mathrm{N}(2.0, 0.2^2)$
	High frequency dissipation	SDSB0 [b_0]	3.0×10^{-4}	$\mathrm{N}(-3.25 \times 10^{-3}, (9.375 \times 10^{-4})^2)$
	High frequency dissipation	SDSB1 [b_1]	0.47	$\mathrm{N}(0.5250, 0.03125^2)$
	High frequency dissipation	PHIMIN [$\phi_{\mathbf{min}}$]	0.003	$\mathrm{N}(0.003, (1.0 \times 10^{-4})^2)$
NL1	Nonlinear interactions (DIA)			
		LAMBDA [λ]	0.25	$\mathrm{N}(0.25, 0.0625^2)$
		NLPROP [C]	10^7	$log_{10}\mathrm{N}(7, 0.1193^2)$

Table 4.2: Experiments presented in chapter 4.

Experiment	Description
Experiment 4.A (a)	Uncertainty analysis for energy dissipation in fully developed conditions.
Experiment 4.A (b)	Uncertainty in wave power from a wave energy converter.
Experiment 4.B	Uncertainty analysis for uncertain input and wind speed in fully developed conditions.

Although this approach tends to minimise the ultimate size of the training data set of the emulator, which obviously reduces computational burden and improves emulator performance, a possible disadvantage is that different designs have to be combined. The "maximin" criteria[32] ensures good spatial coverage for an individual design, but when two such designs are combined there is no such constraint, and points may end up being clustered in an unhelpful way, which at best might reduce efficiency, and at worst give rise to ill-conditioning of the observation covariance matrix. It may seem sensible therefore to generate larger designs, in order to avoid creating additional designs. Training and validation points could then be drawn form the single large design.

Experiments presented in this thesis tend to take the former approach, where design size is minimised initially, and additional LHS designs are subsequently generated[33]. This is generally found to be effective, without causing numerical or validation issues. It is also convenient, because additional point selection algorithms are not required (i.e. a single large design does not need to be sub-sampled), and wave simulations tend to be relatively cheap to produce (particularly in chapters 4 and 5).

4.4 Experiment 4.A (a), the effect of uncertainty about dissipation physics

In the first experiment in this chapter the uncertainty distribution induced in simulated H_s and energy period, T_e, by uncertainty about the tuning of high frequency dissipation physics is obtained. The specific input parameters regarded as uncertain are *sdsa0*, *sdsa1* and *sdsa2* which are distributed according to table 4.1. The output distribution gives us the variance arising due to the input uncertainty and also illustrates whether the simulator is nonlinear in this case. The wind speed was

[32] "Maximin" refers to the maximisation of the minimum interpoint distance.

[33] In practice, it is convenient to create and run four separate designs at the outset.

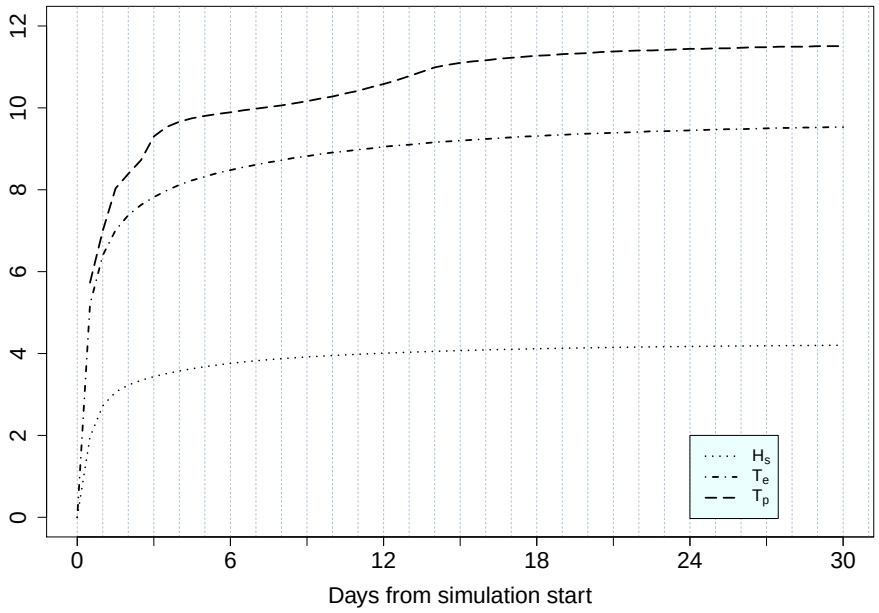


Figure 4.3: Time evolution of wave summary statistics for duration-limited growth.

regarded as known with a constant speed of 10ms^{-1}. The simulator provides output at locations and time intervals specified by the user. The form of the output can also be requested and options include summary wave statistics in fields or, for point output, 1-D wave spectra or the full 2-D spectrum. For this experiment, summary statistics at point output were requested at 12 hourly intervals. The time evolution of H_s and T_e is shown in figure 4.3. We can see that the behaviour of the time evolution of all quantities is asymptotic. The sea state responds within hours and so convergence is rapid during the first few days but continues appreciably over a further week or two. The behaviour of T_p appears to be less regular than either H_s or T_e. This is attributable to the finite spectral resolution and the simulator's determination of the peak. Owing to the nonlinear movement of energy within the spectrum, the peak frequency does not always "follow" the growth of the spectrum. At times this can give rise to a somewhat discontinuous behaviour of T_p, as seen here. Equation (4.1) is satisfied where $t \to \infty$ which is most closely approximated at t=30 days in figure 4.3. Simulation output at t=30 days is therefore used for the analysis in this experiment.

Having output from a designed experiment we now proceed to formulate an emulator in the same way as section 3.5 by fitting a multiple regression model of the form $y \sim 1 + x_{[1,2,3]}$ to the output data to gain some understanding of the response to each input parameter. y denotes the simulator output (H_s at time t=30 days in this case) and $x_{[1,2,3]}$ the input parameters *sdsa0*, *sdsa1* and *sdsa2* respectively.

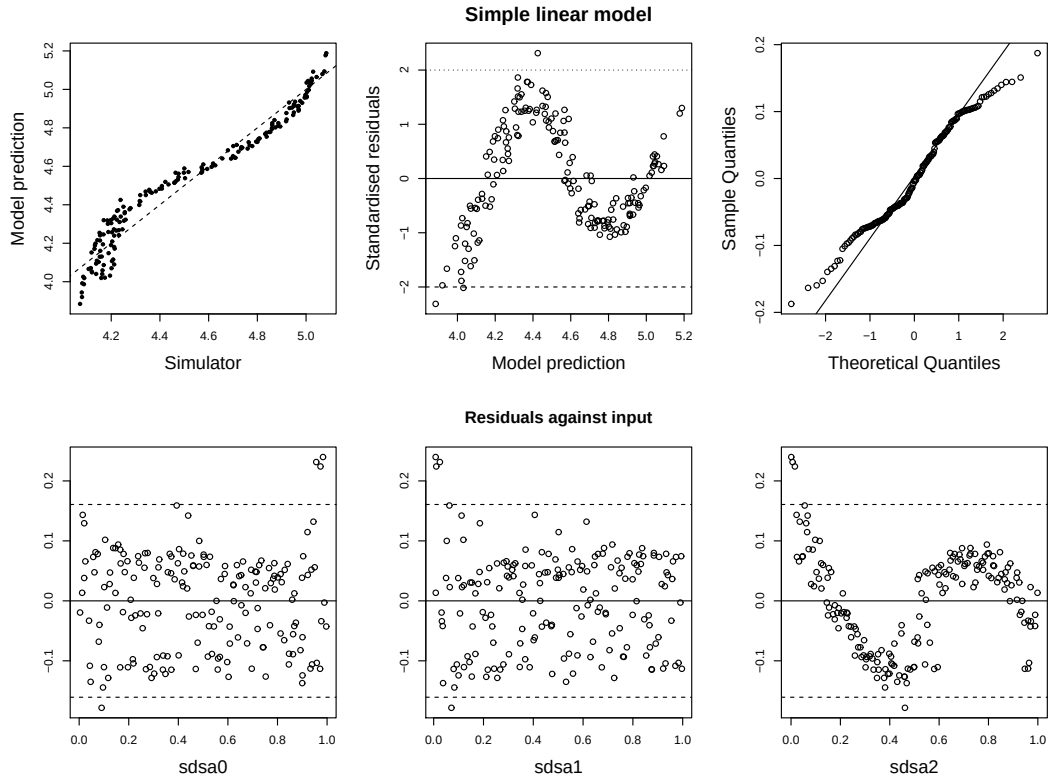


Figure 4.4: Linear regression model fitted to simulated H_S for experiment 4.A

Regression diagnostics are shown in figure 4.4. The strong correlation of the output with the input parameter *sdsa2* is clear, whereas *sdsa0* and *sdsa1* appear to have only a small approximately linear influence. As in section 3.5 this information can be used to formulate a prior mean function for the emulator. The relationship between *sdsa2* and the output data is explained by higher order terms and so terms of up to sixth order in *sdsa2* are employed in the regression basis functions. A new regression model of the form $y \sim 1 + x_{[1,2,3]} + x_3^2 + x_3^3 + x_3^4 + x_3^5 + x_3^6$ is fitted and it can be seen in figure 4.5 that much of the trend has been explained by the model leaving residuals that are approximately normally distributed.

Note that the choice of terms in the mean function is determined by a statistical significance test on the coefficients when performing the linear regression. Typically, those that reject the null hypothesis at 95% confidence level are included. In some cases strong correlation between the output and high order terms of the input is revealed. Less formally, a visual inspection of regression diagnostics will usually justify the use of the first few terms in the mean since the structure of the output is quite clear. If not, there is probably little improvement in the resulting emulator through the inclusion of numerous or higher order terms. Noting the improved

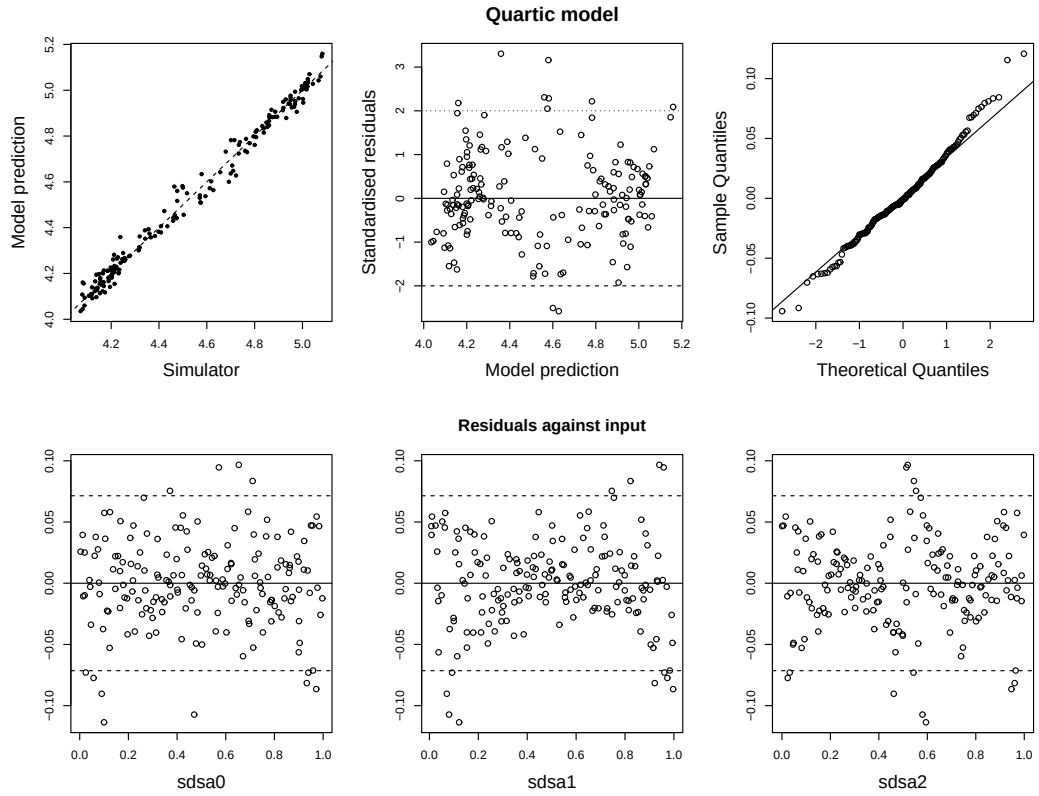
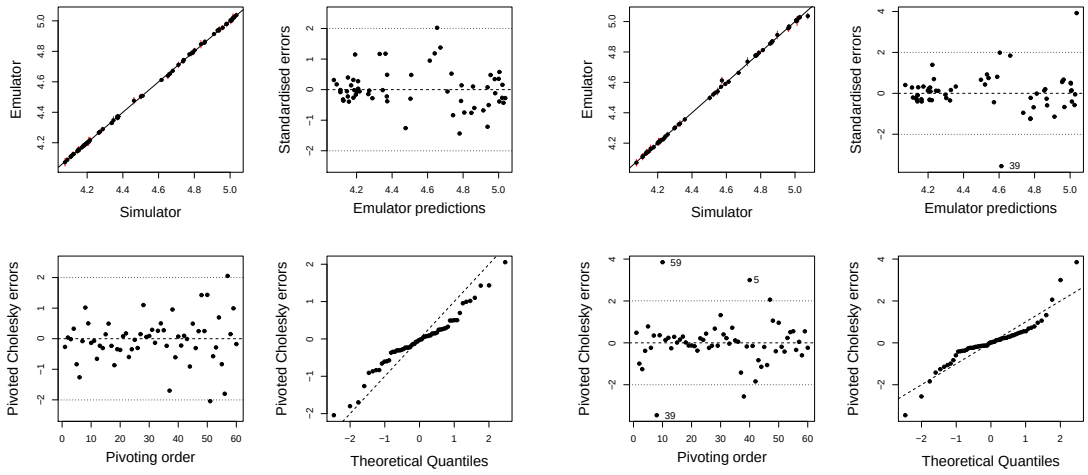


Figure 4.5: Linear regression model incorporating higher order terms fitted to simulator output (experiment 4.A)



(a) Validation with design set #2 **(b)** Validation with design set #3

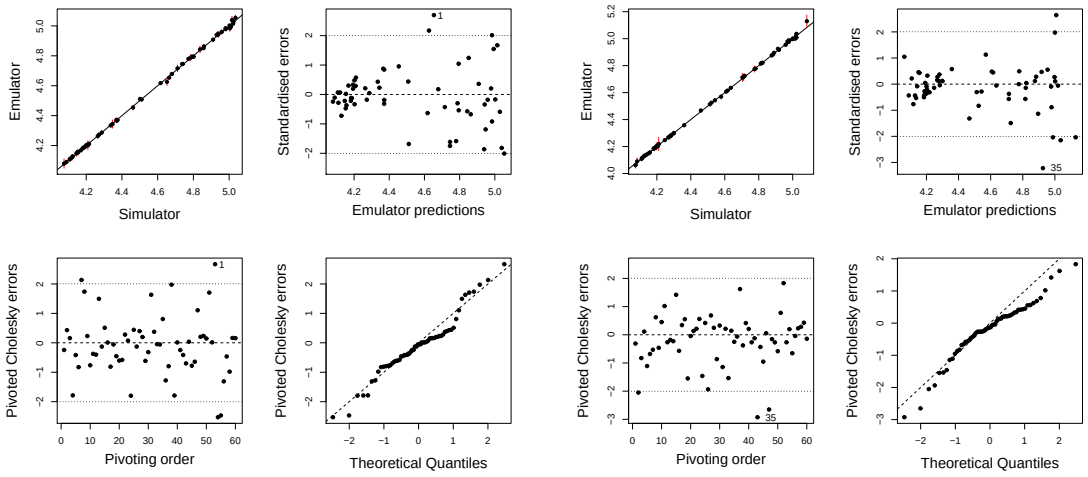
Figure 4.6: Graphical diagnostics for an emulator for H_s built with linear terms in the mean function and Matérn ($\nu = 3/2$) covariance functions.

mean function, two emulators are constructed for comparison and to illustrate the difference in performance. Firstly, only linear terms of the three input parameters are used in the prior mean function, that is $y \sim 1 + x_{[1,2,3]}$. A Matérn ($\nu = 3/2$) covariance function completes the emulator prior specification and the correlation lengths were determined using maximum likelihood estimation. Graphical diagnostics of the emulators performance are shown in figure 4.6. It is evident that this emulator is doing a reasonable job at fitting the data. The Q-Q plot indicates that the distribution of the residuals is deviating slightly from the reference distribution and exhibit light (panel (a)) and heavy (panel (b)) tails. The top left panels in figures 4.6 (a) and (b) show that the emulator mean function is doing a very good job of making point predictions of the simulator. The observed Mahalanobis distances for the two cases were 32.0 and 66.8 respectively. Compared to the reference distribution with mean 60.0 and s.d. 16.2, this suggests reasonable agreement of the model however the lower observed value for the first validation data set suggests over estimation of the variance. The graphical diagnostics also suggest some other possible conflicts. Specifically, figure 4.6 (b) reveals the presence of outlying points. If this was a general trend then it might imply an underestimation of the variance, due perhaps to a poor estimation of the correlation lengths. However, with just one or two isolated points, this is probably indicative of a localised misfit due to the design. A quick inspection of the design points shows that 39 is located at $[0.9310, 0.07404, 0.6072]$, and 59 is located at $[0.00930, 0.9955, 0.9846]$. These are

both on the edge of the design, and indeed no. 59 is right in the corner, and so with regards to any subsequent uncertainly analysis undertaken using the emulator, because the misfit lies well outside the range of interest, it is unlikely to have any measurable effect. In both validation cases shown in figure 4.6 (a) and (b), the Q-Q plot indicates deviation from the expected reference t-distribution implying that the normality assumption for the data might be too strong. In fact, we know that the residuals are highly correlated with *sdsa2* and we can use this information to build a more robust emulator. Other options to make improvements at this point could include the use of a nugget term, addition of design points or a re-assessment of the correlation lengths.

At this point we make some adjustments. Firstly, recalling the findings of the earlier regression exercise, a prior mean function of the form $y \sim 1 + x_{[1,2,3]} + x_3^2 + x_3^3 + x_3^4 + x_3^5 + x_3^6$ is adopted, along with Matérn ($\nu = 5/2$) correlation function. Secondly, on the basis of anticipating some non-stationarity in the data, a nugget term was added to the emulator model. Lastly, the new emulator is rebuilt trained with a slightly larger data set, in an attempt to rectify the localised fitting problems. Noting the outlying points in figure 4.6 (b) a small subset of points were taken from design set #3, including those outliers, and added to the training data. This is likely to bring about a new estimate of the correlation length scales which should also lead to improvement. The correlation length scale parameters were determined again through MLE and the resulting emulator was again validated. Results from validation with designs #2 and #4 are shown in figure 4.7. Observed Mahalanobis distances of 62.5 and 50.8 for the two sets respectively suggest good agreement of the emulator with the data. The graphical diagnostics do not reveal any specific or substantial conflict. It is noticeable that there is a clustering of points exhibiting high predicted variance at H_s values of around 4.2m. The contrast with the predicted variance at larger wave heights indicates heteroscedasticity in the data. In fact this is also evident from the linear regression diagnostics seen in figure 4.5. The lower middle and right hand panels both show non-constant variance. Nonetheless, the emulator appears to be robust so we can now proceed to perform statistical inference.

The objective is to obtain the uncertainty distribution in simulation output (in terms of H_s and T_e) induced by the uncertainty about the input parameters. An uncertainty analysis is thus performed using the emulator mean function in a Monte Carlo type approach. Recall from section 4.2 that input uncertainty distributions were derived from the literature. The analysis thus proceeds by drawing 10,000 samples from the joint distribution, which are then rescaled to the range 0 to 1 and used as input data for the emulator. The emulator is then run for each input point yield-



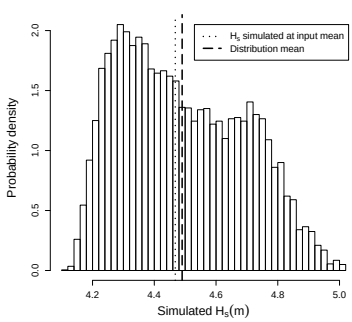
(a) Validation with design set #2

(b) Validation with design set #4

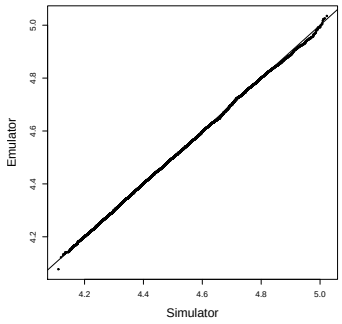
Figure 4.7: Graphical diagnostics for the H_s emulator built with 6th order polynomial mean and Matérn ($\nu = 5/2$) covariance functions.

ing an output distribution for H_s. In this case since two different emulators were built for comparison, one with only linear terms of the input parameters and one with higher order terms, both are used for the uncertainty analysis. The resulting distributions from each emulator are shown in figures 4.8 (a) and (c) respectively. Before discussing specifics of the distribution, a cursory comparison of panels (a) and (c) suggests the difference between the output from the two different emulators is very small. The modifications implemented in the second emulator have made little practical difference in this case. It is reasonable to conclude from this that the we need not be especially concerned about trying to achieve "perfect" output diagnostics when formulating our emulators. That is not to say we can be complacent but using the diagnostics with an element of pragmatism should lead to successful analysis.

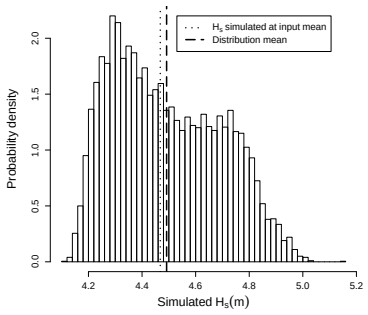
Let us now look at the uncertainty distributions. These tell us about the resulting variance in predictions and nonlinearity within the simulator. Addressing the total uncertainty first, the distribution s.d. is 0.2m but the rapid decay of the tails gives rise to appreciable probability of much of the total range of possible values. For example, we can see that for values of H_s lying between approximately 4.5m and 4.8m, the probability density is approximately uniform. Moreover, the difference between the 12.5 and 87.5 quantiles is almost exactly 0.5m indicating the breadth of the distribution. This result reveals considerable spread resulting from the tuning uncertainty and really draws attention to the problem of relying on predictions



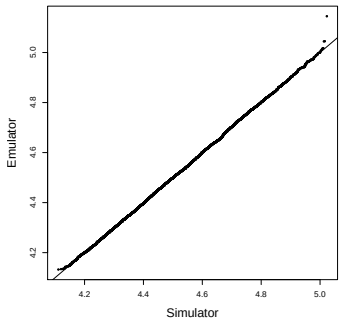
(a) Analysis using emulator 1 (linear terms in the mean).



(b) Q-Q plot comparison of emulator analysis with direct evaluation using WW3.



(c) Analysis with emulator 2 (higher order terms in the mean).



(d) Q-Q plot comparison.

Figure 4.8: Uncertainty analysis for simulated H_s.

based only upon single values (such as the mean input). Regarding the nonlinearity, the input probability distribution is a multivariate normal and any deviation from normality in the output is attributable to nonlinear response. Deviation is clearly evident from these results. It is interesting to note that the distribution exhibits bimodality and has rapidly decaying tails. This is attributable to the response to *sdsa2*, which is evident from figure 4.5. The nonlinearity essentially results in a broadening of the probability distribution but does not cause a large difference between the simulation output at the mean input and the distribution mean.

Interpretation of this result should be taken in the context of tuning an empirical parameterisation. That is, in this case the uncertain parameters are not physical quantities that are part of a well established theory. They arose as a result of a search for the best fitting parameterisation. The nonlinearity is thus probably not attributable to some aspect of wave physics but rather a feature of the author's choice of mathematical expression that was found to be most effective in terms of reproducing observed sea states. However, in general identification of behaviour like this is important and may, for example, be a useful verification of expected behaviour or highlight something unexpected, such as emergent behaviour or an error.

In addition to the uncertainty distributions shown in figure 4.8 (a) and (c), panels (b) and (d) have yet to be discussed. These show a Q-Q comparison between the uncertainty distribution computed using the emulator and the same analysis computed directly using WW3. In this case the simulator is fairly cheap to run so the requisite 10,000 runs could be obtained on a desktop workstation. While the point of using an emulator is to circumvent such a direct approach, it can nonetheless be a useful check where practicable[34]. Inconsistency between the probability distributions is clearly negligible which gives us good confidence that both emulators are performing very well. It is worth highlighting that the 10,000 simulation runs (performed on a desktop workstation) took 18 hours to compute. Contrast this with the 21 minutes it took to obtain the four 50 point latin hypercube designs.

At this point the experiment is extended to include an uncertainty analysis of T_e. Again we have a single dimension of output, derived from the full wave spectrum. Emulators were formulated in the same way as for significant wave height. Given that H_s and T_e are derived from the same data correlation is expected and we might imagine similar results prior to the analysis. Figure 4.9 shows a linear regression of the data and patterns of response similar to those discovered in figure 4.4 are clear. In this case a prior mean function with up to 4th order terms in *sdsa2* was used in combination with a Matérn ($\nu = 5/2$) prior covariance function. This yielded a

[34]Typically running 10,000 simulations, even using a fast code, is highly laborious.

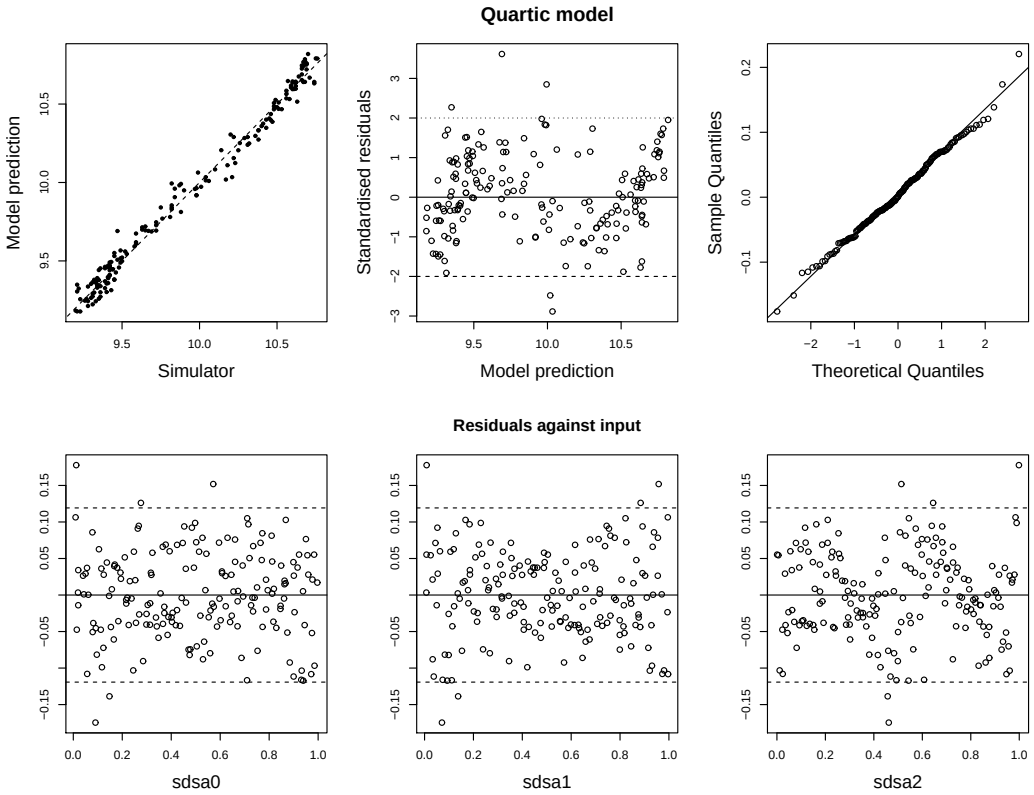
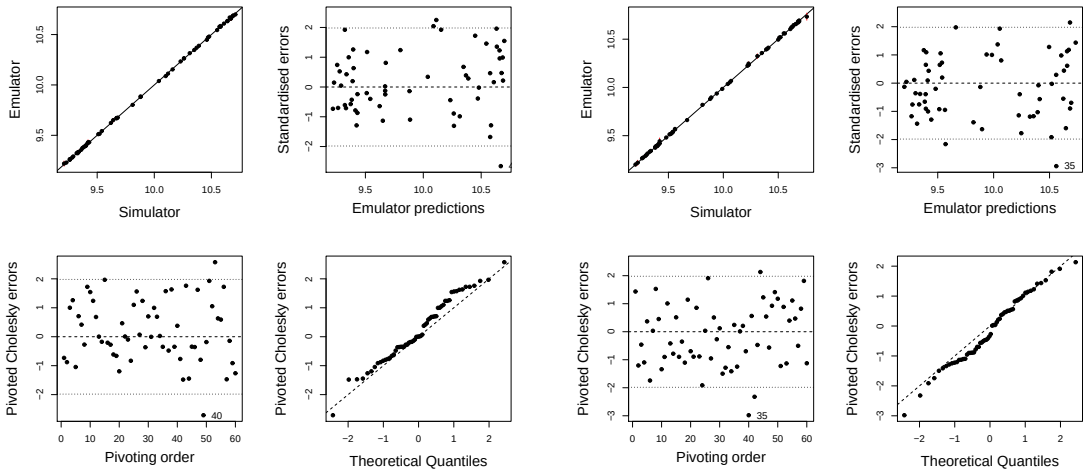


Figure 4.9: Linear regression model fitted to simulator output for T_e (experiment 4.A)



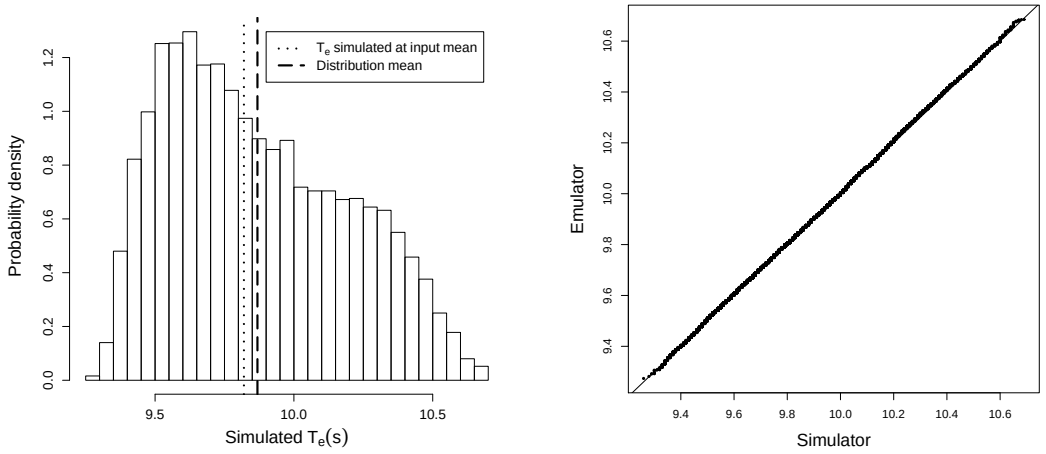
(a) Validation with design set #2

(b) Validation with design set #4

Figure 4.10: Graphical diagnostics for the T_e emulator built with 4th order polynomial mean and Matérn ($\nu = 5/2$) covariance functions.

robust emulator as supported by the graphical validation diagnostics seen in figure 4.10. The resulting uncertainty analysis yields the probability distribution shown in figure 4.11 (a). As expected the shape of the distribution is closely comparable to that for H_s shown in figure 4.8. The same pattern of nonlinearity is evident which was somewhat expected since the two quantities are derived from the same frequency spectrum. By comparison, the values of T_e corresponding to the 12.5 and 87.5 quantiles are 9.5s and 10.3s respectively. We can also see in figure 4.11 (b) that the emulator reproduced the direct uncertainty analysis with great accuracy.

This concludes experiment 4.A (a), which has successfully obtained the uncertainty distributions induced in simulations of H_s and T_e for fully developed sea states. A direct comparison of the results from the emulator with results from the simulator revealed that the emulators reproduced the simulator behaviour almost exactly. The uncertainty distributions, together with regression analysis, have revealed nonlinearity in the simulator response attributable to the input parameter *sdsa2*. In fact, the emulators for H_s and T_e can now be combined in a useful way to provide insight into the field of renewable energy from waves, and this is explored in the next experiment.



(a) Uncertainty distribution obtained with the emulator.

(b) Q-Q plot comparison of emulator analysis with direct simulation.

Figure 4.11: Uncertainty analysis for simulated T_e in duration limited conditions.

4.5 Experiment 4.A (b), uncertainty in wave power from a wave energy converter

The worlds oceans offer an abundant source of energy in the form of waves. Various studies, such as Boud and Barnett (2012), Cornett (2008) and Mackay et al. (2010a) have attempted to quantify the available resource on regional and global scales. The UK is well placed to take advantage of this resource, particularly in the north where long period swell arrives from the west, across the north Atlantic. Recent estimates of the resource available to the UK (Boud and Barnett 2012) suggests that it could be circa 75TWh/yr or a mean power output of approximately 10GW. Over recent years commercial devices have been developed that can extract energy directly from the waves. They come in a variety of designs and are generally referred to as wave energy converters (WEC). The Pelamis Wave Power[35] devices operate offshore and by contrast the Oyster device built by Aquamarine Power[36] operates in the surf zone. Although few devices have been commercially deployed, development is ongoing and there is nonetheless enormous potential for resource extraction. As such, manufacturers, utility companies and governmental agencies have a variety of interests in the potential resource. Knowing about the long term resource associated with the wave climate is important to investors who seek long term financial return, whereas short term variability is key to both implementing grid technology (see e.g. Reikard,

[35] http://www.pelamiswave.com/

[36] http://www.aquamarinepower.com/

2013) and trading in real-time energy markets. It is thus important to understand the uncertainty associated with both the short and long term resource assessment. An analysis was carried out by Mackay et al. (2010a) and Mackay et al. (2010b) in order to assess uncertainty about wave energy resource as predicted by a wave simulator on both short and long time scales. They used statistical post-processing to calibrate wave hindcasts, in order to reduce uncertainty in the prediction of mean wave power extraction from different hindcasts. In order to assess the prediction of short term wave conditions in the context of forecasting incoming wave energy, Reikard et al. (2011) compared the performance of a numerical physics model (the ECMWF WAM model) and statistical methods, including an autoregressive model and an artificial neural network. The study addresses uncertainty in terms of deviation from observed conditions, and concludes that a combined approach using an autoregressive model based upon simulated forecast conditions gives the most reliable prediction. However, Reikard et al. (2011) do not attempt to investigate the underlying sources of uncertainty. In fact, none of these articles investigate the underlying sources of uncertainty in a wave simulator or how these contribute to uncertainty in predicted WEC power extraction. In order to take a step in that direction we can perform an uncertainty analysis, based upon what we have learned about uncertainty in tuning, and utilising information about the power response of a WEC. It is typical for WEC manufacturers to provide a power conversion matrix for a device that specifies the power output of a WEC in a given sea state expressed in terms of H_s and T_e. A number of such conversion matrices for different devices are given in Reikard (2013) which includes in particular the widely studied Pelamis P-1 device. The response table is not reproduced here but a visualisation of the function is shown in figure 4.12. By utilising the emulators formulated in experiment 4.A (a), the response function can be used to compute the uncertainty distribution in power extraction, given a forecast subject to uncertain tuning as is being investigated in this thesis. So we proceed by drawing a sample of 10,000 points from the input probability distribution and run them on the H_s and T_e emulators. The resulting sets of output are used as input to the power conversion matrix. The resulting uncertainty distribution is shown in figure 4.13 (a). The remarkable feature of this result is the extremely narrow peak in an otherwise fairly broad distribution. However the presence of this peak can be explained with closer inspection of the output from the wave simulator and the response function, as seen in figure 4.13 (b). The uncertainty distributions for H_s and T_e are highly correlated. The (2-dimensional) domain is superimposed on the WEC response function, seen in panel (b) and appears like an almost straight line owing to the high degree of correlation between H_s and T_e.

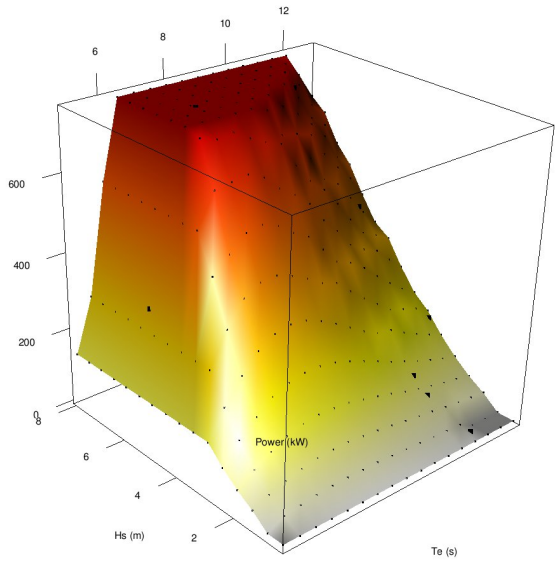
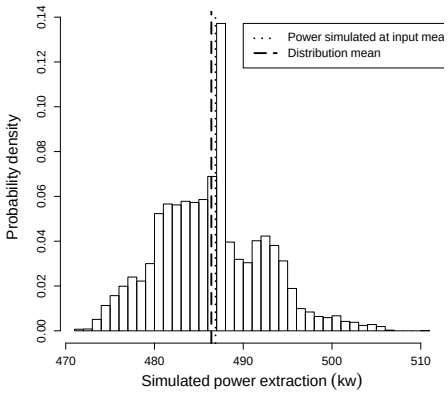
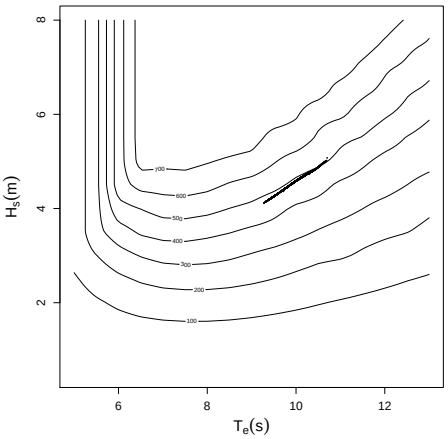


Figure 4.12: Visualisation of the power conversion matrix for the Pelamis P1 device.

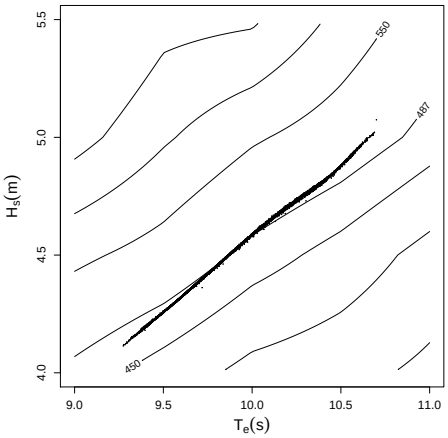
A closeup of the region of response and the joint output distribution is shown in panel (c), which reveals that the correlation causes the joint distribution to align with the contour of the power response function, at a power of 487kw, thus giving rise to to a large sampling frequency in a very narrow range of power output. It may be purely circumstantial that the variability induced in simulated H_s and T_e by the uncertain tuning parameters gives rise to a very peaked distribution (due to the correlation in H_s and T_e). However, if one imagines for a moment that the natural variability in sea states typically exhibits such a correlation between H_s and T_e as we have seen, then it would be advantageous (and indeed quite intentional) that the WEC power response function exhibits contours coincident with the gradient of the correlation. This would in general lead to a more consistent power output which would undoubtedly be more easily supplied to an electricity grid. The details of optimal WEC design are beyond the scope of this work but this simple investigation could easily be extended to explore more thoroughly the implications of uncertainty about resource from a wave simulation. An extension of this work would be to account for uncertainty in the power conversion matrix itself. For example, it may be that in practice many other factors beyond simply the sea state, in terms of H_s and T_e, give rise to variability. In terms of wave energy, the angle of incidence, wave steepness, or even surface obstructions such as an abundance of seaweed could be factors to include. The engineering and electrical generation may also be subject to some variability. Such information could be elicited from experts as described in section 3.2.



(a) Uncertainty distribution for WEC power extraction.



(b) Contour plot of WEC power extraction as a function of H_s and T_e.



(c) Close up view of panel (b).

Figure 4.13: Results of the uncertainty analysis for wave power extraction. Note the extreme peak in panel (a), and in panels (b) and (c), the narrow bivariate probability distribution (black line) that identifies the specific domain for power output associated with the uncertain wave conditions.

This investigation highlights how learning about uncertainty in the wave simulator can be used to obtain probabilistic information in related studies. Only a limited number of sources of uncertainty were considered here although this could easily be extended. Notably this example is not dynamic - a key assumption being that wave conditions are fully developed. A constant wind speed is also assumed but this can be relaxed by regarding the (constant) speed as unknown. In this way the wind speed can be regarded as an additional uncertain input parameter. This approach is taken in the next experiment, where the analysis is expanded to a number of other uncertain input tuning parameters.

4.6 Experiment 4.B, uncertainty about input, dissipation and nonlinear interactions

In this experiment we return to the analysis of the simulator output induced by incomplete knowledge about input parameters and tuning. This experiment is the same as experiment 4.A (a) but the scope of analysis is extended to include a wider range of sources of uncertainty. Now, six tuning parameters associated with the TC96 input and dissipation scheme are regarded as uncertain, these being *sdsa0*, *sdsa1*, *sdsa2*, *sdsb0*, *sdsb1* and *phimin*. In addition, two parameters, λ_{DIA} and C_{DIA} that govern the exact form of the DIA scheme for nonlinear interaction are considered. Uncertainty distributions determined in the elicitation are listed in table 4.1. The forcing wind speed is regarded as constant in time, but uncertain in terms of its constant value. It is specified as $N(10.0\text{ms}^{-1}, 0.5^2)$.

As before, the simulation was run for 30 days to ensure that fully developed conditions were achieved. Proceeding in the usual way, training data was obtained by generating four 60 point LHS designs and running the simulator for each design. Note that in this case the actual design range included $0 < \lambda_{DIA} \leq 0.5$, which turned out to be important, as discussed shortly. Output was obtained for the usual wave summary statistics for each 12 hourly time point. Initially analysis is performed for H_s data at t=30 days. Inspection of the initial linear regression, seen in figure 4.14, reveals a very distinct trend, but more importantly unrealistic predictions of H_s. Values of H_s exceeding 10m are not likely to be observed given a wind speed of approximately 10ms^{-1}. This is not necessarily problematic in terms of building an emulator provided an appropriate model can be found but there is little value in expending effort in constructing an emulator to fit unrealistic or spurious data. The probable cause of the extreme output is that the value of an input parameter has been set too far outside its intended range. To diagnose this problem, the data were log-transformed in order to constrain the more extreme values, and a regression model

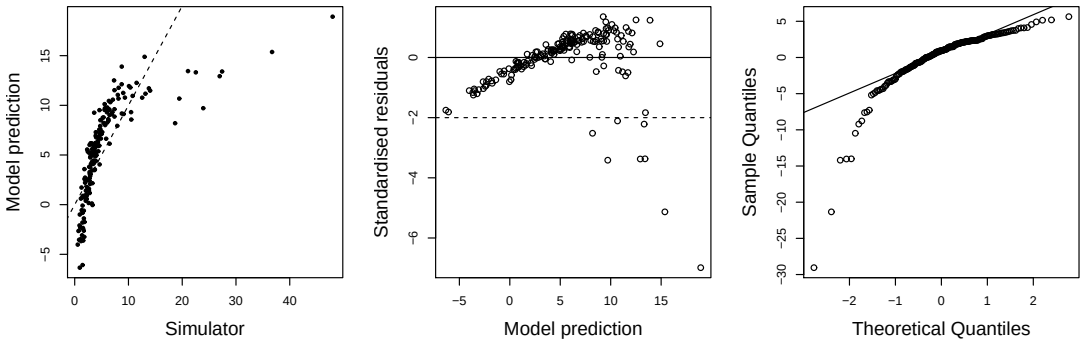


Figure 4.14: Linear regression model fitted to simulator output for H_s (experiment 4.B)

was fitted so that the residuals could be inspected. These are shown in figure 4.15. Significant correlation is only evident for the input parameter, λ_{DIA}, associated with the DIA scheme. More importantly, it appears that this parameter is responsible for the extreme values of H_s, as they are closely correlated to low values (as $\lambda_{DIA} \to 0$). A close inspection of the bottom left panel indicates a uniform relationship between λ_{DIA} and H_s between scaled values of 0.3 and 0.7, corresponding to actual values of 0.15 and 0.4. Outside of this range the variance appears to grow rapidly and the response becomes more nonlinear and extreme. It might therefore be expected that more realistic values of H_s are obtained within this range. This is consistent with some information elicited in section 4.2, that suggested values of λ_{DIA} below 0.12 maybe unrealistic. In this case the extreme values are avoided by redesigning the experiment with $0.125 < \lambda_{DIA} < 0.375$ to alleviate potential fitting problems. This is inefficient in that the output data already obtained is not reused however the simulator is cheap to run so a redesign is a possible option.

Regression modelling of the new data reveals very little correlation of the residuals with output H_s for all parameters. Squared terms were added to the basis functions for the 8th (C_{DIA}) and 9th (wind speed) input parameters of the emulator mean function. A Matérn correlation function ($\nu = 5/2$) was also chosen and the emulator was trained to the $log(H_s)$. A nugget term, equal to 1.5×10^{-4}, was found to give the best fit. The emulator was trained using the whole of the first design set, and half of two subsequent designs in an attempt to capture some extra points and correct localised fitting issues, likely due to the outliers observed during the initial regression. Determination of the correlation length scales was done using maximum likelihood estimation. Only a single maximum was identified by the optimisation routine. Inputs 2, 3 and 7 gave rise to length scale values below unity, being 0.7822,

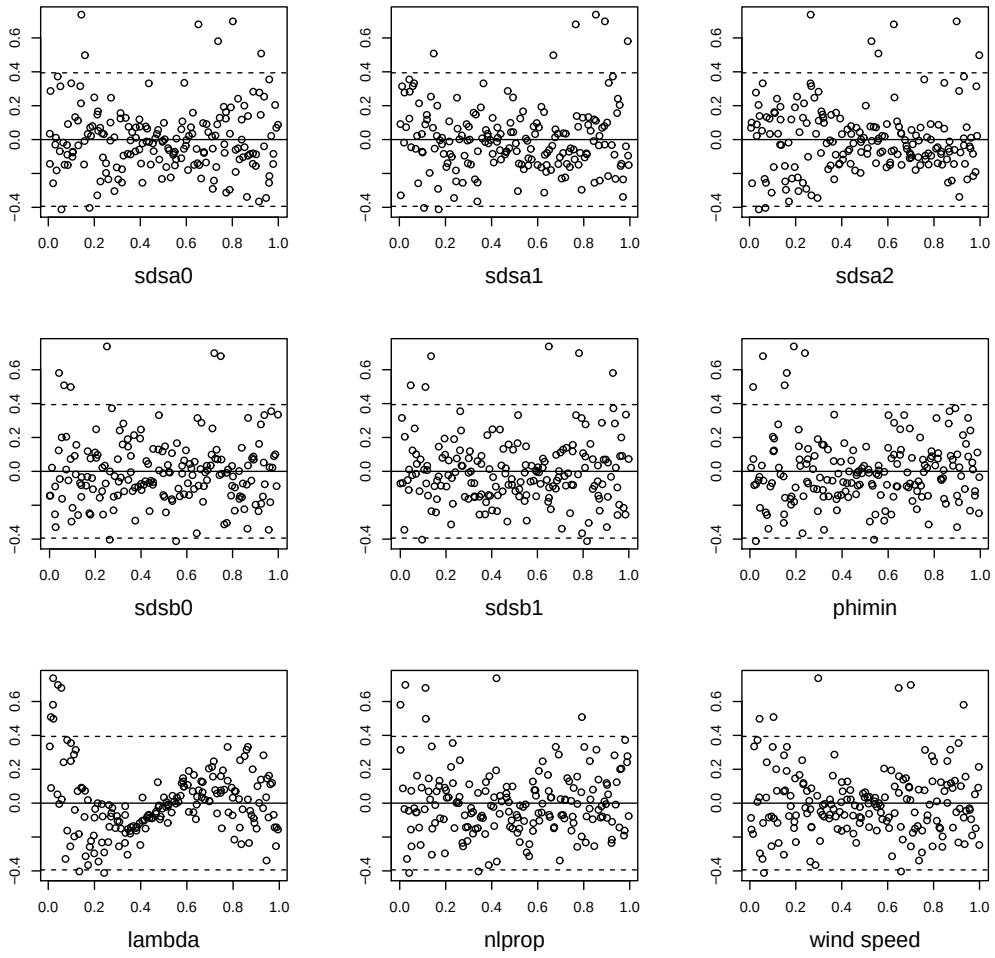
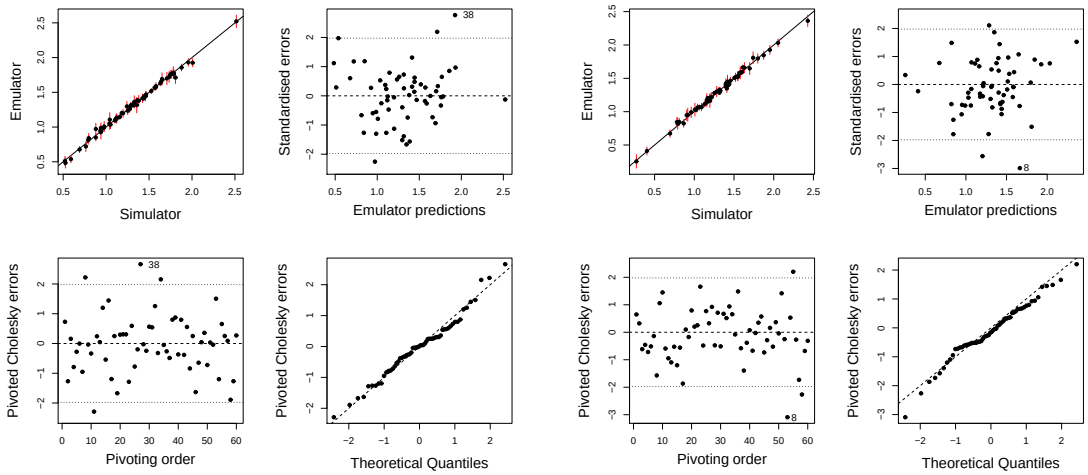


Figure 4.15: Residuals from a linear regression of $log(H_s)$ plotted against input (experiment 4.B)

0.3595 and 0.4384 respectively. An implication of this is that these inputs are the most sensitive, and that the others may be relatively inactive. A sensitivity analysis (see section 3.3) would be needed to obtain quantitative sensitivity measures, but to some extent qualitative information can be inferred from the correlation lengths alone. Output from validation diagnostics is shown in figures 4.16 (a) and (b). Each validation set consisted of 60 points and the emulator performed well, suggesting no conflict with the simulator. The Mahalanobis distances for the two cases were 56.1 and 56.0 respectively, as compared with the reference distribution with mean 60.0 and s.d. 13.2. Proceeding on the basis that the emulator is representing the simulator adequately an uncertainty analysis was performed. The joint distribution as specified in table 4.1 was slightly modified in light of the findings about λ_{DIA}. A distribution of $N(0.25, 0.03125)$ for λ_{DIA} was adopted. 25,000 samples were drawn randomly from the joint distribution, scaled to the appropriate emulator input value

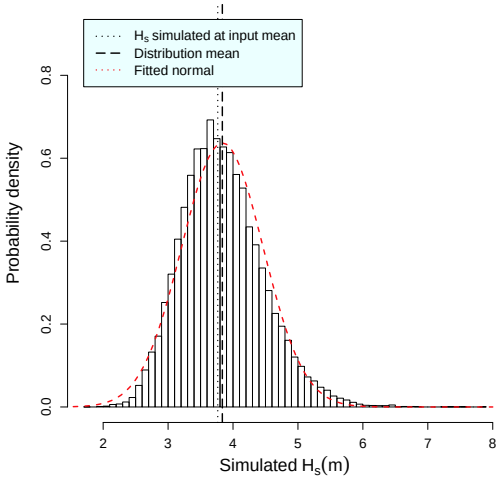


(a) Validation with design set #3 **(b)** Validation with design set #4

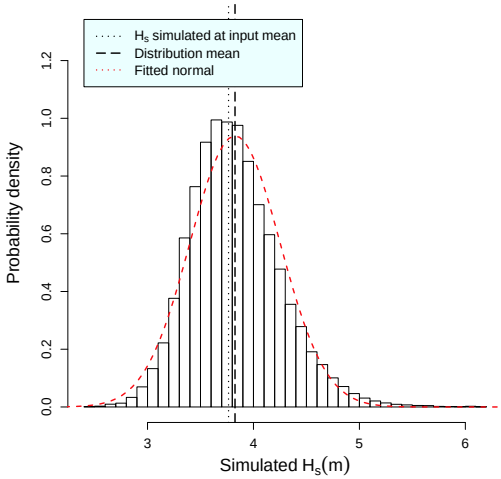
Figure 4.16: Graphical diagnostics for an emulator for H_s built with a mean function of the form $y \sim 1 + x + x_{8,9}^2$ and Matérn ($\nu = 5/2$) covariance function.

(between 0 and 1) and run using the emulator mean function. The resulting distribution in wave height is shown in figure 4.17 (a). The resulting distribution has a mean and s.d. of 3.84m and 0.63m respectively. It deviates from a normal, being slightly skewed, which may be attributable to the log-normal sampling distribution for inputs *sdsa1* and C_{DIA}. However, the fitted normal distribution shows that the deviation is fairly small. The standard deviation of 0.63m is quite substantial given a mean of 3.8m. Using this simple analysis it is possible to investigate the contribution to the variance of the uncertainty due to an individual input, such as wind speed. To do so, the analysis was run again but setting the wind speed distribution to $\mathcal{N}(10, 0.05^2)\text{ms}^{-1}$, a 99% reduction in the variance. The resulting histogram is shown in figure 4.17 (b). The output variance of 0.18m shows a reduction of 0.21m, or 54%. It would therefore be reasonable to conclude that in the case of duration-limited wave conditions wind speed uncertainty is indeed the dominant factor, since the remaining 46% of the variance is explained by the other eight input parameters. The near Gaussian shape of the output distributions indicates that the simulator response is nearly linear.

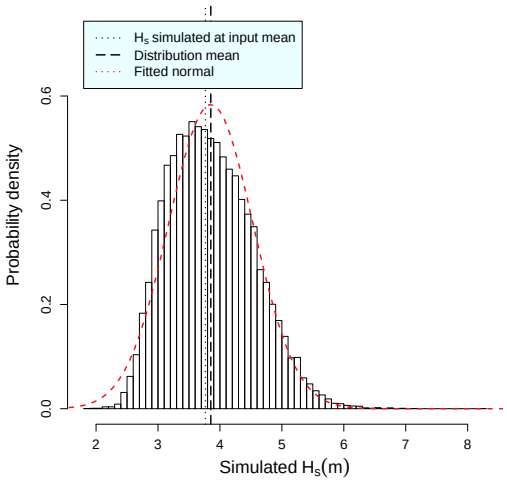
An alternative analysis can be performed by expressing the wind speed as a uniform-distribution, although here it is used more as a tool to see what effect the change of distribution has on the analysis. Use of a prior uniform distribution implies "hard" limits to the range of values, which in most physical situations seems



(a) Uncertainty distribution for H_S



(b) Uncertainty distribution for H_S with wind speed variance reduced by 99%.



(c) Uncertainty distribution for H_S with wind speed distributed as $\sim unif(9.0, 11.0) ms^{-1}$.

Figure 4.17: Uncertainty distributions induced in simulated H_s with three different input uncertainty specifications for wind speed.

implausible[37]. Wind speed would seem to fall into that category too (that is where $P(w.s. = x) > 0$ but $P(w.s. = x + \Delta x) = 0$) however there might be situations where a uniform distribution was a reasonable approximation, such as a wind climate in a specific region for example. Proceeding nonetheless, the input distribution is represented as $wind\,speed \sim \text{unif}(9, 11)\text{ms}^{-1}$. Figure 4.17 (c) shows that the effect is to cause a broadening of the peak of the uncertainty distribution. The effect is consistent with the strong influence of the wind speed, and the broadening is directly attributable to the broad and "flat" uniform distribution. This also shows us that the output uncertainty distribution is very sensitive to the prior distribution for the wind speed, so in any subsequent study this should be carefully considered. No examination of other input parameters is conducted here although a wide range of similar analyses could be performed in order to determine the sensitivity of the output uncertainty to the input specification. More rigorous quantitative sensitivity analysis for other input parameters is carried out in subsequent chapters.

Following on from the analysis of simulated H_s, we now turn our attention to the behaviour of simulated wave period. Within the wave modelling community performance of spectral wave models tends to have more of a focus on wave height rather than period. Perhaps as a result of this, performance of wave simulators with respect to predicting period is sometimes poor. There are many measures of wave period (see chapter 2) so when discussing wave period we have to be cognisant of the wave frequency spectrum. A number of 1-D (direction integrated) spectra from a design set are shown in figure 4.18. Examination of the design space can be very useful in giving us a feel for how input uncertainty manifests itself in the output. The data exhibit only unimodal JONSWAP type spectra (see section 2.4.2), as expected in such a situation[38]. In this case the peak period, T_p, has a clear interpretation and is a useful summary statistic, particularly given that the sea states are well developed. The spectra also show how the shape of the spectral peak changes between design points. For example, it is evident that in the sample shown there is more variability in H_s than T_p. There is also considerable variation in the overall shape of the spectra. It is also clear that the design still spans regions of the simulator input space that give rise to unrealistic conditions. The previous uncertainty analysis generated a distribution for H_s with a mean of 3.83m and a s.d. of 0.63m, so 9.06m (bottom right panel) is somewhat extreme![39].

[37] The temperature of liquid water at atmospheric pressure can clearly not be below 0C or above 100C, so a distribution with a hard limit for possible water temperature, in some situations, could be conceivable.

[38] Bimodal or multimodal spectra typically arise where the wind exhibits temporal variability or there is interaction between wave systems originating from different locations.

[39] A quick inspection of that particular design point informs us that it lies at the very edge of

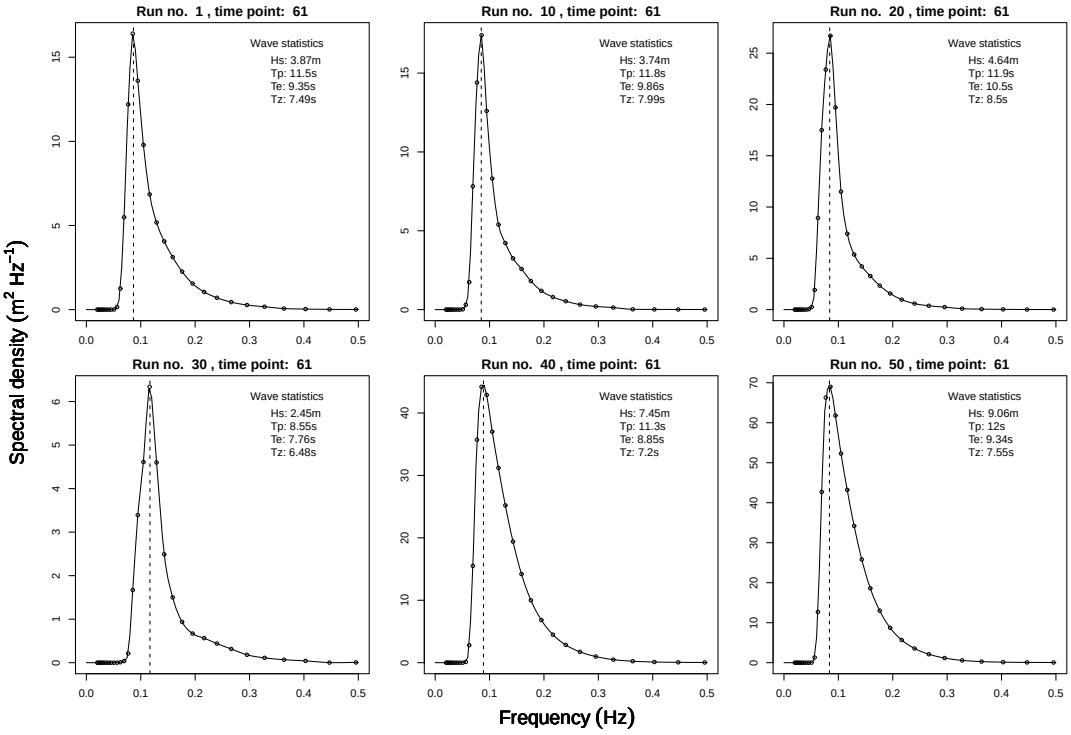
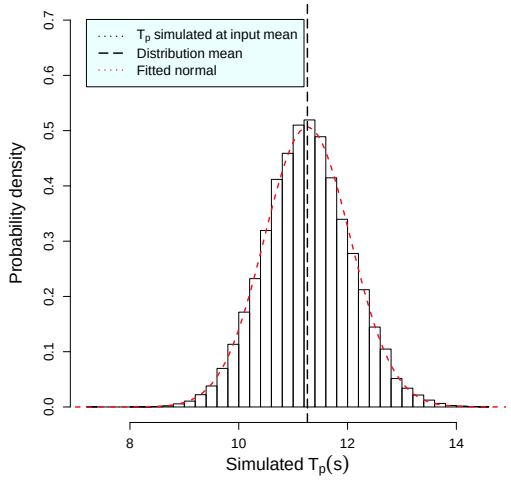


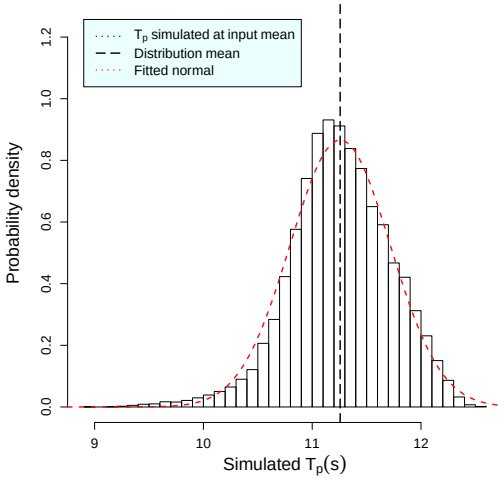
Figure 4.18: A sample of spectra drawn from the first design set. Noting that one time point spans 12 hours, time point 61 is at the end of day 30.

Having confirmed that T_p has a sensible interpretation in this case we proceed to construct an emulator, initially using the same structure as that for H_s with a small nugget value of 1.8×10^{-5}. This was found to be very effective, which is likely due to high correlation between H_s and T_p. Recall that we already saw that there was high correlation between H_s and T_e in the last experiment. Graphical diagnostics are not shown for brevity but they revealed almost exactly t-distributed residuals with no outliers and the Mahalanobis distances were 58.3, 53.6 and 65.7 for three validation sets. The emulator was used to perform the same uncertainty analysis as for the H_s data, the results of which are shown in figure 4.19. The histogram in panel (a) indicates a distribution very close to normal. The difference when the input wind speed variance is removed, shown in panel (b), is however quite dramatic. There is a 65% variance reduction in the output, indicating the strength of influence of the wind, and showing that uncertainty about the simulator tuning has less of an impact on predicted T_p than H_s. The sensitivity of the uncertainty distribution to the wind speed is also highlighted in panel (c) where the use of a uniform prior distribution causes considerable broadening. If wind were not influential we would expect to see little difference between the three figures. A final point is made regarding the tail at

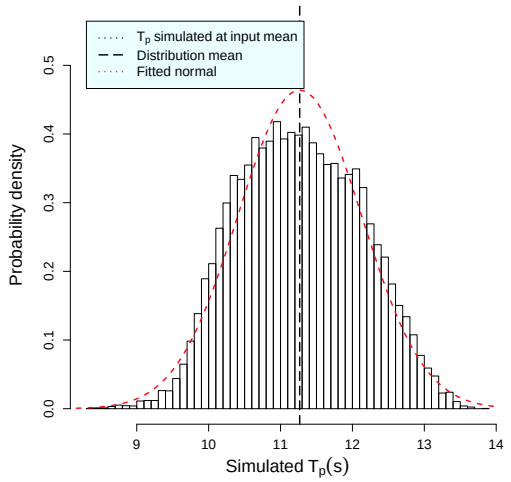
the design space, at location [0.0759,0.94,0.090,0.0095,0.18,0.091,0.040,0.013,0.98]



(a) Uncertainty distribution for T_p.



(b) Uncertainty distribution for T_p with wind speed variance reduced by 99%.



(c) Uncertainty distribution for T_p with wind speed distributed as $\sim unif(9.0, 11.0) ms^{-1}$.

Figure 4.19: Uncertainty distributions induced in simulated T_p with three different input uncertainty specifications for wind speed.

the lower end of the distribution in panel (b). Having removed the dominant effect of the wind, the more nonlinear effects of other tuning parameters become evident. The implication is that if the uncertainty about wind speed was very small, for example if very accurate measuring equipment or simulations were available, then nonlinearity within the wave simulator could become important (although the overall variance would clearly be much reduced).

This analysis concludes experiment 4.B. We have seen that even where eight important tuning parameters are regarded as uncertain, these account for a fairly small proportion of the total uncertainty about the prediction of H_s and T_p in fully developed conditions. The single largest contribution is due to uncertainty about wind speed, where an input normal distribution with a s.d. of only 0.5ms^{-1} accounts for 54% and 65% of variance in simulated H_s and T_p respectively. Nonlinear behaviour in the wave simulator was found to be largely absent. Before concluding this chapter, we briefly explore how an emulator can be used to visualise simulator response to input variability.

4.7 Further applications of an emulator: Visualisation

In this short section a description is provided of how an emulator can be used to visualise output from a simulator. Since the emulator is orders of magnitude faster than the simulator, it is possible to take advantage of its capability of generating large amounts of data. One way of doing so is through visualisation. Many modern data analysis packages can project data in at least 3 dimensions. Here, the R[40] software package with the "rgl" library[41] can be used, together with the emulator mean function, to draw iso-"response"-surfaces in 3-D space. That is, for three different input parameters, the full parameter space can be explored yielding a 3-D map of the output space (H_s for example) and representing this with surfaces of constant output. Colouring the surfaces helps to show the response in the output. Figure 4.20 was produced in such a way making use of one of the emulators formulated in experiment 4.A. Experiment 4.A (a) examined the three input parameters *sdsa0*, *sdsa1* and *sdsa2*. The emulator mean function constructed in section 4.4 is therefore a function of these three variables. Each of the three axes therefore corresponds to an input parameter but note that values of the input parameters re-scaled to between 0 and 1 are shown on the axes. The plotting procedure locates the iso-surfaces through an optimisation process that requires many executions of the target function, which would be rendered prohibitive without the speed of the

[40] http://www.r-project.org/

[41] https://r-forge.r-project.org/projects/rgl/

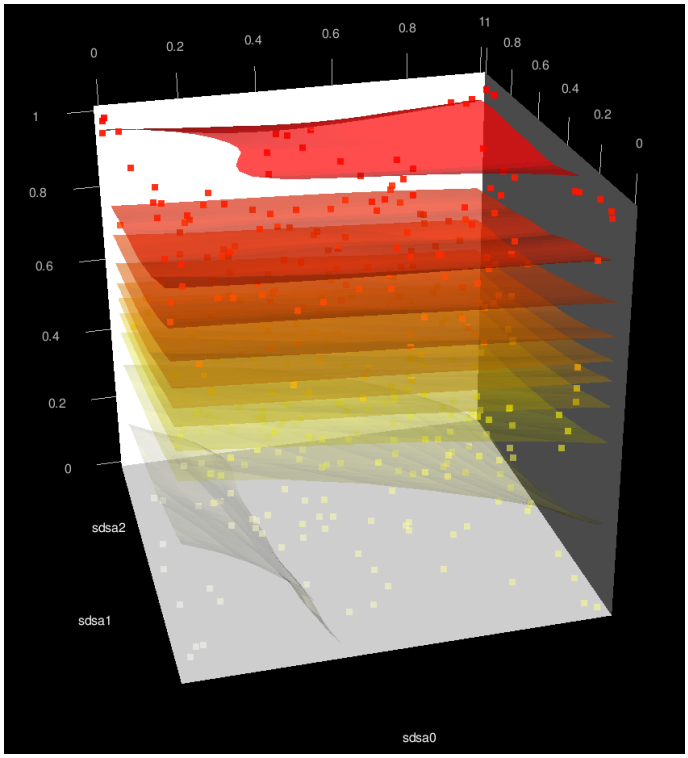


Figure 4.20: Simulator response (in H_s) to variation in input parameters. Input parameters are assinged to the spatial axes and the colours white to red represent values of H_s from low to high.

emulator.

Since the visualisation uses the OpenGL[42] system, the plot can be moved and rotated to examine any interesting features. In figure 4.20 we see from the colour variation and iso-surfaces of constant values of H_s, that there is a broadly linear variation with change in each parameter. However their interactions do lead to some nonlinearity, particularly for low values of *sdsa2*. This can be compared with the findings of the regression diagnostics shown in figure 4.4. The diagnostics revealed that the simulator had a nonlinear response to input *sdsa2* (bottom left panel) and we can in fact see that response in figure 4.20 as *sdsa2* (on the vertical axis) increases.

With some extra work this approach could be extended to allow exploration of a much higher dimensional space. "Browsing" high dimension spaces is not necessarily very useful since the spaces are very large but visualisation can be very helpful where we are interested in specific parameters, or regions of the input space. In the next chapter the main effects of individual inputs are visualised which is a very powerful means of understanding and determining their influence.

[42] http://www.opengl.org/

4.8 Chapter summary

In this chapter we have seen Gaussian process emulators applied to simple simulations of waves produced by WW3. A number of important findings have been made which are discussed below.

4.8.1 Emulation

Emulators have been shown to be an effective method for performing uncertainty analysis for simple configurations of WW3. In both experiments 4.A and 4.B emulators were found to validate based upon a range of statistical tests. Furthermore, in experiment 4.A (a) direct comparison of uncertainty distributions generated by an emulator and directly from the simulator revealed almost identical results. From this we can draw confidence that emulators can be applied to more complex simulations of waves. A final cautionary note is made regarding the experiment design because we have seen that the initial choice of parameter space (see experiment 4.B) gave rise to highly unrealistic output. In addition, we have seen (in figure 4.15) that as parameters are pushed towards their extremes, the uncertainty in simulator output can exhibit non-constant variance. We must be cognisant of such issues.

4.8.2 Uncertainty analysis for uncertain parameters and wind speed

In experiment 4.B, uncertainty analysis was performed for simulated H_s and T_p in fully developed conditions, accounting for uncertainty in the six tuning parameters that govern the high and low frequency dissipation for the TC96 input and dissipation parameterisation, and two parameters that govern the tuning of the DIA scheme for wave-wave nonlinear interactions. Wind speed was also regarded as uncertain, giving rise to uncertainty distributions in simulated H_s and T_p summarised by mean values of 3.83m and 11.26s and variances of 0.392 and 0.607 respectively.

Importantly, an input uncertainty for wind speed defined as $N(10.0\text{ms}^{-1}, 0.5^2)$ contributed 54% and 65% of the output uncertainty for H_s and T_p respectively. This result is a good initial indication of the relative importance of the wind input. Given this result, it is reasonable to expect that wind uncertainty might dominate in a similar way for other more complex cases.

Having made an initial assessment of the influence of uncertainty in tuning parameters and forcing wind speed, in the next chapter this investigation is continued in the context of fetch-limited wave growth.

5 Uncertainty analysis for propagating waves in a channel (1-D)

5.1 Introduction

In the previous chapter wave simulations were run in a single grid cell under duration-limited conditions, in order to explore the effects of uncertainty on input parameters associated with energy dissipation and nonlinear wave-wave interactions. In some realistic cases, such as a large lake where waves are almost entirely due to the local wind conditions (i.e. not arriving from distant locations), it may be possible to use experiments similar to those in the previous chapter, involving only the source functions without propagation, for fore- or hindcasting. However for most practical applications associated with the open ocean it is necessary to propagate energy from place to place. WW3 achieves this by solving the energy balance equation (5.1) complete with advection terms.

$$\frac{\partial E(f,\theta)}{\partial t} + \mathbf{c_g} \cdot \nabla_{\mathbf{x},\mathbf{y}} E(f,\theta) = S_{\text{wind}} + S_{\text{nl}} + S_{\text{ds}} \tag{5.1}$$

In this chapter, spatial propagation is introduced by considering a number of cases related to fetch-limited wave growth (see section 2.4.2). The simulation of fetch-limited wave growth can be done utilising only a single spatial dimension, thus the investigation of this phenomenon provides a useful way of introducing energy propagation, and its associated complexity, while maintaining a fairly simple physical configuration. The experiments presented in this chapter involve the application of emulators to investigate uncertainty on parameter tuning and wind forcing, in much the same way as the previous chapter. The wind forcing in this case is now expressed in a slightly more complex manner. That is, even in a single spatial dimension with constant conditions (effectively independent of time), the wind must still be expressed as a function of space, and uncertainty about the wind must therefore be characterised and appropriately expressed. Specific details about uncertainty associated with forcing wind, input parameters and numerical error are discussed in detail in section 5.3. The scope of the analysis is also extended in this chapter to include probabilistic sensitivity analysis as described by Oakley and O'Hagan (2004) and summarised in chapter 3. This provides a formal and comprehensive evaluation of the influence of input parameters and any interactions.

The scope of this chapter is to derive uncertainty measures for the summary statistics H_s and T_p for a number of scenarios of interest related to fetch-limited conditions. In section 5.2 the physical arrangement, relevant physics and sources

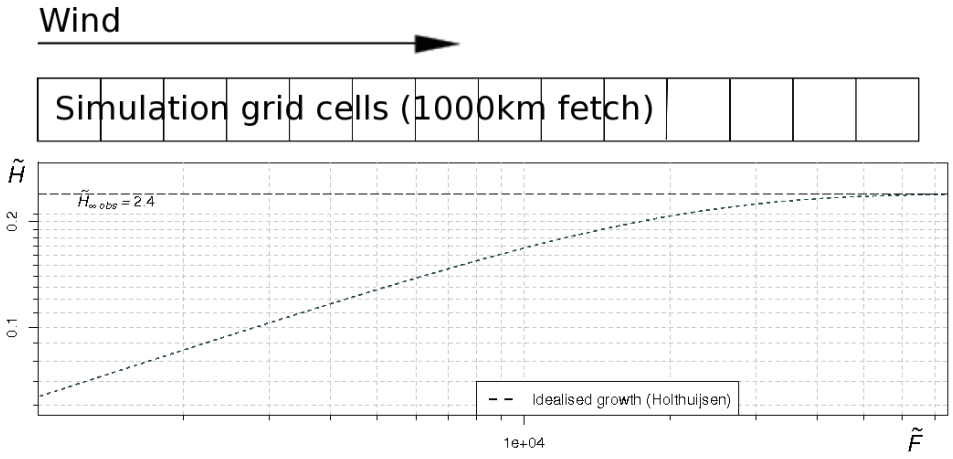


Figure 5.1: Geographic arrangement for a simulation of fetch-limited wave growth, shown with a graph of dimensionless wave height plotted against dimensionless fetch.

of uncertainty are discussed. Section 5.3 details the sources of uncertainty that are specifically considered. Section 5.4 describes the experimental configurations used to perform the analysis, and the results obtained. Experimental results are described in sections 5.5, 5.6 and 5.7 before key findings in this chapter are summarised in section 5.8.

5.2 Fetch-limited wave growth

Fetch-limited wave growth is described in section 2.4.2 but some points are discussed further here in relation to conducting these experiments. As previously described, we have a situation where wind blows steadily, perpendicular to the coastline, over the land and then out to sea. The distance out to sea, or fetch, over which the wind has blown, limits how developed the wave conditions can become assuming that the wind has been blowing continuously for sufficient time. The sea state is thus independent of the duration that the wind has blown, limited only by the fetch. In terms of investigating this through simulation, a geographic arrangement similar to that depicted in figure 5.1 is used. The simulation configuration is comparable to a very long deep water channel. The spatial boundaries are effectively perfect absorbers, so no wave energy crossing the boundary returns, and no wave energy enters at the boundary. The channel is split into a number of grid cells (typically 50, each of 20km in this chapter) with a wind blowing parallel to the direction of the channel. The lower part of figure 5.1 shows a typical profile of H_s along the fetch. The axes show the dimensionless quantities $\tilde{F}$ and $\tilde{H}$. These were defined in section 2.4.2 and provide a wind speed invariant representation of the wave growth. Given a wind speed of 12.5ms^{-1}, the non-dimensional fetch corresponding to 1000km is

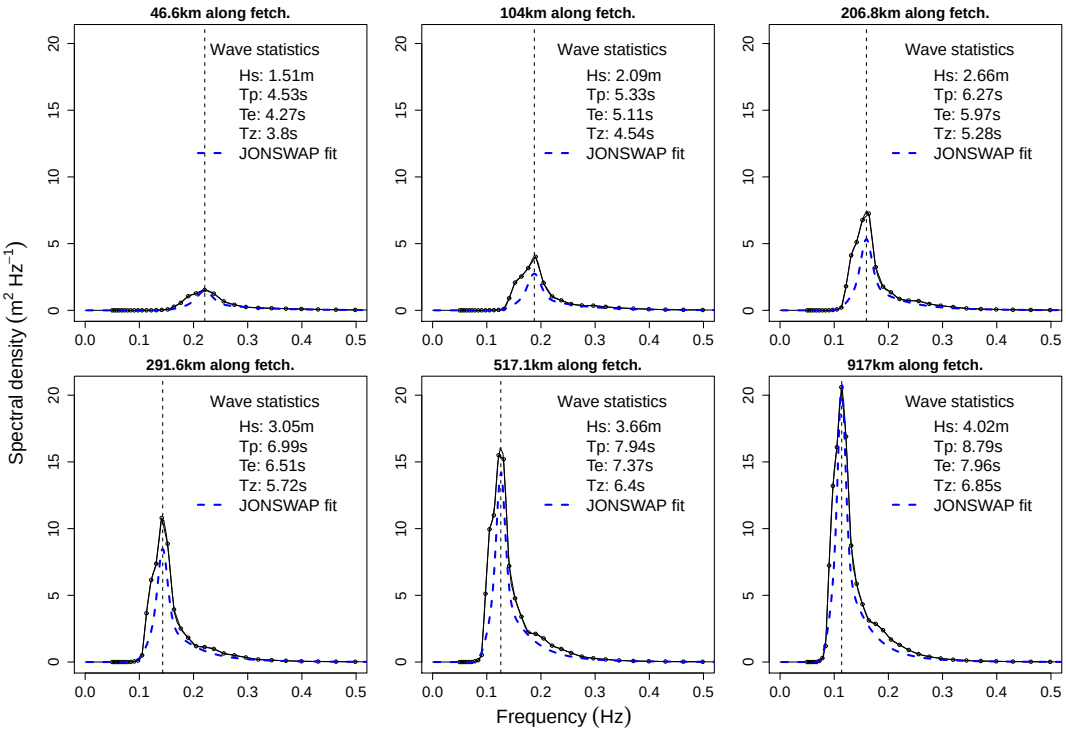


Figure 5.2: Evolution of 1-dimensional wave spectra along the fetch.

approximately 6.3×10^4. Figure 5.1 shows that this corresponds very closely to fully developed conditions (denoted by $\tilde{H}_\infty = 2.4$). Similar growth curves can be obtained for other summary statistics such as T_p. These growth curves derive from the growth of the 1-D spectrum, an example of which, generated by WW3 using default settings, is shown in figure 5.2. The magnitude of the spectral peak increases with fetch as wave energy is accumulated but, through the process of nonlinear wave-wave interactions, energy is transferred to lower frequency waves, and hence the peak period decreases as the sea state develops. For comparison, the spectrum as parameterised in terms of fetch by the functional form derived through the JONSWAP experiment (Hasselmann et al., 1973), is overlaid. The fit is generally good but the simulation seems to over estimate the width of the peak at shorter fetch. This is probably attributable to a focus on getting better performance at long fetch for a global simulations. Examples of poorer performance of WW3 at short fetch have been documented (e.g. Ardhuin et al., 2007).

In this chapter, analysis will not explicitly consider the 1-D spectrum but will focus on the growth curves for wave summary statistics as seen in figure 5.1. Some authors (see e.g. Breugem and Holthuijsen, 2007) have opted to characterise such growth curves by means of fitting empirical functions. Breugem and Holthuijsen

(2007) propose a set of equations as follows,

$$log(\tilde{H}) = \tilde{H}_{\infty}[\tanh(k_1 \ \log(\tilde{F})^{m_1})]^p \tag{5.2}$$

$$log(\tilde{T}) = \tilde{T}_{\infty}[\tanh(k_2 \ \log(\tilde{F})^{m_2})]^q \tag{5.3}$$

Each is characterised by four coefficients, e.g. $\tilde{H}_{\infty}$, k_1, m_1 and p, which are determined by fitting the function to data using a regression method. Note that in the following experiments, p and q are assumed known and equal to 0.572 and 0.187 respectively (consistent with Breugem and Holthuijsen, 2007). The dimensionless quantities $\tilde{F}$, $\tilde{H}$ and $\tilde{T}$ are computed from (2.12), (2.13) and (2.14),

$$\tilde{F} = \frac{gF}{U_{10}^2}$$

$$\tilde{H}_s = \frac{gH_s}{U_{10}^2}$$

$$\tilde{T} = \frac{gT_p}{U_{10}}$$

Breugem and Holthuijsen (2007) used a least-squares approach to fit expressions of the form (5.2) and (5.3) to a variety of observational data from experimental studies. They obtained values for the coefficients $\tilde{H}_{\infty}$, k_1, m_1, $\tilde{T}_{\infty}$, k_2 and m_2 in order to characterise the wave growth. By doing so, the spectral information (that is in terms of simulation, a spectrum at each point along the fetch) is reduced to only three coefficients. The three coefficients are not orthogonal, meaning there is correlation between them, which might hamper subsequent analysis. However, a justification for using a non-orthogonal basis may be to achieve better physical interpretation. In this case, the physics of the wave growth is complex and the use of a (fairly arbitrary choice of) empirical functional representation does little to enlighten us. Furthermore, Breugem and Holthuijsen (2007) do not offer any specific physical justification for their choice. Nonetheless, perhaps some broad insight can be gained so let us consider each coefficient in turn. $\tilde{H}_{\infty}$ correlates exactly with the maximum value of $\tilde{H}_s$, and therefore also total wave energy. Its interpretation is therefore equivalent to H_s in fully developed conditions, which is in fact what we saw in chapter 4. Unfortunately k_1 and m_1 are not so easily interpreted. Owing to the functional form, they are correlated with each other, and both contribute to the gradient of the (log of) growth as seen in figure 5.1. Given that the growth region shows a power law relationship, it may be more useful to regard that region separately and perform analysis with respect to the gradient alone. (This kind of

approach is taken in experiment 5.C). It is difficult to say much more without further analysis, but we might however expect there to be considerable correlation between the results of any uncertainty or sensitivity analysis for each of these coefficients.

Note that this particular representation is used in experiment 5.A only, where inference for parameter uncertainty is performed with respect to these coefficients as generated by the wave simulator. Subsequent experiments in this chapter consider summary waves statistics at specific points along the fetch (5.B), and make use of alternative parameterisations using simple regression methods (5.C). These are discussed in the relevant experiments.

5.2.1 Variations on fetch-limited growth

In reality many variations of idealised fetch-limited growth can occur. Researchers have considered the case of "slanting" fetch for example (Ardhuin et al., 2007), where the wind blows at an angle to the shoreline. Due to the "along-shore" component of wind, this effectively creates waves corresponding to longer fetches along tracks perpendicular to the shore. In keeping with the experimental approach adopted in this chapter and making use of the single channel configuration, other possible variants include modified wind profiles, where the wind is seen to decay as a function of fetch. Imposing the condition that the wind remains constant in time, different kinds of wave growth profiles are created. An experiment to investigate an example of this is conducted in sections 5.7 where the wind speed is configured to decay with distance.

5.3 Sources of uncertainty

5.3.1 Parameter uncertainty

Parameter uncertainty is specified in the same way as for experiments in the previous chapter. No new sources of parameter uncertainty associated with wave physics are introduced. The TC96 input and dissipation scheme remains unchanged together with the DIA scheme for the nonlinear wave interactions source term. The distributions assigned to the input parameters are as previously specified (see table 4.1). Since the input, dissipation and nonlinear source terms are now operating over a spatial domain, with energy being transported, the effects of uncertainty on the processes may differ from the previous findings.

5.3.2 Numerical uncertainty

Experiments conducted in the previous chapter do not involve energy propagation so the calculations are actually source term integrations only. Numerical uncertainty is introduced into the source term integration via the discretisation of the spectral domain. This was previously managed by adopting a (high) resolution of 36 frequency bins spanning the full range of the spectrum (alleviating the need for a long parametric tail). Furthermore, since the analysis data were essentially derived from the source function (i.e. there was no time dependence), then there is no contribution to numerical uncertainty from the choice of time step. The same arguments apply to the experiments conducted in this chapter but since we are now considering advection the relevant sources of numerical uncertainty are mentioned.

Numerics were discussed in section 2.2.4 in detail so the important points are briefly mentioned. Regarding the advection scheme, the third order "ULTIMATE QUICKEST" propagation scheme of WW3 is employed (see Tolman, 2009a). Tolman (1995) found this to be a robust scheme for a range of situations where numerics can cause trouble, but specifically it is able to compute stable fetch-limited growth. Therefore, making use of high spatial and spectral resolution, and choosing appropriate values of the time step to satisfy the CFL condition, numerical uncertainty should be small.

As in chapter 4, the spectral resolution corresponds to 36 frequency bins over a range from 0.050 to 0.67Hz, and grid cells are 20km long over a total fetch of 1000km. The spatial resolution is very much higher than would be adopted for an operational global simulation, where 1 degree of latitude is approximately 111km. Note that since we are investigating propagation in a single direction only, propagation in the perpendicular direction is deactivated in the simulator. This effectively results in homogeneous conditions (like chapter 4) in the perpendicular direction. Given that swell travelling at 20ms^{-1} would take 1000 seconds to cross a cell, the overall simulation time step is typically set to a very conservative value of 500 seconds, which still remains economical.

5.3.3 Uncertainty about wind forcing

For experiments 5.B and 5.C the wind specification is regarded as uncertain. The experimental arrangement requires that the wind speed is specified, with no spatial or temporal variation. Given this simple case we can regard the wind speed as not known precisely and express it as a Gaussian distribution. Noting the discussion of wind speed uncertainty in section 2.2.3, the distribution is specified as $windspeed \sim N(12.5\text{ms}^{-1}, 0.5^2)$. Experiment 5.C considers a wind that decays with fetch, where

Table 5.1: Experiments presented in chapter 5.

Experiment	Description
Experiment 5.A	Investigation of uncertainty about fetch-limited growth.
Experiment 5.B	Exp 5.A modified with a variable wind.
Experiment 5.C	Exp 5.B modified with a steadily decaying wind in space.

the initial wind speed is drawn from the same distribution.

5.4 Experiments

The experiments are listed in table 5.1 and are described below.

Experiments 5.A and 5.B consider the well studied case of fetch-limited growth of waves forced by a constant wind. This also includes consideration of uncertainty on the wind speed, as previously mentioned. Experiment 5.C augments 5.B by introducing a change in the wind profile such that the wind strength diminishes with fetch. In addition to output from uncertainty analyses, sensitivity measures are determined for input parameters which reveals their influence, any resulting nonlinearity and relative importance.

5.4.1 Design summary

Listings of input parameter sampling distributions for each experiment are given in their respective sections. Note that owing to the chronological development of this research, the input distributions used in experiment 5.A (given in table 5.3) differ slightly from those listed in table 4.1. Three or more LHS designs are created for each experiment. Typically the first one or two design sets are used as training data for emulator formulation with an additional design used for validation. Further designs are not always necessary but are cheap enough to produce in this case. They are typically used for further emulator validation or exploratory analysis.

5.4.2 Simulation configuration

All the experiments conducted in this chapter are configured with a wind blowing, constant in time, aligned to the direction of propagation, as depicted in figure 5.1. Propagation of energy perpendicular to the direction of the channel was disabled in the WW3 code. The water depth was set at 1km. To ensure that a steady state in time was achieved for the wave statistics along the fetch, simulations were run until the change in H_s (or T_p) was less than 0.1% per (hourly) time step. This was

Table 5.2: WW3 switch configuration for chapter 5

Swtich	Function
PR3	"ULTIMATE QUICKEST" 3rd order propagation scheme.
FLX2	Friction velocity computed according to TC96 scheme.
WNT1	Linear interpolation of wind in time.
WNX1	Linear interpolation of wind in space.

typically achieved within approximately 48 hours. Important switches relating to the configuration of WW3, that dictate model physics and propagation, are shown in table 5.2. See appendix E for a full switch listing.

5.4.3 Simulation configuration

5.5 Experiment 5.A: Fetch-limited growth

In this first experiment the effect of uncertainty about parameter tuning for TC96 dissipation on wave growth along the fetch is investigated. The approach is to make use of the parameterisation of the wave growth as described in section 5.2. Before discussing the analysis and results it is useful to inspect some output from the simulator. Output from single runs of the wave simulator, showing growth of H_s and T_p, using default tuning are shown in figure 5.3 and figure 5.4 respectively.

The simulation was forced with a wind speed of 12.5ms^{-1}, but the fetch and wave parameters have been scaled to dimensionless quantities. Output from three different simulations is shown to give an indication of variability associated with different numerical schemes and physics. Firstly, the first order scheme is shown for comparison but not routinely used for operational fore- or hindcasting with WW3[43]. The two other simulated data sets utilise the third order propagation scheme but differ in that one uses the DIA nonlinear interaction scheme and the other utilises the full nonlinear computation. Also included is the idealised curve as found by Breugem and Holthuijsen (2007) from observational data. The region of fetch-limited growth is apparent before fully developed conditions are reached near $\tilde{F} = 10^5$. Notably, there is a considerable difference between the three simulations, none of which very closely reproduces the idealised growth curve (given by the dashed line). Interestingly the poorest fit to the idealised case is the simulation run with the most accurate physics. However, this is explained by the fact that by default the TC96 input and dissipation scheme is tuned in conjunction with the (approximate)

[43]The WAM wave model actually employs a first order scheme.

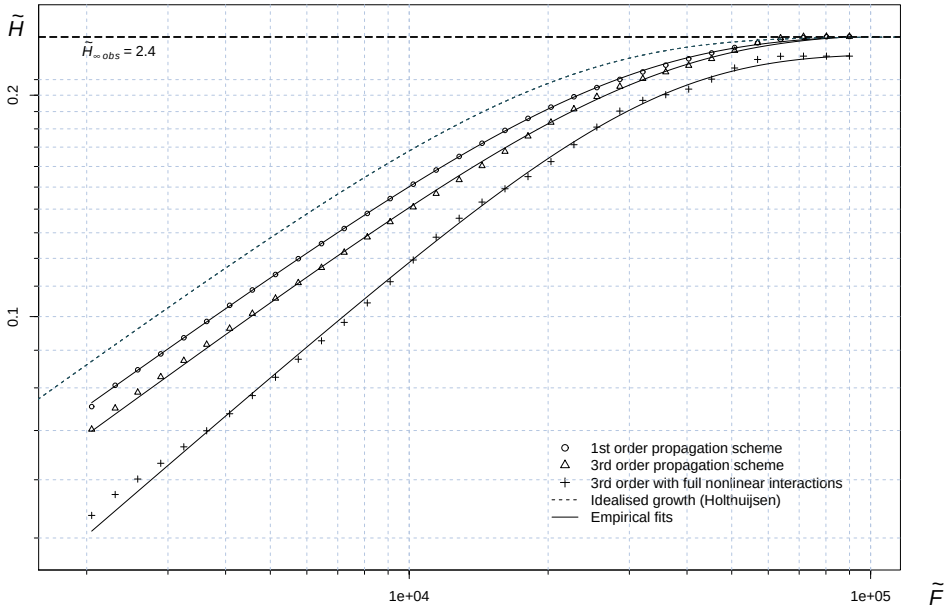


Figure 5.3: Simulations of fetch-limited growth of H_s for different numerical schemes and physics. Forcing wind speed is 12.5ms^{-1}.

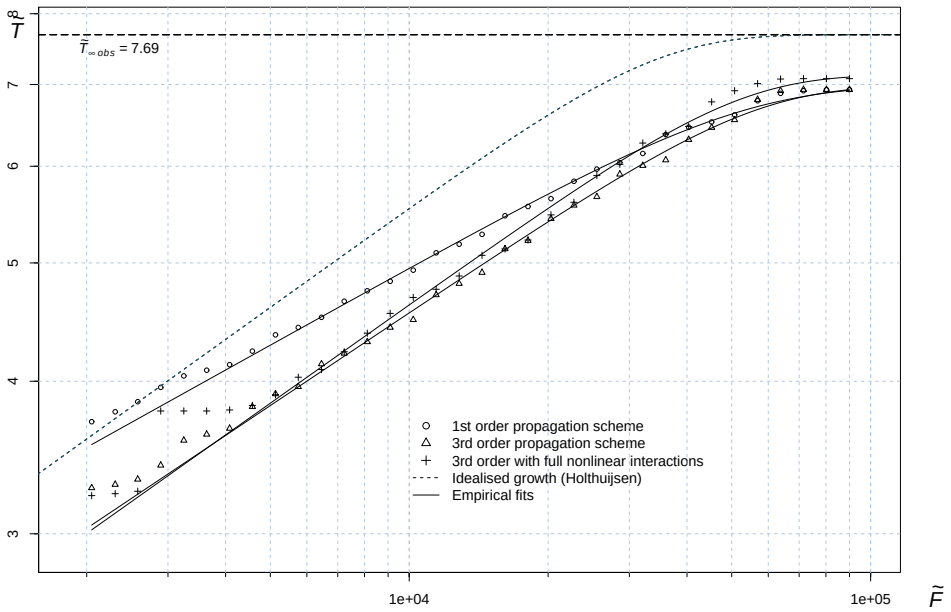


Figure 5.4: Simulations of fetch-limited growth of T_p for different numerical schemes and model physics. Forcing wind speed 12.5ms^{-1}.

Table 5.3: WW3 input parameters and sampling distributions for experiment 5.A

Switch	**Description**	**Tuning parameter** [**Variable Assignment in WW3 manual**]	**Default value**	**Sampling distribution**
ST2	Dissipation (TC96)			
	High frequency dissipation	SDSA0 [a_0]	4.8	$\mathrm{N}(4.8, 0.12^2)$
	High frequency dissipation	SDSA1 [a_1]	1.7×10^{-4}	$\mathrm{N}(1.7 \times 10^{-4}, (4.25 \times 10^{-6})^2)$
	High frequency dissipation	SDSA2 [a_2]	2.0	$\mathrm{N}(2.0, 0.05^2)$
	Low frequency dissipation	SDSB0 [b_0]	3.0×10^{-4}	$\mathrm{N}(3.0 \times 10^{-4}, (7.5 \times 10^{-6})^2)$
	Low frequency dissipation	SDSB1 [b_1]	0.47	$\mathrm{N}(0.47, 0.012^2)$
	Low frequency dissipation	PHIMIN [$\phi_{\mathbf{min}}$]	0.003	$\mathrm{N}(0.003, (7.5 \times 10^{-5})^2)$

DIA nonlinear scheme. It would not therefore be expected that simply "plugging in" the exact physics would immediately give an improved result. The other two cases show similarity and do fit the observational data better, in particular at longer fetch. It is well known that the default TC96 tuning is based upon obtaining better performance for longer fetch, which is clearly supported. The simulator appears to reproduce the growth of H_s better than the growth of T_p but again this is probably due to a focus on obtaining better performance for H_s, which is generally the case.

The experiment proceeds by generating three maximin LHS designs of 80 points over the uncertain parameters shown in table 5.3. Simulation output (in terms of H_s and T_p) is obtained for a total of 34 geographical points along the fetch, these can be seen in figure 5.3. For each run (or equivalently, design point), equations 5.2 and 5.3 are fitted using a nonlinear least squares fitting algorithm. Coefficients $\tilde{H}_\infty$, k_1, m_1, $\tilde{T}_\infty$, k_2 and m_2 then form sets of training data for emulators. A sample of output from six simulations from the first design set is shown in figure 5.5. The points, marked by circles, are output from WW3 and the solid black lines are the function (5.2) fitted using the least squares algorithm. In general it can be seen that the data conform well to the functional shape. Although the idealised function assumes a constant

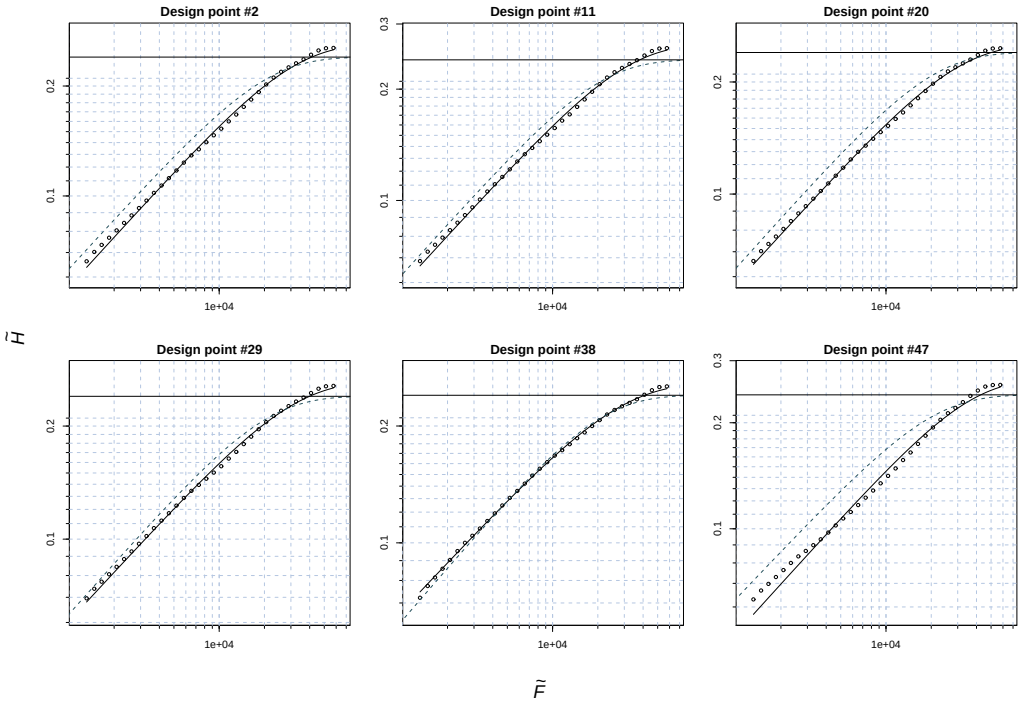
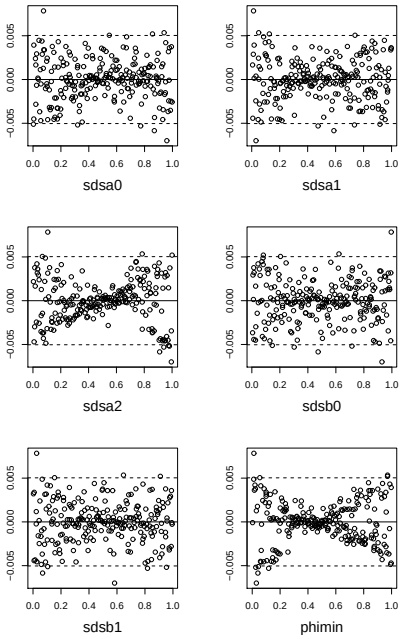


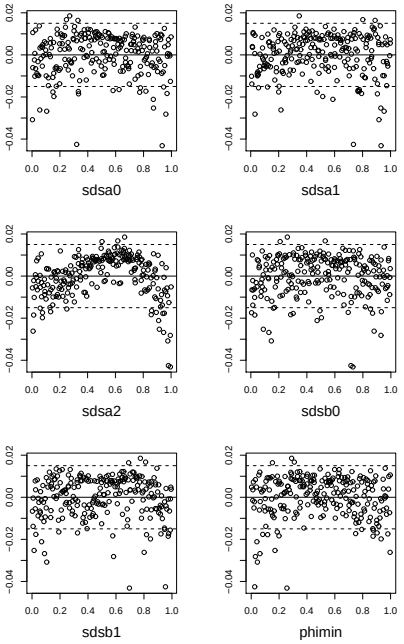
Figure 5.5: Function (5.2) fitted to output from a number of design points.

gradient in the growth region, this is not strictly reproduced by the simulations. This is seen most clearly for run #47. The gradient appears to increase slowly from short fetch before reducing again just before the sea state reaches asymptotic development. In contrast, the fitted function conforms very closely to the simulation output in the fetch-limited region for run #38. This highlights the loss of information due to the fitting process, and we can see that more information has been lost for point #47 than #38. For the analysis, the impact of this is not immediately clear but in this experiment the functional basis appears in general to be a good fit to the data.

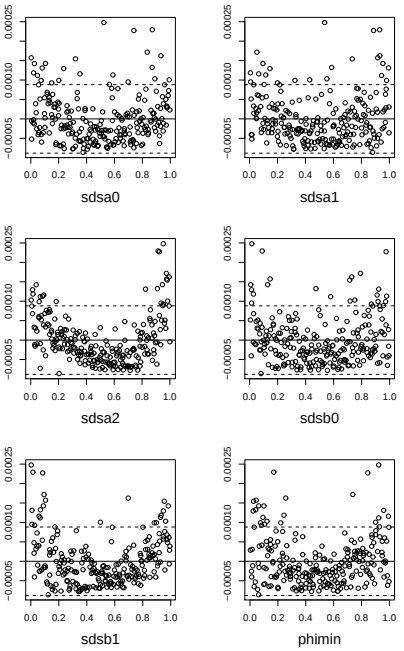
Having obtained sets of the six coefficients (three for growth of H_s and three for T_p), emulators are trained to each. We proceed by using multiple regression to determine trends in the response of the training data to variation in the input parameters. Plots of residuals from a linear model for each input parameter (six in total), for each of the sets of coefficients ($\tilde{H}_\infty$ and k_1, m_1) shown in figures 5.6 (a), (b) and (c) reveals linear response to the input parameters for coefficient $\tilde{H}_\infty$ but some nonlinear response for m_1, k_1. The residuals plotted against *sdsa2* and *phimin*, for the regression of the $\tilde{H}_\infty$ data (figure 5.6 (a)), exhibit heteroscedasticity. The variance clearly diminishes from its maximums at the extremes of the input value range and reaches a minimum at the centre. This could suggest that the parameters were not intended to be adjusted much outside of their central range because



(a) Residuals from linear regression of growth coefficient $\tilde{H}_\infty$



(b) Residuals from linear regression of growth coefficient m_1



(c) Residuals from linear regression of growth coefficient k_1

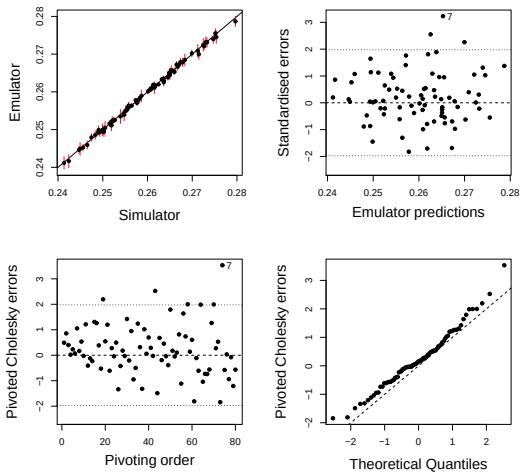
Figure 5.6: Residuals from linear regression of input parameters and wave height growth coefficients.

the simulator response becomes more pronounced. Interestingly the relationship between parameter *phimin* and the residuals reveal a structure that appears to show divergence, particularly at the lower end of the range where an isolated cluster of points is visible. Again this may be symptomatic of setting the parameter to a value well outside of its intended range, thus resulting in spurious simulator output.

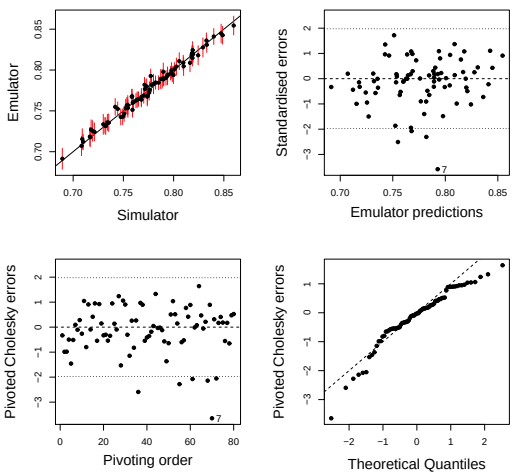
After further regression testing the following choices of emulator prior mean functions were chosen. (For convenience, input parameters are denoted by x with the appropriate subscript.) $\tilde{H}_\infty \sim 1 + x_{[3,5,6]}$, $m_1 \sim 1 + x_{[1,3,5,6]} + x_3^2 + x_3^3$ and $k_1 \sim 1 + x_{[1,3,5,6]} + x_3^2 + x_3^3$. Observe that no mean function has dependence on x_2 or x_4 (corresponding to *sdsa1* and *sdsb0*). This is indicative that the output is largely insensitive to those inputs. It is also interesting to see from figures 5.6 (b) and (c) that the response of m_1 is essentially the inverse of k_1. This suggests that it would be fairly straightforward to represent the wave growth data with only two coefficients rather than three, since m_1 and k_1 are highly correlated. This also seems consistent with the point mentioned earlier that fetch-limited growth region is essentially governed by a power law and therefore could be represented by just a straight line - that is, that only two degrees of freedom are required for the representation. A similar approach is considered in experiment 5.C where wave growth is parameterised using a linear regression model.

Emulators were constructed for the three coefficients using the aforementioned prior mean functions and Matérn ($\nu = 3/2$) covariance functions in all cases. The second design set was used to validate each emulator and the output from the graphical diagnostics is shown in figure 5.7. Some outlying points are apparent, likely due to some localised non-stationarity or heteroscedacity (observed in the linear regression, see figure 5.6). Nugget terms ($\nu = 1.5 \times 10^{-8}, 5 \times 10^{-6}, 5 \times 10^{-4}$) are used in order to improve the posterior variance estimate for some outlying points although validation point number 7 can be seen as an outlier in all diagnostics. In effect the nugget term compensates for the modelling assumptions (e.g. stationarity) being too strong for the data. Furthermore, owing to numerical problems with the covariance matrix coefficient, k_1 required log-transformation when formulating the emulator. This is probably attributable to the very small values of the variance obtained ($\simeq 10^{-8}$). The emulators exhibit observed Mahalanobis distances of 87.4, 77.5 and 77.1 and in general appear to be robust.

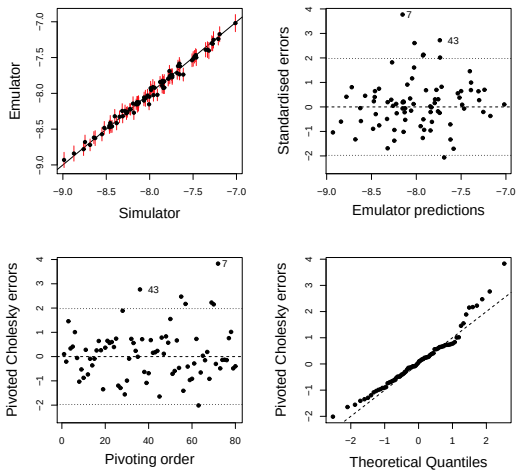
Analysis for T_p is approached in the same way. Emulators for each of the coefficients $\tilde{T}_\infty$, k_2 and m_2 were built with the same prior mean functions and covariance functions as for the H_s coefficients. Figures are not provided for the initial regression and emulator validation diagnostics however it was found that the response of the



(a) Growth coefficient $\tilde{H}_\infty$



(b) Growth coefficient m_1



(c) Growth coefficient k_1

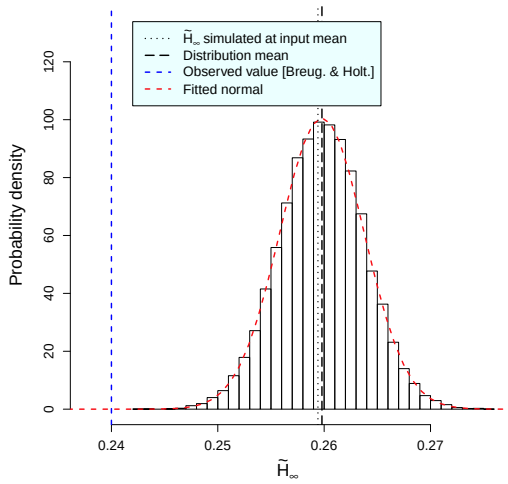
Figure 5.7: Graphical diagnostics for emulators of $\tilde{H}_\infty$, m_1 and k_1. Matérn ($\nu = 3/2$) covariance function used.

coefficients to input parameter uncertainty was similar.

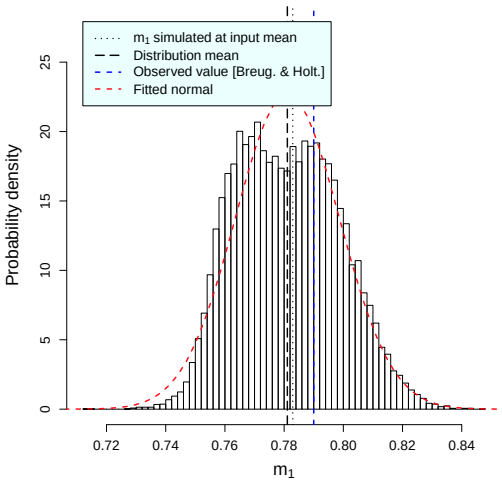
5.5.1 Uncertainty analysis

Having successfully constructed emulators, we proceed to perform uncertainty analysis. This was carried out by simulating realisations from the emulator mean function. The input uncertainty distributions are specified in table 5.3. Distributions resulting from the uncertainty analysis for H_s growth coefficients are shown in figure 5.8. The output distributions for $\tilde{H}_\infty$ and k_1 are fairly symmetrical suggesting broadly linear response. Output for m_1 shows slight bi-modality. More importantly, the variance for $\tilde{H}_\infty$ and k_1 is unable to explain the difference deviation from the observed values. For $\tilde{H}_\infty$ the distribution standard deviation is 0.004, compared with a mean of 0.26, and the real world observed value lies many standard deviations from the mean. $\tilde{H}_\infty$ has a clear physical interpretation and so we can get a sense of the magnitude of this difference. The difference between the mean of the distribution and the observed value is approximately 0.02. This is a dimensionless quantity but making use of (2.13), we find it corresponds to approximately 0.3m at a wind speed of 12.5ms^{-1}. Furthermore, the 2.5 percentile of the uncertainty distribution corresponds to $\tilde{H}_\infty \approx 0.248 \approx 0.19$m. This shows that the tuning uncertainty specified here cannot account for the discrepancy and that some other important sources of uncertainty have not been accounted for. One possible implication is that there are structural errors in the simulator. Another source of uncertainty not considered is that of the wind which is considered further in the next experiment.

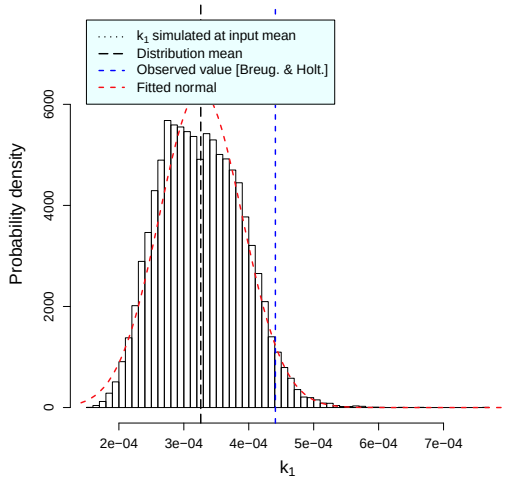
Similar behaviour of the simulator is seen in figure 5.9 for the growth of T_p, which is to a large degree anticipated because H_s and T_p tend to be highly correlated in idealised cases. In considering these results, firstly attention is drawn to the non-linearity in response evidenced by the skewed output distributions. This contrasts with the response of the coefficients relating to growth of H_s. More importantly, for $\tilde{T}_\infty$ in panel (a), the observed value as reported by Breugem and Holthuijsen (2007) is 7.69 which is well above any plausible value drawn from the uncertainty distribution. In fact, the distribution has a standard deviation of 0.07 which is less than 1% of the mean value suggesting that $\tilde{T}_\infty$ is broadly insensitive to the dissipation tuning. Figure 5.9 (a) indicates a very rapid decay in the upper tail suggesting higher values are in general much less likely. The results for m_2 and k_2 seem to be more consistent with the idealised observations, which lie within the ± 3 s.d. range of the output uncertainty distributions. Notably the mean value of k_2 lies very close to the observed value. Since there is correlation between the coefficients, the fact that k_2 alone appears to correspond well with observation is not strong evidence of good



(a) $\tilde{H}_\infty$

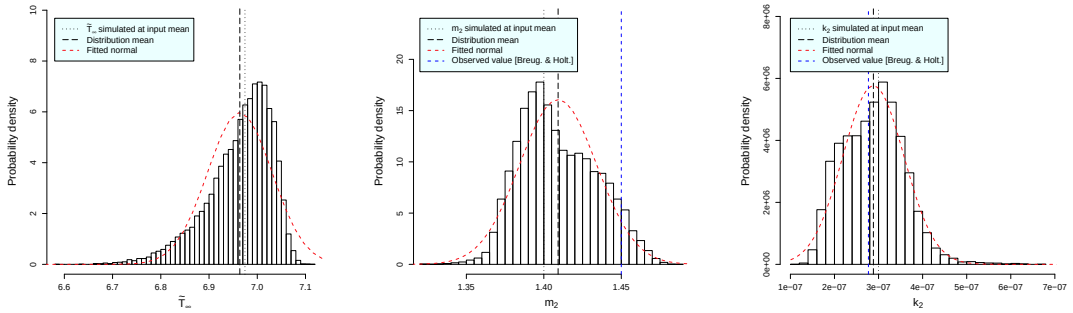


(b) m_1



(c) k_1

Figure 5.8: Uncertainty analysis for H_s growth coefficients.



(a) $\tilde{T}_\infty$ **(b)** m_2 **(c)** k_2

Figure 5.9: Uncertainty analysis for T_p growth coefficients.

performance of the simulator. The large discrepancy between simulated $\tilde{T}_\infty$ and the observed value is clearly indicative of some shortfall. One might therefore seek to account for other sources of uncertainty that specifically influence the response of $\tilde{T}_\infty$ without drastically affecting m_2 and k_2, in order to help explain the discrepancy. However, as noted earlier, the discrepancy may be due to structural error and it may be prudent to improve the representation of the physics in order to find a systematic correction. As a final note on the comparison of the uncertainty analysis results for H_s and T_p there appears to be a stronger nonlinear response in T_p. Given that H_s and T_p are typically highly correlated in cases such as this, it seems unlikely that the differing responses have a physical basis. The dissipation parameterisation is clearly introducing nonlinearity, although owing to the empiricism used in the formulation of the parameterisation, and the difficulty in interpreting the coefficients in a physical sense, drawing more robust conclusions is challenging.

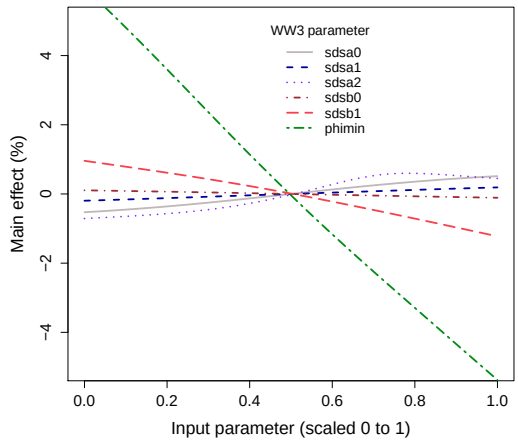
5.5.2 Sensitivity analysis

Before moving on to the next experiment the analysis can be extended by running a sensitivity analysis. This determines quantitatively which input parameters contribute to the variance. The initial regression exercise established that variation in *sdsa2* induced a nonlinear response but did not tell us about the magnitude of the response. Recall from section 3.3 that the *main effect* of an input parameter x_1 is the expectation of the output distribution induced by the (joint) input distribution conditional on x_1. The main effect $z_i(x_1)$ is therefore a function of x_1. Expressed mathematically the main effect $z_i(x_i)$ for input x_i is,

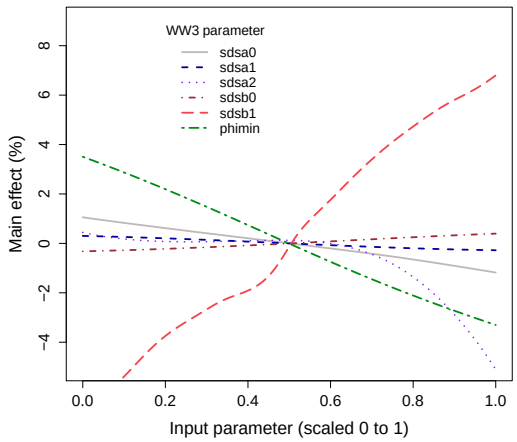
$$z_i(x_i) = E(Y|x_i) - E(Y) \tag{5.4}$$

where Y is the simulator output distribution and the expectation is with respect to the joint input distribution $g(x)$. (See also equation 3.5). The integrals can be computed numerically by simulation using the emulator mean function but we also saw in section 3.4.4 that such sensitivity measures could be computed by analytical integration of the emulator posterior mean and covariance functions. O'Hagan (2011) provides details of how to obtain analytical expressions in the case of a Gaussian correlation function (which makes the mathematics a good deal more tractable) and generic (linear) mean function. However, O'Hagan (2011) also provides a number of general integrals which, it turns out, can be integrated numerically fairly rapidly. Using this approach an arbitrary choice of mean and correlation functions can be used. This approach is made use of here and throughout the rest of this thesis (details of the code is provided in appendix D). Inference for the main effects, computed using this method, are shown in figure 5.10. What we see in each panel are six lines each corresponding to an input parameter. The value on the vertical axis indicates the percentage change in the expectation of the output distribution, conditional on the parameter in question, the specific value (scaled to between 0 and 1 according to the design) of the parameter is read from the horizontal axis. Considering figure 5.10 (a), if we take *phimin* (designated by the green dashed line) as an example, it is clear that variation of this parameter induces the strongest response. The response is close to linear with a negative slope. Note that the x-scale is effectively different for each input, and for *phimin* $[0, 1]$ corresponds to $[2.25 \times 10^{-3}, 3.75 \times 10^{-3}]$. So as *phimin* is increased from 2.25×10^{-3} to 3.75×10^{-3}, the expectation of the simulator output conditional on *phimin* decreases almost linearly. Notice also that the range spans the entire design space of the emulator for *phimin*, the limits of which fall at 4 standard deviations from the mean of the input for the uncertainty and sensitivity analysis. In a sense this is slightly deceptive, because the probability assigned to values at the edge of the design (assuming a Gaussian distribution) is typically vanishingly small, and so contribution to $E(Y)$ is also small. In terms of interpreting figure 5.10, this means that the effect at the limits does not contribute significantly to the analysis[44]. If we consider a different input parameter such as *sdsa2*, it can be seen in figure 5.10 (b) that the response is nonlinear at the high end of its range, but the nonlinearity lies beyond 1 standard deviation (a scaled value of 6.25) from the mean and so has a diminishing effect. By contrast the central region that is within 1 s.d. exhibits a linear response, which is very weak. Considering the effects on $\tilde{H}_\infty$, it can be seen immediately that, with the exception of *phimin*, the input

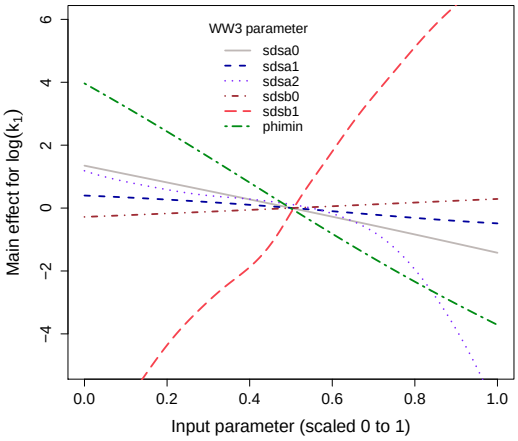
[44]It is for this reason that in general if the emulator performance is poor only for one or two design points at the very edge of the design, we need not be concerned as these effects do not contribute to subsequent analysis.



(a) $\tilde{H}_\infty$



(b) m_1



(c) $log(k_1)$

Figure 5.10: Sensitivity analysis for the coefficients describing the functional representation of growth of H_s

parameters have very little influence at all. Even at the limits of their respective ranges the main effects are 1% or less. Given the low rate of change associated with each parameter we can conclude that in order to induce an effect large enough to obtain simulation output close to the observed value (of 0.24), a sizable perturbation would be necessary in any given parameter. As previously stated, on the basis that the input parameters were not intended to be perturbed by such amounts, it is likely that structural errors are significant.

Interpretation of figure 5.10 (c) has to be mentioned in particular because the analysis is with respect to $log(k_1)$. Recall that some numerical problems were encountered when building the emulator, owing to the very small values of the variance. Since inference has therefore been made with respect to $log(k_1)$ interpretation is less straightforward and results should be treated cautiously. Unlike certain inference measures such as the expectation and variance, the sensitivity measures derived here cannot simply be reverse transformed and expressed in terms of k_1. For example, the main effects will be substantially larger than the approximate range of $\pm 6\%$ (although clearly not $exp(6) \approx 400\%$). What is notable from figure 5.10 (c) is the similarity with figure 5.10 (b). m_1 and k_1 actually have an inverse correspondence but since values of k_1 lie between 0 and 1, the log function inverts it, and as a result the response is very similar to m_1. An alternative way of proceeding would be to draw samples from the emulator mean function using a Monte Carlo approach and compute the sensitivity measures from the transformed output[45]. Main effects for k_1 computed in such a way are shown in figure 5.11. We have recovered the "inverse" behaviour with respect to m_1 and as expected the effects are large with respect to the mean value of k_1. However, given the lack of a clear physical interpretation for k_1 the results still require some consideration.

What we do not see with this kind of plot is the response due to interaction of different input parameters. More is said about this shortly, after discussing the parameter sensitivity measures. In section 3.3 we saw that the total variance induced in the output could be decomposed in terms of the variance due to the effect of each input. A measure $W_i = \text{var}\{z_i(x_i)\}$ was defined that is the amount of variance explained by learning the true value of x_i. These measures are computed in a similar way to the main effects, that is by direct numerical integration of emulator posterior mean and covariance functions. Bar plots showing W_i, or equivalently

[45] One difficulty when computing the integrals numerically (via simulation) is that even for an emulator coded in a package such as R or Matlab, the number of simulations required is computationally demanding. A way around this is to use, for example, the DICE Kriging package (http://cran.r-project.org/web/packages/DiceKriging/index.html) for R which is coded directly in C and so runs very much faster than interpreted code written within the R environment.

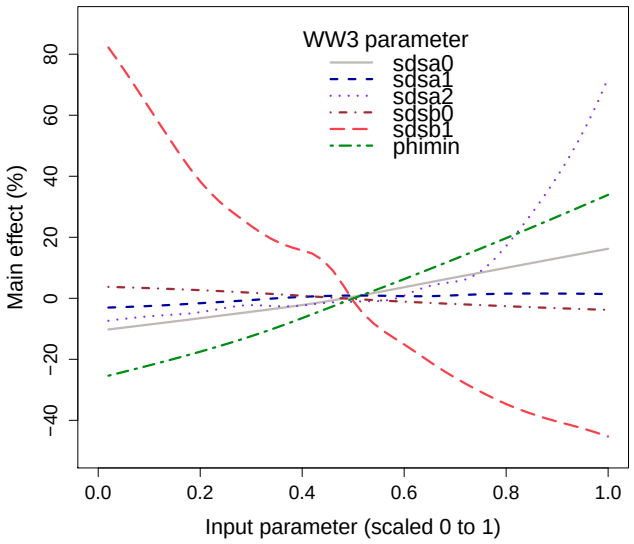
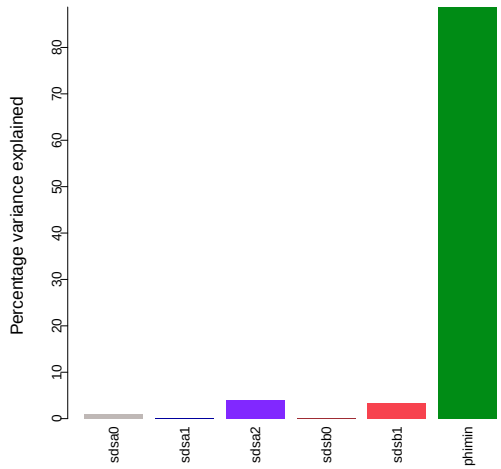
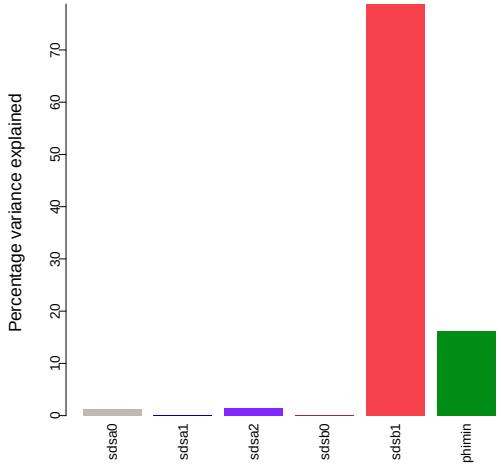


Figure 5.11: Main effects for k_1 computed by Monte Carlo from the emulator mean function.

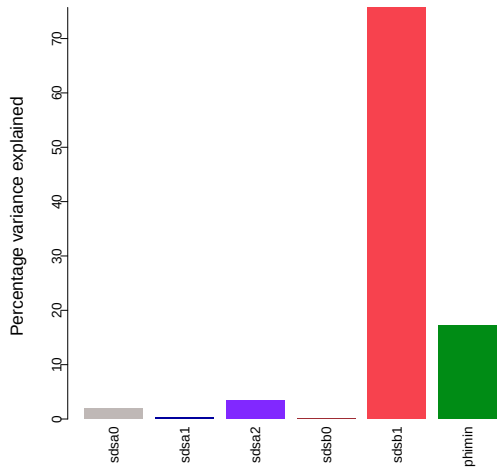
the percentage of variance explained, for each input are given in figure 5.12. The bar plots correspond well to the main effect plots shown in figure 5.10. That is, the parameters exhibiting the largest main effects contribute the most variance. In fact, *sdsb1* and *phimin*, which govern low frequency energy dissipation, dominate the variance. Given the correlation between m_1 and k_1 it is unsurprising that essentially the same response is observed. We can gain some physical insight at this point. Recall that $\tilde{H}_\infty$ corresponds to fully developed conditions (that is, the sea state beyond the region of fetch-limited growth). We can see that its variability, in terms of uncertainty about energy dissipation as governed by the input parameters under examination, is due almost exclusively to *phimin*. So in fact, were we to attempt to improve our simulation of H_s by obtaining output closer to the observed values (see figure 5.9), we could do this for $\tilde{H}_\infty$ by reducing the value of *phimin*. Furthermore, noting the correlation between the coefficients, we can see (quantitatively) what effect this would have on both m_1 and k_1. Following the logical process, we can also see that although m_1 and k_1 would both be affected by a change in *phimin*, they are both primarily governed by *sdsb1*. This potentially gives us a way of compensating for any change due to the adjustment of *phimin* without affecting $\tilde{H}_\infty$. Recalling the inverse relationship between m_1 and k_1, adjustment of *sdsb1* will increase one and decrease the other, so we would remain constrained by their correlation. Nonetheless, these results give us very clear insight into how uncertainty



(a) $\tilde{H}_\infty$

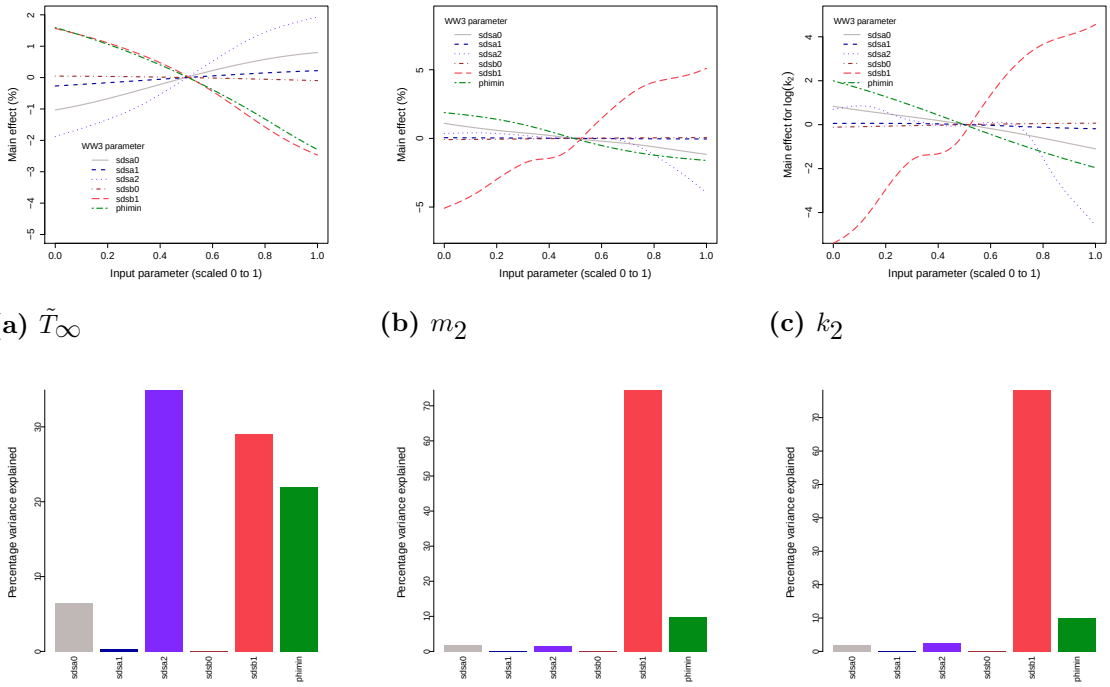


(b) m_1



(c) $\log(k_1)$

Figure 5.12: Proportion of variance in growth coefficients for H_s explained by input parameters.



(a) $\tilde{T}_\infty$ **(b)** m_2 **(c)** k_2

(d) Mean = 7.0; S.D. = 0.070 **(e)** Mean = 1.4; S.D. = 0.026 **(f)** Mean = 2.9×10^{-8}; S.D. = 7.8×10^{-8}

Figure 5.13: Sensitivity analysis for the coefficients describing the functional representation of growth of T_p

about the energy dissipation tuning parameterisation affects the wave simulator in this case.

A further and important point seen by inspection of the bar plots in figure 5.12 is that the variance explained sums to approximately 100%. The specific quantities show that the total main effect variances are 97.0%, 98.6% and 98.7% of the total variance (for $\tilde{H}_\infty$, m_1 and k_1 respectively). The remaining (small) proportion of variance is explained by interactions between parameters. Since the interactions are small, little would be gained from a further analysis so it is not presented here.

We now consider the results from the same kind of sensitivity analysis for the growth of T_p. Recall that the uncertainty distributions, seen in figure 5.9, for the output growth coefficients exhibited nonlinearity so we expect to see evidence of this in the results of a sensitivity analysis which are shown in figure 5.13.

Evidence of nonlinear response is shown in all three cases but most pronounced in panels (b) and (c) with respect to variation in *sdsa2* and *sdsb1*. *sdsa1* and *sdsb0* remain almost inactive. Notice that the response of $\tilde{T}_\infty$ differs from all the other cases seen so far because *sdsa2* actually accounts for the largest proportion of

the variance. Response of m_2 and k_2 are closely comparable to m_1 and k_1. Input parameters *sdsb1* and *phimin* account for the majority of the variance with 7.5% remaining unexplained.

This concludes the analysis for this experiment and so findings are summarised. Uncertainty analysis yielded output probability distributions for the maximum wave height and period at effectively infinite fetch ($\tilde{H}_\infty$ and $\tilde{T}_\infty$). These have a clear physical interpretation and by comparison with observational data, in figures 5.8 (a) and 5.9 (a), we get a sense of the "accuracy" of the simulator given the input uncertainty. Values of these coefficients derived from oceanic observations lie well outside of the range of the output distributions suggesting possible structural errors in the simulator, or that other important sources of uncertainty were not included in the analysis. The uncertainty analysis also generated distributions for the coefficients (m_1, k_1, m_2 and k_2), which seemed to be more commensurate with real world observed values. However, their physical interpretation is less clear. Although they express information about the rate of wave growth, and in a sense can be measured, it is not clear how they correspond to specific aspects of dissipative physical processes. That is, they are not related to quantities we might expect to be relevant such as viscosities, drag co-efficients, turbulence parameters or other similar components of a fluid system that could readily be measured or inferred. As such, they do not tell us much about the physics involved.

Some connection to the theory can be obtained by considering the coefficient $\tilde{H}_\infty$, which specifies the limit of growth at an effectively infinite fetch. Noting that wave height is essentially a measure of the total energy in the system, in a well developed sea state almost all the energy will be stored in the low frequency end of the spectrum (c.f. figure 5.2). Thus, dissipation of this energy will be important in governing the limiting wave height. It is therefore not surprising to find that the input parameter that governs the representation of the low frequency dissipation process, *phimin*, is most influential. TC96 describe *phimin* as a parameter that gives some control over the "full-growth" of waves at long fetch (see their section 5), consistent with findings here. Indeed, *phimin* is by far the dominant source of variance in the output distribution for $\tilde{H}_\infty$ contributing 87% of the total variance. Obtaining physical insight from m_1, k_1, m_2 and k_2 is more difficult because the sea state is a function of fetch, and these coefficients are derived from data along the whole fetch, which covers a range of sea states. In the next experiment simulation output at specific fetch is evaluated directly, rather than making use of any parametric representation. This will allow for the investigation of specific sea states (rather than some kind of average over fetch).

Finally, the results of the analysis of $\tilde{T}_\infty$ are highlighted because this coefficient did in fact exhibit sensitivity to parameters governing high frequency dissipation. Noting that $\tilde{H}_\infty$ and $\tilde{T}_\infty$ are typically well correlated, it might be expected that their sensitivity to parameters was very similar but that appears not to be the case. There are a variety of measures of wave period and some are more sensitive to higher frequency regions of the spectrum than others (e.g. T_z). We therefore might expect high frequency dissipation to be more relevant to younger sea states but given $\tilde{T}_\infty$ characterises a fully developed sea state, the influence of *sdsa2* is curious. However, we should also bear in mind that the total variance for $\tilde{T}_\infty$ is actually very small, with a standard deviation amounting to less than 0.1 (mean value 7.0).

The functional representation of wave growth is clearly a useful means of parameterising the data for fetch-limited growth but provides only limited scope for physical insight. Given that the sea state evolves along the fetch, with the peak period shifting to progressively lower values, it would be insightful to see how the sensitivities of H_s and T_p change with fetch. In the next experiment fetch-limited growth is analysed in a similar way but with consideration of specific points along the fetch. In addition, uncertainty about wind speed is incorporated in to the analysis.

5.6 Experiment 5.B: Fetch-limited wave growth with a variable wind

Experiment 5.A is now extended by regarding the wind speed as uncertain, together with the addition of tuning parameters associated with the DIA nonlinear wave-wave interactions. This is a natural extension given that we want to learn about the effect of wind uncertainty (albeit in this simple case), and also noting that the DIA scheme is so closely linked to both the TC96 input and dissipation parameterisation and good wave model performance.

5.6.1 Experiment configuration and design

The experimental arrangement is exactly the same as for experiment 5.A but with the inclusion of additional uncertain input. The experiment proceeds by performing analysis for specific locations along the fetch. Experiments evaluate the response of H_s and T_p at specific points, without using a dimensionless representation in this case. This choice is made in order to clearly identify the effect of wind uncertainty without scaling the output. Initially, six points along the fetch were chosen, at

distances of 52.3km, 104km, 260km, 367km, 517km and 917km[46]. The spacing of the points reflects the slowing rate of development of the sea state. Results of the analysis is presented only for fetches of 52.3km, 260km and 917km because there was little difference between other points. This is discussed further shortly.

Input parameter specifications for the uncertainty analysis in this experiment are shown in table 5.4[47] The introduction of uncertainty about the tuning of the DIA is a natural progression since the TC96 scheme was tuned in conjunction with a specific choice of DIA tuning. It is instructive to find out whether there is significant interaction between the two parameterisations. Regarding uncertainty about the wind, given the idealised case of wave growth under investigation, variability can only be characterised in limited ways. Most obviously, we can simply draw the wind speed from a Gaussian distribution. This ignores any possible spatial variation or "gustiness", but helps us investigate the notion that may be some average systematic bias in the wind measurement. As in the previous experiment, six LHS designs spanning the parameter space as specified in table 5.4 were generated.

5.6.2 Sensitivity analysis

Simulation output in terms of H_s and T_p at specific points along the fetch is used to train a number of emulators. Linear regression (not shown) revealed that mean functions of the form $H_s \sim 1 + x_{[2,3,4,5,7,8,9]} + x_4^2$ were appropriate and Matérn ($\nu = 3/2$) correlation functions were used. Indices for x correspond to the variables in the same way, with the addition of x_7, x_8 and x_9 corresponding to λ_{DIA}, C_{DIA} and *wind speed* respectively.

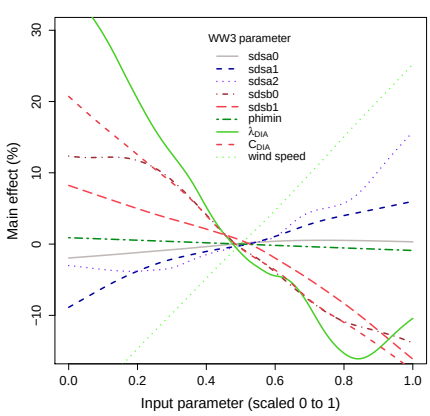
Results of the sensitivity analysis for output H_s at the three locations is shown in figure 5.14. Main effects for each input together with explained variance are shown. At a fetch of 52km, the distribution of H_s has a variance corresponding to a standard deviation of 0.19m. This however increases to 0.29m at a fetch of 207km, and to 0.43m at a fetch of 917km. So it appears that the uncertainty increases approximately linearly with wave height, and in this case is about 10% of H_s. Nonlinear response is evident, particularly due to λ_{DIA} at short fetch. The main effects, seen in 5.14 (a), (c) and (e) reveal the almost linear (and strong) influence of the wind input. The dominance of λ_{DIA}, C_{DIA} and *wind speed* is quite striking. What is also striking is that parameters associated with energy dissipation are now seen to have appreciable influence only at short fetch. At increasing fetch,

[46] The apparently irregular values stem from the logarithmic spacing specified in the WW3 output file.

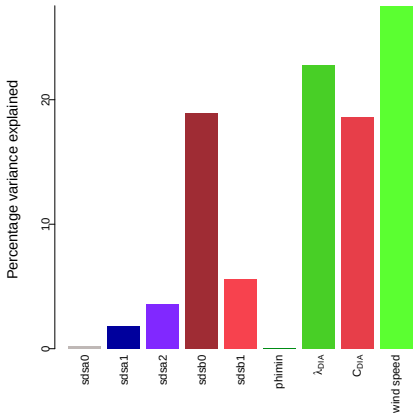
[47] This input specification is consistent with chapter 4 (shown in table 4.1), although with the reduction of the variance of λ_{DIA} made in experiment 4.B.

Table 5.4: WW3 input parameters and sampling distributions for experiment 5.B

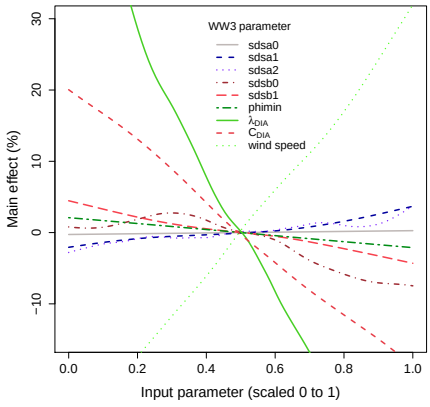
Switch	**Description**	**Tuning parameter [Variable Assignment in WW3 manual]**	**Default value**	**Sampling distribution**
ST2	Dissipation (TC96)			
	High frequency dissipation	SDSA0 [a_0]	4.8	$\mathrm{N}(4.8, 0.2^2)$
	High frequency dissipation	SDSA1 [a_1]	1.7×10^{-4}	$log_{10}\mathrm{N}(-3.293, 0.177^2)$
	High frequency dissipation	SDSA2 [a_2]	2.0	$\mathrm{N}(2.0, 0.2^2)$
	Low frequency dissipation	SDSB0 [b_0]	3.0×10^{-4}	$\mathrm{N}(-3.25 \times 10^{-3}, (9.375 \times 10^{-4})^2)$
	Low frequency dissipation	SDSB1 [b_1]	0.47	$\mathrm{N}(0.5250, 0.03125^2)$
	Low frequency dissipation	PHIMIN [$\phi_{\mathbf{min}}$]	0.003	$\mathrm{N}(0.003, (1.0 \times 10^{-4})^2)$
NL1	Nonlinear interactions (DIA)			
		LAMBDA [λ]	0.25	$\mathrm{N}(0.25, 0.03125^2)$
		NLPROP [C]	10^7	$log_{10}\mathrm{N}(7, 0.1193^2)$
	Wind speed	-	-	$\mathrm{N}(12.5\mathrm{ms}^{-1}, 0.5^2)$



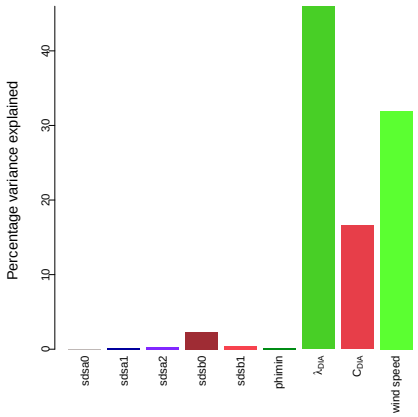
(a) Main effects at 52km



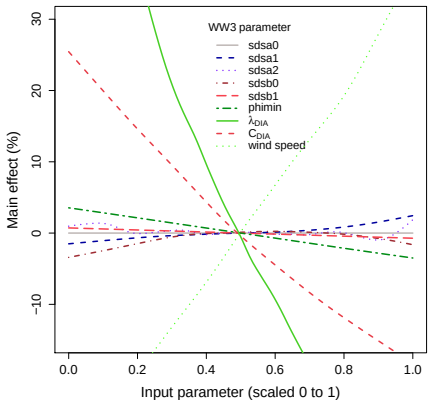
(b) Mean = 1.9m; S.D. = 0.23m



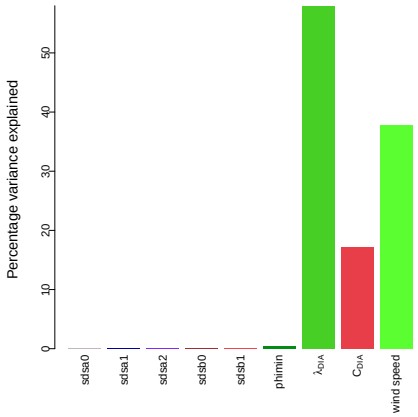
(c) Main effects at 207km



(d) Mean = 3.0m; S.D. = 0.39m



(e) Main effects at 917km



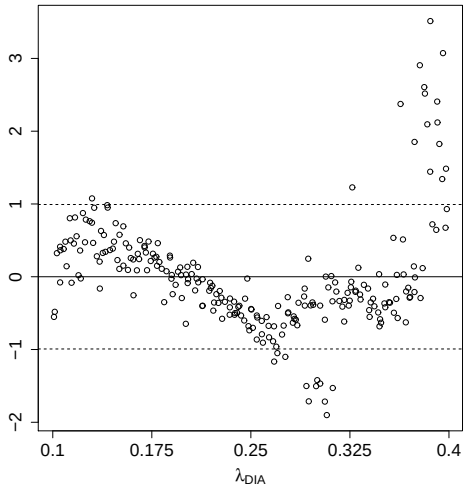
(f) Mean = 4.2m; S.D. = 0.61m

Figure 5.14: Sensitivity analysis for H_s at increasing distance along the fetch.

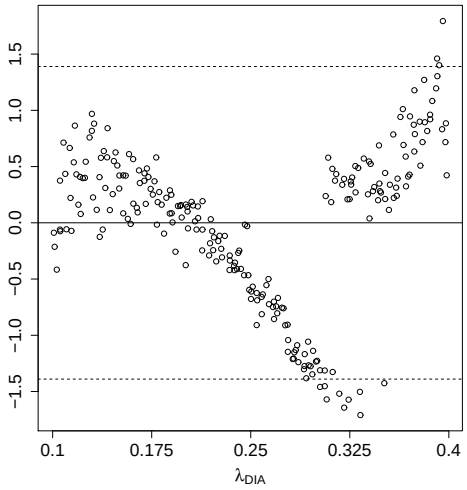
the contribution to the total variance by dissipation diminishes almost entirely.

The results here seem to support the historic beliefs that uncertainty about winds dominated simulation error. Given the input uncertainty is characterised by a s.d. of 0.5ms^{-1}, it has a remarkably strong individual effect, contributing at least 28% to the output variance in each case. However, the input parameters associated with the DIA contribute in excess of 63% of the variance at 200km fetch and beyond. In terms of the wave simulator operation, this is evidence perhaps that the importance of the DIA, and the S_{nl} source term in general, is to a degree overlooked, especially considering the focus in recent years on the development of input and dissipation schemes.

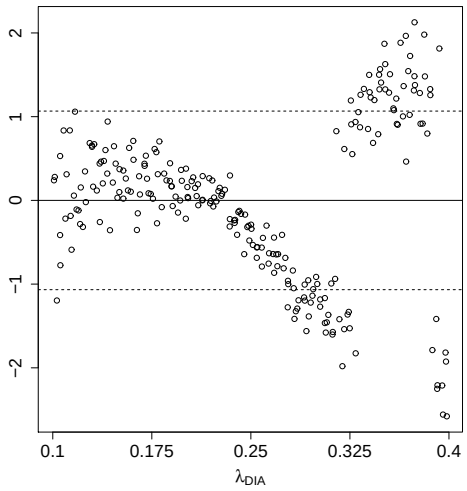
We now move on to discussion of the sensitivity analysis for T_p. Before considering the results, findings of linear regression are described because highly nonlinear response to λ_{DIA} was identified. Other parameters, apart from wind speed, were seen to be largely inactive by comparison. Plots of the residuals from a first order regression model against λ_{DIA} are shown in figure 5.15. The nonlinearity arises from some kind of divergent behavior, as the value of λ_{DIA} is increased beyond approximately 0.31. Although nonlinear response is evident at the shorter fetch of 53km it does not manifest in quite the same way. This behaviour is mathematically interesting but, physically, almost certainly spurious. The most obvious explanation is that λ_{DIA} was never intended to be adjusted much beyond its default value of 0.25. That is, in this particular example, the approximate solution yielded by the DIA scheme with values of λ_{DIA} beyond approximately 0.31, together with some other combinations of input parameter values, results in highly unphysical behaviour. This is problematic in terms of formulating an emulator because an assumption of the statistical model is continuity of the data. This could be avoided by redesigning the experiment to exclude the specific region of input space (where $0.31 < \lambda_{DIA} < 0.4$). Given the input uncertainty specification $\lambda_{DIA} \sim N(0.25, 0.0625^2)$, we see that the problematic region lies beyond 1 s.d. and would therefore make only a limited contribution to any analysis. Formal investigation of that specific region would require the use of a statistical model appropriate to discontinuous data. The output data for 52km fetch appears to be continuous, if somewhat divergent at the high end, so we may be able to obtain some analysis. Making use of a linear function, $T_p \sim 1 + x_{[3,4,7,9]} + x_7^2 + x_7^3 + x_7^8 + x_7^9 + x_7^{10}$ and a nugget term an emulator yields the graphical diagnostics in figure 5.16 (a). Notice that the output data set is narrowly distributed with a mode near the lower end of the scale, between approximately 4 and 7 seconds. The higher (or more extreme) values are sparse and are likely related to high input values of λ_{DIA}. Similar points are also observed in figure 5.15 (a), at



(a) 52km

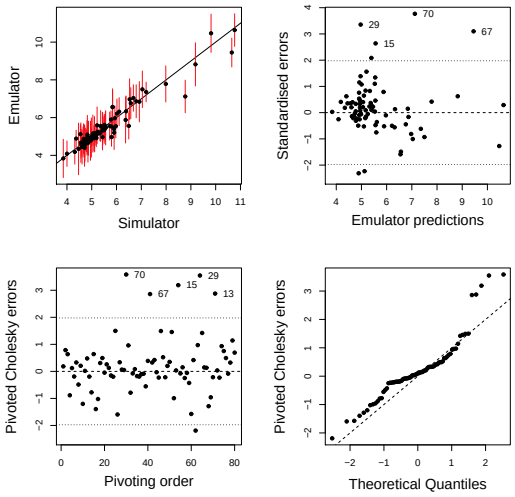


(b) 207km

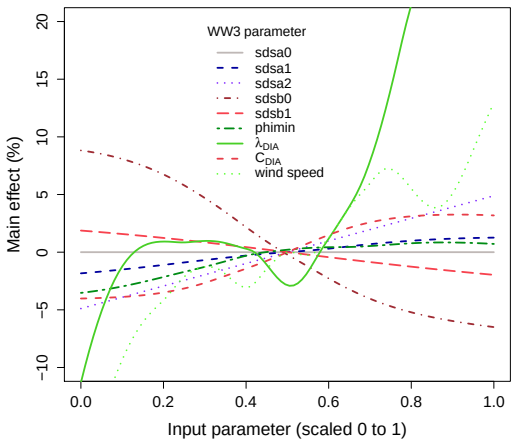


(c) 917km

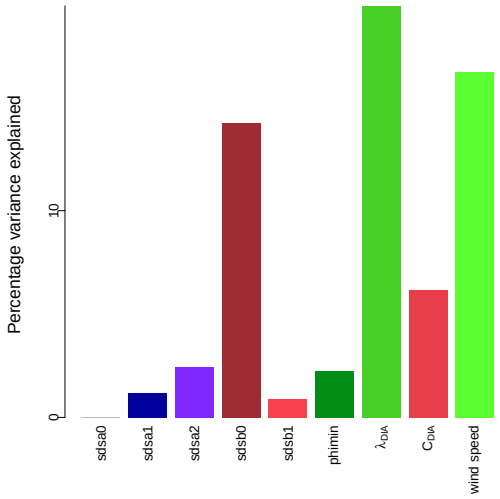
Figure 5.15: Residuals from a linear (1st order) regression model for T_p against λ_{DIA} for different values of fetch. Note that actual values of λ_{DIA} are shown.



(a) Diagnostics



(b) Main effects at 52km



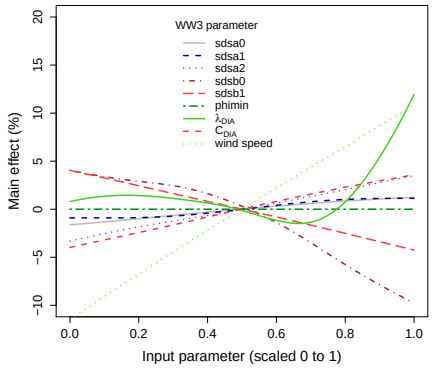
(c) Mean = 5.1s; S.D. = 0.38s

Figure 5.16: Graphical diagnostics and sensitivity analysis for T_p at 52km fetch.

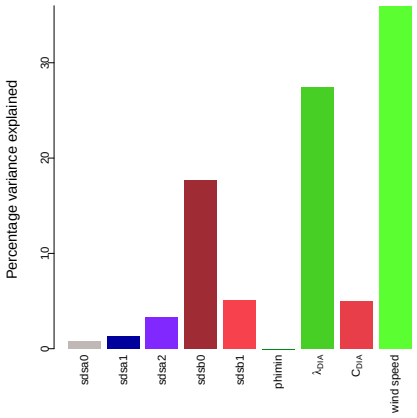
the higher end of the range. With a number of outliers and a fairly irregular error distribution, the emulator does not validate terribly well, which is likely due to the heteroscedasticity of the training data. Results of a sensitivity analysis are shown in figures 5.16 (b) and (c) and reflect the strong influence of λ_{DIA} and wind speed. However we should be cautious in interpreting them due to the poor fit of the emulator. In particular notice the response of the wind speed. It no longer shows a linear relationship that we expect but induces more of a cyclical response around a linear trend, which is symptomatic of the poor emulator performance[48]. Nonetheless, at least qualitatively we can see from figure 5.16 (c) that the input dissipation parameters actually have an appreciable influence at short fetch, very much like the case for H_s. Given the questionable emulator performance, the experiment is redesigned with $0.15 < \lambda_{DIA} < 0.30$ which removes the divergent behaviour arising from higher (and lower) values. Emulators are found to validate far more robustly, although some discontinuity is still evident for the response at short fetch for higher values of λ_{DIA}. A nugget term (=0.5) was included in the emulator in order to compensate for the fact that the model assumptions (i.e. constant variance) appear too strong for the data. Results of the uncertainty and sensitivity analysis are shown in figure 5.17. Total variances remain quite small, rising to approximately s.d. = 0.4s at the longest fetch. Noting that the joint input uncertainty distribution is now different between the experiments for H_s and T_p, we see a much lower variance (as a proportion of the mean) at all fetches. With respect to the results at 52km fetch, notice first that they, at least qualitatively, have not changed substantially. λ_{DIA} and wind speed still dominate the response but a number of dissipation parameters also contribute significantly to the total variance. Notice also that we have recovered the strong linear dependence of T_p on the wind speed. The pattern of influence is comparable to that for the H_s output data with two notable differences. Firstly, although again λ_{DIA} is seen to induce a strong nonlinear response, the form of the response is different for H_s and T_p. Secondly, the response to C_{DIA} is weaker for T_p. Figures 5.17 (d) and (f) reveal how at longer fetch λ_{DIA} and wind speed are the dominant sources of variance. This is consistent with analysis results of the H_s data and no doubt due to the correlation between the two quantities. Note also that the main effect variances account for 96.5%, 99% and 99.5% of the total variance for the output at the three fetches respectively, revealing that there is little parameter interaction.

Before moving on to the final experiment in this chapter, let us briefly summarise the results. Regarding the total variance induced in the output measures, H_s is seen

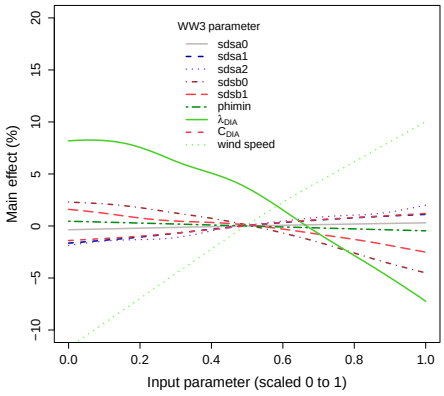
[48]Think of this a bit like the GP mean function "oscillating" between design points.



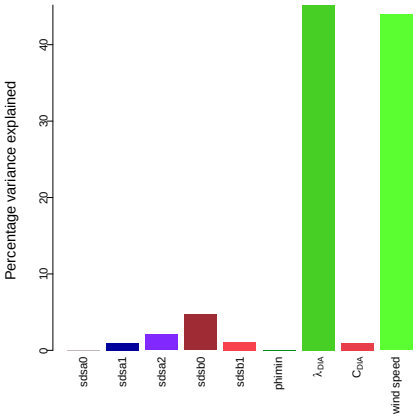
(a) Main effects at 52km



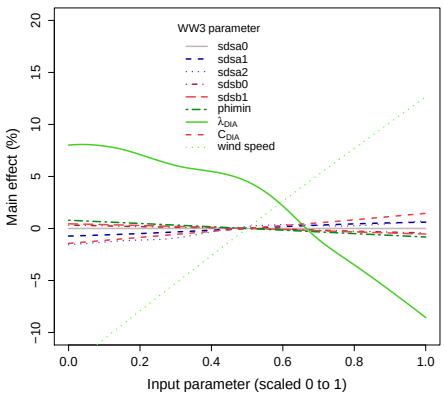
(b) Mean = 5.1s; S.D. = 0.23s



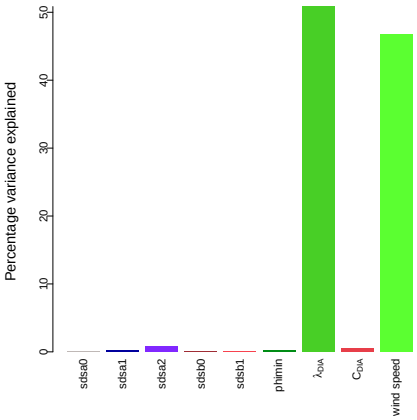
(c) Main effects at 207km



(d) Mean = 6.7s; S.D. = 0.28s



(e) Main effects at 917km



(f) Mean = 9.0s; S.D. = 0.43s

Figure 5.17: Sensitivity analysis for T_p at increasing distance along the fetch.

to be most sensitive to input uncertainty, with a distribution characterised by a s.d. of 0.61m at the longest fetch. This is substantial given a mean of 4.2m. An important finding in regard to the overall objectives in this thesis is the determination that wind input is highly influential. In all situations, that is at short and long fetch, for output H_s and T_p, wind speed was found to contribute at least 27% to output variance. In developed sea conditions this was seen to be as high as 47%. This finding is particularly interesting noting that the input uncertainty for the wind speed, $N(12.5\text{ms}^{-1}, 0.5)$, is arguably conservative. However, it is clear that other input parameters do contribute significantly given their uncertainty specifications. Specifically, the nonlinear wave-wave interactions parameterisation is at least as influential. In all cases, other than for T_p at short fetch (where it was 32%), the combined contribution to output variance was at least 40% and sometimes exceeded 60%. These findings highlight the importance of nonlinear interactions, at least in terms of how the DIA scheme is implemented in WW3. Relatively small (percentage) adjustments in the tuning parameters can induce nonlinear response and influence simulation output substantially. Finally, an important point is that at short fetch for developing seas, energy dissipation *is* clearly more active. There is a big contrast between its influence at short and long fetch, where it is almost entirely absent. With these findings in mind, we now move on to the final experiment where the wind configuration is modified.

5.7 Experiment 5.C: Linear wind decay

5.7.1 Experiment configuration and design

Possibly the simplest extension to the previous experiment is that of a wind speed that decays along the fetch (though remains constant in time). This remains a fairly contrived arrangement but could be envisaged in reality as resulting from some kind of spatial dissipation of the wind. Drop in wind speed is perhaps more easily imagined as a transitory or dynamic effect but over a larger scale it could endure for long enough to result in such a scenario as being investigated here. The objective is to learn about how this variation of fetch-limited wave growth compares to the previous example, and whether it gives rise to any significant difference in response to uncertainty. More specifically, do the same processes act in the same way and does uncertainty about tuning and wind speed have the same effect on those processes.

The experimental configuration here is modified only by the change in forcing wind profile. The profile has an initial wind speed, which is regarded as uncertain

in the same way as the previous experiment, characterised by $N(12.5\text{ms}^{-1}, 0.5^2)$. The wind speed decays to zero at the end of the 1000km fetch. The resulting growth profile obviously differs from the previous case in that steady fully developed conditions are not reached. A situation is expected in which the waves grow along the fetch until the wind speed falls to a point where it can no longer support wave growth. This location is governed in some way by wind speed, energy dissipation and nonlinear interactions. After that point, waves will diminish as the wind speed drops to zero.

The uncertain parameters and experiment design remain the same (as set out in table 5.4 with the adjustment of λ_{DIA}), comprising three maximin LHS designs. Before proceeding to perform the analysis let us examine the evolution of the sea state along the fetch. An arbitrary sample of nine simulations from the first design set are shown in figure 5.18. The output data in this experiment is plotted on a common scale with simulations (generated from exactly the same input) from experiment 5.B, in order to draw a comparison. As anticipated, the growth of H_s gradually departs from the fetch-limited case seen previously due to the decaying wind speed. Variation between samples can be seen, particularly in the overall magnitude of wave height and period. Looking more closely it can be seen that the "transition" region, from growth to decay, that lies at approximately 300km, exhibits variation in terms of its rate of change at the onset of decay. In slight contrast, the evolution of T_p exhibits a steady monotonic increase in all cases, although the slower rate of increase is consistent with the decaying wind. The continued growth along the fetch in spite of the diminishing wind speed is indicative of the continuous process of energy transfer, via wave-wave nonlinear interactions, to lower and lower frequencies (and higher periods). With respect to T_p only, runs #36 and #57 exhibit discontinuity which appears somewhat unphysical. Run #57 in particular contains two erratic step increases. A closer look at the underlying spectra reveals that in these cases it is bimodal, and the step change is attributable to a change in dominant mode at that point. This highlights that T_p must be interpreted with care. Bimodality seems unlikely in this "smooth" and idealised case, and is likely attributable to some region of the design space that results in less realistic output.

In order to examine the transition region at around 300km and investigate waves at long fetch, the conditions at points highlighted by red squares and triangles are chosen for analysis. Emulators are trained in the usual way and the results are discussed in the next section.

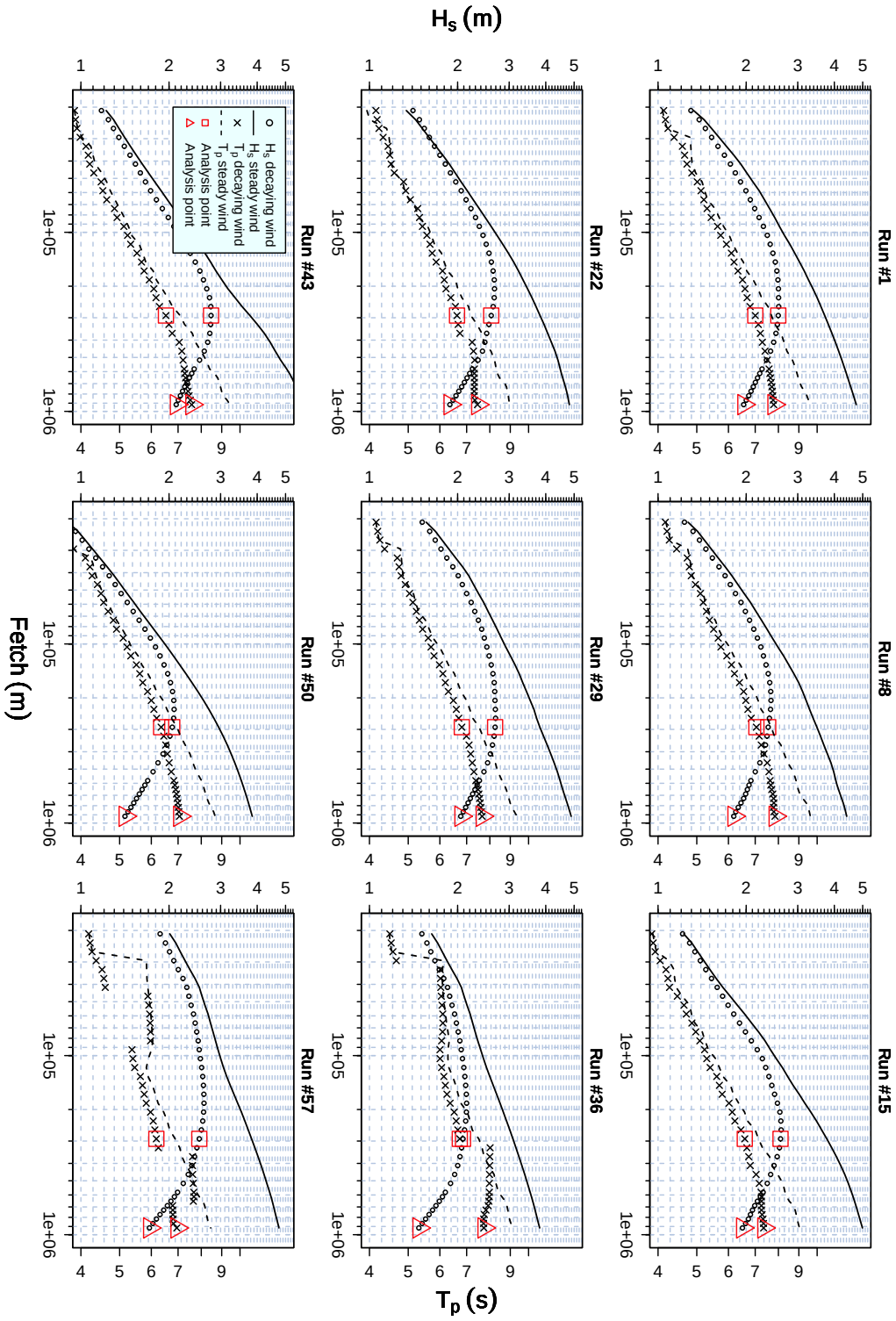


Figure 5.18: Wave growth and decay profiles for different simulations taken from the first design set (experiment 5.C). These are compared with wave growth profiles from experiment 5.B, shown by the solid and dashed lines. Points along the fetch for which an uncertainty analysis was carried out are designated by red squares and triangles.

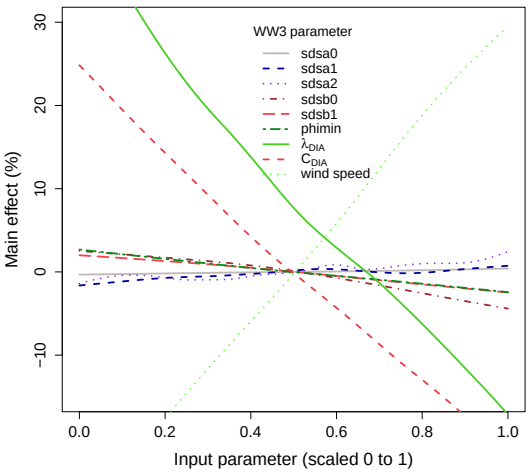
5.7.2 Uncertainty and sensitivity analysis for H_s and T_p

Emulators were formulated and sensitivity analysis for H_s and T_p at fetches of 292km and 917km are carried out (marked by the red squares and triangles respectively, in figure 5.18). Mean functions of the form $H_s \sim 1 + x_{[2,3,4,6,7,8,9]} + x_7^2 + x_7^3$ and $H_s \sim 1 + x_{[2,3,4,6,7,8,9]} + x_7^3$ were used for fetch 292km and 917km respectively. Nonlinear response to λ_{DIA} (denoted by x_7) is again observed. Results for the analysis of H_s are shown in figure 5.19. The results bear resemblance to those for the case of fetch-limited growth at long fetch shown in figure 5.14. Evidently energy dissipation is not very active at both of these locations, in terms of either low or high frequencies. However, as seen previously nonlinear interactions are clearly influential, and wind speed uncertainty is responsible for the largest proportion of variance. There is in fact little difference between the two locations. Given the lack of influence of energy dissipation at longer fetch seen here, we can regard this as further evidence that effects of energy dissipation are more confined to the evolution of waves in young sea states.

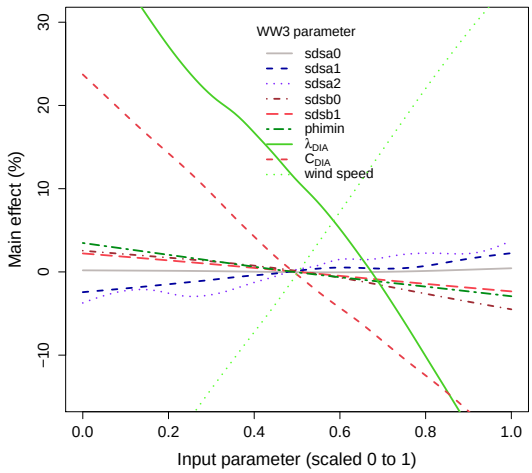
Emulators were also formulated and trained using output for T_p. Linear regression identified that terms up to order 8 in λ_{DIA} explained the trend in output. Mean functions of the form $T_p \sim 1 + x_{[2,3,4,5,7,8,9]} + x_7^2 + x_7^3 + x_7^5 + x_7^6$ and $T_p \sim 1 + x_{[2,3,4,5,7,8,9]} + x_7^2 + x_7^3 + x_7^4 + x_7^6 + x_7^7 + x_7^8$ were used for fetch 292km and 917km respectively. Also note that a large nugget term was required to compensate for heteroscedasticity induced by λ_{DIA}. Results of sensitivity analysis for T_p are shown in figure 5.20. The dominance of the wind speed and lack of sensitivity to other input is consistent with previous findings. As with the results for H_s, we can see that as the sea state develops the influence of the nonlinear interactions slowly diminishes. Note also the highly linear dependence of T_p on the wind speed and the nonlinear dependence on λ_{DIA}.

In general, these results are very similar to the fetch-limited case. Wind speed has been found to be the dominant source of uncertainty in every case although the tuning of the DIA scheme, in particular the parameter λ_{DIA}, is also highly influential. Other input parameters associated with the tuning of both low and high frequency wave energy dissipation of the TC96 parameteristion are largely inactive.

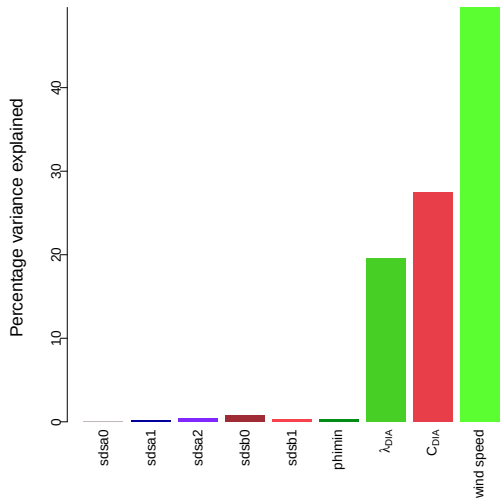
In the next part of this experiment the output is parameterised in a similar way to that used in experiment 5.A. This allows for the analysis of growth and decay *rates*, rather than wave summary statistics at specific points.



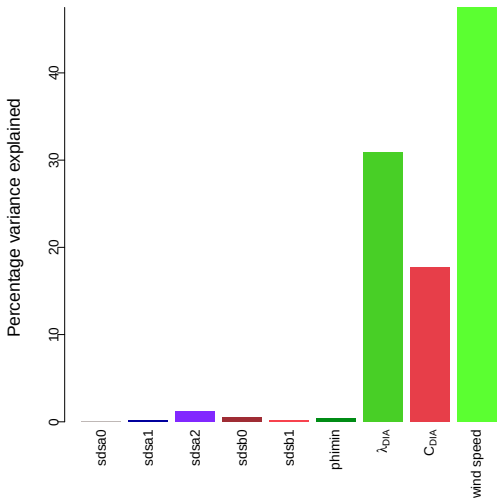
(a) Main effects at 292km



(b) Main effects at 917km

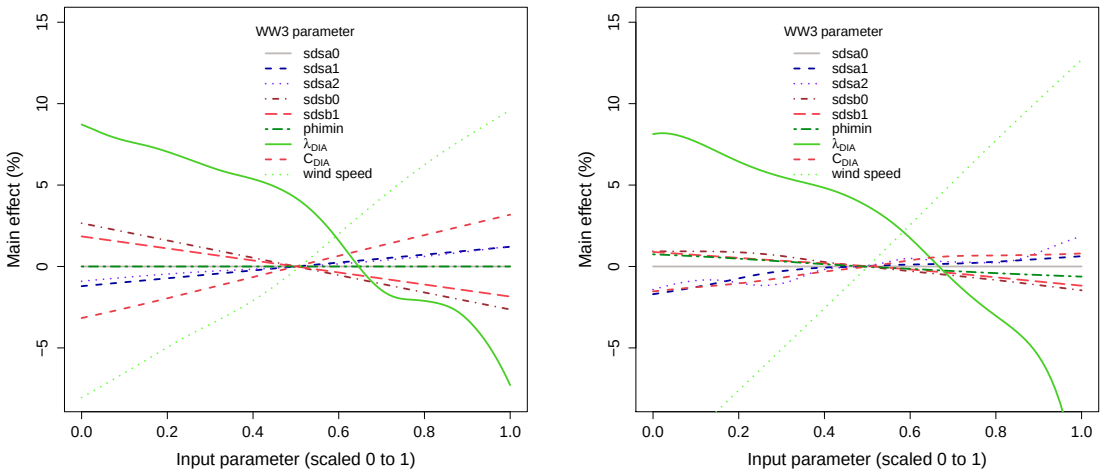


(c) Mean = 2.3m; S.D. = 0.24m



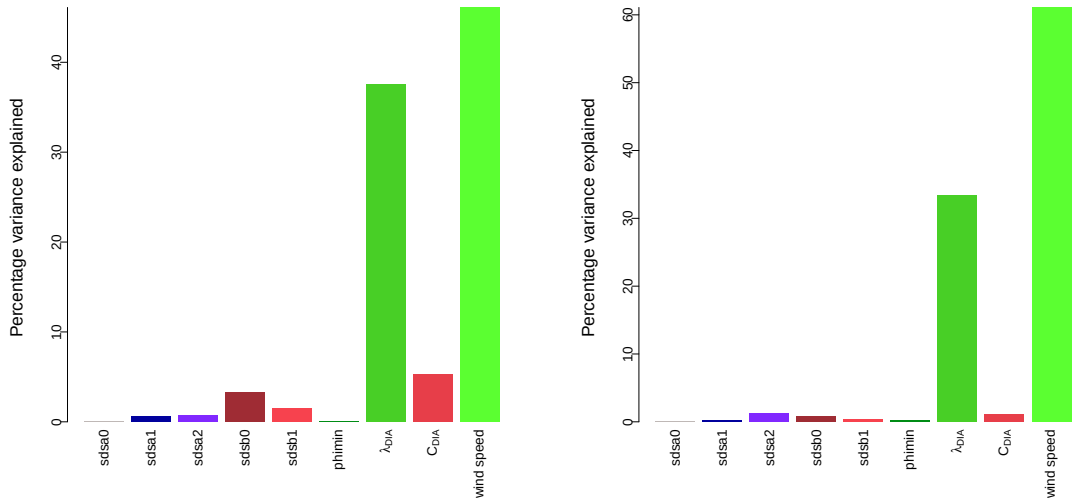
(d) Mean = 1.7m; S.D. = 0.22m

Figure 5.19: Sensitivity analysis for H_s at specific points along the fetch.



(a) Main effects at 292km

(b) Main effects at 917km



(c) Mean = 6.6s; S.D. = 0.25s

(d) Mean = 7.4s; S.D. = 0.29s

Figure 5.20: Sensitivity analysis for T_p at specific points along the fetch.

5.7.3 Uncertainty and sensitivity analysis for growth rates

Considering figure 5.18 we can see variation in the growth and decay process of wave height. Between 200km and 400km the wind drops below a strength sufficient to maintain the growth of wave height, and beyond this point the wave height begins to decay. Further inspection of figure 5.18 suggests that we could define three regions along the fetch. Firstly, there is a region of growth up to approximately 200km. This is comparable to the fetch-limited growth region seen in experiments 5.A and 5.B although here, since the wind speed is continuously decaying along the fetch, the growth rate also decays with fetch. Secondly, between approximately 200km and 400km there is an initial region of decay, where the wave height begins to diminish. This region exhibits a variation in terms of how rapidly the decay occurs. For example runs #43 and #50 seem to show a fairly rapid decay, while runs #29 and #57 show a much broader profile. Thirdly, at fetches beyond 400km a region of fairly steady decay is observed.

In order to investigate the growth, and initial decay regions, the output data is represented parametrically. Second order linear regression models, of the form $H_s \sim a_0 + a_1 F + a_2 F^2$, are fitted to output data for H_s. Note that the models are fitted to the actual data, not its *log* as typically shown in figure 5.21 for example. The use of such models therefore reduces the output data to three coefficients, a_0, a_1 and a_2 corresponding to the three terms. Physically, the constant term in the model essentially corresponds to absolute H_s (this is analogous to $\tilde{H}_\infty$ in experiment 5.A). The first order term (in fetch) corresponds to the rate of change of H_s with fetch, which is the transferral of wind energy to wave energy with fetch. Higher order terms correspond to higher order effects and so in this case the 2nd order term captures the rate of change of net energy transfer. Second order models were found to fit the growth region from approximately 35km to 95km, and the transition region from approximately 200km to 400km. Examples of such models are shown in figure 5.21.

Analysis of the initial growth region is discussed first. The linear model captures the wave growth well, as indicated by the modal average of the multiple R-squared values for the regression fitting for the three design sets, of 0.9978. The three coefficients, a_0, a_1 and a_2 provide information about wave growth, as mentioned above. Analysis therefore proceeds by training emulators to each of the coefficients and running a sensitivity analysis. A few points are mentioned regarding the construction of the emulators. Prior mean functions were determined through the use of simple regression methods as discussed previously. For coefficient a_0 a function of the form $a_0 \sim 1 + x_{[2,3,4,5,7,8,9]} + x_7^3 + x_7^5$. Here we again see higher order terms

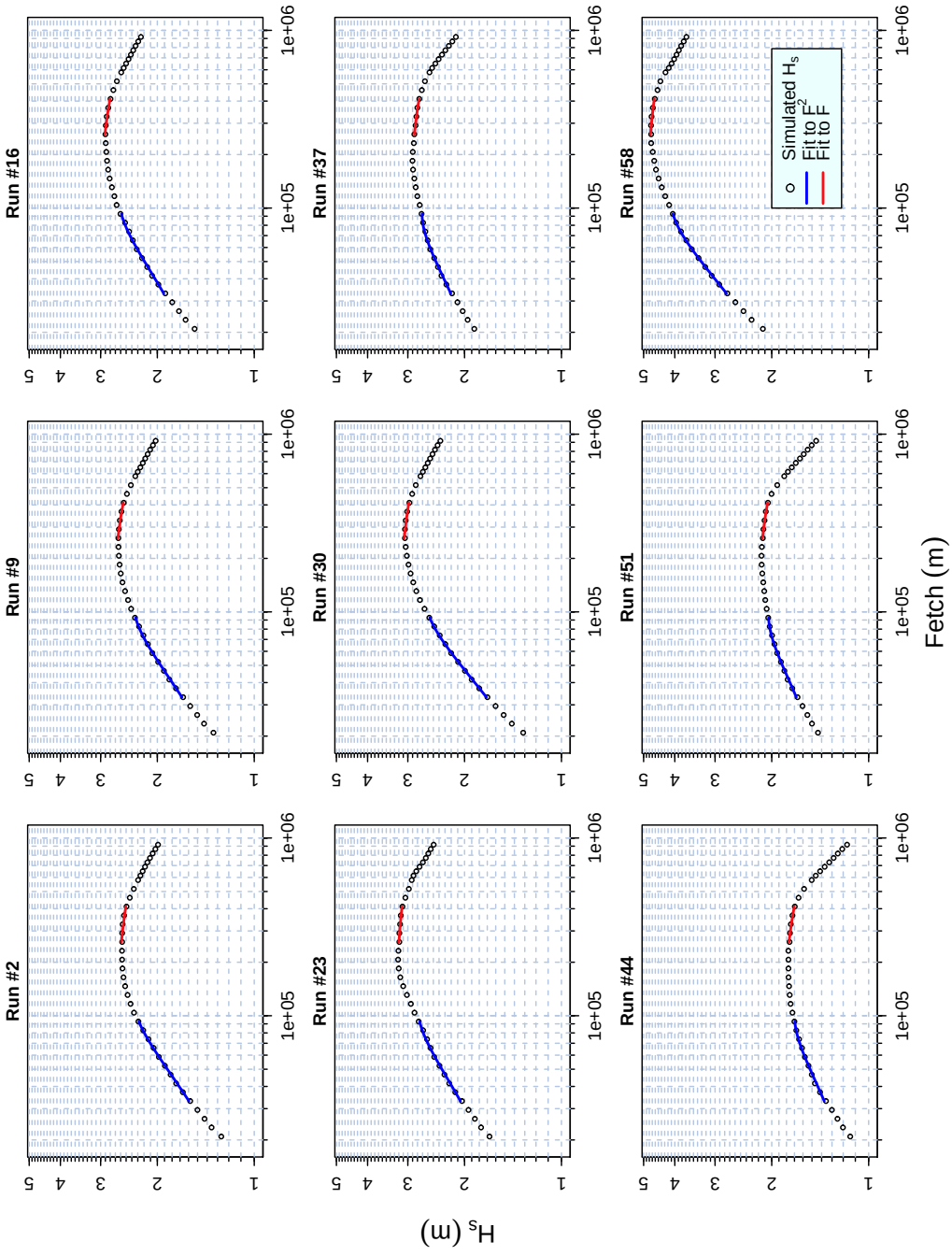
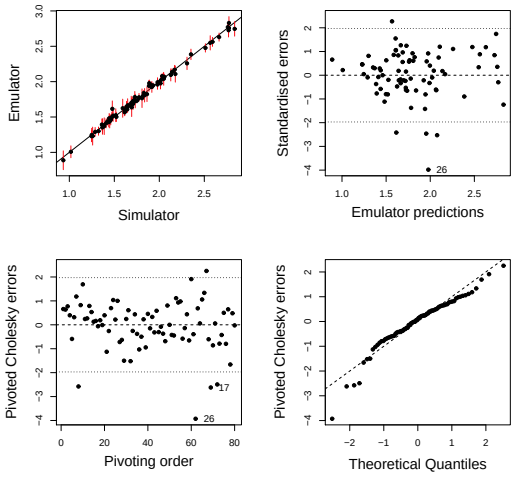


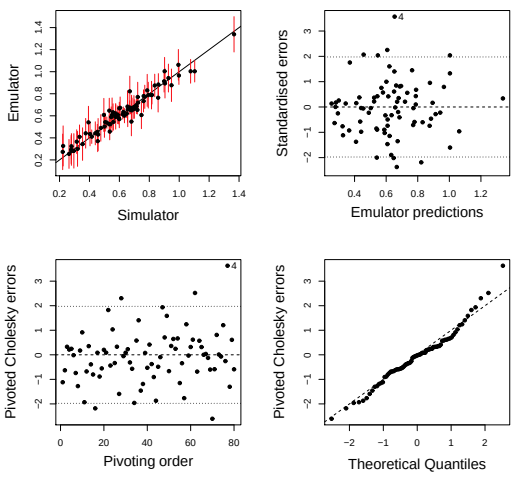
Figure 5.21: Linear regression models fitted to H_s training data. The first (blue) line in the growth region is of the form $H_s \sim a_0 + a_1 F + a_2 F^2$. The second (red) line in the "transition" region is of the form $H_s \sim b_0 + b_1 F + b_2 F^2$.

associated with the response to λ_{DIA}. a_1 has dependence only on λ_{DIA}, C_{DIA} and wind speed with the form, $a_1 \sim 1 + x_{[7,8,9]} + x_7^2 + x_7^3$. a_2 has linear dependence on all input except *sdsa0* and *phimin* (which are inactive). Matérn ($\nu = 3/2$) covariance functions are employed in all cases. Emulator graphical diagnostics are shown in figure 5.22. Emulator performance is good, but notice that the emulator posterior variance appears to increase for predictions of each of the three coefficients. This reflects the fact that the terms in the linear model that is used to represent the simulation output explain less and less of the variance of the output, with increasing order. That is, the higher the order of the term, the more "noisy" it is, and the weaker the relationship to the output data. This seems to suggest that there is weaker higher order dependence on the uncertain input. The emulator for the highest order term, a_2, therefore has the highest prediction uncertainty. The emulator for the coefficient of the lowest order term, a_0, (which is closely related to maximum H_s) makes very good point predictions with low variance, indicating the strong relationship between the input parameters and a_0. Results from the sensitivity analysis are shown in figure 5.23. Again a strong linear response to wind speed is shown in all cases, consistent with previous findings. The influence of the input on coefficient a_0 is similar to that seen for H_s at short fetch in figure 5.14 (b) in the previous experiment. In fact, the results of the uncertainty analysis are comparable with the mean and s.d. of 1.9m and 0.23m seen in that case. A similar comparison can be made between the response of a_1 and the response of H_s at longer fetch ($\sim$ 200km). The influence of input, dominated by nonlinear interactions and wind speed, is very similar. This is physically consistent since, in the absence of higher order effects, the growth rate will closely correspond to the maximum wave height, seen at approximately 200km. The influence of input on a_2 is interesting, being distinctly different from anything we have seen previously. Evidently parameters *sdsb0* and *sdsb1*, both associated with low frequency energy dissipation, are influential and induce nonlinear response, suggesting that energy dissipation is important in higher order physical effects (at least in terms of the simulator's representation of them). The mean value of -0.1 signifies the decline in wave growth. In the absence of clear connections between the input and the details of physical properties or processes that might be associated with higher order effects, it is hard to gain more in-depth physical insight. That is a drawback of the empiricism used in the parameterisations under investigation here. Nonetheless the results do suggest energy dissipation is responsible for higher order behaviour.

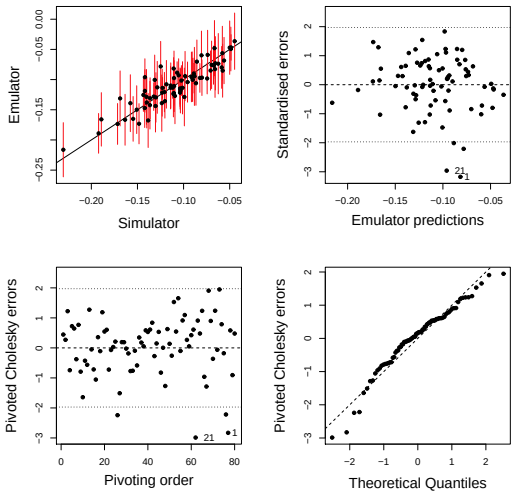
Before concluding this experiment, a similar analysis is performed for the transition region, as indicated by the red lines in figure 5.21. The linear model of the form



(a) Emulator for a_0

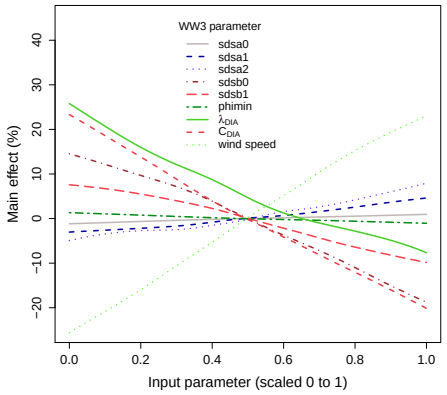


(b) Emulator for a_1

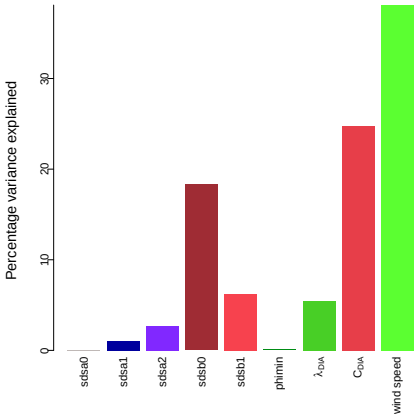


(c) Emulator for a_2

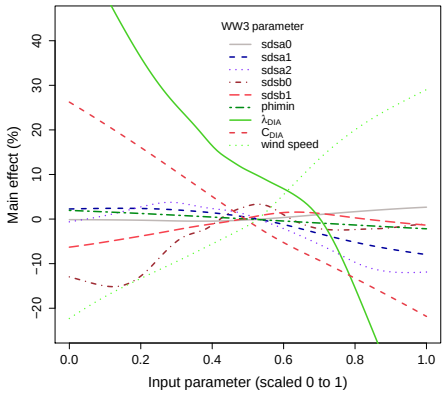
Figure 5.22: Emulator validation diagnostics for regression coefficients a_0, a_1 and a_2.



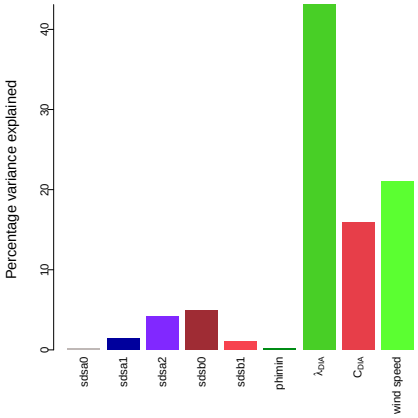
(a) Main effects for a_0



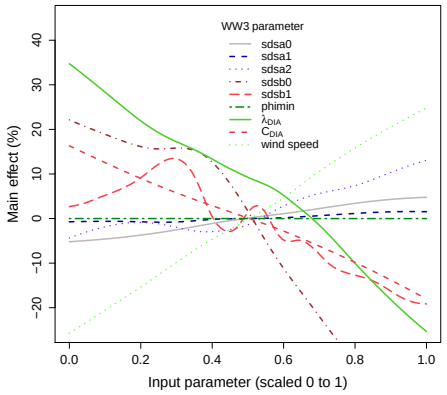
(b) Mean = 1.81; S.D. = 0.19



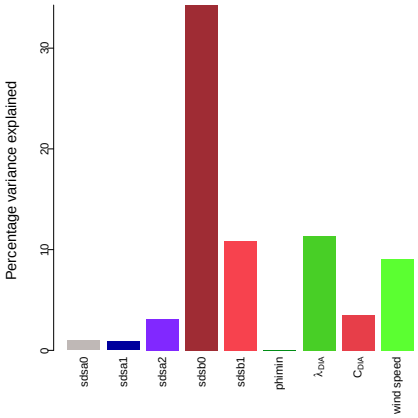
(c) Main effects for a_1



(d) Mean = 0.60; S.D. = 0.096



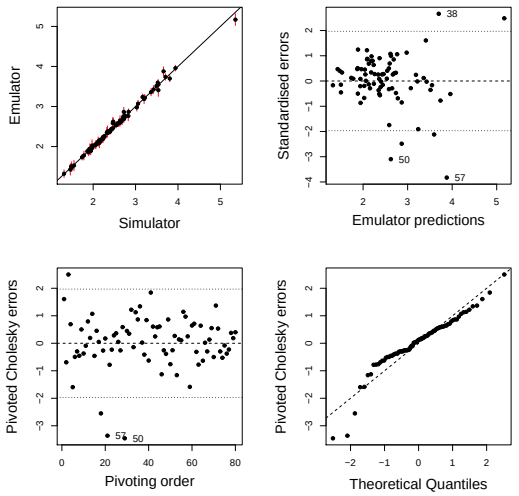
(e) Main effects for a_2



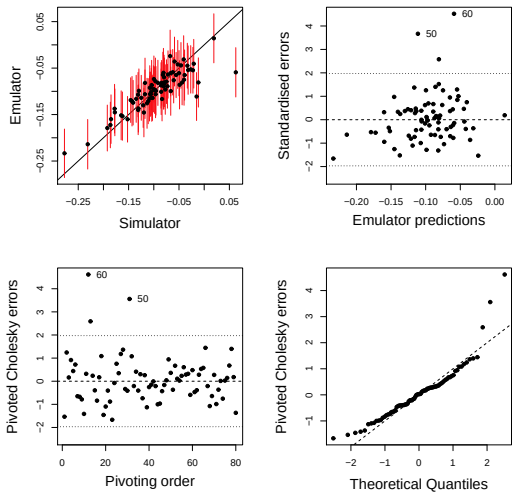
(f) Mean = -0.10; S.D. = 0.022

Figure 5.23: Sensitivity analysis for the coefficients describing the initial growth of wave height.

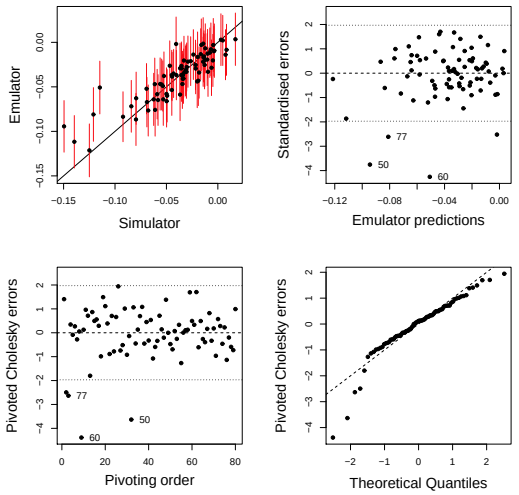
$H_s \sim b_0 + b_1 F + b_2 F^2$ is fitted to the output data. The mode and mean averages of the multiple R-squared for the model are 0.997 and 0.982 respectively suggesting that it captures the variability well. In the same way as before, three emulators are formulated, one for each of the coefficients b_0, b_1 and b_2. As in previous cases we expect to see dependence mainly on wind speed uncertainty and the DIA scheme, given that the sea state is not growing in this region. Diagnostics for the emulators are shown in figure 5.24. A small number of outlying points can be seen suggesting that the modelling assumptions may be too strong in all cases, although the emulator performance looks reasonable. In some cases the linear model was probably more poorly fitted to the simulation output data and therefore we might expect a few outlying points. The diagnostics show a pattern similar to that seen previously, where the emulator fits the coefficient for the lowest order term, b_0 in this case, but exhibits high posterior variance when predicting coefficients for the higher order terms, b_1 and b_2. Again, it is likely therefore that the poorer emulator fitting is due to the additional uncertainty introduced by the regression fitting process. As a result large nugget terms (0.44, 0.33) were used in these cases. In fact it may be reasonable to infer directly from the diagnostics that the output data are explained almost entirely by coefficient a_0. Subsequent sensitivity analysis reveals the results shown in figure 5.25. The response of coefficient b_0 is similar to a_0 seen in figure 5.23 (a) and very similar to the results for the point analyses shown in figure 5.19. We see a very strong influence of *sdsa2* in the response of b_1, which also bears some comparison with the response of coefficient a_2, although the specific pattern is again something new. The proportion of the variance explained by the main effects appears to be dominated by energy dissipation. The response of b_2 looks more familiar, and is in fact comparable to the response of coefficient a_1. Note also that the influence of wind speed appears to be largely absent in the response of both b_1 and b_2. This suggests that while the overall magnitude is strongly dependent upon the wind speed, the way in which the transition from wave growth to decay occurs is not, and influenced predominantly by the simulation physics. This seems physically consistent with the following qualitative description. Since wave height is a measure of total of energy, the maximum wave height, correlated with a_0, is strongly governed by wind speed. However, as the wind speed diminishes and an energy balance is achieved (i.e. the wave height ceases to grow), the influence of the wind rapidly disappears leaving only the simulation physics. The general trend in the decay of wave height appears to be governed strongly by the dissipation physics, which is what we would expect. Figure 5.23 (b) and (f) also suggest that nonlinear interactions (via λ_{DIA}) have a strong influence on the rate of decay.



(a) Emulator for b_0

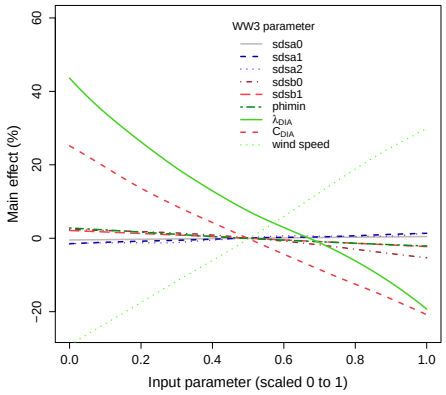


(b) Emulator for b_1

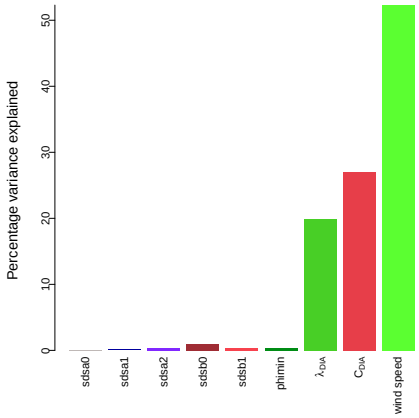


(c) Emulator for b_2

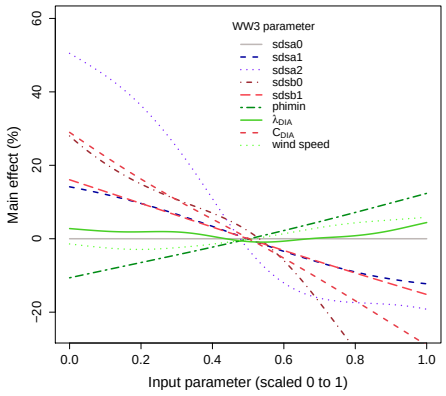
Figure 5.24: Emulator validation diagnostics for regression coefficients b_0, b_1 and b_2.



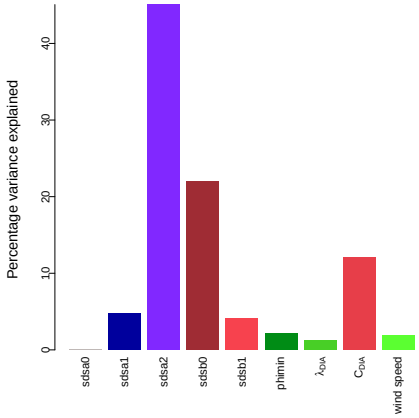
(a) Main effects on b_0



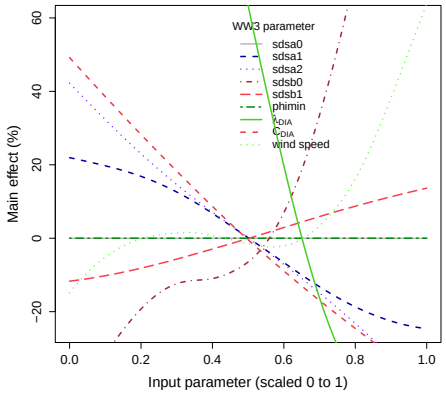
(b) Mean = 2.3; S.D. = 0.24



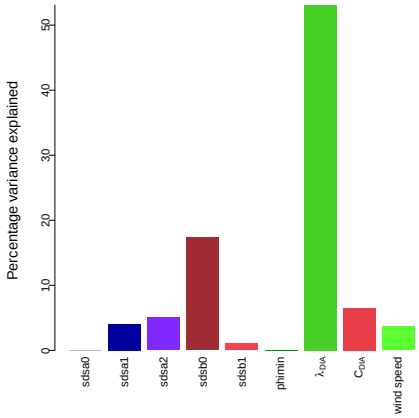
(c) Main effects on b_1



(d) Mean = -0.093; S.D. = 0.019



(e) Main effects on b_2



(f) Mean = -0.018; S.D. = 0.0086

Figure 5.25: Sensitivity analysis for the coefficients of the linear regression fit to the transition region for H_s.

5.7.4 Experiment summary

In this experiment we have learned more about fetch-limited "type" wave growth by examining a situation where wind speed is seen to decay with fetch. Uncertainty and sensitivity analysis of H_s and T_p at specific points along the fetch have revealed very similar results, in terms of the influence of wind speed and wave physics, to that for idealised fetch-limited growth. Specifically, at fetch beyond 200km, only wind speed and nonlinear wave-wave interactions have an appreciable effect. For H_s only, the initial wave growth rate and behaviour in the "transition" region were analysed by using a second order linear regression model to characterise the output data. This appeared to be effective and indicated that higher order effects, in terms of wave growth and decay, and their respective rates as of a function of fetch, were very much governed by the physics of the simulator, particularly energy dissipation. In the absence of a clear connection to specific physical processes for the input parameters, or without looking deeply into the underlying physics, it is difficult to make more robust statements about the underlying processes and how well they are represented in the simulator.

5.8 Chapter summary

In this chapter the results of uncertainty analysis for simulation output of H_s and T_p under conditions of fetch-limited growth, and a variant, are presented. In experiment 5.A, a functional representation of simulated fetch-limited growth was used and sensitivity analysis performed with respect to the effect of uncertain tuning parameters. In experiment 5.B, the scope of the analysis was adjusted and extended to include additional tuning parameters for the DIA nonlinear wave-wave interaction scheme and uncertainty about wind speed. Sensitivity analysis was performed for specific points along the fetch. Experiment 5.C considered a variation of fetch-limited growth by introducing a wind structure that decayed with fetch. In general emulators were found to be very effective although situations arose, largely due to the dramatic effect of large perturbations to λ_{DIA}, where output data had to be carefully considered.

Specific points are summarised in the follow sections.

5.8.1 Wind input

Unsurprisingly wind input is found to be highly influential in developed seas both in terms of H_s and T_p. It typically accounts for between 25% and 50% of output variance in such cases. It is found to be less important (although still very sig-

nificant) at short fetch and also less important with respect to *how* the sea state evolves with fetch. Findings of experiment 5.C show explicitly that where the energy transfer from wind to waves approaches zero, the influence of wind speed also diminishes. Recall also that for all experiments uncertainty on the initial wind speed was characterised by $windspeed \sim N(12.5\text{ms}^{-1}, 0.5^2)$. In chapter 2 this was argued to be conservative so its importance relative to other sources of uncertainty generally appears to be high.

5.8.2 Tuning parameters

Tuning parameters associated with energy dissipation are found to be most influential at fetches of up to 100km. This is consistent with the qualitative argument that energy dissipation is a highly influential process in developing seas where whitecapping is prevalent, and there is a strong energy deficit between wind and waves. This suggests that dissipation is also important in dynamic situations, driven by rapid increases in wind speed. Such situations have not yet been explored but they would no doubt be an interesting research focus. See section 5.8.3 for additional comments.

In contrast, the tuning of the DIA scheme appears to be influential beyond 100km, with its importance increasing fairly quickly with fetch. Specifically, both H_s and T_p are sensitive to perturbation of the parameter λ_{DIA}. For the majority of uncertainty analyses in this chapter its input specification was $\lambda_{DIA} \sim N(0.25, 0.015^2)$, after adjustment owing to unphysical results. In experiment 5.B it was found that for values at the extremes of this range some kind of divergence in the output was evident in some cases (see figure 5.15). Although its physical interpretation is less clear it nonetheless offers a powerful means of adjusting the behaviour (and potentially performance) of the simulator. Given that few published studies exist that examine the DIA scheme in detail (e.g. Tolman, 2004a), particularly in an operational sense, the findings here are important.

5.8.3 Static conditions versus dynamic conditions

Noting the comments in the previous section relating to the importance of energy dissipation in *dynamic* situations, we should be mindful that the investigation so far largely ignores dynamic effects. That is, in this chapter steady state situations have been investigated, where an energy balance between wind and waves exists. We have not explored how uncertainty on the tuning parameters or wind speed influence simulations where there is not a balance, i.e. where there is a (strong) energy imbalance, and the sea state is growing in time. The way uncertainty manifests in

such dynamical situations remains unknown. This may very well differ from the results seen so far. For example, the importance of dissipation physics may change depending on the rate of wave growth and so on. In the next chapter we take another step towards investigating dynamic effects by considering more complex dynamic winds, and their effect on wave direction. Also, more dynamic conditions are investigated in a realistic case in chapter 7.

6 Uncertainty analysis for directional waves in an idealised (2-D) ocean

6.1 Introduction

Having gained an appreciation of how uncertainty in input affects the idealised cases of developed conditions (in chapter 4) and fetch-limited growth (in chapter 5), now complexity of simulation is increased to examine a wider range of phenomenon. In this chapter the experiments continue to investigate uncertainty in the same way, but 2-dimensional wave propagation is allowed. This permits more explicit investigation of directional waves, and also allows for the specification of more complex winds. A brief review of studies into waves arising from a change in wind direction, or turning wind, is given in section 2.4.3. Like the cases of duration- and fetch-limited growth, this kind of physical phenomenon is often used as a benchmark for assessing wave model performance (see e.g. SWAMP, 1985; Komen et al., 1994). In terms of assessing the effect of uncertainty on wave direction, it is a natural starting point. In this chapter wave direction is characterised and expressed by the commonly used summary statistics of mean direction and directional spread.

Two experiments are conducted in this chapter. Firstly, experiment 6.A considers how wave direction responds to a wind that turns abruptly through 90 degrees. Like experiments in chapter 4 this is done using only a single grid cell and so is an investigation of the source function only. This then allows for a comparison with the same experiment conducted with advection on a spatial grid. A grid, of 1000km $\times$ 1000km, is therefore introduced in experiment 6.B, which is conducted in two parts. The forcing wind for the experiment is formulated parametrically, allowing for a more detailed investigation of the effect of wind uncertainty. Uncertainty about wind input, and the approach to parameterising the wind is described in section 6.4.3. The rest of the chapter is structured as follows. Sources of uncertainty and general points on the experimental approach are discussed in section 6.2. Experiment 6.A and 6.B parts (a) and (b) are presented in sections 6.3, 6.4 and 6.5 respectively, followed by a short chapter summary in section 6.6.

6.2 Sources of uncertainty and experiment design.

The experiments conducted in this chapter are summarised in table 6.1 and are described in detail in their respective sections. Sources of parameter uncertainty and their sampling distributions, related to energy dissipation and the DIA scheme, are not changed from experiment 5.B (table 5.4) and are listed in table 6.2. The

Table 6.1: Experiments presented in chapter 6.

Experiment	**Description**
6.A	Investigation of wave direction due to a turning wind, in a single grid cell.
6.B (a)	Exp 6.A with advection, configured on a large spatial grid.
6.B (b)	Exp 6.B (a) modified with wind parameterisation.

Table 6.2: WW3 input parameters and sampling distributions for experiments 6.A, 6.B (a) and (b)

Switch	**Description**	**Tuning parameter [Variable Assignment in WW3 manual]**	**Default value**	**Sampling distribution**
ST2	Dissipation (TC96)			
	High frequency dissipation	SDSA0 [a_0]	4.8	$\mathrm{N}(4.8, 0.2^2)$
	High frequency dissipation	SDSA1 [a_1]	1.7×10^{-4}	$log_{10}\mathrm{N}(-3.293, 0.177^2)$
	High frequency dissipation	SDSA2 [a_2]	2.0	$\mathrm{N}(2.0, 0.2^2)$
	Low frequency dissipation	SDSB0 [b_0]	3.0×10^{-4}	$\mathrm{N}(-3.25 \times 10^{-3}, (9.375 \times 10^{-4})^2)$
	Low frequency dissipation	SDSB1 [b_1]	0.47	$\mathrm{N}(0.5250, 0.03125^2)$
	Low frequency dissipation	PHIMIN [$\phi_{\mathbf{min}}$]	0.003	$\mathrm{N}(0.003, (1.0 \times 10^{-4})^2)$
NL1	Nonlinear interactions (DIA)			
		LAMBDA [λ]	0.25	$\mathrm{N}(0.25, 0.03125^2)$
		NLPROP [C]	10^7	$log_{10}\mathrm{N}(7, 0.1193^2)$

specification of wind uncertainty is however now more complex. In experiment 6.A wind speed alone is specified (in the same way as chapter 5) but a parameterised wind is defined in experiment 6.B. This allows for the specification of uncertainty about wind speed, direction and dynamics (in terms of the time scale for change to occur). A detailed explanation of the parameteristion, together with parameter specifications, is provided in section 6.4.3.

Numerical uncertainty is minimised by adopting a high directional resolution for the wave frequency spectrum, of 36 bins. The spatial grid used in experiment 6.B has dimensions of 1000km $\times$ 1000km and a (high) resolution of 20km per cell (as in chapter 5) thus minimising numerical error associated with energy propagation.

6.2.1 Notes on experimental design

For experiment 6.B, noting the increase in the number of dimensions of the parameter space, typically four maximin LHS designs of 80 points are run for the purpose of exploratory analysis, emulator training and validation. 60 points designs were adopted for experiment 6.A. Note that on the high resolution spatial grid, four days of simulation time for 320 design points (four designs) takes approximately 6 hours when run on 64 cores of a powerful HPC.

6.3 Experiment 6.A: Source term integration for a turning wind (0-D)

The purpose of this experiment is to determine the effect of parameter and wind speed uncertainty on the predicted response and evolution of simulated wave direction. The case of a "turning wind" is considered. As discussed in section 2.4.3, turning winds have been observed on the open ocean and the situation is often used a benchmark case for wave simulator testing and calibration. SWAMP (1985), for example, compare simulations of this phenomenon produced by a number of different simulators. In this experiment their example is followed for the purpose of performing uncertainty and sensitivity analysis. Wave summary statistics, derived from the directional frequency spectrum, typically used to describe wave direction are *mean wave direction* and *directional spread*. These quantities are typically produced as default output from a wave simulation.

In formulating the experiment, SWAMP (1985) advocated making a sudden change in wind direction at the point in time where $f_p = 2f_{PM}$, where f_{PM} is the peak frequency of a fully developed sea state as identified by Pierson and Moskowitz (1964). That is, the sea state is in some sense "halfway" developed. This approach

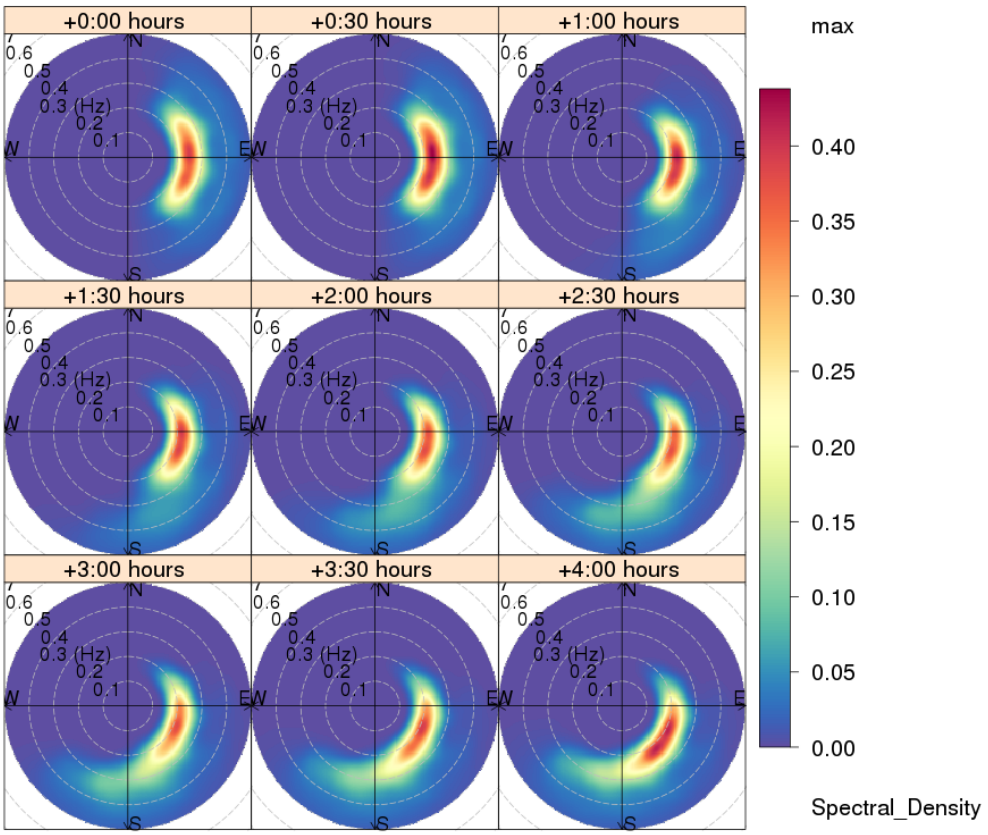


Figure 6.1: Time evolution of the simulated 2-D spectrum generated from a turning wind. Times indicated are relative to the shift in wind direction, not the start of the simulation. Note that the colour corresponding to the spectral intensity has been smoothed for visualisation purposes and may not capture some of the fine detail.

has also been taken here. This experiment utilises the same single cell configuration that was considered in section 4.6 where duration limited conditions were investigated. The configuration uses a forcing wind of 10ms^{-1} in a direction of 270° which was turned abruptly by +90°. The condition $f_p = 2f_{PM} = 0.255$ was found to be satisfied at a time of +04:30 hours from evolution from calm conditions. For sea states exhibiting multi-modal or skewed directional frequency distributions the mean direction may differ from, say, the direction of the peak frequency (assuming a distinct peak exists), so caution is required. It is clearly important to inspect the entire directional spectrum for irregularity. The time evolution of the simulated 2-D spectrum for the (design) mean input is shown in figure 6.1. The development of wind-sea in the new (wind) direction becomes distinctly evident from approximately +1:30 hours onwards. As the sea state evolves the new wind-sea appears to merge with the existing spectrum and the direction of the spectral peak gradually aligns with the new wind direction. Notably the spectral "shape" appears to remain quite

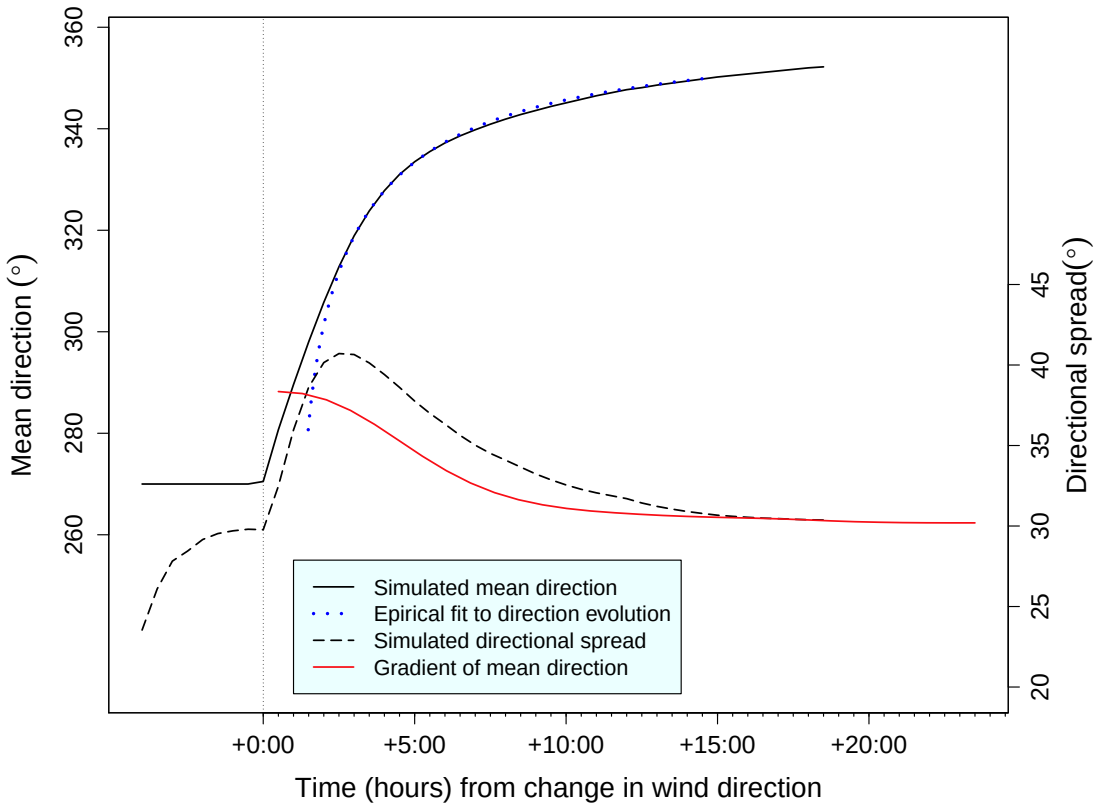


Figure 6.2: Time evolution of mean wave direction when a sudden change in wind direction of +90° occurs, where $f_p = 2f_{PM}$.

consistent throughout the transition. That is, the sudden change in direction does not give rise to the formation of a well defined second spectral peak, but rather there is a steady transition, and the initial spectrum at time +00:00, in a sense, is simply rotated about the origin - it retains its shape in the spectral space. This is in part due to the modest change in direction and wind speed (10ms^{-1}), and with a larger directional shift and higher wind speed we might expect a second spectral peak to form as part of the transition to the new sea state. The 2-D spectrum is now reduced to mean direction and directional spread, the time evolution of which, for the mean run, are shown in figure 6.2. The solid black line shows the time evolution of the mean direction. The change in wind direction at time +00:00 is clearly evident as the wave direction immediately, and rapidly, begins to adjust. The transition is smooth with a fairly linear initial response that has diminished considerably by the time a direction of 340° is reached. The corresponding evolution of directional spread is given by the dashed black line. Notably, we can see that the spread increases from the very beginning of the simulation and then, commensurate with the change in wind direction, increases again more rapidly. Interestingly, after approx-

imately t=+03:00 the gradient (red solid line) of the mean wave direction seems to track the change in directional spread quite closely, although lags it by about 3 hours. The evolution of directional spread is also quite easy to see visually from figure 6.1, with the largest spread happening between t=+2:00 and t=+4:00 hours. A broadening of the directional spectrum is expected given that a new wind-sea in the new direction is being generated concurrent with an existing wind-sea.

The final information provided in figure 6.2 is the blue dotted line showing a fitted directional relaxation model as proposed by Holthuijsen et al. (1987). Recall equation (2.15), which describes how wave direction responds to a changing wind.

$$\frac{\partial \theta_0}{\partial t} = \tilde{\tau} sin(\theta_w - \theta_0)$$

This model was fitted to the simulated data, with the resulting curve shown by the blue line, where $\tilde{\tau} = 0.5$ was found by a nonlinear least squares algorithm. This value is lower than those generally estimated from direct observation by Holthuijsen et al. (1987) although the fit is quite poor immediately after the directional change. The fit is much better after t=+2:00. Compared with the model, the simulator does not appear to give a sufficiently rapid response. However, the relaxation model itself is based upon a number of approximations and, as shown in Holthuijsen et al. (1987), does not give consistent results when compared to observations, observational error being high.

The analysis here therefore investigates how variability within the wave simulator effects predictions of this type of phenomenon. A designed experiment was run using the same design as experiment 4.B. Output from a sample of five runs from the first design set are shown in figure 6.3. The top left panel shows the output from all five runs on the same axis for comparison. This sample suggests that uncertainty in simulator tuning can produce different rates of directional change, but that the direction tends to converge closely after about 20 hours. In terms of performing an analysis, there are a number of options. The simulator output in this case could be parameterised by means of a functional form, in a similar way to the output for fetch-limited growth in chapter 5. We could, for example, fit a relaxation model as discussed (and seen in figure 6.2). However, this form is not flexible enough to fit well in all cases and in the absence of a suitable expression, particularly something with a physical basis, we can proceed by analysing specific points in time. A possible criteria for selecting a specific point could be to choose the point with the highest variance across the design data. Time point t=+4:30 hours exhibits the largest variance (79.5°), which is also apparent qualitatively on inspection of the top left panel of figure 6.3. Use of such a criteria is justified given the objective is to evaluate

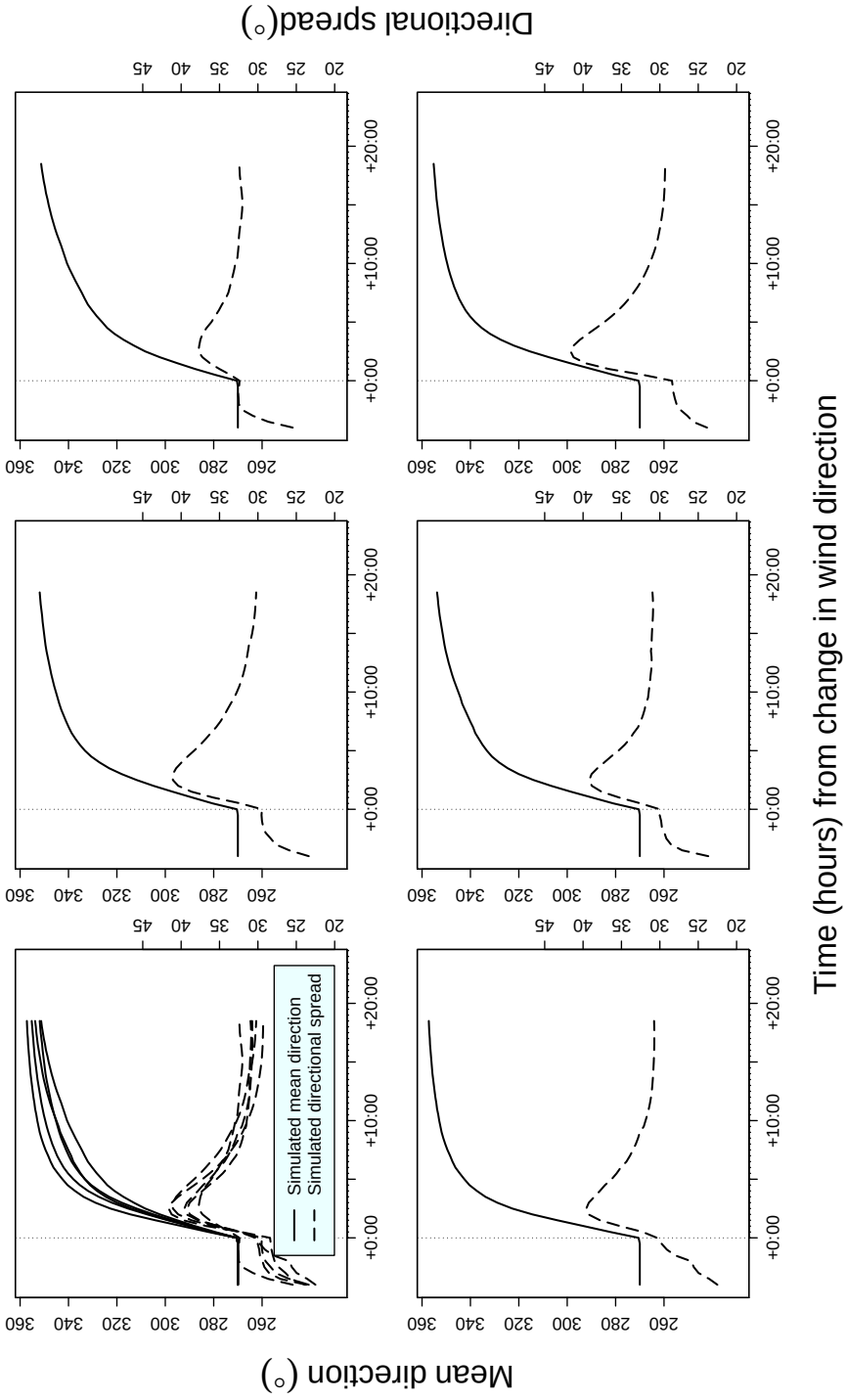
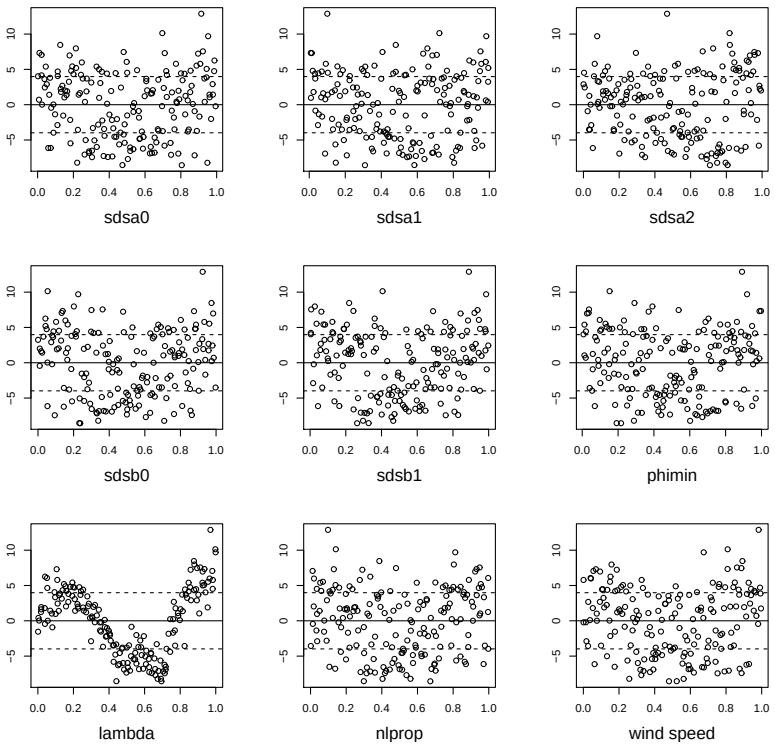


Figure 6.3: Output from five different model runs in the first design set. The top left panel shows all the data in the other five plotted on common axes.

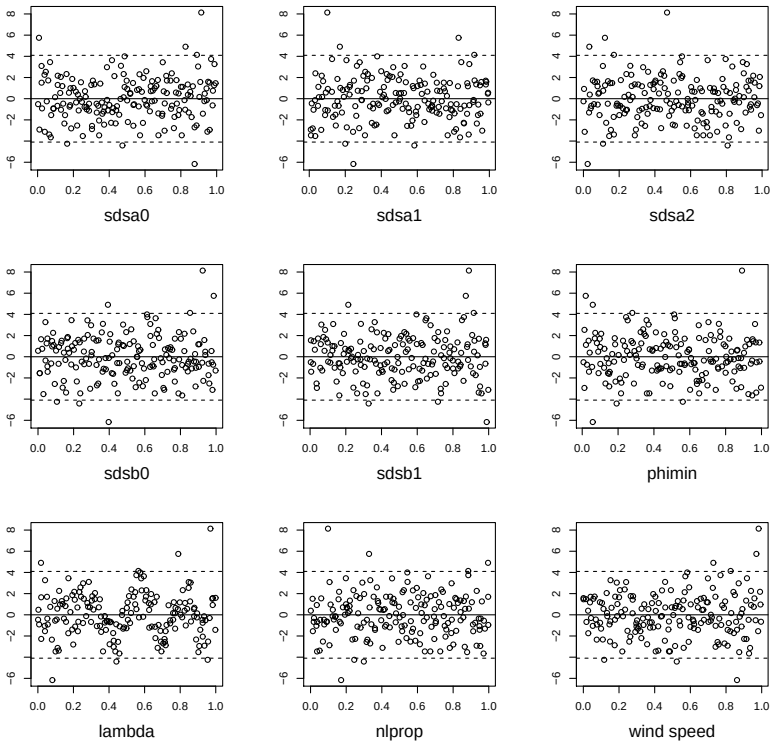
the variance and explain its origin. Note however that all time points in a series appear to be highly autocorrelated and so similar analysis results would be expected for each point. Also notice that because the initial direction is the same in all cases, the magnitude of the direction at most time points is directly correlated to the initial rate of directional change.

So in proceeding to formulate an emulator for mean wave direction at t=+4:30, initial linear regression, shown in figure 6.4 (a), reveals an output dependence very similar to that seen in figure 4.15. A strong nonlinear correlation with λ_{DIA} is obvious. An updated regression model of the form, $mean\,dir \sim 1 + x_{[1,2,3,4,5,6,7,8,9]} + x^2_{[3,4,7]} + x^3_7 + x^4_7$, was formulated residuals from which are shown in figure 6.4 (b). An emulator was constructed using the same formulation for the prior mean function and a Matérn ($\nu = 3/2$) covariance function. Two training data sets (totalling 120 points) were sufficient to build an emulator with robust statistical properties. Results from the emulator validation are shown in figure 6.5. The observed Mahalanobis distance was 55.0 (compared with the reference distribution having mean and s.d. of 60.0 and 13.9) and the graphical diagnostics do not reveal any obvious conflicts. Using the emulator posterior mean function a Monte Carlo uncertainty analysis was performed to obtain an uncertainty distribution, shown in figure 6.6. The fairly rapid decay from the mode into a broad tail is likely due to the strong nonlinear response to λ_{DIA}. Sensitivity analysis confirms this, the results of which are shown in figure 6.7. The total variance, with s.d. $= 4.1°$, is small given the total change in wind direction of 90°. Recall that the nonlinear wave-wave interactions play an import role in the response of the sea to a changing wind direction, so sensitivity to the tuning of the DIA was expected. Note also that the input uncertainty, $\lambda_{DIA} \sim N(0.25, 0.03125)$, is larger than that used in the previous chapter so we might expect other input to be more influential were this reduced. The other input to the simulator has little influence, giving a change of no more than ± 2 degrees over the entire range of variation.

We now move on to consider the behaviour of wave directional spread under conditions of uncertain input. The same design and simulation output were used, a small sample of which we have already seen in figure 6.3. Initial linear regression reveals a response similar to mean wave direction, yielding a linear model of the form $spread \sim 1 + x_{[4,5,7,8]} + x^2_7 + x^3_7 + x^4_7$. Residuals from this model plotted against input are shown in figure 6.8 (a). Four design sets were used to train the emulator in order to reduce the posterior variance on point estimates. Results from the graphical emulator validation are shown in figure 6.8 (b). The observed Mahalanobis distance was 54.8 and the graphical diagnostics do not reveal any obvious conflicts. As before



(a) First order terms only



(b) Higher order terms in λ_{DIA}

Figure 6.4: Residuals from linear regression of mean wave direction plotted against input.

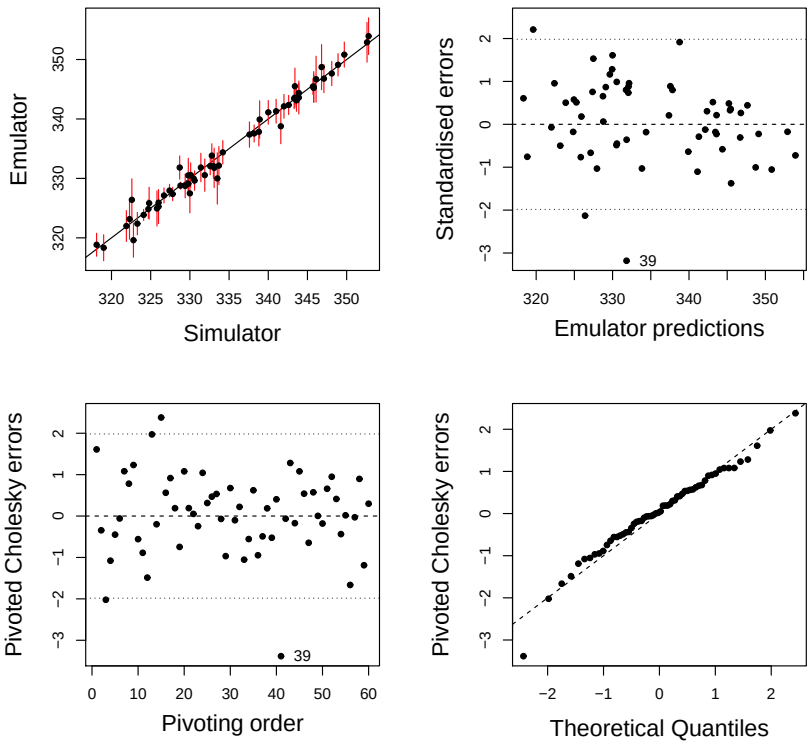


Figure 6.5: Graphical diagnostics for an emulator trained on simulated mean wave direction at t=+4:30. Linear mean and Matérn ($\nu = 3/2$) covariance functions were used.

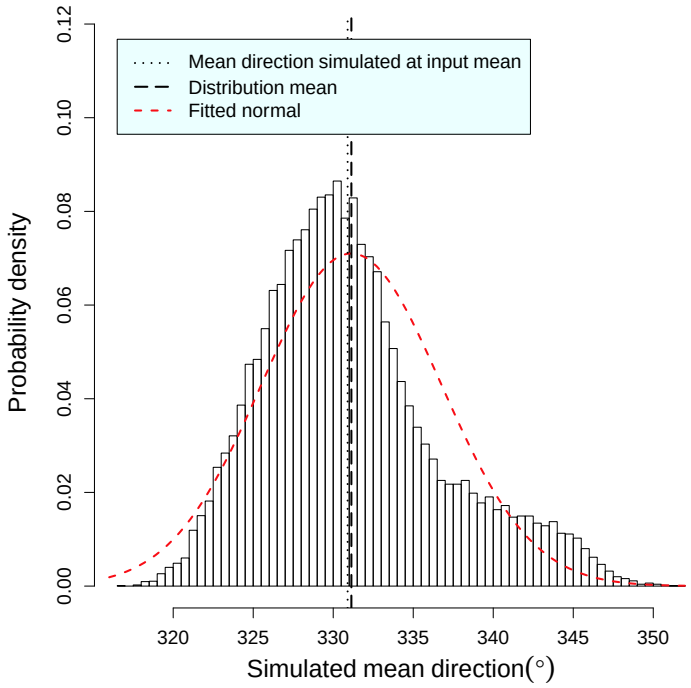
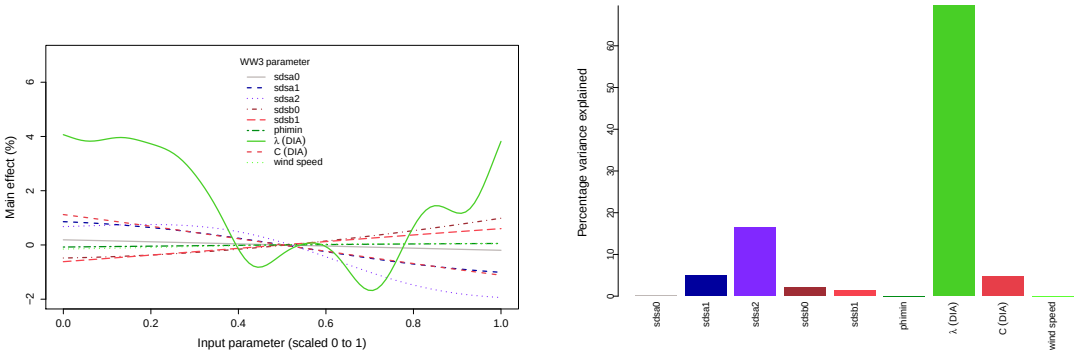
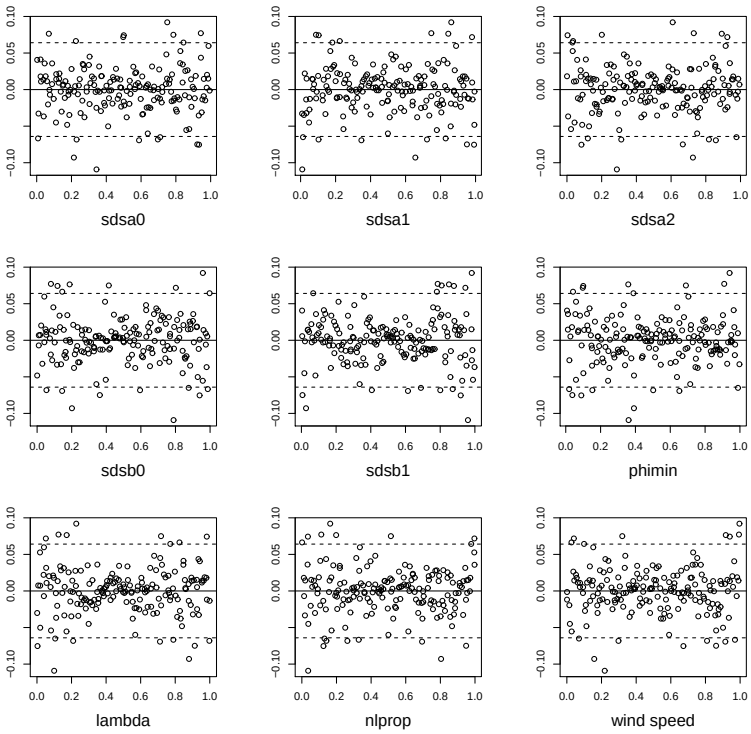


Figure 6.6: Uncertainty analysis for simulated mean wave direction computed at time t=+4:30 hours.

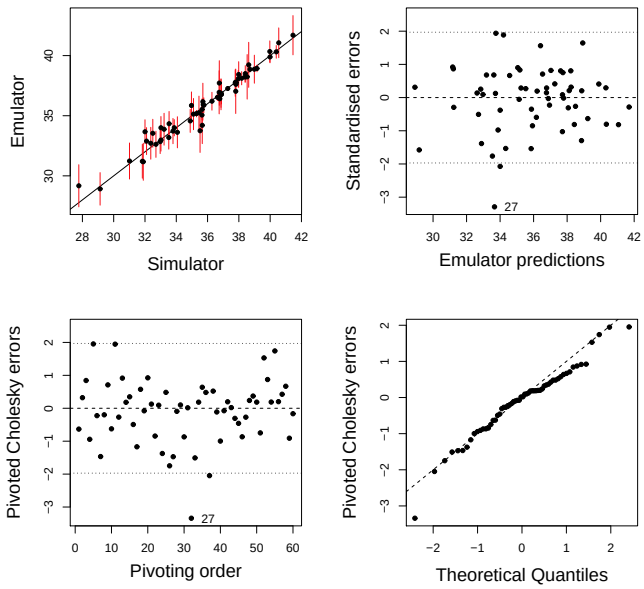


(a) Main effects on mean wave direction. **(b)** Mean = 331; S.D. = 5.6

Figure 6.7: Results of the sensitivity analysis for mean wave direction at t=+4:30 hours.



(a) Residuals from linear regression of wave directional spread plotted against input



(b) Graphical diagnostics for an emulator trained on simulated wave directional spread at t=+4:30. Linear mean and Matérn ($\nu = 3/2$) covariance functions were used

Figure 6.8: Residual plots from linear regression and emulator diagnostics.

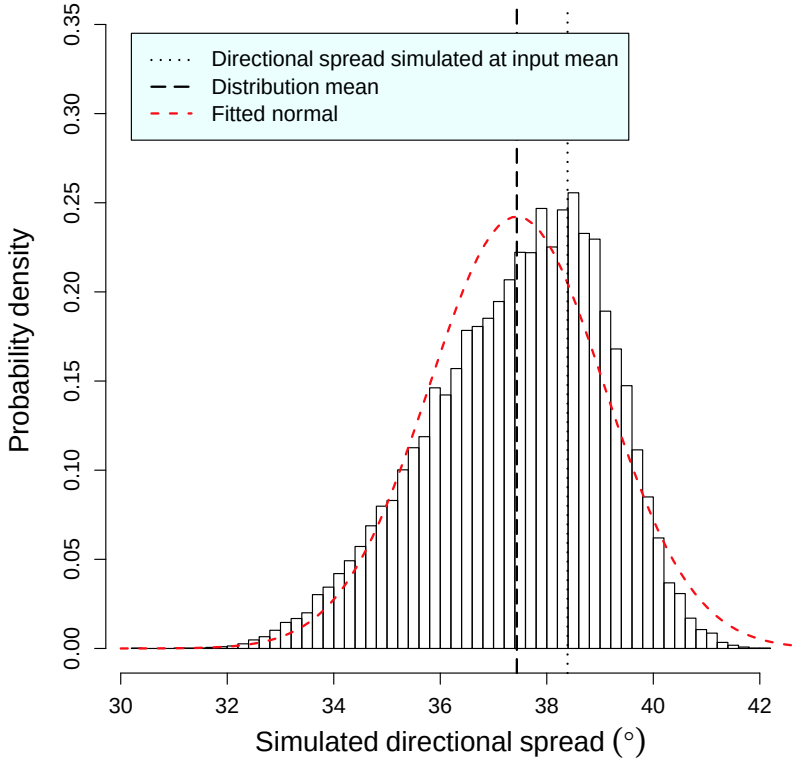


Figure 6.9: Uncertainty analysis for wave directional response computed with the DIA nonlinear interaction scheme. $Mean = 37°, s.d. = 1.7°$. Some nonlinearity is apparent from the skewed distribution.

the emulator posterior mean function was used to perform a Monte Carlo uncertainty analysis. Figure 6.9 shows the output distribution which, interestingly, appears to resemble a reflection of the uncertainty distribution for mean wave direction. This is explained by considering figure 6.2, which shows that after approximately t=+2:30 directional spread becomes inversely correlated with mean direction. Note however that the distribution has s.d. $= 1.6°$, which is small in absolute terms. In an experimental context it is questionable as to whether such a small variance would be detectable. Given the low variance a sensitivity analysis is not performed in this case. It may be that directional spreading is more sensitive to input uncertainty in a different physical context but this is not investigated further here.

This experiment has revealed that tuning of the DIA scheme plays an important role in the simulation of wave direction, again highlighting the importance of nonlinear wave-wave interactions. Noting this importance, and before concluding this experiment, a comparison is made with similar simulations but where the ex-

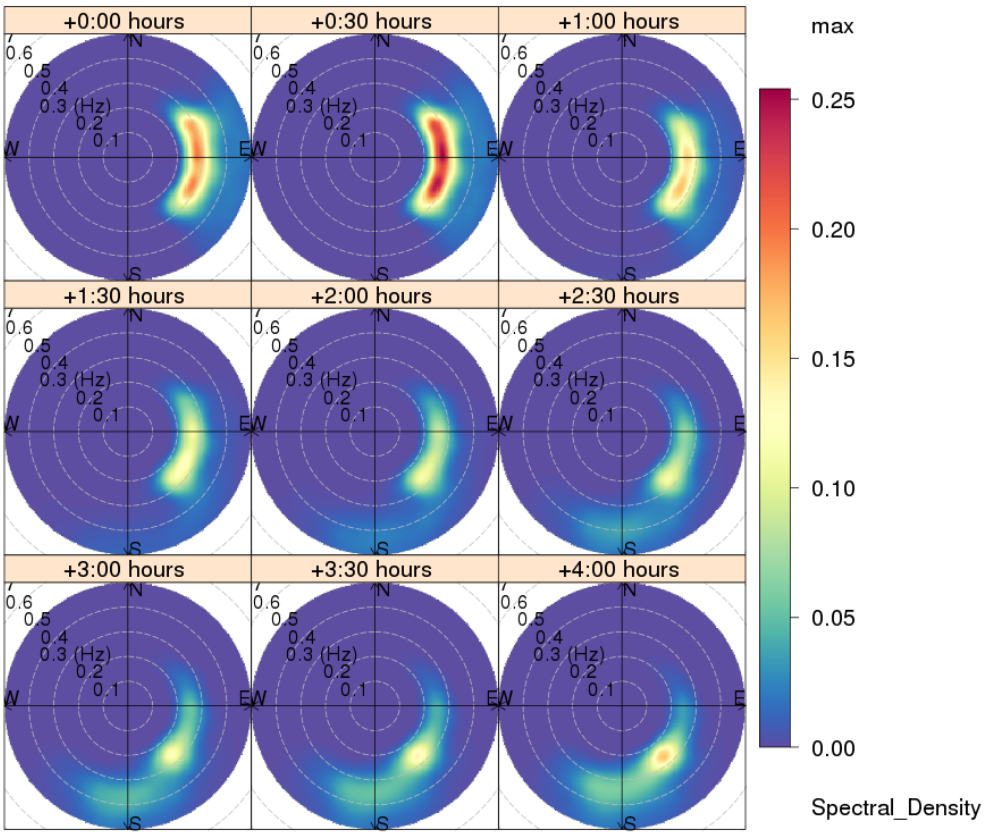


Figure 6.10: Time evolution of the simulated 2-D spectrum generated from a turning wind, as computed using the near exact WRT nonlinear interaction scheme. Times indicated are relative to the shift in wind direction, not the start of the simulation. Note that the colour corresponding to the spectral intensity has been smoothed for visualisation purposes and may not capture some of the fine detail.

act method of computing the nonlinear interactions (denoted WRT[49]) is employed in place of the DIA scheme. The exact method is not routinely utilised by wave modellers and operational centres owing to its computational burden. For these simpler cases however it is viable. The same design data is used to generate four more training data sets[50]. Note that the two dimensions that were previously associated with the two DIA tuning parameters, λ_{DIA} and C_{DIA}, were not used. This is less efficient in terms of a space-filling design but it is less time consuming to progress in this way rather than to specifically create and run a new design. An inspection of the resultant 2-D spectra from the "mean" run are shown in figure 6.10. It is remarkable that even in this simple case there is a noticeable difference

[49]The so called Webb-Resio-Tracy scheme, see section 2.1.4.

[50]A 60 point design took 35 minutes using the WRT computation method, as opposed to 1 minute with the DIA scheme.

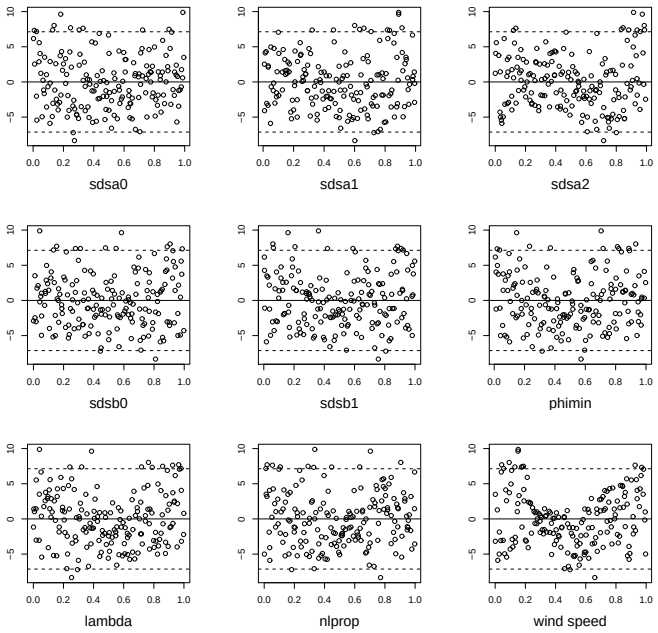


Figure 6.11: Residuals from linear regression of mean wave direction plotted against input.

between the spectra resulting from the DIA approximation and the WRT solution. The former, seen in figure 6.1, shows a "uniform" spectral shape, that exhibits little temporal variation other than its direction. The latter more clearly shows the formation of a new wind-sea (spectral mode) at 270° from approximately t=+02:00 onwards. Furthermore, the initial spectral peak appears to diminish almost entirely before the formation of the new system. While the DIA spectra give the impression of a regular "turning spectra", the WRT spectra have a much more irregular form throughout their evolution. Given that the WRT scheme is a near exact solution we could regard the spectra in figure 6.10 as being closer to the "truth" but this would be complacent. Recall that the dissipation scheme is tuned on the basis of using the approximate DIA scheme and therefore uncertainty is still present. Nonetheless, making use of the WRT scheme means that tuning of the DIA is no longer relevant and so influence of λ_{DIA}, for example, is removed. In order to see whether simulated wave direction responds in the same way to uncertain input, an uncertainty analysis is performed. Using the same training data for mean wave direction at time t=+04:30 (as in the previous cases) linear regression was performed to identify an emulator prior mean function. Residuals plotted against input are shown in figure 6.11. Evidently, after removing uncertainty about tuning the DIA scheme, and the strong nonlinear influence of λ_{DIA}, wind speed exhibits the most nonlinear correlation with the evolution of directional waves. Other inputs exhibit only linear

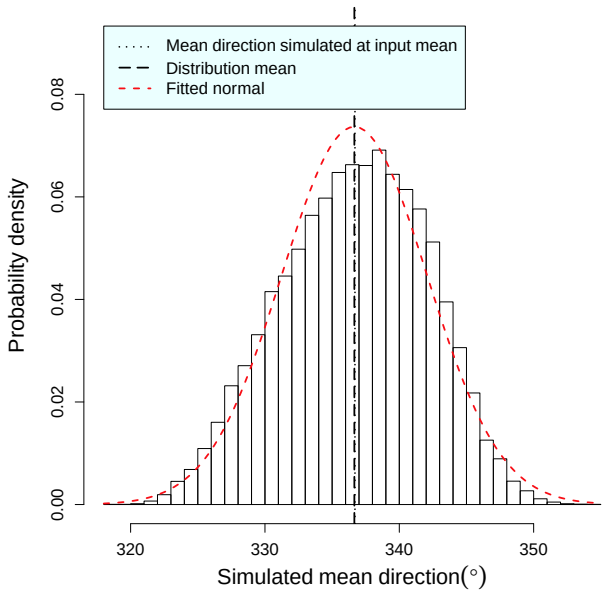


Figure 6.12: Uncertainty analysis for wave directional response computed with the WRT nonlinear interaction scheme. $Mean = 337°, s.d. = 5.4°$. Note that the output at the mean input, and distribution actually coincide in this instance, indicating a very linear response of the simulator to the uncertain input.

relationships to the simulation output. After successfully training an emulator using two design sets with a mean function such that $mean\,dir \sim 1 + x_{[2,3,4,5,9]} + x^2_{[3,9]}$ an uncertainty analysis was performed. It is immediately apparent from figure 6.12 by the fairly close fit of the normal distribution curve that the simulator response is quite linear. This is predictable given that wave energy dissipation does not have particular directional importance. The mean value of 337° is slightly higher than the distribution mean for the DIA scheme (331°), suggesting a slightly closer fit to the relaxation model discussed at the beginning of this experiment. The variance in this case is however almost exactly the same, with a s.d. of 5.6°. This is interesting because having removed λ_{DIA}, the dominant source of uncertainty, we might have expected the variance to have been reduced by an appreciable amount. Given it has not, it would appear that the WRT scheme interacts with the wind input and dissipation to generate an equivalent variance to the DIA (conditional on λ_{DIA} and C_{DIA}).

This analysis brings the experiment to a close. We have seen that mean wave direction is sensitive to parameter tuning, and particularly to λ_{DIA} which has a highly nonlinear influence, when using the DIA scheme. The total variance is modest but shows that output uncertainty may be appreciable in some cases. In contrast to other situations, seen in chapter 5, wind speed is not found to be dominant. In

practice of course wave direction is computed on a spatial grid which can lead to the loss of directional information in some cases (e.g. the garden sprinkler effect occurs due to lack of spectral resolution when propagating waves, see section 2.2.4). A spatial grid is therefore introduced in the next experiment, which also involves a more complex wind specification.

6.4 Experiment 6.B (a): Turning wind with advection

6.4.1 Introduction

This experiment is an extension of experiment 6.A. The objective is to measure the effect of uncertainty about certain aspects of the wind on wave direction. As in experiment 6.A, the simulation involves a turning wind, however in this case complexity has been added by parameterising the wind which allows for control over how the change in wind direction takes place. By making use of the parameterisation, only a single computer experiment is run, but two different analyses are performed, which comprise the two parts (a) and (b) of this experiment. The wind parameterisation is explained in section 6.4.3. In contrast with experiment 6.A, the simulation is now run on a 2-D spatial grid, with a resolution of 50×50 cells and a total extent of 1000×1000 kilometres, comparable with chapter 5.

Before describing the experimental configuration in detail, section 6.4.2 more generally discusses the issue of experimental design for winds. Section 6.4.4 then briefly discusses how uncertainty in initial conditions is controlled before results of the statistical analysis are presented in section 6.4.6.

6.4.2 Experiment design for winds

Before continuing with the experimental description, this section provides a context for the approach to the investigation of winds in this chapter, and also has relevance to chapter 7. Sensitivity analysis in the previous chapters established the importance of accurate wind input, which is widely recognised as crucial to good wave modelling. The wind is stochastic in nature, so to conduct an experiment it was specified as some kind of average - a single number defining the speed in the previous examples. However, the use of an average value clearly ignores any short term variability. Moreover, for global wave simulations temporal resolution is not often better than a few hours so there is little scope to capture short term variation.

In the idealised cases that have been investigated so far the “fixed” wind conditions arise specifically from the intent of researchers to observe such conditions in nature. However, as simulation complexity grows, so does that of the forcing wind.

In experiment 6.A a simple turning wind was precisely specified, but it is easy to imagine real variability in its behaviour, such as the rate of turn, the initial and final wind speeds, and so on. In larger scale realistic cases it may be very difficult to express the variability in such clear terms. Possibly, in localised or specific regions, or on appropriate times scales, winds may be well characterised by a small number of "average" parameters (e.g. a strong prevailing direction). The use of such averages is particularly important in this research because, in order to perform uncertainty analysis using an emulator, a design is required that spans the simulator input space. The input space so far has largely been in relation to input parameters, which lie on convenient scales, and can usually be regarded as uncorrelated. The (average) wind speed can also be regarded conveniently as a parameter for the purpose of design and analysis. However, it should immediately be clear that where the input is stochastic, or has stochastic properties, the "input space" becomes more difficult to define. Specifically, as the representation of the wind becomes more complex (like a realistic wind), it cannot generally be well described by a convenient set of parameters lying in some orthogonal space. Furthermore, even if it could be, how would that representation relate to wind uncertainty - and what in fact do we mean by "wind uncertainty"? In order to investigate complex winds in a way consistent with the approach so far, it is clear that these issues need to be resolved.

To define uncertainty about wind is certainly as much a philosophical problem as a practical one. To do so implies we have obtained, or can elicit, meaningful information or beliefs about wind variability, and then quantify them in a mathematically convenient and tractable way. The type of scientific question would dictate how this might be approached. For example, in the case of a wave *forecast* the winds may be generated by an atmospheric simulation, and it may be possible to obtain some probabilistic information about them by perturbing the input (parameters) of the atmospheric simulator in some meaningful way. In the case of producing a *hindcast* a similar approach might be possible, if using an atmospheric simulator, but typically the objectives of the hindcast would be different. That is, hindcasts are often used as a means of testing and measuring the skill of a simulator, and therefore the forcing winds would probably be of the "best" quality, in an effort to minimise the source of error. Historic wind products sometimes have known uncertainty, such as low or high regional biases, and so on. Such information could then be used, perhaps by simply adjusting and sampling the average wind speed commensurate with the uncertainty, in a similar way to previous experiments, where the wind speed was regarded as a random variable. If the wind data were obtained from observation stations, then it is possible uncertainty information could be derived based upon

known equipment accuracy.

Our beliefs and specification of uncertainty therefore depends on the situation and the questions being asked. Whatever the circumstances may be, in order to make use of an emulator a design space has to be specified. Where some kind of overall systematic error is present, akin to the idealised cases investigated so far, this seems possible. That is, wind speed is regarded as a dimension of input and incorporated into the design. However, where we have reason to believe the form of variability is much more complex, such as could be imagined in the case of forecasting realistic winds, we must find a mathematical basis that allows us to specify the uncertainty in a meaningful way, and that also allows us to incorporate it into a design. As mentioned, if an atmospheric simulator could be perturbed (via input parameters) to provide probabilistic information, it may be possible to directly incorporate that input into a design. The wind uncertainty would then be directly linked to some parametric scale as required. Winds are not generated from an atmospheric simulator in this thesis and so this approach is not discussed further.

If the winds were obtained solely from direct observation and are fairly abundant, a comparable approach might be to analyse the wind climatology in order to identify and quantify the variability in some way. For example a principal component analysis may yield dominant modes and their probability distributions, which could then be used to formulate meaningful uncertainty information. Each mode could then be regarded as a dimension of input and incorporated into a design. This still leaves some important questions, such as whether the "designed" winds are sensible, or even physically realistic. This approach is not pursued further in this thesis but an example of how a design over winds could be generated, using synthetic data, is given in appendix F.

In this experiment a parameterised wind representation is introduced, in order to make inference about a wider range of wind properties. Bearing in mind idealised cases are still under consideration, and we are largely examining the sensitivity of the wave model to idealised changes in wind, a parametric formulation based upon a stream function is proposed. This allows physically realistic wind variability, in terms of average speed and direction, and temporal variation of those properties, to be specified and easily incorporated into a designed experiment. The approach is explained in detail in the experimental discussion in the next section.

6.4.3 Wind configuration

The objective of this experiment is to investigate uncertainty about a wider range of wind behaviour, and its effect on wave direction, in a similar way to the turning

wind. In experiment 6.A the wind turned suddenly through 90°. But consider the situation where there is uncertainty about exactly how far the wind will turn, and how quickly it will turn. How might this affect the prediction of waves? Investigating this situation will provide a deeper insight into the sensitivity of the simulator to wind conditions. In order to perform an uncertainty analysis the wind is now represented parametrically. The wind speed components in the x and y directions are derived from a stream function to ensure consistency with physical theory. This is described shortly. In addition, two parameters, θ_{trans} and λ_{trans}, are incorporated into the generating function to allow for the adjustment of the directional change and transition period respectively.

In terms of the experimental time line, the wind can be thought of as initially blowing from the "west", which is from a direction of 270°. At some time, t_1, the wind begins to change direction before reaching its new direction, $270° + \theta_{trans}$. Note that the new direction is always reached at time t_2, and t_1 can be a maximum of 6 hours prior to t_2. t_1 is controlled by the parameter λ_{trans} which can take values $-1 < \lambda_{trans} \leq 1$. The relationship between λ_{trans}, t_1 and t_2 is therefore $t_1 = t_2 + 6\lambda_{trans}$. Consideration of the consequences of different values of λ_{trans} reveals more about how we can control the behaviour of the wind. Firstly, set $\lambda_{trans} = 0$. t_1 then equals t_2 and so there is no transition time. If $\theta_{trans} = 90°$ then we would recover the same conditions as experiment 6.A, which is an abrupt change of 90° in wind direction. Secondly, if $-1 < \lambda_{trans} < 0$, then the transition time would be anything up to 6 hours, depending on the exact value. Thirdly, if $0 < \lambda_{trans} < 1$ the behaviour changes slightly. The wind maintains its original direction until t_2 when it abruptly changes direction, but t_1 becomes positive so a transition begins *from* the new direction back to the original direction. t_1 is then effectively after t_2, up to a maximum of 6 hours. However, note that $\lambda_{trans} = 0$ corresponds to a six hour transition, and $\lambda_{trans} = 1$ corresponds to a instantaneous directional change, followed rapidly by a reversion (a little like a delta function). In a sense this "breaks" the symmetry of the scheme but it allows for an experimental design over a continuous range that includes $\lambda_{trans} = 0$ without introducing an erratic "pulse" in wind direction. We make use of this feature in experiment 6.B (b) where it is discussed further. Figure 6.13 shows a schematic diagram of the time profile of the wind for three different variations in control parameters. Note that in practice the simulator requires wind updates to be input at specified times. Therefore, where a transition period is specified (i.e. where $\lambda_{trans} \neq 0$), the transition period is divided into twelve time steps. A transition time of 6 hours therefore corresponds to (maximum) time steps of 30 minutes each. Note also that the simulator linearly

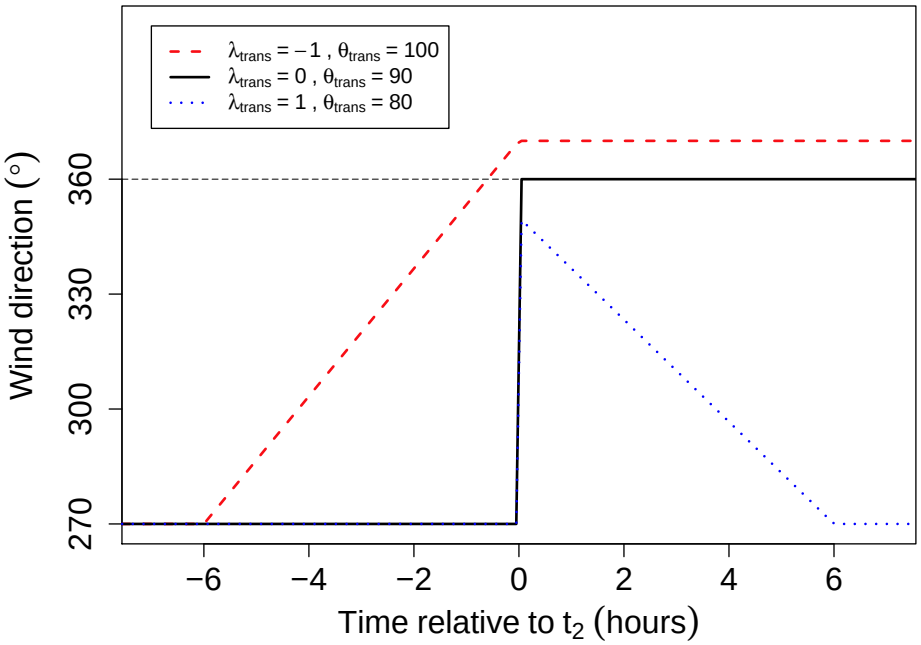


Figure 6.13: Wind directional profile in time for different choices of control parameters.

interpolates winds in time so the transition retains the intended linear change in direction.

In order to create winds that conform to the described situation it is straightforward to create them directly. However, we can make use of a stream function to demonstrate consistency with physical laws in this, albeit, simple case. If the objective was to generate winds in a more complex situation, with greater spatial and temporal variability, it would be necessary to ensure that they were physically consistent in this way. The stream function arises from the continuity equation for incompressible flow which in two dimensions is,

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \tag{6.1}$$

The flow velocity $\mathbf{u} = (u, v, 0)$ can be written in terms of a vector potential $\Psi = (0, 0, \psi(x, y))$ such that $\mathbf{u} = \nabla \times \Psi$ and then,

$$u = \frac{\partial \psi}{\partial x} \quad \text{and} \quad v = -\frac{\partial \psi}{\partial y} \tag{6.2}$$

So by specifying a functional form for Ψ, and obtaining the flow velocity vectors from (6.2) it is possible to obtain fairly arbitrary flows that are physically consistent. In

order to obtain winds consistent with the aforementioned scheme, the following stream function is used.

$$\psi(x,y,t) = s_{wind}\left\{y\ cos\left(\alpha(t,\lambda_{trans})\frac{2\,\pi\,\theta_{trans}}{360}\right) + x\ sin\left(\alpha(t,\lambda_{trans})\frac{2\,\pi\,\theta_{trans}}{360}\right)\right\} \tag{6.3}$$

where $\alpha(t,\lambda_{trans})$ is a linear function of t that results in transitions according to the diagram in figure 6.13. (The complete form is not written for brevity.) Note that (6.3) is linear in x and y and so the resulting velocity components are spatially homogeneous over the grid (consistent with experiment 6.A).

The approach here remains somewhat contrived but is primarily intended as a means of examining the simulator sensitivity to more complex winds. Proceeding then, a single designed experiment is conducted, the output from which is used in both parts of experiment 6.B.

6.4.4 Initial conditions

The new wind specification necessitates further consideration of the initial conditions. This is because the transition period is with respect to a fixed completion time (i.e. not a fixed start time). That is, a longer transition period must begin at an earlier time in the simulation in order to complete at the designated time. In previous experiments initial conditions were typically "integrated out" by running the simulation to a steady state in time. In order to avoid this problem a spin-up period is ensured so that the initial sea state is approximately the same regardless of the transition time. Figure 6.14 shows the evolution of the sea state, in terms of H_s and T_p, in the mean run. We can see that after a period of 36 hours a maximum is reached and the system is in an equilibrium state. Simulations are therefore subject to a spin-up time of 36 hours to minimise initial conditions uncertainty[51].

6.4.5 Experimental configuration and design summary

In addition to the uncertainty distributions for WW3 input parameters specified in table 6.2, the control parameters listed in table 6.3 jointly form the input specification for experiment 6.B. Note that the experiment is designed over ranges of $[-1,1]$ and $[45,135]^\circ$ for λ_{trans} and θ_{trans} respectively. Four 80 point maximin LHS based designs were run.

[51] Note that there may be a small amount of variation in the initial sea state due to the choice of input parameters.

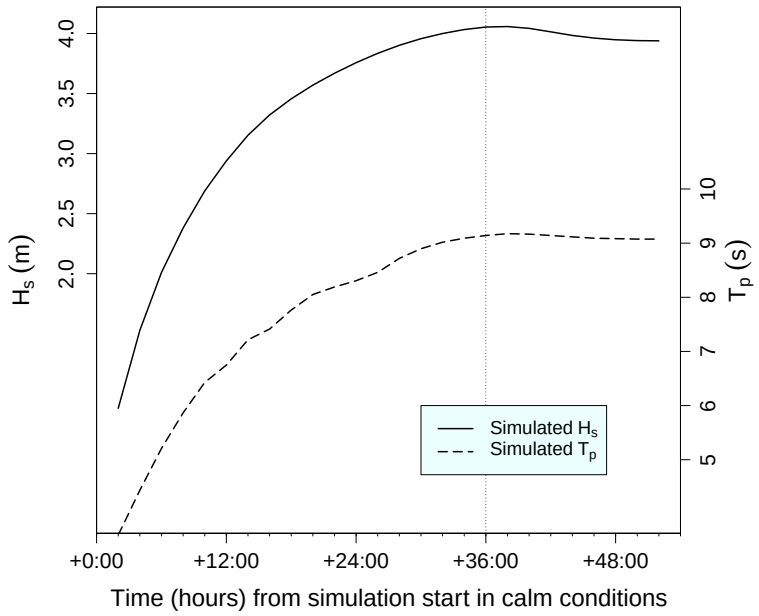


Figure 6.14: Sea state development with time on the 2-D grid.

Table 6.3: Wind control parameters and sampling distributions for experiments 6.A, 6.B (a) and (b)

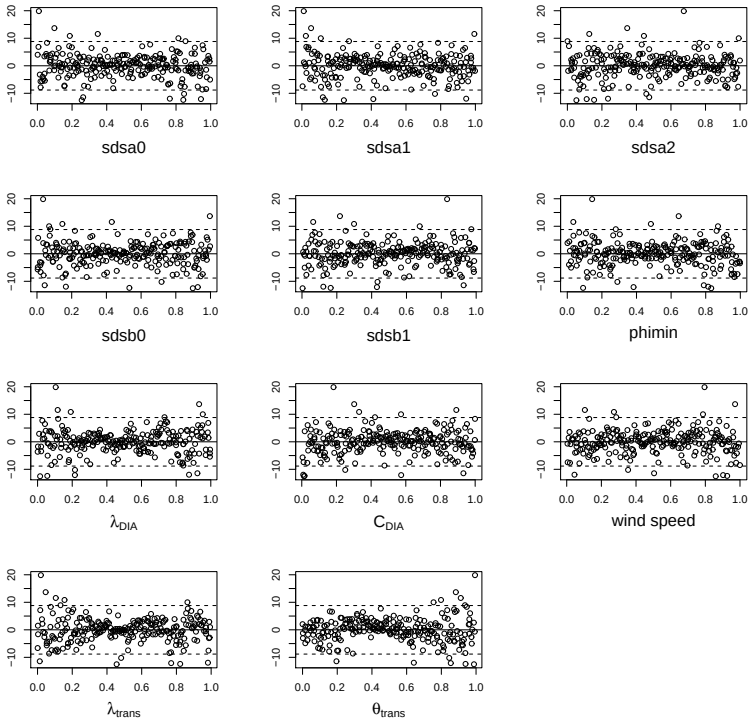
Control variable	**Fixed value in experiment 6.B (a)**	**Sampling distribution in experiment 6.B (b)**
λ_{trans}	0	$Beta(1, 5)$
θ_{trans}	90°	$N(90^\circ, 10^2)$

6.4.6 Analysis

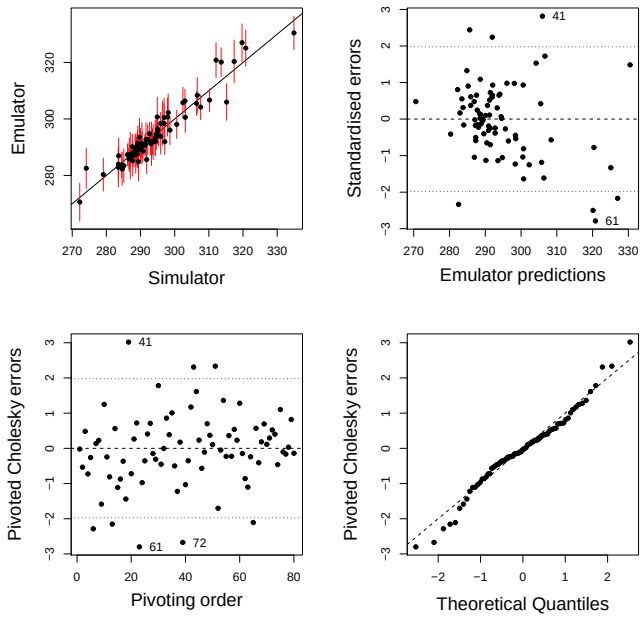
Having discussed the approach to the wind specification and the resulting experiment design, we now move on to the analysis. Some steps in the analysis, including the formulation of the emulator are relevant to experiment parts (a) and (b), since the same emulator is used. Given a spatial grid of 1000km × 1000km output of mean wave direction at a point near the centre of the grid (500km, 545km)[52] is used for analysis. Waves at this point are due not only to the local wind conditions but also due to energy propagated from upwind regions of the grid. Noting the spin-up time of 36 hours and a wind speed of 10.0ms^{-1}, conditions will be fetch-limited at 545km, although well developed. Recall that in experiment 6.A a criteria of $f_p = 2f_{PM}$ was used as a basis to specify the onset of the changing wind direction. An important difference here is that the sea state is more developed ($f_p \approx f_{PM}$).

In experiment 6.A, maximum variance of wave direction was adopted as the criteria for the point (in time) of analysis. However, owing to the experimental design in this case, this criteria ceases to be so meaningful. In fact, the variance is found to increase with time because there is not asymptotic behaviour in all cases, owing to the effect of variability in λ_{trans}. If the same time point for analysis is adopted for consistency, which was t=+4:30, it corresponds to approximately t=+41:00 in this case (if $\lambda_{trans} = 0$), relative to the start of the simulation. Proceeding on this basis in order to draw a comparison an emulator can be constructed. Linear regression suggests that a mean function of the form $mean\,dir \sim 1 + c(1, x_{[3,7,10,11]} + x^2_{[7,10]} + x^3_{[4,10]} + x^4_{[7,10]})$ is appropriate, implying nonlinear response to parameters *sdsa2*, λ_{DIA} and λ_{trans}. Input plotted against residuals is shown in figure 6.15 (a). The emulator, fitted with a nugget term of 0.04, appears to validate well as suggested by diagnostics shown in figure 6.15 (b). Before using the emulator to perform an analysis, consider that in order to re-create experiment 6.A in terms of the input uncertainty specification, λ_{trans} and θ_{trans} can be regarded as known, and set to 0 and 90° respectively. Adopting the same specification for the remaining uncertain input (shown in table 6.2) we can proceed to perform an uncertainty analysis by sampling from the emulator posterior mean function. The resulting uncertainty distribution for mean wave direction, shown in figure 6.16, can be compared with the distribution computed in experiment 6.A, shown in figure 6.6. Immediately it is clear from a simple visual inspection that the two histograms have a similar shape, although their parameters are quite different. The means are 289° and 331° respectively, showing that significantly more directional change has taken place in the earlier experiment.

[52]The 45km offset from the centre is due to an irregular scale in the simulator point output specification.



(a) Residuals from linear regression of mean wave direction plotted against input



(b) Graphical diagnostics for an emulator trained on simulated mean wave direction at t=+41:00 hours.

Figure 6.15: Residual plots from linear regression and emulator diagnostics.

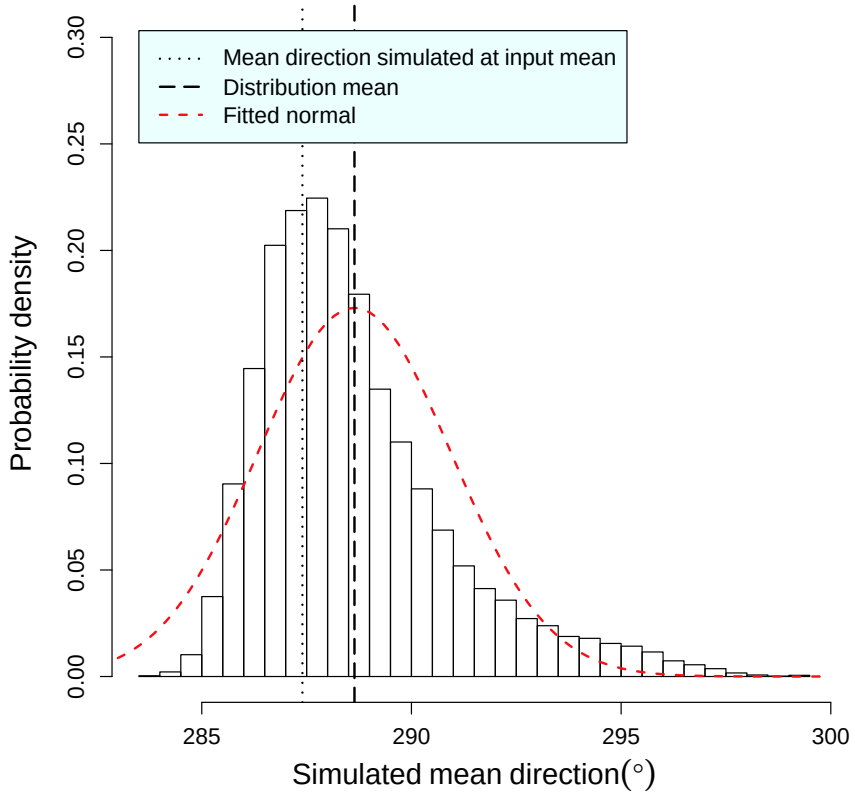


Figure 6.16: Uncertainty analysis for simulated mean wave direction computed at time t=+41:00 hours.

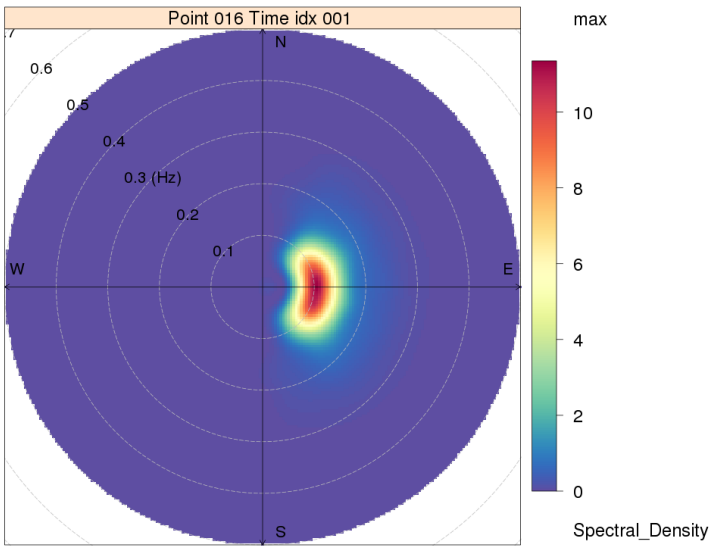
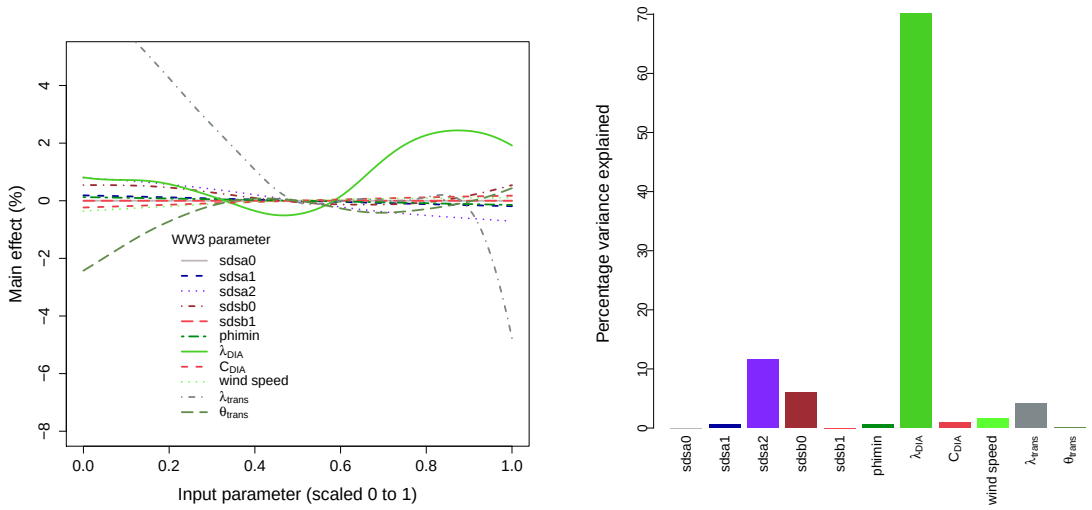


Figure 6.17: 2-D spectrum at a fetch of 545km, at t=+36:00 hours, prior to the change in wind direction.

The variance is also much larger in the earlier case. Consider however that in this case wave energy is constantly propagating into the measurement region, even after the change in wind direction. The waves arriving will retain directional properties from earlier sea states (i.e. the original wind direction) and therefore the directional change will be much slower than for the homogeneous case. Bear in mind also that the new wind sea being created is at much higher frequency and lower energy, and it will take time for nonlinear interactions to transfer energy to lower frequencies in the new prevailing wind direction. Looking at the sea state more explicitly it is possible to learn about how the changes are manifested. The sea state has had time to develop, and therefore has a well defined *directional* peak, seen in figure 6.17. Contrast this with figure 6.1 and note that the directional peak is considerably broader. Where the directional change occurs at an earlier sea state we may see the variance increase significantly. A sensitivity analysis will not reveal why the total variance is low as such, but we want to see how the input parameters contribute to the total. Before reviewing the result a point is made about this particular analysis because a new issue is introduced. The analysis is being performed on the basis that λ_{trans} and θ_{trans} are known exactly, and they are thus specified with zero variance. Without modification, the semi-analytical approach to the computation (see appendix D) produces spurious results because the integration scheme cannot correctly handle a probability distribution with zero variance. In order to avoid the issues (at least temporarily), input distributions with small variances were specified such that $\lambda_{trans} \sim N(0, 0.05^2)$ and $\theta_{trans} \sim N(90, 1.0)$. A standard deviation of 0.05



(a) Main effects on mean wave direction. **(b)** Mean = 289; S.D. = 2.3

Figure 6.18: Sensitivity analysis for mean wave direction at 545km fetch, at t=+36:00 hours.

for the distribution of transition time corresponds to approximately 15 minutes. Figure 6.18 (a) shows the main effects of each input parameter on the mean wave direction. Recall that the main effects show the influence of each parameter over the entire input space of the design, and that here the prior variance of λ_{trans} for the uncertainty analysis is small (thus for the sensitivity analysis tightly constraining its variability to the centre region of the figure). In spite of its highly nonlinear behaviour, the contribution to the total variance from λ_{trans} is therefore small, and λ_{DIA} is again seen to be most influential. With the exclusion of the influence of λ_{trans} these results are highly consistent with the results of experiment 6.A, although the total variance in this case is very much smaller.

Noting the previous comments relating to the maturity of the sea state in this case in contrast to experiment 6.A, by way of better comparison a quick investigation of the case for younger sea states can be undertaken by utilising different output data from this experiment. That is, the geographic grid gives rise to fetch-limited wave conditions and so a variety of sea states are available for analysis. So far in this experiment data near the centre of the grid, at a fetch of 545 km, were analysed. However, by considering wave conditions at shorter fetch, the sea state will be "younger" and more dominated by wind-sea. Output data at 75km, the shortest fetch available in this case[53], is therefore studied. The 2-D spectrum at t=+36:00 is shown in fig 6.19. It is clear to see that the sea state is less developed

[53]Owing to the choice of output points specified in the WW3 simulation configuration.

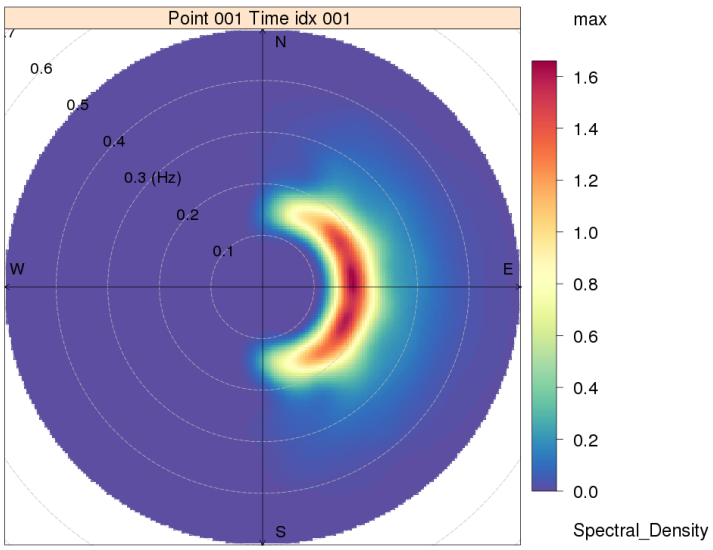
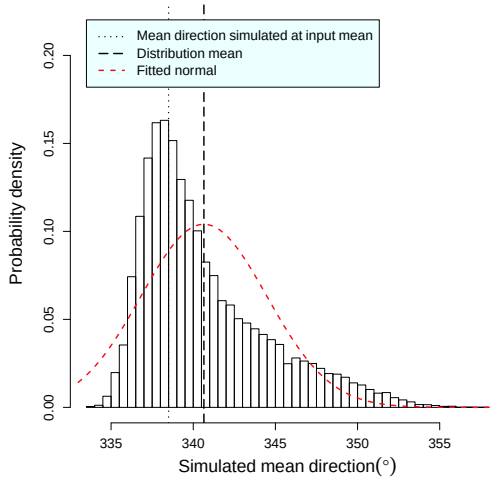


Figure 6.19: 2-d spectrum at a fetch of 75km, at time t=+36:00

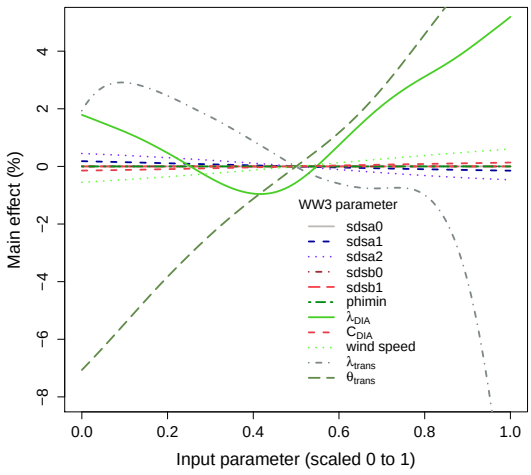
($f_p \approx 0.16$Hz) and that the directional spread is much greater. Proceeding in the same way, an analysis is performed by training an emulator to the output data for mean wave direction at time t=+41:00. Using a mean function of the form $y \sim 1 + x_{[7,10,11]} + x^2_{[7,10]} + x^3_{[7,10]} + x^4_{[10]}$, it was found that the emulator validated well with 160 training points (two design sets). Results of uncertainty and sensitivity analysis are shown in figure 6.20. The output distribution now has mean = 341° and s.d. = 3.9°, which is much closer to the results for experiment 6.A (of mean = 331° and s.d. = 5.6°). The main effects of the input are very similar to those at 545km fetch, but notice now that θ_{trans} has a much stronger and more linear influence. Owing to their small input variances the effects of λ_{trans} and θ_{trans} are not manifested in the uncertainty distribution, which is dominated by λ_{DIA}. Input parameters related to energy dissipation remain largely inactive.

So, in summary, the behaviour of wave direction on a homogeneous grid is closely comparable to that at short fetch. Sensitivity is quite low is all cases but is dominated by λ_{DIA} and control parameters related to the wind. In developed sea states wave direction is less sensitive, and change is (unsurprisingly) observed on longer time scales.

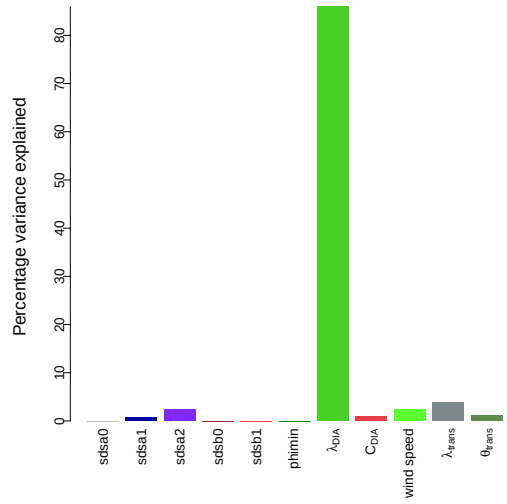
Directional spreading is not investigated for the following reasons. At time point t=+41:00, and 545km along the fetch the mean and variance of directional spread for the design data are 40.5 and 7.0^2 degrees respectively. At time point t=+41:00, and 75km along the fetch the mean and variance of directional spread for the design data are 28.0 and 4.0^2 degrees respectively. Noting that the joint uncertainty distribution



(a) Output uncertainty distribution.



(b) Main effects.



(c) Mean = 341; S.D. = 3.9

Figure 6.20: Uncertainty and sensitivity analysis for mean wave direction at 75km fetch, at t=+36:00 hours.

for the input will constrain that variability considerably, particularly considering the most influential parameters relating to the wind are constrained to a very small variance, it is judged that further analysis is of limited value since there is very little variance to explain.

In the next part of this experiment, the analysis is extended by specifying larger variance for the wind control parameters λ_{trans} and θ_{trans}.

6.5 Experiment 6.B (b): Uncertainty in change in wind direction

Imagine a situation where a change in wind direction could occur but there is uncertainty about exactly *how* it will proceed. That is, it could be a sudden shift in direction, as a weather front moves though for example, or it could be a more gradual change over a period of a few hours. There may also be uncertainty about the new direction of the wind. In such a situation it would be useful to obtain probabilistic information about the possible resulting conditions.

In order to perform such an experiment we can make use of the wind parameterisation, and emulators that have already been formulated. In part (a), the wind control parameters λ_{trans} and θ_{trans} were constrained by a small input uncertainty, but now they are respecified. Elicitation is required to determine the appropriate uncertainty specification, which might typically proceed by examining a set of wind observational data relevant to the region of interest. However, given the idealised and hypothetical nature of this experiment the following is proposed. Firstly, λ_{trans} governs the rate of the directional change. If we suppose that the change takes place on a time scale between 3 hours (with zero probability) and zero hours (with maximum probability), with probability increasing exponentially, the appropriate sampling distribution for λ_{trans} could be something like that shown in figure 6.21. The probability density function is based upon a *Beta*-distribution with shape parameter values of $\alpha = 1.0$ and $\beta = 5.0$. Secondly, a specification for the probability of θ_{trans} is required. In the absence of a review of observational data, it is simply asserted that the directional change is uncertain with a mean value 90°. A Gaussian distribution is therefore specified such that $\theta_{trans} \sim N(90°, 10^2)$. Here we maintain the assumption that all uncertain input is uncorrelated, including λ_{trans} and θ_{trans}. However, were the probability distributions obtained from observations it may prove that this assumption is too strong. The results of uncertainty and sensitivity analysis, given the new input specification, are shown in figure 6.22. Results from both short (75km) and long (545km) fetch are shown and it is interesting to see the difference between the two cases. At short fetch, output variance is dominated by the

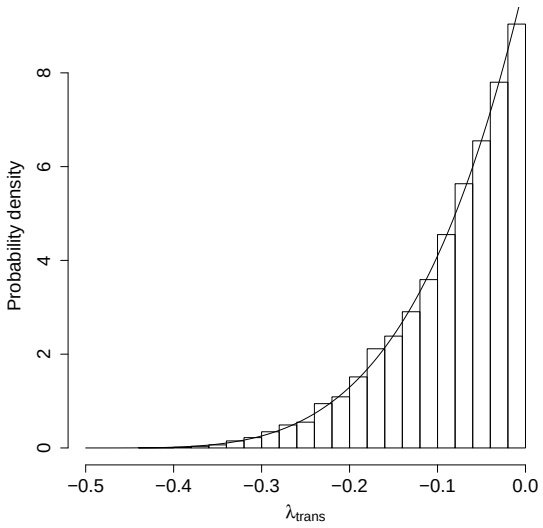


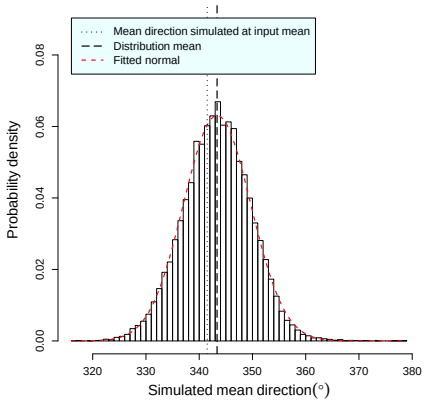
Figure 6.21: Sampling distribution for control parameter λ_{trans}.

influence of θ_{trans} and as a result the uncertainty distribution is highly Gaussian. Contrast this with output at long fetch where we see that λ_{trans} dominates output uncertainty and as a result the distribution is skewed towards lower angles. The difference in response of the two different sea states is quite striking. At short fetch where the sea state is not so well developed the new wind direction is very important to the short term evolution of the wave spectrum. The s.d. of 6.3° is appreciable. At long fetch near the grid centre, the inertia of "the system" is clearly quite large since the variance remains low and the resultant wave direction (at between 5 to 8 hours after the directional change) is largely insensitive to θ_{trans}. It would be reasonable to conclude that in coastal waters at short fetch accurate knowledge of the wind is crucial to good directional simulation. Note finally that λ_{DIA} is seen to be influential in spite of the others sources of uncertainty, and the importance of accurate nonlinear wave-wave interactions is again reiterated. This experiment concludes the investigation of the simulation of a turning wind.

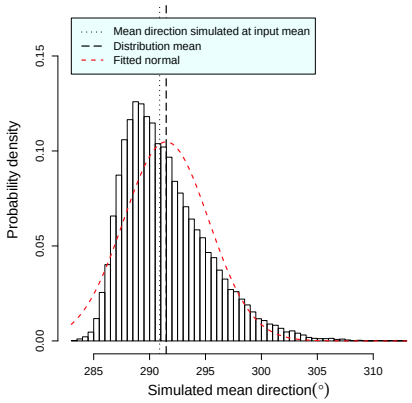
6.6 Chapter summary

The effects of uncertain input on simulated mean wave direction in the context of a wind turning through 90° were analysed. Uncertain input includes parameters relating to energy dissipation and nonlinear wave-wave interactions, together with control parameters that govern the behaviour of the wind. Important results are summarised as follows.

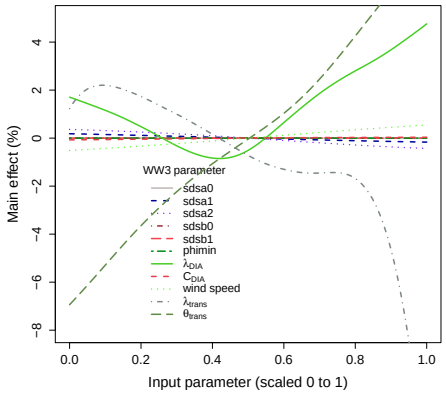
In the analysis of simulation output for a source function only (no advection)



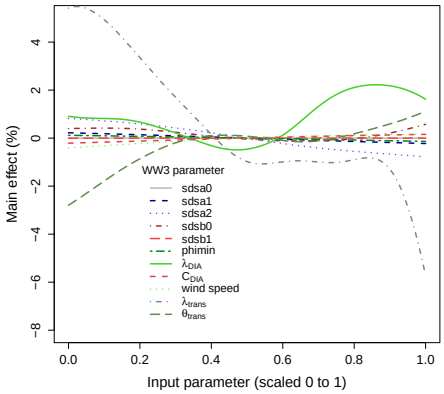
(a) Output uncertainty distribution at 75km fetch.



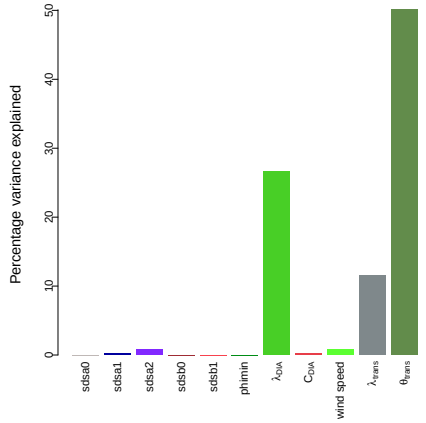
(b) Output uncertainty distribution at 545km fetch.



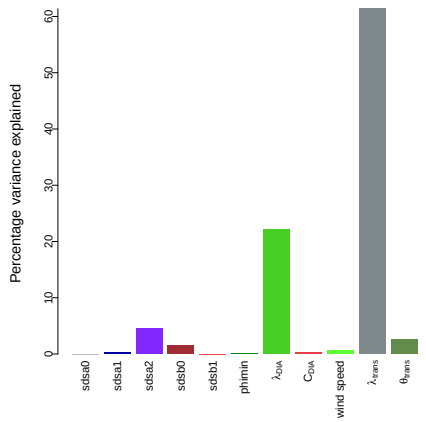
(c) Main effects.



(d) Main effects.



(e) Mean = 343; S.D. = 6.3



(f) Mean = 291; S.D. = 3.8

Figure 6.22: Results from uncertainty and sensitivity analysis at t=+41:00 hours.

and a case with advection at 75km fetch on a 1000km square grid, similar output uncertainty distributions were observed suggesting commonality in how the sea state responds to the change in wind direction. In the case for the source function only, analysis was conditional on the change in wind, and input parameter λ_{DIA} was found to dominate output uncertainty ($> 70\%$ variance). When a parametric representation of the wind was employed for the case at 75km fetch, wind control parameters governing the rate and magnitude of the directional change were found be influential jointly accounting for over 60% of output variance. Mean wave direction was found to be most sensitive ($> 50\%$ variance) to the magnitude of the change in wind direction. Input parameter λ_{DIA} was also found to be highly influential ($> 25\%$).

At longer fetch (545km), output uncertainty was dominated by the rate of change in direction (60%), and input parameter λ_{DIA} (20%). The total variance however was 60% of that at shorter fetch (3.8° as opposed to 6.3°), highlighting the inertia of the more developed sea state.

Results show that input parameters governing energy dissipation were largely inactive in all experiments, indicating that, at least in terms of how WW3 operates, energy dissipation does not play an important role in simple directional phenomenon. This does not seem entirely consistent with van Vledder and Holthuijsen (1993) who conduct numerical experiments and determine that the effect of S_{nl} on turning rate is an order of magnitude smaller than the effect of S_{ds} and S_{in}. However their study is not the same, and is with respect to the exact (WRT) nonlinear scheme, and the DIA is known to exhibit excessive directional spreading. Additional study would be required to say more about this. A qualitative comparison of the full nonlinear wave-wave interaction calculation, and the approximate DIA scheme was made and distinct differences in directional structure of the 2-D wave frequency spectrum were observed. Together with the observed influence of λ_{DIA}, this is further evidence of the importance of tuning the DIA scheme appropriately.

7 Uncertainty analysis for a real 2-D basin: Lake Michigan

Having examined the effect of uncertainty about parameter tuning and wind input for a range of idealised cases we now turn our attention to a real case. This broadens the scope of possible investigation in at least two important ways. Firstly, real wind data can be used. This can include a much higher degree of spatial and temporal variation than we have seen so far and may result in much more pronounced *dynamic* effects. Forcing winds are discussed in detail in the next section. Secondly, it is now possible to incorporate observational data more directly. We have previously seen simulation output compared qualitatively to observational data in chapter 5, in the specific physical case of fetch-limited wave growth. Wind and wave conditions are monitored on Lake Michigan by operational data buoys that provide hourly observations of the direction integrated wave spectrum and summary statistics. This abundant data can be used to learn more about uncertainty in the simulation of waves. Specifically we can obtain measures of departure of simulator output from observations, and learn about bias, and possible structural error in the simulator. This is discussed further in section 7.9. In addition to these important aspects of the investigation, different physical processes may need consideration. For example, the lake is shallow in parts and so bottom friction or shallow water 3-wave nonlinear interactions may be relevant. Although such effects are not investigated in this chapter, primarily because points of observations and analysis are located in deep water, discussion of these additional sources of uncertainty is provided in section 7.3.

The research in this chapter therefore investigates how the effect of tuning uncertainty affects simulations of a real case, and whether the findings that we have seen so far in idealised cases are comparable. At the end of the chapter, emulators are used to determine a choice of parameter values that give better performance in energetic conditions. In the next section, the choice of Lake Michigan as the focus of research is discussed and justified. Section 7.2 describes the typical wind and wave climate observed on the lake, followed by section 7.3 which discusses elicitation of relevant uncertainty information and section 7.4 which summarises the experiments conducted. Since dynamics are now a consideration, before conducting formal statistical analysis, an experiment in section 7.5 investigates "spin-up" of the simulator, in order to identify the typical time scales required to minimise the effects of uncertainty about initial conditions. In sections 7.6, 7.7 and 7.8 a number of wind regimes lasting periods of three to four days, and comparable to those investigated in chap-

ters 5 and 6, are used to force simulations where uncertainty about simulator tuning is present. Emulators are used in a similar way to previous experiments in order to evaluate the effects of parameter uncertainty on simulated output of summary wave statistics at buoy locations. Finally in section 7.9 a simple calibration experiment is performed for prevalent (energetic) wave conditions seen typically in October. Two new parameterisations are identified that give much better performance of the simulator than the default settings.

7.1 Motivation

The choice of Lake Michigan for the final series of experiments is justified as follows. It is a well studied case, the subject of operational forecasts and is a test case for wave model updates (see e.g. Liu et al., 2002; Rogers and Wang, 2007; Ardhuin et al., 2010; Alves et al., 2011). With a total length of 494km, fetches can be long enough to generate waves large enough to be dangerous to shipping and coastal areas. An abundance of real world observational data is available from a number of weather stations situated on and around the lake[54]. Of specific importance are two buoys operated by the U.S. National Data Buoy Center (NDBC), designated NDBC 45002, located in the central north of the lake and NDBC 45007 located in the central south of the lake, as seen in figure 7.1 (bottom right image). These buoys capture various environmental data at hourly intervals including an estimate of the direction integrated (1-D) wave spectrum, together with common summary statistics such as H_s and T_p. The lake is large enough to support the evolution of fairly well developed sea states ("short swell" with $T_p \approx 9s$) and wave heights of approximately 5m can be achieved. However, longer swell is not present which prevents wave systems arriving from other (distant) locations. An important implication of this is that interacting sea states do not readily take place and as such the wave conditions are dominated by the local wind sea, which can be well described by summary statistics. These qualities limit the scope of the analysis to that concerning wind-sea, however much of the analysis conducted in the previous chapters is directly relevant as the focus was on developing seas and directional waves. The absence of interacting sea states, and potentially multi-modal spectra, considerably simplifies the analysis which, at this stage, is advantageous. It is therefore argued that Lake Michigan is a good subject at this stage of the thesis. Progression to a full ocean basin would be a natural (and desirable) follow-up to this study and is discussed briefly in chapter 9.

[54] See http://coastwatch.glerl.noaa.gov/marobs/marobs.html for a complete list.

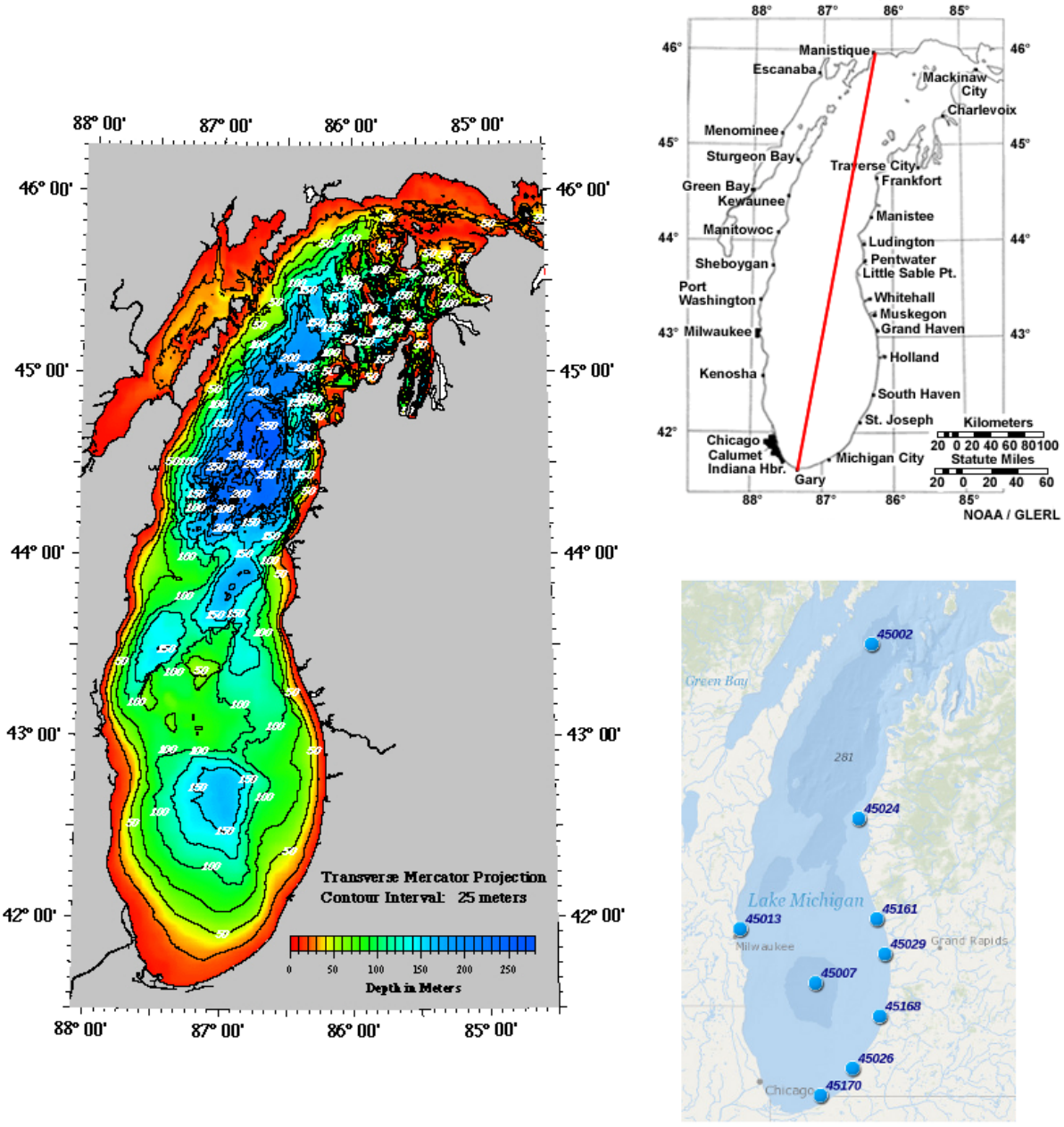
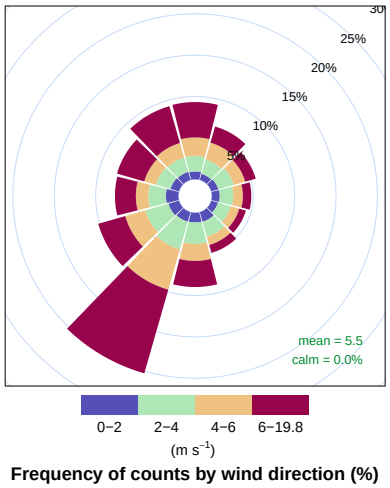
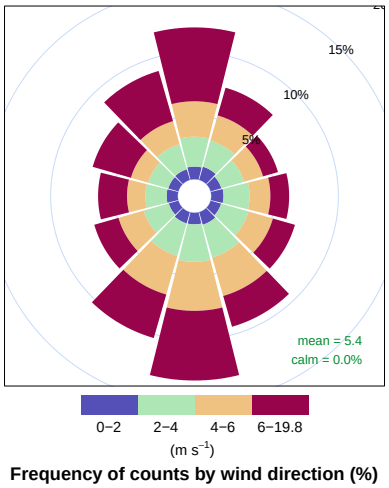


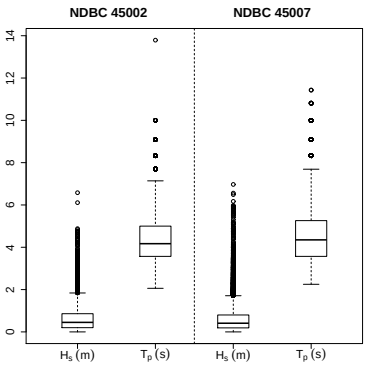
Figure 7.1: Lake Michigan: The left panel shows the lake geography together with its bathymetry. The top right panel shows some additional regional information, together with an indication of the longest fetch, marked by the red line. The bottom right panel shows the location of the buoys NDBC 45002 and 45007.



(a) Winds measured by NDBC 45002.



(b) Winds measured by NDBC 45007.



(c) Box plots showing the distribution of H_S and T_p.

Figure 7.2: Summaries of 5 years (2008 to 2012) of observations at NDBC buoys 45002 and 45007, comprising H_s, T_p and winds.

7.2 Wind regimes and wave climate

Abundant wind data is available from the numerous weather stations located throughout the region of the North American Great Lakes. Hourly wave data is provided by a number of buoys, including NDBC 45002 and 45007 (which are used in this chapter and provide by the NDBC[55]). Summary statistics for the wave and wind climate on the lake measured at the buoys for the five year period between 2008 and 2012 (inclusive) are given in figure 7.2. Wind climate is shown in panels (a) and (b). The prevailing conditions correspond well to the orientation of the lake, that is, in terms of the where the largest extents of open water lie with respect to the buoy locations. For NDBC 45002 this is south-southwest and for NDBC 45007

[55] See e.g. http://www.ndbc.noaa.gov/station_history.php?station=45002

this is to the north and south. Although the surrounding land is fairly low lying, the prevalence of the directions aligned with the lake suggests that it attenuates all other wind directions.

The distributions of H_s and T_p are seen in panel (c). In terms of H_s we can see that the distributions for both buoys are comparable with incidents of slightly larger waves in the south of the lake being more common. The median value of H_s is less than 1 metre for both locations on the lake illustrating that wave conditions are generally subdued. The situation is similar for T_p where the distributions are closely comparable. With approximately 50% of observations of T_p between 2 and 4 seconds, the sea state is generally not developed. However in the south of the lake an appreciable number of observations show T_p exceeding 6 seconds.

Both buoys provide a measurement of the direction integrated spectrum. Although the analysis in this chapter is concerned with wave summary statistics, it is instructive to examine the spectral data. Specifically we are concerned with whether the spectra exhibit bi-modality, which might require more careful analysis. Some examples of spectra observed in October (when conditions tend to be more energetic), 2012 are shown in figure 7.3. We see that they are typically unimodal and resemble the form of a JONSWAP spectra. However, in a few cases we can see that the spectral peak has apparently been attenuated, and the spectral form shows bi-modality. There are different possible explanations for this, one is that given a strong wind, a sudden change in wind direction could bring about the rapid evolution of a wave system in the new direction. The associated peak would then appear at a higher frequency, as seen in figure 7.3, panels 3 and 13 (counting across from the top left). In fact an example of this effect produced by a simulation was seen in section 6.3 (e.g. figure 6.10). This is not necessarily the case however and it could also be that the data are unreliable. For example, the bottom left panel shows that H_s and T_p were not calculated or recorded for some reason (such as internal error or noisy reading at that particular time), thus suggesting the spectral information is also unreliable (and that the bimodality is spurious). In this particular case a review of the data does indeed show a change in wind direction of +70 degrees beginning at approximately 22:00 04/10/2012 and lasting until 03:00 05/10/2012. This provides us some confidence that the buoy is reporting observations accurately. These data therefore reveal that bi-modality can occur, and should cause us to be a little cautious about using T_p as a measure of the spectrum since it may not be well defined in some cases.

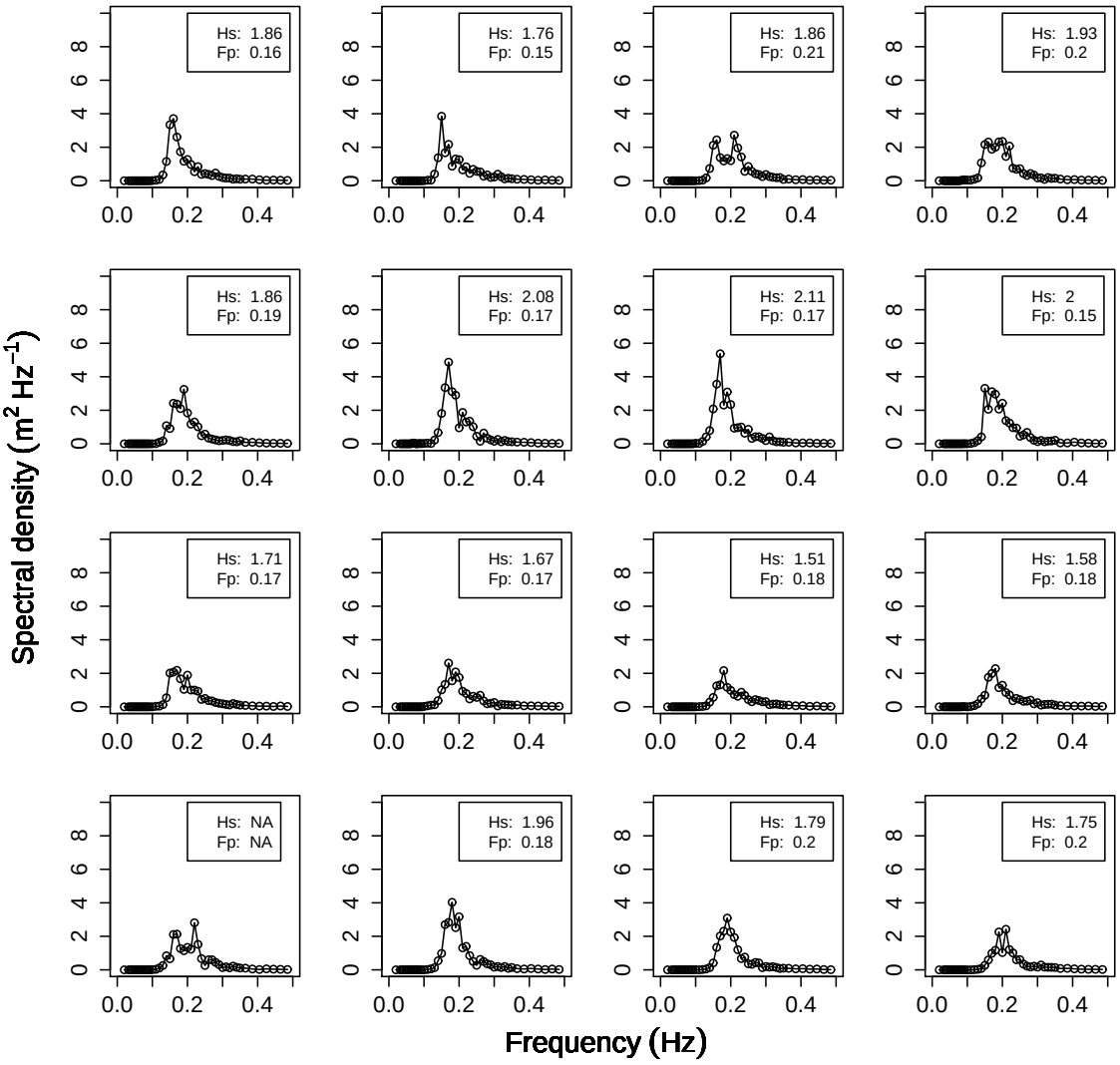


Figure 7.3: Direction integrated (1D) wave spectra observed at NDBC 45002 every 2 hours starting at 00:00 05/10/2012.

7.3 Simulation details, sources of uncertainty and elicitation

A summary of the simulation configuration is given in section 7.3.1 before sources of uncertainty are discussed in the following subsections. Sources include; parameter uncertainty, forcing winds, initial conditions, simulation numerics and bathymetry. In common with previous experiments in this thesis the parameterisation of wave energy dissipation and nonlinear wave-wave interactions is regarded as uncertain and the input specification for these remains unchanged. The remaining sources are discussed in the following subsections and an experiment design summary is given in section 7.4.1.

7.3.1 Simulator configuration

The input specification is replicated from the operational simulation used by the US Naval Research Laboratory[56]. The simulation differs from previous cases in that the geography is specified on a longitude / latitude grid. The geographic resolution is dictated by the bathymetry which is specified with a resolution of 66×127 that yields grid cells of resolution 0.049×0.036 degrees. Compared with a typical global simulation this arrangement is particularly high resolution. A spectral resolution of 29 frequency bins with 36 directional bins is used. In respect of the bathymetry that includes shallow water, in contrast to previous experiments, WW3 is compiled with the "JONSWAP" bottom friction parameterisation. The full switch configuration used to compile the WW3 source code is given in appendix E.

7.3.2 Forcing winds and elicitation of uncertainty about wind conditions

Approximately 100 observation stations, including NDBC 45002 and 45007, are located in and around Lake Michigan and the wider great lakes region. Researchers at the Great Lakes Environmental Research Laboratory (GLERL[57]) produce data sets of winds for Lake Michigan by interpolating the data from all observational platforms. The resulting winds are specified on a longitude / latitude grid with a resolution of 131×251. Note that these winds are at approximately twice the resolution of the geographic grid. However, bearing in mind that they are interpolated from much more sparsely deployed weather stations, the high resolution should not be interpreted as an indication of excellent fidelity.

A communication from GLERL (see appendix G) suggests that little information is readily available as to what uncertainties are associated with the wind data. In

[56] Kindly provided by W. Eric Rogers, NRL, Mississippi.

[57] http://www.glerl.noaa.gov/

principle, given access to the interpolation scheme, and details of the "over-water" correction method it may be possible to specify and express uncertainties in a way that could then be included in the experiment design. This could potentially incorporate knowledge of uncertainties associated with specific weather stations or regions, if available. The approach would then require the formulation of "uncertain" winds, where the uncertainty was expressed using some kind of suitable parameterisation. In effect it would be similar to the approach demonstrated in chapter 6, where synthetic winds were generated through a simplistic parameterisation.

Owing to the complexity of the problem, and lack of readily accessible uncertainty information, the analysis in this chapter is conditional on the winds as given.

7.3.3 Initial conditions

Analysis in previous chapters largely avoided the issue of uncertainty associated with initial conditions by running simulations to an equilibrium state - essentially to a solution independent of time. In the case of dynamic forcing winds, the wave simulator will converge to a solution, but there will be a period of time between the start of the simulation and when convergence has been achieved. The length of that period of time will be a function of the accuracy of the initial conditions. For example, the initial conditions may be a calm sea but if the "real" conditions (that is, the simulation of the real conditions) are more energetic then it will take some time for the simulator to "spin-up". If the initial conditions are a close match to the real conditions then the spin-up period will clearly be shorter.

Thus the uncertainty in initial conditions discussed here describes how close those conditions are to the converged solution. Note that for a wave simulator this differs from the problem of initial condition uncertainty as typically encountered when running atmospheric models, for example, where the dynamics of the system can cause it to diverge drastically if the initial conditions are incorrectly specified. The wave simulator converges regardless, but the time taken to do so will be related to how close the initial conditions match the converged solution. As such a spin-up will generally be necessary to ensure convergence.

This issue is likely to be far less consequential in a simulation of a large lake than in a larger regional or global situation, since the state-space of possible wave conditions is much smaller. Importantly there will be a lack of swell, or waves generated remotely that would be particularly important in a global setting. Clearly in a global simulation, waves generated from a distant storm could take a long time to arrive, and if the specification of initial conditions was incorrect, those distant waves may arrive too late or too early (or not all!). By contrast, a lake lacks this kind

of possibility and we are therefore only concerned with a spin-up to local conditions.

Provided sufficient time is allowed, uncertainty about the initial conditions will be inconsequential. WW3 allows the user to initialise a simulation by attempting to approximate a fetch-limited JONSWAP spectrum at each geographic point from the local wind conditions. This feature reduces spin-up time, and so is utilised for all the simulations in this chapter. The spin-up time required is investigated and determined in experiment 7.A (section 7.5).

7.3.4 Simulation numerics

Simulation numerics are not investigated in this chapter however a few points are noted. Compared with global simulations commonly run with spatial resolutions of around one degree, and a directional frequency spectrum of 24 frequency bins in 24 directions, the resolution used here is very high. The spatial resolution is of the order of hundredths of a degree and the frequency spectrum is represented with 29 frequency bins in 36 directions. This is computationally achievable because the simulation is of small geographic extent, and given the high resolution, numerical uncertainty or errors arising due to coarse resolution, are minimised and not expected to contribute significantly to simulation error.

It is in fact quite likely that due to the fairly small geographic extent of the simulation, and the constraints of the lake on the possible range of wave systems (i.e. limited to fairly localised systems), very similar results could be obtained from a much simplified simulation. That is, one with much lower spatial and spectral resolutions. This would be an interesting extension to this particular investigation but is not addressed further here.

7.3.5 Bathymetry

A potentially important source of uncertainty when simulating waves in shallower water is bathymetry. The research in this thesis is focussed on deep water only and so uncertainty about bathymetry is not specifically investigated. Lake Michigan is shallow at the coast and the effects of bathymetry may be significant in these regions. However, experiments in this chapter are concerned with conditions measured at the two data buoys NDBC 45002 and 45007. The buoys are situated in water depths of 175m and 160m respectively, and recalling that shoaling occurs only at a water depth below half the wavelength, it is clear that the necessary wavelengths will not be achieved (in excess of 300m) and therefore shoaling is not going to affect waves at these points. Note also that the water depth increases from the shoreline in all

Table 7.1: Experiments presented in chapter 7.

Experiment	**Description**
Experiment 7.A	Investigation of simulation spin-up time.
Experiment 7.B	Parameter uncertainty analysis for southerly wind.
Experiment 7.C	Parameter uncertainty analysis for northerly wind.
Experiment 7.D	Parameter uncertainty analysis for a turning wind.
Experiment 7.E	Parameter calibration for wave conditions during October 2012.

locations (see figure 7.1), so no shallow areas that could affect the assumption of deep water conditions are located far from the shore

7.4 Experiments and analysis

Five experiments are presented in this chapter. The experiments are listed in table 7.1 and are described below.

Experiment 7.A measures spin-up duration using winds over a six day period. The objective is to establish a minimum time required to spin-up in order to remove the effects of initial condition uncertainty. The simulator is run with default settings in this case. Experiments 7.B, 7.C and 7.D are uncertainty experiments conditional on three different wind regimes each lasting a few days. The conditions approximate idealised experiments run in previous chapters and the objective is therefore to determine the effect of parameter uncertainty but also make a comparison with previous findings. The objective of experiment 7.E is to show how emulators can be used to calibrate a wave simulator. Output data from experiment 7.B is used for this experiment and a number of different emulators are constructed for simulated H_s. A cost function based upon the squared differences between the simulation and observations is formulated and minimised in order to find optimal tuning parameter values.

7.4.1 Design summary

Table 7.2 lists the parameters that are regarded as uncertain and the distributions assigned to them. For the designed experiments presented in this chapter, six 80 point maximin LHS designs were generated, as described in section 3.4.2. Design ranges are typically ± 4 s.d. about the mean, according to table 7.2.

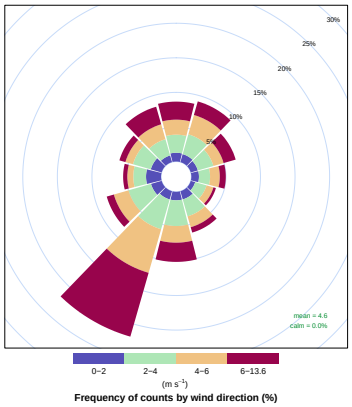
Table 7.2: WW3 input parameters and sampling distributions for experiments 7.B, 7.C, 7.D and 7.E

Switch	**Description**	**Tuning parameter [Variable Assignment in WW3 manual]**	**Default value**	**Sampling distribution**
ST2	Dissipation (TC96)			
	High frequency dissipation	SDSA0 [a_0]	4.8	$\mathrm{N}(4.8, 0.2^2)$
	High frequency dissipation	SDSA1 [a_1]	1.7×10^{-4}	$log_{10}\mathrm{N}(-3.293, 0.177^2)$
	High frequency dissipation	SDSA2 [a_2]	2.0	$\mathrm{N}(2.0, 0.2^2)$
	Low frequency dissipation	SDSB0 [b_0]	3.0×10^{-4}	$\mathrm{N}(-3.25 \times 10^{-3}, (9.375 \times 10^{-4})^2)$
	Low frequency dissipation	SDSB1 [b_1]	0.47	$\mathrm{N}(0.5250, 0.03125^2)$
	Low frequency dissipation	PHIMIN [$\phi_{\mathbf{min}}$]	0.003	$\mathrm{N}(0.003, (1.0 \times 10^{-4})^2)$
NL1	Nonlinear interactions (DIA)			
		LAMBDA [λ]	0.25	$\mathrm{N}(0.25, 0.03125^2)$
		NLPROP [C]	10^7	$log_{10}\mathrm{N}(7, 0.1193^2)$

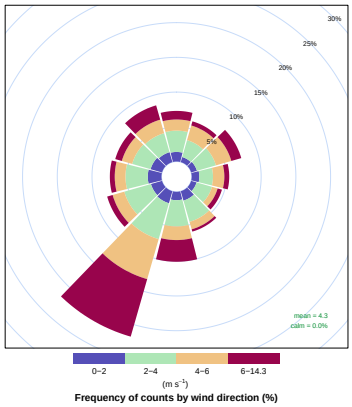
7.5 Experiment 7.A: Spin up times

In this experiment spin-up times are examined but no uncertainty analysis is performed. The objective is to measure the length of time required in order to ensure convergence of the solution in different conditions. Note that WW3 has been configured to initialise the local wave spectrum as a fetch-limited JONSWAP based upon the local wind conditions. This differs from previous experiments where waves developed from calm conditions, and has the advantage of converging to the solution faster. However, simulations initialised during energetic conditions are anticipated to take more simulation time to converge to the true solution. The approach here is therefore to select a period of time during which a range of conditions are present, and run the same simulation a number of times but in each case initialising the simulation at progressively later intervals throughout the period. Note that the issue of spin-up can, to a large degree, be mitigated by a sensible choice of initialisation time (i.e. during calm conditions), which we are generally free to make (but might not always be possible).

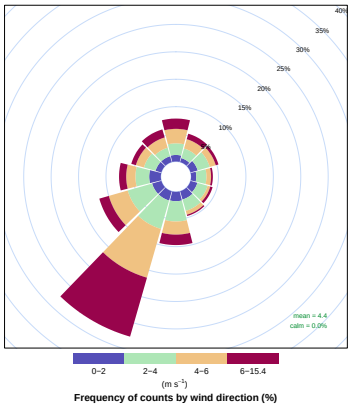
We proceed therefore by reviewing the weather over Lake Michigan throughout the year in order to identify periods of variability. Monthly wind rose diagrams spanning five years (2008 to 2012) for May through to December for winds observed at NDBC 45002 are shown in figure 7.4. Note that during the winter months much of Lake Michigan is covered with ice so observational data are available sporadically, if at all, which causes the spurious looking observations for December, and the lack of data during January to April. Conditions are generally consistent during the spring, and then in August wind strength increases slightly before becoming much stronger and more variable in October. Energetic conditions are the most interesting, most relevant to ocean based activities and safety, and often predicted with the poorest skill. October therefore appears to be a good choice for further investigation. Observations of wind speed and direction measured by NDBC 45002 for the first two weeks in October 2012 are shown in figure 7.5. Observations at the buoy are shown together with an average wind, derived from seven points lying approximately evenly spaced on a line of constant longitude along the lake axis. The data were obtained from the GLERL interpolated winds. The average data provide an indication of the spatial variability which in general is quite low, that is, the winds measured at the buoy are a good representation of the winds throughout the entire lake. It can be seen that during most of this period the wind speed consistently exceeds 5ms^{-1} with highs of 15ms^{-1} in some instances, and there is substantial directional variability. The largest waves are generated by winds from the north or south, aligned to create the longest fetch, and so these conditions are



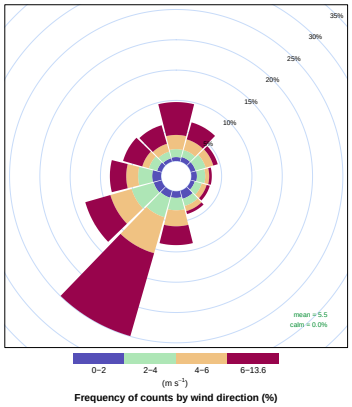
(a) May



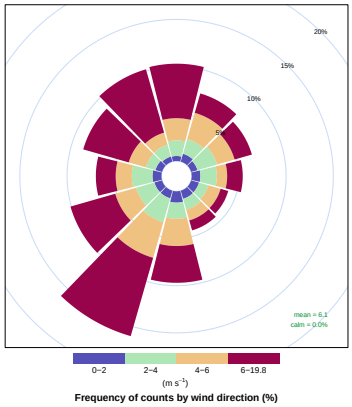
(b) June



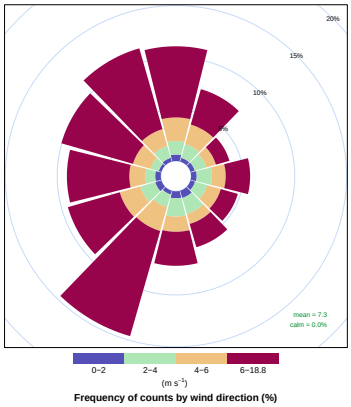
(c) July



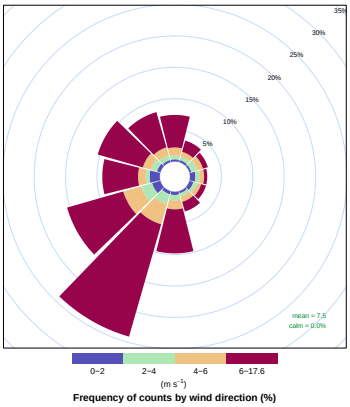
(d) August



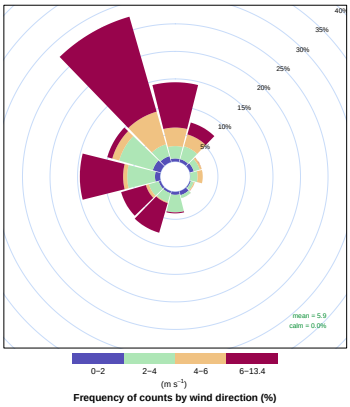
(e) September



(f) October



(g) November



(h) December

Figure 7.4: Summaries of monthly winds during 5 years of observations at NDBC buoy 45002.

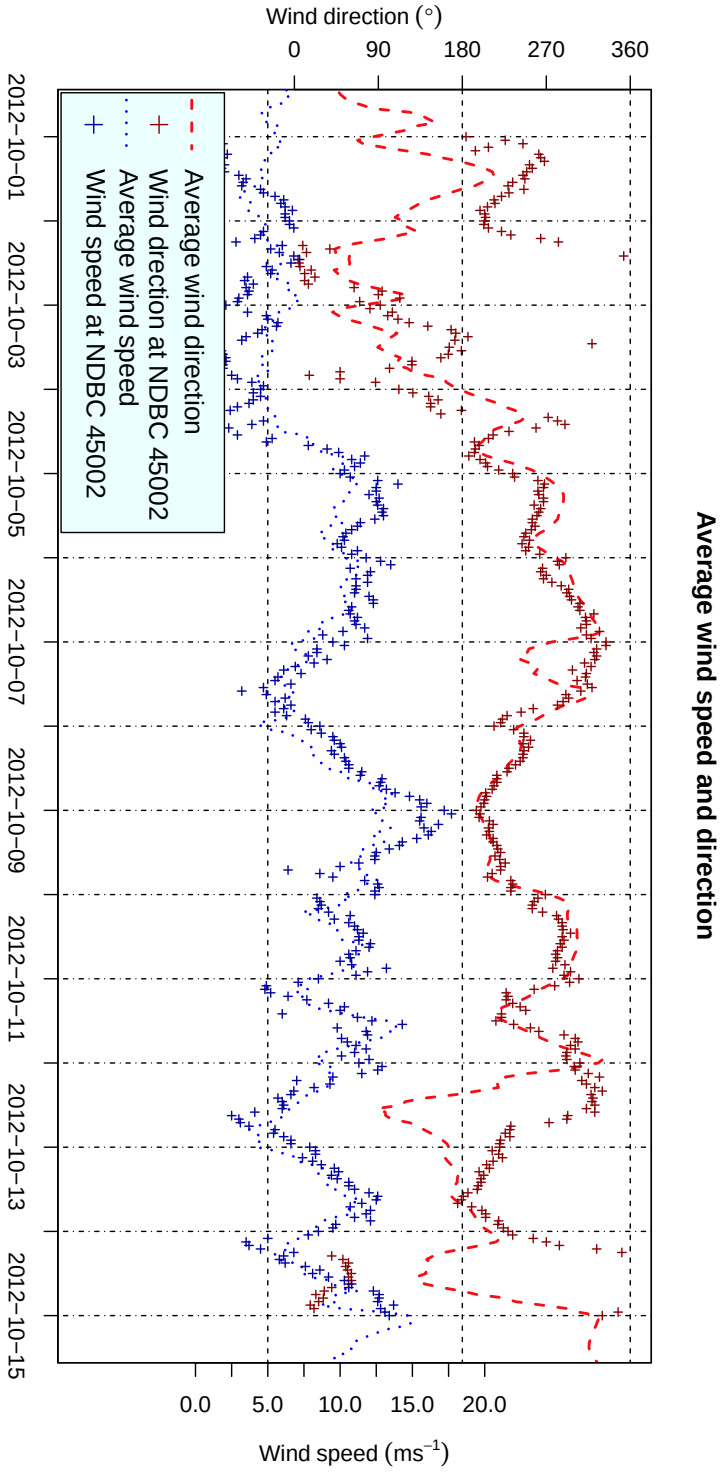
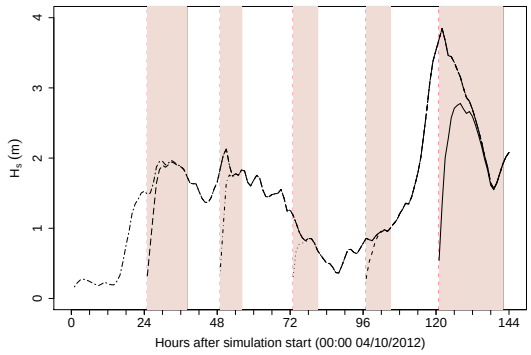
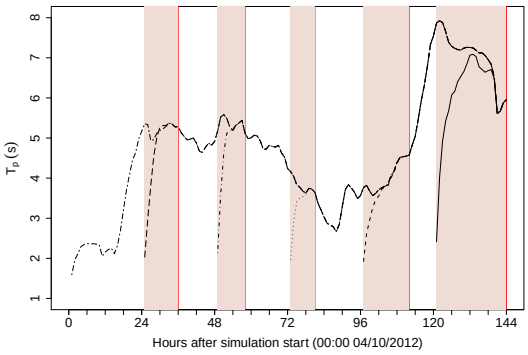


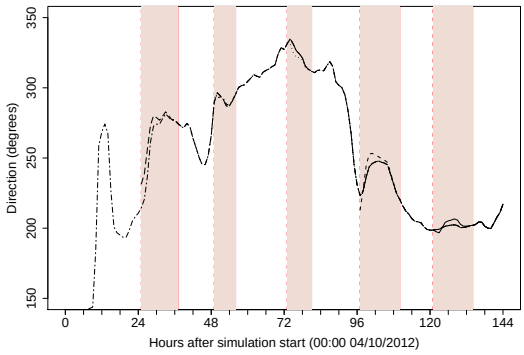
Figure 7.5: Wind conditions at NDBC 45002 during the first two weeks of October. "Average" winds denoted with the orange and blue dashed lines are an average of seven locations evenly spaced running approximately north to south along the axis of the lake. These give an indication of spatial variability compared with the observation at NDBC 45002.



(a) H_s



(b) T_p



(c) Mean wave direction.

Figure 7.6: Six runs of the same experiment where each is started 24 hours after the last. The period of convergence to the solution from the initial conditions in each case is identified by the orange bands. The difference in convergence period can clearly be seen.

of most interest. An example of this can be seen in the period of a few hours before and after 00:00 09/10/2012 where the wind speed is approximately 15ms^{-1} and the direction approaches 180°, which is broadly aligned with the longest fetch. In contrast, between the 05/10/2012 and 07/10/2012, although the wind speed remains fairly strong at approximately 10ms^{-1} the direction can be seen to be approximately 270°, blowing west to east.

In order to examine spin-up times, winds during the period of six days between 00:00 04/10/2012 and 00:00 09/10/2012 were used to force six simulations, each beginning 24 hours after the previous. This provides six different initial conditions each exhibiting different wave conditions. Output from the simulation in terms of H_s, T_p and mean wave direction at NDBC 45002 is shown in figure 7.6. In each panel, the start of each of the six simulations is indicated by the left side (dashed line) of each of the orange rectangles, which coincide with each 24 hour period. During each period, the dashed black lines converge fairly quickly to meet the solid

black line, which is the output from the initial 6-day experiment (i.e. the converged solution). The right edge of each rectangle shows when the simulation achieved convergence. The convergence criteria used in each case is based upon a moving average difference threshold taken over three hours. The thresholds are 0.01m, 0.1s and 1 degree for H_s, T_p and mean wave direction respectively. These thresholds are fairly onerous given the other sources of uncertainty but they serve to ensure any oscillation around the convergence is accounted for. We can see in panel (a) that in this case for $H_s \approx 2$m, the spin-up period is generally around 12 hours or less. However, the spin-up time is considerably longer where $H_s \approx 4$m taking nearly 24 hours to achieve convergence. This pattern is also seen in panels (b) and (c) for T_p and mean wave direction respectively. Note that the wave direction converges fairly rapidly in all cases, which is consistent with wind-sea dominated conditions. A general observation is that the initial conditions of each simulation always appear to be of very low energy. We can see in panels (a) and (b) that both H_s and T_p have to "climb" a long way to reach the converged solution. This is not investigated further here but it is possible that the fetch-limited JONSWAP spectrum imposed is limited by the short west to east fetch. In contrast, the initial wave direction tends to be well aligned to the solution.

Since wave conditions on Lake Michigan rarely exceed 4 metres, ensuring a period of at least 24 hours of simulation time has elapsed before experiments are conducted will eliminate any uncertainty due to initial conditions. This practice is adopted for the remaining experiments in this chapter. This is perhaps excessively onerous because typically simulations are initialised where the wave conditions are significantly smaller than 2 metres. If the computational cost of simulations was very high, a shorter spin-up time might be considered. Having addressed this issue, we now move on to the first experiment which focuses on the same period in October 2012.

7.6 Experiment 7.B: Southerly winds

The objective of this experiment is to investigate how parameter uncertainty affects waves on Lake Michigan generated by southerly winds. Specifically, for the period of time between 08/10/2012 and 10/10/2012 the wind maintains a fairly constant direction that might give rise to conditions similar to the fetch-limited case studied in chapter 5. Owing to the dynamics however there are expected to be differences since neither the direction nor the wind strength remains constant. Uncertainty and sensitivity analysis is conducted for simulation output at the location of NDBC 45002 during the peak conditions.

Observations of the wind from NDBC 45002 and NDBC 45007 during the four day period between 08/10/2012 and 11/10/2012 are shown in figure 7.7. During the period beginning 00:00 08/10 the wind profile is fairly constant in direction while wind speed is seen to increase steadily from 8ms^{-1} to 16ms^{-1}. The direction is approximately aligned with the lake resulting in a fetch spanning much of its 400km length. In order to progress with the uncertainty analysis, the GLERL winds were used to force an ensemble of simulations according to the experimental design described in section 7.4.1. An initial inspection of the simulation output data, together with wave observations at NDBC 45002 and 45007, is shown in figure 7.8. Simulated H_s and T_p are shown by the solid lines, together with observations (denoted with circles and crosses respectively) from NDBC 45002 in panel (a) and NDBC 45007 in panel (b). Output generated using the default input values is shown as dark (red and blue) lines. A sample of 20 points from the first design data set are also shown for comparison, to give an indication of the variability due to tuning uncertainty.

Firstly, we can see from the observations that the largest waves (approximately 4m) are observed at NDBC 45002, consistent with the (approximately) southerly wind. These coincide very closely with the peak wind speed. Looking at the data for NDBC 45002 in panel (a) first, the default parameterisation clearly underpredicts H_s and T_p during the energetic period. The spread in the sample of output data provides an indication of the possible variability in prediction, suggesting that certain choices of parameterisation could give much better performance. However, of the 20 samples shown, only the more "extreme" output appears to coincide with the observations. Moreover, although the spread shows substantial variability in terms of the magnitude of H_s there is a high level of temporal correlation between samples. This implies that variation in possible parameterisations tends only to lead to a change in overall magnitude rather than significantly modifying the temporal structure or dynamic response. We can see that the observed H_s profile at 00:00 09/10/2012 reaches a peak close to 5 meters which is maintained for approximately 12 hours. Every sample run however predicts a decaying H_s profile during this period. This is possibly indicative of a structural error in the simulator, or may be due to the omission of an important source of uncertainty in the analysis, which could include poorly specified forcing winds. Similar trends are seen in the data for T_p. Again, the temporal structure is well captured by the simulator but all of the output samples are underpredictions. It appears, at least from this small sample, that in spite of a fairly large variance on the distributions of input parameters the simulator is unable to closely match the observations.

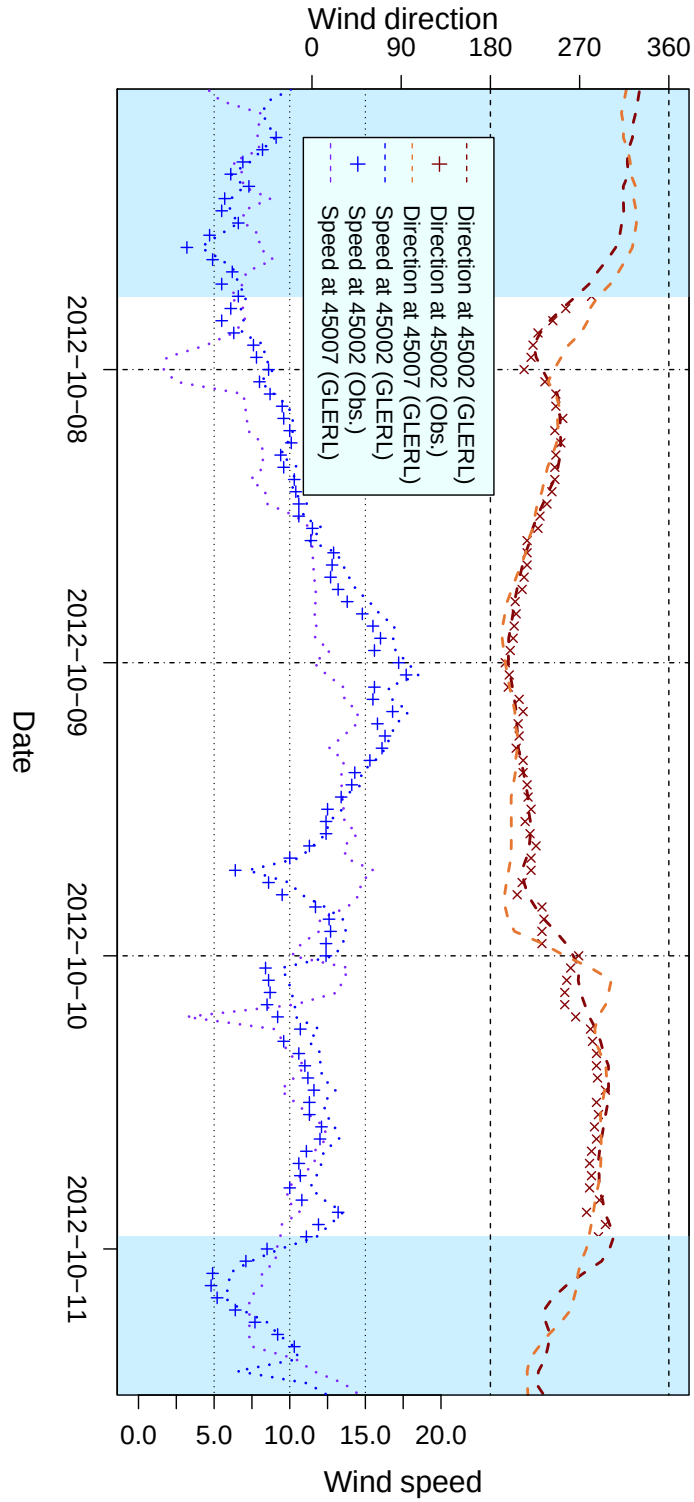
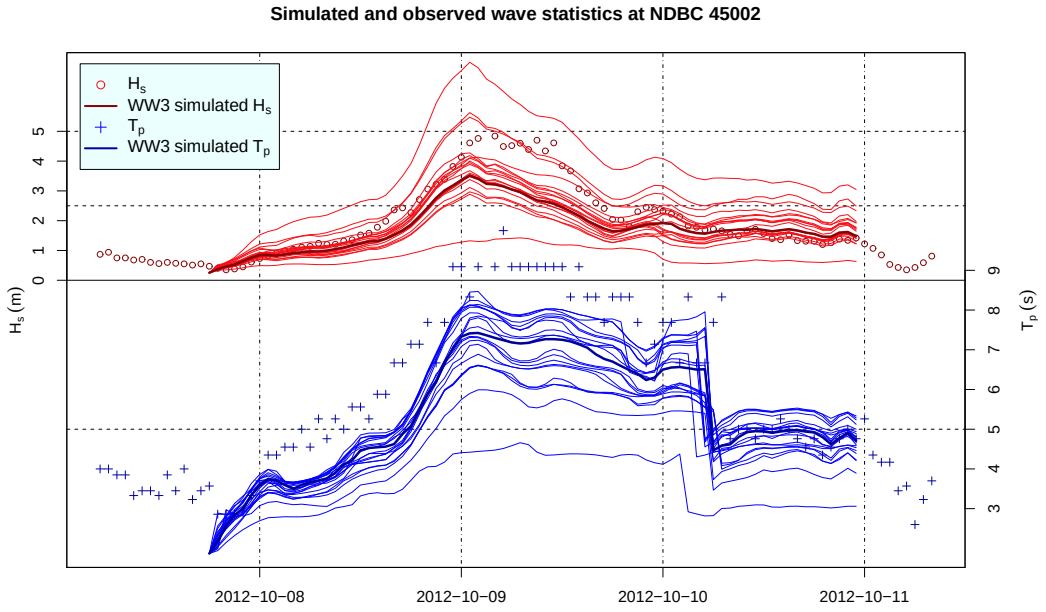
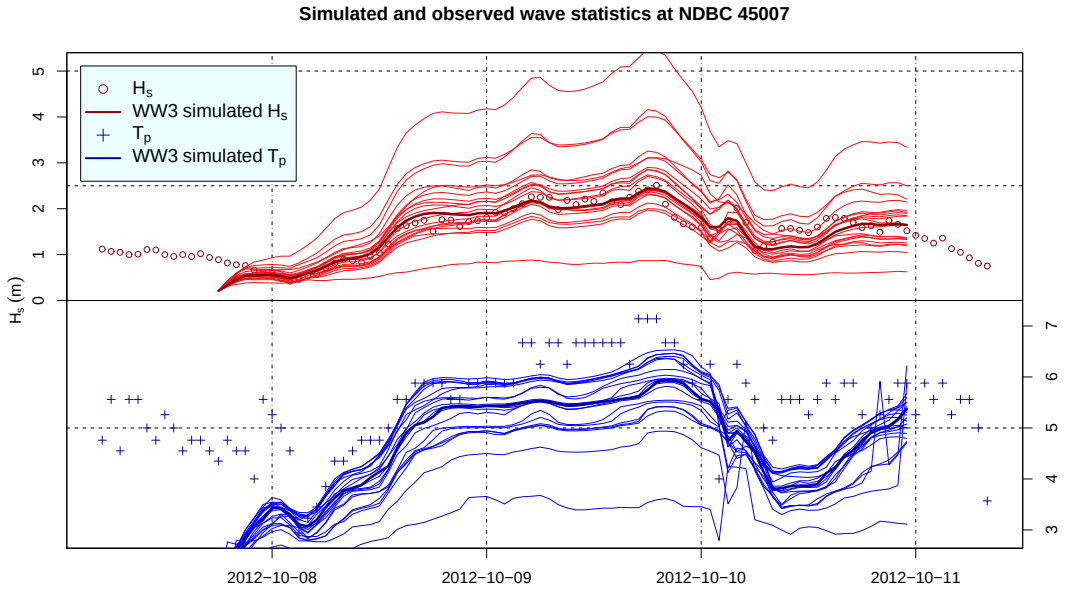


Figure 7.7: A localised average from the GLERL interpolated winds at NDBC 45002 and 45007 during October 2012. Note that the winds observed at NDBC 45002 are included for comparison. A slight difference is noticeable due to the localised averaging.



(a) Sample of simulated output at NDBC 45002, compared with observations.



(b) Sample of simulated output at NDBC 45007, compared with observations.

Figure 7.8: A comparison of simulated output from the default parameterisation with observations of waves at NDBC 45002 (panel a) and NDBC 45007 (panel b). A sample of 20 points from a design ensemble gives an indication of variability in the output.

Moving on to consideration of the data observed at NDBC 45007, shown in figure 7.8 (b) interestingly we see a different story. The default parameterisation gives a good hindcast for H_s, showing good agreement with the observations. Variation in the samples is comparable to that in panel (a) and the temporal structure of simulated output also appears to be in good agreement with observations. Simulation of T_p is improved but once again it is underpredicted, and even the most extreme sample output tends to fall below the observed values. The fact that hindcasts for different locations show markedly different skills again suggests the possibility of structural errors in the simulator which affect more energetic sea states more dramatically. Another cause of this is that the winds could be less accurately measured at NDBC 45002.

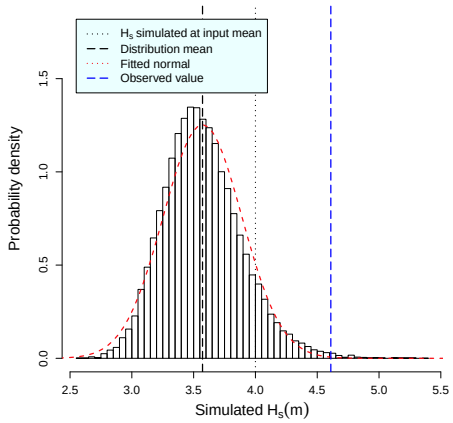
Before attempting to address questions relating to structural errors or biased forcing winds, we wish to look at how the simulator behaves in more detail. Specifically, in chapter 5 uncertainty and sensitivity analysis were performed for simulations of fetch-limited growth and sensitivity measures were determined for the input parameters at different values of fetch. Although the case here is not exactly equivalent to fetch-limited growth it does bear similarity, and we can perform similar analysis. Figure 7.7 shows that the wind direction remains fairly constant between 12:00 08/10/2012 and 18:00 09/10/2012, with a mean direction of 212 degrees and s.d. of 9 degrees. During this period the wind speed increases from approximately 10ms^{-1} to 18ms^{-1} before falling back to about 10ms^{-1}. Given a period of approximately 24 hours with a wind speed of 15ms^{-1}, conditions would be approximately fetch-limited up to about 460km[58]. However, towards the end of this period the wind speed decays towards 10ms^{-1} and conditions may therefore become more dynamic.

Simulation output at NDBC 45002 for two particular points in time are therefore chosen for analysis. The first is at the maximum wave height achieved, seen to be at 01:00 09/10/2012 in figure 7.8. The second is at 18:00 09/10/2012 where, as mentioned, the conditions are postulated to be closer to fetch-limited. Emulators for H_s at both time points were formulated with prior mean functions of $y \sim 1 + x_{[3,4,5,6,7,8]} + x^2_{[3,5,7]} + x^3_{[3,7]}$ and Matérn ($\nu = 3/2$) correlation functions.

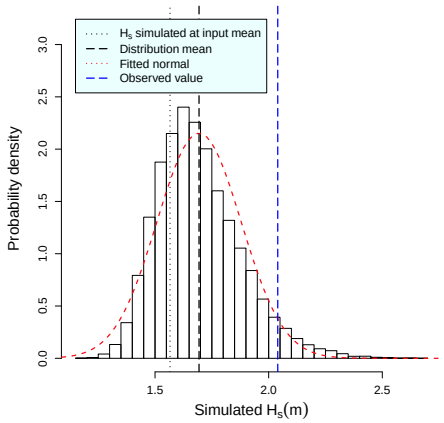
7.6.1 Analysis for H_s

Results of the uncertainty and sensitivity analysis for H_s are shown in figure 7.9. The results for the uncertainty analysis of simulated H_s, figure 7.9 (a), show a broadly linear response to the input uncertainty. Consistent with this, the main effects in

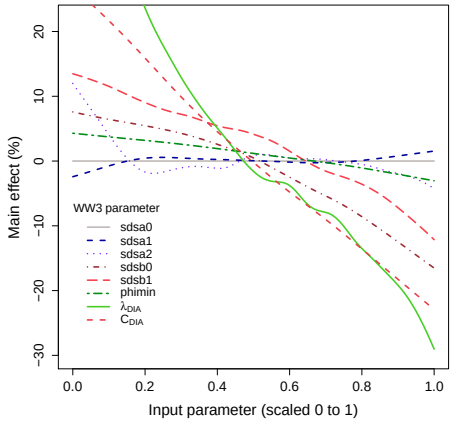
[58]The onset of the fully developed limit occurs at a non-dimensional fetch of approximately 2×10^4, so for a wind speed of 15ms^{-1}, this corresponds to a real fetch of approximately 460km.



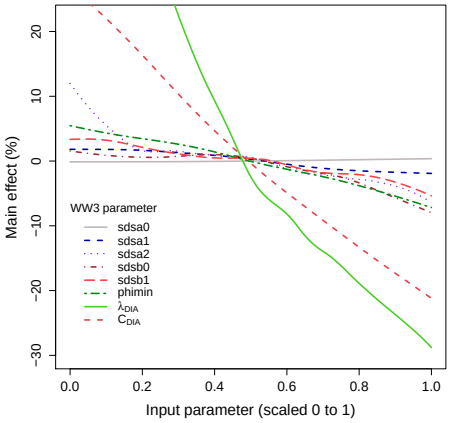
(a) Uncertainty analysis at 01:00 09/10



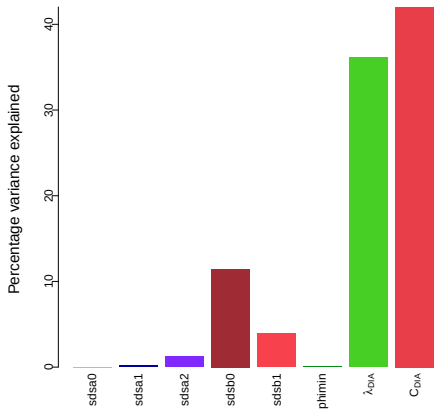
(b) Uncertainty analysis at 18:00 09/10



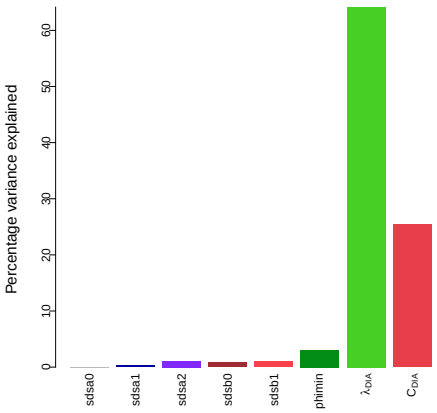
(c) Main effects for H_S at 01:00 09/10



(d) Main effects for H_S at 18:00 09/10



(e) Mean = 3.56m; S.D. = 0.33m



(f) Mean = 1.75m; S.D. = 0.21m

Figure 7.9: Uncertainty and sensitivity analysis for H_s simulated at 01:00 09/10/2012, and 18:00 09/10/2012.

panel (c) also exhibit a broadly linear response, the exceptions being *sdsa2* and λ_{DIA} which showed similar behaviour in many previous experiments. The most significant difference with the analysis at the two time points is that the variance is explained by different inputs parameters, seen in panels (e) and (f). Clearly the tuning of the DIA scheme is most influential in both cases. With reference to figure 5.14 we can draw some comparisons with the analysis for fetch-limited growth that were conducted in experiment 5.B. The output distributions are comparable, but the variances are a little lower in this case. However this is at least in part due to the lack of input wind variability. The analysis for the peak conditions (at 01:00 09/10), seen in figure 7.9 (b), shows a similar pattern of influence across the input parameters to that seen for H_s at short fetch, where the tuning for energy dissipation has an appreciable effect. The variance explained by the dissipation parameters is less than that seen at 52km but greater than that at 207km in experiment 5.B, so we might conclude that the situation is broadly similar to fetch-limited growth at (perhaps) 100km.[59] The analysis for H_s at the second time point reveals parameter influence very much like that at fully developed conditions, that is, comparable to figure 5.14 (f) for experiment 5.B (without input wind variability). This suggests that the sea state is well developed, which is actually what we would expect given that the wind speed is decaying from energetic conditions at this point. In a sense it may be in a state of "relaxation". Evidently in such states the tuning of the dissipation parameters is somewhat irrelevant. This seems reasonable in that dissipation often tends refer to "white-capping", which is prevalent as a sea state grows from strong winds. As the sea relaxes and wave energy is propagated from the local region energy dissipation is much reduced, and so in terms of the simulation, we would not expect to see much influence. We can conclude from this that nonlinear wave-wave interactions appear to have a dominant effect in both idealised "steady state" cases and dynamic conditions. Finally note that the total variance that remains unexplained by the main effects is less than 5.0% in both cases showing that interactions between input parameters is small.

Although we have learned about the relevant effects of the different tuning parameters and parameterisations, it should be born in mind that analysis here is with respect to H_s (and T_p in the next section), both of which are summary measures. As such we do not see how the uncertainty acts upon the form of the complete wave spectrum. That is, although a certain choice of DIA parameterisation might appear to give a better simulation of H_s or T_p in some situations, that does not necessarily mean the underlying frequency spectrum is accurate, or even realistic. A more

[59] This could be easily verified by an additional experiment.

detailed study would be required in order to explore this.

7.6.2 Analysis for T_p

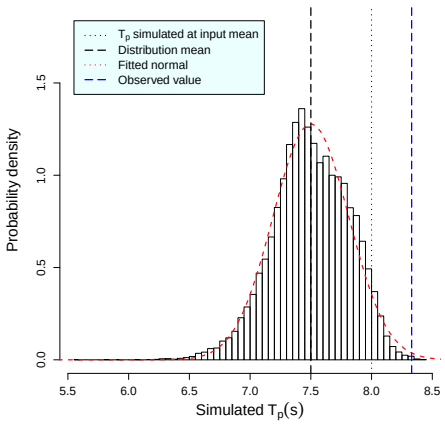
Results of the uncertainty and sensitivity analysis for T_p are shown in figure 7.10. Considerable nonlinearity is apparent in the uncertainty analysis, and the sensitivity analysis reveals that λ_{DIA} accounts for 75% of the variance at the maximum conditions, and 93% in the latter conditions. Panels (c) and (d) show that apart from λ_{DIA}, the other parameters lead to a main effect of no more than a 5%, even at the extremes of the parameter ranges. This compares very closely with the results of experiment 5.B where T_p was also seen to be influenced almost exclusively by λ_{DIA} and wind speed. The size of the variances in experiment 5.B are a little larger than in this case but this is at least partially explained by the lack of wind variability. In contrast with experiment 5.B for the analysis of T_p at short fetch, here there is little interaction between the input variables, calculated to be less than 5% in both cases.

This result is further evidence that T_p is less strongly linked to the dissipation parameterisation (and its tuning) than H_s. In practical terms this means that we have much less control over the adjustment of T_p, and an implication is that it is constrained (either intentionally or otherwise) within the structure of the simulator. Given that there is little means to adjust the performance, the general underestimation of T_p suggests that there are structural errors giving rise to the low bias. The next experiment considers a similar case but uses forcing winds in the opposite direction.

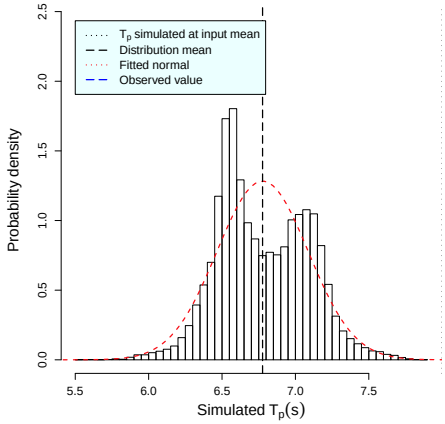
7.7 Experiment 7.C: Northerly winds

This experiment is somewhat complimentary to the previous in that the wind is blowing strongly, and aligned to the length of the lake. In this case however the wind is northerly and so the largest waves are located in the south. The wind profile is shown in figure 7.11. Note that the (apparent) erratic directional changes are actually an artifact of the averaging process, averaging values that cross the 360 to 0 degree boundary. The observational data, indicated by the crosses, show that the direction is sustained for the four day period with very little variation. The wind speed is seen to increase from approximately 5ms^{-1} to 17.5ms^{-1}.

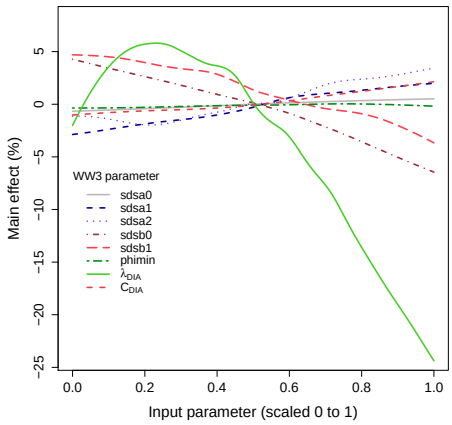
We are interested in seeing how the effect of uncertainty in the simulation changes as the wind speed increases, and whether this is comparable to both the previous experiment, and findings for fetch-limited growth in chapter 5. Figure 7.11 shows that



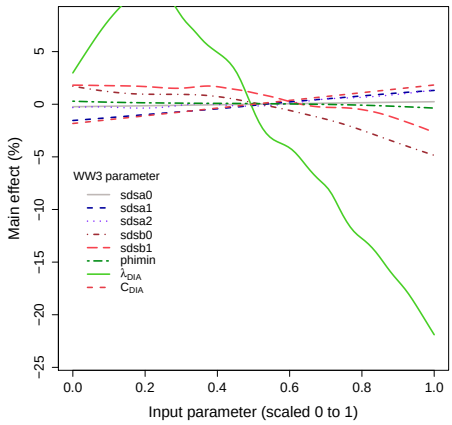
(a) Uncertainty analysis for T_p at 01:00 09/10



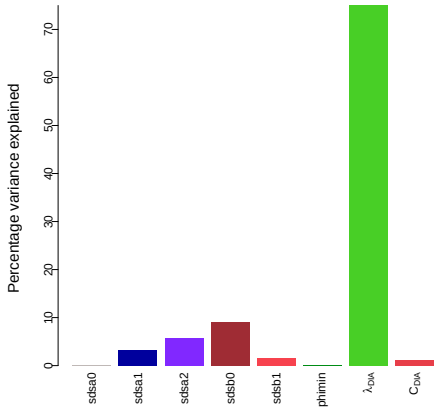
(b) Uncertainty analysis at 18:00 09/10 09/10



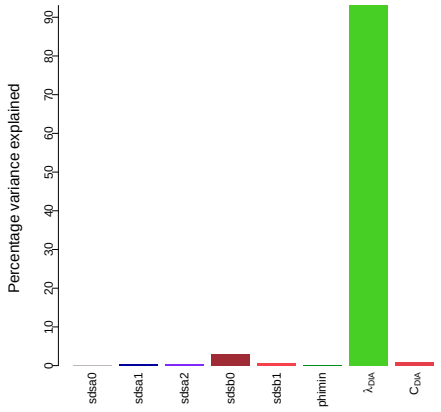
(c) Main effects for T_p at 01:00 09/10



(d) Main effects for T_p at 18:00 09/10



(e) Mean = 7.5s; S.D. = 0.31s



(f) Mean = 6.8s; S.D. = 0.33s

Figure 7.10: Uncertainty and sensitivity analysis for T_p simulated at 01:00 09/10/2012, and 18:00 09/10/2012.

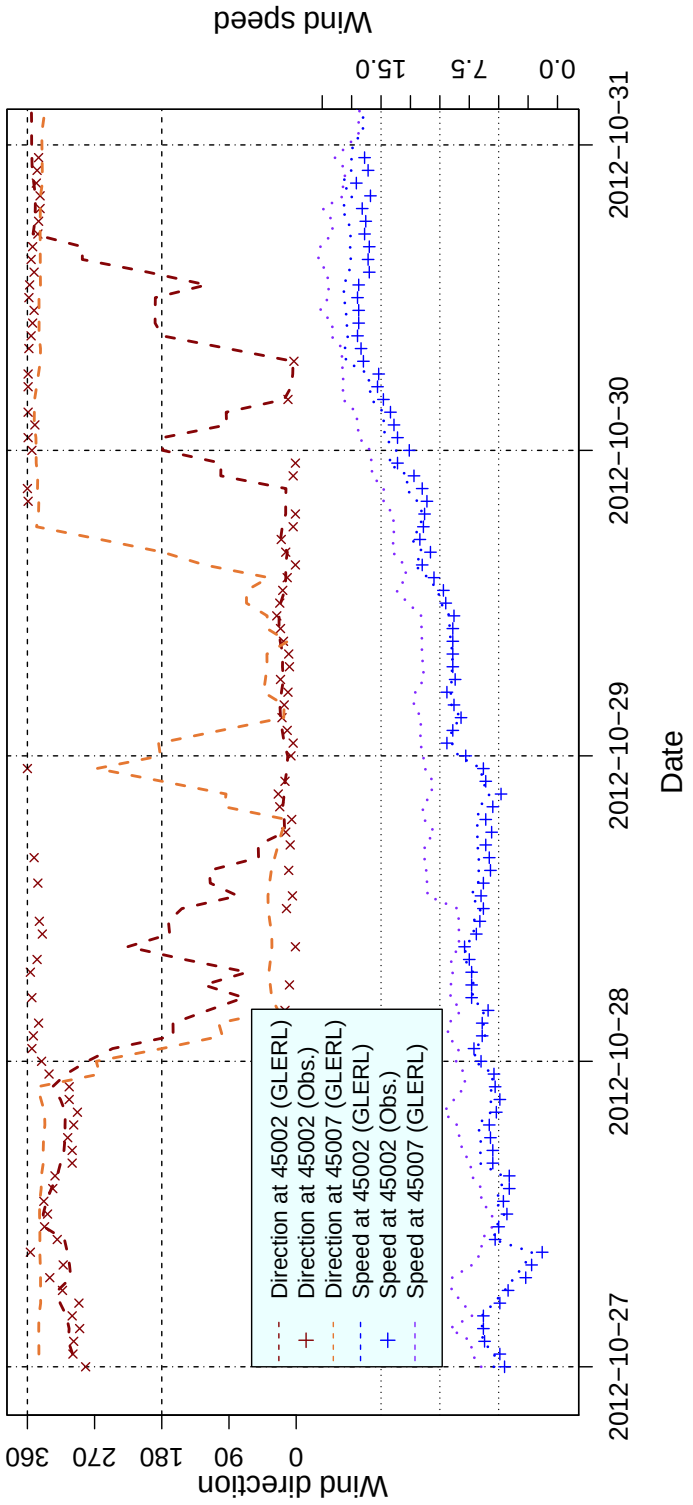


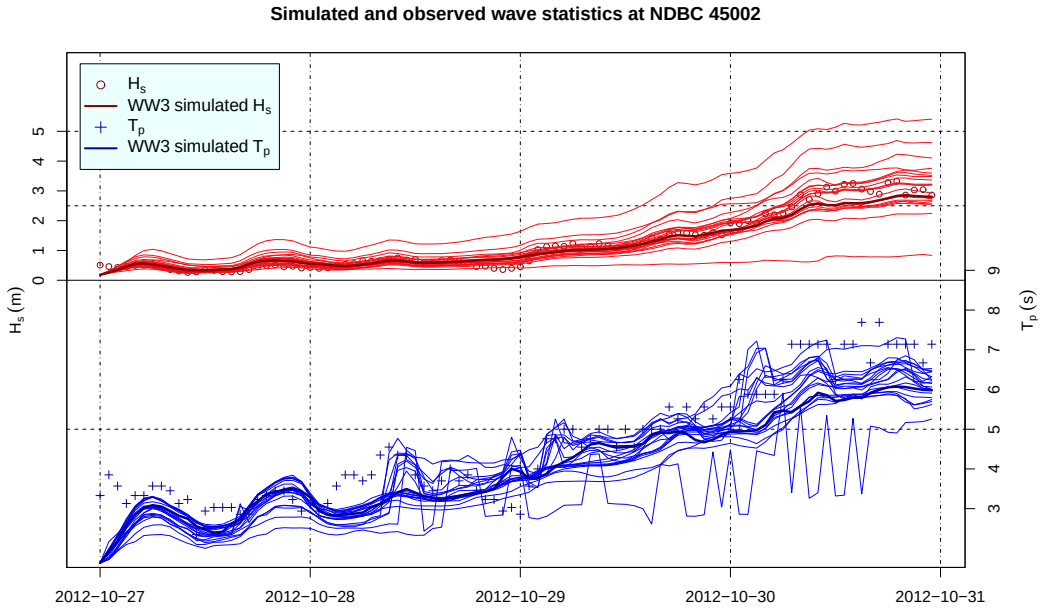
Figure 7.11: A localised average from the GLERL interpolated winds at NDBC 45002 and 45007 during October 2012. Note that the winds observed at NDBC 45002 are included (marked as crosses) for comparison. A slight difference is noticeable due to the localised averaging.

the wind conditions are very steady for the twelve hours following 00:00 29/10/2012 with a wind speed of approximately 10ms^{-1}. Very similar conditions are seen during the 12 hours preceding 00:00 31/10/2012, although the wind speed is approximately 17.5ms^{-1}. Given these conditions we expect the waves at NDBC 45007 to be approximately fetch-limited. Time points 12:00 29/10/2012 and 00:00 31/10/2012 (at NDBC 45007) are therefore chosen for analysis. These results can be compared with the previous experiment, and with experiment 5.B in order to identify commonality.

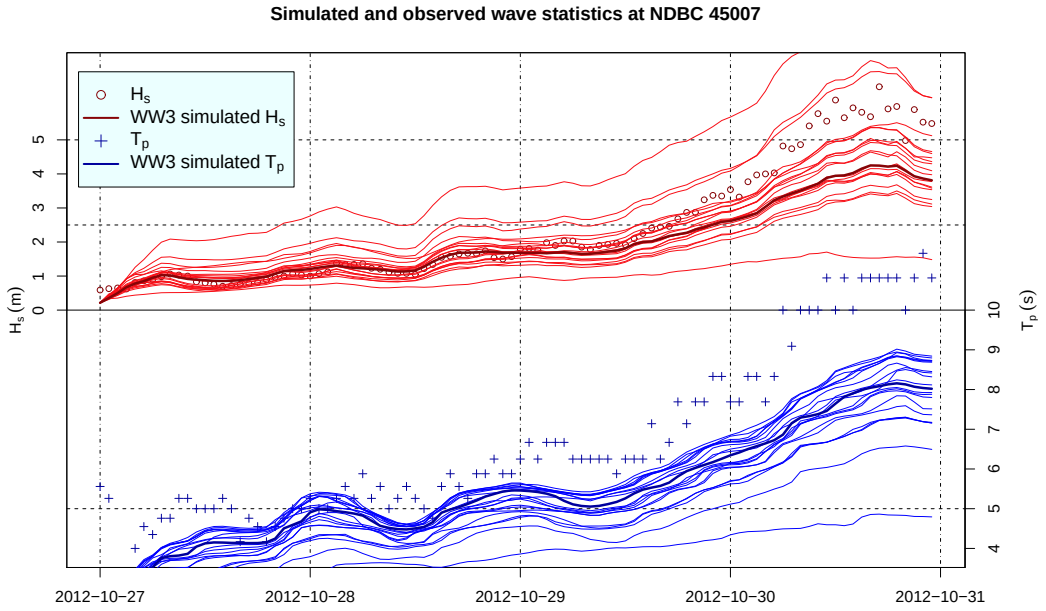
The experiment design is copied identically from experiment 7.B. Before performing that analysis, an initial inspection of some of the simulation output, seen in figure 7.12, reveals some important issues. Firstly, in similarity with experiment 7.B, we see that simulation of smaller waves at shorter fetch, measured at NDBC 45002 in this case, is far better than that for larger waves. Figure 7.12 (a) shows that the default simulation tracks the observations quite closely until about the final 12 hours for H_s and 24 hours for T_p, when it begins to under-predict both measures. However, panel (b) shows that the onset of under-prediction is much earlier for NDBC 45007, and during the peak conditions is quite drastic for both H_s and T_p. We appear to be seeing similar results to experiment 7.B, but with the wind reversed so that the energetic waves at NDBC 45007 are now under-predicted. Secondly, the spread of simulations in the design set seems to suggest that for H_s the observed values fall within the range of output induced by parameter variability. In similarity with experiment 7.B this is not the case for T_p, which is again systematically underpredicted. Thirdly, some strange behaviour is apparent in the simulation of T_p for NDBC 45002. It appears that some combination of parameter values caused spurious simulations of T_p although oddly this only seems to apply to prediction at NDBC 45002. Recall however that discontinuity in output for T_p was seen in experiment 5.B as a result of high values of λ_{DIA} and what we see here maybe the same issue. Note the design range in this case is $0.1 < \lambda_{DIA} \leq 0.4$. The spurious data need not be a concern at this stage provided an emulator can be fitted to the data, and in fact the design point associated with this run lies on the edge of the design. Analysis of the simulation data for H_s and T_p is undertaken in the following sections.

7.7.1 Analysis for H_s

Emulators were formulated in the same way as experiment 7.B and used to perform uncertainty and sensitivity analysis for H_s, the results of which are shown in figure 7.9. Here again we see the same kind of response to input parameter uncertainty that we saw in the previous experiment and the tuning of the DIA scheme is clearly

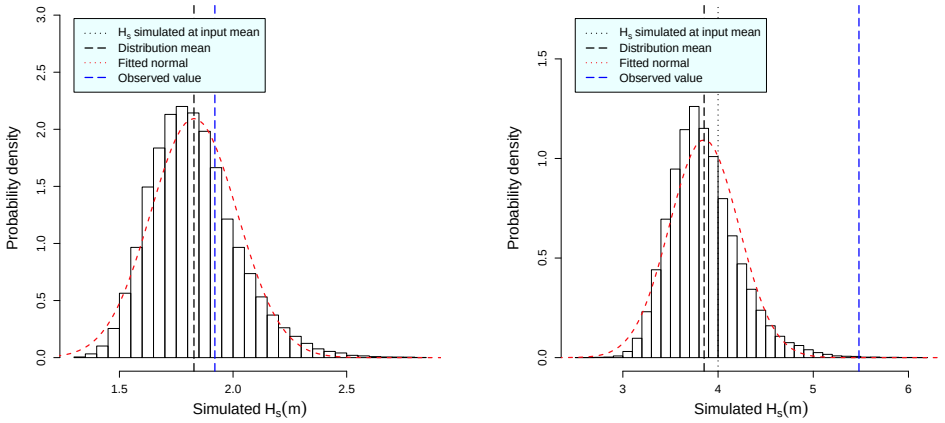


(a) Sample of simulated output at NDBC 45002, compared with observations.



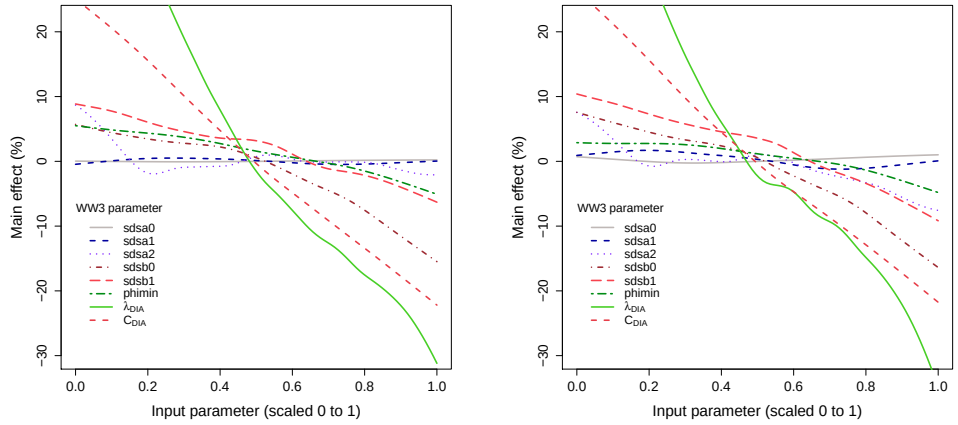
(b) Sample of simulated output at NDBC 45007, compared with observations.

Figure 7.12: A comparison of simulated output from the default parameterisation with observations of waves at NDBC 45002 (panel a) and NDBC 45007 (panel b). A sample of 20 points from a design ensemble gives an indication of variability in the output.



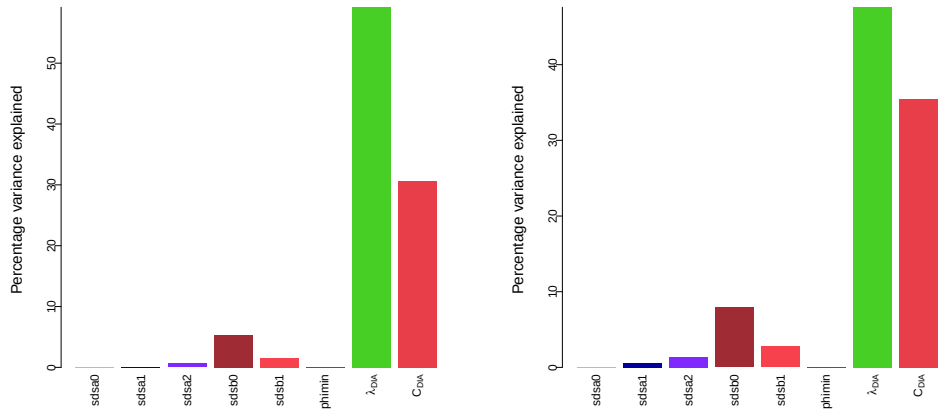
(a) Uncertainty analysis at 12:00 29/10/2012

(b) Uncertainty analysis at 23:00 30/10/2012



(c) Main effects for H_S at 12:00 29/10/2012

(d) Main effects for H_S at 23:00 30/10/2012



(e) Mean = 1.83m; S.D. = 0.20m

(f) Mean = 3.87m; S.D. = 0.38m

Figure 7.13: Uncertainty and sensitivity analysis for H_s

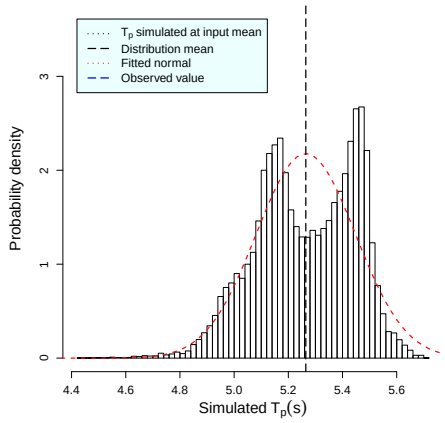
of most influence. There is remarkably little difference between the analysis at the two different time points, which suggests that the physics was similar in both cases. At the latter time point, figure 7.13 (f) seems to indicate that the sea state is growing more than the earlier time point, since it is consistent with the response of conditions at shorter fetch. This is characterised by more influence of the dissipation input parameters and C_{DIA}.

7.7.2 Analysis for T_p

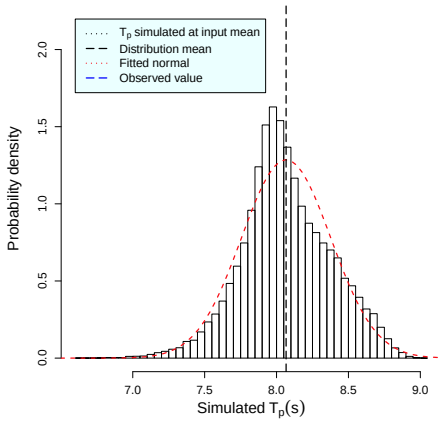
Results of the uncertainty and sensitivity analysis for T_p are shown in figure 7.14. Again it is immediately obvious that the results are similar to those found in the previous experiment. λ_{DIA} accounts for most of the variance in both cases (92% and 87% respectively). The same nonlinear response to its variation is also exhibited in the output uncertainty distribution. Importantly the total variance induced in T_p, as a percentage of the mean value in both cases, is 3.5% and 3.6% respectively, which compares to 11.0% and 9.8% in the case of H_s. This is also consistent with the previous experiment and further highlights that T_p is considerably more constrained than H_s.

7.8 Experiment 7.D: Changing wind direction

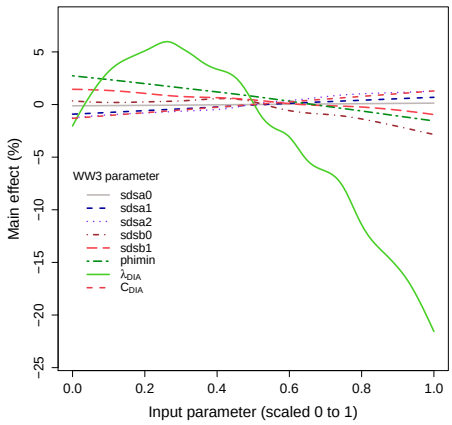
The previous two experiments considered situations that created wave conditions that were broadly fetch-limited. The wind was therefore fairly constant in time. In this experiment we consider a different kind of wind behaviour, in order to analyse the uncertainty induced in simulated wave direction. This allows for some comparison with the findings made in chapter 6 where response to a turning wind was considered. There are however some differences between the two situations. Experiment 6.B investigated the response of the wave simulator to an abrupt change in wind direction (through 90°). Over Lake Michigan such abrupt changes are typically transitory and occur during periods of high variability, thus making them a more complex case to study. The forcing wind in this experiment has been chosen on the basis of exhibiting a consistent change in direction over a short time scale. The two cases also differ in terms of geographic scale, where experiment 6.B was conducted on a 1000km × 1000km square grid with 50 cells each of length 20km, as compared with grid cells with a length of approximately 4km. Nonetheless, the wave conditions in both cases will be dominated by the local wind conditions and so the larger grid size will have a limited impact. A time series showing a summary of the forcing winds in this case is shown in figure 7.15. The winds, that occurred in September



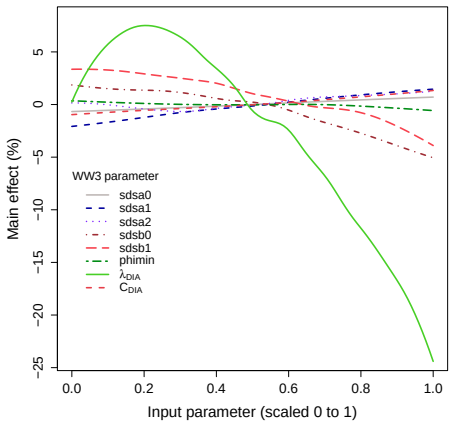
(a) Uncertainty analysis for T_p at 01:00 09/10



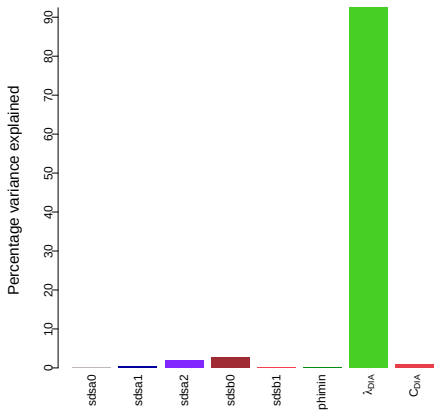
(b) Uncertainty analysis at 18:00 09/10



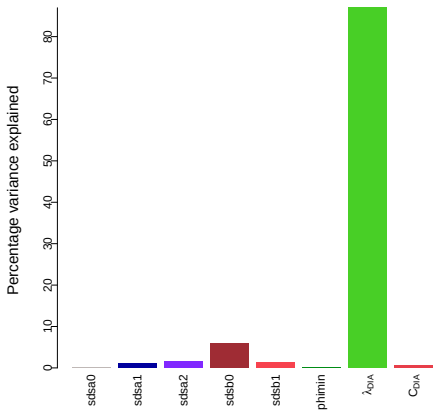
(c) Main effects for T_p at 01:00 09/10



(d) Main effects for T_p at 01:00 09/10



(e) Mean = 5.3s; S.D. = 0.19s



(f) Mean = 8.1s; S.D. = 0.29s

Figure 7.14: Uncertainty and sensitivity analysis for T_p

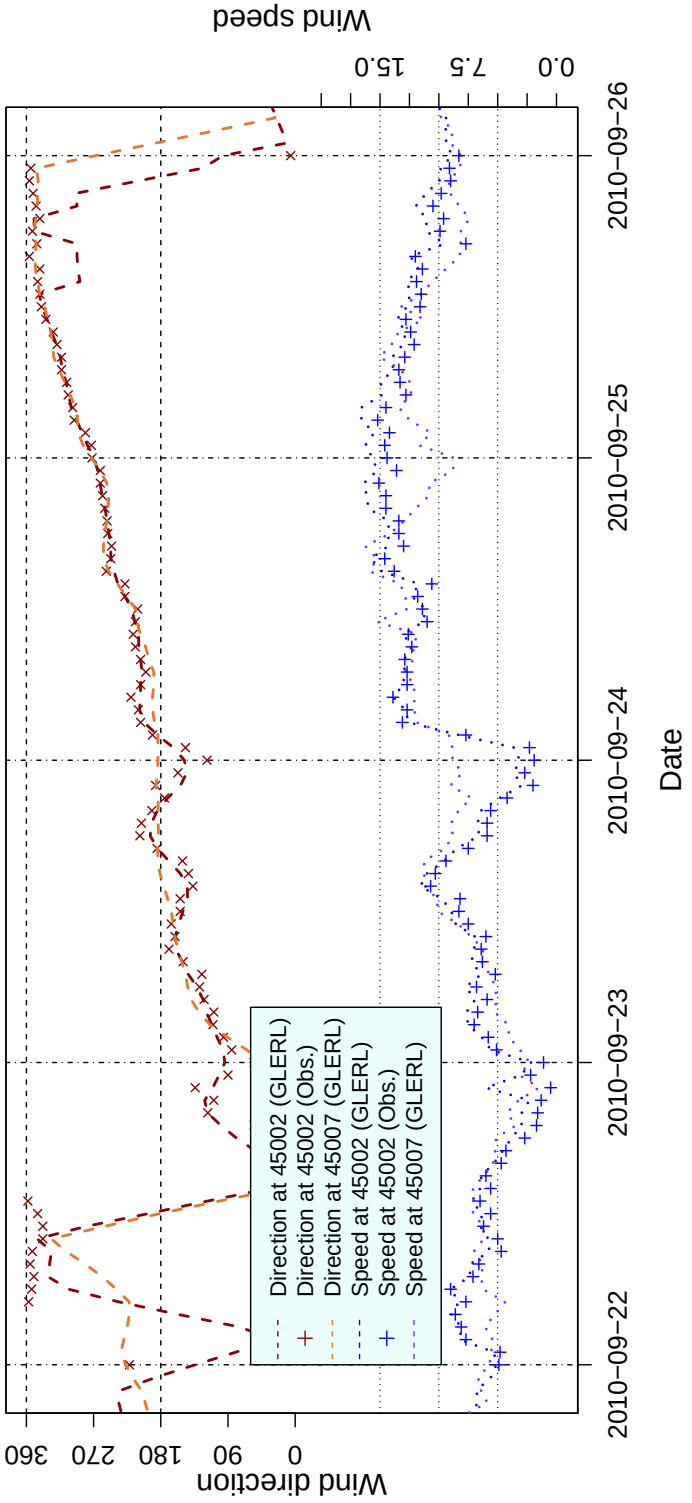


Figure 7.15: A localised average from the GLERL interpolated winds at NDBC 45002 and 45007 during September 2010. Note that the winds observed at NDBC 45002 are included (marked as crosses) for comparison. A slight difference is noticeable due to the localised averaging.

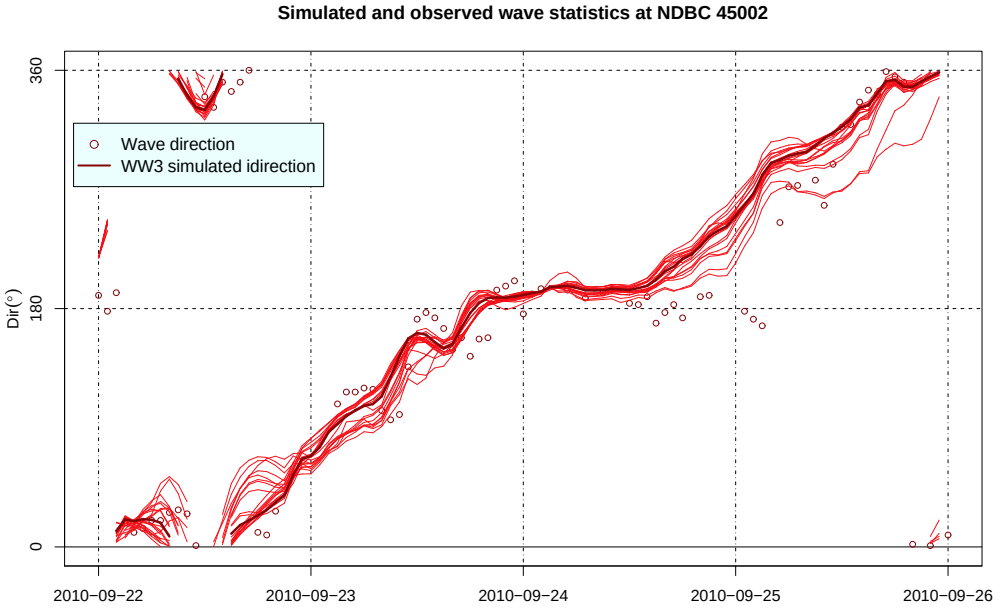


Figure 7.16: A comparison of simulated output from the default parameterisation with observations of waves at NDBC 45002. A sample of 20 points from a design ensemble gives an indication of variability in the output.

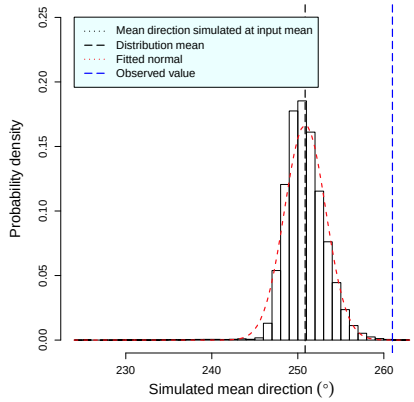
2010, exhibit a steady directional change that swings through 270 degrees from easterly to northerly over a period of three days. Comparing these conditions with the analysis conducted in chapter 6, the directional transition here is gradual as opposed to sudden. Furthermore the wind strength shows some variability, although tends to increase during the period. We may therefore not expect to see quite the same response to parameter uncertainty. The experiment was run according to the same design as experiment 7.B and 7.C. Simulated mean wave direction for the period at NDBC 45002, produced by the default parameterisation, together with output from a sample of the design points is shown in figure 7.16. The default output appears to reproduce the observed data with reasonable accuracy. The spread in the output sampled from the design is generally small and reduces to almost zero at a number of points. Variability appears to be positively correlated to the rate of directional change. It can be seen that during the 20 hours beginning approximately 18:00 23/09/2010, when the direction remains constant, the variability in the design sample is very small. This is in contrast to the subsequent 36 hours when the direction is changing, and the variability is increased. Intuitively this seems reasonable since the waves follow the direction of the wind. In the absence of a changing wind direction, there is no reason why a perturbation in the wave physics should give rise to a change in wave direction. However, clearly perturbed physics can result in a

slower or faster rate of change of direction, as indicated by the increased variability. Departures from the observational data are evident, particularly during the latter part of the simulation. Consistent with the findings of the previous experiments, the variability in the design sample does not suggest that uncertainty in parameter tuning can account for the discrepancy between the simulation and observations.

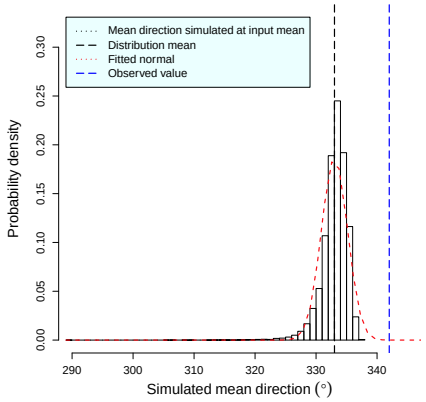
On the basis that the time points that exhibit the highest variance are the most interesting, and important (given that physical processes are most active), uncertainty and sensitivity analysis is performed for the two time points that exhibit the highest variance across the six design data sets. These are 23:00 24/09/2010 and 15:00 25/09/2010 (with s.d.'s of 13.4 and 14.8 degrees respectively). Results are shown in figure 7.17. Both cases show similarity, the most striking point being the response of wave direction with respect to λ_{DIA}. As we have seen previously with wave height and period in a number of experiments, it is highly nonlinear, in contrast to the other uncertain input parameters. In this case the resulting uncertainty distributions in panels (a) and (b) have some interesting properties. They are both skewed, the first in (a) towards the low end, the second in (b) towards the high end, but more importantly both have extremely long tails on the low end, with a minimum value in the distribution (for the given MC sample) of 290°. Inspection of the main effects in panels (c) and (d) show why this comes about. At a scaled value of 0.7 (corresponding to a value of λ_{DIA} of approximately 0.31) the response begins to fall very rapidly in both cases. The input probability distribution for λ_{DIA} is $N(0.250, 0.03125^2)$ which results in $P(\lambda = 0.31) \approx 0.05$. So values of λ_{DIA} greater than 0.31 are sampled with a low probability but owing to the rapidly changing response it is clear that the corresponding values of wave direction will be many standard deviations from the output distribution mean. Thus the resulting output distribution has a very long tail. Note that the value of 0.31 for λ_{DIA} corresponds closely with the discontinuous behaviour in response found in experiment 5.B.

In both cases λ_{DIA} contributes approximately 50% to the output variance. Response to the other input parameters is fairly linear but together they contribute 44% and 40% of the remainder respectively, leaving 6% and 10% unexplained by the main effects. This shows that there is some interaction between the inputs but it is fairly small. Moreover, the total variances of 6.7 (s.d. = 2.6°) and 5.6 (s.d. = 2.4°) are also small, given that the observed values lie more than 4 s.d. from the mean. This suggests that other sources of uncertainty are required to explain the large difference, and that model structural error may be present.

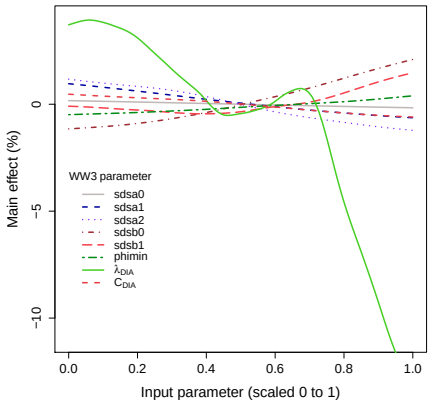
This brief investigation into the directional response of the wave simulator in a realistic case has highlighted at least two important points. Firstly, at the two



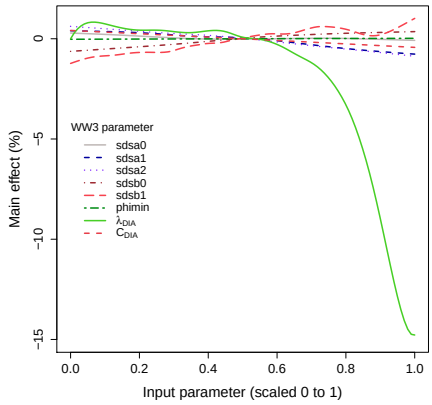
(a) Uncertainty analysis for mean wave direction at 23:00 23/09/2010



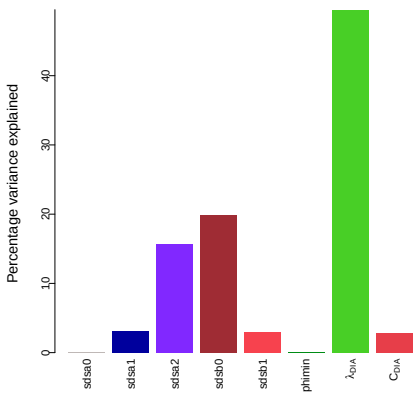
(b) Uncertainty analysis for mean wave direction at 15:00 25/09/2010



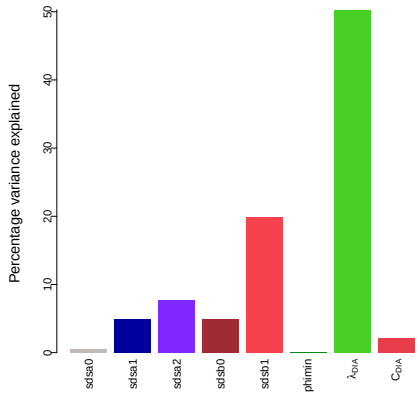
(c) Main effects for wave direction at 23:00 23/09/2010



(d) Main effects for wave direction at 15:00 25/09/2010



(e) Mean = 251 degrees; S.D. = 2.6 degrees



(f) Mean = 332 degrees; S.D. = 2.4 degrees

Figure 7.17: Uncertainty and sensitivity analysis for mean wave direction simulated at 23:00 23/09/2010, and 15:00 25/09/2010.

different time points during the wind transition, that showed the highest variation in the simulation, the total variance in the results of an uncertainty analysis was found to be small (s.d. $\approx 2.5°$), and could not account for the difference between the simulation output and the observed values. Possible explanations for this include unreliable observations, inaccurate forcing winds or structural error in the model. In the absence of a rigorous estimate for either the observations or the wind uncertainty, determination of possible structural error is difficult. However, given that discrepancy between simulation output and observation seen in experiments 7.B and 7.C were substantial only for the most energetic conditions, we might conclude that any uncertainty in the forcing winds, or the observations, was highly correlated with wind speed. This is no doubt an important area of further study. Secondly, we have seen that perturbation of the tuning parameters associated with energy dissipation tends to induce an approximately linear response in contrast with the nonlinear wave-wave interactions, specifically λ_{DIA}. The response due to λ_{DIA} seen in experiment 6.B was also highly nonlinear but of a different form. Nonetheless, in this experiment like many others, the simulator response tends to be dominated by uncertainty about λ_{DIA}.

7.9 Experiment 7.E: Parameter calibration for high winds

Noting the poor performance of WW3 with respect to simulating the more energetic wave conditions on Lake Michigan, this chapter is concluded with an example of a possible means of better calibrating the simulator. In order to do this we need a means of evaluating the performance of the simulator in order to identify the "best" calibration. A simple approach could involve formulating a cost function based on the mean squared difference between simulation output and observations, and this is followed here, although with the following considerations. Such an approach is statistically naive and does not account for observational error or uncertainty in the model discrepancy. That is, in contrast to the analysis conducted so far, beliefs about the best (or most informative) input parameter values are not incorporated and the resulting model discrepancy term is not statistically modelled. In the proposed approach the input parameter space is constrained by the design space of the emulator, but otherwise no preference for specific parameter values is expressed. The resulting ("optimal") choice of input parameters is therefore dictated by the choice of observational data (with no consideration for observational error) and could potentially lead to unphysical choices of parameterisations and over-fitting.

There is a growing literature on more comprehensive statistical approaches to this kind of problem. The method of "history-matching" (see e.g. Craig et al., 1996 and

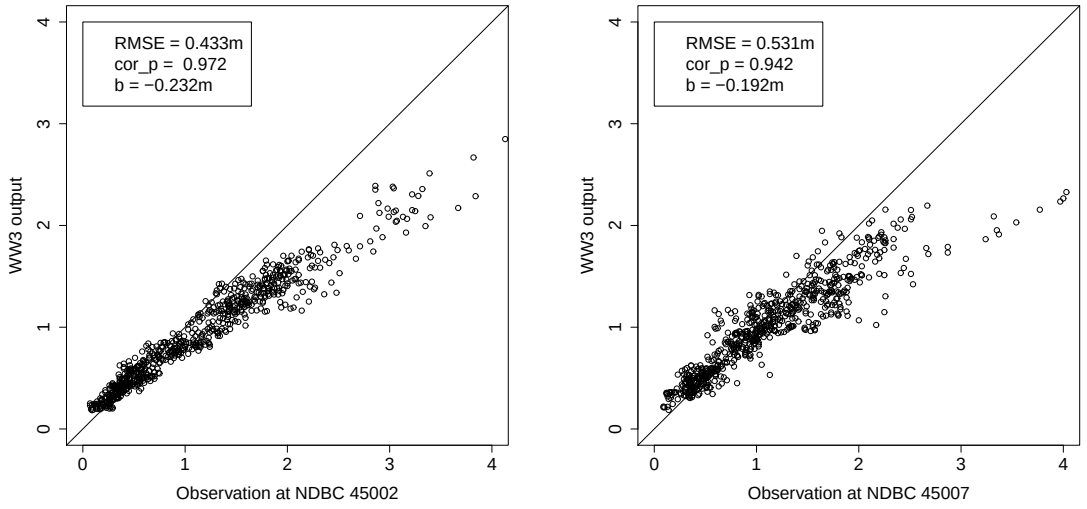
Williamson et al., 2013) involves consideration of a broad input parameter space and proceeds to "rule out" unphysical regions of that space based upon comparison with observational data. This approach incorporates uncertainty information but is also efficient, and helps avoid wasted effort associated with the modelling of regions of the input space that result in complex, nonlinear and unphysical output[60]. History-matching can be used in conjunction with a probabilistic calibration methodology. Such a methodology is described by Kennedy and O'Hagan (2001), which is briefly summarised in section 3.4.5. They formulate a model discrepancy term but rather than it being a simple measure of error, it is a complete probabilistic description of how the simulator output relates to observations (as a function of the uncertain inputs). It is in fact modelled as a GP much like the simulator response is emulated in this thesis. Trucano et al. (2006) commend the approach of Kennedy and O'Hagan (2001) but point out that the methods required for analysing the resulting posterior probability distributions are complicated. Noting the apparent lack of literature describing the calibration of a wave simulator, probabilistic or otherwise, a simpler approach is undertaken here (acknowledging its shortfalls). However, methods such as those described above are being increasingly employed for complex simulators of various kinds (see Williamson et al., 2013 and references therein) and it would be a very natural next step to bring them to bear on a wave simulator. This is mentioned briefly in section 9.1.

The initial approach then is to formulate a simple cost function based on predictions and observations of H_s at both buoys 45002 and 45007. The choice of ignoring T_p, or any more detailed information derived from the spectrum, is two fold. Firstly, it is substantially simpler to consider H_s alone rather than to derive more complex cost functions, and secondly we have seen that the simulator consistently under-predicts T_p even using "extreme" choices of input parameter values. Any calibration process based upon T_p would preferentially select choices of parameter values that necessarily result in "extreme" parameterisations. We therefore choose to avoid this issue by using data associated only with H_s on the basis that simpler cost functions can be formulated and useful progress can be made. This does not preclude improved simulations of T_p resulting from the calibration although any improvement would be coincidental[61].

Hourly observations of H_s are available at both NDBC 45002 and NDBC 45007 for the entire month of October 2012. A comparison of these with output from

[60] Recall that here we have seen larger values of λ_{DIA} give rise to apparently unphysical output.

[61] Note however that H_s and T_p tend to be highly correlated and it is not unreasonable to expect that a new parameterisaton identified through a calibration exercise based solely on H_s will also improve simulation of T_p.



(a) **(b)**

Figure 7.18: Simulated H_s compared with observations at NDBC buoys 45002 and 45007 for the entire month of October 2012.

a simulation with default tuning parameter values, in the form of a scatter plot, is shown in figure 7.18. The underprediction of H_s at medium to high values is immediately clear. Three commonly used statistics for quantifying the performance are the root mean square error ($RMSE$), bias (b, defined here as the difference between the mean of each data set) and Pearson correlation (cor_P).

Ideally we would wish to calibrate with respect to all of the available data, which in this case would be the entire time series. However, the emulators used so far have been univariate in their prediction, and so to utilise such emulators in the calibration process we would need an emulator for each time point. Other alternatives are also possible. For example, since there is clearly a high degree of autocorrelation in the data a more efficient representation for the time series could be found and the resulting data set used as training data. In the example here only a limited number of points are used with a view to formulating an emulator for each point. Eight different time points are specified (identified by large crosses in figure 7.19) at regular intervals throughout the month. The calibration points include a range of observations, that exhibit both positive and negative prediction errors. A total of sixteen calibration points are used, eight associated with each buoy. Calibration is performed by minimising the sum of the squared distances at each of these points. The general trend of the simulation is to underestimate H_s and so at some points, notably the third, forth and fifth, the deviation is very large. The resulting calibration will represent a trade-off between fitting the simulator output

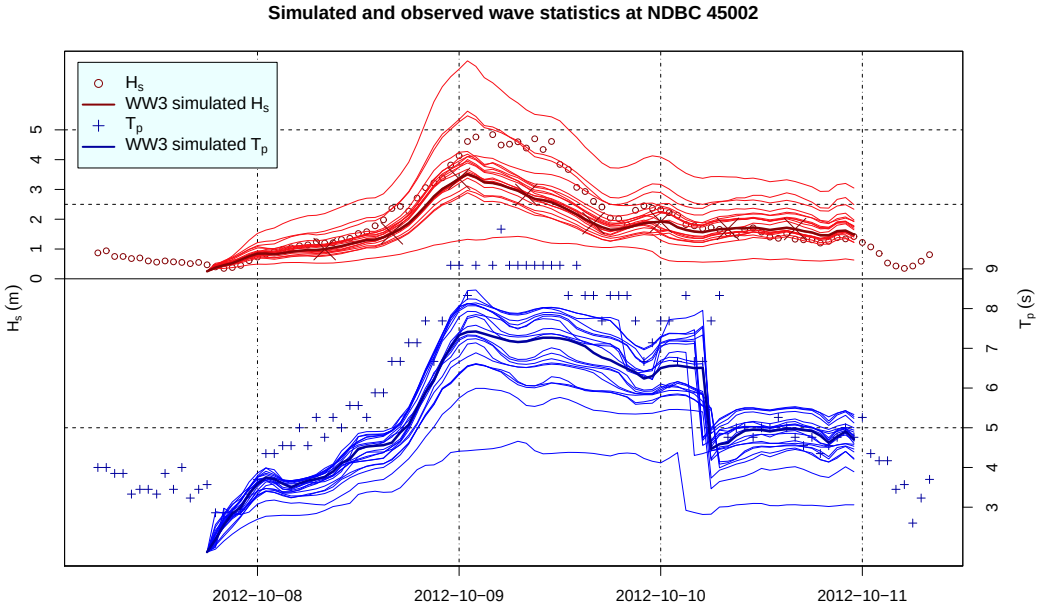
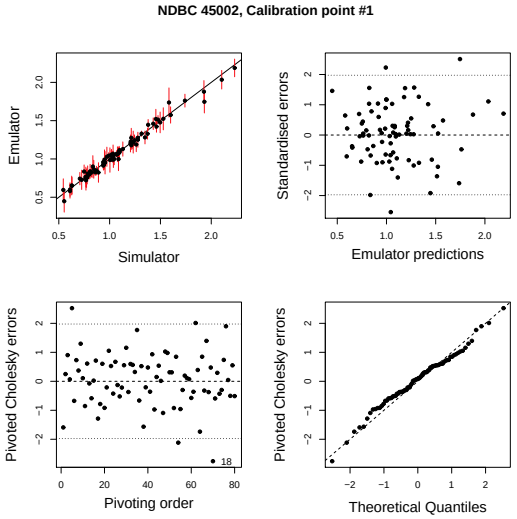


Figure 7.19: As figure 7.8. Large (red) crosses show the calibration points used to formulate the cost function.

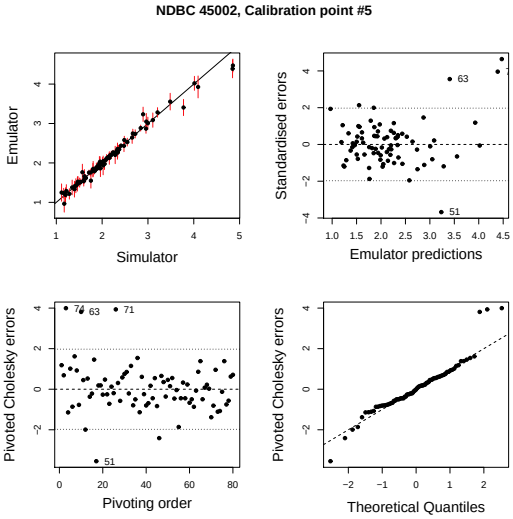
to both the lower values of H_s (which are simulated quite well by default) and the more extreme points (which tend to be significantly underestimated).

In practice this means that a total of sixteen emulators are required, one for each calibration point. Making use of a common emulator formulation, using a mean function of the form $y \sim x_{3,4,5,6,7,8} + x^2_{3,5,7} + x^3_{3,7}$ and a Matérn ($\nu = 3/2$) correlation function, it is straightforward and fairly quick to construct the emulators. When using larger numbers of emulators in this way (and some researchers have advocated using hundreds or even thousands) there is a risk that some will be fitted quite poorly, owing to the degree of automation required. With fewer emulators, and where a consistent formulation is adopted, it is straightforward to inspect the diagnostics and gain more confidence in their performance. A sample of diagnostics is shown in figure 7.20. Some of the validation tests show outlying points but on the whole the emulators appear robust. Having successfully formulated the emulators we proceed by using the emulator posterior mean functions to formulate a cost function. The form of the cost function initially chosen is based upon a simple sum of the squared differences between the simulator and observations, as follows,

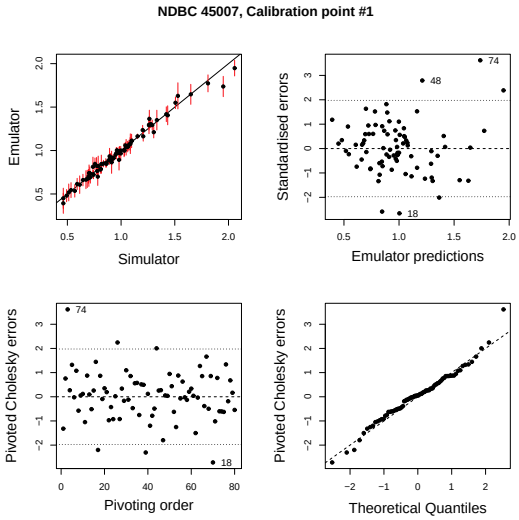
$$\mathcal{C} = \sum_{i=1}^{\mathcal{P}} (H_{s(Sim.45002)_i} - H_{s(Obs.45002)_i})^2 + \alpha \sum_{i=1}^{\mathcal{P}} (H_{s(Sim.45007)_i} - H_{s(Obs.45007)_i})^2 \quad (7.1)$$



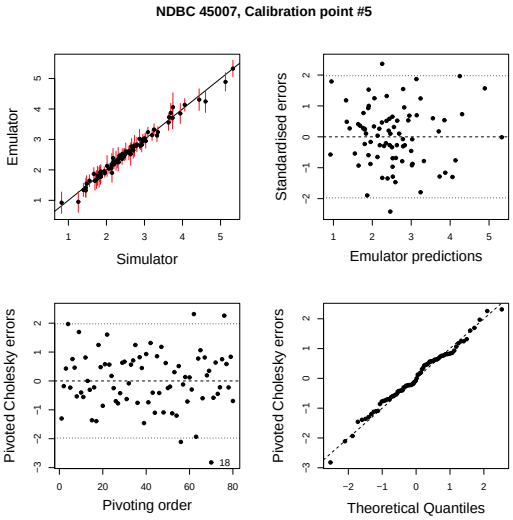
(a) Mahalanobis distance = 66.4



(b) Mahalanobis distance = 111.0



(c) Mahalanobis distance = 81.2



(d) Mahalanobis distance = 74.0

Figure 7.20: Examples of graphical diagnostics for the sixteen emulators formulated for the calibration process. The reference distribution for the Mahalanobis distance is mean = 80.0, s.d. = 15.8.

which can also be written as,

$$\mathcal{C} = B^{\mathrm{T}} S B \tag{7.2}$$

where,

$$B = \begin{bmatrix} H_{s(Sim.45002)_1} - H_{s(Obs.45002)_1} \\ \vdots \\ H_{s(Sim.45002)_{\mathcal{P}}} - H_{s(Obs.45002)_{\mathcal{P}}} \\ H_{s(Sim.45007)_1} - H_{s(Obs.45007)_1} \\ \vdots \\ H_{s(Sim.45007)_{\mathcal{P}}} - H_{s(Obs.45007)_{\mathcal{P}}} \end{bmatrix} \tag{7.3}$$

and S is a $2\mathcal{P} \times 2\mathcal{P}$ diagonal matrix with the first $\mathcal{P}$ diagonal elements set to 1, and the remainder set to α. Writing (7.1) in this way highlights how the cost function can be interpreted in terms of uncertainty about the observations. The matrix S appears very much like the inverse of the covariance matrix V in equation (3.31) for the Mahalonobis distance. In that context, we could assume that S represents some kind of Gaussian measurement error or uncertainty. Firstly, then, noting that S is diagonal, we are implying that observations are independent and uncorrelated. This is clearly not a reasonable assumption since the time series is autocorrelated. Secondly, with the inclusion of α in the diagonal, we are able to adjust the "variance" associated with the observations made at NDBC 45007. A higher value of α implies a higher weight, or lower variance, associated with these observations. Furthermore, with a constant variance given to each data buoy we are asserting that the variance is not a function of the wave height, and as such giving a higher weight to larger observations. We may expect uncertainty in the observations to scale with their magnitude, thus in some sense "moderating" the influence of high valued observations. With the cost function formulated as given, "moderation" (i.e. re-weighting of the observations) can be achieved rather crudely using α, although in the absence of any information on measurement error, its adjustment must be made pragmatically. Also, this affects all observations made at each data buoy simultaneously, rather than individual measurements. If more detailed information about observation error were available it could be incorporated in S. Noting the comments above, we can see that the cost function has a number of limitations. Nonetheless, it represents a starting point for the calibration process in this particular case, and could be developed, as described, were more information on observation uncertainties available.

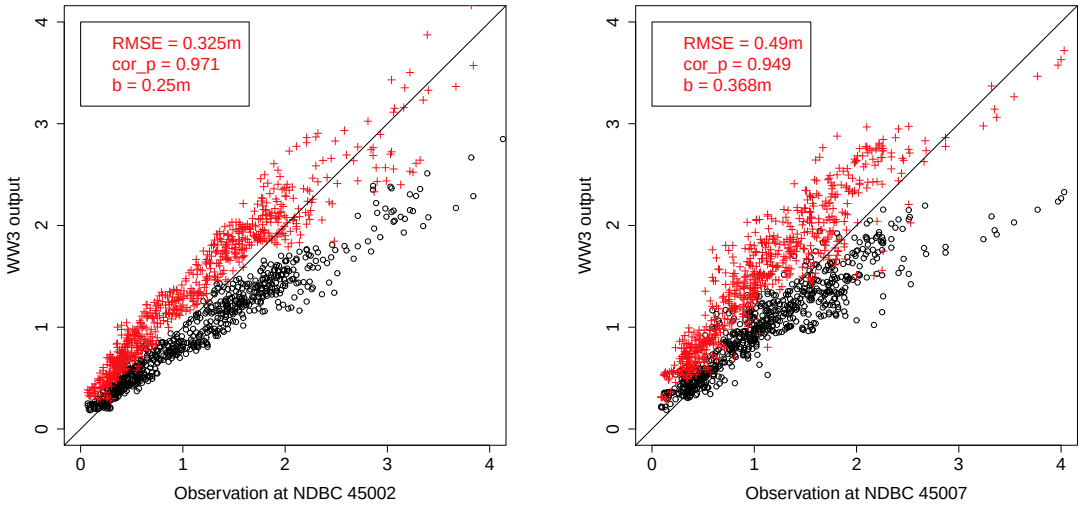
In this case, $\mathcal{P}$ is set to 8, and α is set to 1 initially. Making use of an exponential transform to modify the input domain of optimisation to $[-\infty, 0]$, the cost function

Table 7.3: Tuning parameter values found through calibration (7.E).

Parameter	**Calibration #1** ($\alpha = 1$)	**Calibration #2** ($\alpha = 5$)	**Calibration #3** ($\alpha = 10$)
sdsa0	4.94	4.8	4.91
sdsa1	6.31×10^{-5}	6.31×10^{-5}	0.00316
sdsa2	1.44	1.71	2.12
sdsb0	-0.007	-0.00698	-0.007
sdsb1	0.68	0.7	0.4
phimin	0.001	0.003	0.0023
λ_{DIA}	0.28	0.28	0.29
C_{DIA}	3.33×10^6	3.33×10^6	8.32×10^6

is then minimised using a constrained quasi-Newton method. This is necessary in order to avoid evaluating the emulators outside their validated range of $[0, 1]$. The optimisation method is described in Byrd et al. (1995).

Tuning parameter values found to minimise the cost function (7.1) with $\alpha = 1$ are shown in the second column of table 7.3. Using the new parameterisation, results of a simulation for the month of October 2012 are shown in figure 7.21. The comparison of these with observation are shown by red crosses, together with a similar comparison made between the default parameterisation and observations, shown by black circles. What we can see immediately is that the simulator now has a positive bias in conditions where H_s is below approximately 3 metres. However, at the most extreme values of H_s, above 3 metres, we see that the simulator is performing very much better. Noting that the new parameterisation results in a strongly positive bias, it appears that the process of fitting the more extreme values has had the undesirable effect of causing an increase in simulated H_s in all conditions. In effect it has over-fitted the extreme data. On the assumption that the minimisation of the cost function located the true minimum, it would seem that there is no combination of tuning parameters values within the range of the analysis that can give good estimates of both low and high values of H_s. The calibration process is therefore finding a balance between fitting calm conditions (which are simulated well by default), and fitting extreme conditions (which are simulated poorly by default). In order to penalise parameterisations that cause deviation from the (generally good) simulation in calmer conditions (e.g. at NDBC 45007 in this case), the cost function can be modified using the weighting parameter α. Increasing the value of α has the desired effect, so the calibration process was therefore repeated for $\alpha = 5, 10$. The values of the parameters found during the calibration are given in table 7.3 (columns 3 and 4). Before moving on to consider the results let us consider the values of the

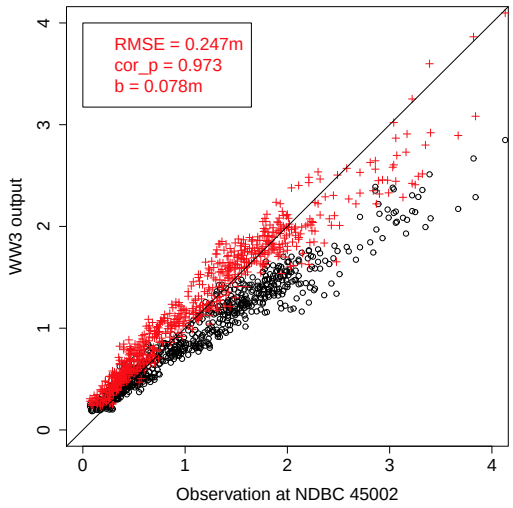


(a) $\alpha = 1$ **(b)** $\alpha = 1$

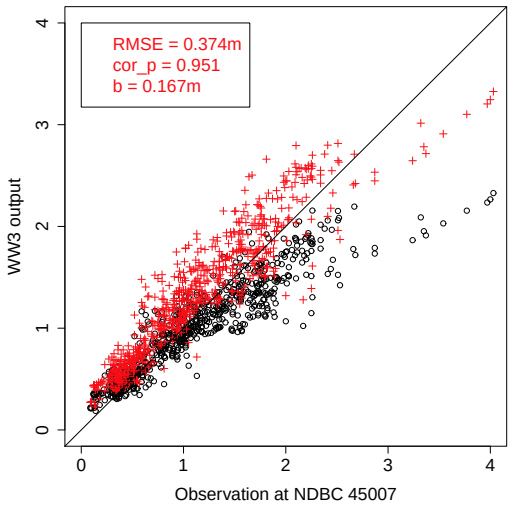
Figure 7.21: Simulated H_s using the default parameterisation (black circles) and the parameterisation after calibration (red crosses), compared with observations.

tuning parameters identified. Recalling that these parameters do not generally have a specific physical interpretation, the physical insight we can gain may be somewhat limited. However, it is clear that those parameters seen previously to be influential are active here. Specifically, *sdsa2* shows an increasing trend as the cost function penalises more extreme conditions. *sdsb0*, which has previously shown to give a negative linear response in H_s, is found to be optimal at its low design limit. C_{DIA} is found to be optimal at its low design limit in the first two cases but as the cost function penalises more extreme conditions, higher values are preferential. Since this typically induces an inverse response in simulated H_s, this behaviour is also consistent with that seen previously. An important question raised here is, given that some parameter values have been pushed to their design limits, would a more effective parameterisation be found if the experiment was designed over a wider parameter space? The answer would appear to be that they might very well be, although any further extension to the parameter space would be further away from our beliefs according to the original elicitation. The method of history-matching could be a useful tool to explore this further.

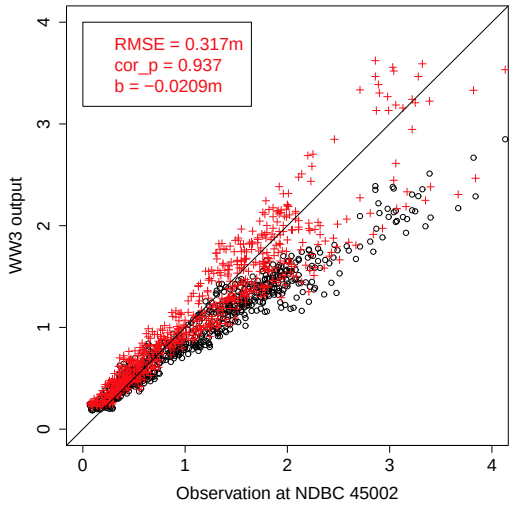
Proceeding without further consideration of the specifics of the parameterisations, comparison of simulation results with observations for calibration with $\alpha = 5, 10$ are shown in figure 7.22. We can see that with the higher weighting on the calibration points for NDBC 45007 (where calmer conditions are typically observed), the resulting parameterisation gives output with a reduced bias. The



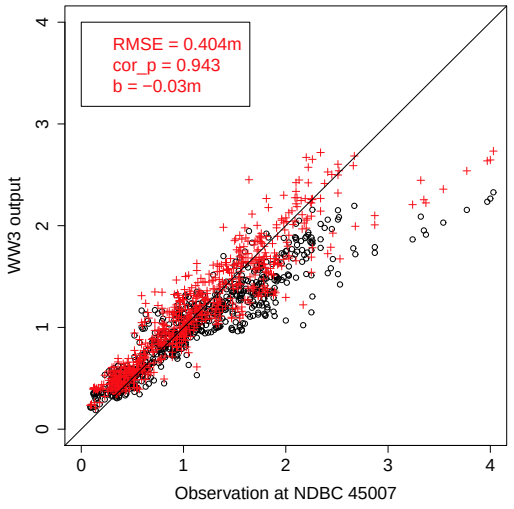
(a) $\alpha = 5$



(b) $\alpha = 5$



(c) $\alpha = 10$



(d) $\alpha = 10$

Figure 7.22: Simulated H_s compared with observations at NDBC 45002 and 45007 using the default parameterisation (black circles), and after calibration (red crosses) with $\alpha = 5, 10$.

Table 7.4: Comparison of performance statistics for the default WW3 parameterisation and alternatives found through calibration (7.E).

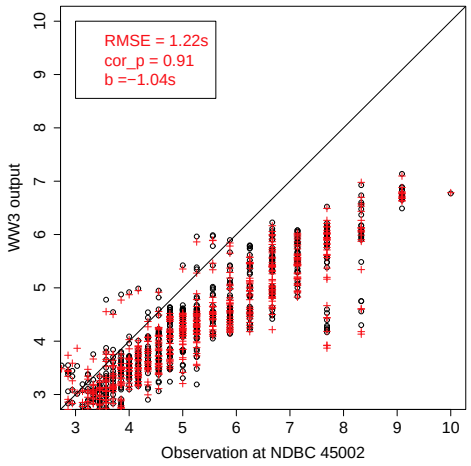
	NDBC 45002				NDBC 45007			
	Default	$\alpha = 1$	$\alpha = 5$	$\alpha = 10$	Default	$\alpha = 1$	$\alpha = 5$	$\alpha = 10$
$RMSE$	0.433	0.325	0.247	0.317	0.531	0.490	0.374	0.404
cor_P	0.972	0.971	0.973	0.937	0.942	0.949	0.951	0.943
b	-0.23	0.250	0.078	-0.021	-0.19	0.368	0.167	-0.03

top panels, for $\alpha = 5$, show that a bias remains, particularly for predictions at NDBC 45007, however good performance is still seen for the more extreme values of H_s. The bottom panels, for $\alpha = 10$, show that the positive bias has been largely removed but at the price of poorer performance at the more extreme values, particularly at NDBC 45007. Nonetheless, regarding the simulator's skill in predicting conditions for the period of October 2012, where stormier conditions are expected (and it probably applies more generally to such conditions), both new parameterisations give better performance than the default. These are summarised in table 7.4. From a safety point of view, underestimation of wave conditions is generally far more dangerous than overestimation[62], especially in extreme conditions. Consequently it could be argued that either of the new parametrisations ($\alpha = 5$ or $\alpha = 10$) should be implemented for stormy weather. The fact that the default scheme dramatically underpredicts in extreme conditions is certainly something that needs correcting.

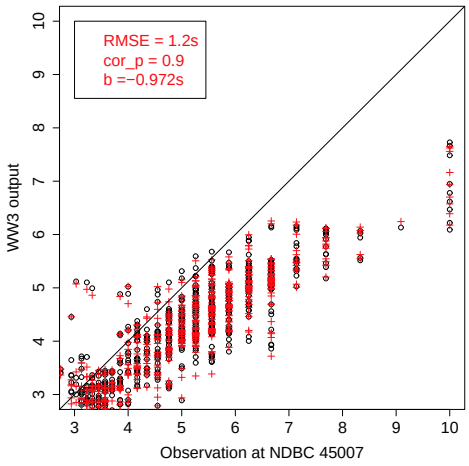
Although the calibration process did not incorporate observational data for T_p at either buoy, it is instructive to see how well T_p is simulated using the new parameterisations. Scatter plots are shown in figure 7.23. The poor performance in all cases is quite apparent. We have seen throughout this thesis that T_p is fairly insensitive to changes in tuning and this case is very similar. In spite of the considerable performance improvement in the simulation of H_s obtained through the calibration process, the new parameterisations do not appear to have much affect on T_p. The largest improvement is seen for $\alpha = 10$ where the $RMSE$ and bias at NDBC 45002 are reduced by 18% and 32% respectively. These are significant improvements but the overall predictive skill remains fairly poor.

In summary of this experiment some further comments are made. Firstly, the calibration process illustrated here is simple and does not formulate model discrepancy or take into account observation uncertainty. The choice of cost function is also somewhat pragmatic, and involves only a limited amount of the available in-

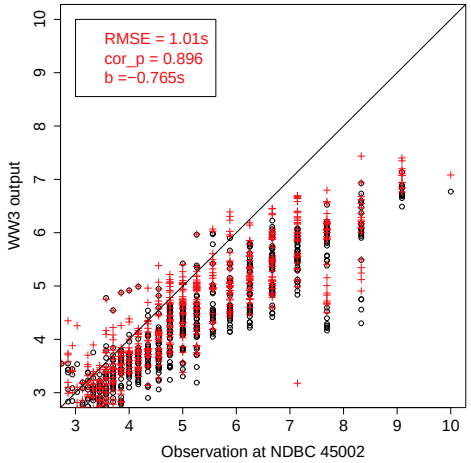
[62]Other costs may be associated with overestimation of conditions, but these are usually measured in economic terms, for example due to the unnecessary suspension of public services.



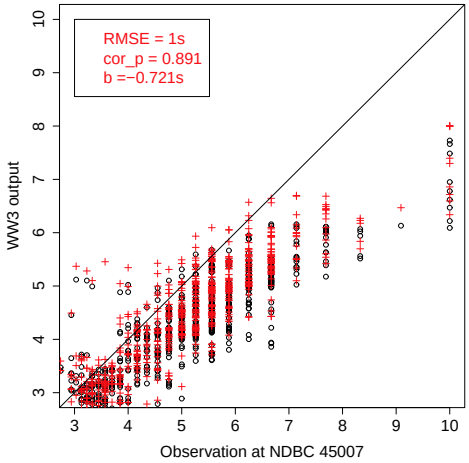
(a) $\alpha = 5$



(b) $\alpha = 5$



(c) $\alpha = 10$



(d) $\alpha = 10$

Figure 7.23: Simulated T_p compared with observations after calibration. Note performance statistics for the default case at NDBC 45002 are $RMSE = 1.19s, cor_P = 0.907, b = -1.01s$ and at NDBC 45007 are $RMSE = 1.21s, cor_P = 0.889, b = -0.958s$.

formation. However what has been shown is that emulators can be used to great effect when calibrating a simulator. It has been clearly demonstrated that improved performance can be obtained quite cheaply. Owing to project time constraints this particular problem is not considered in more detail here but it is easy to see that there are a range of possible avenues to explore that might give even better results. Specifically, the form of the cost function could be investigated, which might involve the inclusion of more information via the use of additional emulators. A more sophisticated approach might involve history-matching (Williamson et al., 2013) or the more complete probabilistic framework of Kennedy and O'Hagan (2001). Development of a more efficient representation of the time series data, or an emulator that accounts for the autocorrelation would also take this problem forward. Any one of these approaches might give further improvement in parameterisation for the conditions on Lake Michigan, and could no doubt be easily extended to other regional or global wave modelling problems.

7.10 Chapter summary

In this chapter uncertainty about tuning of the wave simulator has been investigated in the context of a realistic example. The issue of uncertainty about initial conditions was considered in experiment 7.A and determined that typically at least 24 hours spin-up time was required in the most energetic conditions. Experiments 7.B, 7.C and 7.D performed uncertainty and sensitivity analysis for different wind regimes in energetic conditions. In all cases uncertainty with respect to the DIA parameterisation for nonlinear wave-wave interactions was found to dominate output variance (always contributing between approximately 50% and 90%). Furthermore in energetic cases, tuning uncertainty could not account for the discrepancy between simulation output, in terms of H_s and T_p, and values observed by data buoys. Close temporal correlation between the simulation output and the observed quantities suggest that either a bias exists in the forcing winds or observations of waves in more energetic conditions, or that structural error in the simulator causes a systematic underprediction of waves during more energetic conditions. Finally in experiment 7.E a calibration exercise was performed in order to obtain a better parameterisation for the simulation of H_s in energetic conditions (typically seen in October). Sixteen emulators were used in order to explore the parameter space and minimise a cost function. Two resulting parameterisations improved the skill of the simulator for the period of October 2012.

8 Conclusions

In this thesis the effect of uncertainty on input to a numerical wave simulator has been investigated. The objectives were to perform quantitative uncertainty analysis, in order to shed light on some fundamental, and thus far unanswered, questions relating to uncertainty in wind-wave modelling. The motivation for this work was detailed in chapters 1 and 2. The proposed theory and methods, particularly relating to Gaussian process emulators, were discussed in chapter 3. Chapter 4 set out the scope of the uncertainty analysis, which also included elicitation of the input uncertainty. This was followed by analysis of simple wave model configurations and concluded by an investigation into how the uncertainty might affect predictions of generated wave power from a wave energy converting device. Chapter 5 extended the uncertainty analysis to situations of fetch-limited wave growth, making use of energy propagation in 1-dimension, and introduced sensitivity analysis for input parameters. Chapter 6 investigated simulations with a second spatial dimension, allowing for the investigation of directional waves, and the specification of more complex forcing winds. Finally, chapter 7 investigated realistic simulations on Lake Michigan and drew comparisons with findings from idealised cases. Also, an updated parameterisation was determined that gave improved performance in energetic wind conditions.

In the following sections final comments are made and conclusions drawn with respect to the findings of this research. Firstly the effectiveness of emulators is summarised. Secondly some general comments are made about the effect of uncertainty on simulations of waves. Thirdly, individual sources of uncertainty are discussed before providing some final comments.

8.1 Effectiveness of emulators

The approach to performing uncertainty and sensitivity analysis using Gaussian process emulators has proved very effective. Three uncertain input parameters were specified in chapter 4 and up to twelve were used in chapter 6. By identifying appropriate choices of the mean function through standard regression techniques, and typically making use of a nugget term, emulators were found to robustly validate. Detailed consideration of the behaviour of emulators in simple cases, shown in section 3.5.5 elucidated how problems can arise. However, results of uncertainty analysis using emulators, when compared with a direct Monte Carlo analysis on the simulator (see e.g. figure 4.8), revealed no appreciable difference between the two. In subsequent chapters emulators have successfully been used;

1. To perform quantitative uncertainty and sensitivity analysis with respect to parameter uncertainty, for a range of output wave summary statistics in a range of idealised and realistic cases;

2. To identify nonlinear response to simulation input;

3. As surrogate models to build a fast cost function based upon simulated and observed data, that can be minimised in order to find an improved parameterisation.

Recall also that efficient code (see appendix D) was written to perform uncertainty and sensitivity analysis by direct numerical integration of the emulator posterior functions.

There is one specific example of where significant problems were encountered when attempting to fit an emulator, in chapter 5 (see figure 5.15). For input parameter $\lambda_{DIA} > 0.31$ the response of T_p at fetches of 207km and 917km (and probably in between) exhibits a distinct discontinuity. Adjustment to the design was therefore made in order to exclude the discontinuous region because a requirement is that the training data is continuous. The source of the discontinuity remains unexplained but we can see that it manifests only with respect to variability in λ_{DIA} for output T_p in certain sea states. No other discontinuities were discovered and it is not in general expected to be a feature of simulator output (or any underlying physical process). Note that Tolman (2004b) identifies bifurcation in λ_{DIA} (for $0.16 < \lambda_{DIA} \leq 0.20$) when optimising the DIA with respect to the exact nonlinear computation. There is no immediate evidence that these two phenomenon are related but he speculates that the cause could be either a false minimum in the optimisation routine or genuinely as a result of strongly nonlinear integration.

8.2 Uncertainty in wave model output, and sensitivity to input

8.2.1 General comments

General comments are summarised into three main points. The first relates to overall uncertainty induced by uncertain parameter tuning and wind forcing. The second highlights the issue of parameter uncertainty and how this relates to structural error. The third, captures more generally the value of the work in this thesis in terms of providing a basis for further research. This also incorporates some notes on the nonlinear nature of wave physics and speculates as to what further research might reveal.

Based upon elicited uncertainty information (see chapter 4), output probability distributions for H_s and T_p have been computed using emulators in a number of cases. Output uncertainty is typically characterised with a s.d. of approximately 10% and <5% of absolute value of H_s and T_p respectively. Such results are observed consistently throughout this thesis whether in highly idealised cases or in the dynamic cases of simulations of Lake Michigan. In the case of Lake Michigan, simulations were conditional on the winds but nonetheless comparable output variance was found. This highlights firstly, that our uncertainty in tuning does give rise to appreciable output uncertainty. In practical terms this means for example that 2.5% of waves will have significant height exceeding 3.6m, where the mean significant height is 3m. Secondly, it is notable that T_p is considerably more constrained than H_s by the simulator. That is, the same input uncertainty results in approximately half of the output uncertainty for T_p compared to H_s. When compared to observational data this appears to contribute to poor simulator performance for T_p.

We have also seen than in spite of the large output variance induced by the input uncertainty, that in conditions of strong winds particularly, parameter uncertainty cannot fully account for the discrepancy between output and observations. Calibration of the simulator (see section 7.9) resulted in a reduction in RMSE associated with H_s for stormy conditions but at the cost of performance in less energetic conditions. Furthermore, T_p data was not included in this process, although it is clear from from the large systematic underprediction that no amount of parameter adjustment could result in particularly good performance. Bearing this in mind, two points are made. Firstly, clearly parameter tuning of the TC96 and DIA schemes *can* lead to significant performance improvement in certain conditions, particularly for H_s. Secondly however, the structure of the parameterisations appears to constrain adjustment to systematic increase in wave height predictions in all cases, and cannot give rise to the change necessary for accurate T_p prediction without drastic intervention (which seems unrealistic). Note also that in the realistic cases forced by strong winds, simulation output for H_s with mean 4m and s.d. 0.4m clearly cannot account for observed values of approaching 6m. Poor performance in energetic conditions is well known and improvements have been made but findings here are further evidence of important structural errors in the simulator. In the absence of uncertainty information about the observational data (wind speeds and wave heights) it is difficult to make more detailed statements. Proceeding on the basis that the observational error is small we can say with confidence that some kind of substantial structural error is present. This seems particularly likely when considering the simulator's sensitivity to the forcing wind. Although strong, in the

fetch-limited case at approximately 200km, the main effect of the wind on H_s suggested that an increase of 1ms^{-1} wind speed would result in 15% increase in H_s. This clearly falls well short of the uncertainty that would be required to explain the above mentioned discrepancy (of nearly 2m, or 50%). It therefore seems unlikely that forcing wind speed uncertainty is contributing significantly and that structural error is the main issue. More discussion about how the specific sources of uncertainty influence output could be used to direct improvement in the presence of structural error are given shortly.

As a final, more general point on parameter uncertainty, this work has identified little interaction between tuning parameters. In idealised cases this is perhaps not surprising owing to the linear combination of source terms in the formulation of the energy balance equation. However, given the clearly nonlinear nature of ocean waves (including nonlinear wave-wave interactions), we might have expected generally to see more nonlinearity and interaction between components of source terms, regardless of their specific formulation. That is, it may be unrealistic to expect accurate representation of the physical processes without more explicit nonlinear behaviour being evident in the analysis. Rogers et al. (2012) makes a point of highlighting that certain nonlinear phenomenon (relating to wind input and wave breaking) are captured in their new parameterisation. The parameterisation of Ardhuin et al. (2010) now implemented in WW3 is based upon the WAM4 input source term, with a new dissipation scheme, and offers a very wide range of tuning parameters. This may be indicative of the more complex (and nonlinear?) representation that might be required, and in light of the findings made in this thesis it would extremely interesting to examine the inner workings of that particular expression of the physics. This is explored further in section 9.1.

8.3 Sensitivity to wind input

Historically forcing wind has been regarded as the dominant source of uncertainty. Findings in this thesis generally seem to support this point of view. Specifically, simulations have shown very strong sensitivity to wind speed and directional uncertainty in all relevant experiments. Where winds are subject to biases of the order of at least 1ms^{-1}, especially over long periods of time, we might expect considerable resultant error in wave simulation output.

However, this is clearly not the whole story. As noted above, even the strong sensitivity cannot account for the large discrepancy between output and observations in certain conditions, and it is clear that other source terms also induce strong sensitivity. To say more about how wind uncertainty manifests in general is far

from straightforward. Performing analysis for uncertainty about forcing winds is extremely challenging - in fact it is very difficult to even state what we mean by wind uncertainty. As such, in any given case we are likely only to be able to make general statements, particularly on large or global scales. On smaller scales it may be possible to undertake more detailed experiments, as was done with the synthetic winds in experiment 6.B, and these may be useful for obtaining better localised results.

Although the simulations for Lake Michigan in chapter 7 are conditional on the wind input, it is clear that simulator performance is superior (and generally very good) in less energetic conditions. Alves et al. (2011) finds similar behaviour for Lake Michigan when using the previous release of WW3 (as used here) and demonstrates improved simulation using a version of the parameterisation of Ardhuin et al. (2010). This highlights that structural errors are being addressed, and that given higher quality winds, a focus on addressing shortfalls in the representation of the physics is indeed the most prudent approach at present.

8.3.1 Energy dissipation

We have seen that parameters that govern energy dissipation in the TC96 scheme are most influential in growing sea states. For example, this was seen in 5.14 panel (b) which showed growth of H_s at a fetch of 52km. This should of course be expected because wave breaking is most active at this time. However, even where they are most active, they tend to account for a modest proportion of the total variance, typically $<25\%$. With confidence in the input probability distributions (that is, if the elicitation process was reasonable), that would suggest that energy dissipation is, in most cases, of less importance than other sources of uncertainty (i.e. wind forcing and nonlinear wave-wave interactions).

The "pattern" of influence, that is the relative proportions of output variance explained, appears to be remarkably consistent across experiments of the same kind (e.g. idealised fetch-limited growth in chapter 5 compared with similar growth on Lake Michigan in chapter 7, compare figure 5.17 (b) with figure 7.10 (c)). Typically *sdsa0* and *phimin* are virtually inactive, whereas *sdsa1, sdsa2, sdsb0* and *sdsb1* are influential, although often induce fairly linear response. There is no apparently obvious connection to be made between their relative importances, and their relationship to the physical process, which is either low or high frequency energy dissipation in this case. It would therefore seem reasonable that these input parameters could be regarded only as set of "knobs" that could be used to arbitrarily re-tune the parameterisation as necessary, without much further consideration for their physical

interpretation. In fact, although not consistent with the elicitation previously conducted, their input probability distributions (or more specifically their variances) could be somewhat arbitrarily increased to obtain a wider range of control. The relative influence of each parameter would then be adjusted. Such an approach might give more insight into the operation of the wave model, although the input would not necessarily capture our beliefs. We can only speculate as to whether the authors of the original work ever believed the credible range of parameters to be so wide.

8.3.2 Nonlinear wave-wave interactions

The substantial, and typically highly nonlinear, influence of the tuning of the DIA scheme (via parameters λ_{DIA} and C_{DIA}) cannot be understated. This is particularly interesting given recent work such as Zakharov and Badulin (2011) who argue that the dominance of nonlinear interactions in the dynamics of the wave spectrum are somewhat overlooked. These findings would in fact suggest that effort should be expended seeking improved (but computationally cheap) parameterisations for the nonlinear integral (see also section 9.2‘ below). It is no surprise that this is indeed on-going (Tolman, 2013a) but given the huge expenditure in developing improved input and dissipation schemes over recent years, one has to question whether the balance of research effort is sensible.

Research in this thesis has not focussed on the detailed sensitivity of the full 2-D nonlinear transfer function spectrum to DIA adjustment (like the work of Tolman (2004a)) but rather on the actual impact of tuning of the DIA as implemented in a wave simulator. It is particularly interesting that Tolman (2004a) found optimal values of λ_{DIA} of between 0.16 and 0.20 whereas tuning in section 7.9 suggests values closer to 0.28. To take this forward a detailed study of the resultant spectra should be undertaken to examine why this is the case. It would seem that the nonlinear source term is a particularly important aspect of wave modelling that has perhaps been overlooked for too long. Future research should not only focus on improved parameterisations alone, but part of any study should also incorporate analysis of its interaction with other source terms, perhaps by using emulator methods. This is particularly important noting that new parameterisations, such as that presented by Tolman (2013a) introduce further tunable parameters.

8.4 Final thoughts

Returning to the original questions posed in chapter 1, we are now in much stronger position to provide answers, at least within the scope of experiments conducted

in this thesis. It may also be possible to speculate beyond these examples with a reasonable degree of confidence.

1. *How influential is the uncertainty in forcing winds in wave model predictions?* Regarding the wind speed, with a distribution characterised by $N(12.5\text{ms}^{-1}, 0.5^2)$ it is typically the dominant source of uncertainty, and accounts for between than 25% and 50% of output variance of both H_s and T_p. For higher wind speed or more variable winds we might expect this to be proportionately higher. In cases involving directional waves, uncertainty about the wind dynamics was also shown to dominate output uncertainty (>50%). This suggests it is likely to be the dominant source of uncertainty in a much broader range of cases.

2. *What choice of model input parameter values gives the best performance when hindcasting wave conditions?* Sensitivity analysis has revealed detailed quantitative information about the effect of each parameter on simulated H_s and T_p in a range of cases. This alone is sufficient to make adjustments pragmatically, but a more formal approach based upon real world observations has been demonstrated in chapter 7 to yield "optimal" parameter values that give improved performance in certain conditions. Given the approach was relatively simple, success here strongly suggests that even using the outdated[63] TC96 and DIA parameterisations, improved performance could be gained in localised regions, or even on larger scales without the need for expensive reformulation of the physics. Moreover, recently developed parameterisations could benefit enormously from the application of the methods employed here.

3. *Has wave modelling reached its limit?* Over the past few years alone the development of wave modelling has been considerable. Improvement in model physics (e.g. Ardhuin et al., 2010) has resulted in improved performance. However, that improvement was made without the potential benefits afforded by the methods presented here which are argued to provide a solid foundation for future research of this kind. Noting these potential benefits, in terms of both incorporating our beliefs into analysis, and learning about the effects of uncertainty and wave model sensitivity, it seems entirely reasonable to believe that its limits have not been reached. In particular the full benefits of recent work are unlikely to have yet been realised. It is the conclusion of this author that the application of these methods in different ways, and in any number of cases, would bring immediate gains.

[63] Recalling the abundance of recent research output and update to WW3.

Some brief further thoughts on potentially rich areas of related future research, and how to exploit them to maximum effect, are given in chapter 9.

9 Notes on future work

9.1 Uncertainty and sensitivity analysis for Wavewatch III v4.18 and larger scale simulations.

The timing of this project seems a little unfortunate with regards to the timing and availability of major updates to WW3. WW3 version 4.18 was released publicly in the second quarter of 2014 and incorporated many new physical parameterisations relating to energy input and dissipation and nonlinear wave-wave interactions. The work in this thesis lays an important foundation in terms of what we might expect to see with regards to the investigation and adjustment of other wave model parameterisations and in light of this it would be very interesting to examine in detail, using the methods presented here, recently developed physical schemes. In particular, the work of Ardhuin et al. (2010) and Tolman (2013a) offer many demonstrable performance enhancements (including on Lake Michigan). However, neither of these has been examined rigorously by methods used in this thesis, either when operating independently or together in a simulation. The methods of history-matching and probabilistic calibration could also be implemented. More importantly, what has yet to be done is apply these methods to larger scale simulations. A sensible next step, that would allow for the investigation of quite complex realistic situations while alleviating certain issues such as highly variable boundary conditions, is suggested to be the Gulf of Mexico.

9.2 Surrogate modelling of nonlinear wave-wave interactions source term

GP emulators have been used to perform uncertainty analysis with respect to a wave model in this thesis but they do have other applications. Andrianakis and Challenor (2012a) used an emulator to act as a fast (and accurate) representation of a (computationally expensive) fine scale parameterisation for convective mixing. Tolman et al. (2005) present preliminary results from experimentation of using an artificial neural network (ANN) to act as a fast surrogate for the nonlinear source term. It would appear that if an ANN could be trained to give useful results, that a GP emulator could also. The problem is not however trivial. A key issue is that the enormous variability in realistic sea states leads to a massive input / output space (i.e. some kind of space incorporating all possible sea states). Some kind of highly efficient dimension reduction technique would also be required. Tolman et al. (2005) considered only a set of single peaked spectrum thus limiting the problem for the

purpose of investigation. Nonetheless, wave spectra do have certain characteristics that might, to a degree, circumvent this problem. The nonlinear source term also exhibits a very specific structure (in frequency / direction space) that may therefore be less problematic to deal with than it might first appear. This would also lead to an interesting experimental design problem.

9.3 Bathymetry

Over recent years, in part owing to the gathering momentum in the marine renewables industry, modelling of the coastal zone has become important. This can be a particularly complex problem owing to the richer diversity of physical processes involved. An important aspect of this modelling is the representation of the bathymetry. Uncertainty is either introduced through low resolution, or even absent bathymetric data, but it is also introduced because bottom topology can change over time due to sediment transport or erosion. Extensions to the methods described here could be used to perform uncertainty analysis that takes into account bathymetric uncertainty.

A Representing the 1-D spectrum

Unidirectional sea states, particularly in more developed seas, typically exhibit a spectral shape that resembles the well known JONSWAP spectrum (in 1-D). For some kind of distribution of sea states expressed in the form of a 1-D spectrum (perhaps obtained from observational data buoys), it may be straightforward to capture the variability using an empirical or parametric basis. Within the wave model, the spectrum is represented by, typically, between 20 and 40 frequency bins. However, since the size of each bin is correlated with its neighbours, owing to the spectral shape, then there could be a number of ways of reducing the number of dimensions required to represent a spectrum. In this appendix, the use of a Gaussian process model is briefly explored as a means of representing the 1-D spectrum in a flexible way, with a view to emulating simulation output with a lower loss of information than occurs when considering summary statistics (e.g. H_s and T_p) alone.

We start by assuming that the spectrum exhibits a well defined peak frequency, f_p. Spectral intensity tends to decay rapidly on either side of the peak, as frequency increases or decreases. This point is therefore used as a design point for a Gaussian process model. Other design points may then be specified with respect to the peak frequency. Examples are shown in figure A.1, where design points are chosen based upon a logarithmic series, specifically, $[0.08, \mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_3, f_p]$Hz, where $\mathcal{L}_1$ comprises five points on a base 10 logarithmic scale between 0.912 and f_p, $\mathcal{L}_2$ comprises five points between $1.07f_p$ and $1.8f_p$ and $\mathcal{L}_3$ comprises four points between $2.3f_p$ and 0.6Hz. Working in this way we can avoid problems with modelling the spectral peak, that arise due to extreme gradient, that would be encountered for more arbitrary (e.g. LHS) designs. The Gaussian process model is formed in the same way as emulators described in chapter 3, with a mean function linear in frequency and a Matérn ($\nu = 3/2$) correlation function.

From a visual inspection, we can see from the examples shown in figure A.1 that the approach appears to perform quite well up to frequencies of approximately 0.3Hz. However, beyond this, the behaviour of the GP mean function, particularly in the bottom two panels (at longer fetch), likely due to the positioning of the design points and fairly short correlation lengths scales (0.035, 0.044, 0.024, 0.024 respectively), diverges from the spectral function in an unhelpful way. It is clear that further investigation of this approach is required before it could be more formally implemented, but nonetheless it appears promising. A logical next step would be to develop a prior mean function that more closely represents the spectral shape. This would likely facilitate a more efficient design requiring fewer than the 10 to 15

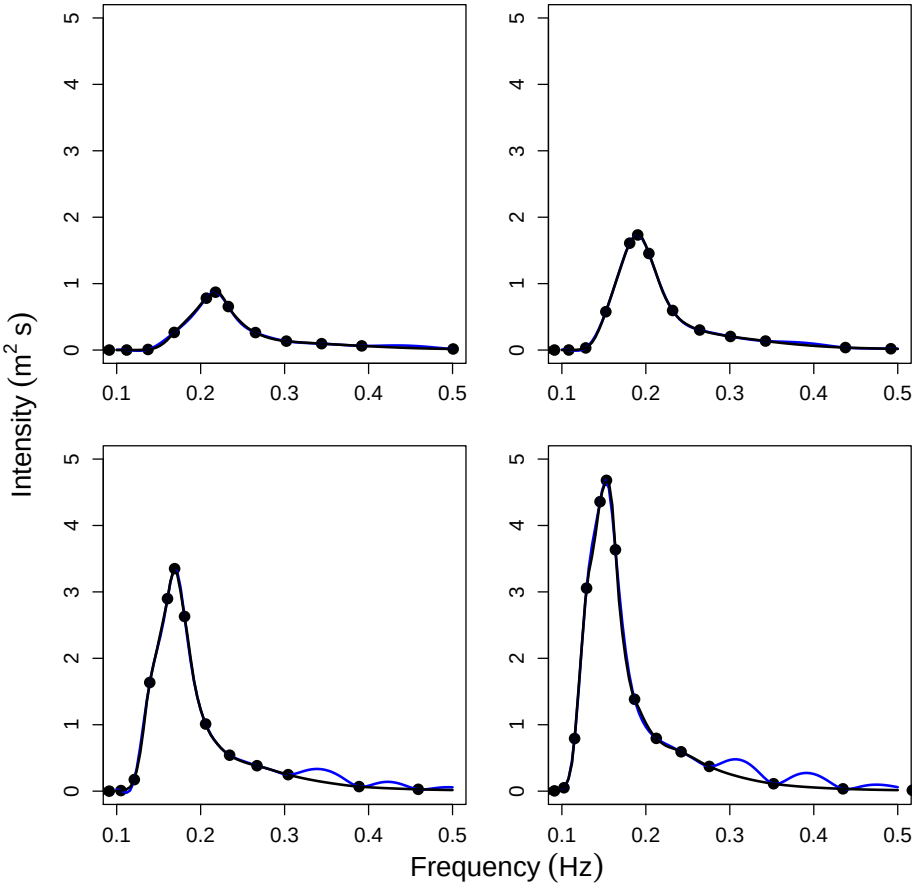


Figure A.1: A Gaussian process model fitted to a number of fetch-limited spectra (at increasing fetch) as generated by WW3. The spectral shape (solid black line) is modelled by a GP posterior mean function (solid blue line) using up to 14 design points (sold black circles.)

design points used here. Each design point could then be emulated separately or jointly when performing uncertainty or sensitivity analysis for spectral wave model output.

Table B.1: WW3 programme execution sequence

Programme	Purpose	Description
ww3_grid	Grid preprocessor.	Configures the spatial grid, together with relevant numerical (CFL parameters) and tuning parameters via namelists.
ww3_strt	Initial conditions preprocessor.	Configures initial conditions for the wave spectrum, e.g. calm or JONSWAP.
ww3_prep	Field preprocessor.	Pre-processes various input fields, such as winds. Typically supplied as separate input files.
ww3_shel	Numerical solver.	This is the parallelised part of the code. This takes input prepared by the above programmes.
ww3_outp	Point output processor.	On completion this programme interrogates the (binary) output, and provides point output in specific files. Other similar programmes can provide tracks or fields of output if requested.

B Running designed experiments with Wavewatch III

B.1 Wavewatch III sequence of execution

The operation of WW3 for the purposes of running designed experiments is summarised here, although the reader is directed to Tolman (2009a) for extensive discussion of its operation. WW3 comprises a number of auxiliary programs that are run in sequence. These are briefly described in table B.1. Each program reads an input file that specifies its various configuration information. The input files are named according to the programme name, e.g. *ww3_grid.inp*. This cannot be changed but it can be linked to any other file name for convenience. A batch of input files typically needs to be created for the ensemble run. If the files are large, i.e. for ensembles of winds, these may need to be created "on-the-fly" to avoid exceeding storage allocation. A listing of the shell batch execution script (*ww3_multi_cell.sh*) is given below.

```
1 #!/bin/sh
2 #PBS -S /bin/bash
3 #PBS -l nodes=2:ppn=12
4 #PBS -l walltime=00:45:00
5
6 module load R
7 module load openmpi/1.6.4/gcc_of2
```

```
 8
 9 #--------------------------------------#
10 # Filename: ww3_multi_cell.sh
11 # Script to run ten cells with spatially
12 # variable wind patterns.
13 # Serial or parallel capability.
14 #
15 #--------------------------------------#
16    TIME_NOW="`date +%d%h%H%M%S`"
17
18 # Number of procs.
19 #   NUM_PROCS=8
20 # PR3 propagation.
21    PATH_E=/home/bwt1g10/ww3_3.14/built_exes/xy.PR3.lin.mpi
22    PATH_SER=/home/bwt1g10/ww3_3.14/built_exes/xy.PR3.lin.shrd
23
24    SHEL_COM="mpirun -np ${NUM_PROCS} ${PATH_E}/ww3_shel"
25
26    echo
27    echo " Operating in MPI mode, ${NUM_PROCS} processors."
28    echo
29
30 # Input directory.
31 #   echo "Workdir: $PBS_O_WORKDIR"
32 #   PATH_H=$PBS_O_WORKDIR
33 #   PATH_H=$PWD
34 #   cd ${PATH_H}
35    PATH_I=${PATH_H}/input
36
37 # Create a new output directory and path.
38 #   mkdir ${PATH_H}/output/${TIME_NOW}
39 #   PATH_O=${PATH_H}/output/${TIME_NOW}
40    PATH_O=/scratch/bwt1g10/`echo ${PATH_H} | cut -d'/' -f5,6`/output/${TIME_NOW}
41    mkdir -p ${PATH_O}
42
43 # Find the type of run (e.g. training, validation etc.)
44    RUN_TYPE=`ls -l ${PATH_I}/vars | gawk '{ print $10 }' | sed -e 's/^.*_\(.*\)_.*_
          .*$/\1/'`
45    RUN_SUBTYPE=`ls -l ${PATH_I}/vars | gawk '{ print $10 }' | sed -e 's/^.*_\(.*\)_
          .*$/\1/'`
46 #   PARAMS_SUMMARY=${PATH_O}/${RUN_TYPE}_av_hs
47    SPECTRA_SUMMARY=${PATH_O}/${RUN_TYPE}_spectra
48    CELL_HS_AV=${PATH_O}/cell_hs_average
49    CELL_TE_AV=${PATH_O}/cell_te_average
50    TIME_EVOLVE_HS=${PATH_O}/time_evolve_hs
51    TIME_EVOLVE_TE=${PATH_O}/time_evolve_te
52
53 # Create a file to record the code config that was used.
54    echo ${PATH_E} > ${PATH_O}/run_info
55    echo ${PATH_SER} >> ${PATH_O}/run_info
56
57 # File to record successful and failed runs.
58    SUCCESS_SUMMARY=${PATH_O}/${RUN_TYPE}_success
59    FAILED_SUMMARY=${PATH_O}/${RUN_TYPE}_failed
60
61 # Copy important input files.
62    cp input/*grid_*.inp ${PATH_O}
```

```
63    cp input/*shel_*.inp ${PATH_O}
64    cp input/winds_* ${PATH_O}
65    cp input/vars ${PATH_O}
66
67 # Capture which paramters are considered.
68    NUM_INPUT_FILES=`ls input/grid_* | grep -c 'grid'`
69 #   NUM_INPUT_FILES=1
70    COUNT="1"
71    PARAM_NAMES=`cat input/grid_1.inp | grep -B2 '\$ End Parameters' | head -1 | sed
          -e 's/^..//'`
72 #   echo "${PARAM_NAMES}  Hs(m) L(m)  Tmean(s)" > ${PARAMS_SUMMARY}
73 #   echo "${PARAM_NAMES}" > ${SPECTRA_SUMMARY}
74
75 # Begin Wavewatch job.
76 #----------------------------------------#
77
78 # Loop over input files.
79 # Note that each loop assumes all relevant parameterised files (e.g. grid, shel,
       etc) have the same file number.
80    while [ "${COUNT}" -le "${NUM_INPUT_FILES}" ]
81       do
82
83          GRID_FILE="input/grid_${COUNT}.inp"
84          SHEL_FILE="input/shel_${COUNT}.inp"
85          STRT_FILE="input/ww3_strt1_bt.inp"
86          FIELDS_FILE="input/prep_${COUNT}.inp"
87          OUTPUT=${PATH_O}/ww3_output_${COUNT}
88          FAILED_RUN="0"
89
90 # 1. Grid pre-processor -----------------------------------------------------
91
92          echo " WW3: Executing grid preprocessor."
93
94          ln -s ./${GRID_FILE} ./ww3_grid.inp > ${OUTPUT} 2>&1
95          ln -s ./input/bottom_${COUNT} ./bottom_${COUNT}
96          echo " WW3: Executing grid preprocessor." >> ${OUTPUT}
97 #         ${PATH_E}/ww3_grid >> ${OUTPUT} 2>&1
98          ${PATH_SER}/ww3_grid >> ${OUTPUT} 2>&1
99          rm ./ww3_grid.inp ./bottom_${COUNT}
100
101 # 2. Initial conditions -----------------------------------------------------
102
103         echo " WW3: Configuring initial conditions."
104
105 #         ln -s ./${STRT_FILE} ./ww3_strt.inp >> ${OUTPUT} 2>&1
106         echo " WW3: Configuring initial conditions." >> ${OUTPUT}
107         ${PATH_SER}/ww3_strt >> ${OUTPUT} 2>&1
108 #         rm ./ww3_strt.inp
109
110 # 3. Input fields -----------------------------------------------------------
111
112         echo " WW3: Configuring fields."
113
114         ln -s ./${FIELDS_FILE} ./ww3_prep.inp >> ${OUTPUT} 2>&1
115         ln -s ./input/winds_${COUNT} ./winds_${COUNT}
116         echo " WW3: Configuring fields." >> ${OUTPUT}
117         ${PATH_SER}/ww3_prep >> ${OUTPUT} 2>&1
```

```
118          rm ./ww3_prep.inp ./winds_${COUNT}
119
120 # 4. Main program --------------------------------------------------------
121
122          echo " WW3: Executing shell."
123
124 #         ln -s ./${SHEL_FILE} ./ww3_shel.inp >> ${OUTPUT} 2>&1
125          echo " WW3: Executing shell." >> ${OUTPUT} 2>&1
126          ${SHEL_COM} >> ${OUTPUT} 2>&1
127 # Captured failure.
128          if [ $? != 0 ]
129             then
130                FAILED_RUN="1"
131          fi
132
133 #         rm ./ww3_shel.inp
134
135 # 5. Point output -------------------------------------------------------
136
137          echo " WW3: Table of mean wave paramters."
138
139          for OUTP_FILES in `ls input/ww3_outp*`
140
141             do
142                ln -s ${OUTP_FILES} ./ww3_outp.inp
143                ${PATH_SER}/ww3_outp >> ${OUTPUT} 2>&1
144                rm ./ww3_outp.inp
145             done
146
147 #          echo " WW3: GrADS output."
148
149 #          ln -s ${PATH_I}/ww3_gx_outp_bt.inp ./gx_outp.inp
150 #          ${PATH_E}/gx_outp >> ${OUTPUT} 2>&1
151 #          rm ./gx_outp.inp
152
153 #   ln -s $path_a/colorset.gs .
154 #   ln -s $path_a/spec.gs     .
155 #   ln -s $path_a/source.gs   .
156 #   echo 'script ww3_ts1'         > spec_ids
157 #   echo 'WAVEWATCH III TEST'    >> spec_ids
158
159 ## 5a. Set up GrADS if necessary.
160 #
161 #   grads -pc "run spec"
162 #   gxps -c -i plot.grads -o $path_o/spec.ps
163 #   rm -f plot.grads
164 #
165 #   echo '   GrADS sources'
166 #
167 #   grads -pc "run source"
168 #   gxps -c -i plot.grads -o $path_o/source.ps
169 #
170 # 6. Gather relevant output data. ---------------------------------------
171
172          echo " WW3: Wavewatch execution complete."
173
174          if [ "${FAILED_RUN}" = "0" ]
```

```
175             then
176
177 # Capture the parameter values.
178                PARAMS=`cat ${GRID_FILE} | grep -B1 '\$ End Parameters' | head -1 |
                       sed -e 's/^..//'`
179
180 # Loop over tab files.
181                for TAB_FILES in `ls tab*ww3`
182                   do
183                      cat ${TAB_FILES} | sed -e '1i '"${PARAM_NAMES}"'' \
184                      -e '1i '"${PARAMS}"'' \
185                      > ${PATH_O}/`echo ${TAB_FILES} | sed -e 's/\.ww3//'`_${COUNT}
186                   done
187
188                echo ${COUNT} >> ${SUCCESS_SUMMARY}
189
190 # Move transfer files.
191 #                mv ww3.68060123.hs ${PATH_O}/ww3.68060123.hs_${COUNT} 2> /dev/null
192                for SPC_FILES in `ls ww3.*.spc`
193                   do
194                      mv ${SPC_FILES} ${PATH_O}/${SPC_FILES}_${COUNT} 2> /dev/null
195                   done
196
197 # Move the restart file.
198                mv restart.ww3 ${PATH_O} 2> /dev/null
199
200             else
201                PARAMS=`cat ${GRID_FILE} | grep -B1 '\$ End Parameters' | head -1 |
                       sed -e 's/^..//'`
202                echo ${COUNT} >> ${FAILED_SUMMARY}
203          fi
204
205 # Clean up.
206          rm *ww3 2> /dev/null
207 #         rm wind.ww3 2> /dev/null
208 #         mkdir ${PATH_O}/dump_${COUNT} 2> /dev/null
209 #         mv *ww3 ${PATH_O}/dump_${COUNT} 2> /dev/null
210
211          COUNT=`expr ${COUNT} + 1`
212
213       done
214
215 # Following completion copy the summary files for download.
216 #   cp ${CELL_HS_AV} ${PATH_O}/../${RUN_TYPE}_${RUN_SUBTYPE}_hs_av
217 #   cp ${CELL_TE_AV} ${PATH_O}/../${RUN_TYPE}_${RUN_SUBTYPE}_te_av
218 #   cp ${TIME_EVOLVE_HS}_1 ${PATH_O}/../${RUN_TYPE}_${RUN_SUBTYPE}_hs_time
```

../code/ww3_multi_cell.sh

B.2 Designed experiments

The running of a designed experiment proceeds by writing an input file that contains all of the parameter values in columns, with the appropriate column header (e.g. var_sdsa0), typically written out from R. Template files are then created for all

relevant WW3 input files that need to be modified. A place holder for each variable is written into the template, as ¡var_sdsa¿. A processing script called *generate.sh* then writes the appropriate variable values into the place holders. Typically at least the grid input file needs to be rewritten *ww3_grid.inp* but if custom winds are used for each ensemble member then *ww3_prep.inp* will also need writing. Input files for the *ww3_shel* and *ww3_strt* may also need creating. Once template files have been created (and checked!) and the list of parameters is ready, the *generate.sh* shell script writes out the entire ensemble input. For winds, this also instigates automatic R processing to write the wind files.

```
#!/bin/sh

##########################################
# Script for writing multiple input files #
#----------------------------------------#
# Version 2.0 13/10/2011                 #
# * Takes 25 parameters                  #
# * Writes R files and runs R to generate #
#   the input wind fields.               #
#                                        #
##########################################

# Assign variables  #
#===================#

   VARS_FILE="vars"

# Process the files #
#===================#

   VAR_LIST=`cat ${VARS_FILE} | head -1`
   VAR_NAME_1=`echo ${VAR_LIST} | gawk '{ print $1 }'`
   VAR_NAME_2=`echo ${VAR_LIST} | gawk '{ print $2 }'`
   VAR_NAME_3=`echo ${VAR_LIST} | gawk '{ print $3 }'`
   VAR_NAME_4=`echo ${VAR_LIST} | gawk '{ print $4 }'`
   VAR_NAME_5=`echo ${VAR_LIST} | gawk '{ print $5 }'`
   VAR_NAME_6=`echo ${VAR_LIST} | gawk '{ print $6 }'`
   VAR_NAME_7=`echo ${VAR_LIST} | gawk '{ print $7 }'`
   VAR_NAME_8=`echo ${VAR_LIST} | gawk '{ print $8 }'`
   VAR_NAME_9=`echo ${VAR_LIST} | gawk '{ print $9 }'`
   VAR_NAME_10=`echo ${VAR_LIST} | gawk '{ print $10 }'`
   VAR_NAME_11=`echo ${VAR_LIST} | gawk '{ print $11 }'`
   VAR_NAME_12=`echo ${VAR_LIST} | gawk '{ print $12 }'`
   VAR_NAME_13=`echo ${VAR_LIST} | gawk '{ print $13 }'`
   VAR_NAME_14=`echo ${VAR_LIST} | gawk '{ print $14 }'`
   VAR_NAME_15=`echo ${VAR_LIST} | gawk '{ print $15 }'`
   VAR_NAME_16=`echo ${VAR_LIST} | gawk '{ print $16 }'`
   VAR_NAME_17=`echo ${VAR_LIST} | gawk '{ print $17 }'`
   VAR_NAME_18=`echo ${VAR_LIST} | gawk '{ print $18 }'`
   VAR_NAME_19=`echo ${VAR_LIST} | gawk '{ print $19 }'`
   VAR_NAME_20=`echo ${VAR_LIST} | gawk '{ print $20 }'`
   VAR_NAME_21=`echo ${VAR_LIST} | gawk '{ print $21 }'`
   VAR_NAME_22=`echo ${VAR_LIST} | gawk '{ print $22 }'`
```

```
44    VAR_NAME_23=‘echo ${VAR_LIST} | gawk ’{ print $23 }’‘
45    VAR_NAME_24=‘echo ${VAR_LIST} | gawk ’{ print $24 }’‘
46    VAR_NAME_25=‘echo ${VAR_LIST} | gawk ’{ print $25 }’‘
47
48    COUNT="2"
49    VARS=‘cat ${VARS_FILE} | sed -ne ’’${COUNT}’p’‘
50
51    while [ "${VARS}" != "" ]
52       do
53
54          VAR_1=‘echo ${VARS} | gawk ’{ print $1 }’‘
55          VAR_2=‘echo ${VARS} | gawk ’{ print $2 }’‘
56          VAR_3=‘echo ${VARS} | gawk ’{ print $3 }’‘
57          VAR_4=‘echo ${VARS} | gawk ’{ print $4 }’‘
58          VAR_5=‘echo ${VARS} | gawk ’{ print $5 }’‘
59          VAR_6=‘echo ${VARS} | gawk ’{ print $6 }’‘
60          VAR_7=‘echo ${VARS} | gawk ’{ print $7 }’‘
61          VAR_8=‘echo ${VARS} | gawk ’{ print $8 }’‘
62          VAR_9=‘echo ${VARS} | gawk ’{ print $9 }’‘
63          VAR_10=‘echo ${VARS} | gawk ’{ print $10 }’‘
64          VAR_11=‘echo ${VARS} | gawk ’{ print $11 }’‘
65          VAR_12=‘echo ${VARS} | gawk ’{ print $12 }’‘
66          VAR_13=‘echo ${VARS} | gawk ’{ print $13 }’‘
67          VAR_14=‘echo ${VARS} | gawk ’{ print $14 }’‘
68          VAR_15=‘echo ${VARS} | gawk ’{ print $15 }’‘
69          VAR_16=‘echo ${VARS} | gawk ’{ print $16 }’‘
70          VAR_17=‘echo ${VARS} | gawk ’{ print $17 }’‘
71          VAR_18=‘echo ${VARS} | gawk ’{ print $18 }’‘
72          VAR_19=‘echo ${VARS} | gawk ’{ print $19 }’‘
73          VAR_20=‘echo ${VARS} | gawk ’{ print $20 }’‘
74          VAR_21=‘echo ${VARS} | gawk ’{ print $21 }’‘
75          VAR_22=‘echo ${VARS} | gawk ’{ print $22 }’‘
76          VAR_23=‘echo ${VARS} | gawk ’{ print $23 }’‘
77          VAR_24=‘echo ${VARS} | gawk ’{ print $24 }’‘
78          VAR_25=‘echo ${VARS} | gawk ’{ print $25 }’‘
79
80          LOC_COUNT=‘expr ${COUNT} - 1‘
81
82 # Loop over the dfferent input templates files.
83          for TEMPLATE in ‘ls inp_template*‘
84             do
85
86                TEMP_TYPE=‘echo ${TEMPLATE} | cut -f3 -d’_’‘
87                OUTPUT_FILE="${TEMP_TYPE}_${LOC_COUNT}.inp"
88
89                cat ${TEMPLATE} | sed \
90                -e ’/^\$ End Parameters/i \$ ’${VAR_NAME_1}’ ’${VAR_NAME_2}’ ’${VAR_
                      NAME_3}’ ’${VAR_NAME_4}’ ’${VAR_NAME_5}’ ’${VAR_NAME_6}’ ’${VAR_
                      NAME_7}’ ’${VAR_NAME_8}’ ’${VAR_NAME_9}’ ’${VAR_NAME_10}’ ’${VAR
                      _NAME_11}’ ’${VAR_NAME_12}’ ’${VAR_NAME_13}’ ’${VAR_NAME_14}’ ’$
                      {VAR_NAME_15}’ ’${VAR_NAME_16}’ ’${VAR_NAME_17}’ ’${VAR_NAME_18}
                      ’ ’${VAR_NAME_19}’ ’${VAR_NAME_20}’ ’${VAR_NAME_21}’ ’${VAR_NAME
                      _22}’ ’${VAR_NAME_23}’ ’${VAR_NAME_24}’ ’${VAR_NAME_25}’’ \
91                -e ’/^\$ End Parameters/i \$ ’${VAR_1}’ ’${VAR_2}’ ’${VAR_3}’ ’${VAR
                      _4}’ ’${VAR_5}’ ’${VAR_6}’ ’${VAR_7}’ ’${VAR_8}’ ’${VAR_9}’ ’${
                      VAR_10}’ ’${VAR_11}’ ’${VAR_12}’ ’${VAR_13}’ ’${VAR_14}’ ’${VAR_
                      15}’ ’${VAR_16}’ ’${VAR_17}’ ’${VAR_18}’ ’${VAR_19}’ ’${VAR_20}’
```

```
                    '${VAR_21}' '${VAR_22}' '${VAR_23}' '${VAR_24}' '${VAR_25}'' \
92              -e 's/<'${VAR_NAME_1}'>/'${VAR_1}'/g' \
93              -e 's/<'${VAR_NAME_2}'>/'${VAR_2}'/g' \
94              -e 's/<'${VAR_NAME_3}'>/'${VAR_3}'/g' \
95              -e 's/<'${VAR_NAME_4}'>/'${VAR_4}'/g' \
96              -e 's/<'${VAR_NAME_5}'>/'${VAR_5}'/g' \
97              -e 's/<'${VAR_NAME_6}'>/'${VAR_6}'/g' \
98              -e 's/<'${VAR_NAME_7}'>/'${VAR_7}'/g' \
99              -e 's/<'${VAR_NAME_8}'>/'${VAR_8}'/g' \
100             -e 's/<'${VAR_NAME_9}'>/'${VAR_9}'/g' \
101             -e 's/<'${VAR_NAME_10}'>/'${VAR_10}'/g' \
102             -e 's/<'${VAR_NAME_11}'>/'${VAR_11}'/g' \
103             -e 's/<'${VAR_NAME_12}'>/'${VAR_12}'/g' \
104             -e 's/<'${VAR_NAME_13}'>/'${VAR_13}'/g' \
105             -e 's/<'${VAR_NAME_14}'>/'${VAR_14}'/g' \
106             -e 's/<'${VAR_NAME_15}'>/'${VAR_15}'/g' \
107             -e 's/<'${VAR_NAME_16}'>/'${VAR_16}'/g' \
108             -e 's/<'${VAR_NAME_17}'>/'${VAR_17}'/g' \
109             -e 's/<'${VAR_NAME_18}'>/'${VAR_18}'/g' \
110             -e 's/<'${VAR_NAME_19}'>/'${VAR_19}'/g' \
111             -e 's/<'${VAR_NAME_20}'>/'${VAR_20}'/g' \
112             -e 's/<'${VAR_NAME_21}'>/'${VAR_21}'/g' \
113             -e 's/<'${VAR_NAME_22}'>/'${VAR_22}'/g' \
114             -e 's/<'${VAR_NAME_23}'>/'${VAR_23}'/g' \
115             -e 's/<'${VAR_NAME_24}'>/'${VAR_24}'/g' \
116             -e 's/<'${VAR_NAME_25}'>/'${VAR_25}'/g' \
117             -e 's/<winds_file_name>/winds_'${LOC_COUNT}'/g' \
118             -e 's/<bottom_file_name>/bottom_'${LOC_COUNT}'/g' \
119             > "${OUTPUT_FILE}"
120          done
121
122       COUNT=`expr ${COUNT} + 1`
123       VARS=`cat ${VARS_FILE} | sed -ne ''${COUNT}'p'`
124
125    done
126
127 # Run R to generate the wind files.
128    rm bottom_* 2> /dev/null
129    for R_FILES in `ls windsR* bottomR*`
130       do
131
132          R CMD BATCH ${R_FILES}
133          mv ${R_FILES} R_input_files
134          rm *Rout
135       done
```

../code/generate_ww.sh

B.3 WW3 directory structure

The directory structure used for running the code is as follows.

```
1 fetch_limited_2.d
2 |-- input
3 |   |-- R_input_files
```

```
|   |   |-- bottomR_1.inp
|   |   `-- windsR_1.inp
|   |-- archive
|   |   |-- inp_template_simple_windsR
|   |   |-- ww3_outp_table_param_centre.inp
|   |   |-- ww3_outp_table_param_east.inp
|   |   |-- ww3_outp_table_param_north.inp
|   |   |-- ww3_outp_table_param_south.inp
|   |   `-- ww3_outp_table_param_west.inp
|   |-- bottom_1
|   |-- control_var
|   |-- create_fetch_points.R
|   |-- design_data
|   |   |-- archive
|   |   |   `-- fetch_lim1_design_1_data
|   |   |-- fetch_lim2.tar
|   |   |-- fetch_lim2_design_1_data
|   |   |-- fetch_lim2_design_2_data
|   |   |-- fetch_lim2_design_3_data
|   |   |-- fetch_lim2_design_4_data
|   |   |-- fetch_lim2_design_5_data
|   |   `-- fetch_lim2_design_6_data
|   |-- generate_ww.sh
|   |-- grid_1.inp
|   |-- inp_bottom_50cell
|   |-- inp_template_bottomR
|   |-- inp_template_grid
|   |-- inp_template_prep
|   |-- inp_template_windsR
|   |-- mean_var
|   |-- mean_var_beta
|   |-- prep_1.inp
|   |-- simple_1.inp
|   |-- stream_function_winds_sq.R
|   |-- template_shel
|   |-- test_var
|   |-- test_winds_file
|   |-- vars -> ./mean_var_beta
|   |-- w3_outf_params1.inp
|   |-- winds_1
|   |-- ww3_outp_1Dspec_021200.inp
|   |-- ww3_outp_2Dspec_020000_p36_series.inp
|   |-- ww3_outp_2Dspec_021200_fetch.inp
|   |-- ww3_outp_2Dspec_021600_fetch.inp
|   |-- ww3_outp_2Dspec_042300_fetch.inp
|   |-- ww3_outp_means_centre_fetch_020000_series.inp
|   |-- ww3_outp_means_centre_fetch_021600.inp
|   |-- ww3_outp_means_centre_fetch_042300.inp
|   |-- ww3_outp_means_centre_fetch_p36_010000_series.inp
|   |-- ww3_outp_table_param_north_fetch_020000_series.inp
|   |-- ww3_outp_table_param_south_fetch_020000_series.inp
|   |-- ww3_shel.inp
|   |-- ww3_strt_FLjonswap.inp
|   `-- ww3_strt_calm.inp
|-- output
|-- run_batch.sh
|-- run_reports
```

```
|   |-- run_batch.sh.e1271242
|   |-- run_batch.sh.e1271838
|   |-- run_batch.sh.e1272418
|   |-- run_batch.sh.e1313311
|   |-- run_batch.sh.e324434
|   |-- run_batch.sh.e346310
|   |-- run_batch.sh.e347398
|   |-- run_batch.sh.e358601
|   |-- run_batch.sh.o1271242
|   |-- run_batch.sh.o1271838
|   |-- run_batch.sh.o1272418
|   |-- run_batch.sh.o1313311
|   |-- run_batch.sh.o324434
|   |-- run_batch.sh.o346310
|   |-- run_batch.sh.o347398
|   `-- run_batch.sh.o358601
|-- ww3_multi_cell.sh
|-- ww3_multi_cell_login.sh
|-- ww3_shel.inp -> ./input/ww3_shel.inp
`-- ww3_strt.inp -> ./input/ww3_strt_FLjonswap.inp

7 directories, 72 files
```

../code/fetch_limited_2_tree

C Towards an analytical solution for uncertainty analysis using the Matérn correlation function

O'Hagan (2011) derives analytical expressions for a number of important results when conducting inference with a GP emulator. This was made more straightforward by the use of joint normal distributions for the prior uncertainties, and a prior Gaussian correlation function (3.10) for the emulator. Subsequent multiplication required during the calculation results in another Gaussian form, which leads to a closed form solution. In the case where, say a Matérn correlation function is used, no such closed form solution exists. Here a brief analysis of this problem is presented.

We proceed in a similar fashion to O'Hagan (2011) section 4.1 by considering the scalar function,

$$c(x, x_k)\varphi(x) = \big(1 + D|x - x_k|\big)\, exp\big(-D|x - x_k|\big)\, exp\big(-\frac{1}{2}(x - m)^{\mathrm{T}} B(x - m)\big) \tag{C.1}$$

where $D = \sqrt{3}/l$ and l is a correlation length scale parameter.

Immediately we see that the presence of the *abs* function hampers our progress. We wish to calculate the following integral,

$$\int_{-\infty}^{\infty} c(x, x_k)\varphi(x)dx \tag{C.2}$$

and consideration of the function $exp(-|x - x'|)$ seen in figure C.1 reveals that integration can be done in two parts by setting the limits of integration from $-\infty$ to x' in the first sum, and then x' to ∞ in the second. So in terms of our functions of x and x_k that is,

$$\int_{-\infty}^{\infty} c(x, x_k)\varphi(x)dx = \int_{-\infty}^{x_k} c(x, x_k)\varphi(x)dx + \int_{x_k}^{\infty} c(x, x_k)\varphi(x)dx \tag{C.3}$$

At this point we expand out the terms and rearrange to give,

$$\int_{-\infty}^{x_k} c(x, x_k)\varphi(x)dx = \int_{-\infty}^{x_k} \mathbb{P}(x, x_k) \exp\big(\mathbb{Q}(x, x_k)\big)dx \tag{C.4}$$

$$\int_{x_k}^{\infty} c(x, x_k)\varphi(x)dx = \int_{x_k}^{\infty} \mathbb{P}'(x, x_k) \exp\big(\mathbb{Q}'(x, x_k)\big)dx \tag{C.5}$$

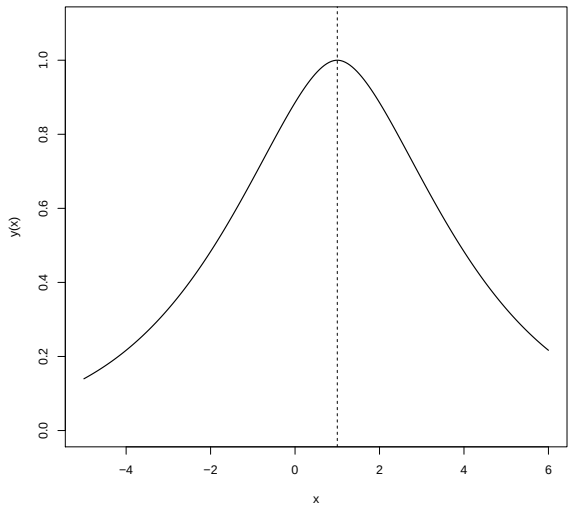


Figure C.1: Function $y = exp(-|x - x'|)$, where $x' = 1$.

where,

$$\mathbb{P}(x, x_k) = \big(1 - D(x - x_k)\big) \tag{C.6}$$

$$\mathbb{P}'(x, x_k) = \big(1 + D(x - x_k)\big) \tag{C.7}$$

$$\mathbb{Q}(x, x_k) = D(x - x_k) - \frac{1}{2}(x - m)^{\mathrm{T}} B(x - m) \tag{C.8}$$

$$\mathbb{Q}'(x, x_k) = -D(x - x_k) - \frac{1}{2}(x - m)^{\mathrm{T}} B(x - m) \tag{C.9}$$

Noting that $\mathbb{P}(x, x_k)$ multiplies the exponential, in order to find closed form solutions we require that $\mathbb{P}(x, x_k) = a\mathbb{R}(x) + \mathbb{S}(x_k)$ where $\mathbb{R}(x) = \frac{d}{dx}\mathbb{Q}(x, x_k)$, thus yielding a term $\frac{d}{dx}\mathbb{Q}(x, x_k)\exp(\mathbb{Q}(x, x_k))$.

So working on the expressions for $\mathbb{P}$ and $\mathbb{Q}$ for the moment (we will address $\mathbb{P}'$ and $\mathbb{Q}'$ for later), after some re-arranging we find,

$$\frac{d}{dx}\mathbb{Q}(x, x_k) = -(Bx - Bm - D) \tag{C.10}$$

And after more work it can be shown that,

$$\mathbb{P}(x, x_k) = \left[1 + Dx_k - Dm - \frac{D^2}{B}\right] - \frac{D}{B}\left[Bx - Bm - D\right] \tag{C.11}$$

where,

$$a\mathbb{R}(x) = -\frac{D}{B}\left[Bx - Bm - D\right] \tag{C.12}$$

$$\mathbb{S}(x_k) = \left[1 + Dx_k - Dm - \frac{D^2}{B}\right] \tag{C.13}$$

Thus $a = D/B$ and so,

$$a\int_{-\infty}^{x_k} \mathbb{R}(x)\exp\big(\mathbb{Q}(x, x_k)\big)dx = \frac{D}{B}\exp\big(\mathbb{Q}(x_k, x_k)\big) \tag{C.14}$$

We now have to deal with the integral for the remaining $\mathbb{S}(x_k)$ term. Rearranging the terms in $\mathbb{Q}(x, x_k)$, and making use of the standard solution for the definite integral of a Gaussian form,

$$\int_{-\infty}^{x_k} \mathbb{S}(x_k)\exp\big(\mathbb{Q}(x, x_k)\big)dx =$$
$$\Big(1 + Dx_k - Dm - \frac{D^2}{B}\Big)exp\Big(\frac{D^2}{2B} - Dx_k + Dm\Big)\sqrt{\frac{\pi}{4B/2}}\left(erf\left(\sqrt{\frac{B}{2}}(x_k - (\frac{D}{B} + m))\right) + 1\right) \tag{C.15}$$

Summing (C.14) and (C.15) then gives an expression for (C.4), which is the first part of the desired integration. Similar expressions can be derived for (C.5) but are not shown here, however the total sum then equals the full integration (C.2).

D Numerical integration code for arbitrary (linear) mean and correlation functions

A detailed guide to the computation of variance based sensitivity measures is provided in the MUCM Toolkit[64]. A large number of integrals have to be computed, and for mathematically tractable choices of emulator covariance function (namely Gaussian) and joint distribution for the analysis inputs (typically independent Gaussian), the entire computation can be done analytically. However as noted in chapter 3, it is often more efficient to adopt a covariance function other than a Gaussian, such as one of the Matérn family of functions. Now the resulting integrations become more difficult, largely owing to the intractable *abs* function (see appendix C for an indication).

D.1 Integrals

It is fairly straightforward, if a little laborious, to integrate the emulator function numerically, in order to obtain the integrals listed in the MUCM Toolkit. Computation is thus slower than an analytical solution but substantially better than attempting to use the emulator code as part of a Monte Carlo integration scheme. Not all the integrals are discussed but R_{hh} and R_{ht} are considered. O'Hagan (2011) lists (in section 2.4) a number of integrals used to conduct various inference. Of particular note are,

$$\mathrm{E}^*[M] = \int m^*(x)\, dg(x) \tag{D.1}$$

$$\mathrm{Var}^*[M] = \int\int v^*(x, x')\, dg(x)\, dg(x') \tag{D.2}$$

$$\mathrm{E}^*[V] = (I_1 - \mathrm{Var}^*[M]) + (I_2 - \mathrm{E}^*[M]^2) \tag{D.3}$$

where,

$$I_1 = \int v^*(x, x)\, dg(x) \tag{D.4}$$

$$I_2 = \int m^*(x)^2\, dg(x) \tag{D.5}$$

Given the expressions for the emulator posterior mean and covariance functions, (3.26) and (3.27) respectively, it is fairly easy to see how (D.1) are computed by a number of other integrals given by O'Hagan (2011), section 3.1. For example if we

[64] http://mucm.aston.ac.uk/MUCM/MUCMToolkit/index.php?page=ProcVarSAGP.html

integrate (3.26) with respect to $g(x)$,

$$\mathrm{E}^*[M] = \int m^*(x)\, dg(x) = \int h(x)^{\mathrm{T}}\hat{\beta} + t(x)^{\mathrm{T}}\mathrm{e}\, dg(x) \tag{D.6}$$

$$= \int h(x)^{\mathrm{T}}\, dg(x)\hat{\beta} + \int t(x)^{\mathrm{T}}\, dg(x)\mathrm{e} \tag{D.7}$$

$$= R_h^{\mathrm{T}}\hat{\beta} + R_t^{\mathrm{T}}\mathrm{e} \tag{D.8}$$

where $\mathrm{e} = \mathrm{A}^{-1}(\mathbf{y} - \mathrm{H}\hat{\beta})$. In section 4.1, O'Hagan (2011) goes on to provide analytical solutions for R_h^{T} and R_t^{T}, given choices of $h(x) = [1, x]^{\mathrm{T}}$ and Gaussian correlation function for the prior model. In the general case of linear mean with basis functions incorporating higher order terms of x, and arbitrary correlation function, analytical solutions are not readily available. However, it is possible to make use of the integral forms given in section 3.2 by simply integrating numerically to obtain each term. On a modern computer system, and depending on the number of dimensions and size of the design matrix, the integrations tend to take seconds to minutes. Code is listed below for a number of integrals listed in O'Hagan (2011), although the remainder are used only for the calculation of $\mathrm{Var}^*[V]$, which has not been used in this thesis.

```
1  # source("/home/ben/phd/code/R/experiments/canals/scale_3a_thesis.d/R_Hs_125_peak_
       grad/auto_unc_sens_analysis_v1.2.R")
2  #=========================================================================#
3  # Perform emulator based uncertainty and sensitivity analysis.
4  # This is "semi" analytical...
5  #=========================================================================#
6
7  # Set number of design points.
8     i_n <- num_design_points
9
10 # Specify which emulator to analyse.
11    emulator_index <- 1
12
13    nu <- 0
14    delta <- mat_max_delta[,emulator_index]
15    beta <- mat_beta
16
17    print(paste("Emulator: Delta = ", delta))
18    print(paste("Emulator: Beta = ", beta))
19    print(paste("Emulator: Sigma^2 = ",sigmasq[emulator_index]))
20
21 # Set up the pieces to perform the integrals.
22 # Conventions used from MUCM toolkit.
23    invA <- array_matinvA[,,emulator_index]
24    G <- invA %*% H
25    W <- solve(t(H) %*% G)
26    e <- invA %*% ( mat_fD - H %*% beta )
27
28    P1 <- e
29    P3 <- invA %*% H
30    P2 <- solve( t(H) %*% P3 )
```

```
31
32 # List of dimensions.
33    seq_dim <- seq(1,i_p)
34
35 #-----------------------------------------------------------------------#
36 # Emulator design range (analysis must be within these ranges).
37    param_labels <- c("sdsa0","sdsa1","sdsa2","sdsb0","sdsb1","phimin",expression(
          lambda[DIA]),expression(C[DIA]),"windspeed")
38    des_par_ran <- cbind(
39                                  rbind(4.0000,5.6000),
40                                  rbind(-4.0000,-2.5850),
41                                  rbind(1.2000,2.8000),
42                                  rbind(-0.00700,0.00050),
43                                  rbind(0.4000,0.6500),
44                                  rbind(0.0026,0.0034),
45                                  rbind(0.15,0.3),
46                                  rbind(6.523,7.477),
47                                  rbind(10.5,14.5)
48                                 )
49
50 # Specify the sampling distributions.
51 # Either mean and s.d. or range limits for uniform.
52    p1_dist <- c(4.8000,0.2000)
53 # LOG10
54    p2_dist <- c(-3.293,0.1769)
55    p3_dist <- c(2.000,0.2000)
56    p4_dist <- c(-0.003250,0.0009375)
57 #    p4_dist_unif <- c(0.05*des_par_ran[2,4]+0.95*des_par_ran[1,4],0.95*des_par_ran
       [2,4]+0.05*des_par_ran[1,4])
58    p5_dist <- c(0.5250,0.03125)
59    #p5_dist <- c(0.450,0.080)
60    p6_dist <- c(0.0030,0.00010)
61    #p6_dist <- c(0.0025,0.00050)
62    p7_dist <- c(0.25,0.015)
63 # nlprop log10
64 #    p8_dist <- c(6.523,7.477)
65    p8_dist <- c(7,0.1193)
66 # Wind speed (normal).
67    p9_dist <- c(12.5,0.5)
68
69 # Re-scale the ranges.
70    func_log_rescale <- function(x,RA) { ( log10(x)-RA[1])/(RA[2]-RA[1]) }
71    func_lin_rescale <- function(x,RA) { ( ( x-RA[1] ) / (RA[2]-RA[1]) ) }
72    func_lin_rescale_sd <- function(x,RA) { x / (RA[2]-RA[1]) }
73
74 # Generate the sampling distributions.
75    m_var <- rbind(
76                c(func_lin_rescale(p1_dist[1],des_par_ran[,1]),func_lin_rescale_sd(p1
                    _dist[2],des_par_ran[,1])),
77                c(func_lin_rescale(p2_dist[1],des_par_ran[,2]),func_lin_rescale_sd(p2
                    _dist[2],des_par_ran[,2])),
78                c(func_lin_rescale(p3_dist[1],des_par_ran[,3]),func_lin_rescale_sd(p3
                    _dist[2],des_par_ran[,3])),
79 # p4
80                c(func_lin_rescale(p4_dist[1],des_par_ran[,4]),func_lin_rescale_sd(p4
                    _dist[2],des_par_ran[,4])),
81 #           runif(unc_MC_res,func_lin_rescale(p4_dist_unif[1],des_par_ran[,4]),func
```

```
    _lin_rescale(p4_dist_unif[2],des_par_ran[,4])),
82              c(func_lin_rescale(p5_dist[1],des_par_ran[,5]),func_lin_rescale_sd(p5
                    _dist[2],des_par_ran[,5])),
83              c(func_lin_rescale(p6_dist[1],des_par_ran[,6]),func_lin_rescale_sd(p6
                    _dist[2],des_par_ran[,6])),
84              c(func_lin_rescale(p7_dist[1],des_par_ran[,7]),func_lin_rescale_sd(p7
                    _dist[2],des_par_ran[,7])),
85              c(func_lin_rescale(p8_dist[1],des_par_ran[,8]),func_lin_rescale_sd(p8
                    _dist[2],des_par_ran[,8])),
86              c(func_lin_rescale(p9_dist[1],des_par_ran[,9]),func_lin_rescale_sd(p9
                    _dist[2],des_par_ran[,9]))
87             )
88
89 # The input uncertainty distribution is multivariate normal characterised by:
90 # mean m and covariance matrix B.
91    m <- m_var[,1]
92    vec_var <- m_var[,2]^2
93    B <- diag(1/vec_var, i_p, i_p)
94
95 # Gaussian correlation matrix.
96    C <- diag(1/delta^2, i_p, i_p)
97
98 # Integration control.
99    sub_divs <- 250
100   rel_tol <- 1e-7
101
102 #=========================================================================#
103 # Uncertainty analysis.
104 # We compute various uncertainty measures by integrating the emulator
105 # functions.
106 # Linear mean assumed, arbitrary correlation function.
107 #=========================================================================#
108 # Set up various functions.
109 #-------------------------------------------------------------------------#
110 # Prob density function.
111    func_w_x <- function(x,i)
112    {
113       func_in <- cbind( rep(x,length(i)) , rep(i,each=length(x)) )
114       apply( X=func_in,1,FUN=function(x) { ((2*3.141593)^-0.5)*sqrt(B[x[2],x[2]]) *
               exp(-0.5*(x[1]-m[x[2]]) * B[x[2],x[2]] * (x[1]-m[x[2]])) } )
115    }
116
117 #   func_w_x <- function(x,i)
118 #   {
119 #      ((2*3.141593)^-0.5)*sqrt(B[i,i]) * exp(-0.5*(x-m[i]) * B[i,i] * (x-m[i]))
120 #   }
121
122 #-------------------------------------------------------------------------#
123 # Gaussian correlation function, c(x,x_prime).
124 #   func_int_cx <- function(x2,x1,i,j)
125 #   {
126 #      exp( -( ( C[i,i] * ( as.vector(x1) - x2 )^2 ) ) ) *
127 #      ((2*3.141593)^-0.5)*sqrt(B[i,i]) * exp(-0.5*(x2-as.vector(m[i])) * B[i,i] *
    (x2-as.vector(m[i]))) *
128 #      ((2*3.141593)^-0.5)*sqrt(B[j,j]) * exp(-0.5*(x1-as.vector(m[j])) * B[j,j] *
    (x1-as.vector(m[j])))
129 #   }
```

```
# Gaussian correlation function.
#   func_tx <- function(x,i,k) { exp(-((x-D[k,i]) * C[i,i] * (x-D[k,i]))) }
# Gaussian "R_t" expression.
#   func_Rt <- function(x,i,k) { ((2*3.141593)^-0.5)*sqrt(B[i,i]) * exp(-0.5*(x-m[i
    ]) * B[i,i] * (x-m[i])) * exp(-((x-D[k,i]) * C[i,i] * (x-D[k,i]))) }
#-----------------------------------------------------------------------#
# Matern correlation function order 3/2, c(x,x_prime).
   DD <- sqrt(3) / delta
   func_int_cx <- function(x2,x1,i,j)
   {
      sapply(X=x2,FUN=function(x,x1) { prod( 1 + ( DD[i] * abs( as.vector(x1) - x )
           ) ) },x1=x1 ) *
      exp( -( DD[i] * abs( as.vector(x1) - x2 ) ) ) *
      ((2*3.141593)^-0.5)*sqrt(B[i,i]) * exp(-0.5*(x2-as.vector(m[i])) * B[i,i] * (
          x2-as.vector(m[i]))) *
      ((2*3.141593)^-0.5)*sqrt(B[j,j]) * exp(-0.5*(x1-as.vector(m[j])) * B[j,j] * (
          x1-as.vector(m[j])))
   }
# Matern correlation function, t(x).
   func_tx <- function(x,i,k) { (1+DD[i]*abs(x-D[k,i])) * exp(-DD[i]*abs(x-D[k,i]))
        }
# Matern "R_t" expression.
   func_Rt <- function(x,i,k) { ((2*3.141593)^-0.5)*sqrt(B[i,i]) * exp(-0.5*(x-m[i
       ]) * B[i,i] * (x-m[i])) * (1+DD[i]*abs(x-D[k,i])) * exp(-DD[i]*abs(x-D[k,i])
       ) }
#-----------------------------------------------------------------------#
# Matern correlation function order 5/2, c(x,x_prime).
#   DDa <- sqrt(5) / delta
#   DDb <- 5 / (3 * delta^2)
#   func_int_cx <- function(x2,x1,i,j)
#   {
#      sapply(X=x2,FUN=function(x,x1) { prod( 1 + ( DDa[i] * abs( as.vector(x1) - x
     ) ) + ( DDb[i] * abs( as.vector(x1) - x )^2 ) ) },x1=x1 ) *
#      exp( -( DDa[i] * abs( as.vector(x1) - x2 ) ) ) *
#      ((2*3.141593)^-0.5)*sqrt(B[i,i]) * exp(-0.5*(x2-as.vector(m[i])) * B[i,i] *
    (x2-as.vector(m[i]))) *
#      ((2*3.141593)^-0.5)*sqrt(B[j,j]) * exp(-0.5*(x1-as.vector(m[j])) * B[j,j] *
    (x1-as.vector(m[j])))
#   }
# Matern correlation function, t(x).
#   func_tx <- function(x,i,k) { ( 1 + DDa[i]*abs(x-D[k,i]) + DDb[i]*abs(x-D[k,i])
    ^2 ) * exp(-DDa[i]*abs(x-D[k,i])) }
# Matern "R_t" expression.
#   func_Rt <- function(x,i,k) { ((2*3.141593)^-0.5)*sqrt(B[i,i]) * exp(-0.5*(x-m[i
    ]) * B[i,i] * (x-m[i])) * ( 1 + DDa[i]*abs(x-D[k,i]) + DDb[i]*abs(x-D[k,i])^2 )
     * exp(-DDa[i]*abs(x-D[k,i])) }
#-----------------------------------------------------------------------#
   func_int_Rt <- function(l,m) { integrate(func_Rt,lower=-Inf,upper=Inf,i=l,k=m,
       subdivisions=sub_divs,rel.tol=rel_tol)$value }
# Gaussian "R_tt" expression.
   func_Rtta <- function(x,i,k,l) { func_Rt(x,i,k) * func_tx(x,i,l) }

#-----------------------------------------------------------------------#
# E = E*E[f(X)]] = t(R_h) %*% beta + t(R_t) %*% e
#-----------------------------------------------------------------------#
# R_h = E_x[ h(x) | m,B ]
# For linear mean this trivially reduces to the mean m.
```

```
#   R_h <- c(1,m)

# In the general case where there are higher order terms, we integrate numerically.
# This is tricky because we don't "know" which terms of x the mean function incorporates.
   func_h_w1 <- function(x,i,q) { h_x_val <- matrix(0,1,i_p); h_x_val[1,i] <- x; func_h(h_x_val)[q] * func_w_x(x,i) }
   func_h_w2 <- function(x,i,q) { sapply(X=x,FUN=func_h_w1,i=i,q=q) }
   func_h_indx <- function(i) { h_idx <- matrix(0,1,i_p); h_idx[1,i] <- 1; which( func_h(h_idx) == 1)[-1] }

   print(paste(" Computing R_h..."))

   R_h <- 1
   for (i in 1:i_p)
   {
      for (q in c(func_h_indx(i)))
      {
         R_h[q] <- integrate(func_h_w2,lower=-Inf,upper=Inf,i=i,q=q,subdivisions=1000,rel.tol=1e-9)$value
      }
   }

#-------------------------------------------------------------------------#
# R_t
#-------------------------------------------------------------------------#
   print(paste(" Computing R_t..."))

   R_t1_temp <- 0
   for (j in 1:i_n)
   {
      int_sum <- 0
      for (i in 1:i_p)
      {
         int_sum[i] <- func_int_Rt(i,j)
      }
      R_t1_temp[j] <- prod(int_sum)
#      R_t1[j] <- (1-nu) * prod(int_sum)
   }
   R_t1 <- (1-nu) * R_t1_temp

   E <- t(R_h) %*% beta + t(R_t1) %*% e
   print(paste("Emulator E         : ",format(E,digits=4)))

#-------------------------------------------------------------------------#
# U
#-------------------------------------------------------------------------#
   func_int_U_inner <- function(x,i,j) { integrate(func_int_cx,lower=-Inf,upper=Inf,x1=x,i=i,j=j,subdivisions=sub_divs,rel.tol=rel_tol)$value }

   func_U_inner1 <- function(x,i,j)
   {
      sapply(X=x,FUN=func_int_U_inner,i=i,j=j)
   }

   print(paste(" Computing U..."))

```

```
225 # U outer integral.
226    int_sum1 <- 0
227    for (j in 1:i_p)
228    {
229       ii <- 0
230       while(TRUE)
231       {
232          rel_tol1 <- rel_tol * 10^ii
233          int_test <- NULL
234          try(int_test <- integrate(func_U_inner1,lower=-Inf,upper=Inf,i=j,j=j,
                 subdivisions=sub_divs,rel.tol=rel_tol1)$value,silent=TRUE)
235          ii <- ii + 1
236
237          if (!is.null(int_test) ) break;
238       }
239       int_sum1[j] <- int_test
240       #int_sum1[j] <- integrate(func_U_inner1,lower=-Inf,upper=Inf,i=j,j=j,
              subdivisions=sub_divs,rel.tol=1e-6)$value
241       prod(int_sum1)
242    }
243    U1 <- (1-nu) * prod(int_sum1)
244
245 #-------------------------------------------------------------------------#
246 # E_Var_f(X) = (I1 - V) + (I2 - E)
247 # I1 = sigmasq * (U_tilda - sum(diag(invA %*% R_tt) + sum(diag(W %*% (R_hh - 2*R_ht
        %*% G + t(G) %*% R_tt %*% G)))))
248 # I2 = t(beta) %*% R_hh %*% beta + 2*t(beta) %*% R_ht %*% e + t(e) %*% R_tt %*% e
249 #-------------------------------------------------------------------------#
250 # R_hh = E_x[ h(x)t(h(x)) | m,B ]
251 # Results from Rh * Rh.
252 # Should be the same for linear mean, irrespective of correlation function?
253 #-------------------------------------------------------------------------#
254
255 # In the general case where there are higher order terms, we integrate numerically.
256 # A bit mucky!
257 # For i=j only a single integral required.
258    func_h1 <- function(x,i,q) { h_x_val <- matrix(0,1,i_p); h_x_val[1,i] <- x; func
          _h(h_x_val)[q] }
259    func_hh1a <- function(x,i,q,r) { func_h1(x,i,r) * func_h1(x,i,q) * func_w_x(x,i)
           }
260    func_hh1b <- function(x,i,q,r) { sapply(X=x,FUN=func_hh1a,i=i,q=q,r=r) }
261
262 # Some off-diagonals require integration over both respective dimensions of X so
       double integrals required.
263    func_hh_w <- function(x,x1,i,j,q,r) { func_h_w2(x,j,r) * func_h_w2(x1,i,q) }
264    func_int_hh_w1 <- function(x,i,j,q,r)
265    {
266       integrate( func_hh_w,lower=-Inf,upper=Inf,x1=x,i=i,j=j,q=q,r=r,subdivisions=
              sub_divs,rel.tol=rel_tol)$value
267    }
268    func_int_hh_w2 <- function(x,i,j,q,r) { sapply(X=x,FUN=func_int_hh_w1,i=i,j=j,q=
          q,r=r) }
269
270 # Create a temporary matrix.
271    R_hh2 <- matrix(0,i_q,i_q)
272
273    print(paste(" Computing R_hh..."))
```

```
274
275 # Matrix is symmetrical so compute half of it, then transpose and copy.
276    for (i in 1:i_p)
277    {
278       for (q in c(func_h_indx(i)))
279       {
280          for (j in 1:i_p)
281          {
282             for (r in c(func_h_indx(j)))
283             {
284                if (i==j && q > r)
285                {
286                   R_hh2[q,r] <- integrate(func_hh1b,lower=-Inf,upper=Inf,i=i,q=q,r=
                          r,subdivisions=sub_divs,rel.tol=rel_tol)$value
287                }
288                else if (i!=j && q > r)
289                {
290                   R_hh2[q,r] <- integrate(func_h_w2,lower=-Inf,upper=Inf,i=i,q=q,
                          subdivisions=sub_divs,rel.tol=rel_tol)$value *
291                                    integrate(func_h_w2,lower=-Inf,upper=Inf,i=j,q=r,
                                        subdivisions=sub_divs,rel.tol=rel_tol)$value
292                   #R_hh3b[q,r] <- integrate(func_int_hh_w2,lower=-Inf,upper=Inf,i=i
                          ,j=j,q=q,r=r,subdivisions=sub_divs,rel.tol=rel_tol)$value
293                }
294             }
295          }
296       }
297    }
298
299 # Transpose and copy.
300    R_hh1 <- R_hh2 + t(R_hh2)
301
302 # Then do the diagonal.
303    for (i in 1:i_p)
304    {
305       for (q in c(func_h_indx(i)))
306       {
307          R_hh1[q,q] <- integrate(func_hh1b,lower=-Inf,upper=Inf,i=i,q=q,r=q,
                subdivisions=sub_divs,rel.tol=rel_tol)$value
308       }
309    }
310
311 # Just copy R_h into the first column and first row to save computation.
312 # Note this only works if the first entry of func_h is 1.
313    R_hh1[,1] <- R_h
314    R_hh1[1,] <- R_h
315
316 #-----------------------------------------------------------------------#
317 # R_ht
318 # Results from Rh * Rt.
319 #-----------------------------------------------------------------------#
320 # Create the integrable functions.
321 # Very similar to func_h1 but returns 1 for non-active elements.
322    func_h2 <- function(x,q,i) { h_x_val <- matrix(1,1,i_p); h_x_val[1,i] <- x; func
          _h(h_x_val)[q] }
323    func_h2a <- function(x,q,i) { sapply(X=x,FUN=func_h2,i=i,q=q) }
324
```

```
325    func_R_ht1 <- function(x,q,i,x_k)
326    {
327       func_h2a(x,q,i) * func_Rt(x,i,x_k)
328    }
329
330 # This builds the "standard" R_ht for uncertainty analysis.
331 # So we integrate over all x, and the function is thus not of x or i_cond.
332 # q designates the row of the R_ht matrix.
333    func_int_R_ht1 <- function(q,x_k)
334    {
335       int_sum <- 0
336       for (i in 1:i_p)
337       {
338          int_sum[i] <- integrate(func_R_ht1,lower=-Inf,upper=Inf,q=q,i=i,x_k=x_k,
                subdivisions=sub_divs,rel.tol=rel_tol)$value
339       }
340       prod(int_sum)
341    }
342
343    print(paste(" Computing R_ht..."))
344
345    R_ht1 <- matrix(0,i_q,i_n)
346    for (a in 1:i_n)
347    {
348       for (b in 1:i_q)
349       {
350          R_ht1[b,a] <- func_int_R_ht1(b,a)
351       }
352    }
353
354 #-------------------------------------------------------------------------#
355 # R_tt
356 # Square matrix arising from R_t * R_t.
357 # Function of x_k and x_l.
358 #-------------------------------------------------------------------------#
359 # Create an array of integratations for later use.
360    array_int_Rtt <- array(0,dim=c(i_n,i_n,i_p))
361
362    print(paste(" Computing R_tt..."))
363
364    for (a in 1:i_n)
365    {
366    print(paste(" R_tt integration col:",a))
367       for (b in 1:i_n)
368       {
369          for (i in 1:i_p)
370          {
371             ii <- 1
372             while(TRUE)
373             {
374                rel_tol1 <- rel_tol * 10^ii
375                int_test <- NULL
376                try(int_test <- integrate(func_Rtta,lower=-Inf,upper=Inf,i=i,k=b,l=a
                      ,subdivisions=sub_divs,rel.tol=rel_tol1)$value,silent=TRUE)
377                ii <- ii + 1
378
379                if (!is.null(int_test) ) break;
```

```
380             }
381
382             array_int_Rtt[b,a,i] <- int_test
383             #array_int_Rtt[b,a,i] <- integrate(func_Rtta,lower=-Inf,upper=Inf,i=i,k
                    =b,l=a,subdivisions=sub_divs,rel.tol=1e-3)$value
384          }
385       }
386    }
387
388    R_tt1_temp <- matrix(0,i_n,i_n)
389    for (a in 1:i_n)
390    {
391       for (b in 1:i_n)
392       {
393          R_tt1_temp[b,a] <- prod(array_int_Rtt[b,a,])
394       }
395    }
396    R_tt1 <- (1-nu)^2 * R_tt1_temp
397
398 #-------------------------------------------------------------------------#
399 # Complete integrals.
400 #-------------------------------------------------------------------------#
401 # U_tilda
402    U_tilda <- 1
403
404 # V
405    V <- sigmasq[emulator_index] * (U1 - t(R_t1) %*% invA %*% R_t1 + t(R_h - t(G) %*
           % R_t1) %*% W %*% (R_h - t(G) %*% R_t1))
406    print(paste("Emulator variance V         : ",format(V,digits=4)))
407 # I1
408    I1 <- sigmasq[emulator_index] * (U_tilda - sum(diag(invA %*% R_tt1)) + sum(diag(
           W %*% (R_hh1 - 2*R_ht1 %*% G + t(G) %*% R_tt1 %*% G))))
409 # I2
410    I2 <- t(beta) %*% R_hh1 %*% beta + 2*t(beta) %*% R_ht1 %*% e + t(e) %*% R_tt1 %*
           % e
411
412    E_Var_fX <- (I1 - V) + (I2 - E^2)
413 # Print outs.
414    print("-----------------------------------------")
415    print(paste("E*[ E[f(X)] ]          = E : ", format(E,digits=5)))
416    print(paste("E*[ Var[f(X)] ] = E_Var_fX :  ", format(E_Var_fX,digits=5)," (s.d.
           = ",format(sqrt(E_Var_fX),digits=3),")",sep=""))
417
418 #=========================================================================#
419 # Sensitivity analysis.
420 # Linear mean assumed, arbitrary correlation function.
421 #=========================================================================#
422 # Interactions.
423 #-------------------------------------------------------------------------#
424 # This function generates the elements of a vector for
425 # the mean function, assuming linear mean function.
426    func_R_h1 <- function(xx,ii)
427    {
428       temp_mat <- t(matrix(rep(R_h,length(xx)),i_q,length(xx)))
429       temp_mat[,func_h_indx(ii)] <- t(apply(matrix(xx,nrow=length(xx),ncol=i_p),1,
              func_h))[,func_h_indx(ii)]
430       temp_mat
```

```
   }

# Function for the conditional expectation of the mean.
# From this we can derive the main effects.
# We create functions depending on which order interactions,
# because the code gets complicated!
# x must be a matrix of n*length(i_cond), where x is
# dimension length(i_cond), with n entries.
#---------------------------------------------------------------------#
# Main effects.
#---------------------------------------------------------------------#
# Here, x and i_cond must be 2D.
# R_h2 is modified for 2 conditions (might work for more but untested yet).
# Note that xx input must be formatted with the first "column" of xx being values
    for the first conditional dimension.
# Second "column" for the second. I.e. not alternating.
   func_R_h2 <- function(xx,ii)
   {
      n_x_vals <- length(xx)/length(ii)
      temp_mat <- t(matrix(R_h,nrow=i_q,ncol=n_x_vals))
      loc_count <- 1
      for (jj in ii)
      {
         h_x_val <- matrix(0,nrow=n_x_vals,ncol=i_p)
         h_x_val[,jj] <- matrix(xx,nrow=n_x_vals)[,loc_count]
         temp_mat[,func_h_indx(jj)] <- t(apply(h_x_val,1,func_h))[,func_h_indx(jj)]
         loc_count <- loc_count + 1
      }
      temp_mat
   }

# Pre-compute R_t integral to save time.
      R_t_table <- matrix(0,nrow=i_n,ncol=i_p)
      for (k in 1:i_n)
      {
         R_t_table[k,] <- sapply(X=seq_dim,func_int_Rt,m=k)
      }

   func_E_cond_1 <- function(x,i_cond)
   {
      n_x_vals <- length(x)
      mat_R_t2a <- matrix(0,n_x_vals,i_n)
      int_sum <- array(0,dim=c(i_n,i_p,n_x_vals))
      for (k in 1:i_n)
      {
         int_sum[k,i_cond,] <- func_tx(x,i_cond,k)

         int_sum[k,seq_dim[-i_cond],] <- R_t_table[k,seq_dim[-i_cond]]
         mat_R_t2a[,k] <- apply(t(int_sum[k,,]),1,prod)
      }
      R_t2e <- apply(mat_R_t2a,1,function(xx) { xx %*% e })
      R_h2b <- func_R_h2(x,i_cond) %*% beta
      R_h2b + R_t2e
   }

#---------------------------------------------------------------------#
# Sensitivity variances for single conditional variable.
```

```
# i) Hora & Imam importance measure.
#    Equivalently MUCM sensitivity variance.
#    E*[Vw] = E*[E[E[f(X)|Xw]^2]] - E*[E[f(X)]^2]
#------------------------------------------------------------------------#
# We create conditions for U_cond, R_hh_cond, R_ht_cond and R_tt_cond.
# (Not the erroneous emulator mean function, because this omits the
# emulator variance.)
#------------------------------------------------------------------------#

#------------------------------------------------------------------------#
# U_cond
# The conditional dimensions simply integrate to unit.
# This function can also be used for multiple conditionals.
#------------------------------------------------------------------------#
# U outer integral.
   func_U_cond <- function(i_cond)
   {
      int_sum1 <- 0
      for (j in c(1:i_p)[-i_cond])
      {
         ii <- 0
         while(TRUE)
         {
            rel_tol1 <- rel_tol * 10^ii
            int_test <- NULL
            try(int_test <- integrate(func_U_inner1,lower=-Inf,upper=Inf,i=j,j=j,
                subdivisions=sub_divs,rel.tol=rel_tol1)$value,silent=TRUE)
            ii <- ii + 1

            if (!is.null(int_test) ) break;
         }
         int_sum1[j] <- int_test
         #int_sum1[j] <- integrate(func_U_inner1,lower=-Inf,upper=Inf,i=j,j=j,
             subdivisions=sub_divs,rel.tol=rel_tol)$value
      }
      int_sum1[i_cond] <- 1
      (1-nu) * prod(int_sum1)
   }

#------------------------------------------------------------------------#
# R_hh_cond = E_x[ h(x)t(h(x)) | m,B,x1 ]
# Conditional terms only get integrated once!
# Remaining terms, even if they are the "same" index are integrated twice
# since this is a double integral.
#------------------------------------------------------------------------#
   func_R_hh_cond <- function(i_cond)
   {
# Create a temporary matrix.
      R_hh2_cond <- matrix(0,i_q,i_q)

# Matrix is symmetrical so compute half of it, then transpose and copy.
      for (i in 1:i_p)
      {
         for (q in c(func_h_indx(i)))
         {
            for (j in 1:i_p)
            {
```

```
542               for (r in c(func_h_indx(j)))
543               {
544                  if (i==i_cond && i==j && q > r)
545                  {
546                     R_hh2_cond[q,r] <- integrate(func_hh1b,lower=-Inf,upper=Inf,i=
                           i,q=q,r=r,subdivisions=sub_divs,rel.tol=rel_tol)$value
547                     #R_hh2_cond[q,r] <- integrate(func_h_w2,lower=-Inf,upper=Inf,i
                           =i,q=q,subdivisions=sub_divs,rel.tol=rel_tol)$value *
548                     #              integrate(func_h_w2,lower=-Inf,upper=Inf,i=j,q=
                           r,subdivisions=sub_divs,rel.tol=rel_tol)$value
549                  }
550                  else if (q > r)
551                  {
552                     R_hh2_cond[q,r] <- integrate(func_h_w2,lower=-Inf,upper=Inf,i=
                           i,q=q,subdivisions=sub_divs,rel.tol=rel_tol)$value *
553                                        integrate(func_h_w2,lower=-Inf,upper=Inf,i=
                                            j,q=r,subdivisions=sub_divs,rel.tol=rel
                                            _tol)$value
554                  }
555               }
556            }
557         }
558      }
559
560 # Transpose and copy.
561      R_hh_cond <- R_hh2_cond + t(R_hh2_cond)
562
563 # Then do the diagonal.
564      for (i in 1:i_p)
565      {
566         for (q in c(func_h_indx(i)))
567         {
568            if (i==i_cond)
569            {
570               R_hh_cond[q,q] <- integrate(func_hh1b,lower=-Inf,upper=Inf,i=i,q=q,r
                     =q,subdivisions=sub_divs,rel.tol=rel_tol)$value
571            } else {
572               R_hh_cond[q,q] <- integrate(func_h_w2,lower=-Inf,upper=Inf,i=i,q=q,
                     subdivisions=sub_divs,rel.tol=rel_tol)$value *
573                                 integrate(func_h_w2,lower=-Inf,upper=Inf,i=i,q=q,
                                     subdivisions=sub_divs,rel.tol=rel_tol)$value
574            }
575         }
576      }
577
578 # Just copy R_h into the first column and first row to save computation.
579 # Note! This only works if the first entry of func_h is 1.
580      R_hh_cond[,1] <- R_h
581      R_hh_cond[1,] <- R_h
582      R_hh_cond
583   }
584
585 #-----------------------------------------------------------------------#
586 # R_ht_cond
587 #
588 # The order of integration is tricky. If q does not correspond to the conditional
      dimension,
```

```
589 # then the element of R_ht_cond is essentially a double integral, one over h(x) and
        one
590 # over t(x).
591 #-------------------------------------------------------------------------#
592    func_int_R_ht_cond1 <- function(x,q,x_k,i_cond)
593    {
594       if ( is.element(q,func_h_indx(i_cond)) )
595       {
596          int_sum <- prod(R_t_table[x_k,-i_cond])
597          int_sum_cond <- func_R_ht1(x=x,q=q,i=i_cond,x_k=x_k)
598       } else {
599          int_sum <- R_h[q] * prod(R_t_table[x_k,-i_cond])
600          int_sum_cond <- func_Rt(x=x,i=i_cond,k=x_k)
601       }
602       prod(int_sum) * int_sum_cond
603    }
604
605 # Wrapper for second integration.
606    func_R_ht_cond1 <- function (x,q,x_k,i_cond)
607    {
608       sapply(X=x,FUN=func_int_R_ht_cond1,q=q,x_k=x_k,i_cond=i_cond)
609    }
610
611 # Do the integration.
612    func_int_R_ht_cond2 <- function (q,x_k,i_cond)
613    {
614       integrate(func_R_ht_cond1,lower=-Inf,upper=Inf,q=q,x_k=x_k,i_cond=i_cond,
              subdivisions=sub_divs,rel.tol=rel_tol)$value
615    }
616
617 # Function to create the matrix.
618    func_R_ht_cond <- function(i_cond)
619    {
620       R_ht1 <- matrix(0,i_q,i_n)
621       for (a in 1:i_n)
622       {
623 #      print(paste(" R_ht_cond col:",a))
624          for (b in 1:i_q)
625          {
626             R_ht1[b,a] <- func_int_R_ht_cond2(q=b,x_k=a,i_cond)
627          }
628       }
629    R_ht1
630    }
631
632 #-------------------------------------------------------------------------#
633 # R_tt_cond
634 # R_tt conditional on i_cond...
635 # Function of x_k and x_l.
636 #-------------------------------------------------------------------------#
637    func_int_Rtt_cond1 <- function (x,x_k,x_l,i_cond)
638    {
639 # Use pre-computed integrals.
640       int_sum <- R_t_table[x_k,] * R_t_table[x_l,]
641       int_sum[i_cond] <- func_Rtta(x=x,i=i_cond,k=x_k,l=x_l)
642       prod(int_sum)
643    }
```

```
644
645 # Wrapper to integrate over the conditional.
646    func_Rtt_cond1 <- function (x,x_k,x_l,i_cond)
647    {
648       sapply(X=x,FUN=func_int_Rtt_cond1,x_k=x_k,x_l=x_l,i_cond=i_cond)
649    }
650
651 # Do the integration.
652    func_int_Rtt_cond2 <- function (x_k,x_l,i_cond)
653    {
654       integrate(func_Rtt_cond1,lower=-Inf,upper=Inf,x_k=x_k,x_l=x_l,i_cond=i_cond,
           subdivisions=200,rel.tol=1e-7)$value
655    }
656
657 # Now create the matrix.
658 #   print(paste(" Computing R_tt_cond..."))
659
660    func_R_tt_cond <- function(i_cond)
661    {
662       R_tt_cond_temp <- matrix(0,i_n,i_n)
663       for (a in 1:i_n)
664       {
665 #         print(paste(" R_tt_cond col:",a))
666          for (b in 1:i_n)
667          {
668             R_tt_cond_temp[b,a] <- func_int_Rtt_cond2(b,a,i_cond)
669          }
670       }
671       (1-nu)^2 * R_tt_cond_temp
672    }
673
674 #-------------------------------------------------------------------------#
675 # Compute:
676 #    E*[Vw] = E*[E[E[f(X)|Xw]^2]] - E*[E[f(X)]^2]
677 #-------------------------------------------------------------------------#
678 # E*[E[f(X)]^2]
679 #-------------------------------------------------------------------------#
680    E_fX_2 <- V + E^2
681 #-------------------------------------------------------------------------#
682 # For E*[E[E[f(X)|Xw]^2]] we loop over each dimension to get each sensitivity
       variance.
683 # This method is much slower than dealing with the emulator mean and covariance
       functions
684 # directly (see below).
685 #-------------------------------------------------------------------------#
686 #   E_E_fX_w2 <- 0
687 #   U_cond <- 0
688 #   array_R_hh_cond <- array(0,dim=c(i_q,i_q,i_p))
689 #   array_R_ht_cond <- array(0,dim=c(i_q,i_n,i_p))
690 #   array_R_tt_cond <- array(0,dim=c(i_n,i_n,i_p))
691 #   sen_var1 <- 0
692 #
693 #   print("-------------------------------------------")
694 #   print("Sensitivity variances")
695 #
696 #   for (i in 1:i_p)
697 #   {
```

```
## U_cond.
#      U_cond[i] <- func_U_cond(i)
## R_hh_cond.
#      print(paste(" Computing R_hh_cond #",i,"..."))
#      array_R_hh_cond[,,i] <- func_R_hh_cond(i)
## R_ht_cond.
#      print(paste(" Computing R_ht_cond #",i,"..."))
#      array_R_ht_cond[,,i] <- func_R_ht_cond(i)
## R_tt_cond.
#      print(paste(" Computing R_tt_cond #",i,"..."))
#      array_R_tt_cond[,,i] <- func_R_tt_cond(i)
## E_E_fX_w2.
#      E_E_fX_w2[i] <- sigmasq[emulator_index] *
#                      (
#                        U_cond[i] -
#                        sum(diag(invA %*% array_R_tt_cond[,,i])) +
#                        sum(diag(W %*% (array_R_hh_cond[,,i] - 2*array_R_ht_cond
    [,,i] %*% G + t(G) %*% array_R_tt_cond[,,i] %*% G)))
#                      ) +
#                        t(e) %*% array_R_tt_cond[,,i] %*% e + 2*t(beta) %*% array_
    R_ht_cond[,,i] %*% e + t(beta) %*% array_R_hh_cond[,,i] %*% beta
#
#      sen_var1[i] <- E_E_fX_w2[i] - E_fX_2
#      print(paste("Variable ",i,"     : ",format(sen_var1[i],digits=5)," (",format
    (100 * sen_var1[i]/E_Var_fX,digits=3),"% )"),sep="")
#   }
#
#-------------------------------------------------------------------------#
# Print summary.
#-------------------------------------------------------------------------#
#   for (i in 1:i_p)
#   {
#      print(paste("Variable ",i,"     : ",format(sen_var1[i],digits=5)," (",format
    (100 * sen_var1[i]/E_Var_fX,digits=3),"% )"),sep="")
#   }
#   sen_var_unexp1 <- E_Var_fX - sum(sen_var1)
#   print("-----------------------------------------")
#   print("Unexplained variance")
#   print(
#           paste("E*[ Var[f(X)] ] - sum(sensitivity variances)  : ",
#           format(sen_var_unexp1,digits=3),
#           " (",
#           format(100 * sen_var_unexp1/E_Var_fX,digits=3),
#           "% )",
#           sep="")
#         )
#
#=========================================================================#
# Alternative.
# Here we integrate over the margins, except the conditional(s).
# This function is over one marign.
#=========================================================================#
   func_mean_cond <- function(x,i_cond)
   {
      n_x_vals <- length(x)
      mat_R_t2a <- matrix(0,n_x_vals,i_n)
      int_sum <- array(0,dim=c(i_n,i_p,n_x_vals))
```

```
751       for (k in 1:i_n)
752       {
753          int_sum[k,i_cond,] <- func_tx(x,i_cond,k)
754
755          int_sum[k,seq_dim[-i_cond],] <- R_t_table[k,seq_dim[-i_cond]]
756          mat_R_t2a[,k] <- apply(t(int_sum[k,,]),1,prod)
757       }
758       R_t2e <- apply(mat_R_t2a,1,function(xx) { xx %*% e })
759       R_h2b <- func_R_h2(x,i_cond) %*% beta
760       (R_h2b + R_t2e)^2
761    }
762
763 #-----------------------------------------------------------------------#
764 # Integration over the "square" of the covariance function.
765 #-----------------------------------------------------------------------#
766
767 # Functions for first and second terms.
768    func_Rt_invA_Rt <- function(xx) { sum(diag( invA %*% (xx %*% t(xx)) )) }
769    func_cov_p2 <- function(xx)
770                   {
771                      Rh_p2 <- xx[(i_n+1):(i_n+i_q)]
772                      Rt_p2 <- xx[1:i_n]
773                      sum(diag( W %*% ( (Rh_p2 - t(G) %*% Rt_p2) %*% t(Rh_p2 - t(G)
                           %*% Rt_p2) ) ) )
774                   }
775 # Covariance function.
776    func_cov_cond <- function(x,i_cond)
777    {
778       n_x_vals <- length(x)
779       mat_R_t2a <- matrix(0,n_x_vals,i_n)
780       int_sum <- array(0,dim=c(i_n,i_p,n_x_vals))
781       for (k in 1:i_n)
782       {
783          int_sum[k,i_cond,] <- func_tx(x,i_cond,k)
784
785          int_sum[k,seq_dim[-i_cond],] <- R_t_table[k,seq_dim[-i_cond]]
786          mat_R_t2a[,k] <- apply(t(int_sum[k,,]),1,prod)
787       }
788 # Compute part 1.
789       cov_p1 <- apply(mat_R_t2a,1,func_Rt_invA_Rt)
790 # Compute part 2.
791       cov_p2 <- apply(X=cbind(mat_R_t2a,func_R_h2(x,i_cond)),1,FUN=func_cov_p2 )
792 # Add together.
793       cov_p1 + cov_p2
794    }
795
796 # Create a function that can be integrated over each margin to obtain the
       sensitivity variance.
797    func_mean_cond_w <- function(x,i_cond) { func_mean_cond(x,i_cond) * func_w_x(x,i
          _cond) }
798    func_int_mean_cond <- function(i_cond) { integrate(func_mean_cond_w,lower=-Inf,
          upper=Inf,i_cond=i_cond,subdivisions=sub_divs,rel.tol=rel_tol)$value }
799    func_cov_cond_w <- function(x,i_cond) { func_cov_cond(x,i_cond) * func_w_x(x,i_
          cond) }
800    func_int_cov_cond <- function(i_cond) { integrate(func_cov_cond_w,lower=-Inf,
          upper=Inf,i_cond=i_cond,subdivisions=sub_divs,rel.tol=rel_tol)$value }
801
```

```
802 #------------------------------------------------------------------------#
803 # Qucik plot of main effects.
804 #------------------------------------------------------------------------#
805    line_cols <- c(22,30,31,32,33,258,50,34,47,652,67,89,102,104)
806    X11()
807    par(oma=c(0,0,0,0),mgp=c(3,1,0),mar=c(6,6,6,2)+0.1)
808    plot( NULL,
809          xlim=c(0,1),ylim=c(-0.15,0.15),
810          main=expression(Main~effects~of~uncertain~parameters~on~fetch~coefficient~
                 H[infinity]),
811          xlab=expression(X[w]),ylab=c("Main effect"),
812          cex.main=1.8,cex.lab=1.5,cex.axis=1.3 )
813
814    legend(x=0.1, y=-0.005, legend=param_labels, lty=c(1:6), lwd=c(rep(2.5,6)), col=
           c(colours()[line_cols[1:8]],"black"), title=expression(WW3~parameter), ncol
           =1, title.adj=0.1, bty="n",cex=1.6, y.intersp=0.6)
815
816    for (i in 1:i_p)
817    {
818       lines(seq(0,1,,100),(func_E_cond_1(seq(0,1,,100),i) / as.vector(E)) - 1,lty=c
              (i),lwd=2.5,col=colours()[line_cols[i]])
819    }
820
821 #------------------------------------------------------------------------#
822 # Generate sensitivity variances.
823 #------------------------------------------------------------------------#
824    print("-------------------------------------------")
825    print("Sensitivity variances")
826
827    E_E_fX_w2a <- 0
828    sens_var2 <- 0
829    for (i in 1:i_p)
830    {
831       E_E_fX_w2a[i] <- sigmasq[emulator_index] *
832                        ( func_U_cond(i) - func_int_cov_cond(i) ) + func_int_mean_
                             cond(i)
833
834       sens_var2[i] <- E_E_fX_w2a[i] - E_fX_2
835       print(paste("Variable ",i,"     : ",format(sens_var2[i],digits=5)," (",format
              (100 * sens_var2[i]/E_Var_fX,digits=3),"% )"),sep="")
836    }
837    sen_var_unexp2 <- E_Var_fX - sum(sens_var2)
838    print("-------------------------------------------")
839    print("Unexplained variance")
840    print(
841           paste("E*[ Var[f(X)] ] - sum(sensitivity variances)  : ",
842           format(sen_var_unexp2,digits=3),
843           " (",
844           format(100 * sen_var_unexp2/E_Var_fX,digits=3),
845           "% )",
846           sep="")
847         )
848
849    ABC <- barplot( 100*sens_var2/E_Var_fX, plot=FALSE )
850    X11()
851    par(oma=c(0,0,0,0),mgp=c(3,0.25,0),mar=c(6,6,6,2)+0.1)
852    barplot( 100*sens_var2/E_Var_fX,
```

```
853            main=expression(Percentage~variance~explained),
854            horiz=FALSE,
855 #           names.arg=seq(1,i_p),
856            xlab=c("Input parameter"),
857            ylab=c("Percentage"),
858            col=colors()[line_cols],
859            border=NA,
860            cex.main=1.8,cex.lab=1.5,cex.axis=1.3,cex.names=1.3 )
861    axis(1, at=ABC, labels=param_labels, cex.axis=1.2, tick=FALSE, las=2)
862
863 #=========================================================================#
864 # Sensivity variances for 1st order interacting variables.
865 #=========================================================================#
866 # We create a function that can be integrated over each margin to obtain the
        sensitivity variance for interacting variables.
867 # Note this works only for the interaction of 2 conditions.
868 #-------------------------------------------------------------------------#
869 # Function for conditional mean for two dimensions.
870 #-------------------------------------------------------------------------#
871 # Function for the conditional mean of h(x). x must be length 2.
872    func_R_h2a <- function(x,i_cond)
873    {
874       R_h_cond <- R_h
875       h_x_val <- vector("double",length=i_p)
876       h_x_val[i_cond] <- x
877       R_h_cond[func_h_indx(i_cond)] <- func_h(h_x_val)[func_h_indx(i_cond)]
878       R_h_cond
879    }
880
881 # Function for integration later. x must be length 2.
882    func_E_cond2_w <- function(x,i_cond)
883    {
884       mat_R_t2a <- 0
885       int_sum <- matrix(0,nrow=i_n,ncol=i_p)
886       for (k in 1:i_n)
887       {
888          int_sum[k,i_cond[1]] <- func_tx(x[1],i_cond[1],k)
889          int_sum[k,i_cond[2]] <- func_tx(x[2],i_cond[2],k)
890
891 #         int_sum[k,seq_dim[-i_cond]] <- sapply(X=seq_dim[-i_cond],func_int_Rt,m=k)
892          int_sum[k,seq_dim[-i_cond]] <- R_t_table[k,seq_dim[-i_cond]]
893          mat_R_t2a[k] <- prod(int_sum[k,])
894       }
895       R_t2e <- mat_R_t2a %*% e
896       R_h2b <- func_R_h2a(x,i_cond) %*% beta
897       (R_h2b + R_t2e)^2 * func_w_x(x[1],i_cond[1]) * func_w_x(x[2],i_cond[2])
898    }
899
900 # Integrable function over x1. i_cond is length 2.
901    func_E_cond2_w_vec1 <- function(x1,x2,i_cond)
902    {
903       X2 <- cbind(x1,rep(x2,length(x1)))
904       apply(X2,1,func_E_cond2_w,i_cond=i_cond)
905    }
906 # Then integrate.
907    func_E_cond2_w_int1 <- function(x,i_cond)
908    {
```

```
909       integrate(func_E_cond2_w_vec1,lower=-Inf,upper=Inf,x2=x,i_cond=i_cond,
              subdivisions=sub_divs,rel.tol=rel_tol)$value
910    }
911 # Integrable function over x1. i_cond is length 2.
912    func_E_cond2_w_vec2 <- function(x,i_cond)
913    {
914       sapply(X=x,FUN=func_E_cond2_w_int1,i_cond=i_cond)
915    }
916 
917 #------------------------------------------------------------------------#
918 # Function for conditional covariance for two dimensions.
919 #------------------------------------------------------------------------#
920 # Covariance function.
921    func_cov_cond2_w <- function(x,i_cond)
922    {
923       mat_R_t2a <- 0
924       int_sum <- matrix(0,nrow=i_n,ncol=i_p)
925       for (k in 1:i_n)
926       {
927          int_sum[k,i_cond[1]] <- func_tx(x[1],i_cond[1],k)
928          int_sum[k,i_cond[2]] <- func_tx(x[2],i_cond[2],k)
929 
930          int_sum[k,seq_dim[-i_cond]] <- R_t_table[k,seq_dim[-i_cond]]
931          mat_R_t2a[k] <- prod(int_sum[k,])
932       }
933 # Compute part 1.
934       cov_p1 <- func_Rt_invA_Rt(mat_R_t2a)
935 # Compute part 2.
936       cov_p2 <- func_cov_p2(c(mat_R_t2a,func_R_h2a(x,i_cond)))
937 # Add together.
938       (cov_p1 + cov_p2) * func_w_x(x[1],i_cond[1]) * func_w_x(x[2],i_cond[2])
939    }
940 
941 # Integrable function over x1. i_cond is length 2.
942    func_cov_cond2_w_vec1 <- function(x1,x2,i_cond)
943    {
944       X2 <- cbind(x1,rep(x2,length(x1)))
945       apply(X2,1,func_cov_cond2_w,i_cond=i_cond)
946    }
947 # Then integrate.
948    func_cov_cond2_w_int1 <- function(x,i_cond)
949    {
950       integrate(func_cov_cond2_w_vec1,lower=-Inf,upper=Inf,x2=x,i_cond=i_cond,
              subdivisions=sub_divs,rel.tol=1e-5)$value
951    }
952 # Integrable function over x1. i_cond is length 2.
953    func_cov_cond2_w_vec2 <- function(x,i_cond)
954    {
955       sapply(X=x,FUN=func_cov_cond2_w_int1,i_cond=i_cond)
956    }
957 
958 #========================================================================#
959 # Find one interaction.
960 #------------------------------------------------------------------------#
961 #   i_w <- c(2,3)
962 #    <- integrate(func_E_cond2_w_vec2,lower=-Inf,upper=Inf,i_cond=i_w,subdivisions=
    sub_divs,rel.tol=rel_tol)$value
```

```
963 #   print(paste("Variables",i_w[1],"&",i_w[2],":", format(int_cond2a - E^2,digits
        =5)))
964 #   print("-------------------------------------------")
965 #=========================================================================#
966 # Loop over all 1st order interactions.
967 # Pretty slow, requires hours.
968 #-------------------------------------------------------------------------#
969 # Integrate over the "outer" conditional to obtain the complete integral.
970    int_E_cond2 <- matrix(0,i_p,i_p)
971    int_cov_cond2 <- matrix(0,i_p,i_p)
972    iw_list <- seq_dim
973 #   iw_list <- c(7,8,9)
974    for (i_w1 in 1:(length(iw_list)-1))
975    {
976       for (i_w2 in (i_w1+1):length(iw_list))
977       {
978          i_w <- c(iw_list[i_w1],iw_list[i_w2])
979          if (int_cov_cond2[i_w[1],i_w[2]] == 0)
980          {
981             int_E_cond2[i_w[1],i_w[2]] <- integrate(func_E_cond2_w_vec2,lower=-Inf,
                   upper=Inf,i_cond=i_w,subdivisions=sub_divs,rel.tol=1e-5)$value
982             int_cov_cond2[i_w[1],i_w[2]] <- integrate(func_cov_cond2_w_vec2,lower=-
                   Inf,upper=Inf,i_cond=i_w,subdivisions=sub_divs,rel.tol=1e-5)$value
983             print(paste("Variables",i_w[1],"&",i_w[2],"(est.):", format(int_E_cond2
                   [i_w[1],i_w[2]] - E^2,digits=3)))
984             print("-------------------------------------------")
985          }
986       }
987    }
988
989 # Write tables.
990 #   write.table(int_E_cond2, file = "interaction_tables/int_E_cond2_1.txt", append
       = FALSE)
991 #   write.table(int_cov_cond2, file = "interaction_tables/int_cov_cond2_1.txt",
       append = FALSE)
992
993    E_E_fX_inter <- matrix(0,i_p,i_p)
994    E_V_w <- matrix(0,i_p,i_p)
995    var_inter <- matrix(0,i_p,i_p)
996
997    print("-------------------------------------------")
998    print("First order interaction variances.")
999    for (i_w1 in 1:(length(iw_list)-1))
1000   {
1001      for (i_w2 in (i_w1+1):length(iw_list))
1002      {
1003         i_w <- c(iw_list[i_w1],iw_list[i_w2])
1004         E_E_fX_inter[i_w[1],i_w[2]] <- sigmasq[emulator_index] * ( func_U_cond(i_w
               ) - int_cov_cond2[i_w[1],i_w[2]] ) + int_E_cond2[i_w[1],i_w[2]]
1005         E_V_w[i_w[1],i_w[2]] <- E_E_fX_inter[i_w[1],i_w[2]] - E_fX_2
1006         var_inter[i_w[1],i_w[2]] <- E_V_w[i_w[1],i_w[2]] - sens_var2[i_w[1]] -
               sens_var2[i_w[2]]
1007         print(
1008               paste("E*[ Var_Tw ] ",
1009               i_w[1]," & ",i_w[2]," : ",
1010               format(var_inter[i_w[1],i_w[2]],digits=3),
1011               " ( ",
```

```
1012                format( (100 / E_Var_fX) * var_inter[i_w[1],i_w[2]],digits=3),
1013                "% )",
1014                sep="")
1015               )
1016       }
1017    }
1018    print("-------------------------------------------")
1019    print("Total variance explained by interactions.")
1020    print(paste("Sum( E*[ Var_Tw ] ) : ",format((100 / E_Var_fX) * sum(var_inter),
            digits=3),"%"))
```

../code/auto_unc_sens_analysis_v1.2.R

E Wavewath configuration details

Switch listings. See Tolman (2009a) (section 5.4 p.122) for full details.

Switch configuration used in chapter 4.

F90 SHRD NOGRB LRB4 XYG PR3 FLX2 LN1 ST2 STAB2 NL1 BT0 DB0 TR0 BS0 XX0 WNT1 WNX1 CRT1 CRX1

Switch configuration used in chapter 7.

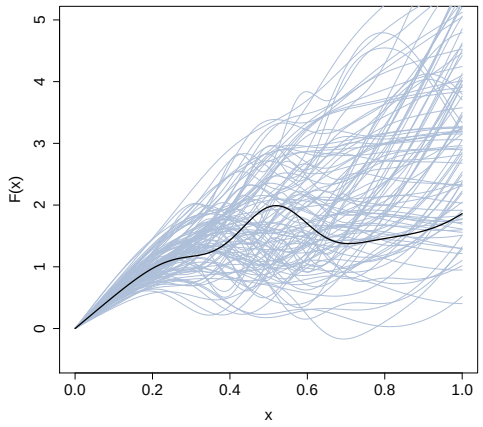
LLG PR3 FLX2 SEED ST2 STAB2 NL1 BT1 DB0 TR0 BS0 XX0 WNX1 WNT1 CRX1 CRT1 O0 O1 O2 O3 O4 O5 O6 O7 F90 NOGRB SHRD LRB4

F A possible approach to designing winds from observational data: A synthetic example.

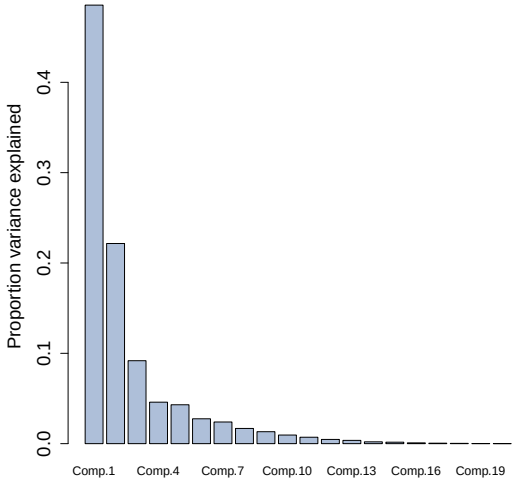
We begin by making use of a modified version of the "toy model" employed in section 3.5.5. The explicit form of the model is,

$$y(x|a,b,c,d) = 2.5x\left(1 + 0.7cos(dx)\right) + ax + \left\{0.4 * exp\left(\frac{-(x-b)}{0.2^2}\right) * sin(cx)\right\} \tag{F.1}$$

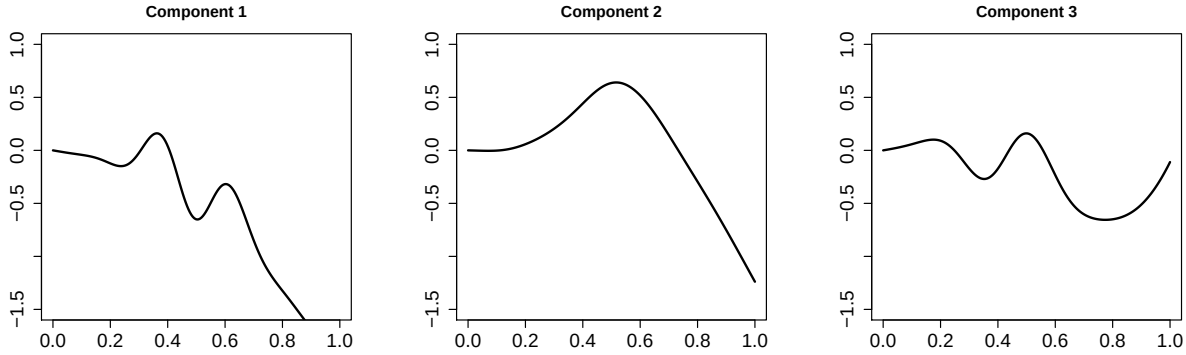
A sample of responses y for $0 < x < 1$ is generated by sampling a, b, c, d from a joint Gaussian distribution given by $[a, b, c, d]^{\mathrm{T}} \sim N(\mu_{TOY}, C_{TOY})$ where $\mu_{TOY} = [1.0, 0.5, 15.0, 3.5]^{\mathrm{T}}$ and C_{TOY} is a diagonal matrix with diagonal $[1.0, 0.05, 10.0, 2.0]^2$. y could be imagined as a wind speed directional component and x a coordinate in space or time (although the example is only illustrative). The samples could be obtained from repeated observations of a specific geographic area, time period, etc. In figure F.1 (a) we see the "mean" response, compared with 500 samples drawn from the aforementioned probability distribution. If we regard this data as observed, and pretend we do not know about the source function (F.1), we can proceed to use a principal component decomposition. This provides a set of empirical basis functions, which lends itself to the formulation of an experimental design. After performing the decomposition, we find the proportion of variance explained by each component, shown in panel (b). The scores associated with each component decay fairly quickly and 99.1% of the variance is explained by the first twelve components. Reconstruction of the first observation in the original sample using a limited number of components (up to 14) is shown in figure F.2. In this case we might therefore expect to utilise the first, say, twelve components as a basis to describe the wind with little loss of information. The next logical step is then to use the basis to create an experiment design, based upon uncertainty as suggested by the sampled data. Examination of the distribution of scores (or eigenvalues) for each component, shown in figure F.3, suggests appropriate ranges for a design and input distributions for an analysis. Proceeding to formulate a LHC sampling based design, that spans the first twelve components, yields response profiles corresponding to design points as shown in figure F.4. If, as we are imagining, that these responses correspond to some kind of wind input (such as a directional component profile in space or time), then in principle this approach might work. That is, an uncertainty analysis would yield the probability distribution of wave conditions corresponding to the probable variation in wind conditions. A sensitivity analysis would reveal the response of the simulator to each principal component. However, it remains to be seen how effective



(a) Mean and sample data.



(b) Variance explained by successive empirical components.



(c) First three components.

Figure F.1: Sampled response data and its decomposition into principal components.

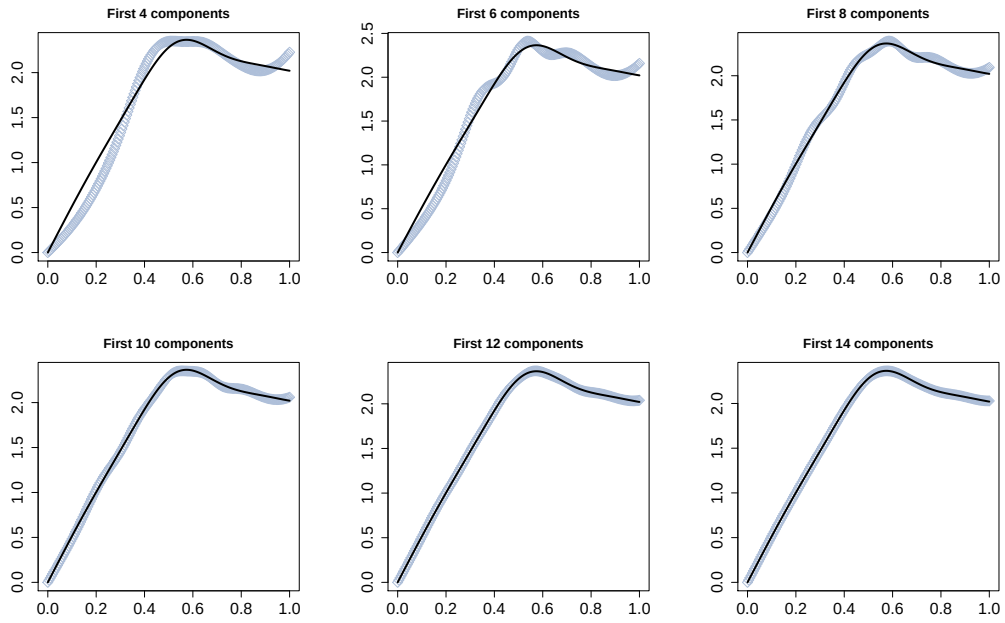


Figure F.2: A single sampled response, reconstructed using a limited number of modes. Improvement is evident with the inclusion of additional modes.

this approach is when applied to realistic winds. It is unclear how many principal components would be required to capture sufficient variance. In this synthetic case the modes decayed fairly rapidly, but if the number of required components was high (say > 50) this could cause computational problems, aside from any physics or interpretation issues. Important questions remain as to whether the designed winds are physically sensible. That is, arbitrarily specifying different weights for each component (as arises during the design process) may create physically unrealistic, or even nonsensical winds. The design space may be highly complex, and comprise many "unphysical" regions (i.e. unrealistic winds) that are separated in unpredictable ways. This would make interpretation of the results challenging.

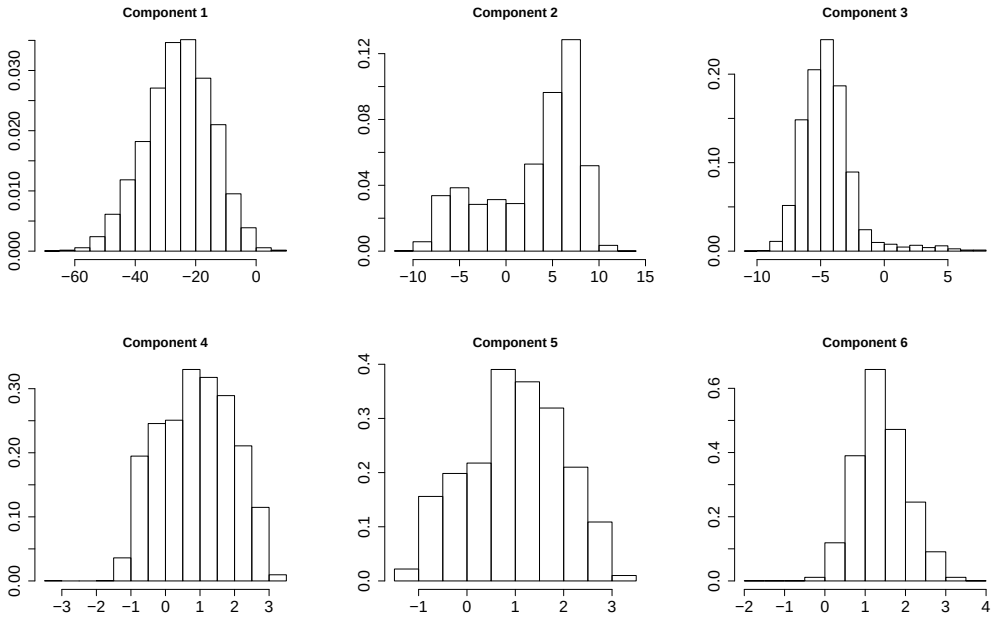


Figure F.3: Probability histograms for the first six response modes.

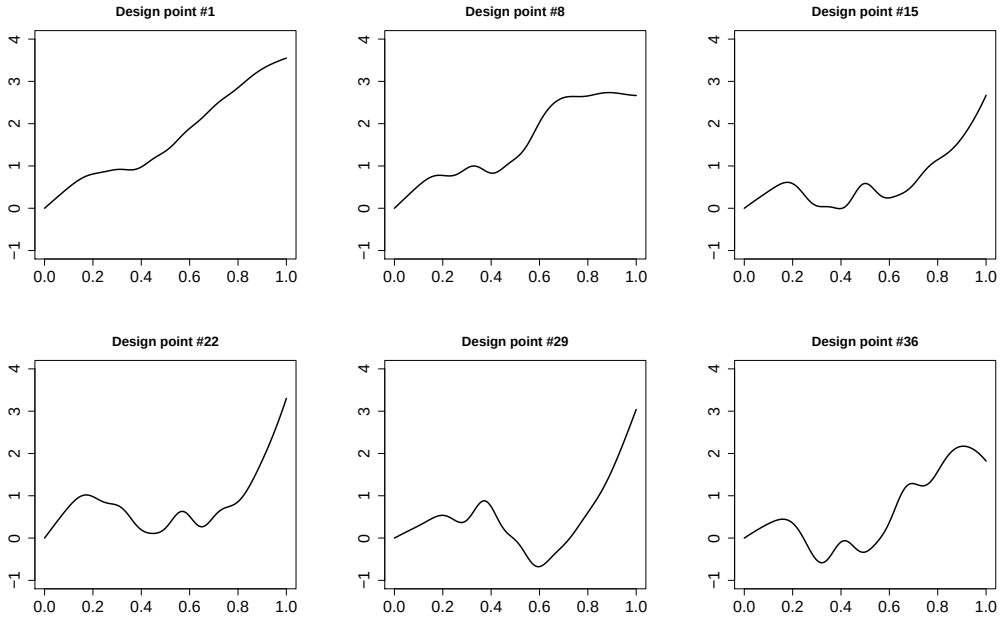


Figure F.4: Response profiles corresponding to a number of design points.

G Details pertaining to simulation of Lake Michigan.

G.1 Communication from GLERL on wind uncertainty.

——— Original Message ———

Subject: Re: Historic wind data for Lake Michigan - 2005

Date: Tue, 4 Jun 2013 12:37:02 -0400

From: Gregory Lang - NOAA Federal gregory.lang@noaa.gov

To: Ben Timmermans ben.timmermans@soton.ac.uk

CC: Dave Schwab djschwab@umich.edu, Eric J Anderson Eric.J.Anderson@noaa.gov, Gregory Lang Gregory.Lang@noaa.gov

Hi Ben,

I've cc'd the project lead (Dr. Eric Anderson) and the former project lead, Dr. Dave Schwab (retired). They may have more insight to the full depth of your questions.

The wind data is indeed sparse in areas, especially over the lake, especially in winter, and especially over parts of Canada. We use a natural neighbor technique, after first applying an over-water correction to land stations, based on the wind direction.

We use NOAAPORT as our data source: http://coastwatch.glerl.noaa.gov/marobs/php/dat

I don't know of particular biases with this technique. Measurement uncertainty is always present, as we utilize many airports, automated c-man stations, and buoys. We exclude ship and Coast Guard station data.

Greg

Bibliography

Abdalla, S., and Cavaleri, L., 2002: Effect of wind variability and variable air density on wave modeling. *Journal of Geophysical Research*, **107**, 17–1 – 17–2.

Alves, J. H. G. M., Banner, M. L., and Young, I. R., 2003: Revisiting the Pierson-Moskowitz asymptotic limits for fully developed wind waves. *Journal of Physical Oceanography*, **33**, 1274–1298.

Alves, J.-H. G. M., Chawla, A., Tolman, H. L., Schwab, D. J., Lang, G., and Mann, G., 2011: The Great Lakes wave model at NOAA/NCEP: Challenges and future developments. *12th International Workshop on Wave Hindcasting and Forecasting*.

Anderson, T. R., 2010: Progress in marine ecosystem modelling and the "unreasonable effectiveness of mathematics". *Journal of Marine Systems*, **81**, 4–11.

Andrianakis, I., and Challenor, P., 2012a: Super-parameterisation of oceanic deep convection using emulators. www.mucm.ac.uk/UCM2012/Forms/Downloads/Andrianakis.pdf.

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

Ardhuin, F., Chapron, B., and Collard, F., 2009: Observation of swell dissipation across oceans. *Geophysical Research Letters*, **36**.

Ardhuin, F., Erick Rogers, W. E., Babanin, A. V., Bidlot, J.-F., Magne, R., Roland, A., van der Westhuysen, A., Queffeulou, P., Lefevre, J.-M., Aouf, L., and Collard, F., 2010: Semiempirical dissipation source functions for ocean waves. Part I: Definition, calibration, and validation. *Journal of Physical Oceanography*, **40**, 1917–1941.

Ardhuin, F., Herbers, T. H. C., van Vledder, G. P., Watts, K. P., and et H. Graber, R. J., 2007: Swell and slanting-fetch effects on wind wave growth. *Journal of Physical Oceanography*, **37**, 908–931.

Babanin, A., and Rogers., W., 2014: Generation and limiters of rogue waves. *International Journal of Ocean and Climate Systems*, **5**, 38–49.

Babanin, A. V., Kahka N. Tsagareli, K., Young, I. R., and Walker, D. J., 2010: Numerical investigation of spectral evolution of wind waves. Part II: Dissipation term and evolution tests. *Journal of Physical Oceanography*, **40**, 667–683.

Babanin, A. V., and van der Westhuysen, A. J., 2007: Physics of "saturation-based" dissipation functions proposed for wave forecast models. *Journal of Physical Oceanography*, **38**, 1831–1841.

Babanin, A. V., Young, I. R., and Banner, M. L., 2001: Breaking probabilities for dominant surface waves on water of finite constant depth. *Journal of Geophysical Research: Oceans*, **106**, 11659–11676.

Banerjee, S., Carlin, B. P., and Gelfand, A. E., 2004: *Hierarchical modeling and analysis for spatial data*. Chapman & Hall.

Banner, M. L., Babanin, A., and Young, I., 2000: Breaking probability for dominant waves on the sea surface. *Journal of Physical Oceanography*, 3145–3160.

Bastos, L. S., and O'Hagan, A., 2009: Diagnostics for gaussian process emulators. *Technometrics*, **51**, 524–439.

Bayarri, M. J., Berger, J. O., Cafeo, J., Garcia-Donato, G., Liu, F., Palomo, J., Parthasarathy, R. J., Paulo, R., Sacks, J., and Walsh, D., 2007: Computer model validation with functional output. *Annals of Statistics*, **35(5)**, 1874–1906.

Bidlot, J., Janssen, P., and Abdalla, S., 2007: A revised formulation of ocean wave dissipation and its model impact. Tech. rep., European Centre for Medium-Range Weather Forecasts, ECMWF, Shinfield Park, Reading, RG2 9AX, England.

Bidlot, J.-R., Holmes, D. J., Wittmann, P. A., Lalbeharry, R., and Chen, H. S., 2002: Intercomparison of the performance of operational ocean wave forecasting systems with buoy data. *Weather and Forecasting*, **17**, 287–310.

Booij, N., and Holthuijsen, L. H., 1987: Propagation of ocean waves in discrete spectral wave models. *Journal of Computational Physics*, **68**, 307–326.

Booij, N., Ris, R. C., and Holthuijsen, L. H., 1999: A third-generation wave model for coastal regions: Model description and validation. *Journal of Geophysical Research*, **104**, 7649–7666.

Boud, R., and Barnett, C., 2012: UK Wave Energy Resource. Tech. Rep. B/C001/29603/003, Carbon Trust.

Breugem, W. A., and Holthuijsen, L. H., 2007: Generalized shallow water wave growth from lake george. *Journal of waterway, port, coastal, and ocean engineering*, **133**, 173–182.

Bricheno, L. M., Soret, A., Wolf, J., Jorba, O., and Baldasano, J. M., 2013: Effect of high-resolution meteorological forcing on nearshore wave and current model performance. *Journal of Atmospheric and Oceanic Technology*, **30**, 1021–1037.

Brito, M., Griffiths, G., Ferguson, J., Hopkin, D., Mills, R., Pederson, R., and Macneil, E., 2012a: A behavioral probabilistic risk assessment framework for managing autonomous underwater vehicle deployments. *Journal of Atmospheric and Oceanic Technology*, **29**, 1689–1703.

Brito, M., Griffiths, G., and Mowlem, M., 2012b: Exploring antarctic subglacial lakes with scientific probes: a formal probabilistic approach for operational risk management. *Journal of Glaciology*, **58(212)**, 1085–1097.

Byrd, R. H., Lu, P., Nocedal, J., and Zhu, C., 1995: A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, **16**, 1190–1208.

Camus, P., Mendez, F. J., and Medina, R., 2011: A hybrid efficient method to downscale wave climate to coastal areas. *Coastal Engineering*, **58**, 851–862.

Cavaleri, L., 2009: Wave modeling - missing the peaks. *Journal of Physical Oceanography*, **39**, 2757–2778.

Cavaleri, L., Fox-Kemper, B., and Hemer, M., 2012: Wind waves in the coupled climate system. *Bulletin of the American Meteorological Society*, **93**, 1651–1661.

Cavaleri, L., Hasselmann, K., and Hasselmann, S., 2007: Wave modelling: The state of the art. *Progress in Oceanography*, **75**, 603–674.

Challenor, P., 2011: Designing a computer experiment that involves switches. *Journal of Statistical Theory and Practice*, **5**, 47–57.

Challenor, P., 2013: Experimental design for the validation of kriging metamodels in computer experiments. *Journal of Simulation*, **7**, 290–296.

Chu, P. C., Qi, Y., Chen, Y., Shi, P., and Mao, Q., 2004: South China Sea wind-wave characteristics. Part I: Validation of Wavewatch-III using TOPEX/Poseidon data. *Journal of Atmospheric and Oceanic Technology*, **21**, 1718–1733.

Clayton, A. M., Lorenc, A. C., and Barker, D. M., 2013: Operational implementation of a hybrid ensemble/4D-Var global data assimilation system at the met office. *Quarterly Journal of Royal Meteorological Society*, **139**, 1445–1461.

Conti, S., and O'Hagan, A., 2010: Bayesian emulation of complex multi-output and dynamic computer models. *Journal of Statistical Planning and Inference*, **140**.

Cornett, A. M., 2008: A global wave energy resource assessment. *Proceedings of the Eighteenth (2008) International Offshore and Polar Engineering Conference*.

Cornford, D., Nabney, I. T., and Williams, C. K. I., 2002: Modelling frontal discontinuities in wind fields. *Journal of Nonparametric Statistics*, **14**, 43–58.

Courant, R., Friedrichs, K., and Lewy, H., 1967: On the partial difference equations of mathematical physics [1928]. *IBM Journal of Research and Development*, **11**, 215–234.

Craig, P., Goldstein, M., Seheult, A., and Smith, J., 1996: Bayes linear strategies for matching hydrocarbon reservoir history. *Bayesian Statistics*, **5**, 69–95.

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

Cressie, N., 1990: The origins of Kriging. *Mathematical Geology*, **22**, 239–252.

Cressie, N., 1993: *Statistics for Spatial Data (Revised Edition)*. Wiley.

de Leon, S. P., and Soares, C. G., 2005: On the sheltering effect of islands in ocean wave models. *Journal of Geophysical Research*, **110**.

Dette, H., and Pepelyshev, A., 2010: Generalized Latin hypercube design for computer experiments. *Technometrics*, **52**, 421–429.

Feng, H., Vandemark, D., Quilfen, Y., Chapron, B., and Beckley, B., 2006: Assessment of wind-forcing impact on a global wind-wave model using the topex altimeter. *Ocean Engineering*, **33**, 1431–1461.

Filipot, J.-F., Ardhuin, F., and Babanin, A. V., 2010: A unified deep-to-shallow water wave-breaking probability parameterization. *Journal of Geophysical Research*, **115**, 1–15.

French, S., 2003: Modelling, making inferences and making decisions: The roles of sensitivity analysis. *Top*, **11**, 229–251.

Fricker, T. E., and Oakley, J. a., 2013: Multivariate Gaussian process emulators with nonseparable covariance structures. *Technometrics*, **55**, 47–56.

Gemmrich, J., Zappa, C. J., Banner, M. L., and Morison, R. P., 2013: Wave breaking in developing and mature seas. *Journal of Geophysical Research: Oceans*, **118**, 4542–4552.

Gilks, W., Richardson, S., and Speigelhalter, D., 1996: *Markov chain Monte Carlo in practice*. Chapman and Hall.

Good, P., Jones, C., Lowe, J., Betts, R., and Gedney, N., 2013: Comparing tropical forest projections from two generations of Hadley Centre earth system models, HadGEM2-ES and HadCM3LC. *Journal of Climate*, **26**, 495–511.

Gouldby, B., Mendez, F., Guanche, Y., Rueda, A., and Minguez, R., 2014: A methodology for deriving extreme nearshore sea conditions for structural design and flood risk analysis. *Coastal Engineering*, **88**, 15–26.

Gramacy, R. B., and Lee, H. K. H., 2010: Cases for the nugget in modeling computer experiments. arXiv:1007.4580.

Gregoire, L. J., Valdes, P. J., Payne, A. J., and Kahana, R., 2010: Optimal tuning of a GCM using modern and glacial constraints. *Climate Dynamics*.

Hanley, K. E., and Belcher, S. E., 2009: Wave-driven wind jets in the marine atmospheric boundary layer. *Journal of Atmospheric Sciences*, **65**, 2646–2660.

Hasselmann, K., 1962a: On the nonlinear transfer of energy in gravity-wave part 1: General theory. *Journal of Fluid Mechanics*, **12**, 481–500.

Hasselmann, K., 1962b: On the nonlinear transfer of energy in gravity-wave part 2: Conservation theorems; wave-particle analogy; irreversibility. *Journal of Fluid Mechanics*, **15**, 273–281.

Hasselmann, K., 1974: On the spectral dissipation of ocean waves due to white-capping. *Boundary-layer meteorology*, **6**, 107–127.

Hasselmann, K. T. P., Barnett, E., Bouws, H., Carlson, D. E., Cart-wright, K., Enke, J. A., Ewing, H., Giennapp, D. E., Hasselmann, P., Kruseman, A., Meersburg, P., Mfiller, D. J., Olbers, K., Richter, W., Sell, and Walden, H., 1973: *Measurements of wind-wave growth and swell decay during the Joint North SeaWave Project (JONSWAP)*.

Hasselmann, S., and Hasselmann, K., 1985: Computation and parameterisations of the nonlinear transfer of energy in gravity-wave part 1: A new method for efficient computations of the exact nonlinear transfer integral. *Journal of Physical Oceanography*, **15**, 1369–1377.

Hasselmann, S., Hasselmann, K., Allender, J. H., and Barnett, T. P., 1985: Computation and parameterisations of the nonlinear transfer of energy in gravity-wave part 2: Parameterisations of the nonlinear energy transfer for application in wave models. *Journal of Physical Oceanography*, **15**, 1378–1391.

Haylock, R. G., and O'Hagan, A., 1996: On inference for outputs of computationally expensive algorithms with uncertainty on the inputs. J. M. Bernardo, J. O. Berger, D. A. P., and A. F. M. Smith, Eds., *Bayesian Statistics 5*, Oxford University Press, 629–637.

Hemer, M. A., Fan, Y., Mori, N., Semedo, A., and Wang, X. L., 2013: Projected changes in wave climate from a multi-model ensemble. *Nature Climate Change*, **3**, 471–476.

Henderson, D. M., and Segur, H., 2013: The role of dissipation in the evolution of ocean swell. *Journal of Geophysical Research: Oceans*, **118**, 5074–5091.

Higdon, D., Gattiker, J., Williams, B., and Rightley, M., 2008: Computer model calibration using high-dimensional output. *Journal of the American Statistical Association*, **103**, 570–583.

Holthuijsen, L. H., 2007: *Waves in Oceanic and Coastal Waters*. Cambridge University Press.

Holthuijsen, L. H., Kuik, A. J., and Mosselman, E., 1987: The response of wave directions to changing wind directions. *Journal of Physical Oceanography*, **17**, 845–853.

Hsu, Y. L., Dykes, J. D., and O'Reilly, W. C., 2004: User's manual for long-range swell forecasting model. Tech. Rep. NRL/MR/7320–04-8719, Naval Research Centre.

Huang, C. J., Qiao, F., Song, Z., and Ezer, T., 2011: Improving simulations of the upper ocean by inclusion of surface waves in the Mellor-Yamada turbulence scheme. *Journal of Geophysical Research*, **116**, 1–13.

Jacques, J., Lavergne, C., and Devictor, N., 2006: Sensitivity analysis in presence of model uncertainty and correlated inputs. *Reliability Engineering and System Safety*, **91**, 1126–1134.

Janssen, P., Hansen, B., and Bidlot, J.-R., 1997: Verifiation of the ECMWF wave forecasting system against buoy and altimeter data. *Weather and Forecasting*, **12**, 763–784.

Janssen, P. A., 2002: Nonlinear four wave interactions and freak waves. Tech. Rep. 366, ECMWF.

Janssen, P. A., 2008: Progress in ocean wave forecasting. *Journal of Computational Physics*, **227**, 3572–3594.

Janssen, P. A., yvind Breivik, Mogensen, K., Vitart, F., Balmaseda, M., Bidlot, J.-R., Leutbecher, S. K. M., Magnusson, L., and Molteni., F., 2013: Air-sea interaction and surface waves. Tech. Rep. 712, ECMWF.

Janssen, P. A. E. M., 1991: Quasi-linear theory of wind-wave generation applied to wave forecasting. *Journal of Physical Oceanography*, **21**, 1631–1642.

Kahma, K., and Calkoen, C., 1992: Reconciling discrepancies in the observed growth of wind-generated waves. *Journal of Physical Oceanography*, **22**, 1389–1405.

Katz, R. W., Craigmile, P. F., Guttorp, P., Haran, M., Sans, B., and Stein, M. L., 2013: Uncertainty analysis in climate change assessments. *Nature Climate Change*, **3**, 769–771.

Kennedy, M., Anderson, C., O'Hagan, A., Lomas, M., Woodward, I., and Gosling, J. P., 2008: Quantifying uncertainty in the biospheric carbon flux for England and Wales. *Journal of the Royal Statistical Association A*, **171**, 109–135.

Kennedy, M., and O'Hagan, A., 2000: Predicting the output from a complex computer code when fast approximations are available. *Biometrika*, **87**, 1–13.

Kennedy, M. C., and O'Hagan, A., 2001: Bayesian calibration of computer models. *Journal of the Royal Statistical Association B*, **63**, 425–464.

Komen, G. J., Cavaleri, L., Donelan, M., Hasselmann, K., Hasselmann, S., and Janssen, P. A. E. M., 1994: *Dynamics and Modelling of Ocean Waves*. Cambridge University Press, 532 pp.

Komen, G. J., Hasselmann, S., and Hasselmann, K., 1984: On the existence of a fully developed wind-sea spectrum. *Journal of Physical Oceanography*, **14**, 1271–1285.

Leckler, F., Ardhuin, F., Filipot, J.-F., and Mironov, A., 2013: Dissipation source terms and whitecap statistics. *Ocean Modelling*, **70**, 62–74.

Lee, L. A., Carslaw, K. S., Pringle, K. J., Mann, G. W., and Spracklen, D. V., 2011: Emulation of a complex global aerosol model to quantify sensitivity to uncertain parameters. *Atmospheric Chemistry and Physics*, **11**, 12253–12273.

Leonard, B. P., 1979: A stable and accurate convective modelling procedure based on quadratic upstream interpolation. *Computer Methods in Applied Mechanics and Engineering*, **18**, 59–98.

Leonard, B. P., 1991: The ultimate conservative difference scheme applied to unsteady one-dimensional advection. *Computer Methods in Applied Mechanics and Engineering*, **88**, 17–74.

Liu, P. C., Schwab, D, J., and Jensen, R. E., 2002: Has wind-wave modeling reached its limit? *Ocean Engineering*, **29**, 81–98.

Loeppky, J. L., Sacks, J., and Welch, W. J., 2009: Choosing the sample size of a computer experiment: A practical guide. *Technometrics*, **51**, 366–376.

Longuet-Higgins, M. S., 1952: On the statistical distribution of the heights of sea waves. *Journal of Marine Research*, **11**, 245–265.

Mackay, E. B., Bahaj, A. S., and Challenor, P. G., 2010a: Uncertainty in wave energy resource assessment. Part 1: Historic data. *Renewable Energy*, **35**, 1792–1808.

Mackay, E. B., Bahaj, A. S., and Challenor, P. G., 2010b: Uncertainty in wave energy resource assessment. Part 2: Variability and predictability. *Renewable Energy*, **35**, 1809–1819.

Mahalanobis, P. C., 1936: On the generalized distance in statistics. *Proceedings of the National Institute of Sciences of India*, vol. 2, 49–55.

Mahjoobi, J., Etemad-Shahidi, A., and Kazeminezhad, M., 2008: Hindcasting of wave parameters using different soft computing methods. *Applied Ocean Research*, **30**, 28–36.

Makarynskyy, O., and Makarynska, D., 2007: Wave prediction and data supplementation with artifcial neural networks. *Journal of Coastal Research*, **23**, 951–960.

Masson, D., 1990: Observations of the response of sea waves to veering winds. *Journal of Physical Oceanography*, **20**, 1876–1885.

McKay, M. D., Backman, R. J., and Conover, W. J., 1979: A comparison of three methods for selecting values of input in the analysis of output from a computer code. *Technometrics*, **21**, 239–245.

Mease, D., and Bingham, D., 2006: Latin hyperrectangle sampling for computer experiments. *Technometrics*, **48**, 467–477.

Meehl, G. A., Covey, C., Delworth, T., Atif, M. L., Mcavaney, B., Mitchell, J. F. B., Stouffer, R. J., and Taylor, K. E., 2007: The WCRP CMIP3 multimodel dataset: A new era in climate change research. *Bulletin of the American Meteorological Society*, **88**, 1383–1394.

Miles, J. W., 1957: On the generation of surface waves by shear flows. *Journal of Fluid Mechanics*, 185–204.

Millar, D., Smith, H., and Reeve, D., 2007: Modelling analysis of the sensitivity of shoreline change to a wave farm. *Ocean Engineering*, **34**, 884–901.

Mitsuyasu, H., 2002: A historical note on the study of ocean surface waves. *Journal of Oceanography*, **58**, 109–120.

Oakley, J., 1999: *Bayesian Uncertainty Analysis for Complex Computer Codes*. Ph.D. thesis, University of Sheffield.

Oakley, J. E., and O'Hagan, A., 2004: Probabilistic sensitivity analysis of complex models: A Bayesian approach. *Journal of the Royal Statistical Association B*, **66**, 751–769.

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

O'Hagan, A., Buck, C. E., Daneshkhah, A., Eiser, J. R., Garthwaite, P. H., Jenkinson, D. J., Oakley, J. E., and Rakow, T., 2006: *Uncertain Judgements*. Wiley.

O'Hagan, A., and Forster, J., 2004: *Kendall's Advanced Threory of Statistics 2B: Bayesian Inference*. Oxford University Press, 2nd edition edn.

O'Hagan, T., 2011: Uncertainty analysis: the variance of the variance. MUCM document at http://mucm.aston.ac.uk/MUCM/MUCMToolkit/index.php?page=ProcUAGP.html.

Oreskes, N., Shrader-Frechette, K., and Belitz, K., 1994: Verification, validation, and confirmation of numerical models in the earth sciences. *Science*, **263**, 641–646.

Palmer, T., 2011: A CERN for climate change. *Physics World*, **(24)**, 14–15.

Perrie, W., Susilo, A., and Toulany, B., 2010: A new approximation for nonlinear wave-wave interactions. *Ocean Modelling*, **33**, 159–176.

Perrie, W., and Toulany, B., 1995: The response of ocean waves to turning winds. *Journal of Physical Oceanography*, **25**, 1116–1129.

Perrie, W., Toulany, B., Resio, D. T., Roland, A., and Auclair, J.-P., 2013: A two-scale approximation for wave-wave interactions in an operational wave model. *Ocean Modelling*, **70**, 38–51.

Phillips, O. M., 1957: On the generation of waves by turbulent wind. *Journal of Fluid Mechanics*, **2**, 417–445.

Phillips, O. M., 1958: The equilibrium range in the spectrum of wind-generated waves. *Journal of Fluid Mechanics*, **4**, 426–434.

Pierson, W. J., 1952: A unified mathematical theory for the analysis, propagation, and refraction of storm generated ocean surface waves. Tech. rep., New York University, College of Engineering, Dept. of Meteorology.

Pierson, W. J., and Moskowitz, L., 1964: A proposed spectral form for fully developed wind seas based on the similarity theory of S. A. Kitaigorodskii. *Journal of Geophysical Research*, **69(24)**, 5181–5190.

Ponce de Leon, S., and Guedes Soares, C., 2008: Sensitivity of wave model predictions to wind fields in the Western Mediterranean sea. *Coastal Engineering*, **55**, 920–929.

Qian, P., 2009: Nested Latin hypercube designs. *Biometrika*, **96**, 957–970.

Rasmussen, C. E., and Williams, C., 2006: *Gaussian Processes for Machine Learning*. No. 026218253X, The MIT Press.

Reikard, G., 2013: Integrating wave energy into the power grid: Simulation and forecasting. *Ocean Engineering*, **73**, 168–178.

Reikard, G., Pinson, P., and Bidlot, J.-R., 2011: Forecasting ocean wave energy: The ECMWF wave model and time series methods. *Ocean Engineering*, **38**, 1089–1099.

Resio, D., and Perrie, W., 1992: A numerical study of nonlinear energy fluxes due to wave-wave interactions, part 1. Methodology and basic results. *Journal of Fluid Mechanics*, **223**, 603–629.

Rice, S. O., 1944: Mathematical analysis of random noise. *Bell System Technical Journal*, **23**, 282–332.

Rogers, W., Kaihatu, J., Petit, H., Booij, N., and Holthuijsen, L., 2002: Diffusion reduction in an arbitrary scale third generation wind wave model. *Ocean Engineering*, **29**, 1357–1390.

Rogers, W. E., 2002: The U.S. navy's global wind-wave models: An investigation into sources of errors in low-frequency energy predictions. Tech. Rep. NRL/FR/7320–02-10,035, Naval Research Laboratory.

Rogers, W. E., Babanin, A., and Wang, D. W. C., 2012: Observation-consistent input and whitecapping dissipation in a model for wind-generated surface waves: Description and simple calculations. *Journal of Atmospheric and Oceanic Technology*, 1329–1346.

Rogers, W. E., and Wang, D. W. C., 2007: On validation of directional wave predictions: Review and discussion. *Journal of Atmospheric and Oceanic Technology*, **24**, 504–520.

Rogers, W. E., Wittmann, P. A., Wang, D. W. C., Clancy, R. M., and Hsu, Y. L., 2005: Evaluations of global wave prediction at the fleet numerical meteorology and oceanography center. *Weather and Forecasting*, **20**, 745–760.

Rougier, J., 2009: Efficient emulators for multivariate deterministic functions. *Journal of Computational and Graphical Statistics*, **17(4)**, 827–843.

Rougier, J., and Cruifix, M., 2012: Draft chapter in "conceptual issues in climate modeling". Submitted 2012, see J. Rougier website.

Rougier, J., and Goldstein, M., 2014: Climate simulators and climate projections. *Annual Review of Statistics and Its Application*, **1**, 103–123.

Rougier, J., Guillas, S., Maute, A., and Richmond, A. D., 2009a: Expert knowledge and multivariate emulation: The thermosphere-ionosphere electrodynamics general circulation model (TIE-GCM). *Technometrics*, **51**, 414–424.

Rougier, J., Sexton, D. M. H., Murphy, J. M., and David Stainforth, ., 2009b: Analyzing the climate sensitivity of the HadSM3 climate model using ensembles from different but related experiments. *Journal of Climate*, **22**, 3540–3557.

Rusu, E., and Soares, C. G., 2009: Numerical modelling to estimate the spatial distribution of the wave energy in the Portuguese nearshore. *Renewable Energy*, **34**, 1501–1516.

Sacks, J., Welch, W. J., Mitchell, T. J., and P., W. H., 1989: Design and analysis of computer experiments. *Statistical Science*, **4**, 409–435.

Sahu, S. K., Yip, S., and Holland, D., 2011: A fast Bayesian method for updating and forecasting hourly ozone levels. *Environmental Ecological Statistics*, **18**, 185–207.

Saltelli, A., Chan, K., and Scott, E. M., 2000: *Sensitivity Analysis*. Wiley series in probability and statistics, John Wiley and Sons.

Saltelli, A., Ratto, M., Tarantola, S., and Campolongo, F., 2005: Sensitivity analysis for chemical models. *Chemical Reviews*, **105**, 2811–2827.

Santner, J. S., Williams, B. J., and Notz, W. I., 2003: *The Design and Analysis of Computer Experiments*. New York: Springer-Verlag.

Sexton, D., Grubb, H., Shine, K., and Folland, C., 2003: Design and analysis of climate model experiments for the efficient estimation of anthropogenic signals. *Journal of Climate*, **16(9)**, 1320–1336.

Sexton, D. M. H., Murphy, J. M., Collins, M., and Webb, M. J., 2012: Multivariate probabilistic projections using imperfect climate models. Part I: Outline of methodology. *Climate Dynamics*, **38**, 2513–2542.

Smith, R. S., Gregory, J. M., and Osprey, A., 2008: A description of the FAMOUS (version XDBUA) climate model and control run. *Geoscientific Model Development*, **1**, 53–68.

Snyder, R. L., Dobson, F. W., Elliott, J. A., and Long, R. B., 1981: Array measurements of atmospheric pressure fluctuations above surface gravity waves. *Journal of Fluid Mechanics*, **102**, 1–59.

Sobol, I. M., 1967: On the distribution of points in a cube and the approximate evaluation of integrals. *USSR Computational Mathematics and Mathematical Physics*, **7**, 86–112.

Stein, M. L., 1999: *Interpolation of Spatial Data: Some Theory for Kriging*. Springer.

Sverdrup, H., and Munk, W., 1947: Wind, sea and swell : theory of relations for forecasting. Tech. rep.

SWAMP, G., 1985: *Ocean Wave Modeling*. Plenum Press.

Tamura, H., Waseda, T., and Miyazawa, Y., 2010: Impact of nonlinear energy transfer on the wave field in pacific hindcast experiments. *Journal of Geophysical Research*, **115**.

Taylor, K. E., Stouffer, R. J., and Meehl, G. A., 2012: An overview of CMIP5 and the experiment design. *Bulletin of the American Meteorological Society*, **93**, 485–498.

Tolman, H. L., 1992: Effects of numerics on the physics in a third-generation wind-wave model. *Journal of Physical Oceanography*, **22**, 1095.

Tolman, H. L., 1995: On the selection of propagation schemes for a spectral wind-wave model. Office note 411, NWS/NCEP.

Tolman, H. L., 2002a: Alleviating the garden sprinkler effect in wind wave models. *Ocean Modelling*, **4**, 269–289.

Tolman, H. L., 2002b: Testing of Wavewatch III version 2.22 in NCEPs NWW3 ocean wave model suite. Tech. rep., National Oceanic and Atmospheric Administration.

Tolman, H. L., 2002c: Validation of Wavewatch III version 1.15. Tech. rep., National Oceanic and Atmospheric Administration.

Tolman, H. L., 2003: Treatment of unresolved islands and ice in wind wave models. *Ocean Modelling*, **5**, 219–231.

Tolman, H. L., 2004a: Inverse modeling of discrete interaction approximations for nonlinear interactions in wind waves. *Ocean Modelling*, **6**, 405–422.

Tolman, H. L., 2004b: Optimum discrete interaction approximations for wind waves. Part 2: Convergence of model integration. Technical note 247, Environmental Modeling Center Marine Modeling and Analysis Branch.

Tolman, H. L., 2009a: *User manual and system documentation of WAVEWATCH III TM version 3.14*. Environmental Modeling Center Marine Modeling and Analysis Branch, U. S. Department of Commerce National Oceanic and Atmospheric Administration National Weather Service National Centers for Environmental Prediction 5200 Auth Road, Camp Springs, MD 20746.

Tolman, H. L., 2009b: *User manual and system documentation of WAVEWATCH III(TM) version 3.14*. Environmental Modeling Center Marine Modeling and Analysis Branch.

Tolman, H. L., 2013a: A generalized multiple discrete interaction approximation for resonant four-wave interactions in wind-wave models. *Ocean Modelling*, **70**, 11–24.

Tolman, H. L., 2013b: The NOPP operational wave model improvement project. *Ocean Modelling*, **70**, 2–10.

Tolman, H. L., and Chalikov, D. V., 1996: Source terms in a third-generation wind-wave model. *Journal of Physical Oceanography*, **26**, 2497–2518.

Tolman, H. L., and Grumbine, R. W., 2013: Holistic genetic optimization of a generalized multiple discrete interaction approximation for wind waves. *Ocean Modelling*, **70**, 25–37.

Tolman, H. L., Krasnopolsky, V. M., and Chalikov, D. V., 2005: Neural network approximations for nonlinear interactions in wind wave spectra: direct mapping for wind seas in deep water. *Ocean Modelling*, **8**, 253–278.

Tracy, B., and Resio, D. T., 1982: Theory and calculation of the nonlinear energy transfer between sea waves in deep water. Tech. Rep. WES Report 11, US Army Corps of Engineers.

Trucano, T. G., Swilera, L. P., Igusa, T., Oberkampf, W. L., and Pilch, M., 2006: Calibration, validation, and sensitivity analysis: What's what. *Reliability Engineering and System Safety*, **91**, 1331–1357.

Tsagareli, K., Babanin, A., Walker, D., and Young, I., 2009: Numerical investigation of spectral evolution of wind-waves. Part I: Wind-input source function. *Journal of Physical Oceanography*, **40**, 656–666.

Tucker, M. J., and Pitt, E. G., 2001: *Waves in Ocean Engineering*. Elsevier Science Ltd.

Uppala, S. M., P. W. Kallberg, A. J., et al., 2005: The ERA-40 re-analysis. *Quarterly Journal of Royal Meteorological Society*, **131**, 2961–3012.

van der Westhuysen, A. J., Zijlema, M., and Battjes, J. A., 2007: Nonlinear saturation-based whitecapping dissipation in SWAN for deep and shallow water. *Coastal Engineering*, **54**, 151–170.

van Vledder, G., and Holthuijsen, L. H., 1993: The directional response of ocean waves to turning winds. *Journal of Physical Oceanography*, **23**, 177–192.

Vogelzang, J., Stoffelen, A., Verhoef, A., and Figa-Saldaa, J., 2011: On the quality of high-resolution scatterometer winds. *Journal of Geophysical Research*, **116**, 1–14.

WAMDI, 1988: The WAM model - a third generation ocean wave prediction model. *Journal of Physical Oceanography*, **18**, 1775–1810.

Wang, C., and Neal, R. M., 2012: Gaussian process regression with heteroscedastic or non-gaussian residuals. *arXiv:1212.6246*, 1–19.

Webb, D. J., 1978: Nonlinear transfers between sea waves. *Deep-Sea Research*, **25**, 279–298.

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

Williamson, D., Goldstein, M., and Blaker, A., 2012: Fast linked analyses for scenario-based hierarchies. *Journal of the Royal Statistical Society C*, **61**, 665–691.

Wolf, J., 2008: Coupled wave and surge modelling and implications for coastal flooding. *Advances in Geoscience*, **17**, 19–22.

Yin, J., Ng, S., and Ng, K., 2011: Kriging metamodel with modified nugget-effect: The heteroscedastic variance case. *Computers and Industrial Engineering*, **61**, 760–777.

Young, I., Babanin, A., and Zieger, S., 2013: The decay rate of ocean swell observed by altimeter. *Journal of Physical Oceanography*, **43**, 2322–2333.

Zakharov, V. E., and Badulin, S. I., 2011: On energy balance in wind driven seas. *Oceanology*, **440**, 1440–1444.