Quantification of modelling uncertainties in turbulent flow simulations

Wouter Nico Edeling

2015-ENAM-0007

École doctorale n° 432 : Science des Métiers de l'ingénieur

Doctorat ParisTech

T H È S E

pour obtenir le grade de docteur délivré par

l'École Nationale Supérieure d'Arts et Métiers

Spécialité " Mécanique des Fluides "

présentée et soutenue publiquement par

Wouter EDELING

14-04-2015

Quantification of Modelling Uncertainties in Turbulent Flow Simulations

Directeur de thèse : **Paola CINNELLA**
Co-encadrement de la thèse : **Hester BIJL**

Jury

M. Gianluca IACCARINO, Prof. Dr., Departement of Mechanical Engineering, Stanford University	Rapporteur
M. Chris LACOR, Prof. Dr., Departement of Mechanical Engineering, Vrije Universiteit Brussel	Rapporteur
M. Siddhartha MISHRA, Prof. Dr., Departement Mathematik, ETH Zürich	Examinateur
M. Arnold HEEMINK, Prof. Dr., Delft Institute of Applied Mathematics, TU Delft	Examinateur
M. Richard DWIGHT, Prof. Dr., Aerodynamics group, TU Delft	Examinateur
Mme. Paola CINNELLA, Prof. Dr., DynFluid, Arts et Métiers	Examinateur
M. Fred VAN KEULEN, Prof. Dr., Departement of Precision and Microsystems Engineering, TU Delft	Examinateur
Mme. Hester BIJL, Prof. Dr., Faculty of Aerospace Engineering, TU Delft	Examinateur

T H È S E

Arts et Métiers ParisTech - Centre de Paris
Laboratoire Dynfluid

QUANTIFICATION OF MODELLING UNCERTAINTIES IN TURBULENT FLOW SIMULATIONS

WOUTER NICO EDELING

QUANTIFICATION OF MODELLING UNCERTAINTIES IN TURBULENT FLOW SIMULATIONS

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. ir. K. C. A. M. Luyben,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op dinsdag 14 April 2015 om 15:00 uur

door

Wouter Nico EDELING

ingenieur Luchtvaart en Ruimtevaart,
geboren te Leiderdorp, Nederland.

This thesis is approved by the promotors:

Prof. dr. ir. drs. H. Bijl.
Prof. dr. P. Cinnella

Composition of the doctoral committee:

Rector Magnificus,	voorzitter
Prof. dr. ir. drs. H. Bijl,	Technische Universiteit Delft, promotor
Prof. Dr. P. Cinnella,	Arts et Métiers ParisTech, promotor
Prof. Dr. R.P. Dwight	Technische Universiteit Delft, supervisor

Independent members:

Prof. Dr. G. Iaccarino	Stanford University
Prof. Dr. Ir. C. Lacor	Vrije Universiteit Brussel
Prof. Dr. M. Siddhartha	Swiss Federal Institute of Technology Zürich
Prof Dr. Ir. A. Heemink	Technische Universiteit Delft
Prof. Dr. F. Scarano	Technische Universiteit Delft, reserve member

Dr. R.P. Dwight has made important contributions to the realisation of this thesis.

This is a dual diploma together with the following partner institute for higher education: Arts et Métiers ParisTech, Paris, France, stipulated in the diploma supplement and approved by the Board for Doctorates.

Keywords: uncertainty quantification, prediction, RANS turbulence models, Bayesian statistics, Bayesian model-scenario averaging

An electronic version of this dissertation is available at
`http://repository.tudelft.nl/`.

Remember that all models are wrong;
the practical question is how wrong do they have to be to not be useful.

George Edward Pelham Box

CONTENTS

1

INTRODUCTION

In this thesis we use the framework of Bayesian statistics to quantify the effect of modelling uncertainties on the predictions made with eddy-viscosity turbulence models. The two keywords here are 'modelling uncertainties' and 'predictions'. Firstly, by uncertainties we mean a lack of knowledge that is present in multiple aspects of turbulence models. Not only uncertain input coefficients, but also the assumptions inherent in the mathematical form of the turbulence models should be acknowledged. Furthermore, the scenario in which a model is applied can also introduce uncertainties. Secondly it is important to note that, while all turbulence models are calibrated to fit a set of experimental data, this does prove their predictive capability. In fact, due to the mentioned modelling uncertainties, the validity of a single deterministic prediction is simply unknown. To circumvent this problem, a predictive simulation with quantified uncertainty should be made. The authors of [7] define the term 'predictive simulation' as follows

> *The systematic treatment of model and data uncertainties and their propagation through a computational model to produce predictions of quantities of interest with quantified uncertainty.*

Performing such predictive simulations for RANS turbulence models is the general aim of this thesis.

This thesis begins with a brief introduction to uncertainty quantification with the Bayesian statistical framework. In Chapter 2 this framework is applied to the $k-\varepsilon$ turbulence model. We perform multiple calibrations under different flow scenarios within a single, computationally inexpensive, class of flow. This allows us to investigate the variability of the input coefficients, and to develop a predictive methodology that uses the observed variability in order to make a prediction for an unmeasured scenario. In Chapter 3 Bayesian calibrations are performed on a set of different turbulence models. A more robust predictive framework based on Bayesian Model-Scenario Averaging is also developed in this chapter. In order to apply the developed uncertainty quantification techniques to computationally expensive flows, efficient surrogate models are required. This

is the subject of Chapter 4, in which we develop a version of the Simplex Stochastic Collocation method with improved performance characteristics. This collocation method is merged with our Bayesian Model-Scenario Averaging framework in Chapter 5, with the goal of obtaining turbulence model uncertainty estimates for a computationally expensive transonic flow over an airfoil. In the final chapter we give our conclusions regarding the developed uncertainty quantification methodologies, and we outline our recommendations for future research.

1.1. COMPUTER PREDICTIONS WITH QUANTIFIED UNCERTAINTY

Historically, scientific predictions were made by either theories (mathematical models), or observations (experiments). In many instances, a mathematical model has parameters in need of tuning, so that the model may represents the physical phenomenon of interest as best as possible. The closure coefficients found in turbulence models are an example of this. Unfortunately, the exact values of these parameters are not always known, and may well not exist, which introduces a source of error in the model. Generally, these coefficients are tuned based upon available experimental data. However, the observations themselves are also not free from error, since the measurements are corrupted by imperfect instruments or slight variations in the experimental conditions.

The rise of the digital computer has led to the establishment of the third pillar of science, i.e. computer modelling and simulation. This allowed for mathematical models of great complexity to be simulated, or perhaps more accurately, to be approximated. Because, before a mathematical model can be simulated on a computer it often needs to be discretized, introducing yet another source of error. Thus, all possible methods for scientific prediction are encumbered by their own specific source of uncertainty, which is schematically depicted in Figure 1.1

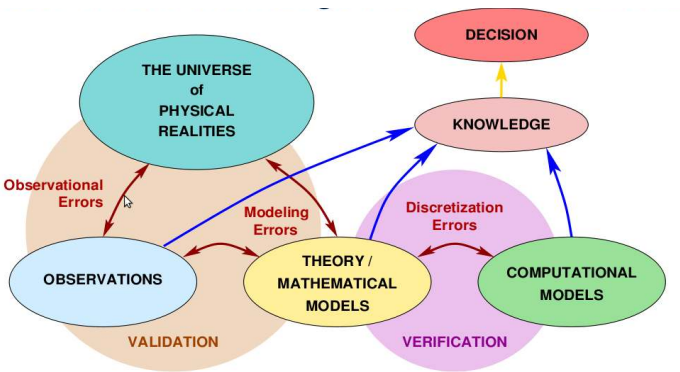


Figure 1.1: The imperfect paths to knowledge, source [7]

A framework which provides a methodology for coping with these imperfections is Bayesian statistics, named after the Reverend Thomas Bayes [1]. The Bayesian framework is discussed in further detail in Section 1.2. Here we first discuss the stages of predictive science, as defined in [7]

1. **Identifying the quantities of interest:** A computer simulation should begin with a clear specification of the goals of the simulation. This is the specification of the target output, or 'quantities of interest' (QoI). This is of great importance because models might be well suited to represent one target functional, but completely inadequate of representing another. For instance, a potential flow model may be adequate to predict aerodynamic lift on airfoils at low angle of attack, but it is totally incapable of predicting (friction and pressure) drag.

2. **Verification:** This is the process designed to detect errors due to the discretization of the mathematical model, and errors due to incorrect software implementation, i.e. software bugs.

3. **Calibration** Calibration is the activity of adjusting the unknown input parameters of the model until the output of the model fits the observed data as best as possible. This amounts to solving an inverse problem, which is often ill-posed. Nonetheless, as we shall see in Section 2.4, the Bayesian framework provides us with a means of regularization.

4. **Validation** The validation process in meant to assess to what degree the (calibrated) model is capable of predicting the QoIs of step 1, and can thus be considered as a forward problem. This process requires a carefully designed program of experiments in order to determine how much the model deviates from observation. These observations are only used for comparative purposes, in no way are they used to inform the model. A complication is that the QoIs are not always accessible for observation. It is up to the modeller to determine what degree of disagreement invalidates the model for predicting the specified QoI's.

The Bayesian framework represents uncertainty as probability, which is normally used to represent a random process. However, one type of uncertainty, namely *aleatoric* uncertainty, does indeed arise through natural random variations of the process. This type of uncertainty is irreducible, in that more data or better models will not reduce it. *Epistemic* uncertainty on the other hand, arises from a lack of knowledge about the model, e.g. unknown model parameters or mathematical form. This type of uncertainty is usually dominant and it can in principle be reduced. Epistemic uncertainty is also represented through probability in Bayesian statistics. In this case, the uncertainty represents our confidence in some proposition, given all the current observational data. More importantly, in the Bayesian framework this confidence can be updated once more data becomes available.

Within the Bayesian framework, the validation phase requires uncertainties to be propagated through the model, as it is a forward problem. For this purpose one might use Stochastic Collocation Methods, as described in Chapter 4. But first a Bayesian calibration must be performed. It should be noted however, that uncertainty propagation can occur during the calibration phase as well, for instance when the effect of the prior distribution is examined. A general overview of Bayesian statistics is given in the next section, followed by a section outlining the application of Bayesian statistics to computer models.

1

1.2. GENERAL BAYESIAN DATA ANALYSIS

By Bayesian data analysis, we mean practical methods for making inferences from data using probability models for quantities we observe and for quantities we wish to learn about. The general process for Bayesian data analysis can be broken down into the following three steps

1. **Setting up a full probability model:** a joint-probability distribution for both the observed and unobserved quantities in the problem.
2. **Conditioning on observed data:** calculating the conditional *posterior* probability distribution of the unobserved quantities given the observed data.
3. **Evaluating the fit of the model:** evaluate if the model fits the data, how sensitive are the results to the assumptions of step 1 etc.

Note that the second and third step correspond to the calibration and validation phase of Section 1.1 respectively. However, it should also be noted that the breakdown of steps in Section 1.1 is designed with a clear ultimate goal in mind (the prediction of the defined quantities of interest), which is lacking from the above statement.
Inferences [1] are made for two kinds of unobserved quantities, i.e.

1. *Parameters* that govern the model, which we denote by the column vector $\boldsymbol{\theta}$.
2. *Future predictions* of the model. If we let $\mathbf{z} = (z_1, z_2, \cdots, z_n)$ [2] denote the observed data, then the currently unknown (but possibly observable) future predictions are denoted by $\tilde{\mathbf{z}}$.

Finally, we also have a class of *explanatory* variables $\mathbf{x}$. These are variables that we do not bother to model as random, but who could possibly could be moved into the $\mathbf{z}$ (or possibly $\boldsymbol{\theta}$) category if we choose to do so. For instance exactly specified boundary conditions, or applied pressure gradients fall into this category. Our final word on notation is that we use $p(\cdot)$ denote a probability density function.

In short, the aim of Bayesian data analysis is to draw conclusions about $\boldsymbol{\theta}$ (calibration) through the conditional posterior distribution $p(\boldsymbol{\theta} \mid \mathbf{z})$, or about $\tilde{\mathbf{z}}$ (prediction) through $p(\tilde{\mathbf{z}} \mid \mathbf{z})$. We can achieve this via the application of *Bayes' rule*

$$p(\boldsymbol{\theta} \mid \mathbf{z}) = \frac{p(\mathbf{z} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})}{p(\mathbf{z})} \tag{1.1}$$

where the law of total probability states that $p(\mathbf{z}) = \int p(\mathbf{z} \mid \boldsymbol{\theta}) p(\boldsymbol{\theta}) d\boldsymbol{\theta}$. Bayes' rule follows immediately from the definition of conditional probability, see for instance [4]. Since

[1] Statistical inference is the process of drawing conclusions from numerical/observed data about unobserved quantities.
[2] Depending on the dimension of the individual entries z_i, $\mathbf{z}$ can be either a vector or a matrix.

this denominator does not depend upon $\boldsymbol{\theta}$, it is often omitted to yield the unnormalized version of (1.1)

$$p(\boldsymbol{\theta} \mid \mathbf{z}) \propto p(\mathbf{z} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta}) \tag{1.2}$$

The term $p(\mathbf{z} \mid \boldsymbol{\theta})$, i.e. the distribution of the data given the parameters is called the *likelihood function*, and it provides the means for updating the model once more data becomes available. The term $p(\boldsymbol{\theta})$ is the *prior distribution* of $\boldsymbol{\theta}$, i.e. it represents what we know about the parameters before the data became available.

If our prior knowledge of $\boldsymbol{\theta}$ is limited, then choosing $p(\boldsymbol{\theta})$ as a uniform distribution which includes all possible values for $\boldsymbol{\theta}$ is often appropriate [3]. Such a prior is called a *non-informative prior*. If we do have some prior information about the distribution of $\boldsymbol{\theta}$ we can encode it into $p(\boldsymbol{\theta})$ to obtain a so-called *informative prior*. A special class of informative priors are the *conjugate* priors, which are priors that have the same parametric form as the posterior. For instance, if the posterior is normally distributed then $p(\boldsymbol{\theta}) = \exp\left(A\boldsymbol{\theta}^2 + B\boldsymbol{\theta} + C\right)$ is the family of conjugate priors with the same parametric form as the normal distribution.

A *proper* prior is one that integrates to 1 and does not depend upon the data. However, for a non-informative prior this is not always the case. Assume that we have a normal likelihood, then a non-informative prior (one proportional to a constant) integrates to infinity over $\boldsymbol{\theta} \in (-\infty, \infty)$. For the normal case this is not a real problem because the resulting posterior distribution (a constant times the normal likelihood) is still proper, i.e. it has a finite integral. But, we have no guarantee that this will happen for all distributions. Therefore, when using improper priors, one must always check if the resulting posterior distribution is proper.

The posterior predictive distribution conditional on the observed $\mathbf{z}$ can be written as

$$p(\tilde{\mathbf{z}} \mid \mathbf{z}) = \int p(\tilde{\mathbf{z}}, \boldsymbol{\theta} \mid \mathbf{z})\, d\boldsymbol{\theta} = \int p\left(\tilde{\mathbf{z}} \mid \boldsymbol{\theta}, \mathbf{z}\right) p(\boldsymbol{\theta} \mid \mathbf{z})\, d\boldsymbol{\theta} = \int p\left(\tilde{\mathbf{z}} \mid \boldsymbol{\theta}\right) p(\boldsymbol{\theta} \mid \mathbf{z})\, d\boldsymbol{\theta} \tag{1.3}$$

The last step follows because $\tilde{\mathbf{z}}$ and $\mathbf{z}$ are assumed to be conditionally independent given $\boldsymbol{\theta}$, i.e. $p(\tilde{\mathbf{z}}|\mathbf{z}, \boldsymbol{\theta}) = p(\tilde{\mathbf{z}}|\boldsymbol{\theta})$.

A general feature of Bayesian analysis is that the posterior distribution is centered around a point which represents a compromise between the prior information and the observed data. This compromise will be increasingly controlled by the data if the sample size, i.e. n in $\mathbf{z} = (z_1, \cdots, z_n)$, increases.

Note that since we have to choose a probability model for $p(\cdot)$, we have no guarantee the the chosen model is indeed a good model. This is something that should be evaluated in the third step of Bayesian data analysis, or perhaps better suited, the validation phase described in chapter 1.1. A second option is to do the Bayesian analysis using multiple stochastic models. In this case some models can already be invalidated after the calibration phase, as described in for instance [2].

We close this section with some useful formulas from probability theory

[3] This is known as the 'principle of insufficient reason', i.e. if we have no prior information on the parameters, it is reasonable to model them with equal probability.

$$\mathbb{E}_{\mathbf{z}}(\boldsymbol{\theta}) = \mathbb{E}_{\mathbf{z}}(\mathbb{E}_{\boldsymbol{\theta}}(\boldsymbol{\theta} \mid \mathbf{z})) \tag{1.4}$$

$$\mathrm{Var}_{\mathbf{z}}(\boldsymbol{\theta}) = \mathbb{E}_{\mathbf{z}}(\mathrm{Var}_{\boldsymbol{\theta}}(\boldsymbol{\theta} \mid \mathbf{z})) + \mathrm{Var}_{\mathbf{z}}(\mathbb{E}_{\boldsymbol{\theta}}(\boldsymbol{\theta} \mid \mathbf{z})) \tag{1.5}$$

Equation (1.4) states that the prior mean of $\boldsymbol{\theta}$ is the average of all possible posterior means over the distribution of the data. Equation (1.5) states that the posterior variance is on average *smaller* than the prior variance by an amount that depends upon the variance of the posterior means over the distribution of the data. This is in line with intuition, since one would expect that the posterior distribution (due to the incorporation of the data) shows less variation than the prior distribution. Thus, the larger the variation of $\mathbb{E}(\boldsymbol{\theta} \mid \mathbf{z})$, the more potential we have for reducing the uncertainty in $\boldsymbol{\theta}$. In other words, if $\boldsymbol{\theta}$ is very sensitive to the data, then given a certain measured $\mathbf{z}$ we will be able to obtain an informed distribution of $\boldsymbol{\theta}$. We will see examples of this in Chapters 2 and 3.

1.3. BAYESIAN DATA ANALYSIS APPLIED TO COMPUTER MODELS

Kennedy and O'Hagan [5] wrote a general paper which deals with the Bayesian calibration of (deterministic) computer models. It was the first paper of its kind that took into account all sources of uncertainty arising in the calibration (and subsequent prediction) of computer models. These sources of uncertainty are

1. **Parametric uncertainty:** This is the uncertainty which arises due to insufficient knowledge about the model parameters $\boldsymbol{\theta}$. In the context of computer models, $\boldsymbol{\theta}$ can be thought of as a vector of unknown code inputs.

2. **Model inadequacy:** Even if we would know the exact value of $\boldsymbol{\theta}$, there is still no such thing as a perfect model. Due to for instance modeling assumptions, the model output will not equal the true value of the process. This discrepancy is the model inadequacy. Since the process itself may exhibit some natural variations, the model inadequacy is defined as the difference between the mean value of the real-world process and the model output at the true values of the input $\boldsymbol{\theta}$.

3. **Residual variability:** Computer codes may not always give the same output, even though the input values remain unchanged. This variation is called residual variability. It may arise because the process itself is inherently stochastic, but it may also be that this variation can be diminished is we could recognize and specify some more model conditions. This can be tied the the discussion of aleatoric and epistemic uncertainties of Section 1.1. In any case, the Bayesian framework does not discriminate between these two types of uncertainties by representing them both with probability distributions.

4. **Observation error:** This is the difference between the true value of the process and its measured value.

5. **Code uncertainty:** The output of a computer code, given a set of inputs $\boldsymbol{\theta}$ is in principle not unknown. However, if the code is computationally expensive it might

be impractical to actually run the code at every input configuration of interest. In this case uncertainty about the code output can also be incorporated into the Bayesian framework.

The authors of [5] use Gaussian processes, mainly for convenience, to model both the computer code and the model inadequacy. This model includes the relationship between the observations z_i, the true process $\zeta(\cdot)$ and the model output $\delta(\cdot,\cdot)$ subject to code uncertainty. It is given by

$$z_i = \zeta(\mathbf{x}_i) + e_i = \rho\delta(\mathbf{x}_i, \boldsymbol{\theta}_b) + \eta(\mathbf{x}_i) + e_i, \tag{1.6}$$

where e_i is the observational error for the i^{th} observation. The observational errors e_i are modelled as independently normally-distributed variables $N(0,\lambda)$. Note that the zero mean indicates that it is assumed that on average the measured values z_i are correct, i.e. there is no bias in the measurement. The term ρ is a constant (unknown) hyper parameter and $\eta(\cdot)$ is a term to represent the model inadequacy. Equation (1.6) implies that the true value is modelled as

$$\zeta(\mathbf{x}_i) = \rho\delta(\mathbf{x}_i, \boldsymbol{\theta}_b) + \eta(\mathbf{x}_i), \tag{1.7}$$

where $\boldsymbol{\theta}_b$ are the 'best-fit' model parameters. Thus, these are the values for $\boldsymbol{\theta}$ which lead to the closest fit to the experimental data, such that (1.7) is valid. Obviously the best-fit values of $\boldsymbol{\theta}$ are unknown *a-priori*, hence the need for calibration. Also note that if the code can be sampled at every point of interest (no code uncertainty), the stochastic term $\rho\delta$ reduces to the deterministic code output $\mathbf{y}$ computed using $\boldsymbol{\theta}_b$. Equation (1.6) is just one way of modelling the relationship between code output and the real-life process, in Chapter 2 we will use a different form. This chapter also describes the calibration procedure and the modelling choice made for η.

The described statistical framework has found many applications in physical problems, for instance in thermal problems [3] or in climate-change models [10]. Applications to structural mechanics also exist, such as [8, 11]. In the field of fluid mechanics the Bayesian framework is also extensively used to quantify the uncertainty in numerical flow models. For instance the authors of [9] use a Bayesian multi-model approach to quantify the uncertainty in groundwater models, and in [6] thermodynamic uncertainties in dense-gas flow computations are accounted for. In this thesis the Bayesian framework is applied to turbulent flow problems, which is the subject of the next chapter.

REFERENCES

[1] Mr. Bayes and M. Price. An essay towards solving a problem in the doctrine of chances. by the late rev. mr. bayes, frs communicated by mr. price, in a letter to john canton, amfrs. *Philosophical Transactions (1683-1775)*, pages 370–418, 1763.

[2] S.H. Cheung, T.A. Oliver, E.E. Prudencio, S. Prudhomme, and R.D. Moser. Bayesian uncertainty analysis with applications to turbulence modeling. *Reliability engineering and systems safety*, 96(9):1137–1149, 2011.

[3] D. Higdon, C. Nakhleh, J. Gattiker, and B. Williams. A bayesian calibration approach to the thermal problem. *Computer Methods in Applied Mechanics and Engineering*, 197(29):2431–2441, 2008.

[4] H.P. Hsu. *Schaum's outline of theory and problems of probability, random variables, and random processes*. Schaum's Outline Series, 1997.

[5] M.C. Kennedy and A. O'Hagan. Bayesian calibration of computer models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(3):425–464, 2001.

[6] X Merle and P Cinnella. Bayesian quantification of thermodynamic uncertainties in dense gas flows. *Reliability Engineering & System Safety*, 2014.

[7] T. Oden, R. Moser, and O. Ghattas. Computer predictions with quantified uncertainty. Technical report, ICES-REPORT 10-39, The institute for Computational Engineering and Sciences, The University of Texas at Austin, 2010.

[8] F. Perrin, B. Sudret, and M. Pendola. Bayesian updating of mechanical models-application in fracture mechanics. *18ème Congrès Français de Mécanique (Grenoble 2007)*, 2007.

[9] R. Rojas, Sa. Kahunde, L. Peeters, O. Batelaan, L. Feyen, and A. Dassargues. Application of a multimodel approach to account for conceptual model and scenario uncertainties in groundwater modelling. *Journal of Hydrology*, 394(3):416–435, 2010.

[10] D.A. Stainforth, T. Aina, C. Christensen, M. Collins, N. Faull, DJ Frame, JA Kettleborough, S. Knight, A. Martin, JM Murphy, et al. Uncertainty in predictions of the climate response to rising levels of greenhouse gases. *Nature*, 433(7024):403–406, 2005.

[11] B. Sudret, M. Berveiller, F. Perrin, and M. Pendola. Bayesian updating of the long-term creep deformations in concrete containment vessels. In *Proc. 3rd Int. ASRANET Colloquium, Glasgow*, 2006.

2

BAYESIAN ESTIMATES OF PARAMETER VARIABILITY IN THE $k-\varepsilon$ TURBULENCE MODEL

2.1. INTRODUCTION

Computational Fluid Dynamics (CFD) and Reynolds-averaged Navier-Stokes (RANS) simulations in particular form an important part of the analysis and design methods used in industry. These simulations are typically based on a deterministic set of input variables and model coefficients. However real-world flow problems are subject to numerous uncertainties, e.g. imprecisely known parameters, initial- and boundary conditions. For input uncertainties described as probability density functions (pdfs), established methods exist for determining the corresponding output uncertainty [5, 6, 40]. Furthermore, numerical predictions are affected by numerical discretization errors and approximate physical models (turbulence models in RANS). The former may be estimated and controlled by means of mesh refinement (e.g. Ref. 7), but no analogue exists for the latter. This error, which we call *model inadequacy* in the following, is therefore the only major source of simulation error that remains difficult to estimate. It is therefore the bottleneck in the trustworthiness of RANS simulations. This chapter describes an attempt to construct an estimate of model inadequacy in RANS for a limited set of flows, and for a single turbulence closure model, tke $k-\varepsilon$ model in this case.

Within the framework of RANS, many turbulence models are available, see e.g. Ref. 39 for a review. There is general agreement that no universally-"best" RANS turbulence closure model is currently known; the accuracy of models is problem-dependent [42]. Moreover, each turbulence model uses a number of closure coefficients which are classically determined by calibration against a database of fundamental flows [30]. Model performance may strongly depend on these values, which are often adjusted to improve

This chapter is based on: W.N. Edeling, P. Cinnella, R.P. Dwight, H.Bijl, Bayesian estimates of the parameter variability in the $k-\varepsilon$ model, Journal of Computational Physics, 258 (2014) 73–94.

model accuracy for a given set of problems, or for a specific flow code. They are almost always assumed to be constant in space and time. For a given model there is sometimes no consensus on the best values for these coefficients, and often intervals are proposed in the literature [26].

Our approach is to represent model inadequacy by uncertainty in these coefficients. Summarized we proceed as follows: (1) we define the class of flows for which we wish to estimate the error, in this article turbulent boundary-layers for a range of pressure gradients. (2) We collect experimental data for a number of flows of this class. (3) We use Bayesian model updating to calibrate the closure coefficients against each flow in this data-set, resulting in posterior distributions on the coefficients for each flow [8]. (4) We summarize the large amount of posterior information using highest posterior-density (HPD) intervals. This summary gives intervals on the coefficients which represent both the spread of coefficients within the flow-class, as well as the ability of the calibration to provide information about the values these coefficients should take in each flow. (5) For a new flow of the class, for which there might be no experimental data, we then perform a simulation using the model with the specified coefficient uncertainties. The resulting interval on the model output is our probabilistic estimate of the true flow.

Representing model inadequacy by uncertainty in closure coefficients is reasonable since the coefficients are empirical: they must be seen as "tuning parameters" associated to the model and, in general, they are not expected to be flow-independent. Furthermore each coefficient is involved in an approximation of the underlying physics, and therefore is closely related to some component of the model inadequacy. Finally an error estimate based on coefficient uncertainty has the virtue of being geometry-independent – that is we do not need to assume a particular flow topology to apply the estimate. We do not claim that it is possible to approximate all turbulence model inadequacy in this way. The method does rely on being able to approximate most of it, and we demonstrate that this is possible for the limited class of flows we consider.

The key step in the method is the calibration of the coefficients. For the calibration phase we follow the work of Cheung et al. [3], in which a Bayesian approach was applied to the calibration of the Spalart-Allmaras [32] turbulence model, taking into account measurement error [18]. In that work, for a given statistical model, the coefficients were calibrated once on all the available measured velocity profiles and wall-shear stress components. Model inadequacy was treated with a multiplicative term parameterized in the wall-normal direction with a Gaussian process, following the framework of Kennedy and O'Hagan [12]. In the present work, we perform an analysis by performing separate calibrations on multiple flows in our class, using the $k-\varepsilon$ model, with Launder-Sharma damping functions [16]. Using uniform priors and calibrating against a large, accurate data-set containing boundary-layer profiles at different pressure gradients, results in informative coefficient posteriors for each flow. The multiplicative model inadequacy term is retained to capture the part of the error which cannot be captured by the closure coefficients alone.

We choose the pressure gradient as the independent variable in our flow class because it is known to have a large impact on the performance of $k-\varepsilon$ model [14, 27, 37]. Approaching this problem in a Bayesian context allows us to estimate how much this deficiency can be reduced by choice of closure coefficients alone, and how much the co-

efficients have to vary to match measurements at all pressure-gradients. The spread of coefficients is an indication of flow-independence of the model, and we expect better models to have smaller spreads.

This chapter is laid out as follows: we briefly outline the $k-\varepsilon$ model in Section 2.2. Section 2.3 describes the experimental data used for the calibration, and Section 2.4 describes our calibration framework, in particular the statistical model and priors. The results, including verification, HPD analysis of calibration posteriors, and prediction using the obtained coefficient uncertainties are described in Section 2.5. Specifically, our confidence interval estimate for error due to turbulence modelling inadequacy is given in Section 2.5.7. Finally, Section 2.6 summarizes the main findings and provides guidelines for future research.

2.2. THE $k-\varepsilon$ TURBULENCE MODEL

The general simulation approach considered in this chapter is the solution of the RANS equations for turbulent boundary layers, supplemented by a turbulence model. RANS equations remain up to now the most advanced and yet computationally acceptable simulation tool for engineering practice, since more advanced strategies, like Large Eddy Simulation (see e.g. Ref. 28) are yet too expensive for high-Reynolds flows typically encountered in practical applications. Under the assumption of incompressibility, the governing equations for a boundary-layer flow are given by

$$\frac{\partial \bar{u}_1}{\partial x_1} + \frac{\partial \bar{u}_2}{\partial x_2} = 0, \tag{2.1a}$$

$$\frac{\partial \bar{u}}{\partial t} + \bar{u}_1 \frac{\partial \bar{u}_1}{\partial x_1} + \bar{u}_2 \frac{\partial \bar{u}_1}{\partial x_2} = -\frac{1}{\rho}\frac{\partial \bar{p}}{\partial x_2} + \frac{\partial}{\partial x_2}\left[(\nu + \nu_T)\frac{\partial \bar{u}_1}{\partial x_2}\right], \tag{2.1b}$$

where ρ is the constant density, $\bar{u}_i$ is the mean velocity in x_i direction and ν is the kinematic viscosity. The eddy viscosity ν_T is meant to represent the effect of turbulent fluctuations on the mean flow, and is calculated here through the $k-\varepsilon$ turbulence model:

$$\nu_T = C_\mu f_\mu \frac{k^2}{\tilde{\varepsilon}}, \tag{2.2a}$$

$$\frac{\partial k}{\partial t} + \bar{u}\frac{\partial k}{\partial x_1} + \bar{v}\frac{\partial k}{\partial x_2} = \nu_T \left(\frac{\partial \bar{u}}{\partial x_2}\right)^2 - \varepsilon + \frac{\partial}{\partial x_2}\left[\left(\nu + \frac{\nu_T}{\sigma_k}\right)\frac{\partial k}{\partial x_2}\right], \tag{2.2b}$$

$$\frac{\partial \tilde{\varepsilon}}{\partial t} + \bar{u}\frac{\partial \tilde{\varepsilon}}{\partial x_1} + \bar{v}\frac{\partial \tilde{\varepsilon}}{\partial x_2} = C_{\varepsilon 1} f_1 \frac{\tilde{\varepsilon}}{k}\nu_T \left(\frac{\partial \bar{u}}{\partial x_2}\right)^2 - C_{\varepsilon 2} f_2 \frac{\tilde{\varepsilon}^2}{k} + E + \frac{\partial}{\partial x_2}\left[\left(\nu + \frac{\nu_T}{\sigma_\varepsilon}\right)\frac{\partial \tilde{\varepsilon}}{\partial x_2}\right], \tag{2.2c}$$

see Ref. 39. Here, k is the turbulent kinetic energy and $\tilde{\varepsilon}$ is the isotropic turbulent dissipation, i.e. the term that controls the dissipation rate of k. The isotropic dissipation (which is zero at the wall) is related to the dissipation ε by $\varepsilon = \varepsilon_0 + \tilde{\varepsilon}$, where ε_0 is the value

of the turbulent dissipation at $x_2 = 0$. The system (2.2a)-(2.2c) contains several closure coefficients and empirical damping functions, which act directly on these coefficients. Without the damping functions the $k-\varepsilon$ model would not be able to provide accurate predictions in the viscous near-wall region [39]. The Launder-Sharma $k-\varepsilon$ model [16] is obtained by specifying these damping functions as follows

$$f_\mu = \exp\left[-3.4/(1+Re_T/50)\right],\ f_1 = 1,$$
$$f_2 = 1-0.3\exp\left[-Re_T^2\right],\ \varepsilon_0 = 2\nu\left(\frac{\partial\sqrt{k}}{\partial x_2}\right)^2,$$
$$E = 2\nu\nu_T\left(\frac{\partial^2\bar{u}}{\partial x_2^2}\right)^2, \tag{2.3}$$

where $Re_T \equiv k^2/\tilde{\varepsilon}\nu$. In the case of the Launder-Sharma $k-\varepsilon$ model, the closure coefficients have the following values

$$C_\mu = 0.09, \quad C_{\varepsilon 1} = 1.44, \quad C_{\varepsilon 2} = 1.92,$$
$$\sigma_k = 1.0, \quad \sigma_\varepsilon = 1.3. \tag{2.4}$$

We do not expect these values to be generally applicable 'best' values, and other $k-\varepsilon$ models do use different values. For instance, the Jones-Launder model [11], which only differs from (2.3) by a slightly different f_μ, uses

$$C_\mu = 0.09, \quad C_{\varepsilon 1} = 1.55, \quad C_{\varepsilon 2} = 2.0,$$
$$\sigma_k = 1.0, \quad \sigma_\varepsilon = 1.3. \tag{2.5}$$

We refer to Wilcox [39] for further discussion on $k-\varepsilon$ type models and their limitations.

2.2.1. CLASSICAL IDENTIFICATION OF CLOSURE COEFFICIENTS

The values of the closure coefficients in (2.4) are classically chosen by reference to fundamental flow problems. We illustrate how the nature of the coefficients leads to some ambiguity regarding their values, and how flow independent single best values are unlikely to exist. One such a fundamental flow problem often considered is homogeneous, isotropic, decaying turbulence. In this case the k and ε equations (2.1a)-(2.2c) (without damping functions) simplify to

$$\frac{dk}{dt} = -\varepsilon, \tag{2.6}$$
$$\frac{d\varepsilon}{dt} = -C_{\varepsilon 2}\frac{\varepsilon^2}{k}. \tag{2.7}$$

These equations can be solved analytically to give

$$k(t) = k_0\left(\frac{t}{t_0}\right)^{-n}, \tag{2.8}$$

with reference time $t_0 = nk_0/\varepsilon_0$ and $n = 1/(C_{\varepsilon 2} - 1)$. And thus,

$$C_{\varepsilon 2} = \frac{n+1}{n}. \tag{2.9}$$

The standard value for n is such that $C_{\varepsilon 2} = 1.92$. However, this is by no means a hard requirement and other models do use different values for $C_{\varepsilon 2}$. For instance, the RNG $k-\varepsilon$ model uses a modified $\tilde{C}_{\varepsilon 2} = 1.68$ and the $k-\tau$ model (essentially a $k-\varepsilon$ model rewritten in terms of $\tau = k/\epsilon$ [33]) uses $C_{\varepsilon 2} = 1.83$ [39]. Also, the experimental data from Ref. 20 suggests that most data agrees with $n = 1.3$, which corresponds to $C_{\varepsilon 2} = 1.77$.

The coefficient C_μ is calibrated by considering the approximate balance between production and dissipation which occurs in free shear flows, or in the inertial part of turbulent boundary layers. This balance can be expressed as

$$\mathscr{P} = \nu_t \left(\frac{\partial \bar{u}_1}{\partial x_2}\right)^2 = C_\mu \frac{k^2}{\varepsilon}\left(\frac{\partial \bar{u}_1}{\partial x_2}\right)^2 = \varepsilon. \tag{2.10}$$

Equation (2.10), can be manipulated together with the turbulent-viscosity hypothesis $-\overline{u_1' u_2'} = \nu_t \partial \bar{u}_1/\partial x_2$ to yield $-\overline{u_1' u_2'} = \varepsilon(\partial \bar{u}/\partial x_2)^{-1}$, which in turn yields

$$C_\mu = \left(\frac{\overline{u_1' u_2'}}{k}\right)^2. \tag{2.11}$$

The DNS data from Ref. 13 can be used to show that $\overline{u_1' u_2'} \approx -0.30k$ (except close to the wall), such that $C_\mu = 0.09$ is the recommended value. Again however, different models use different values for C_μ, such as $C_\mu \approx 0.085$ in the case of the RNG $k-\varepsilon$ model.

Another fundamental flow to be considered is fully developed (so $Dk/Dt = D\varepsilon/Dt = 0$) channel flow. The resulting simplified governing equations allows us to find the following constraint amongst several parameters [26]

$$\kappa^2 = \sigma_\varepsilon C_\mu^{1/2}(C_{\varepsilon 2} - C_{\varepsilon 1}), \tag{2.12}$$

where κ is the von-Karman constant. It should be noted that the suggested values (2.4) satisfy this constraint only approximately. Using (2.4) in (2.12) gives $\kappa \approx 0.43$, instead of the 'standard' value of 0.41.

The following constraint (between $C_{\varepsilon 1}$ and $C_{\varepsilon 2}$) can be found by manipulating the governing equations of uniform (i.e. $\partial \bar{u}_1/\partial x_2 =$ constant) shear flows [26]

$$\left(\frac{\mathscr{P}}{\varepsilon}\right) = \frac{C_{\varepsilon 2} - 1}{C_{\varepsilon 1} - 1}, \tag{2.13}$$

where the non-dimensional parameter $\mathscr{P}/\varepsilon$ is the ratio between the turbulent production $\mathscr{P}$ and dissipation ε. Tavoulakis et. al. [36] measured $\mathscr{P}/\varepsilon$ for several uniform shear flows. They reported values between 1.33 and 1.75, with a mean around 1.47. Note however, that (2.13) becomes 2.09 with the standard values for $C_{\varepsilon 1}$ and $C_{\varepsilon 2}$, which is significantly different from the mentioned experimental values.

The parameter σ_k can be considered as a turbulent Prandtl number, defined as the ratio of the momentum eddy diffusivity and the heat-transfer eddy diffusivity. These

quantities are usually close to unity, which is why the standard value for σ_k is assumed to be 1.0. As noted in Ref. 25, no experimental data can be found to justify this assumption. And again, we see a range of recommended values amongst the different variations of the $k-\varepsilon$ model. For instance, the RNG $k-\varepsilon$ model uses $\sigma_k = 0.72$ [39].

The parameter σ_ε controls the diffusion rate of ε, and its value can be determined by using the constraint (2.12), i.e.

$$\sigma_\varepsilon = \frac{\kappa^2}{C_\mu^{1/2}(C_{\varepsilon 2} - C_{\varepsilon 1})}. \tag{2.14}$$

Finally, it should be noted that the 'constant' value of the von Karman constant (0.41) is being questioned. An overview of experimentally determined values for κ is given in Ref. 41, which reports values of κ in $[0.33, 0.45]$

2.2.2. NUMERICAL SOLUTION OF THE $k-\varepsilon$ MODEL

To obtain efficient numerical solutions for the boundary-layer problem (2.1a)-(2.2c) we used the program EDDYBL of Ref. 38, which we modified slightly to make it more suitable for our purpose. EDDYBL is a two-dimensional (or axisymmetric), compressible (or incompressible) boundary-layer program for laminar, transitional and turbulent boundary layers. This program has evolved over three decades and is based on a code originally developed by Price and Harris in 1972 [38]. The advantage of using a boundary-layer approximation rather than a full RANS code, is that a boundary-layer code allows for quicker numerical simulations, and thus avoids the need of a surrogate model.

Parabolic systems of equations such as the boundary-layer equations can, in general, be solved using unconditionally stable numerical methods. EDDYBL uses the variable-grid method of Blottner [1], which is a second-order accurate finite-difference scheme designed to solve the turbulent boundary-layer equations. This scheme uses a three-point forward-difference formula in the stream-wise direction, central differencing for the normal convection term and conservative differencing for the diffusion terms.

We verify that the discretization error is small enough such it does not dominate over the uncertainties we want to quantify. The rate at which the grid-point spacing increases in normal direction is set such that the first grid point satisfies $\Delta y^+ < 1$, which provides a good resolution in the viscous layer. Initially, the maximum number of points in the normal direction is set to 101, although EDDYBL is capable of adding more points if needed to account for boundary-layer growth. The maximum number of stream-wise steps is set high enough such that EDDYBL has no problems reaching the specified s_{stop}, i.e. the final arc length in stream-wise direction. Using this setup we verify that the discretization errors are substantially smaller than the uncertainties present in the model and data. To give an example of the magnitude of the discretization error, we computed the boundary layer over the curved airfoil-shaped surface of Ref. 29 with $s_{stop} = 20.0\,[ft]$ for both our standard mesh with the first grid point below $y^+ = 1$, and on a finer mesh with the first 15 points below $y^+ = 1$. The maximum relative error between the two predicted velocity profiles was roughly 0.3%, which is well below the expected variance in the model output that we might see due to for instance the uncertainty in the closure coefficients. Discretization error is assumed to be negligible hereafter.

2.3. EXPERIMENTAL DATA

EDDYBL comes with configuration files which mimic the experiments described in the 1968 AFOSR-IFP-Stanford conference proceedings [4]. From this data source, we selected one zero pressure-gradient flow, and 12 flows from other types of available pressure gradients, which range from favourable ($d\bar{p}/dx < 0$) to strongly adverse ($d\bar{p}/dx > 0$) gradients. These 13 flows are described in table 2.1. The identification number of each flow is copied from Ref. 4. According to Ref. 37, the flows are identified as being 'mildly adverse', 'moderately adverse' etc, based upon qualitative observations of the velocity profile shape with respect to the zero-pressure gradient case. We plotted the experimentally determined, non-dimensional, streamwise velocity profiles in Figure 2.1. As usual, the normalized streamwise velocity is defined as $u^+ \equiv \bar{u}_1/\sqrt{\tau_w/\rho}$, where τ_w is the wall-shear stress. The normalized distance to the wall, displayed on the horizontal axis of Figure 2.1, is $y^+ \equiv x_2\sqrt{\tau_w/\rho}/\nu$. Too much weight should not be given to the classifications of the severity of the adverse gradients, since some flows (such as 2400) experience multiple gradient types along the spanwise direction. Also, when we try to justify the classification based upon the velocity profile shape we find some discrepancies. For instance, based upon the profile shape alone, we would not classify flow 1100 as mildly adverse, or 2400 as moderately adverse.

To obtain an estimate of the spread in closure coefficients, we calibrate the $k-\varepsilon$ model for each flow of table 2.1 separately, using one velocity profile as experimental data.

Use of experimental data in the viscous wall region is worthy of a separate discussion. On one hand, Reynolds stresses tend to zero when approaching the wall, so that calibrating the turbulence model using data from the first few wall units does not really make sense; moreover, in the whole viscous layer the model is dominated by damping functions (2.3), not calibrated here, introduced to enforce asymptotic consistency as $y^+ \to 0$. As a consequence, little information is obtained from the measurements here. On the other hand, obtaining reliable measurements close to the wall can be difficult due to limited spatial resolution, see e.g. [10]. Therefore, most experimental datasets of [4] do not include points in this region. Additional difficulties may arise according to the experimental technique in use: for instance, outliers due to additional heat losses near the wall are not uncommon in hot-wire measurements, and special corrections are needed to fix the problem [15]. Numerical results from Direct Numerical Simulations (DNS) could be used instead of experimental data sets. Nevertheless, there is little DNS data with high enough values of the friction Reynolds number Re_τ to allow for a sufficiently extended logarithmic region (see [24] for a recent survey).

More generally, our goal is to introduce and test a methodology that can be applied to complex, high-Re flows for which DNS is simply not feasible. The effect of excluding near-wall data from the calibration data set is investigated through numerical experiments presented in Section 2.5.2.

2.4. METHODOLOGY: CALIBRATION AND PREDICTION

Our methodology consists of two major parts: calibration and prediction. In the calibration stage (Section 2.4.1) posterior distributions on closure coefficients are identified

Table 2.1: Flow descriptions, source [4].

Identification	Type	Description
1400	Zero	Equilibrium boundary layer at constant pressure
1300	Fav	Near-equilibrium boundary layer in moderate negative pressure gradient
2700	Fav	Equilibrium boundary layer in mild negative pressure gradient
6300	Fav	Near-equilibrium boundary layer growing beneath potential flow on model spillway
1100	Mild adv	Boundary layer in diverging channel
2100	Mild adv	Boundary layer on large airfoil-like body; pressure gradient first mildly negative, then strongly positive, with eventual separation
2500	Mild adv	Equilibrium boundary layer in mild positive pressure gradient
2400	Mod adv	Initial equilibrium boundary layer in moderate positive pressure gradient; pressure gradient abruptly decreases to zero, and flow relaxes to new equilibrium
2600	Mod adv	Equilibrium boundary layer in moderate positive pressure gradient
3300	Mod adv	Boundary layer, initially at constant pressure, developing into equilibrium flow in moderate positive pressure gradient
0141	Str adv	Boundary-layer with strong adverse pressure gradient, source [14]
1200	Str adv	Boundary layer in diverging channel with eventual separation
4400	Str adv	Boundary layer in strong positive pressure gradient

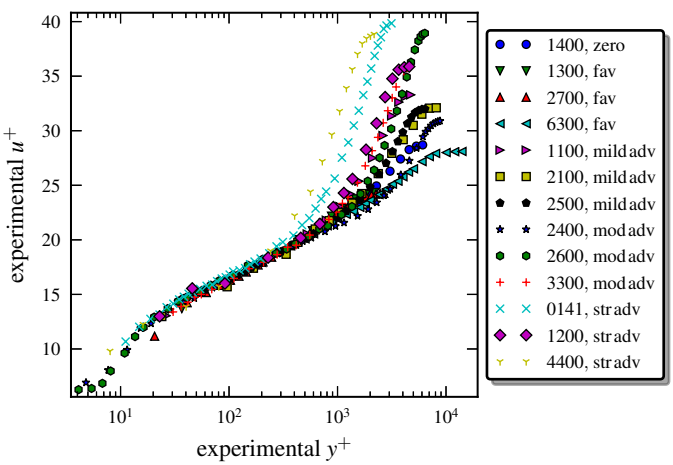


Figure 2.1: The experimental data from Ref. 4.

for each of a set of 13 boundary-layer flows. These posteriors are then summarized with Highest Posterior Density (HPD) intervals in Section 2.4.3. The results will give a first indication of the extent to which posterior distributions of turbulence closure coefficients $\boldsymbol{\theta}$ are case-dependent. This stage can also be seen as a setup stage for the predictive mechanism of our methodology. To make predictions of a new, unseen flow, we combine the 13 posterior distributions for $\boldsymbol{\theta}$ using p-boxes (Section 2.4.4). These p-boxes encapsulate the effect of both the 13 individual posterior uncertainties (due to the data not exactly identifying a single optimal $\boldsymbol{\theta}$, but rather a probability distribution different from a Dirac function), and the variability of $\boldsymbol{\theta}$ between cases.

2.4.1. CALIBRATION FRAMEWORK

Bayesian calibration requires selection of joint prior distribution for the calibration parameters and a joint pdf (or statistical model) describing the likelihood function.

In our turbulence model calibration we have a large number of accurate observations, and a belief that model inadequacy will dominate the error between reality and prediction. In this situation we expect the prior on closure coefficients to be substantially less influential than the joint pdf. We therefore impose uniform priors on closure coefficients, on intervals chosen to: (i) respect mild physical constraints, and (ii) ensure the solver converges in most cases.

After the calibration we verify that the posterior pdf is not unduly constrained by the prior intervals: if we find that one of the informed marginal posteriors is truncated, we simply re-perform the calibration with a wider prior range for the truncated coefficient. We also perform a posterior model checking, in the sense that we verify that sufficient overlapping between the posterior model distribution and the calibration data interval exists.

To specify the joint likelihood we start from the framework of Cheung et. al. [3], who use a multiplicative model inadequacy term, modelled as a Gaussian process in the wall-normal direction. Multiplicative error models are less common than additive errors (like Equation (1.6) in Section 1.3), but may be useful in many engineering situations: here, it allows to enforce automatically that the random velocity profiles satisfy a no-slip wall condition (see [3]). By considering multiple different flows we have additional modelling choices. Unlike Cheung et. al., we choose to calibrate closure coefficients and model-inadequacy hyper-parameters independently for each flow, and examine the variability between flows in a post-calibration step.

Let the experimental observations from flow-case $k \in \{1, \cdots, N_C\}$ be $\mathbf{z}_k = [z_k^1, \cdots, z_k^{N_k}]$. Here N_k is the number of scalar observations in flow-case k, and z_k^i is the scalar observation at location $y_k^{+,i} > 0$, where in the following we work in y^+-units. Following Ref. 3, we assume the observation noise $\boldsymbol{\lambda}_k = [\lambda_k^1, \cdots, \lambda_k^{N_k}]$ is known and uncorrelated at all measurement points. Furthermore, the closure coefficients and flow parameters for case k are denoted $\boldsymbol{\theta}_k$ and $\mathbf{t}_k$ respectively. The flow parameters include specification of the pressure-gradient as a function of the x-coordinate. The observation locations $\mathbf{y}_k^+$, noise $\boldsymbol{\lambda}_k$, and flow parameters $\mathbf{t}_k$ are modelled as precisely known explanatory variables. In the case that substantial uncertainties existed in the experiments these could be modelled stochastically as nuisance parameters.

A statistical model accounting for additive Gaussian noise in the observations via an additive term and model inadequacy via a multiplicative term is: $\forall k \in \{1, \cdots, N_C\}$

$$\mathbf{z}_k = \zeta_k(\mathbf{y}_k^+) + \mathbf{e}_k, \tag{2.15a}$$

$$\zeta_k(\mathbf{y}_k^+) = \eta_k(\mathbf{y}_k^+) \cdot u^+(\mathbf{y}_k^+, \mathbf{t}_k; \boldsymbol{\theta}_k), \tag{2.15b}$$

where $u^+(\cdot,\cdot;\cdot)$ is the simulation code output, and the multiplication is applied element-wise to its arguments. Observational noise is modelled as

$$\mathbf{e}_k \sim \mathrm{N}(\mathbf{0}, \Lambda_k), \quad \Lambda_k := \mathrm{diag}(\boldsymbol{\lambda}_k),$$

and the model-inadequacy term $\eta_k(\cdot)$ is a stochastic process in the wall-distance y^+ modelling the relative error between the code output and the true process. Therefore (2.15a) represents the difference between the true process ζ_k and the measurement observations, and (2.15b) the difference between ζ_k and model predictions. Together they relate $\boldsymbol{\theta}_k$ to $\mathbf{z}_k$.

Cheung et. al. consider three models of this form, which differ only in the modelling of η. They compared the posterior evidence, and showed that modelling η as a correlated Gaussian process yielded by far highest evidence of the three models considered [3]. We therefore adopt the same strategy and model each η_k as a Gaussian process with unit mean (dropping the subscript k for convenience):

$$\eta \sim \mathrm{GP}(1, c_\eta), \tag{2.16}$$

and the simple, homogeneous covariance function

$$c_\eta(y^+, y^{+\prime} \mid \boldsymbol{\gamma}) := \sigma^2 \exp\left[-\left(\frac{y^+ - y^{+\prime}}{10^\alpha l}\right)^2\right], \tag{2.17}$$

where $y+$ and $y^{+\prime}$ represent two different measurement points along the velocity profile, and l is a user-specified length scale. We fix this length scale to 5.0, which is the y^+ value that denotes the end of the viscous wall region. The smoothness of the model-inadequacy term is controlled by the correlation-length parameter α, and its magnitude by σ. Both α and σ are unknown, and must be obtained via calibration from the data, and form a hyper-parameter vector $\boldsymbol{\gamma} := [\alpha, \sigma]$.

A more boundary-layer specific model than (2.16), is described in Ref. 23. It attempts to account for the multi-scale structure of the boundary layer by allowing the correlation length to vary in y^+ direction. Together (2.15b) and (2.16) imply the relative model inadequacy σ is independent of y^+, and that the correlation length is the same throughout the boundary layer. This model may be generalized to multiple dimensions by replacing $\eta(\cdot)$ with a multi-dimensional Gaussian process. See [12] for a thorough discussion of the role of the covariance term.

A consequence of the above modelling choices is that the true process ζ is also modelled as a Gaussian process:

$$\begin{aligned} \zeta \mid \boldsymbol{\theta}, \boldsymbol{\gamma} &\sim \mathrm{GP}(\mu_\zeta, c_\zeta) \\ \mu_\zeta(y^+ \mid \boldsymbol{\theta}) &= u^+(y^+, \mathbf{t}; \boldsymbol{\theta}) \\ c_\zeta(y^+, y^{+\prime} \mid \boldsymbol{\theta}, \boldsymbol{\gamma}) &= u^+(y^+, \mathbf{t}; \boldsymbol{\theta}) \cdot c_\eta(y^+, y^{+\prime} \mid \boldsymbol{\gamma}) \cdot \\ &\qquad u^+(y^{+\prime}, \mathbf{t}; \boldsymbol{\theta}), \end{aligned} \tag{2.18}$$

which is still centered around the code output. The assumption of normality is made mainly for convenience, and more general forms are possible. See Section 2.4.5 for a discussion on the choice of statistical model.

The likelihood evaluated at the measurement locations $y^{+,i}$ can now be written for each flow case k independently as:

$$
\begin{aligned}
p(\mathbf{z} \mid \boldsymbol{\theta}, \boldsymbol{\gamma}) &= \frac{1}{\sqrt{(2\pi)^N |K|}} \exp\left[-\frac{1}{2}\mathbf{d}^T K^{-1}\mathbf{d}\right], \\
\mathbf{d} &:= \mathbf{z} - u^+(\mathbf{y}^+) \\
K &:= \Lambda + K_\zeta .
\end{aligned} \tag{2.19}
$$

where

$$
\left[K_\zeta\right]_{ij} := c_\zeta(y^{+,i}, y^{+,j} | \boldsymbol{\theta}, \boldsymbol{\gamma}).
$$

Since in general the computational grid does not coincide with measurement locations we linearly interpolate the code output at $y^{+,i}$ where needed.

Note that the statistical model is based only on velocity data, and does not include constraints on other physical quantities, like e.g. the Reynolds stresses or the turbulent kinetic energy k. This choice is consistent with the fact that our version of $k-\varepsilon$ is not realizable, and as such it does not contain any modelling assumption to prevent the normal Reynolds stresses from becoming negative, but only enforces them to satisfy physical constraints through the application of limiters. If we were to calibrate realizable $k-\varepsilon$ we would try to preserve the realizability conditions. This would require adding constraints to the likelihood function so that zero probability is assigned to parameter combinations leading to unrealizable turbulence states.

2.4.2. PRIORS FOR $\boldsymbol{\theta}$ AND $\boldsymbol{\gamma}$

Unlike Cheung et. al., we do not treat all closure coefficients as independent random variables in the prior. Instead we use the physical relations described in Section 2.2.1 to constrain the value of two closure coefficients. Specifically we fix $C_{\epsilon 1}$, by rewriting (2.13) as

$$
C_{\epsilon 1} = \frac{C_{\epsilon 2}}{\mathscr{P}/\varepsilon} + \frac{\mathscr{P}/\varepsilon - 1}{\mathscr{P}/\varepsilon}, \tag{2.20}
$$

where, similar to Ref. 25, we fix the ratio $\mathscr{P}/\varepsilon$ to 2.09. In our results, this choice locates the mode of the posterior for $C_{\epsilon 2}$ relatively close to the standard value of 1.92. If we instead would have used a different (experimentally determined) value of $\mathscr{P}/\varepsilon$, the mode $C_{\epsilon 2}$ would be located elsewhere. Whether or not our choice is reasonable has to be determined by the ability of the posterior distributions to capture the observed data, as outlined in Section 2.5.3. Two other possibilities we do not employ are: (i) to move $\mathscr{P}/\varepsilon$ into $\boldsymbol{\theta}$ and calibrate it along with the other parameters with some suitable prior, or (ii) model $\mathscr{P}/\varepsilon$ as an aleatory uncertainty, using the $\mathscr{P}/\varepsilon$ data from Ref. 36 to construct an

approximate pdf $p(\mathscr{P}/\varepsilon)$. Also, we fix σ_ε using (2.12). Such a choice avoids running the boundary-layer code with non-physical parameter combinations.

All priors, for both the closure coefficients $\boldsymbol{\theta}$ and hyper-parameters $\boldsymbol{\gamma}$, are independent uniform distributions. The choice of interval end-points was made based on three factors: the spread of coefficients recommended in the literature, the range of coefficients for which the solver was stable, and avoidance of apparent truncation of the posterior at the edge of the prior domain. The range we used is specified in Table 2.2. We

Table 2.2: The empirically determined range (absolute and relative to nominal value) of the uniform prior distributions.

coefficient	left boundary	right boundary
$C_{\epsilon 2}$	1.15 (-40%)	2.88 (+50%)
C_μ	0.054 (- 40 %)	0.135 (+50 %)
σ_k	0.450 (-45 %)	1.15 (+50 %)
κ	0.287 (-30 %)	0.615 (+50 %)
σ	0.0	0.1
$\log\alpha$	0.0	4.0

chose uniform distributions because we lack confidence in more informative priors for these parameters. We note however that some reasonable, informative priors can be obtained using the classical framework for coefficient identification (c.f. Section 2.2.1) in combination with multiple experimental measurements from different sources [25].

To obtain samples from the posterior distributions $p(\boldsymbol{\theta} \mid \mathbf{z})$, we employed the Markov-chain Monte Carlo (McMC) method [9]. We subsequently approximated the marginal pdf of each closure coefficient using kernel-density estimation, using the last 5,000 (out of a total of 40,000) samples from the Markov chain. It was observed that at 35,000 samples, the Markov chain was in a state of statistical convergence.

2.4.3. SUMMARIZING POSTERIORS: HPD INTERVALS

The methodology of Section 2.4.1 will be applied to 13 flow cases, resulting in 13 posteriors on $[\boldsymbol{\theta}, \boldsymbol{\gamma}]$. The large amount of information (see e.g. Figure 2.2 and 2.3 in the results section) is difficult to visualize. In other words the posteriors must be summarized, and we do this with intervals. In the remainder we make the assumption that closure coefficients are approximately independent and uncorrelated. This is justified by Figure 2.3. This assumption allows us to work with 1d marginal pdfs of the coefficients, rather than the full multi-dimensional posteriors.

We compute Highest Posterior Density (HPD) intervals on 1d marginals to summarize the posteriors. An HPD interval is a Bayesian credible interval which satisfies two main properties, namely:

1. The density for every point inside the interval is greater than that for every point outside the interval.

2. For a given probability content $1-\beta$, $\beta \in (0,1)$, the interval is of the shortest length.

We use the algorithm of Chen et. al. [2] to approximate the HPD intervals using the obtained McMC samples. To do so, we first sort the samples of the Q closure coefficients θ^q, $q = 1,2,\cdots,Q$ in ascending order. Then, if we let $\{\theta^q_j,\ j = 1,2,\cdots,J\}$ be the McMC samples from $p\left(\theta^q \mid \mathbf{z}\right)$, the algorithm basically consists of computing all the $1-\beta$ credible intervals and selecting the one with the smallest width. For a given j, we can use the empirical cumulative-distribution function to approximate the $1-\beta$ interval by computing the first θ^q_s which satisfies the inequality

$$\sum_{i=j}^{J} \mathbb{1}_{\theta^q_i \leq \theta^q_s} \geq \left[J\left(1-\beta\right)\right], \tag{2.21}$$

where $\mathbb{1}_{\theta^q_i \leq \theta^q_s}$ is the indicator function for $\theta^q_i \leq \theta^q_s$ and $\left[J\left(1-\beta\right)\right]$ is the integer part of $J\left(1-\beta\right)$. Secondly, if we let $\theta^q_{(i)}$ be the smallest of a set $\{\theta^q_i\}$, then the first θ^q_s for which (2.21) is satisfied simply is $\theta^q_{([J(1-\beta)])}$. Thus, the j^{th} credible interval is given by $\theta^q_{(j+[J(1-\beta)])} - \theta^q_{(j)}$ and the HPD interval for θ^q is found by solving

$$\min_j \theta^q_{(j+[J(1-\beta)])} - \theta^q_{(j)}, \quad 1 \leq j \leq J - \left[J\left(1-\beta\right)\right]. \tag{2.22}$$

The algorithm of Chen assumes a uni-modal posterior pdf, although it could possibly be extended to deal with multi-modal pdfs [2]. This assumption is shown to be justified in our case in Section 2.5.5.

2.4.4. PREDICTIVE FRAMEWORK: P-BOXES

So far we have only discussed identification of $\boldsymbol{\theta}$ for flows with data. The final purpose of this work is to establish uncertainties on $k-\varepsilon$ model predictions under flow conditions $\mathbf{t}^*$ at which no measurements are available. To achieve this we must assess the effect of all sources of uncertainty on the model prediction at $\mathbf{t}^*$. We require a method that can simultaneously represent solution variability within- and between the posteriors $p(\boldsymbol{\theta},\boldsymbol{\gamma} \mid \mathbf{z}_k,\mathbf{t}_k)$, $k = 1,2,\cdots,N_C$, where $N_C = 13$ in this work.

Our approach is to construct a probability box (p-box) for the output of the model at the new condition $\mathbf{t}^*$, using coefficients sampled from the HPD intervals of each of the 13 cases. A p-box is commonly used to visualize possible outcomes due to a combination of epistemic and aleatory uncertainty [22]. Examples of p-boxes are shown in Figure 2.13. In our case - roughly speaking - the slant of the box represents the width of individual HPD intervals, and the width of the box the variability between HPD intervals obtained from calibrations against different data sets. More precisely they can be interpreted as bounds on the ζ (or u^+) value at any particular probability level, and therefore can be used to construct confidence intervals on the true process ζ or code output u^+.

We define our p-box as follows: For flow case k, let $\boldsymbol{\Theta}^{50}_k$ be a uniformly distributed random variable on the box given by the 50% HPD intervals of the posterior for $\boldsymbol{\theta}$. Propagate this variable through the flow-code for u^+ to obtain

$$Z_k(y^+_*) := u^+(y^+_*, t^*; \boldsymbol{\Theta}^{50}_k),$$

a random variable representing the effect of posterior uncertainty in case k on the model output at conditions t^*. Here y_*^+ is the location at which an uncertainty estimate is required, which need not correspond to a sample location in any of the calibrated flow cases. Let $F_k(u^+) = \mathbb{P}(Z_k \leq u^+)$ be the cumulative density function of Z_k. Then the p-box $\mathbb{D}$ is defined by

$$\begin{aligned} \mathbb{D}(u^+) &:= \{r \in [0,1] \mid \underline{F}(u^+) \leq r \leq \overline{F}(u^+)\}, \\ \overline{F}(u^+) &:= \min_{k\in\{1,\ldots,13\}} F_k(u^+), \\ \underline{F}(u^+) &:= \max_{k\in\{1,\ldots,13\}} F_k(u^+) \end{aligned} \tag{2.23}$$

i.e. the envelope formed by this collection of k cdfs. To construct a worst-case 90% confidence interval we find the end-points $\underline{u}^+$ and $\overline{u}^+$ such that

$$\begin{aligned} \underline{F}(\underline{u}^+) &= 0.05, \\ \overline{F}(\overline{u}^+) &= 0.95. \end{aligned}$$

The interval $\mathbb{I} = [\underline{u}^+, \overline{u}^+]$ is our final estimate of solution uncertainty due to modelling inadequacy in the $k-\varepsilon$ model for $u^+(y_*^+)$ at conditions $\mathbf{t}^*$.

To construct the p-box in (2.23) numerically we use empirical cdfs:

$$F_k(u^+) = \mathbb{P}\left(Z_k \leq u^+\right) \approx \frac{1}{S}\sum_{j=1}^{S} \mathbb{1}_{u_j^+ \leq u^+}, \tag{2.24}$$

where u_j^+ are S samples from Z_k obtained using Monte-Carlo. An approximation to $\mathbb{D}$ is then readily calculated.

Note that because Z_k is based only on the flow-code output and not on the true process ζ, the effect of the model inadequacy term η is not included in our estimates. If a variable other than u^+ were of interest, we could define the p-box in exactly the same way (using skin-friction C_f as an example):

$$Y_k(x) := C_f(x, t^*; \boldsymbol{\Theta}_k^{50}).$$

That this is possible is a consequence of representing model inadequacy via closure coefficients. It is not possible if we base estimates on η-like terms. For the validity of the confidence intervals we are relying on the uncertainties in $\boldsymbol{\theta}$ accounting for the majority of model error. We acknowledge that the choice of 50% HPD intervals plays a role in the p-box size, and this tuning parameter could be eliminated by replacing $\boldsymbol{\Theta}_k^{50}$ and $\boldsymbol{\Gamma}_k^{50}$ by the posteriors for case k, at the cost of increasing the size of the p-boxes.

2.4.5. DISCUSSION

In the above we are attempting to capture model error in two different ways:

1. Via the traditional (Kennedy and O'Hagan) Gaussian process (GP) "model discrepancy" term, $\eta(\cdot)$, and

2. Via representing the variability of θ across flows (using p-boxes).

In doing so we have introduced some redundancy, since model errors can be captured by either one of these two methods, or a combination of them both. Our aim is to capture the majority of model error via θ-variability, for the following reasons.

The Gaussian process $\eta(\cdot)$ has a very general form, able to capture a large space of smooth model-data error as a function of y^+. It has no physical content (other than the assumption of smoothness). Its form is geometry and flow-topology dependent: e.g. in our case the calibrated $\eta(\cdot)$ can not be used to predict error in any other variable than u^+, in any other geometry. The natural generalization to more complex flows is a multi-dimensional Gaussian process, which would necessitate estimation of a large number of hyper-parameters.

In contrast, varying $\boldsymbol{\theta}$ does not allow for general velocity profiles. Only those profiles that satisfy the governing equations (for some value of $\boldsymbol{\theta}$) are represented, hence certain constraints based on physical modelling are automatically satisfied. Since each component of $\boldsymbol{\theta}$ corresponds to some empirical modelling assumption, one might also see $\boldsymbol{\theta}$ as an (incomplete) parameterization of modelling assumptions. Furthermore, irrespective of flow topology, uncertainties on $\boldsymbol{\theta}$ can be propagated through the simulation code to estimate model error in the output. That is, distributions of $\boldsymbol{\theta}$ derived from calibration on a class of boundary-layer flows, can be applied to estimate model error on any flow. The accuracy will depend on the extent to which boundary-layer modelling error dominates in the new flow, e.g. the error in fully attached subsonic flows over aerofoils might be successfully judged.

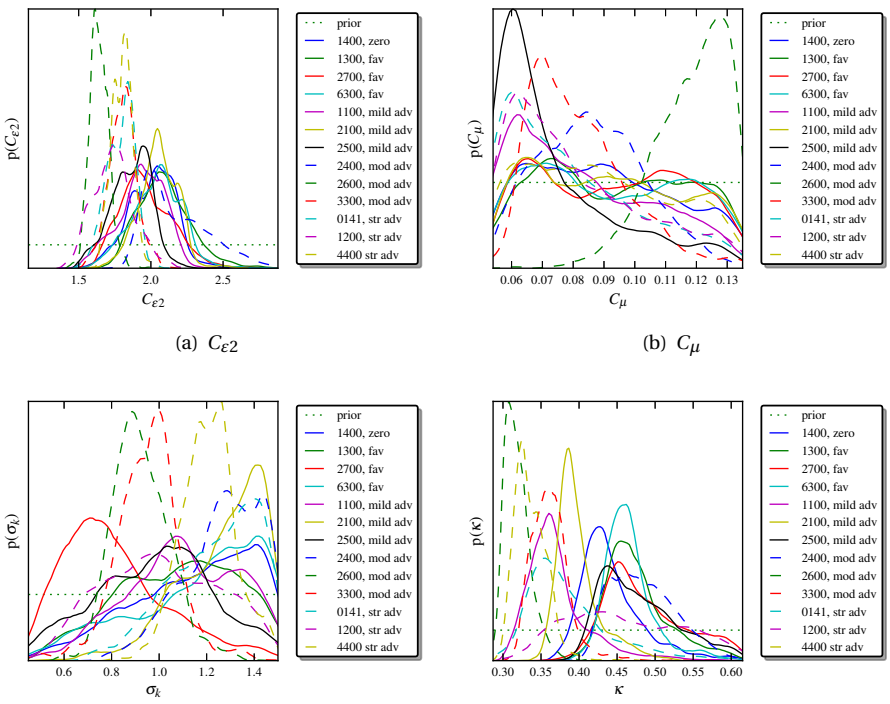
In short we consider representing model error in model-coefficient space has several advantages, compared to representing it in the data space. However all error can not be represented in $\boldsymbol{\theta}$-space – e.g. error due to model form is not accounted for. Therefore, to characterize the real relationship between data and model output in the statistical model, and to prevent over-fitting, η is necessary in (2.15b). Of course any choice of statistical model represents a modelling assumption, and in Section 2.5.6 we examine the sensitivity of $\boldsymbol{\theta}$-posteriors to the choice of η.

2.5. RESULTS AND DISCUSSION

2.5.1. MARGINAL POSTERIOR PDFS

Calibration of the $k-\varepsilon$ model, using the experimental data described in Section 2.3, and the statistical model of Section 2.4 was performed. The marginal posterior pdfs of all four parameters in $\boldsymbol{\theta}$, for all of the 13 calibration cases are shown in Figure 2.2. There we see that the data has been informative for $C_{\varepsilon 2}$ in all cases, resulting in sharply peaked posteriors. This is in contrast to C_μ, which has been only weakly informed. Experience suggests that for these cases the level of informativeness is predicted by the sensitivity of the data $u^+(\boldsymbol{\theta})$ to the individual parameters. Parameters with the largest sensitivities are the best identified, see section 2.5.4. The calibrations have provided us with a bit more information regarding σ_k, since more posterior distributions show clear modes compared to the C_μ results. The spread of coefficients between cases is also visible. For instance $C_{\varepsilon 2}$ values cluster around the center of the prior interval, while κ is sharply identified at distinctly different values for the different flow conditions.

To examine dependencies between variables, we plot two-variable marginal posteri-



(a) $C_{\varepsilon 2}$ (b) C_μ

(c) σ_k (d) κ

Figure 2.2: The marginal posterior distributions of the coefficients for the 13 cases of Table 2.1.

ors of all pairs of closure coefficients, see Figure 2.3. This figure is constructed from the McMC traces for flow 3300, and is typical of the other flow cases which are not shown. Any correlation between two coefficients will be visible in such a plot. As can be seen, there is a weak negative correlation between $C_{\varepsilon 2}$ and C_μ, and also between $C_{\varepsilon 2}$ and σ_k. However, overall the coefficients appear approximately independent, as they were in the prior. This observed independence may not be a coincidence, it may be a result of the design of the turbulence model in which the individual coefficients parameterize separate modelling assumptions.

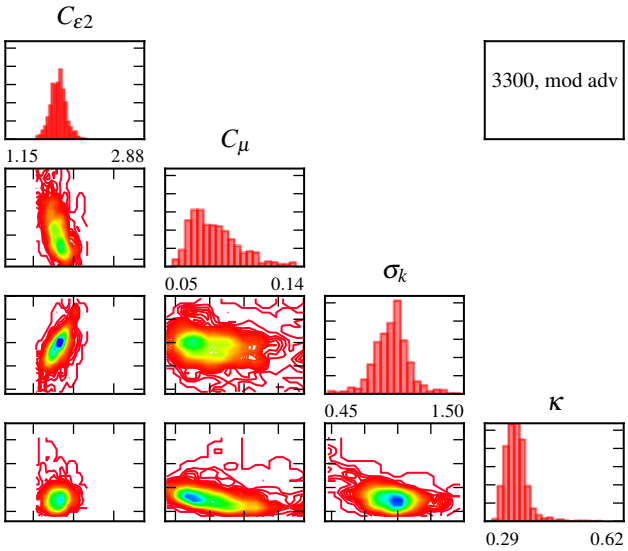
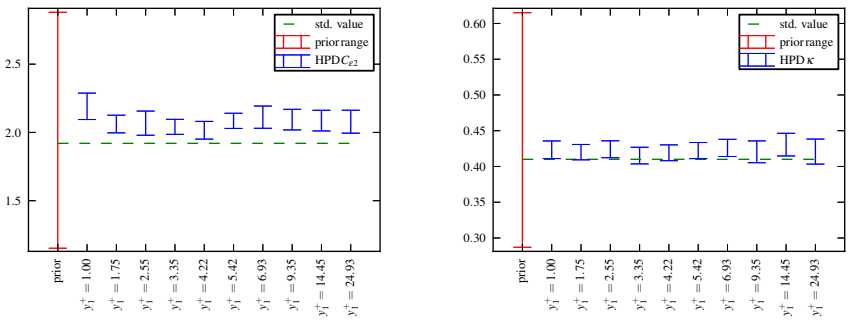


Figure 2.3: A two-dimensional contour plot of the posterior $\boldsymbol{\theta}$ samples from flow 3300.

2.5.2. y^+-CUTOFF SENSITIVITY

In Section 2.3 we mention that we omit data in the viscous-wall region. To examine if our posterior distributions are critically affected by not including near-wall data, we perform a simple sensitivity analysis using flow case 1400. Originally, this flow contains data points from $y^+ = 38.1$ onward. We add 10 artificial data points to $\mathbf{z}$ between this value and $y^+ = 1$. Next we perform 10 separate calibrations where each time we shift the lower bound of $\mathbf{z}$ one point closer to the original lower bound of $y^+ = 38.1$. We visualize the results in Figure 2.4 by plotting the HPD intervals for $C_{\varepsilon 2}$ and κ versus y_1^+, i.e. the y^+ value corresponding to the first data point. From this figure it becomes clear that the inclusion of near-wall data does not significantly alter the posteriors in this case. The other parameters in both $\boldsymbol{\theta}$ and $\boldsymbol{\gamma}$ show similar behaviour.

The sparsity of available experimental near-wall data, together with the indication that the posterior distributions do not change significantly justifies our exclusion of near-wall data.



(a) HPD intervals of $C_{\varepsilon 2}$

(b) HPD intervals of κ

Figure 2.4: The HPD intervals corresponding to calibrations with different y^+ cutoff values.

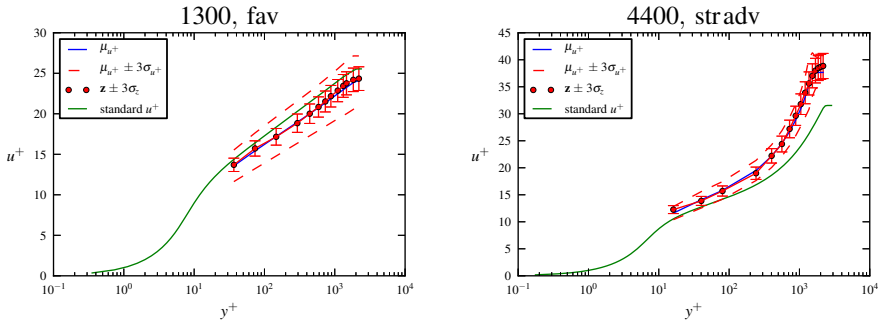
2.5.3. POSTERIOR MODEL CHECK

In Bayesian analysis it is good practice to assess the fit of the chosen model. We expect all observed data used in the calibration for each flow case to lie within the range of the posterior predictive distribution of the true process ζ for that case. It should be noted that this is not the same as validating the model, since it only ensures that the chosen model is capable of reproducing the observed data. It does not guarantee that it can also be used for predictions. In our model the variability in ζ can be broken down into that due to the explicit model inadequacy term $\eta(\mathbf{y}^+)$, and that due to uncertainty in $\boldsymbol{\theta}$. The former can be obtained directly from (2.16) and the calibrated values of $\boldsymbol{\gamma} = [\sigma, \alpha]$. The latter is just the posterior of $u^+(\boldsymbol{\theta})$, and can be computed using the u^+ traces stored during the McMC calibration run. This is equivalent to propagating posterior samples of $\boldsymbol{\theta}$ through the $k-\varepsilon$ model as in Monte-Carlo.

In Figure 2.5 we show only the uncertainty due to $\boldsymbol{\theta}$ for two flows. The posterior prediction for u^+ encompasses all the experimental data, and this is true for all the flows described in Table 2.1, even those in which a large η was predicted. They are therefore all consistent (in the sense of Ref. 19: existence of an overlap between the predictions and the region of experimental uncertainty). In addition, the calibrated models approximate the data better than the uncalibrated models in all cases. Based on this we judge the calibrations successful.

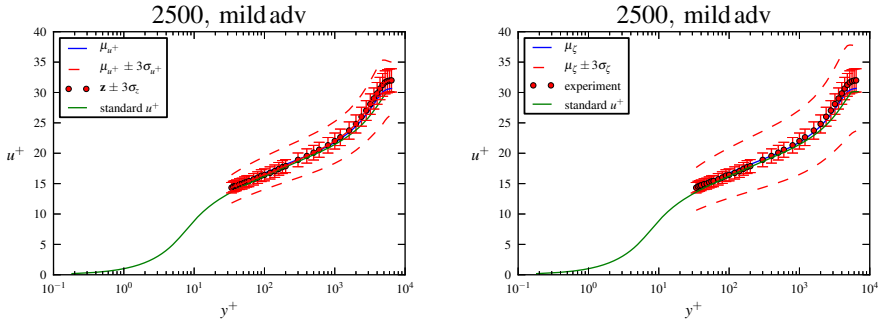
To illustrate the effect of also using η, we compare posterior distributions of u^+ and ζ in Figure 2.6. The mean of both distributions is the same, which could be inferred from (2.18). Thus, including a model inadequacy term of the form (2.16) results in a posterior distribution of the true process with the same mean as the posterior u^+ distribution, but a larger variance. When making predictions with the model this contribution to the variance should be included.

It is instructive to examine the effect of the coefficient uncertainties on quantities that are not directly measured. To that end we show the standard-deviation cloud for the normalized turbulent kinetic energy k in Figure 2.7. Since the condition $k > 0$ is enforced by limiters for all samples, the posterior probability of negative k is guaranteed to be



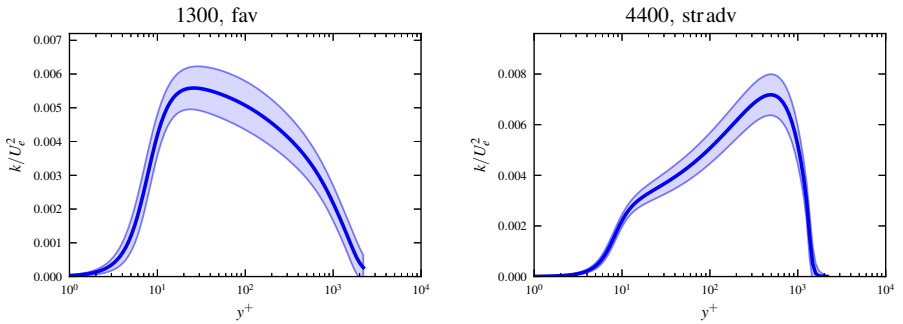
(a) Favourable flow. (b) Strongly-adverse flow.

Figure 2.5: The mean and 3 standard deviations of posterior $u^+(\mathbf{y}^+, \boldsymbol{\theta})$ samples of a favourable, and a strongly adverse flow (bottom). The green line indicates the solution of the $k-\varepsilon$ model using the standard values (2.4), and the red dots represent the experimental data with error bars.



(a) u^+ posterior. (b) $\boldsymbol{\zeta}$ posterior.

Figure 2.6: The posterior distribution of u^+, and the posterior distribution of $\boldsymbol{\zeta}$.



(a) flow case 1300. (b) flow case 1200.

Figure 2.7: A one standard-deviation cloud (i.e. $\mu_k \pm \sigma_k$) of k/U_e^2 for flow cases 1300 and 1200.

zero. However, since we do not have experimental data for the normalized turbulent kinetic energy, we cannot verify if its posterior distribution is consistent with the true process of k.

2.5.4. SOBOL INDICES

Figure 2.2 shows that, for a given flow case k, there is significant variation in the amount of information contained in the posterior closure-coefficient distributions. In an attempt to elucidate this behaviour, we perform global sensitivity analysis on u^+ with respect to $\boldsymbol{\theta}$. If the measurement data is very sensitive to a particular parameter or set of parameters in $\boldsymbol{\theta}$, we expect the corresponding posterior distribution to be well informed and *vice versa*. In particular we use Sobol' indices S_w, defined as [31]

$$
\begin{aligned}
D &= \mathrm{Var}_{\mathscr{W}}\left\{u^+\right\},\\
D_w &= \mathrm{Var}_w\left\{\mathbb{E}_{w'}(u^+|\boldsymbol{\theta}_w)\right\},\\
S_w &= D_w/D
\end{aligned}
$$

where $w \subset \{1,\ldots,Q\} = \mathscr{W}$ indexes the components of $\boldsymbol{\theta}$, $w' = \mathscr{W} \setminus w$, and $\mathrm{Var}_w\{\cdot\}$ indicates variance taken over priors of $\boldsymbol{\theta}_w$, etc. The indices S_w satisfy $\sum_{\mathscr{P}(\mathscr{W})} S_w = 1$, where $\mathscr{P}(\mathscr{W})$ is the power set of $\mathscr{W}$. A value close to unity for S_i can be interpreted as the coefficient corresponding to $i \in \mathscr{W}$ being responsible for most of the total variance in u^+ on its own (also without interaction effects with other parameters). A value close to zero indicates an uninfluential parameter. The D_w are computed using a polynomial approximation to the response in the stochastic space [17], given which the expectation and variance can be evaluated analytically [34, 35]. The D_w are computed using a 3rd-order polynomial approximation to the response in the stochastic space [17], given which the expectation and variance can be evaluated analytically [34, 35]. The method of [17] requires running $(p+1)^d$ deterministic simulations, where p is the polynomial order of the approximation and d is the number of uncertain parameters. We contrasted the results with those obtained from a 2nd-order polynomial approximation, and found no significant devi-

ations. This makes us confident about the convergence of the stochastic method with respect to the polynomial order.

Using the described setup, we calculate the main effect indices S_i with $i = 1,2,3,4$ corresponding to $\{C_{\varepsilon 2}, C_\mu, \sigma_k, \kappa\}$ for $u^+(y^+)$. The results are shown in Figure 2.8. The ranking from most sensitive parameter to least sensitive one for the velocity profile is $C_{\varepsilon 2}, \kappa, \sigma_k, C_\mu$. This is the same ranking that we get when we sort the coefficients from most informed posterior distribution to least informed one, see Figure 2.2. Thus, the very low sensitivity of our data to the value of C_μ is an explanation for the lack of information in the posterior C_μ distributions.

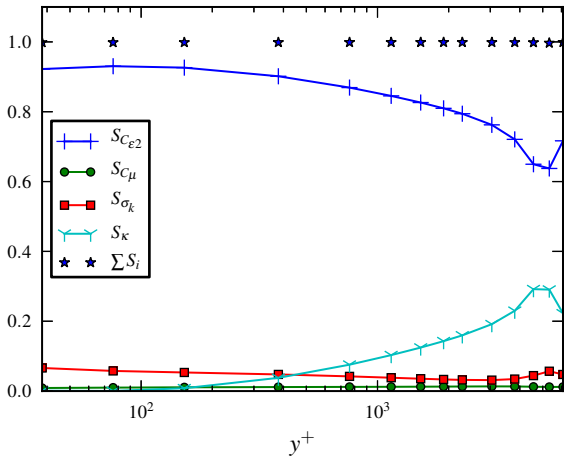


Figure 2.8: The Sobol indices S_i for flow 1400, with the velocity profile as QoI. The horizontal axis represents the direction normal to the wall for the stream-wise location $s = 16.3[ft]$.

This sensitivity analysis can be used in experimental design. E.g. we could ask: would adding friction coefficients C_f to the dataset lead to better informed distributions? Using the Sobol indices for C_f, we attempt to answer the question in the absence of C_f data. The S_i corresponding to the friction coefficient can be found in Figure 2.9. The influence of C_μ is still very low, therefore we would not expect to significantly improve its identification. This is consistent with the results from [3], where the SA turbulence model was calibrated using both velocity profiles and friction coefficients. Still, some of their posterior distributions were uninformative as well.

In Figures (2.8)-(2.9) we also show $\sum S_i$, i.e. the sum of all displayed Sobol indices. This sum is very close, but not equal to 1. This indicates that for the considered range in the closure coefficients, the interaction effects are low, i.e. the Sobol indices corresponding a combination of closure coefficients are small.

2.5.5. COEFFICIENT VARIABILITY ACROSS TEST-CASES

The HPD summaries of the $\boldsymbol{\theta}$-posteriors are shown in Figure 2.10 for all test-cases. In these plots the ordering of the test-cases on the horizontal axis corresponds roughly to increasingly unfavorable pressure gradients. The spread of the posterior modes of $C_{\varepsilon 2}$ is

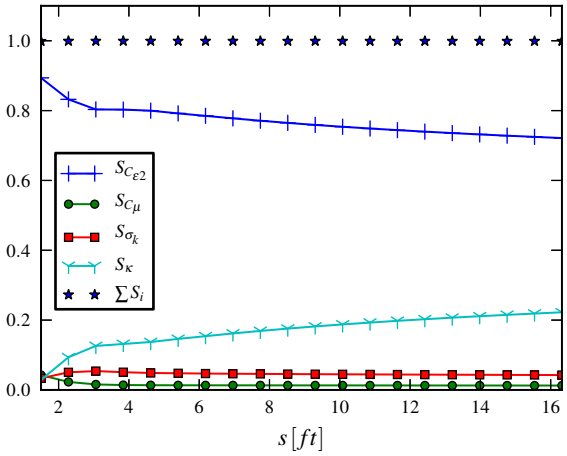


Figure 2.9: The Sobol indices S_i for flow 1400, with C_f as QoI. The horizontal axis represents the stream-wise direction.

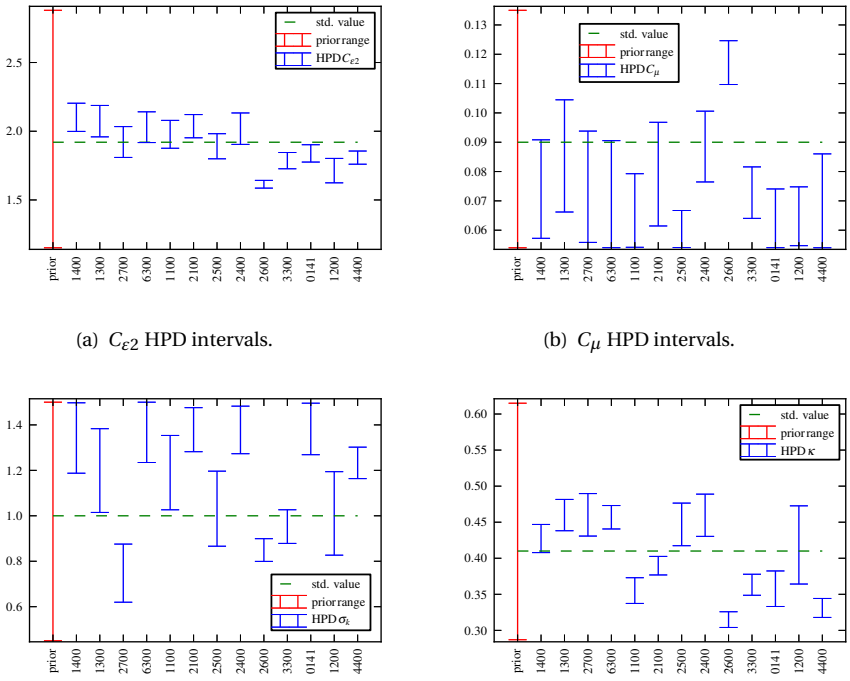
quite concentrated, they all lie relatively close to the standard value of 1.92. The small width of the HPD intervals (compared to the prior range), indicates that the posterior distributions have been well-informed by the data. Also notice that a slight downward trend of the HPD intervals can be observed with increasing pressure gradient, suggesting some physical effect not present in the model.

The spread of C_μ is relatively small, with most distributions centered close to 0.06, consistently lower than the standard value (0.09). The only clear exception is flow 2600, which also showed some outlier behaviour for $C_{\varepsilon 2}$. The parameter σ_k shows a larger spread, although in general values above the standard value of 1.0 are preferred.

Most individual pdfs of κ are quite well informed, but the modes are spread roughly between 0.31 and 0.46. Previous studies have looked at the spread of the von Karman constant. An overview is given in Ref. 41, which reports values of κ between $[0.33, 0.45]$, roughly similar to the spread that we have observed. The spread of the κ HPD intervals in Figure 2.10 can be qualitatively explained by considering the deviation of the experimental velocity profiles of Figure 2.1 from the standard log law $1/\kappa \ln\left(y^+\right) + C$. As can be seen from Figure 2.1, from roughly $y^+ = 30$ the velocity profiles overlap onto the standard log law. However, around $y^+ = 200$ the first profiles start to deviate from this law. Qualitatively, the profiles which show a larger deviation from the log law, are also the ones which show a lower κ HPD interval compared to the rest.

The HPD intervals of the hyper-parameters σ and $\log\alpha$ can be found in Figure 2.11. Most posterior modes of σ are located near the bottom edge of the domain, indicating that η is small, and the experimental data can be matched well with appropriate choice of $\boldsymbol{\theta}$ alone. The posterior modes for $\log(\alpha)$ all lie between 2.5 and 3.5, indicating that the remaining model error is correlated over a large fraction of the boundary layer [3]. In other words, a smooth model inadequacy term is preferred.

Figure 2.11 shows three clear outliers to this trend, flows 2400, 2500 and 1200. High



(a) $C_{\varepsilon 2}$ HPD intervals.

(b) C_μ HPD intervals.

(c) σ_k HPD intervals.

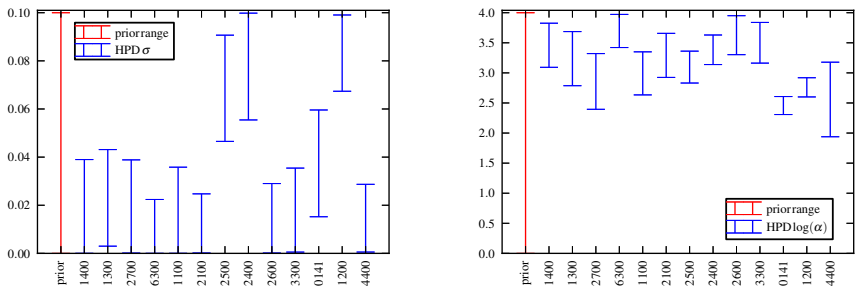
(d) κ HPD intervals.

Figure 2.10: The 50 % HPD intervals of $\boldsymbol{\theta}$ for the 13 cases of Table 2.1.

σ values in these cases indicate that the data can not be matched with $\boldsymbol{\theta}$ alone.

2.5.6. STATISTICAL MODEL SENSITIVITY

As mentioned in Section 2.4.5, the choice of $\eta(\cdot)$ represents a modelling assumption. We acknowledge that our choice of $\eta(\cdot)$ is not the best possible one in terms of physical modelling, e.g. the $\eta(\cdot)$ of [23] includes more physics. However, our goal is to capture model inadequacy mainly through $\boldsymbol{\theta}$, while relying on η to capture the *remaining* error. Hence, it is important to examine the sensitivity of the marginal $\boldsymbol{\theta}$ posteriors to the form of η. Therefore, we re-computed all calibrations with $\eta = 1$. In Figure 2.12 we show the HPD intervals for both $\eta = 1$ and the original $\eta(\cdot)$ of (2.16). Notice that the HPD intervals of $\eta = 1$ are smaller, which was to be expected since no uncertainty due to model inadequacy is added. More importantly though, the spread of closure coefficients across flow cases is roughly the same for both statistical models. This indicates that it is indeed possible to represent the bulk of the model inadequacy in $\boldsymbol{\theta}$ space, since choosing a radically different $\eta(\cdot)$ does not seem lead to a significantly different spread of $\boldsymbol{\theta}$.



(a) σ HPD intervals.

(b) $\log(\alpha)$ HPD intervals.

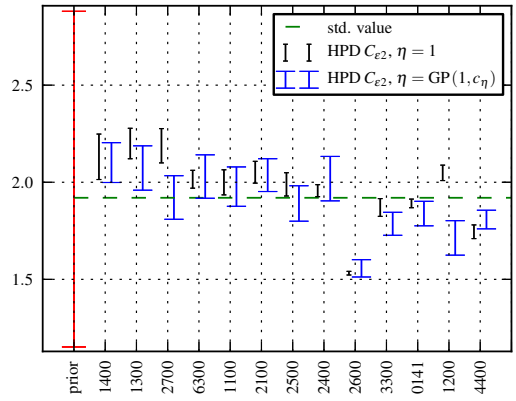
Figure 2.11: The 50 % HPD intervals of $\boldsymbol{\gamma}$ for the 13 cases of Table 2.1.

2.5.7. PREDICTION WITH UNCERTAINTIES

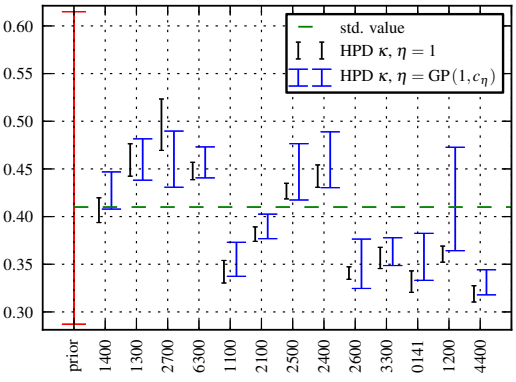
The calibrations described above contain quantitative information on the accuracy of the $k-\varepsilon$ model for flat-plate flows. We apply the methodology described in Section 2.4.4 to estimate u^+ and the associated uncertainty for a new flat-plate flow not in the calibration set. We use the data (used only as ground-truth) from Ref. 21, which is boundary-layer data on a cylinder in axially symmetric flow. This is flow 3600 from the 1968 AFOSR-IFP-Stanford conference [4]. The results for three y^+ stations ($y^+_* = 46.2$, 267.0 and 1039.7) are given in Figure 2.13, together with boxes representing the experimental data $\pm 3\sigma$.

The width of the p-boxes clearly dominates their slant, suggesting that case-to-case variability dominates uncertainty in individual posteriors. Therefore adding more experimental data to the 13 calibrations will not much reduce our overall uncertainty. Had we not included the strongly-adverse pressure-gradient cases, the width of the p-box would be significantly smaller. However this would simply correspond to a more restricted range of flows for which the analysis is valid. To tighten these error estimates, the class of considered flows must be restricted to flows in some sense similar to the new flow. It is not presently clear how to do this automatically. We note that our estimates of simulation error are larger than estimates of experimental error, as expected.

Finally we extract confidence intervals from the p-boxes of Figure 2.13. In Figure 2.14 we show 90% confidence intervals on u^+ values for case 3600, and compare them with the measurement data as a reference. Note that all intervals extracted from the p-boxes are consistent with the experimental data – indicating that modelling inadequacy has been successfully bounded for this case. It is clearly a tight bound on the low-u^+ side. That the upper bounds are not tight in some locations, is an indication that uncertainty in the closure coefficients remains substantial. To put this in perspective, consider that, for reasonable values of the coefficients, the $k-\varepsilon$ model can produce solutions anywhere within our confidence intervals.

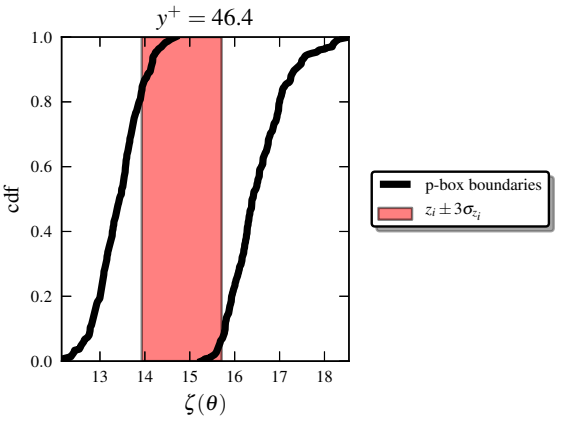


(a) $C_{\varepsilon 2}$ HPD intervals.

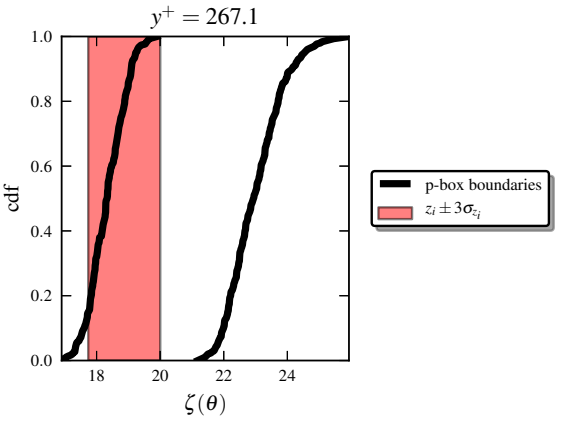


(b) κ HPD intervals.

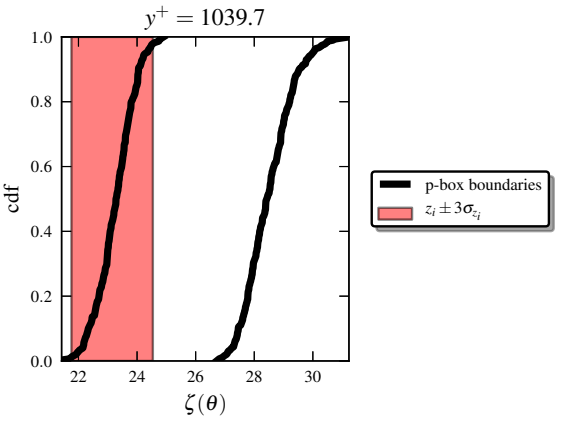
Figure 2.12: The HPD intervals for $C_{\varepsilon 2}$ and κ for two statistical models.



(a) The u^+ p-box for flow 3600 at $y_*^+ = 46.2$.



(b) The u^+ p-box for flow 3600 at $y_*^+ = 267.0$.



(c) The u^+ p-box for flow 3600 at $y_*^+ = 1039.7$.

Figure 2.13

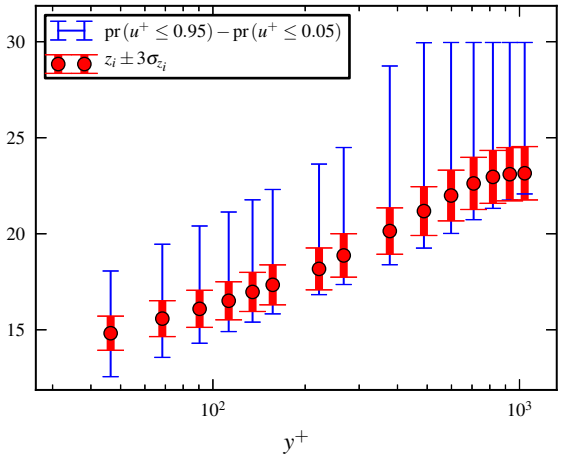


Figure 2.14: The 90 % confidence intervals extracted from p-boxes.

2.6. CONCLUSION

Using a Bayesian framework, we performed 13 separate calibrations of the closure coefficients in the standard $k-\varepsilon$ model. The experimental data on which we calibrated consisted of velocity profiles from 13 boundary-layer flows, each subject to a different pressure gradient. This allowed us to investigate the resulting spread of the posterior coefficient distributions, caused by the range of considered pressure gradients. To summarize the spread we perform a Highest Posterior Density (HPD) analysis on all posterior distributions, which gives us 13 credible intervals of most-likely values for both the closure coefficients and the hyper-parameters used to parameterize the model-inadequacy term.

The results show a significant variation of coefficient-posteriors across this (very limited) range of flows, for $C_{\varepsilon 2}$ and especially for κ. Other coefficients were not identified well enough by the data to allow for clear conclusions about their variability. In any case it is clear that a single calibration does not provide us with reliable coefficients, or an estimate of model error. On the other hand, we use all 13 posteriors to build an error estimate containing both between-case coefficient variability, and individual posterior uncertainty. This estimate is a probability-box (p-box) for model output, from which confidence intervals can be constructed. We demonstrate the methodology for a 14th flat-plate flow not from the calibration set. The resulting confidence intervals include the reference measurements, indicating that the estimate is not too narrow.

The next chapter will involve attempting to narrow the confidence intervals further, by selection of a subset of cases from the calibration set that represent the new flow best. Furthermore we will apply the analysis to other turbulence models, to determine if the conclusions hold for models that are generally considered more accurate than $k-\varepsilon$. The use of Bayesian model averaging to combine all the resulting information into a predictive framework is promising.

Acknowledgements The present work was supported by the French 'Agence Nationale de la Recherche' (ANR) under contract ANR-11-MONU-008-002.

REFERENCES

[1] FG Blottner. Variable grid scheme applied to turbulent boundary layers. *Computer Methods in Applied Mechanics and Engineering*, 4(2):179–194, 1974.

[2] M.H. Chen and Q.M. Shao. Monte carlo estimation of bayesian credible and hpd intervals. *Journal of Computational and Graphical Statistics*, 8(1):69–92, 1999.

[3] S.H. Cheung, T.A. Oliver, E.E. Prudencio, S. Prudhomme, and R.D. Moser. Bayesian uncertainty analysis with applications to turbulence modeling. *Reliability engineering and systems safety*, 96(9):1137–1149, 2011.

[4] DE Coles and EA Hirst. Computation of turbulent boundary layers. In *Proceedings of AFOSR-IFP Stanford Conference*, volume 2, 1968.

[5] B. J. Debusschere, H.N. Najm, P. P. Pébay, O. M. Knio, R. G. Ghanem, and O. P. Le Maître. Numerical challenges in the use of polynomial chaos representations for stochastic processes. *SIAM, J. Sci. Comput.*, 26(2):698–719, 2001.

[6] Richard Dwight and Zhong-Hua Han. Efficient uncertainty quantification using gradient-enhanced Kriging. In *Proceedings of 11th AIAA Conference on Non-Deterministic Approaches, Palm Springs CA. AIAA-2009-2276*, pages 1–21. AIAA, 2009.

[7] J.H. Ferziger and M. PERIĆ. Further discussion of numerical errors in cfd. *International Journal for Numerical Methods in Fluids*, 23(12):1263–1274, 1996.

[8] A. Gelman, JB Carlin, HS Stern, and DB Rubin. Bayesian data analysis. 2004. *Chapman&Hall, London*, 2, 2004.

[9] W.K. Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[10] T Gunnar Johansson. Near-wall measurements and wall shear stress. In *Progress in Wall Turbulence: Understanding and Modeling*, pages 377–384. Springer, 2011.

[11] WP Jones and B.E. Launder. The prediction of laminarization with a two-equation model of turbulence. *International Journal of Heat and Mass Transfer*, 15(2):301–314, 1972.

[12] M.C. Kennedy and A. O'Hagan. Bayesian calibration of computer models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(3):425–464, 2001.

[13] J. Kim, P. Moin, and R. Moser. Turbulence statistics in fully developed channel flow at low reynolds number. *Journal of Fluid Mechanics*, 177(1):133–166, 1987.

[14] SJ Kline. The 1980-81 afosr-httm-stanford conference on complex turbulent flows: Comparison of computation and experiment. In *Objectives, evaluation of data, specifications of test cases, discussion and position papers*, volume 1, 1981.

[15] CF Lange, F Durst, and M Breuer. Correction of hot-wire measurements in the near-wall region. *Experiments in fluids*, 26(5):475–477, 1999.

[16] BE Launder and BI Sharma. Application of the energy-dissipation model of turbulence to the calculation of flow near a spinning disc. *Letters Heat Mass Transfer*, 1:131–137, 1974.

[17] GJA Loeven, JAS Witteveen, and H. Bijl. Probabilistic collocation: an efficient non-intrusive approach for arbitrarily distributed parametric uncertainties. In *Proceedings of the 45th AIAA Aerospace Sciences Meeting*, volume 6, pages 3845–3858, 2007.

[18] L. Lourenco and A. Krothapalli. On the accuracy of velocity and vorticity measurements with piv. *Experiments in Fluids*, 18(6):421–428, 1995.

[19] P.E. McSharry and L.A. Smith. Consistent nonlinear dynamics: identifying model inadequacy. *Physica D: nonlinear phenomena*, 192(1):1–22, 2004.

[20] M.S. Mohamed and J.C. Larue. The decay power law in grid-generated turbulence. *Journal of Fluid Mechanics*, 219(1):195–214, 1990.

[21] H.L. Moses. The behavior of turbulent boundary layers in adverse pressure gradients. Technical report, DTIC Document, 1964.

[22] W.L. Oberkampf and C.J. Roy. *Verification and validation in scientific computing*. Cambridge University Press, 2010.

[23] T.A. Oliver and R.D. Moser. Bayesian uncertainty quantification applied to rans turbulence models. In *Journal of Physics: Conference Series*, volume 318, page 042032. IOP Publishing, 2011.

[24] Sergio Pirozzoli and Matteo Bernardini. Probing high-reynolds-number effects in numerical boundary layers. *Physics of Fluids*, 25:021704, 2013.

[25] PDA Platteeuw, GJA Loeven, and H. Bijl. Uncertainty quantification applied to the k-e model of turbulence using the probabilistic collocation method. In *AHS Adaptive Structures Conference; 10 th AIAA Non-Deterministic Approaches Conference; 9 th AIAA Gossamer Spacecraft Forum and the 4 th AIAA Multidisciplinary Design Optimization Specialist Conference*. American Institute of Aeronautics and Astronautics, 1801 Alexander Bell Drive, Suite 500, Reston, VA, 20191-4344, USA,, 2008.

[26] S.B. Pope. *Turbulent flows*. Cambridge Univ Pr, 2000.

[27] W. Rodi and G. Scheuerer. Scrutinizing the k-ε turbulence model under adverse pressure gradient conditions. *Journal of fluids engineering*, 108:174, 1986.

[28] P. Sagaut. *Large eddy simulation for incompressible flows: an introduction*. Springer, 2005.

[29] G.B. Schubauer and PS Klebanoff. Investigation of separation of the turbulent boundary layer. Technical report, DTIC Document, 1950.

[30] T. Sjögren and A.V. Johansson. Development and calibration of algebraic nonlinear models for terms in the reynolds stress transport equations. *Physics of Fluids*, 12:1554, 2000.

[31] I.M. Sobol. Global sensitivity indices for nonlinear mathematical models and their monte carlo estimates. *Mathematics and computers in simulation*, 55(1-3):271–280, 2001.

[32] P.R. Spalart and S.R. Allmaras. A one-equation turbulence model for aerodynamic flows. *La recherche aérospatiale*, 1(1):5–21, 1994.

[33] C.G. Speciale, R. Abid, and Anderson E.C. A critical evaluation of two equation model for near-wall turbulence. *AIAA Journal*, 90, 1990.

[34] B. Sudret. Global sensitivity analysis using polynomial chaos expansions. *Reliability Engineering & System Safety*, 93(7):964–979, 2008.

[35] G. TANG, MS ELDRED, and L.P. SWILER. Global sensitivity analysis for stochastic collocation expansion. *CSRI SUMMER PROCEEDINGS 2009*, page 100, 2010.

[36] S. Tavoularis and U. Karnik. Further experiments on the evolution of turbulent stresses and scales in uniformly sheared turbulence. *Journal of Fluid Mechanics*, 204(1):457–478, 1989.

[37] D.C. Wilcox. Comparison of two-equation turbulence models for boundary layers with pressure gradient. *AIAA Journal (American Institute of Aeronautics and Astronautics);(United States)*, 31(8), 1993.

[38] D.C. Wilcox. *Companion Software:Turbulence Modeling for CFD Third Edition*, 2006.

[39] D.C. Wilcox, American Institute of Aeronautics, and Astronautics. *Turbulence modeling for CFD*, volume 3. DCW industries La Canada, CA, 2006.

[40] N. Zabaras and B. Ganapathysubramanian. A scalable framework for the solution of stochastic inverse problems using a sparse grid collocation approach. *Journal of Computational Physics*, 227(9):4697–4735, 2008.

[41] E.S. Zanoun, F. Durst, and H. Nagib. Evaluating the law of the wall in two-dimensional fully developed turbulent channel flows. *Physics of Fluids*, 15:3079, 2003.

[42] Z. Zhang, W. Zhang, Z.J. Zhai, and Q.Y. Chen. Evaluation of various turbulence models in predicting airflow and turbulence in enclosed environments by cfd: Part 2—comparison with experimental data from literature. *Hvac&R Research*, 13(6):871–886, 2007.

3

PREDICTIVE RANS SIMULATIONS VIA BAYESIAN MODEL-SCENARIO AVERAGING

3.1. INTRODUCTION

As described in the prededing chapter, in the Bayesian framework model error is typically accounted for with a stochastic term added to (or multiplying) the output of the simulation. For instance Kennedy and O'Hagan [12] define "model inadequacy" as the discrepancy that remains between the true process and the prediction of the simulation code evaluated using the best-fit parameter values. We believe however that modelling this discrepancy directly is inappropriate when the simulation is an approximation of continuum mechanics, which is characterized by perfectly known conservation laws (e.g. conservation of momentum), containing imperfect, subordinate, empirical models, such as turbulence closure models in RANS. In this context, it makes much more sense to include stochastic modelling of inadequacy at the level of the empirical model, not at the level of the simulation code output. This choice has the ancillary benefit that predictions of quantities of other type than the calibration data can be made – and in other geometries – while still incorporating estimates of model inadequacy. This is not possible in the original Kennedy and O'Hagan framework.

There are two natural ways to incorporate stochastic model inadequacy into turbulence closure models: via the Reynolds-stress tensor [8], and via the closure coefficients [7]. The former authors proposed a physically-motivated method for perturbing the Reynolds stress tensor, but the uncertainty is determined by physically possible states, and not informed by experimental data. In [7] the present authors used Bayesian statistics to estimate the model error of the $k-\varepsilon$ turbulence model [14], via multiple posterior distributions of the closure coefficients (see also Chapter 2). However our use of

This chapter is based on: W.N. Edeling, P. Cinnella, R.P. Dwight, Predictive RANS simulations via Bayesian Model-Scenario Averaging, Journal of Computational Physics, 275 (2014) 65–91.

p-boxes limited the flexibility of the predictions, leading to model error estimates that contained the true solution, but which were impractically large. Hence, our goal is to quantify model error in the predictions of RANS codes via the closure coefficients, while obtaining more realistic error bars.

Our novel approach is again based on Bayesian statistics [9], and can be summarized as follows: we first choose a *class* of flows for which we wish to make reliable predictions with quantified model error, in this work the class is flat-plate boundary-layers at a variety of pressure gradients. We select a number of examples of this class for which we have reliable experimental data; these flows form our *calibration scenarios*. Finally we select a set of turbulence closure models, that are both expected to perform acceptably on the flow-class, but which are also heterogeneous in their modelling approaches, in this work we use $k-\varepsilon$, $k-\omega$, Spalart-Allmaras, Baldwin-Lomax and stress-ω models [24]. For each model and each scenario we perform Bayesian calibration to obtain posteriors on closure coefficients [2] as well as on model probabilities, i.e. probabilities of recovering the data by using a given model with the calibrated coefficients. For all models a strong dependence of coefficients on scenario is observed. This variation is interpreted as the extent to which the coefficients must be adjusted to match scenarios in the calibration set; informally a kind of coefficient uncertainty. Precisely, to make predictions for some Quantity of Interest (QoI) in some unmeasured scenario, we use Bayesian Model Averaging (BMA) [11] to collate the individual coefficient posteriors. A key step is propagating these posteriors through the simulation code for the prediction scenario. By using the freedom inherent in BMA to define prior scenario probabilities, we automatically assign a higher weight to those calibration scenarios which allow recovering consistent predictions for a new scenario with all alternative models considered; these are often found to be similar to the prediction scenario. The prediction is summarized by the mean and variance of the posterior predictive distribution [5, 21]. Because in our framework the variation between scenarios is at least as important as the variation between models, we speak of *Bayesian Model-Scenario Averaging* (BMSA), rather than BMA.

This chapter is laid out as follows: the five turbulence models are described in Section 3.2, and the scenarios together with experimental data in Section 3.3. Section 2.4 lays out our calibration framework, Section 3.4.1 prediction with Bayesian Model-Scenario Averaging, and Section 3.4.2 automatic scenario weighting. Section 3.5 discusses results.

3.2. TURBULENCE MODELS

Our model set consists of five different turbulent models of varying degrees of complexity. It includes the Launder-Sharma $k-\varepsilon$ model, which was described in Section 2.2. Another included two-equation model is the Wilcox (2006) $k-\omega$ model, and we also consider the one-equation Spalart-Allmaras model. The simplest turbulence model in our set is the algebraic Baldwin-Lomax model, and the most complex one is the stress-ω model. All except the $k-\varepsilon$ model are briefly outlined below. For a more detailed description of their mathematical structure we refer to [24].

3.2.1. THE WILCOX (2006) $k-\omega$ MODEL

This model is also a two-equation model, which computes the eddy viscosity as $\nu_T = k/\tilde{\omega}$, where $\tilde{\omega}$ is a quantity related to the specific dissipation $\omega = \varepsilon/k$ [24]. The following closure coefficients are present: α, β_o, β^*, σ, σ^* and σ_{do}. Again, a constraint between parameters, equivalent to (2.12), can be found [24]

$$\alpha = \frac{\beta}{\beta^*} - \frac{\kappa^2}{2\sqrt{\beta^*}}, \tag{3.1}$$

which we use to fix α. The nominal values of the remaining coefficients can be found in Table 3.1.

3.2.2. THE SPALART-ALLMARAS MODEL

This is a one-equation model where a transport equation is solved for a viscosity-like variable $\tilde{\nu}$ [23]. It contains the following seven closure coefficients: C_{b1}, C_{b2}, σ, C_{w2}, C_{w3}, C_{v1} and κ. The coefficient C_{w1} is constraint by the values of the other coefficients as

$$C_{w1} = \frac{C_{b1}}{\kappa^2} + \frac{1 + C_{b2}}{\sigma}, \tag{3.2}$$

the other 7 parameters are considered for calibration.

3.2.3. THE BALDWIN-LOMAX MODEL

Unlike the preceding models, which require the solution of one or two additional transport equations, this model assumes an algebraic expression for ν_T [1]. It is therefore the simplest model in our set. Like the other models it also contains closure coefficients, namely A_0^+, C_{cp}, C_{kleb}, C_{wk}, α and κ, which are all considered for calibration. Again, their nominal values can be found in Table 3.1.

3.2.4. THE STRESS-ω MODEL

All preceding models utilize the Boussinesq hypothesis to close the RANS equations, which assumes that the principal axes of the Reynolds-stress tensor τ_{ij} are coincident with those of the mean strain-rate tensor S_{ij} [24]. The constant of proportionality between τ_{ij} and S_{ij} is the eddy viscosity ν_T, which is calculated by means of a turbulence model, and appears in (2.1a). However, the validity of the Boussinesq hypothesis is questionable, see e.g. [22] for a discussion. A class of (more complex) turbulence models attempting to remedy these shortcomings are stress-transport models. These directly write a model transport equation for the Reynolds-stress tensor τ_{ij}, rather than relying on the linear relationship between τ_{ij} and S_{ij} suggested by the Boussinesq hypothesis. An ancillary transport equation for a turbulent length scale is also required, (see [24] for further details). One such a model is the stress-ω model of Wilcox [24]. As it uses ω from the $k-\omega$ model for certain terms in the τ_{ij} transport equation, it contains all the same closure coefficients as the Wilcox (2006) $k-\omega$ model, plus two additional coefficients, denoted as C_1 and C_2.

Table 3.1: Traditional closure coefficient values [24].

$k-\varepsilon$		$k-\omega$		SA		BL		stress-ω	
κ	0.41	κ	0.41	κ	0.41	κ	0.4	κ	0.41
C_μ	0.09	β^*	0.09	c_{v1}	7.1	A_0^+	26.0	β^*	0.09
$C_{\varepsilon 2}$	1.92	β_o	0.0708	c_{b1}	0.1355	α	0.0168	β_o	0.0708
σ_k	1.0	σ^*	0.6	c_{b2}	0.622	C_{cp}	1.6	σ^*	0.6
		σ	0.5	σ	2/3	C_{kleb}	0.3	σ	0.5
		σ_{do}	0.125	c_{w2}	0.3	C_{wk}	1.0	σ_{do}	0.125
				c_{w3}	2.0			C_1	9/5
								C_2	10/19

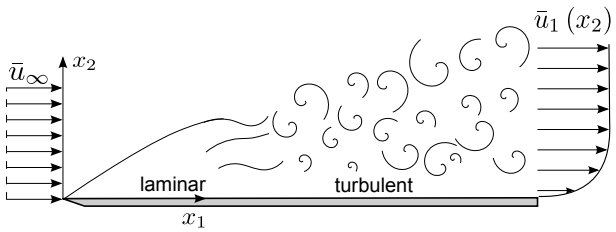


Figure 3.1: A schematic overview of a flat-plate turbulent boundary layer. Shown are the uniform inflow velocity $\bar{u}_\infty$, a schematic visualisation of the instantaneous flow field, and the averaged velocity $\bar{u}_1\,(x_2)$

.

3.3. TURBULENT BOUNDARY-LAYER CONFIGURATION

In the following, we investigate the predictive capabilities of our Bayesian methodology for a specific class of flows, again incompressible turbulent boundary layers subject to different pressure gradients. Figure 3.1 depicts a sketch of a turbulent boundary layer.

3.3.1. EXPERIMENTAL BOUNDARY-LAYER DATA

EDDYBL comes with configuration files which mimic the experiments described in the 1968 AFOSR-IFP-Stanford conference proceedings [4]. From this data source, we selected one zero pressure-gradient flow, and 13 flows from other types of available pressure gradients, which range from favorable ($d\bar{p}/dx_1 < 0$) to strongly adverse ($d\bar{p}/dx_1 > 0$) gradients. These 14 flows are described in table 3.2. The identification number of each flow is copied from [4]. We plotted the experimentally determined, non-dimensional, streamwise velocity profiles in Figure 3.2. As usual, the normalized streamwise velocity is defined as $u^+ \equiv \bar{u}_1/u_\tau$. As mentioned in Section 2.3, too much weight should not be given to the classifications of the severity of the adverse gradients in Table 3.2, since some flows (such as 2100 and 2400) experience multiple gradient types along the spanwise direction.

Therefore, it can be more informative to also inspect which flows are in equilibrium. There are two main forces acting on a boundary-layer. First, the pressure gradient acts

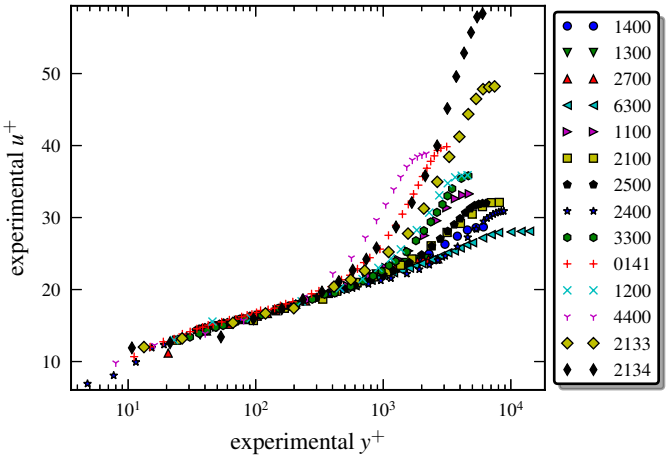


Figure 3.2: Our experimental data set.

with a force per unit length and width as $\delta^* \mathrm{d}\bar{p}/\mathrm{d}x_1$, where δ^* is the boundary-layer displacement thickness. The second force on the boundary layer comes from the wall-shear stress τ_w. A boundary-layer is said to be in equilibrium when the ratio of these two forces is constant along the length of the boundary layer, i.e. when the so-called equilibrium parameter $\beta_T := (\delta^*/\tau_w)(\mathrm{d}\bar{p}/\mathrm{d}x_1)$ is constant [3]. We plot the absolute value of the experimentally determined β_T values in figure 3.3. Flows such as 6300 with constant β_T are simple flows for which the turbulence models should be able to make accurate predictions. Other flows are more complex. We especially expect that models will encounter difficulties in the last part of flow 2100, i.e. flows 2133 and 2134.

We calibrate each model of Section 3.2 for each flow of Table 3.2 separately, using one velocity profile as experimental data. We again omit any experimental data in the viscous wall region. Remember we investigated the effect of not including near-wall data on the posterior distributions in Section 2.5.2, where we found that the posteriors were not significantly affected by this exclusion.

3.3.2. SENSITIVITY ANALYSIS OF BOUNDARY-LAYER PROBLEM

To get an idea of which coefficients can be informed by our data, we perform a variance-based, global-sensitivity analysis before we start the calibrations. We expect that we can obtain informative posterior distributions for the parameters with a high sensitivity measure, whereas the ones with low sensitivities are likely to yield posterior distributions that do not differ much from the uniform prior distributions. To this end we call $\boldsymbol{\theta} \in \mathbb{R}^Q$ a random vector, the components of which are Q uncertain closure coefficients. We carry out a sensitivity analysis of the output quantity $u^+ = u^+\left(\mathbf{y}^+;\boldsymbol{\theta}\right)$ by computing the Sobol indices S_w, as described in Section 2.5.4.

We considered uniform input pdfs for all coefficients, with the boundaries located at $\pm 10\%$ of the nominal values displayed in Table 3.1. The results can be found in Figure 3.4. In case of the $k-\varepsilon$ model with the constraints (2.12)-(2.13) applied, the most influential

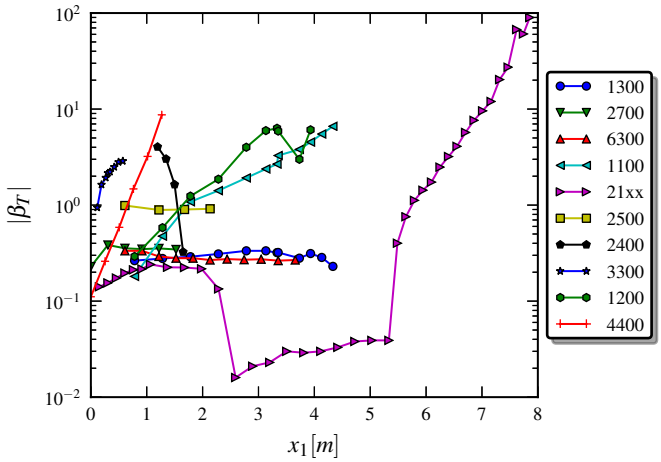
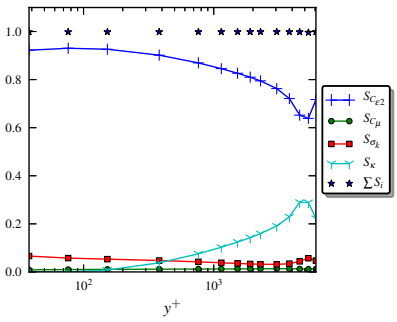


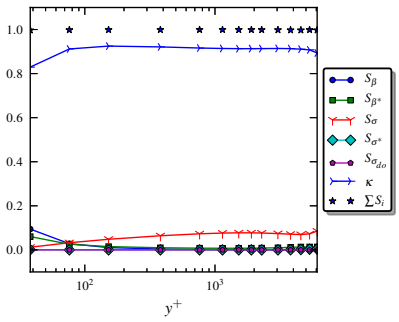
Figure 3.3: Experimental $|\beta_T|$ values for all flows but 1400 and 0141. Source [4].

Table 3.2: Flow descriptions, source [4].

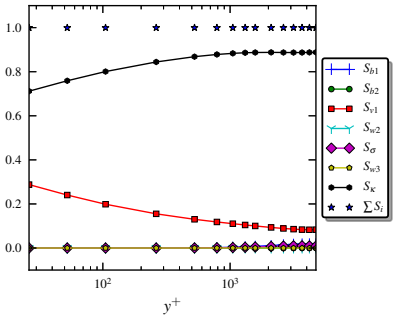
Identification	**Type**	**Description**
1400	Zero	Equilibrium boundary layer at constant pressure
1300	Fav	Near-equilibrium boundary layer in moderate negative pressure gradient
2700	Fav	Equilibrium boundary layer in mild negative pressure gradient
6300	Fav	Near-equilibrium boundary layer growing beneath potential flow on model spillway
1100	Mild adv	Boundary layer in diverging channel
2100, 2133, 2134	Div	Boundary layer on large airfoil-like body; pressure gradient first mildly negative, then strongly positive, with eventual separation
2500	Mild adv	Equilibrium boundary layer in mild positive pressure gradient
2400	Div	Initial equilibrium boundary layer in moderate positive pressure gradient; pressure gradient abruptly decreases to zero, and flow relaxes to new equilibrium
3300	Mod adv	Boundary layer, initially at constant pressure, developing into equilibrium flow in moderate positive pressure gradient
0141	Str adv	Boundary-layer with strong adverse pressure gradient, source [13]
1200	Str adv	Boundary layer in diverging channel with eventual separation
4400	Str adv	Boundary layer in strong positive pressure gradient



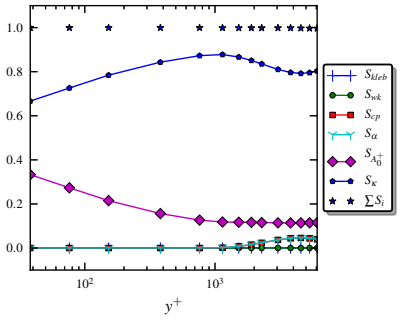
(a) The $k-\varepsilon$ model.



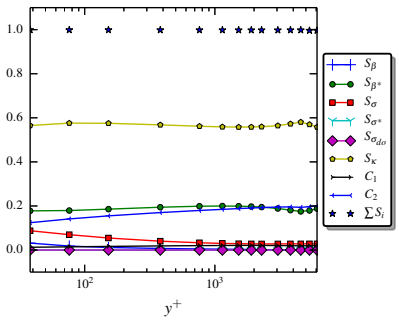
(b) The $k-\omega$ model.



(c) The Spalart-Allmaras model.



(d) The Baldwin-Lomax model.



(e) The stress-ω model.

Figure 3.4: Sobol indices of the considered turbulence models for flow case 1400.

parameter is $C_{\varepsilon 2}$, followed by κ. The parameter $C_{\varepsilon 2}$ is the proportionality constant of the dissipation term in the transport equation of ε. The von Karman constant κ enters the model through the constraint on σ_ε (2.12), which is the proportionality constant of the diffusion term in the ε equation.

The Sobol indices of the $k-\omega$ model were calculated with the constraint (3.1) applied. We can see that κ is by far the most influential parameter. Again, κ enters the model through a constraint. This time the constraint is on α, which regulates the production of ω. The second parameter that has an impact on the computed u^+ is σ, which is the proportionality constant of the diffusion term of the ω transport equation.

For the SA model we see that again κ is the most influential parameter when it comes to u^+ profiles. And, like the two preceding models, it also enters the model through a constraint. Here, the constraint (3.2) is on C_{w1}, which regulates the dissipation of $\tilde{\nu}$. A second influential parameter is C_{v1}, which appears in the function f_{v1}. In turn, this function relates $\tilde{\nu}$ to the eddy viscosity as $\nu_T = f_{v1}\tilde{\nu}$.

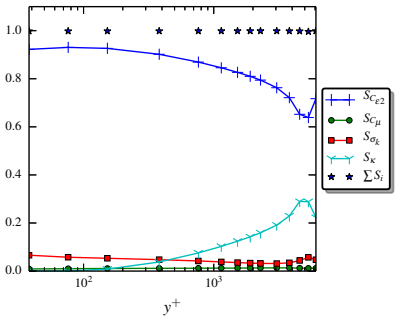
For the Baldwin-Lomax model the most influential parameter is κ as well, although it enters the model directly though algebraic expressions of the eddy viscosity ν_T. The u^+ profiles are sensitive to A_0^+ as well, which is a parameter in the van Driest damping function [24].

Finally, for the stress-ω model we find that again κ, which performs exactly the same function as for the $k-\omega$ model, is the most influential, although to a lesser extend compared to the other 4 models. Unlike the $k-\omega$ model however, the parameters β and β^* also carry some weight.

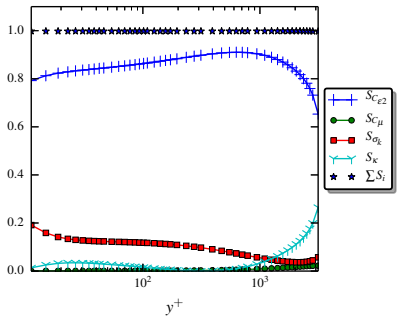
So far we have examined Sobol indices for one particular flow case, i.e. 1400. The variation over different flow cases is not very large, although small changes can occur, see Figure 3.5 for an example, which compares Sobol indices associated to the closure coefficients of $k-\varepsilon$ computed for flow case 1400, and for the strong adverse pressure gradient case 0141. In both cases $C_{\varepsilon 2}$ is by far the most influential. And in both cases κ becomes more influential than σ_k when we move away from the wall. However, in the strong adverse case the intersection point is at a higher y^+ value, and before that σ_k has a higher Sobol index than it had in case 1400.

3.4. STATISTICAL METHODOLOGY

Similar to our approach in Chapter 2, the first step in our method consists of calibrating the closure coefficients of each turbulence model of Section 3.2 against different sets of experimental data. The calibration framework is already described in Section 2.4, and the selected priors for all turbulence models are described in the next section. Thus we obtain multiple sets of posterior parameter distributions for each model by calibrating it to each flow of Table 3.2. We summarize this large amount of information by calculating the Highest Posterior Density intervals of Section 2.4.3. The final phase consists of using the posterior parameter distributions to make predictions for a new flow case. To achieve this we use the Bayesian Model-Scenario Averaging (BMSA) framework described in Section 3.4.1.



(a) Sobol indices for $k-\varepsilon$ model for case 1400.



(b) Sobol indices for $k-\varepsilon$ model for case 0141.

Figure 3.5: Variation of Sobol indices over flow cases for the $k-\varepsilon$ model.

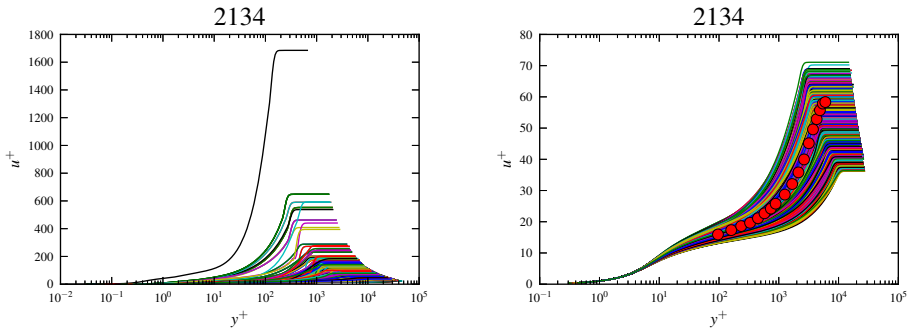
PRIORS FOR $\boldsymbol{\theta}$ AND $\boldsymbol{\gamma}$

All priors, for both the closure coefficients $\boldsymbol{\theta}$ and hyper-parameters $\boldsymbol{\gamma}$, are independent uniform distributions. The choice of interval end-points was mainly on the range of coefficients for which the solver was stable and provides physical solutions. By physical solutions we mean u^+ profiles that are not excessively large. Take for instance the stress-ω model. If we assume a uniform prior where we perturb the coefficients by ±50% from their nominal values, we found u^+ values that were extremely large. The worst case we encountered is depicted in Figure 3.6(a). In the calibration phase this does not pose a problem. The McMC chain will automatically reject the bad samples and move to a part of the stochastic domain that does generate samples close to the data. However, our predictive methodology (Section 3.4.1), involves the propagation of multiple posterior distributions through the code at validation settings. In this case the extremely unphysical samples can re-occur, although with low probability. This was observed especially in the case of the stress-ω model. Therefore we chose the prior domain such that these extreme outliers are not possible. At the same time we checked that the solution variation, obtained by prior propagation, was still large enough to easily encompass the data, see e.g. Figure 3.6(b).

For the $k-\varepsilon$ model, which does not suffer from extreme outliers, we re-used our posterior distributions from Chapter 2. For the stress-ω and $k-\omega$ models the prior domain boundaries of each coefficient were set at ±20% of their respective nominal values. For the Spalart-Allmaras and Baldwin-Lomax models we used ±30%.

3.4.1. BAYESIAN SCENARIO AVERAGING: PREDICTION

Calibration according to Section 2.4 gives values of $\boldsymbol{\theta}$ that reproduce the single scenario $\mathbf{S}$ accurately. The variance of $p(\boldsymbol{\theta}|z)$ only contains information about how well $\boldsymbol{\theta}$ has been identified by $\mathbf{z}$, but nothing about the variability needed in $\boldsymbol{\theta}$ when varying the scenario $\mathbf{S}$. It is this latter information that is relevant for predicting model inadequacy in some unmeasured scenario $\mathbf{S}'$. In the remaining derivation the effect of $\boldsymbol{\gamma}$ is integrated out, that is we work with marginal posteriors of $\boldsymbol{\theta}$ only.



(a) 5000 samples obtained by propagating ±50% priors through the stress-ω model configured for case 2134. Clearly some extremely non-physical samples are present considering that the maximum experimental u^+ value is 58.3.

(b) 5000 samples obtained by propagating ±20% priors through the stress-ω model configured for case 2134. The red dots are the experimental data.

Figure 3.6: Outlier analysis as a function of the prior domain.

To capture this we consider a set of K measured scenarios $\mathscr{S} = \{\mathbf{S}_1, \ldots, \mathbf{S}_K\}$ chosen to encompass the class of flows we wish to predict. In this work they are a subset of the 14 cases described in Section 2.3. The class of flows we may expect to reliably predict is therefore the class of attached flat-plate boundary-layers at any pressure-gradient within the range of cases in $\mathscr{S}$. However, the predictive capability of our $\boldsymbol{\theta}$ for a QoI outside this class is investigated in Section 3.5.4. The measurements associated with these scenarios are denoted $\mathbf{z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_K\}$. Furthermore, to capture some model error due to the form of the turbulence model, we consider the set of I models $\mathscr{M} = \{M_1, \cdots, M_I\}$. Here we use the 5 models described in Section 3.2.

For each turbulence model and each scenario the calibration procedure results in a posterior on $\boldsymbol{\theta}$:

$$\hat{\boldsymbol{\theta}}_{i,k} \sim \boldsymbol{\theta} | M_i, \mathbf{S}_k, \mathbf{z}_k, \quad i = 1, \ldots, I,\ k = 1, \ldots, K,$$

whereby the nature of $\boldsymbol{\theta}$ depends on the model. Formally we write $\boldsymbol{\theta} \in \Theta_i$ for $i = 1, \ldots, I$ depending on the context.

Now let Δ be the Quantity of Interest in an unmeasured scenario $\mathbf{S}' \notin \mathscr{S}$. This is a velocity at $y^{+\prime}$, modelled as

$$\Delta = u_i^+(y^{+\prime}, \mathbf{S}'; \boldsymbol{\theta}), \tag{3.3}$$

but it could equally be a quantity of entirely different type, such as turbulent kinetic energy. In (3.3), $u_i^+(\cdot)$ indicates the simulation code under turbulence model M_i. Evaluating $p(\Delta|\mathbf{z})$ conditional on $\mathscr{M}$ and $\mathscr{S}$ is commonly called Bayesian Model Averaging (BMA) – but in our case could equally be called Bayesian Model-Scenario Averaging (BMSA), since the choice of $\mathscr{S}$ is critical to the validity of the resulting estimate. In par-

ticular

$$
\begin{aligned}
p(\Delta|\mathbf{z}) &\overset{(a)}{=} \sum_{i=1}^{I}\sum_{k=1}^{K}\int_{\Theta_i} p(\Delta, M_i, \mathbf{S}_k, \boldsymbol{\theta}|\mathbf{z}_k)\,\mathrm{d}\boldsymbol{\theta} \\
&\overset{(b)}{=} \sum_{i=1}^{I}\sum_{k=1}^{K}\int_{\Theta_i} p(\Delta|M_i,\boldsymbol{\theta})\,p(\boldsymbol{\theta}|M_i,\mathbf{S}_k,\mathbf{z}_k)\mathbb{P}(M_i,\mathbf{S}_k|\mathbf{z}_k)\,\mathrm{d}\boldsymbol{\theta} \\
&\overset{(c)}{=} \sum_{i=1}^{I}\sum_{k=1}^{K}\int_{\Theta_i} \underbrace{p(\Delta|M_i,\boldsymbol{\theta})\,p(\hat{\boldsymbol{\theta}}_{i,k})}_{p(\hat{\Delta}_{i,k})}\mathbb{P}(M_i|\mathbf{S}_k,\mathbf{z}_k)\mathbb{P}(\mathbf{S}_k)\,\mathrm{d}\boldsymbol{\theta}.
\end{aligned} \tag{3.4}
$$

Here we use $\mathbb{P}(\cdot)$ to denote a probability mass function (pmf) rather than a pdf. The equalities in the above are established using: (a) the law of total probability, and the fact that scenario k depends only on $\mathbf{z}_k$; (b) the definition of conditional probability (twice), and the conditional independence of Δ from $\mathbf{z}$ and $\mathbf{S}_k$ given $\boldsymbol{\theta}$; (c) the definition of conditional probability, and the conditional independence of $\mathbf{S}_k$ and $\mathbf{z}_k$. The braced term $\hat{\Delta}_{i,k}$ is equivalent to the posterior $\hat{\boldsymbol{\theta}}_{i,k}$ propagated through the simulation code for Δ (3.3). The weights $\mathbb{P}(M_i|\mathbf{S}_k,\mathbf{z}_k)$ are commonly interpreted as the level of evidence for a given model [11], in this case specific to a given scenario. Finally we are free to choose $\mathbb{P}(\mathbf{S}_k)$, and it is reasonable to preferentially weight those scenarios that are (in some sense) similar to the prediction scenario $\mathbf{S}'$.

The posterior probability of model M_i can be calculated through another application of Bayes' rule

$$
\mathbb{P}\left(M_i \mid \mathbf{S}_k, \mathbf{z}_k\right) = \frac{p\left(\mathbf{z}_k \mid M_i, \mathbf{S}_k\right)\mathbb{P}\left(M_i \mid \mathbf{S}_k\right)}{\sum_{j=1}^{J} p\left(\mathbf{z}_k \mid M_j, \mathbf{S}_k\right)\mathbb{P}\left(M_j \mid \mathbf{S}_k\right)}, \tag{3.5}
$$

$\forall k \in \{1,2,\cdots,K\}$, where additionally

$$
p\left(\mathbf{z}_k|M_i,\mathbf{S}_k\right) = \int_{\Theta_i} p\left(\mathbf{z}_k|\boldsymbol{\theta},M_i,\mathbf{S}_k\right)p\left(\boldsymbol{\theta}|M_i,\mathbf{S}_k\right)\mathrm{d}\boldsymbol{\theta}. \tag{3.6}
$$

Furthermore, the axioms of probability require that

$$
\sum_{i=1}^{I}\mathbb{P}\left(M_i \mid \mathbf{S}_k\right) = 1, \quad \text{and} \quad \sum_{k=1}^{K}\mathbb{P}(\mathbf{S}_k) = 1. \tag{3.7}
$$

Following Draper [5], and the derivation in [17], the leading moments of $p(\Delta|\mathbf{z})$ can be written as

$$
\mathbb{E}\left[\Delta|\mathbf{z}\right] = \sum_{i=1}^{I}\sum_{k=1}^{K}\mathbb{E}[\hat{\Delta}_{i,k}]\mathbb{P}\left(M_i|\mathbf{S}_k,\mathbf{z}_k\right)\mathbb{P}\left(\mathbf{S}_k\right), \tag{3.8}
$$

$$
\begin{aligned}
\mathrm{Var}\left[\Delta \mid \mathbf{z}\right] = &\sum_{i=1}^{I}\sum_{k=1}^{K}\mathrm{Var}[\hat{\Delta}_{i,k}]\mathbb{P}\left(M_i \mid \mathbf{S}_k,\mathbf{z}_k\right)\mathbb{P}\left(\mathbf{S}_k\right) \\
&+ \sum_{i=1}^{I}\sum_{k=1}^{K}\left(\mathbb{E}[\hat{\Delta}_{i,k}] - \mathbb{E}\left[\Delta \mid \mathbf{S}_k,\mathbf{z}_k\right]\right)^2\mathbb{P}\left(M_i \mid \mathbf{S}_k,\mathbf{z}_k\right)\mathbb{P}\left(\mathbf{S}_k\right) \\
&+ \sum_{k=1}^{K}\left(\mathbb{E}\left[\Delta \mid \mathbf{S}_k,\mathbf{z}_k\right] - \mathbb{E}\left[\Delta \mid \mathbf{z}\right]\right)^2\mathbb{P}\left(\mathbf{S}_k\right),
\end{aligned} \tag{3.9}
$$

Here, $\mathbb{E}[\hat{\Delta}_{i,k}]$ and $\mathrm{Var}[\hat{\Delta}_{i,k}]$ are the mean and variance obtained by propagating $\hat{\boldsymbol{\theta}}_{i,k}$ though the code of Δ. Furthermore,

$$\mathbb{E}[\Delta \mid \mathbf{S}_k, \mathbf{z}_k] = \sum_{i=1}^{I} \mathbb{E}[\hat{\Delta}_{i,k}] \mathbb{P}(M_i \mid \mathbf{S}_k, \mathbf{z}_k).$$

This decomposition of the variance into 3 (positive) terms, allows for additional interpretation. The first term on the right-hand side is called the *within-model, within-scenario* variance. This can be considered a measure of the effect of the variance in all the $\hat{\boldsymbol{\theta}}_{i,k}$ on the variance in $\Delta|\mathbf{z}$ – specifically an average weighted according to the importance of each model and scenario. The second term is the *between model, within scenario* variance. It is large when when different models applied to the same scenario give different predictions. The last term is the *between scenario* variance. It accounts for the fact that the same model M_i, calibrated under different scenarios, results in different posterior $\boldsymbol{\theta}$ and therefore different predictions of Δ [7].

3.4.2. SMART SCENARIO WEIGHTING

In order to close the above system we must specify $\mathbb{P}(M_i|\mathbf{S}_k)$ and $\mathbb{P}(\mathbf{S}_k)$. As we will show in Section 3.5.3, simply specifying a uniformly distributed $\mathbb{P}(\mathbf{S}_k)$ yields an unacceptably large variance. This is due to the large spread of $\hat{\boldsymbol{\theta}}_{i,k}$ between scenarios, see Section 3.5.2, resulting in a large spread of Δ. We would like to preferentially weight those scenarios in $\mathscr{S}$ which are – in some relevant sense – similar to $\mathbf{S}'$. Furthermore we would like to do this in an automatic, geometry- and flow-independent way.

We propose choosing $\mathbb{P}(\mathbf{S}_k)$ to preferentially weight those scenarios for which all models in $\mathscr{M}$ give similar mean predictions of Δ, under the coefficients $\hat{\theta}_{i,k}$. Equally, we assign low probabilities to scenarios with large scatter in the model predictions of Δ. The rationale is that, if $\mathbf{S}'$ is very similar to $\mathbf{S}_k$, the 4 models are expected to give similar predictions under $\hat{\boldsymbol{\theta}}_{i,k}$, as they have each been calibrated to the same data $\mathbf{z}_k$, containing physical processes similar to $\mathbf{S}'$. If $\mathbf{S}'$ is completely different to $\mathbf{S}_k$, the choice of $\hat{\boldsymbol{\theta}}_{i,k}$ is likely to be inappropriate, and the models – provided they are sufficiently heterogeneous – will deliver different predictions.

In particular we define the scenario probabilities as

$$\begin{aligned} \mathscr{E}_k &= \sum_{i=1}^{I} \| \mathbb{E}[\hat{\Delta}_{i,k}] - \mathbb{E}[\Delta \mid \mathbf{S}_k, \mathbf{z}_k] \|_2 \\ \mathbb{P}(\mathbf{S}_k) &= \frac{\mathscr{E}_k^{-p}}{\sum_{k=1}^{K} \mathscr{E}_k^{-p}}, \quad \forall \mathbf{S}_k \in \mathscr{S}. \end{aligned} \tag{3.10}$$

Here $\mathscr{E}_k$ represents the measure for the prediction similarity, and p is a tuning parameter, controlling the degree to which the preferred scenarios are weighted. Setting $p = 0$ yields uniform probabilities, and as $p \to \infty$ a single scenario is selected (provided that the $\mathscr{E}_k$ have distinct values).

3.4.3. NUMERICAL EVALUATION

In CFD evaluating the likelihood (2.19) involves running a CFD code. Here, the posteriors $\hat{\boldsymbol{\theta}}_{i,k}$ are approximated with Markov-chain Monte-Carlo (McMC) methods [10, 16], which provide samples $\hat{\boldsymbol{\theta}}^n_{i,k}, n = 1,2,\ldots,N$ from $\hat{\boldsymbol{\theta}}_{i,k}$. This chain approximation is thereafter used to approximate everything else by Monte-Carlo, e.g.:

$$\mathbb{E}(\hat{\Delta}_{i,k}) \approx \frac{1}{N}\sum_{n=1}^{N} u_i^+\left(y^{+\prime}, \mathbf{S}'; \hat{\boldsymbol{\theta}}^n_{i,k}\right). \tag{3.11}$$

Clearly the process outlined here is impractically costly for an expensive CFD simulation code applied to a complex geometry – here it is possible due to the use of a boundary-layer code. Our intention in this paper is to explore the statistical framework, and determine to what extent prediction of model inadequacy is possible, under the assumption that numerical costs can be handled. Success will motivate us to study numerical techniques for improving the computational efficiency of the procedure. There are many possible approaches, including: surrogate modelling [6], Occam's window [11], zero-variance McMC methods [18], etc. Also, a means of reducing the computational cost is described in Section 3.5.5. Furthermore, the calibration step is perhaps necessary for simple geometries only, giving a database of coefficient distributions $\hat{\boldsymbol{\theta}}_{i,k}$ which can then be applied to more complex geometries. A flowchart of our entire procedure, from calibration to BMA prediction, can be found in Appendix A.

3.5. RESULTS

3.5.1. HPD INTERVALS OF COEFFICIENT POSTERIORS

To evaluate the success of the individual calibrations, we inspect the coefficient posterior distributions $\hat{\boldsymbol{\theta}}_{i,k}$. Instead of plotting full multi-dimensional distributions, we plot HPD intervals of the two most sensitive parameters for each model, shown in Figures 3.7 and 3.8. The scenarios are presented roughly in order of increasing pressure-gradient, but with the zero-pressure-gradient case (1400) first. The coefficient κ was the most well-informed coefficient for all models except $k-\varepsilon$, for which it was the second most well-informed coefficient. As can be seen by the small width of the individual HPD intervals compared to the prior width, these coefficients have been informed by the data. This was to be expected considering the results of the sensitivity analysis of Section 3.3.2. The remaining closure coefficients were not informed well enough by the data to be interesting.

The four κ's presented in Figure 3.7 are not the same κ: the coefficient has a different meaning and influence in each model. Neither should they be confused with any model-independent, or "true" von Karman constant. Nonetheless, ideally κ would be independent of scenario for a given model. Figure 3.7 clearly shows statistically significant variation in calibrated coefficients across scenarios. This may be interpreted as pressure-gradient dependence in the models, and we might conclude, for example, that Spalart-Allmaras can reproduce attached boundary-layer flows with a single set of coefficients better than Baldwin-Lomax can.

This observed coefficient variation also supports our choice of a statistical model with scenario-dependent coefficients. A standard calibration approach is to assume

there exists a single best value of model coefficients $\boldsymbol{\theta}^\star$, and then use all available data to inform this value, with model inadequacy captured by an η-like term only [12]. These results question the validity of that assumption in the case of turbulence closure models.

Figure 3.8 shows the HPD intervals of the 2nd most sensitive coefficients (or the 1st most sensitive in the case of $k-\varepsilon$). In general coefficients are less well informed and the scenario scatter is greater, supporting the view that optimal closure coefficients are strongly dependent upon the pressure gradient.

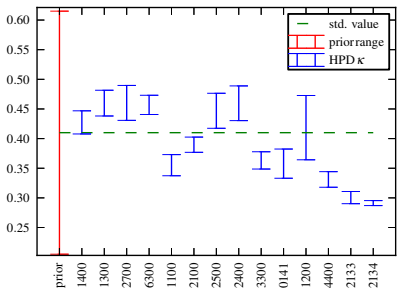
The HPD intervals of the hyper-parameter σ_{hp} give insight into the model error that remains after optimal closure coefficients have been found for each scenario, see Figure 3.9. HPD interval values close the the 0.1 boundary indicates that the turbulence model has to rely on the η term in order to capture the data better. All models encounter flow cases where this occurs, although some more than others. Especially the Spalart-Allmaras model shows high HPD intervals of σ_{hp} in most cases. This indicates that, of the models in our set $\mathscr{M}$, it is worst at matching the experimental velocity profiles exactly. As we shall see in the next section, BMA provides a coherent mechanism for penalizing models that do not perform well on the calibration dataset.

3.5.2. POSTERIOR MODEL PROBABILITY

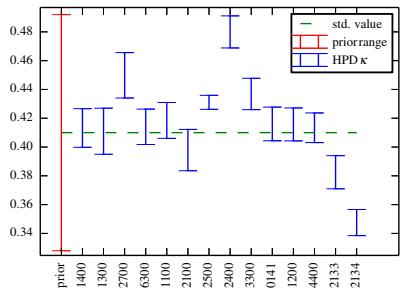
For a given flow case, once we have calibrated each model in the set $\mathscr{M}$ on the data $\mathbf{z}_k$ we can calculate the posterior model probability (3.5). This posterior model probability $\mathbb{P}(M_i \mid \mathbf{S}_k, \mathbf{z}_k)$ can be interpreted as the evidence for model M_i given the available data, and the other models in $\mathscr{M}$. It should be emphasized that the posterior model probabilities are conditional on the choice of models in $\mathscr{M}$, i.e. $\mathbb{P}(M_i \mid \mathbf{S}_k, \mathbf{z}_k, \mathscr{M})$. Since in our analysis the set $\mathscr{M}$ remains fixed we drop it from the notation for the sake of brevity.

We compute $\mathbb{P}(M_i \mid \mathbf{S}_k, \mathbf{z}_k)$ for each flow case k in the set $\mathscr{S}$, assuming a uniform probability mass function for $\mathbb{P}(M_i \mid \mathbf{S}_k)$. The results can be found in Figure 3.10. From this figure it becomes clear that also the posterior model probability is dependent upon the applied flow case. Which model has the highest posterior probability given the data can change significantly from one flow case to another. Thus, both the posterior closure coefficients and the posterior model probabilities are functions of the pressure gradient. There is no clear "best" $\boldsymbol{\theta}$ or M_i that will outperform its competitors in $\mathscr{M}$ for every scenario in $\mathscr{S}$. For a predictive case that is not in $\mathscr{S}$ it would be hard to select the best model and coefficient set *a priori*, given that in prediction we do not have access to experimental data by definition.

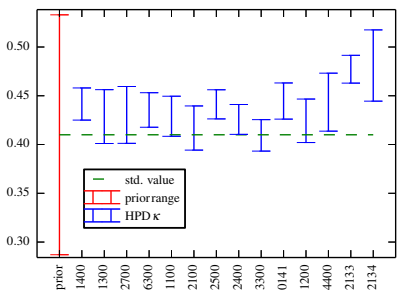
More speculatively, these results could be interpreted as a explanation for why it is that no clear "winner" amongst turbulence closure models has been found, see e.g. [26], and why there are a wide spread of closure coefficients recommended and used in literature and industry. Note in particular that no clear superiority of the more advanced Reynolds-Stress model over the other is noticed, even if it does exhibit somewhat higher probabilities for strong adverse pressure gradient cases. On the other hand, no clear inferiority of the algebraic model is observed either, not even for adverse pressure gradient scenarios. Indeed – based on these results – the practice of tuning a set of deterministic closure coefficients for specific applications seems futile, even if that tuning is done in a rigourous way. Predictions made with existing models and deterministic coefficients seem to be subject to a degree of model error that is not substantially reducible by pa-



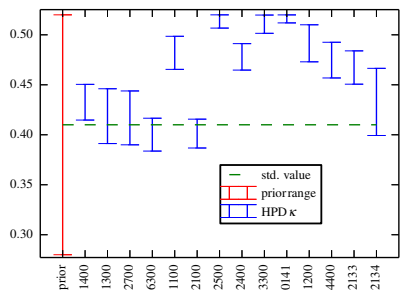
(a) κ of the $k-\varepsilon$ model.



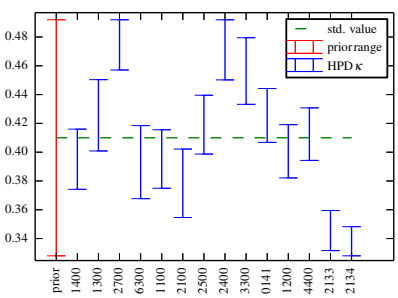
(b) κ of the $k-\omega$ model.



(c) κ of the Spalart-Allmaras model.

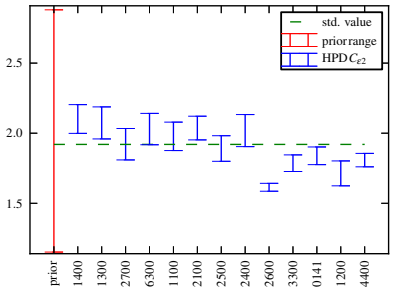


(d) κ of the Baldwin-Lomax model.

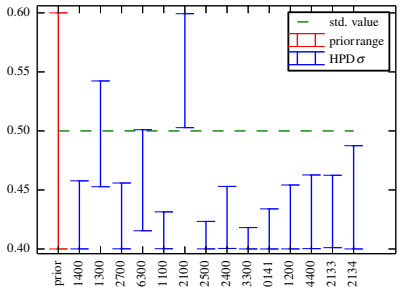


(e) κ of the stress-ω model.

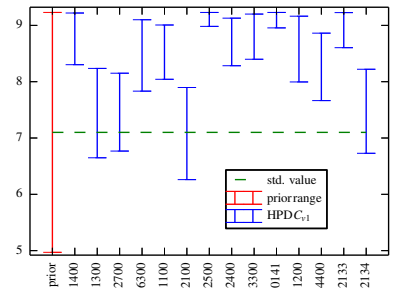
Figure 3.7: The HPD intervals of κ for all $\left(M_i, \mathbf{S}_k\right)$ combinations.



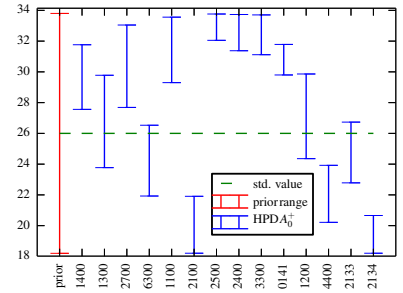
(a) $C_{\varepsilon 2}$ of the $k-\varepsilon$ model.



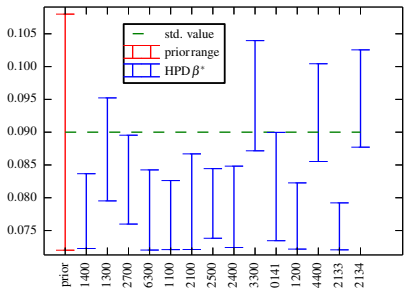
(b) σ of the $k-\omega$ model.



(c) C_{v1} of the Spalart-Allmaras model.

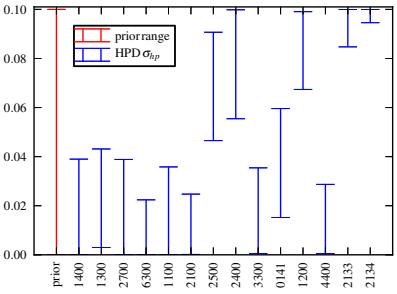


(d) A_0^+ of the Baldwin-Lomax model.

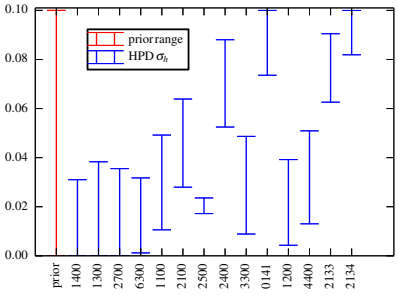


(e) β^* of the stress-ω model.

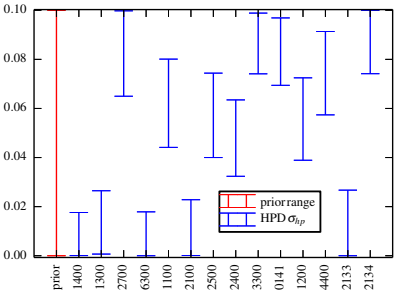
Figure 3.8: The HPD intervals of second-most well-informed closure coefficients for all $(M_i, \mathbf{S}_k)$ combinations.



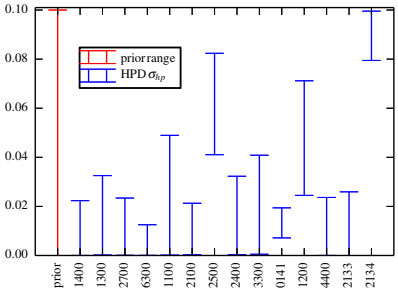
(a) σ_{hp} of the $k-\varepsilon$ model.



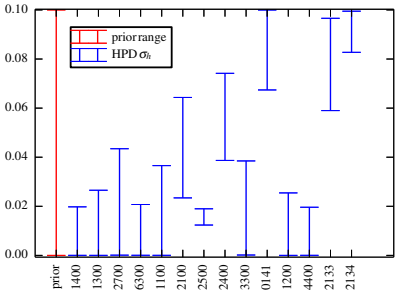
(b) σ_{hp} of the $k-\omega$ model.



(c) σ_{hp} of the Spalart-Allmaras model.



(d) σ_{hp} of the Baldwin-Lomax model.



(e) σ_{hp} of the stress-ω model.

Figure 3.9: The HPD intervals hyper parameter σ_{hp} for all $(M_i, \mathbf{S}_k)$ combinations.

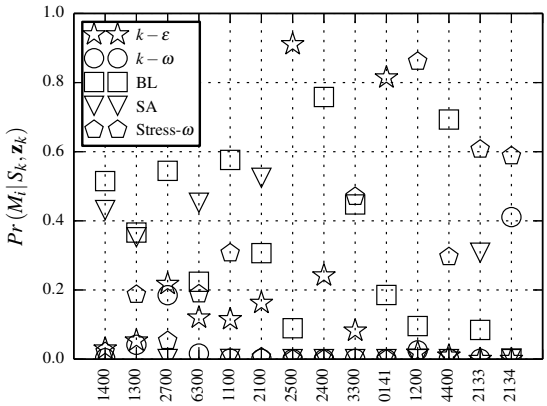


Figure 3.10: The distribution of $\mathbb{P}\left(M_i \mid \mathbf{S}_k, \mathbf{z}_k\right)$ for all models and scenarios.

rameter tuning. For this reason we choose to make model- and scenario-averaged predictions, using our framework of Section 3.4.1.

3.5.3. PREDICTIONS WITH BAYESIAN MODEL-SCENARIO AVERAGING

In this section the methodology of Section 3.4.1 is applied to prediction of boundary-layer flows at pressure gradients for which (nominally) no data is available. Our goal is to assess the predictive capability of (3.8)-(3.9). Our predictions are in the form of mean and variance of velocity profiles for the unmeasured flows, that is $\mathbb{E}(\Delta|\mathbf{z})$ and $\mathrm{Var}(\Delta|\mathbf{z})$ where $\Delta = u^+(y^+)$.

We proceed as follows: one case is selected from the 14 flows in Table 2.1 as validation flow, $\mathbf{S}'$. The experimental data for this case is completely excluded from the analysis, and only used for comparison purposes in the final plots. The remaining cases form the set of calibration scenarios $\mathscr{S}$, so that $K = |\mathscr{S}| = 13$. Each of these cases is calibrated independently, following Section 2.4, to give the coefficient posteriors per scenario and per model, $\hat{\boldsymbol{\theta}}_{i,k}$. The moments $\mathbb{E}[\hat{\Delta}_{i,k}]$ and $\mathrm{Var}[\hat{\Delta}_{i,k}]$ are obtained by propagating the $K \times I$ McMC chains from the calibration step through (3.3). The chains are our best available representation of the coefficient posteriors, and therefore no uncertainty information gets lost in this step. Finally (3.8)-(3.9) are evaluated.

To close (3.8) it remains only to specify the probability mass function (pmf) for $\mathbb{P}(\mathbf{S}_k)$. This could be based on expert opinion. For instance if one believes that the current prediction flow is subject the mildly-adverse pressure gradients, the posterior parameter distributions that were calibrated under similar circumstances could be favored through $\mathbb{P}(\mathbf{S}_k)$. Ideally an automatic procedure is preferred, or if no information is available one must admit ignorance and select a uniform pmf. In the following sections first predictions based on this last choice are examined as a worst-case; then a smart automated selection of $\mathbb{P}(\mathbf{S}_k)$ is developed

RESULTS OF UNIFORM $\mathbb{P}(\mathbf{S}_k)$

In the worst-case we are completely ignorant of the relationship between $\mathbf{S}'$ and the members of $\mathscr{S}$, and select a uniform pmf for $\mathbb{P}(\mathbf{S}_k)$, i.e. assuming that each pressure-gradient scenario is equally likely. We show the results for two validation flow cases in Figure 3.11, an "easy" case (flow 1300), and a "difficult" case subject to a strongly adverse pressure gradient (flow 4400). The mean prediction for case 1300 in Figure 3.11(a) is reasonable, it falls within the region of experimental uncertainty for all y^+. However the standard deviation of the prediction $\sigma_\Delta := \sqrt{\mathrm{Var}[\Delta \mid \mathbf{z}]}$ is almost 2 u^+ units. As can be seen from the 1σ, 2σ and 3σ quantiles, plotted in blue, this is very large, and a clear over-estimation of the true model error. Nonetheless the fact that the correct solution lies within the range of our prediction should be interpreted as a successful prediction – albeit perhaps not a very useful one.

Thanks to the variance decomposition in (3.9), the source of the large variance can be diagnosed, see Figure 3.11(b). It seems the contributions from all three sources are of the same order, but dominated somewhat by the *between scenario* variance. It is exactly this variance that stands to be reduced most by smart weighting of scenario probabilities.

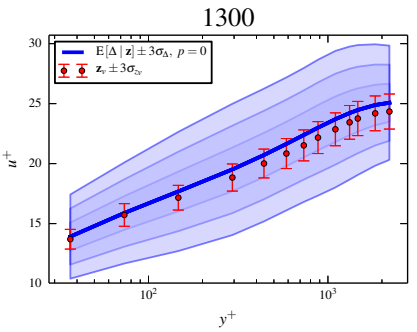
The story for case 4400 is essentially the same, where again the variance is a substantial over-estimation.

One could argue that it is better to be conservative in predictions of uncertainty in the QoI, however the amount of uncertainty should also not be so high as to make the prediction useless. We are of the opinion that the amount of uncertainty (in these relatively simple flows) crosses that line. For instance the results for flow case 4400 show an uncertain maximum $3\sigma_\Delta$ range of ± 15 u^+ units in the defect layer, roughly 43% of the mean.

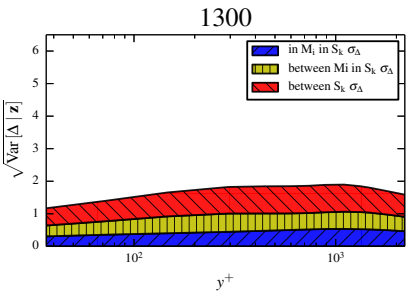
Ultimately the source of this large variance is the large spread of closure coefficient values across the calibration cases. As more extreme cases are added to $\mathscr{S}$ – with the goal of extending the range of validity of the predictions – this variance will only increase. This result echoes our previous results using p-boxes in Edeling et al. [7] which also lead to very large error bars. Unlike p-boxes however, our Bayesian Scenario Averaging framework is not limited to a uniform scenario weighting.

RESULTS OF SMART $\mathbb{P}(\mathbf{S}_k)$ WEIGHTING

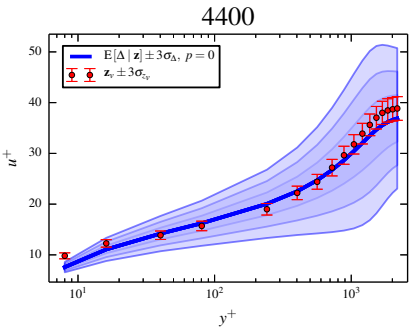
In order to bring σ_Δ down to realistic levels we use the model prediction-spread based $\mathbb{P}(\mathbf{S}_k)$ defined in (3.10). To clarify, we plot the model prediction spread for validation case 1300 in Figure 3.12. This corresponds to the envelope of predictions provided by the different models, using posterior coefficients associated to a particular calibration scenario. Figure 3.12 displays such a prediction spread for different calibrations scenarios. Note that the spread is quite small for predictions made with posteriors coming from cases similar to 1300, and that it steadily grows as we move to the predictions made with $\hat{\boldsymbol{\theta}}_{i,k}$ calibrated under strongly-adverse $\mathbf{S}_k$. It is this behavior that is used to inform $\mathbb{P}(\mathbf{S}_k)$ via $\mathscr{E}_k$, and the effect is amplified by selecting a $p \geq 1$. The resulting scenario weights for $p = 1, 4, 8$ are shown in 3.13. Notice that in both cases the sensor favours cases with pressure-gradients similar to the validation scenario. For case 1300 it selects favorable pressure-gradients, and for case 4400 adverse gradients. In both cases with $p = 8$ most scenarios have an almost-zero weight and can be neglected.



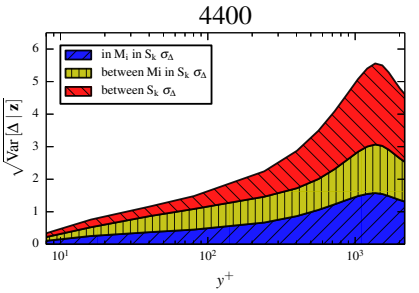
(a) The BMSA prediction for validation case 1300.



(b) The BMSA standard deviation for validation case 1300.



(c) The BMSA prediction for validation case 4400.



(d) The BMSA standard deviation for validation case 4400.

Figure 3.11: Selected BMSA predictions with uniform $\mathbb{P}(\mathbf{S}_k)$.

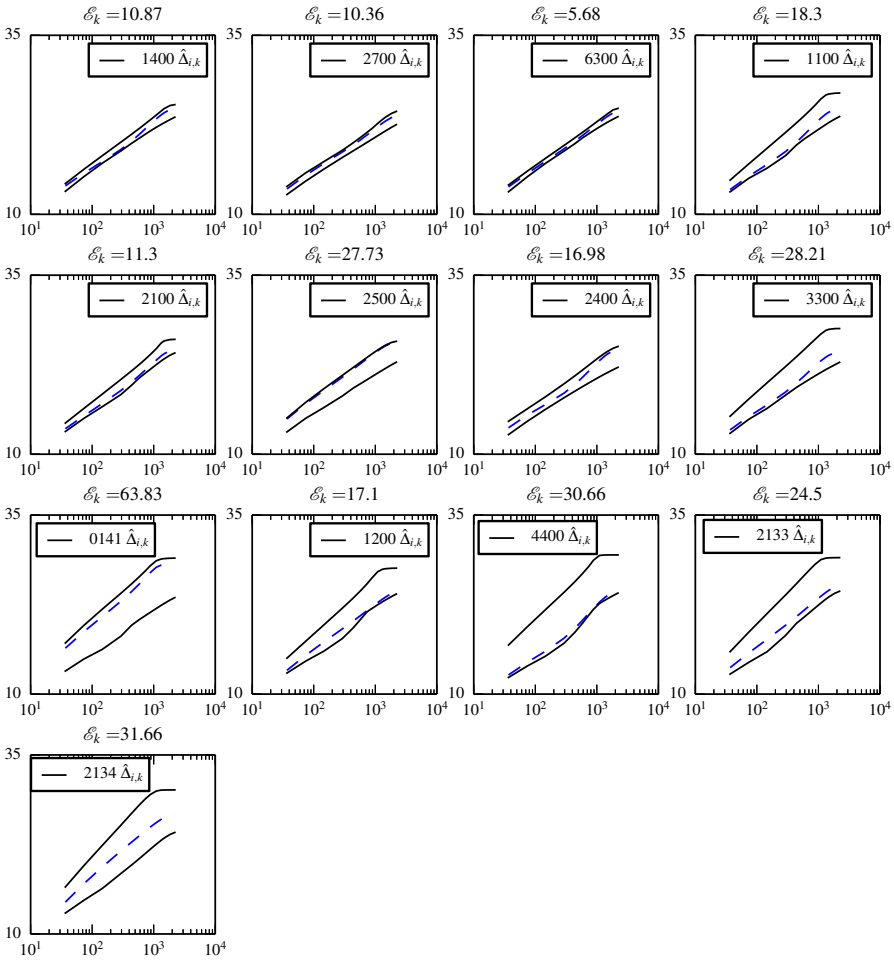
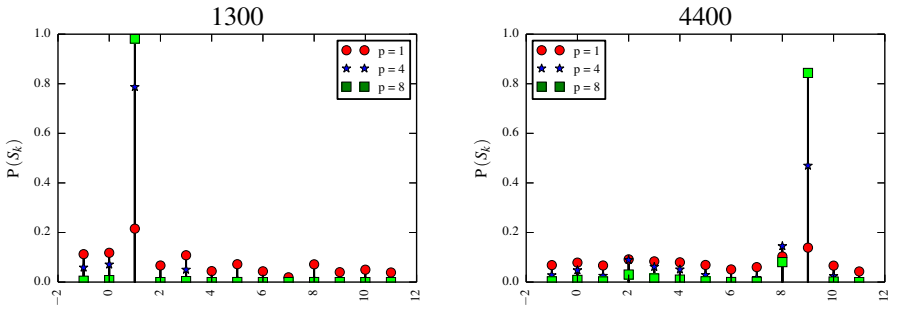


Figure 3.12: The model prediction spread for validation case 1300. Each subplot shows the envelope (solid lines) formed by the 5 $\hat{\Delta}_{i,k}$, $i = 1,\cdots,5$, as well as the BMA prediction (dotted line) for a given $\mathbf{S}_k$. $\mathscr{E}_k$ is a measure for the spread as defined in (3.10). The unit of all vertical axes is u^+, and the horizontal axes display y^+ units.



(a) Validation case 1300.

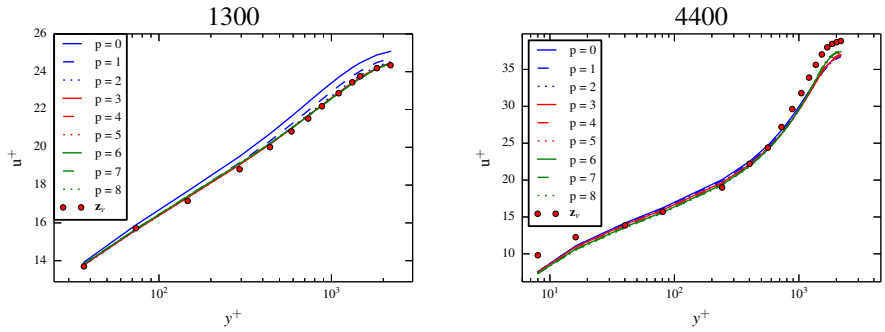
(b) Validation case 4400.

Figure 3.13: Two $\mathscr{E}_k$-based $\mathbb{P}(\mathbf{S}_k)$ distributions.

We now examine how this new $\mathbb{P}(\mathbf{S}_k)$ affects the BMSA predictions, for different values of p. In Figure 3.14a we plot $\mathbb{E}[\Delta|\mathbf{z}]$ for $p = 0, 1, \cdots, 8$ for case 1300. We see that for $p = 0, 1$ the predictions are slightly off the experimental data. For all $p > 1$ the predictions overlap and match the data quite well. However, in most cases we found that the value of p had no significant effect on the BMSA prediction $\mathbb{E}[\Delta|\mathbf{z}]$, see e.g. Figure 3.14b. The posterior variance (3.9) on the other hand is significantly affected by p. This can be inferred from Figure 3.15, where we show the same predictions with quantified uncertainty as in the previous section, but using the smart weighting (with $p = 8$). As discussed, the BMSA prediction mean for 1300 is improved, and now lies right on top of the measurement points. In addition the maximum σ_Δ value has more than halved from 2.0 to 0.8. It is now comparable to the level of measurement error indicating that our prediction is as accurate as our reference experimental data. For 4400 the prediction with $p = 0$ was already reasonable, and remains unchanged for $p > 0$. Again, the maximum σ_Δ has decreased, in this case from 5.5 to roughly 3.7 u^+-units.

From Figures 3.15(b) and (d), we observe the variance reduction was most strong for *between scenario* variance as expected, but *between model, within scenario* variance also took a substantial hit. The remaining variance is dominated by the term representing the lack of identification of closure coefficients in the individual calibrations, i.e. the variance of $\hat{\boldsymbol{\theta}}_{i,k}$. This can be reduced by using more data (and more informative data) in the calibrations, to better inform coefficients. Alternatively, approximating each $\hat{\boldsymbol{\theta}}_{i.k}$ by its maximum *a posteriori* (MAP) estimate, would reduce the computational cost of evaluating $\mathbb{E}(\hat{\Delta}_{i,k})$, and set the blue region in the above plots to zero. Given that the blue region dominates the variance however, this is an approximation of questionable validity.

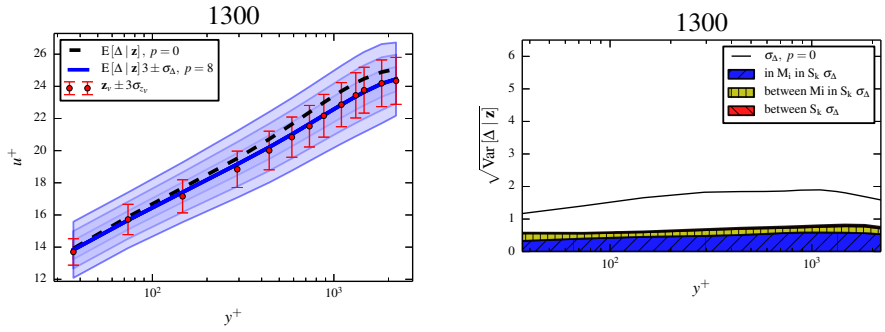
Finally, to demonstrate the generality and robustness of these results, the above analysis is performed for prediction of each one of the 14 boundary-layer cases, each time using data from the remaining 13 cases only. The results are summarized in Table 3.3. There we show the set of $\mathbf{S}_k$ which make up 75 % or more of $\mathbb{P}(\mathbf{S}_k)$, the relative error of the BMSA prediction (3.8) defined as $\epsilon'_\infty := \|\mathbb{E}[\Delta \mid \mathbf{z}] - \mathbf{z}_v\|_\infty / \|\mathbf{z}_v\|_\infty$ and the coefficient of variation (CoV), given by $C_{cov} := \sigma_\Delta / \mathbb{E}[\Delta \mid \mathbf{z}]$. The CoV is a measure of the solution



(a) Validation case 1300.

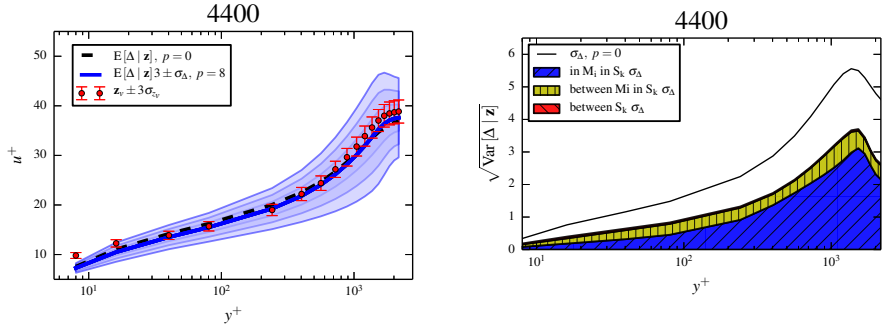
(b) Validation case 4400.

Figure 3.14: $\mathbb{E}[\Delta|\mathbf{z}]$ vs p for two validation flow cases.



(a) The new and old BMSA prediction for validation case 1300.

(b) The new BMSA standard deviation for validation case 1300.



(c) The new and old BMSA prediction for validation case 4400.

(d) The new BMSA standard deviation for validation case 4400.

Figure 3.15: Selected BMSA predictions with a $\mathscr{E}_k$-based $\mathbb{P}(\mathbf{S}_k)$.

variability with respect to the mean. Note that most validation cases favour a $\mathbf{S}_k$ with a pressure-gradient classification similar to their own, only the mild- and moderately adverse $\mathbf{S}'$ tend to mix different type of scenarios. Also, almost all BMSA predictions are satisfactory, which can be inferred from the fact that they have both small ϵ'_∞ and max C_{cov}. The only real exceptions are cases 2133 and 2134. These are cases with an extremely adverse pressure gradients, for which all turbulence models in our set $\mathscr{M}$ struggle to make accurate predictions.

Table 3.3: Results validation flow cases using (3.10).

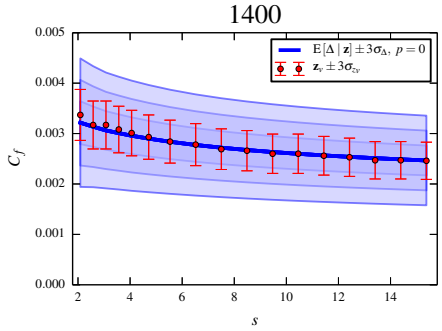
Val. case	$\{\mathbf{S}_k \mid \sum_k \mathbb{P}(\mathbf{S}_k) \geq 0.75\}$	ϵ'_∞	max C_{cov}
1400 (zero)	6300	0.030	0.042
1300 (fav)	6300	0.013	0.042
2700 (fav)	6300, 1300	0.030	0.058
6300 (fav)	1300	0.011	0.045
1100 (mild adv)	1200,2100,6300	0.062	0.061
2100 (mild adv)	6300,1400	0.028	0.055
2500 (mild adv)	1100,1400,6300	0.039	0.057
2400 (mod adv)	2500,1100,2700,1400	0.013	0.067
3300 (mod adv)	1100	0.046	0.065
0141 (strong adv)	4400	0.051	0.081
1200 (strong adv)	1100,2133,2100	0.043	0.091
4400 (strong adv)	1200	0.050	0.108
2133 (very strong adv)	0141,4400,1200	0.14	0.099
2134 (very strong adv)	0141,4400,1200	0.29	0.099

3.5.4. C_f PREDICTION

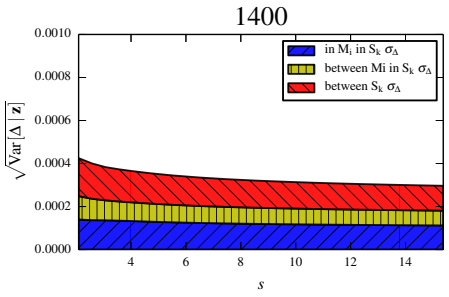
Up this point the QoI has always been u^+ profiles, and all posterior $\hat{\boldsymbol{\theta}}_{i,k}$ have been informed using only experimental u^+ data. The question now arises whether these distributions are also effective when we try to predict a QoI of a different nature. To that end we make predictions for a variety of different skin-friction profiles, using the same procedure, distributions and posterior model probabilities as in the preceding sections. Thus, the only change is our new QoI, the dimensionless skin-friction coefficient $C_f := \tau_w/(\rho u_\infty^2/2)$. Here τ_w is the wall-shear stress and u_∞ is the freestream velocity.

The results for a uniform $\mathbb{P}(\mathbf{S}_k)$ distribution and a large range of different scenarios are shown in Figure 3.16. We see that all predictions fall within the range of experimental uncertainty, and in most cases are quite well centered on the data points. Only the 3300 prediction is slightly off center. The amount of uncertainty in the predictions is reasonable, $\mathbb{E}[\Delta|\mathbf{z}] \pm \sigma_\Delta$ being in most cases contained within the region of experimental uncertainty.

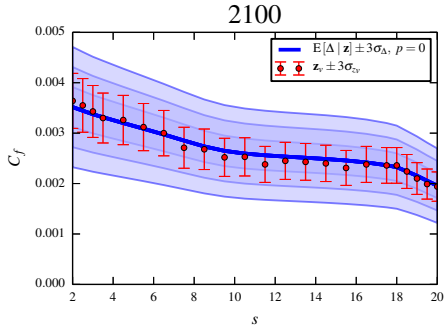
As in the case of the u^+ predictions, we now examine the effect of our scenario-weighting procedure, starting with the influence of p on $\mathbb{E}[\Delta|\mathbf{z}]$. In Figure 3.17a we plot $\mathbb{E}[\Delta|\mathbf{z}]$ vs p on the same vertical scale as used in Figure 3.16. It is again clear that the



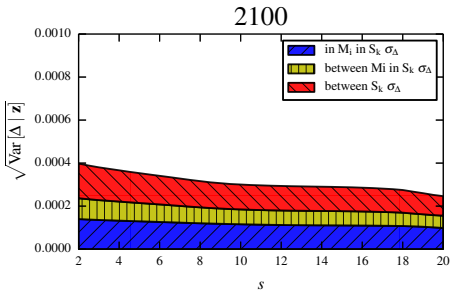
(a) The BMSA prediction for case 1400.



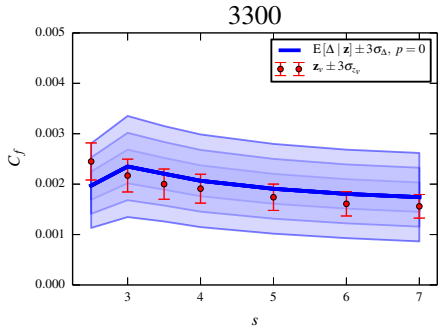
(b) The BMSA std. dev. for case 1400.



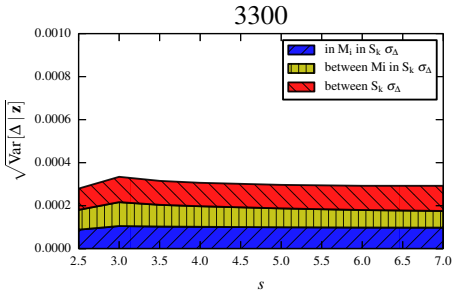
(c) The BMSA prediction for case 2100.



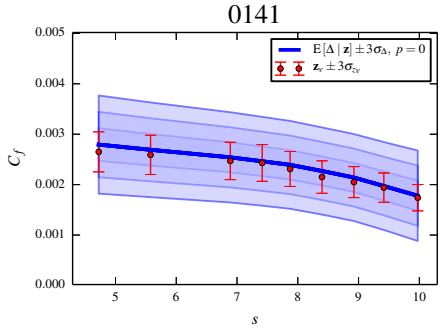
(d) The BMSA std. dev. for case 2100.



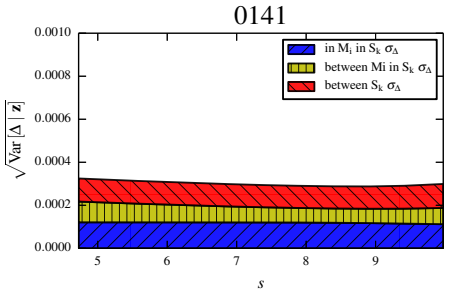
(e) The BMSA prediction for case 3300.



(f) The BMSA std. dev. for case 3300.

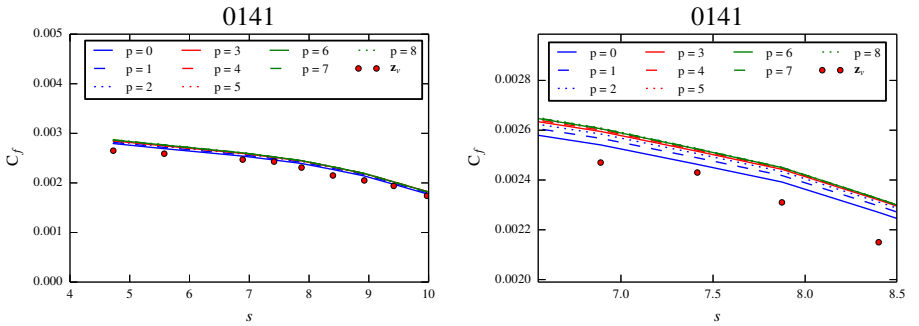


(g) The BMSA prediction for case 0141.



(h) The BMSA std. dev. for case 0141.

Figure 3.16: Selected BMSA predictions for C_f with uniform $\mathbb{P}(\mathbf{S}_k)$.



(a) Validation case 0141.

(b) Validation case 0141, zoomed in.

Figure 3.17: $\mathbb{E}[\Delta|\mathbf{z}]$ vs p. Here, $\Delta = C_f$.

BMSA predictions are robust, as they are relatively invariant considering the large range of the exponent p. When zooming in on $\mathbb{E}[\Delta|\mathbf{z}]$ (see Figure 3.17b), we see that unlike the u^+ case, the predictions move slightly away from the data with increasing p, but basically become insensitive to p starting from $p = 3$. When we examine the model prediction spread for 0141 in Figure 3.18, it becomes clear that predictions can be more in agreement with each other for flow cases not similar to 0141, which underlines the importance of considering different calibration scenarios. Also, including posterior distributions calibrated for QoIs other than u^+ could possibly be beneficial. The prediction with quantified uncertainty for the same validation case and $p = 3$ is shown in Figure 3.19. We can again see that the variance is affected more by p than the prediction.

3.5.5. REDUCTION OF COMPUTATIONAL EFFORT - SCENARIO-AVERAGED POSTERIORS

The full BMSA approach, with I models and posterior distributions coming from K scenarios requires us to propagate $I \times K$ $p(\hat{\boldsymbol{\theta}}_{i,k})$ through the code for Δ. In order to use BMSA, only convergence in mean and variance is required for each propagation. This poses no problem for our current computational setup. Full propagation can be achieved within a single day with moderate computational resources, in our case parallel propagation using 14 cores. Calibration takes longer, but is still easily achieved. A single calibration can take anywhere from 1 hour up to (at the very most) 12 hours, depending on the turbulence model, the streamwise location of the data, and the number of specified steps in the Markov chain. With the mentioned computational resources, all flow-case model combinations can be calibrated in roughly a week time. This needs to be performed only once, after which the posteriors can be re-used (also for complex flow topologies) as much as needed in the predictive phase of the method.

However, the required propagation by BMSA using the same number of models and scenarios is unlikely to be computationally tractable when considering more complex flow topologies. This could be alleviated, e.g. through the use of surrogate models [19,

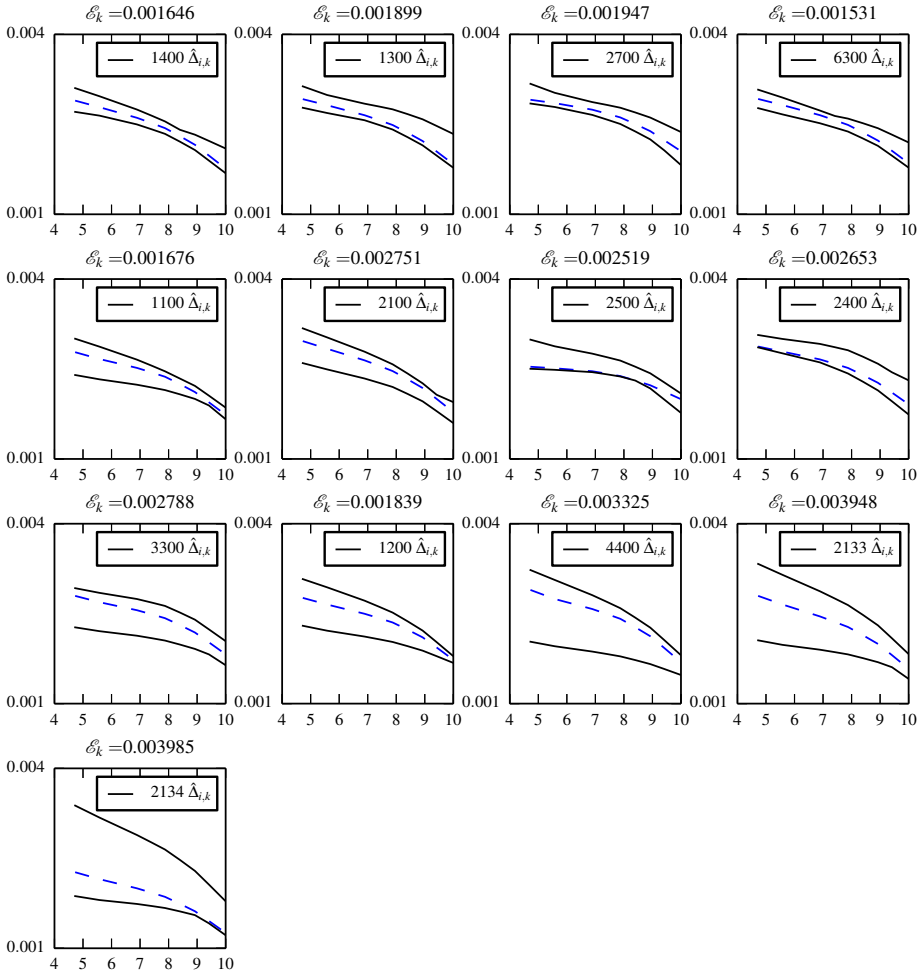
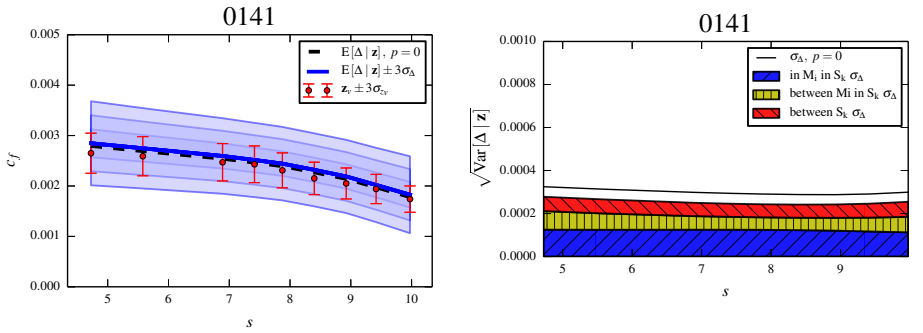


Figure 3.18: The model prediction spread of C_f for validation case 0141. Each subplot shows the envelope (solid lines) formed by the 5 $\hat{\Delta}_{i,k}$, $i = 1, \cdots, 5$, as well as the BMA prediction (dotted line) for a given $\mathbf{S}_k$. $\mathscr{E}_k$ is a measure for the spread as defined in (3.10). The vertical axes all display the skin-friction coefficient C_f, while the horizontal axes show the distance s.



(a) The BMSA prediction for case 0141.

(b) The BMSA std. dev. for case 0141.

Figure 3.19: BMSA prediction for C_f with $\mathscr{E}_k$-based $\mathbb{P}(\mathbf{S}_k)$ and $p=3$.

20, 25], and will be explored in future work. But even with the use of surrogate models, performing $I \times K$ propagations could prove to be computationally expensive. Therefore we propose to replace the $I \times K$ $p(\hat{\boldsymbol{\theta}}_{i,k})$ by I scenario-averaged $p(\tilde{\boldsymbol{\theta}}_i)$ defined as

$$p(\tilde{\boldsymbol{\theta}}_i) := p(\boldsymbol{\theta}|M_i,\mathbf{z}) = \sum_{k=1}^{K} p(\boldsymbol{\theta}|M_i,\mathbf{S}_k,\mathbf{z})\,\mathbb{P}(\mathbf{S}_k)\,. \tag{3.12}$$

Instead of using (3.8), the prediction for Δ is now made with

$$\mathbb{E}[\Delta|\mathbf{z}] = \sum_{i=1}^{I} \mathbb{E}[\tilde{\Delta}_i]\mathbb{P}(M_i|\mathbf{z})\,, \tag{3.13}$$

where $\mathbb{E}[\tilde{\Delta}_i] := \mathbb{E}[\Delta|M_i,\mathbf{z}]$ is obtained by propagating $p(\tilde{\boldsymbol{\theta}}_{i,k})$ through Δ and

$$\mathbb{P}(M_i|\mathbf{z}) = \sum_{k=1}^{K} \mathbb{P}(M_i|\mathbf{S}_k,\mathbf{z})\,\mathbb{P}(\mathbf{S}_k)\,. \tag{3.14}$$

Furthermore, the posterior variance (3.9) now reduces to

$$\mathrm{Var}[\Delta|\mathbf{z}] = \sum_{i=1}^{I} \mathrm{Var}[\tilde{\Delta}_i]\mathbb{P}(M_i|\mathbf{z}) + \sum_{i=1}^{I} \left(E[\tilde{\Delta}_i] - \mathbb{E}[\Delta|\mathbf{z}]\right)^2 \mathbb{P}(M_i|\mathbf{z})\,. \tag{3.15}$$

The first of two terms in (3.15) is denoted as the *in-model variance*, and the second one is called the *between-model variance*. Note that (3.13)-(3.15) is not a traditional BMA, as the influence of the different $\mathbf{S}_k$ is still implicitly present. Of course, propagating scenario-weighted posteriors is likely to provide different results with respect to a full BMSA using the same scenario weights, if significant nonlinear interaction effects are present. However, the computational effort is reduced significantly, since now only I propagations are required to compute the prediction and its variance. Scenario probabilities $\mathbb{P}(\mathbf{S}_k)$

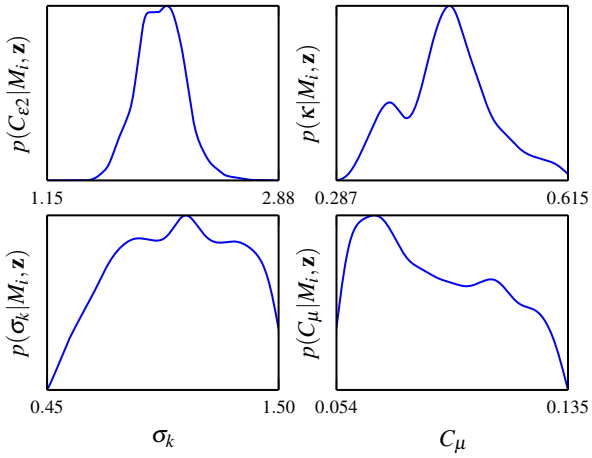


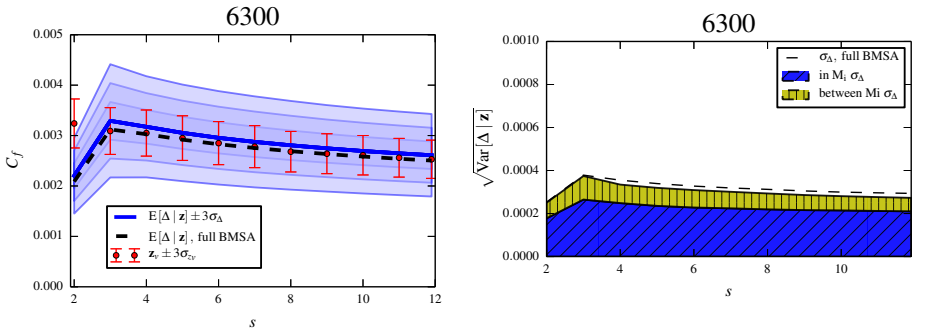
Figure 3.20: The marginals of $p(\tilde{\boldsymbol{\theta}}_i)$ in case of the $k-\varepsilon$ model.

now have to be specified *before* we propagate, and thus we lose the ability of using an automated sensor such as (3.10).

Instead, we must rely on expert opinion to specify the distribution of the $\mathbf{S}_k$. As proof of concept, let us imagine an expert RANS user, interested in predicting C_f for a favourable pressure gradient case. Furthermore suppose this user selects from the closure-coefficient database two favourable (1300, 2700), one zero (1400) and one mildly-adverse (1100) $\mathbf{S}_k$, and sets for each of these scenarios $\mathbb{P}(\mathbf{S}_k) = 0.25$. Next, for each model one $p(\tilde{\boldsymbol{\theta}}_i)$ is created via (3.12). As an example we show the marginals of $p(\tilde{\boldsymbol{\theta}}_i)$ for the $k-\varepsilon$ model in Figure 3.20. The prediction (3.13) and standard deviation of (3.15) are shown in Figure 3.21. To contrast, the full BMSA prediction and standard deviation are plotted as well. Although the full BMSA prediction lies closer to the validation data, the $\mathbb{E}[\Delta|\mathbf{z}]$ computed with (3.13) is not far off. This is especially true for the two σ_Δ, which almost lie on top of each other. Finally, when examining Figure 3.21b we see that through $\mathrm{Var}[\tilde{\Delta}_i]$, most of the between-scenario uncertainty has now been incorporated into the in-model standard deviation, which is significantly larger than the between-model σ_Δ.

3.5.6. DISCUSSION - CLOSURE COEFFICIENT DATABASE

The full BMSA approach described would be clearly extremely computationally expensive if applied to complex flow problems. We have deliberately made no simplifications – such as using MAP estimators, moment methods, or surrogate models – in order to assess the performance of the approach in the best-case scenario. Clearly, application to complex flow topologies requires some performance optimization. For instance, the fact that (3.10) usually picks $\mathbf{S}_k$ similar to $\mathbf{S}'$ suggests that we might set certain $\mathbb{P}(\mathbf{S}_k)$ to 0 by hand based upon physical reasoning, saving code runs. More specifically, this approach would deduct I from the total number of required propagations. Another option with a more significant reduction of computational effort was outlined in Section 3.5.5, i.e. the



(a) The prediction for case 6300.

(b) The BMSA std. dev. for case 6300.

Figure 3.21: Prediction for C_f obtained using 5 scenario-averaged posterior distributions. The full BMSA results are depicted by the dotted lines.

use of scenario-averaged posterior distributions. This particular method requires the user to *a priori* specify $\mathbb{P}(\mathbf{S}_k)$, which could also be done based upon physical reasoning. This $\mathbb{P}(\mathbf{S}_k)$ can then be used to create one $\mathbf{S}_k$-averaged posterior coefficient distribution per model, reducing the required number of propagations to I. Initial results look promising, showing only a small deviation from the full BMSA prediction. Such a strategy could allow $\mathscr{S}$ to contain many hundreds of scenarios, as it is the user who decides which of these to incorporate into $p(\tilde{\boldsymbol{\theta}}_i)$. In addition to different scenarios, we envision the closure coefficient database to contain posteriors calibrated for different QoIs as well. So far we have only used u^+ data to inform our posteriors. Our results indicate that we can still use these distributions to make robust predictions of C_f profiles, but some of the optimality is lost compared to the case where u^+ is the QoI. Enriching the database with distributions calibrated, amongst others, for C_f profiles could possibly remedy this. Also it would also provide a user with a more natural choice of which $\mathbf{S}_k$ to include, plus it could inform certain coefficient distributions that are just not informed by u^+ data. Posterior distributions that were calibrated using, e.g., both u^+ and C_f data could be included as well.

For many propagation techniques, e.g. [25], the computational cost increases with the number of uncertain coefficients. Therefore, further gains could come by reducing the dimensionality of the problem via a Sobol analysis. For instance our results of Section 3.3.2 show that for boundary-layer flows there are only 2 influential parameters for each model under consideration. If in other flow topologies a similar situation occurs, we might fix the non-influential parameters.

The hope is that this framework could be equally successful when applied to more general and complex classes of flows. There are two approaches here. As described earlier, since closure coefficient posteriors are topology independent, we can apply the "coefficient-database" from our limited class of flows to general flow problems. Then only uncertainty propagation needs to be performed for the complex problem to obtain predictions. Many techniques are available for efficiently propagating pdfs through a

computer code, notably sparse-grid stochastic-collocation methods [19, 20, 25]. In the worst-case, MAP estimators can be used instead of full pdfs. The success of this approach will depend on to what extent the features in the complex flow are represented in the database.

The second approach is to build a new database $\mathscr{S}$, tailored to the class of flows to be predicted. This would entail performing new calibrations. If possible, these calibrations should be done using cheap codes, and subsequently again use the topology-independent nature of $\boldsymbol{\theta}$. If this is not possible, Bayesian calibration could also be performed using a surrogate model as a replacement for the expensive computer code, see e.g. [15]. The scenarios in the new set should be (a) representative of this class, and (b) have rich experimental data available. This need not include only (or even mainly) flows of the same type – for wing aerodynamics, flat-plate boundary-layers are likely to be relevant. Models performing poorly – as judged by $\mathbb{P}(M_i|\mathbf{S}_k,\mathbf{z}_k)$ – could be culled from the set to reduce the effort needed in the coming predictions. The calibration step must only be performed once to set up the database. After which predictions are again a matter of uncertainty propagation only.

To summarize, we remain somewhat within the classical paradigm of RANS calibration, in that we calibrate on simple flow problems, the result of which we will apply to more complex problems. However, the classical paradigm: i) uses one point estimate of $\boldsymbol{\theta}$ and subsequently assumes generality of this estimate, taking no form of parameter error into account; ii) does not account for model-form error. To stochastically model these errors, we build a database of posterior distributions for several alternative turbulence models, which could be calibrated under a wide range of different scenarios and/or QoIs. If computationally allowable, a full BMSA could be performed to obtain predictions with quantified uncertainty, where the results coming from certain posteriors can be automatically favoured through a sensor like (3.10). If not, posteriors could be averaged over a weighted set of user-selected scenarios. This approach requires the construction of only one surrogate model for each turbulence model in $\mathscr{M}$, increasing its range of applicability compared to the full BMSA.

3.6. CONCLUSION

We performed Bayesian calibrations of the closure coefficients of 5 turbulence models: the Launder-Sharma $k-\varepsilon$, Wilcox $k-\omega$, Spalart-Allmaras and Baldwin-Lomax model and Wilcox stress-ω. Each model was calibrated for 14 scenarios consisting of flat-plate boundary-layers at varying pressure-gradients, using experimental velocity measurements. Substantial variation in closure coefficients for all models was observed across this large range of simple flow cases.

In order to synthesize these results, and make predictions for unmeasured flows, we utilized Bayesian Model Averaging, with an emphasis on scenarios, i.e. Bayesian Scenario Averaging. The framework requires the computation of the posterior model probabilities, which can be thought of as a measure of consistency that a model has with the experimental data. We found that, like the closure coefficients, the posterior model probabilities vary greatly with the applied pressure-gradient scenario. These results suggest that there is no single best choice of turbulence model or closure coefficients, and no obvious way to choose an appropriate model *a priori*.

Instead we used BMSA to make stochastic predictions of unmeasured velocity profiles. Closing BMSA with uniform scenario weighting produced predictions which matched reference data, but with excessively large variance. We therefore developed a smart scenario sensor, to automatically preferentially weight those scenarios in the calibration set that are similar to the prediction case. This resulted in substantially improved predictors, both in terms of mean and variance. For almost all of 14 prediction cases, the predictions lie within one standard deviation of the experimental validation data, and the variance was of the same order as the experimental measurement error.

Despite the fact that only experimental u^+ data was used to inform our posterior distributions, we also used their topology independent nature to make predictions for skin-friction profiles. Although these posterior distributions were not optimal for this particular quantity of interest, we were still able to make consistently more robust predictions than can be expected from a single turbulence model.

Finally, an approach was suggested to significantly reduce the computational effort required by the predictive phase of our methodology. By averaging the posterior distributions over different scenarios, the number of pdf propagations drops from $I \times K$ to just I. Here, I is the number of turbulence models and K the number of scenarios. Even though expert opinion is now needed to specify the scenario weighting, initial results look promising.

We see the methods developed here as a step towards providing estimates of the error due to turbulence closure models, and thereby lending a robust predictive power to RANS. The next step is to apply them to more complex flow configurations. In this case it will be computationally intractable to sample directly from the computer code, and hence the use of surrogate models becomes necessary. The next section describes one such an efficient surrogate modelling technique.

Acknowledgements: The present work was supported by the French 'Agence Nationale de la Recherche' (ANR) under contract ANR-11-MONU-008-002.

REFERENCES

[1] B. S. Baldwin and H. Lomax. *Thin layer approximation and algebraic model for separated turbulent flows*, volume 257. American Institute of Aeronautics and Astronautics, 1978.

[2] S.H. Cheung, T.A. Oliver, E.E. Prudencio, S. Prudhomme, and R.D. Moser. Bayesian uncertainty analysis with applications to turbulence modeling. *Reliability engineering and systems safety*, 96(9):1137–1149, 2011.

[3] F.H. Clauser. Turbulent boundary layers in adverse pressure gradients. *J. Aeron. l Sci.*, pages 91–108, 1954.

[4] D.E. Coles and E.A. Hirst. Computation of turbulent boundary layers. In *Proceedings of AFOSR-IFP Stanford Conference*, volume 2, 1968.

[5] D. Draper. Assessment and propagation of model uncertainty. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 45–97, 1995.

[6] R. Dwight and Z.H. Han. Efficient uncertainty quantification using gradient-enhanced Kriging. In *Proceedings of 11th AIAA Conference on Non-Deterministic Approaches, Palm Springs CA. AIAA-2009-2276*, pages 1–21. AIAA, 2009.

[7] WN Edeling, Pasquale Cinnella, Richard P Dwight, and Hester Bijl. Bayesian estimates of parameter variability in the< i> k</i>–< i> ε</i> turbulence model. *Journal of Computational Physics*, 258:73–94, 2014.

[8] M. Emory, J. Larsson, and G. Iaccarino. Modeling of structural uncertainties in reynolds-averaged navier-stokes closures. *Physics of Fluids (1994-present)*, 25(11):–, 2013.

[9] A. Gelman, J.B. Carlin, H.S. Stern, and D.B. Rubin. Bayesian data analysis. 2004. *Chapman&Hall, London*, 2, 2004.

[10] W.K. Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[11] J.A. Hoeting, D. Madigan, A.E. Raftery, and C.T. Volinsky. Bayesian model averaging: a tutorial. *Statistical science*, pages 382–401, 1999.

[12] M.C. Kennedy and A. O'Hagan. Bayesian calibration of computer models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(3):425–464, 2001.

[13] SJ Kline. The 1980-81 afosr-httm-stanford conference on complex turbulent flows: Comparison of computation and experiment. In *Objectives, evaluation of data, specifications of test cases, discussion and position papers*, volume 1, 1981.

[14] B.E. Launder and B.I. Sharma. Application of the energy-dissipation model of turbulence to the calculation of flow near a spinning disc. *Letters Heat Mass Transfer*, 1:131–137, 1974.

[15] Youssef Marzouk and Dongbin Xiu. A stochastic collocation approach to bayesian inference in inverse problems. *Communications in Computational Physics*, 6:826–847, 2009.

[16] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E. Teller, et al. Equation of state calculations by fast computing machines. *Journal of Chemical Physics*, 21(6):1087, 1953.

[17] P.D. Meyer, M. Ye, M.L. Rockhold, S.P. Neuman, and K.J. Cantrell. Combined estimation of hydrogeologic conceptual model, parameter, and scenario uncertainty with application to uranium transport at the hanford site 300 area. Technical report, Pacific Northwest National Laboratory (PNNL), Richland, WA (US), 2007.

[18] A. Mira, R. Solgi, and D. Imparato. Zero variance markov chain monte carlo for bayesian estimators. *Statistics and Computing*, 23(5):653–662, 2013.

[19] F. Nobile, R. Tempone, and C.G. Webster. A sparse grid stochastic collocation method for partial differential equations with random input data. *SIAM Journal on Numerical Analysis*, 46(5):2309–2345, 2008.

[20] R. Pecnik, J.A.S. Witteveen, and G. Iaccarino. Uncertainty quantification for laminar-turbulent transition prediction in rans turbomachinery applications. *AIAA Paper 2011*, pages 1–14, 2011.

[21] R. Rojas, Sa. Kahunde, L. Peeters, O. Batelaan, L. Feyen, and A. Dassargues. Application of a multimodel approach to account for conceptual model and scenario uncertainties in groundwater modelling. *Journal of Hydrology*, 394(3):416–435, 2010.

[22] François G Schmitt. About boussinesq's turbulent viscosity hypothesis: historical remarks and a direct evaluation of its validity. *Comptes Rendus Mécanique*, 335(9):617–627, 2007.

[23] P.R. Spalart and S.R. Allmaras. A one-equation turbulence model for aerodynamic flows. *La recherche aérospatiale*, 1(1):5–21, 1994.

[24] D.C. Wilcox, American Institute of Aeronautics, and Astronautics. *Turbulence modeling for CFD*, volume 3. DCW industries La Canada, CA, 2006.

[25] J.A.S. Witteveen and G. Iaccarino. Simplex stochastic collocation with eno-type stencil selection for robust uncertainty quantification. *Journal of Computational Physics*, 239:1–21, 2013.

[26] Z. Zhang, W. Zhang, Z.J. Zhai, and Q.Y. Chen. Evaluation of various turbulence models in predicting airflow and turbulence in enclosed environments by cfd: Part 2—comparison with experimental data from literature. *Hvac&R Research*, 13(6):871–886, 2007.

4

IMPROVED SIMPLEX-STOCHASTIC COLLOCATION METHOD APPLIED TO COMPLEX SIMULATION CODES.

4.1. INTRODUCTION

In order to make reliable predictions of a physical system using a computer code it is necessary to understand what effect the uncertainties in the inputs have on the output Quantity of Interest (QoI). Attempts to do so while keeping the computational cost low can be found in [12, 20, 27], which rely on Smolyak-type sparse-grid stochastic-collocation methods. Whereas traditional collocation-type methods [4, 16] use full-tensor product formulas to extend a set of one-dimensional nodes to higher dimensions, sparse-grid methods build a sparse interpolant using a constrained linear combination of one-dimensional nodes. This can provide a grid with a potential a reduction in support nodes of several orders of magnitude.

Although computationally more efficient than full-tensor approaches, sparse-grid methods add points equally in all dimensions, irrespective of whether the response surface is locally smooth or discontinuous. Therefore further gains can be achieved through adaptive stochastic-collocation schemes which have been developed in recent years. For instance Ma et al. [17] proposed an Adaptive Sparse-Grid (ASG) collocation method where the probabilistic space is spanned by linear finite-element basis functions. During each iteration the probability space is refined locally through an error measure based upon the hierarchical surplus, defined as the difference between the interpolation of the previous level and the newly added code sample. Although the space is refined locally, unphysical oscillations can still occur due to the lack of sample points on some of the edges of the local support of the basis functions. The Simplex-Stochastic Collocation (SSC) Method of Witteveen and Iaccarino [35] circumvents this problem by discretizing

This chapter is based on: W.N. Edeling, P. Cinnella, R.P. Dwight, Improved Simplex-Stochastic Collocation Method Applied To Complex Simulation Codes, Submitted to Journal of Computational Phyics, 2014

the domain into simplices by means of a Delaunay triangulation, and enforcing the so-called Local-Extremum Conserving limiter to suppress unphysical oscillations. Furthermore, it computes Essentially Non-Oscillatory (ENO) stencils [36] of the sample points which allow for high-order polynomial interpolation. Further features include randomized sampling, the ability to deal with non-hypercube probability spaces and it can be extended to perform interpolation with subcell resolution [37].

Besides schemes which efficiently sample the probabilistic space, there are other means of dealing with the curse of dimensionality, i.e. the exponential increase in the amount of computational cost with increasing dimension. Physical systems often have a low effective dimension, meaning that only a few coefficients are influential, and only low-order correlations between the input coefficients have a significant impact on the output. To capitalize on this behavior, High-Dimensional Model-Reduction (HDMR) techniques can be applied [25]. Here, an n-dimensional function is exactly represented as a hierarchical sum of 2^n component functions of increasing dimensionality. In the case of low effective dimension, the n-dimensional function can be approximated well by a truncated expansion. The main idea is the solve several low-dimensional subproblems instead of one high-dimensional one, which greatly reduces the computational cost. A well-known member of this class of decompositions is the analysis of variance (ANOVA) decomposition. In [10], Foo et al. successfully coupled their Multi-Element Probabilistic Collocation Method [11] with a HDMR-ANOVA decomposition to problems with up to 600 dimensions. Although they achieved a significant reduction in the computational cost compared to approaches with full-tensor products, the number of points required to sufficiently sample these extremely high-dimensional spaces is still of the order of millions or even more. Furthermore, in [18] Ma et al. coupled their previously mentioned ASG method with the so-called cut-HDMR technique of [25]. This approach is computationally more efficient than ANOVA-HDMR, as it does not require the evaluation of multi-dimensional integrals. Besides truncating the cut-HDMR expansion at a certain order, the authors of [18] also made their approach dimension adaptive through weights which identify the dimensions that contribute the most to the mean of the stochastic problem. They applied this approach to several easily-evaluated test problems of very high dimensionality, i.e. up to a stochastic dimension of 500. Again, their results represent a significant reduction in the required number of code samples for a certain error level compared to full tensor grids, but in absolute terms the number of samples is still very high.

Even though some our numerical examples in this article are quickly evaluated, we will operate under the assumption that all of them come from expensive computer codes. In this setting it will be intractable to sufficiently sample a probabilistic space of dimension $\mathcal{O}(100)$. Therefore we will investigate means to efficiently create surrogate models of complex simulation codes with a moderate number of uncertain input parameters. Of course, the term 'moderate dimensionality' is system dependent. So to clarify, we consider the dimensionality moderate when the number of inputs parameters falls within the range of 5 to 10. Many problems of engineering interest are simulated using codes with similar dimensionality, e.g. turbulence models [9, 13], groundwater models [26] or thermodynamic equations-of-state [6, 19].

Due to its adaptivity, high-polynomial order and Runge-phenomenon free interpola-

tion, the focus of our investigation is the SSC method. The SSC method has been applied in [3] to optimization under uncertainty of a Formula 1 tire brake intake. After a one-dimensional perturbation analysis, 3 variables were selected for analysis. Furthermore, in [7, 8] the SSC method is efficiently coupled with Downhill-Simplex optimization in a setting for robust design optimization. Several example problems are considered, but again the maximum number of uncertain variables is 3. In [35] the authors note that the cost of the Delaunay triangulation becomes prohibitively large from a dimensionality of 5 onward. They suggested to replace the Delaunay triangulation with a scheme where simplices are formed by selecting the nearest points from randomly placed Monte Carlo (MC) samples. Using this approach the SSC method was applied to a continuous QoI with 15 uncertain parameters.

This chapter is aimed at reducing the computational burden of the SSC method, while keeping the Delaunay triangulation. We investigate two separate techniques. First we propose the use of new alternative stencils based upon the set-covering problem [14]. The main idea is to use the fast increase in number of simplex elements with polynomial order to create a minimal set of stencils which covers the entire probabilistic domain. Afterwards, only this set is used for interpolation. This allows for a more efficient implementation of the SSC method. Our results show that these stencils are capable of reducing the computational cost. We furthermore present a new method for avoiding problems will the ill-conditioning of the sample matrix, and the provide a new formula for placing uniformly distributed samples in any simplex of arbitrary dimension. Secondly, inspired by the work of Ma. et al. [18], we integrate the SSC method into the cut-HDMR framework. This has all the same advantages, e.g. dimension adaptivity, but not the disadvantages related to the ASG method such as linear interpolation and the possible occurrence of the Runge phenomenon. Unlike the authors of [18], we apply our method a complex computer code for which obtaining samples is expensive. For both proposed techniques we perform a detailed analysis of the error and we give a discussion on computational cost as a function of the number of input parameters.

This chapter is organised as follows. In Section 4.2 we present the baseline SSC method as developed by Witteveen and Iaccarino. Next, in Sections 4.3 and 4.4, we described the Set-Covering stencils and cut-HDMR respectively. Section 2.5 holds the obtained results and the discussion. Finally, we give our conclusions in Section 4.6.

4.2. SIMPLEX-STOCHASTIC COLLOCATION METHOD

In the next section we give a general outline of the Simplex-Stochastic Collocation (SSC) Method as developed by Witteveen et. al. For a more detailed description we refer to [34–36, 38].

4.2.1. GENERAL OUTLINE BASELINE SSC METHOD

The SSC method was introduced as a non-intrusive method intended for robust and efficient propagation of uncertainty through computer models. It differs from traditional collocation methods, e.g. [4, 16], in two main ways. First, for multi-dimensional problems it employs the Delaunay triangulation to discretize the probability space into simplex elements, rather than relying on the more common tensor product of one-dimensional

abscissas [22]. Using a multi-element technique has the advantage that mesh adaptation can be performed, such that only regions of interest are refined. Secondly, the SSC method is capable of handling non-hypercube probability spaces [35].

The response surface of the QoI $u(\boldsymbol{\xi})$ is denoted by $w(\boldsymbol{\xi})$ and it is constructed using a set of n_s samples from the computational model, $\mathbf{v} = \{v_1, \cdots, v_{n_s}\}$. Here, $\boldsymbol{\xi}$ is a vector of n_ξ random input parameters $\boldsymbol{\xi}(\boldsymbol{\omega}) = (\xi_1(\omega_1), \cdots, \xi_{n_\xi}(\omega_{n_\xi})) \in \Xi \subset \mathbb{R}^{n_\xi}$. Furthermore, we define Ξ to be the parameter space and $\boldsymbol{\omega} = \left(\omega_1, \cdots, \omega_{n_\xi}\right) \in \Omega \subset \mathbb{R}^{n_\xi}$ is a vector containing realizations of the probability space $(\Omega, \mathscr{F}, P)$ with $\mathscr{F}$ the σ-algebra of events and P a probability measure. The variables in $\boldsymbol{\omega}$ are distributed uniformly as $\mathscr{U}(0,1)$, and the input parameters can have any distribution $f_{\boldsymbol{\xi}}$, although for the sake of simplicity we restrict ourselves in this paper to $f_{\boldsymbol{\xi}} = \mathscr{U}(\boldsymbol{\xi}_i^a, \boldsymbol{\xi}_i^b)$, with the bounds $\boldsymbol{\xi}_i^a$ and $\boldsymbol{\xi}_i^b$. We perform all our analysis on the hypercube $K_{n_\xi} := [0,1]^{n_\xi}$, and we use a linear map in order to go from K_{n_ξ} to the parameter domain $\boldsymbol{\xi}$. Our goal is to propagate $f_{\boldsymbol{\xi}}$ through the computational model in order to asses the effect of $f_{\boldsymbol{\xi}}$ on the m-th statistical moment of $u(\boldsymbol{\xi}(\boldsymbol{\omega}))$, i.e. we wish to compute

$$\mu_u^{(m)} = \int_\Xi u(\boldsymbol{\xi})^m f_{\boldsymbol{\xi}}(\boldsymbol{\xi}) \mathrm{d}\boldsymbol{\xi}. \tag{4.1}$$

Note that u can also be a function of a physical coordinate $\mathbf{x}$ or other deterministic explanatory variables, but for brevity we omit $\mathbf{x}$ from the notation. Since the SSC method discretizes the parameter space Ξ into n_e disjoint simplices $\Xi = \Xi_1 \cup \cdots \cup \Xi_{n_e}$, the m^{th} statistical moment is approximated as

$$\mu_u^{(m)} = \int_\Xi u(\boldsymbol{\xi})^m f_{\boldsymbol{\xi}}(\boldsymbol{\xi}) \mathrm{d}\boldsymbol{\xi} \approx \mu_w^{(m)} = \sum_{j=1}^{n_e} \int_{\Xi_j} w_j(\boldsymbol{\xi})^m f_{\boldsymbol{\xi}}(\boldsymbol{\xi}) \, \mathrm{d}\boldsymbol{\xi}. \tag{4.2}$$

Here, w_j is a local polynomial function of order p_j associated with the j-th simplex Ξ_j such that

$$w(\boldsymbol{\xi}) = w_j(\boldsymbol{\xi}), \quad \text{for } \boldsymbol{\xi} \in \Xi_j, \tag{4.3}$$

and

$$w_j(\boldsymbol{\xi}_{k_{j,l}}) = v_{k_{j,l}}. \tag{4.4}$$

Note that (4.4) is simply the interpolation condition. The subscript $k_{j,l} \in \{1, \cdots, n_s\}$ is a global index which refers to the k-th added computational sample, while j refers to a certain simplex element. Furthermore, $l = 0, \cdots, N_j$ is a local index used to count the number of samples from $\mathbf{v}$ involved in the construction of w_j. The $N_j + 1$ number of points needed for n_ξ-dimensional interpolation of order p_j is given by

$$N_j + 1 = \frac{(n_\xi + p_j)!}{n_\xi! p_j!}, \tag{4.5}$$

and the local interpolation function w_j itself is given by the expansion

$$w_j(\boldsymbol{\xi}) = \sum_{l=0}^{N_j} c_{j,l} \boldsymbol{\Psi}_{j,l}(\boldsymbol{\xi}). \tag{4.6}$$

The choice of basis polynomials $\mathbf{\Psi}_{j,l}$, and the determination of the interpolation coefficients $c_{j,l}$ is dealt with in Section 4.2.2. Note that for a given n_ξ, the maximum allowable order p_j based on the number of samples n_s can be inferred from (4.5).

Which $N_j + 1$ points are used in (4.6) is determined by the interpolation stencil S_j. The selection can be made based on the nearest-neighbour principle [35]. In this case the first $n_\xi + 1$ points are the vertices $\{\boldsymbol{\xi}_{k_{j,0}}, \cdots, \boldsymbol{\xi}_{k_{j,n_\xi}}\}$ of the simplex Ξ_j, which would suffice for $p_j = 1$. For higher degree interpolation, neighbouring points $\boldsymbol{\xi}_k$ are added based on their proximity to the center of simplex Ξ_j, i.e. based on their ranking according to

$$\|\boldsymbol{\xi}_k - \boldsymbol{\xi}_{center_j}\|_2, \tag{4.7}$$

where those $\boldsymbol{\xi}_k$ of the current simplex Ξ_j are excluded. The simplex center $\boldsymbol{\xi}_{center_j}$ is defined as

$$\boldsymbol{\xi}_{center_j} = \frac{1}{n_\xi + 1} \sum_{l=0}^{n_\xi} \boldsymbol{\xi}_{k_{j,l}}. \tag{4.8}$$

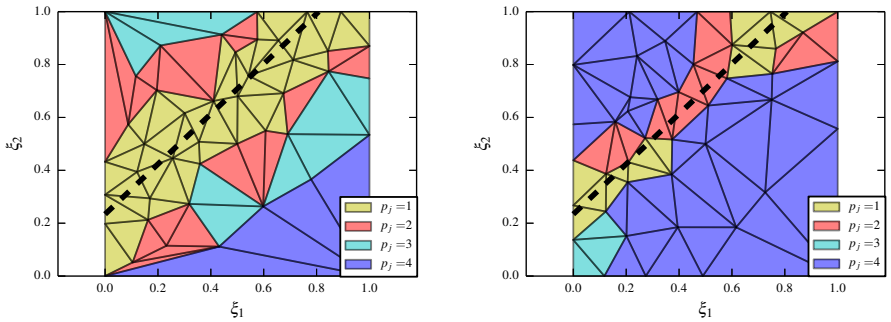
The nearest neighbor stencil (4.7) leads to a p_j distribution that can increase quite slowly when moving away from a discontinuity. An example of this behavior can be found in Figure 4.1(a), which shows the Delaunay triangulation with a discontinuity running through the domain. An alternative to nearest-neighbor stencils are stencils created according to the Essentially Non-Oscillatory (ENO) principle [36]. The idea behind ENO stencils is to have higher degree interpolation stencils up to a thin layer of simplices containing the discontinuity. For a given simplex Ξ_j, its ENO stencil is created by locating all the nearest-neighbor stencils that contain Ξ_j, and subsequently selecting the one with the highest p_j. This leads to a Delaunay triangulation which captures the discontinuity better than its nearest-neighbor counterpart. An example can be found in Figure 4.1(b). Unless otherwise stated, for all subsequent baseline SSC surrogate models we will use ENO-type stencils.

The initial samples, at least in the case of hypercube probability spaces, are located at the 2^{n_ξ} corners of the hypercube K_{n_ξ}. Furthermore, one sample is placed in the middle of the hypercube. Next, the initial grid is adaptively refined based on an appropriate error measure. This error measure can either be based on the hierarchical surplus between the response surface of the previous iteration and new a sample v_k, or on the geometrical properties of the simplices. The latter option is more reliable in multiple stochastic dimensions as it is not based on the hierarchical surplus in a single discrete point [34]. The geometrical refinement measure is given by

$$\bar{e}_j = \bar{\Omega}_j \bar{\Xi}_j^{2O_j}. \tag{4.9}$$

It contains the probability and the volume of simplex Ξ_j, i.e.

$$\bar{\Omega}_j = \int_{\Xi_j} f_{\boldsymbol{\xi}}(\boldsymbol{\xi}) \mathrm{d}\boldsymbol{\xi} \quad \text{and} \quad \bar{\Xi}_j = \int_{\Xi_j} \mathrm{d}\boldsymbol{\xi}, \tag{4.10}$$



(a) Delaunay triangulation color coded by p_j with nearest neighbor stencils.

(b) Delaunay triangulation color coded by p_j with ENO stencils.

Figure 4.1: Delaunay triangulation for two stencil types with a discontinuity running along the dotted line.

where $\bar{\Xi} = \sum_{j=1}^{n_e} \bar{\Xi}_j$. The probability $\bar{\Omega}_j$ can be computed by Monte-Carlo sampling and $\bar{\Xi}_j$ via the relation

$$\bar{\Xi}_j = \frac{1}{n_\xi!}|\det(D)|, \quad D = \left[\boldsymbol{\xi}_{k_{j,1}} - \boldsymbol{\xi}_{k_{j,0}} \quad \boldsymbol{\xi}_{k_{j,2}} - \boldsymbol{\xi}_{k_{j,0}} \quad \cdots \quad \boldsymbol{\xi}_{k_{j,n_\xi+1}} - \boldsymbol{\xi}_{k_{j,0}}\right] \in \mathbb{R}^{n_\xi \times n_\xi}. \tag{4.11}$$

Finally, the order of convergence O_j is given by [34]

$$O_j = \frac{p_j + 1}{n_\xi}. \tag{4.12}$$

The simplex with the highest $\bar{e}_j$ is selected for refinement. To ensure a good spread of the sample points, only randomly-selected points inside a simplex sub-element Ξ_{sub_j} are allowed. The vertices of this sub-element are computed by

$$\boldsymbol{\xi}_{sub_{j,l}} = \frac{1}{n_\xi} \sum_{\substack{l^*=0 \\ l^* \neq l}}^{n_\xi} \boldsymbol{\xi}_{k_{j,l^*}}, \tag{4.13}$$

see Figure 4.2 for a two-dimensional example. In order to place random samples in an arbitrary simplex we derive an analytical map $M_{n_\xi} : K_{n_\xi} \to \Xi_j$, see Section 4.2.2. The SSC algorithm can be parallelized by selecting the $N < n_e$ simplices with the N largest $\bar{e}_j$ for refinement. By using (4.13) only the simplex interiors will get refined (see again Figure 4.2), and the boundaries of the hypercube will never be sampled outside the initial 2^{n_ξ} points. To avoid this, we do not use (4.13) if a simplex element located at the boundary is selected for refinement. Instead, we randomly place samples at the longest simplex edge which is at the boundary, ±10% from the edge center.

Note that (4.9) is probabilistically weighted through Ω_j and that it assigns high $\bar{e}_j$ to those simplices with low p_j since in general $\bar{\Xi}_j \ll 1$. Effectively this means that (4.9) is a solution-dependent refinement measure which refines simplices near discontinuities

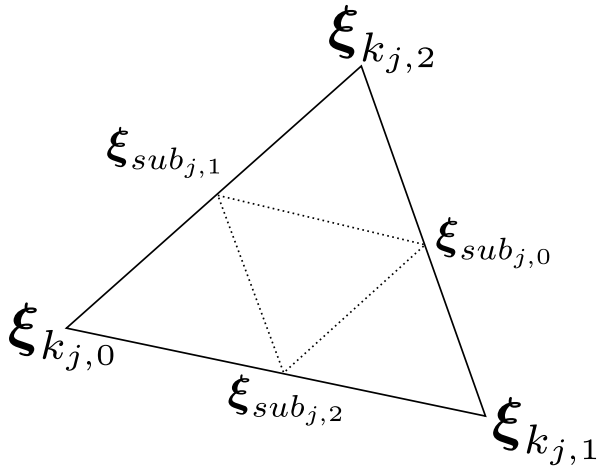


Figure 4.2: The sub simplex (dotted line) of a two-dimensional simplex. Upon refinement one sample is placed at a randomly selected location inside the sub simplex in order to avoid clustering of points.

since the SSC method automatically reduces the polynomial order if a stencil S_j crosses a discontinuity. It achieves this by enforcing the so-called Local Extremum Conserving (LEC) limiter to all simplices Ξ_j in all S_j. The LEC condition is given by

$$\min_{\boldsymbol{\xi}\in\Xi_j} w_j(\boldsymbol{\xi}) = \min \mathbf{v}_j \wedge \max_{\boldsymbol{\xi}\in\Xi_j} w_j(\boldsymbol{\xi}) = \max \mathbf{v}_j, \tag{4.14}$$

where $\mathbf{v}_j = \{v_{k_{j,0}}, \cdots, v_{k_{j,n_\xi}}\}$ are the samples at the vertices of Ξ_j. If w_j violates (4.14) in one of its $\Xi_j \in S_j$, the polynomial order p_j of that stencil is reduced, usually by 1. Since polynomial overshoots occur when trying to interpolate a discontinuity, p_j is reduced the most in discontinuous regions. Interpolating a function on a simplex with $p_j = 1$ and $\mathbf{v}_j$ located at its vertices always satisfies (4.14) [34]. This ensures that $w(\boldsymbol{\xi})$ is able to represent discontinuities without suffering from the Runge phenomenon. In practice, (4.14) is enforced for all Ξ_j in all S_j via random sampling of the w_j. If for a given w_j (4.14) is violated for any of the randomly placed samples $\boldsymbol{\xi}_j$, the polynomial order of the corresponding stencil reduced. Again, wow we sample the n_ξ-dimensional simplices is described in Section 4.2.2. The computational cost of enforcing (4.14) is investigated in Section 4.5.

The procedure of enforcing the LEC condition, computing a refinement measure and subsequently refining certain selected simplices is either repeated for a maximum of I iterations, or halted when a sufficient level of accuracy is obtained. This level of accuracy can be estimated through an error measure based upon the hierarchical surplus [34]. As mentioned, this is the difference between the response surface w_j and the newly added code sample $v_{k_{j,ref}}$ at the refinement location $\boldsymbol{\xi}_{k_{j,ref}}$, i.e.

$$\epsilon(\boldsymbol{\xi}_{k_{j,ref}}) = w_j(\boldsymbol{\xi}_{k_{j,ref}}) - v_{k_{j,ref}} \tag{4.15}$$

This is a point estimate of the error, located at what will be a vertex in the new refined Delaunay grid. To assign error estimates to the simplices rather than to vertices, the error $\tilde{\epsilon}_j$ is introduced. For each Ξ_j, $\tilde{\epsilon}_j$ is simply the absolute value of (4.15) of its most recently added vertex $\boldsymbol{\xi}_{k^*}$. Since adding vertices will change the Delaunay discretization we relate the error of the previous simplex to the new one via

$$\hat{\epsilon}_j \approx \tilde{\epsilon}_j \left(\frac{\bar{\Xi}_j}{\bar{\Xi}_{k^*,ref}} \right)^{\mathscr{O}_j} \tag{4.16}$$

[35]. The ratio $\bar{\Xi}_i/\bar{\Xi}_{k^*,ref}$ represents the change in volume from its old size $\bar{\Xi}_{k^*,ref}$, i.e. the volume of the simplex which was refined by $\boldsymbol{\xi}_{k*}$, to its new size $\bar{\Xi}_j$. Finally, each individual $\hat{\epsilon}_j$ is combined in a global error estimate via the following root mean square (RMS) error norm

$$\hat{\epsilon}_{rms} = \sqrt{\sum_{j=1}^{n_e} \Omega_j \hat{\epsilon}_j^2}. \tag{4.17}$$

The entire baseline SSC method is given in pseudo code in B.

4.2.2. IMPROVEMENTS ON THE BASELINE SSC METHOD

Before discussing our new stencil technique in Section 4.3.1, we will introduce some additions to the baseline SSC method not discussed in the original references [34–36, 38].

POISED SAMPLE SEQUENCE

The authors of [32] write (4.6) in matrix form, constraining $\boldsymbol{\Psi}_{j,l}$ to the class of monomials, and subsequently solve explicitly for the coefficients $c_{j,l}$. They note that although they had no difficulties in solving this system, the matrix could have a high condition number. This poses no real problem for $n_\xi \leq 3$, but for higher dimensions it can become problematic. To cope with this we impose an additional condition on the construction of the stencils S_j such that the interpolation problem is *poised*, meaning that the sample matrix is non-singular [21]. In the following discussion we drop the subscript j until further notice to make the notation more concise.

To construct the interpolating monomials, let us define the collection consisting of $N+1$ n_ξ-dimensional multi-indices $\underline{i} := (i_1, \cdots, i_d, \cdots, i_{n_\xi})$, where for all $\underline{i}$ we have $|\underline{i}| := i_1 + \cdots + i_{n_\xi} \leq p_j$ and each i_d is an integer between 0 and n_ξ. Furthermore, for a given vertex $\boldsymbol{\xi}_l = (\xi_{1,l}, \cdots, \xi_{n_\xi,l})$ belonging to stencil S, let us define its $\underline{i}$-th power to be $\boldsymbol{\xi}_l^{\odot \underline{i}} := \xi_{1,l}^{i_1} \times \cdots \times \xi_{n_\xi,l}^{i_{n_\xi}}$. The sample matrix Ψ, a multi-dimensional Vandermonde matrix, can then be written as

$$\Psi = \begin{bmatrix} \boldsymbol{\xi}_0^{\odot \underline{0}} & \boldsymbol{\xi}_0^{\odot \underline{1}} & \cdots & \boldsymbol{\xi}_0^{\odot \underline{N}} \\ \boldsymbol{\xi}_1^{\odot \underline{0}} & \boldsymbol{\xi}_1^{\odot \underline{1}} & \cdots & \boldsymbol{\xi}_1^{\odot \underline{N}} \\ \vdots & & \vdots & \vdots \\ \boldsymbol{\xi}_N^{\odot \underline{0}} & \boldsymbol{\xi}_N^{\odot \underline{1}} & \cdots & \boldsymbol{\xi}_N^{\odot \underline{N}} \end{bmatrix} \in \mathbb{R}^{(N+1)\times(N+1)}. \tag{4.18}$$

As an example, the l-th row of (4.18) in lexicographical order for $p_j = 2$ will look like $\begin{bmatrix} 1 & \xi_{1,l} & \xi_{2,l} & \xi_{1,l}^2 & \xi_{1,l}\xi_{2,l} & \xi_{2,l}^2 \end{bmatrix}$. The coefficients c_l in (4.6) can now be obtained by solving the system

$$\begin{bmatrix} \boldsymbol{\xi}_0^{\odot 0} & \boldsymbol{\xi}_0^{\odot 1} & \cdots & \boldsymbol{\xi}_0^{\odot N} \\ \boldsymbol{\xi}_1^{\odot 0} & \boldsymbol{\xi}_1^{\odot 1} & \cdots & \boldsymbol{\xi}_1^{\odot N} \\ \vdots & & \vdots & \vdots \\ \boldsymbol{\xi}_N^{\odot 0} & \boldsymbol{\xi}_N^{\odot 1} & \cdots & \boldsymbol{\xi}_N^{\odot N} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_N \end{bmatrix} = \begin{bmatrix} v_0 \\ v_1 \\ \vdots \\ v_N \end{bmatrix}, \tag{4.19}$$

where $\{v_0, \cdots, v_N\}$ are the code samples belonging to stencil S. Once the c_l are known, we can interpolate to any point $\boldsymbol{\xi}$ in the domain spanned by S.

We define $\Delta \equiv \det(\Psi)$, and note that the whole approach hinges on the well-poisedness condition $\Delta \neq 0$. This condition is relatively easy violated during the SSC procedure in higher dimensions. For instance, if for $n_\xi = 4$ we determine the maximum allowable p using (4.5) on the initial Delaunay grid we obtain $p_{max} = 2$. However, every stencil in this case will have $\Delta = 0$. Also situations where a stencil has too many vertices located in the same direction (e.g. due to edge refinement at the boundary of K_{n_ξ}), can lead to a zero determinant of (4.18). Thus, for $n_\xi > 1$ the poisedness condition $\Delta \neq 0$ imposes constraints on the geometrical distribution of the $\boldsymbol{\xi}_l$. From [5, 21] we know

Theorem 1. *The $N+1$ vertices $\boldsymbol{\xi}_0, \cdots, \boldsymbol{\xi}_N \in \mathbb{R}^{n_\xi}$ are polynomially poised iff they are not a subset of any algebraic hypersurface of degree $\leq p$.*

An algebraic hypersurface in $\mathbb{R}^{n_\xi}$ is a $n_\xi - 1$ dimensional surface embedded in a n_ξ-dimensional space constrained to satisfy an equation $f(\xi_1, \cdots, \xi_{n_\xi}) = 0$. The degree is given by f.

The authors of [5] devised a Geometric Characterization (GC) condition which allows us to detect if a set of vertices is poised, i.e.

Definition 1. *GC condition: For each $\boldsymbol{\xi}_l$ in a set of $N+1$ vertices in $\mathbb{R}^{n_\xi}$, there exists p distinct hyperplanes $G_{1,l}, \cdots, G_{p,l}$ such that i) $\boldsymbol{\xi}_l$ does not lie on any of these planes, and ii) all other $\boldsymbol{\xi}_k$, $k = \{0, \cdots, N\} \backslash \{l\}$ lie on at least one of these hyperplanes. Mathematically speaking i) and ii) amount to*

$$\boldsymbol{\xi}_i \in \bigcup_{k=0}^{p} G_{k,l} \quad \text{if } i \neq l, \quad \forall i = 1, 2, \cdots, N. \tag{4.20}$$

Theorem 2. *Let $\{\boldsymbol{\xi}_l\}$ be a set of $N+1$ vertices in $\mathbb{R}^{n_\xi}$. If $\{\boldsymbol{\xi}_l\}$ satisfies the GC condition, then $\{\boldsymbol{\xi}_l\}$ admits a unique interpolation of degree $\leq p$ [5].*

Due to its geometrical configuration, a single simplex Ξ_j in $\mathbb{R}^{n_\xi}$ always satisfies the GC condition for $p = 1$, see Figure 4.3 for a three-dimensional example. For a given vertex $\boldsymbol{\xi}_l \in \Xi_j$, we always have one hyperplane containing the face of the simplex made up by all vertices except $\boldsymbol{\xi}_l$. Thus, Theorem 2 implies that simplex Ξ_j will lead to a Ψ with $\Delta \neq 0$ and $p_j = 1$.

We use this result to obtain a set of well-poised ENO stencils $S_j \ \forall \ j = 1, \cdots, n_e$, in a way that is similar to the construction of the ENO-stencils as described in [36]. Only if

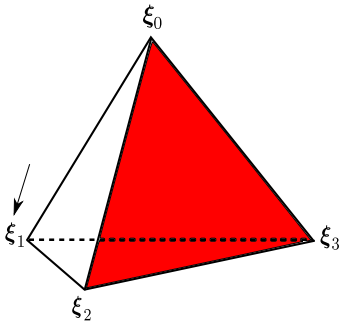


Figure 4.3: When selecting node $\boldsymbol{\xi}_1$, there exists one ($p = 1$) plane which contains all other points except $\boldsymbol{\xi}_1$. This is true for all nodes in the simplex.

during the enforcement of the LEC condition (4.14) we encounter a stencil S_j for which $\Delta = 0$, we collect a set of k candidate nearest-neighbor stencils $\{S_{j,i}\}_{i=1}^{k}$ which all contain simplex Ξ_j. We then select the S_j which has the highest p_j and $\Delta \neq 0$. In the worst case scenario we get $p_j = 1$, where S_j contains only the vertices of Ξ_j itself and for which $\Delta \neq 0$ is guaranteed by Theorem 2. If we have multiple S_j with $p_j > 1$ which satisfy these conditions, we select the one with the smallest average Euclidean distance to the cell center $\boldsymbol{\xi}_{center_j}$.

SIMPLEX SAMPLING

Simplices are refined by randomly placing a point inside the sub-simplex (4.13). If we would like to uniformly sample a line section with the end points $[\xi_0, \xi_1]$ we would use the mapping

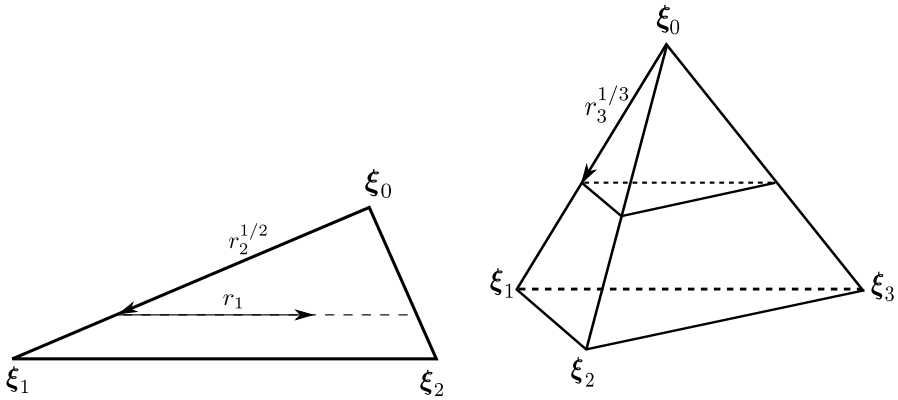
$$M_1 = \boldsymbol{\xi}_0 + r_1(\boldsymbol{\xi}_1 - \boldsymbol{\xi}_0), \tag{4.21}$$

where $r_1 \sim \mathscr{U}[0,1]$. Generating points inside a triangle can be done with

$$M_2 = \boldsymbol{\xi}_0 + r_2^{1/2}(\boldsymbol{\xi}_1 - \boldsymbol{\xi}_0) + r_2^{1/2} r_1(\boldsymbol{\xi}_2 - \boldsymbol{\xi}_1) \tag{4.22}$$

which maps points $\{r_1, r_2\}$ inside the unit square K_2 to points inside a triangle described by the vertices $\{\xi_0, \xi_1, \xi_2\}$ [30]. The working principle of (4.22) is shown in Figure 4.4(a). The parameter $r_2^{1/2}$ selects a line segment parallel to the edge $[\xi_0, \xi_1]$, while r_1 selects a point along the chosen line segment. The exponent $1/2$ ensures that uniformly distributed points in the square yield uniformly distributed points in the triangle. This can be shown by considering the length of the chosen line segment, which increases linearly when $r_2^{1/2}$ moves from $\boldsymbol{\xi}_0$ to $\boldsymbol{\xi}_1$. Since we require a uniform distribution of points, and considering $r_1 \sim \mathscr{U}[0,1]$, the pdf of $r_2^{1/2}$ should be linear as well. If we have the random variable $X = r^{1/\tau}$ with $r \sim \mathscr{U}[0,1]$ and $\tau \in \mathbb{N}_{>0}$, we find the cumulative distribution function (cdf) of X as

$$F_X(x) = \mathbb{P}(X \leq x) = \mathbb{P}\left(r^{1/\tau} \leq x\right) = \mathbb{P}\left(r \leq x^{\tau}\right) = x^{\tau}. \tag{4.23}$$



(a) Triangle. (b) Tetrahedron.

Figure 4.4: Selecting a point inside a triangle and tetrahedron.

And thus we have the pdf $f_X(x) = \mathrm{d}F_X/\mathrm{d}x = \tau x^{\tau-1} \sim \text{Beta}(\tau, 1)$. Therefore, in order to have a linear pdf for $r^{1/\tau}$, we must set $\tau = 2$.

It is suggested in [30] that (4.22) can be extended to higher dimensions, although no specific formulas are given. Hence, we use the same principle to select uniformly distributed points inside a tetrahedron, see Figure 4.4(b). Here, the parameter $r_3^{1/3}$ selects a triangle parallel to the base of the tetrahedron. From there we use $r_2^{1/2}$ and r_1 as before to select a point on this triangle. The exponent $1/3$ again ensures that the point distribution will be uniform. Note that the area of the selected triangles increases quadratically as $r_3^{1/3}$ moves from $\boldsymbol{\xi}_0$ to $\boldsymbol{\xi}_1$. Hence, it must be distributed as $\text{Beta}(3,1)$. We can now derive an expression for M_3 using the geometrical similarities between the base triangle and the selected parallel triangle, which gives us

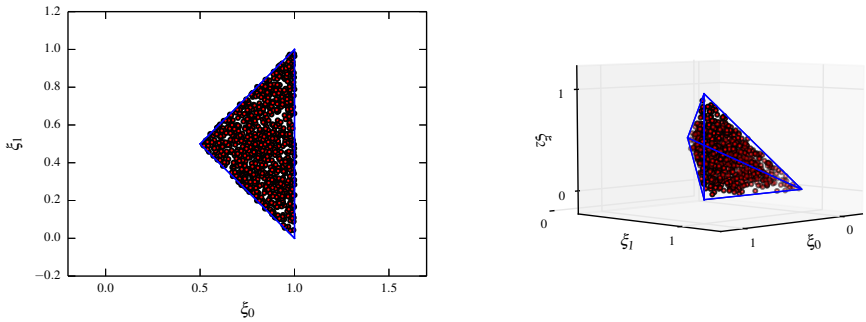
$$M_3 = \boldsymbol{\xi}_0 + r_3^{1/3}(\boldsymbol{\xi}_1 - \boldsymbol{\xi}_0) + r_3^{1/3} r_2^{1/2}(\boldsymbol{\xi}_2 - \boldsymbol{\xi}_1) + r_3^{1/3} r_2^{1/2} r_1(\boldsymbol{\xi}_3 - \boldsymbol{\xi}_2). \tag{4.24}$$

When comparing (4.21), (4.22) and (4.24) we see a pattern emerge which suggests that the map from a n_ξ-dimensional hypercube to a n_ξ-dimensional simplex with vertices $\{\boldsymbol{\xi}_0, \cdots, \boldsymbol{\xi}_{n_\xi}\}$ in $\mathbb{R}^{n_\xi}$ and uniform point distribution is

$$M_{n_\xi} = \boldsymbol{\xi}_0 + \sum_{i=1}^{n_\xi} \prod_{j=1}^{i} r_{n_\xi - j + 1}^{\frac{1}{n_\xi - j + 1}} (\boldsymbol{\xi}_i - \boldsymbol{\xi}_{i-1}), \tag{4.25}$$

where again the r_q are distributed as $\mathcal{U}[0,1]$.

To test (4.25) in 2 and 3 dimensions we can simply plot samples points, an example of which can be found in Figure 4.5. We also placed 10^4 samples in simplices up to 8



(a) Samples mapped from the unit square K_2 to a triangle.

(b) Samples mapped from the unit cube K_3 to a tetrahedron.

Figure 4.5: An example of the map (4.25) for $n_\xi = 2,3$ and 1000 samples.

dimensions, and found no outliers. The proof that (4.25) produces uniformly distributed samples can be found in E.

4.3. SSC SET-COVERING METHOD

In this section we describe an alternative stencil creating methodology, which results in a computational speed up in higher dimensions.

4.3.1. SET COVERING STENCILS

As previously mentioned, Section 4.5 will show that the enforcement of the LEC condition can become computationally expensive for high n_ξ and p_j. This is especially true for smooth response surfaces of the QoI. For many of stencils for our discontinuous problem, the LEC condition is violated and p_j is reduced which in turn significantly lowers the total required number of surrogate model evaluations (n_w) needed to check (4.14). This does not happen very often when the response surface is smooth. As a consequence of the exponential nature of n_w (see Section 4.5.1), we also see an exponential increase in the computation time needed to construct the surrogate model. Note that this increase is due to the SSC procedure, and thus is additional to the time needed to sample the computer code.

However, the problem lies not only with the exponential increase of n_w, but also in the extremely large overlap of the stencils S_j. Note that the standard SSC method enforces the LEC condition for all simplices Ξ_j in all stencils S_j. Hence, in each simplex Ξ_j, w_j is evaluated the same number of times as Ξ_j appears in all stencils S_j. For a two-dimensional example see Figure 4.6. There are two stencils, denote them S_r and S_q, associated to two different simplices Ξ_r and Ξ_q. The dark coloured simplices are the ones which appear in both stencils. Thus, when the LEC condition is checked for both stencils, w_r but also w_q is evaluated in the dark simplices. Moreover, since there are n_e stencils, the overlap will be large, and many different w_j will be evaluated in the same

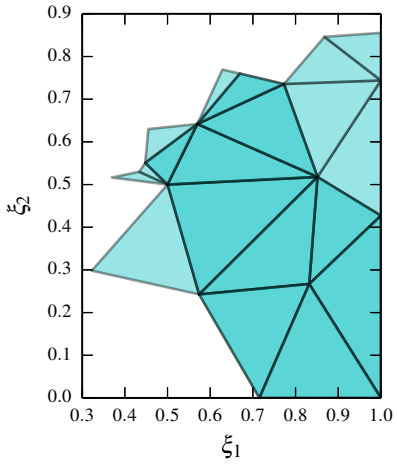


Figure 4.6: Two stencils which overlap each other. The dark simplices are shared by both stencils.

simplex element. This is no bottleneck for problems of low-dimensionality, but if the dimension increases this overlap will make the LEC condition very costly to enfore, see Section 4.5.1.

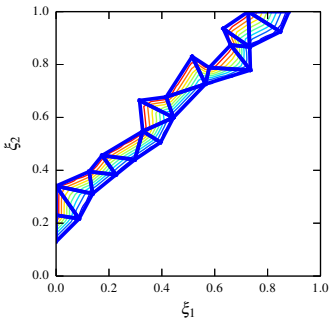
We propose an alternative technique for problems with higher n_ξ, using Set-Covering (SC) stencils based on the well-know set-covering problem [14], stated as follows in SSC terminology:

Set Covering Problem. *Let $X_j = \{\Xi_{j,0}, \cdots, \Xi_{j,K}\}$ be the set of all simplices that are inside the domain spanned by the vertices of stencil S_j. Then, Given the set $\mathscr{X} = \{X_1, \cdots, X_{n_e}\}$, and the set of all simplices $\mathscr{U} = \{\Xi_1, \cdots, \Xi_{n_e}\}$, find the smallest subset $\mathscr{C} \subseteq \mathscr{X}$ that covers $\mathscr{U}$, i.e. for which*

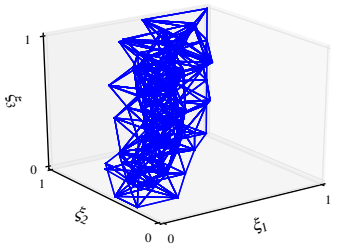
$$\mathscr{U} \subseteq \bigcup_{X_j \in \mathscr{C}} X_j$$

holds.

It is shown in [14] that the set-covering problem is NP-complete, and thus no fast solution is known. We could approximate $\mathscr{C}$ by the greedy algorithm, which at each step simply selects the X_j with the largest number of uncovered simplices. We then would have to check the LEC condition for all stencils in $\mathscr{S}_{sc}$, defined as the set of S_j corresponding to the $X_j \in \mathscr{C}$. For (high-dimensional) problems with a maximum polynomial order $p_{max} > 1$, the number of stencils in $\mathscr{S}_{sc}$ will be significantly lower than n_e. However, this approach would still require to construct all $X_j \in \mathscr{X}$. Also, many of the X_j could potentially cross a discontinuity, leading to a violation of the LEC condition and the subsequent reduction in size of X_j. When this happens the SC property of $\mathscr{C}$ can no longer be guaranteed. Thus, an iterative approach would be necessary which runs until $\mathscr{S}_{sc}$ satisfies both the SC and LEC property.



(a) 2D discontinuous simplices plus contourlines from the QoI. Notice that the QoI is essentially flat outside the discontinuous simplices for this particular problem.



(b) 3D discontinuous simplices.

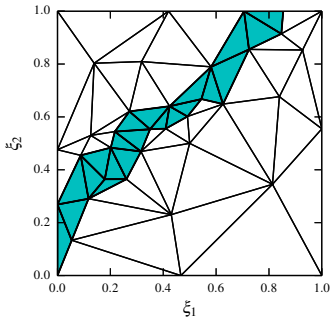
Figure 4.7: Discontinuous simplices identified by (4.26).

For reasons of computational efficiency, we want to avoid this iterative approach as much as possible, and thus not rely completely on the LEC condition to turn a set of nearest-neighbor stencils into a set of ENO stencils. Hence we will use the information contained in $\mathbf{v}$ regarding the discontinuity location to create a small set of SC stencils that also resemble ENO stencils, i.e. which do not cross a discontinuity. We will denote these stencils as SCENO stencils. The Ξ_j through which the discontinuity runs are identified by simply imposing a threshold v_t on the maximum jump observed in $\mathbf{v}$ at each simplex. Then, the set of discontinuous simplices can be defined as

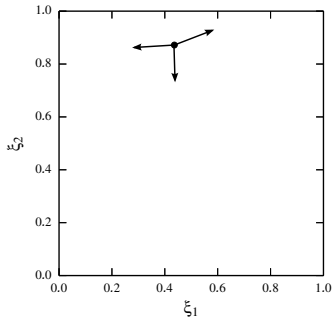
$$\mathscr{D} = \{\Xi_j \,|\, |\max \mathbf{v}_{k_{j,l}} - \min \mathbf{v}_{k_{j,l}}| \geq v_t,\ l = 0, \cdots, n_\xi,\ j = 1, \cdots, n_e\} \tag{4.26}$$

In our case the threshold value is set to $v_t = 1.0$. A two and three-dimensional visualisation of the $\Xi_j \in \mathscr{D}$ can be found in Figure 4.7. We furthermore redefine the set $\mathscr{C}$ as the set containing all the simplices Ξ_j that are currently covered by a stencil S_j.

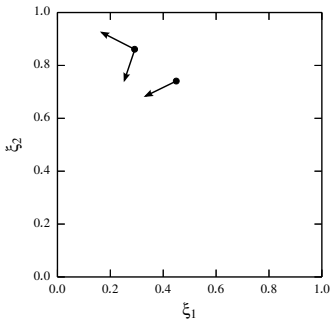
The general outline for constructing the SCENO stencils is now as follows. For the $\Xi_j \in \mathscr{D}$ we set $p_j = 1$ and $\mathscr{C} = \mathscr{C} \cup \mathscr{D}$, i.e. we add all discontinuous simplices to the set of covered simplices as well. Next, we select a $\Xi_j^* \in \mathscr{U} \setminus \mathscr{C}$, defined as the simplex with the largest number of uncovered neighbouring simplices. If there are multiple Ξ_j^* which have the maximum number of uncovered neighbouring simplices (i.e. at most $n_\xi + 1$), we select from those the Ξ_j^* with the large volume $\bar{\Xi}_j$. For the selected simplex we grow its stencil by adding neighbouring Ξ_j which are not covered yet, i.e. which are not in $\mathscr{C}$. We continue growing the stencil either until there are no more uncovered stencils or until S_j is large enough to allow interpolation or order p_{max}. We then move to the next Ξ_j^* and repeat until $\mathscr{C}$ covers $\mathscr{U}$. For a graphical representation of the stencil construction we refer to Figure 4.8. It is important to note that our main goal is to find a set $\mathscr{C}$ with a cardinality significantly less that n_e, rather than approximating the true minimal $\mathscr{C}$ of the SC problem as closely as possible.



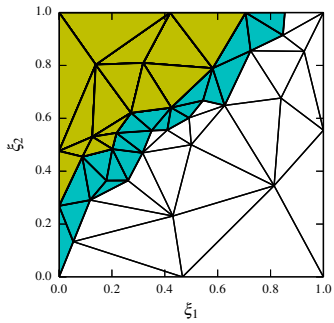
(a) Identify discontinuous simplices $\mathscr{D}$ and set their interpolation order to 1. Add all $\Xi_j \in \mathscr{D}$ to $\mathscr{C}$.



(b) Select uncovered simplex Ξ_j^* and add its neighbours to $\mathscr{C}$. Add corresponding nodes to S_j.



(c) Add uncovered neighbours of previously added simplices to $\mathscr{C}$. Add corresponding nodes to S_j.



(d) Iterate until the stencil size is large enough or until no more uncovered neighbours are available. Goto (b).

Figure 4.8: A two-dimensional example of the SC stencil construction.

This approach assures that we have a relatively small set $\mathscr{S}_{sc}$ for which: i) each simplex Ξ_j is present in only one X_j, ii) that not all $X_i \in \mathscr{X}$ need be calculated, iii) that no X_j crosses a discontinuity, and iv) the $\Xi_j \in \mathscr{D}$ are interpolated linearly. The result is that the number of times the LEC condition needs to be checked is reduced significantly. Only for those S_j associated to the $X_j \in \mathscr{C} \setminus \mathscr{D}$ it is still necessary to check for interpolation overshoots, since the $\Xi_j \in \mathscr{D}$ are guaranteed to be LEC due to their linear interpolation. The property of SCENO stencils mentioned under iii) also means that the number of times the LEC condition is violated is reduced, although not always to zero due to reasons of ill conditioning of the sample matrix (4.18). This is especially true for high n_ξ. An approach as described in Section 4.2.2 would render some of the advantages mentioned under i) - iv) void. Reducing p_j for ill-conditioned stencils will increase the cardinality of $\mathscr{S}_{sc}$, and all $X_j \in \mathscr{X}$ should be calculated in order to look for alternative stencils. Instead we directly solve an ill-conditioned system (4.19) in the non-null subspace of the solution as described in [15]. This method utilizes Gauss-Jordan elimination with complete pivoting to identify the null subspace of a singular matrix Ψ, i.e. $\Psi_{null}\mathbf{c}_{null} = 0$. This partitions the linear system as depicted below,

$$\begin{bmatrix} \Psi_{range} & \cdots \\ \cdots & \Psi_{null} \end{bmatrix} \begin{bmatrix} \mathbf{c}_{range} \\ \mathbf{c}_{null} \end{bmatrix} = \begin{bmatrix} \mathbf{v}' \\ \vdots \end{bmatrix}, \tag{4.27}$$

where $\Psi_{range}\mathbf{c}_{range} = \mathbf{v}'$ is the non-null subspace in which we can obtain accurate solutions. In the case of an ill-conditioned system, the null subspace is closely approximated by a space where the pivots ψ_{ii} are very small but not exactly equal to zero. The start of this 'near-null' subspace is identified by the first pivot ψ_{ii} for which the condition $|\psi_{ii}/\eta_c| < \epsilon$ holds, where η_c is the largest pivot of Ψ and ϵ is a very small parameter, which we set equal to machine precision ($\epsilon \approx 2.22 \cdot 10^{-16}$). In both the ill-conditioned and singular case the detrimental effect of Ψ_{null} on the solution is eliminated by a so-called zeroing operation, which basically replaces Ψ_{null} by an identity matrix of equal dimension and sets $\mathbf{c}_{null} = 0$. Thus, effectively speaking those coefficients $c_{j,l}$ which have been overwhelmed by round-off error are automatically cut out of the expansion (4.6). In our experiments we found that the dimension of Ψ_{null}, i.e. the nullity of Ψ, is small compared to the dimension of the full Ψ, see Table 4.1 for some typical examples at $n_\xi = 6$.

Table 4.1: Examples of ill-conditioned systems. We show the dimension n_ξ, the polynomial order of the stencil, the nullity and condition number of the sample matrix Ψ, and finally the condition number of the non-null Ψ_{range}.

n_ξ	p_j	dimension Ψ	nullity Ψ	cond. Ψ	cond. Ψ_{range}
6	2	28×28	1	1.36e+17	9.39e+3
6	3	82×82	1	1.31e+17	2.40e+4
6	3	84×84	2	2.85e+17	2.66e+3

If the system of equations is well-posed, the algorithm amounts to regular Gauss-Jordan elimination with complete pivoting. In any case, the quality of the response surface is checked via the LEC condition.

In C the algorithm for constructing the SCENO stencils is displayed in detail, and the results of the SSC-SC method can be found in Section 4.5.

4.4. HIGH-DIMENSIONAL MODEL-REDUCTION TECHNIQUES

The use of SCENO stencils makes the SSC method much more computationally efficient when $n_\xi \approx 5,6$. Much beyond that the exponential increase of n_e will render even this approach intractable when only moderate computational resources are available.

However, in physical systems it is often found that only a few parameters are influential, and only low-order correlations between the input parameters have a significant impact on the output. To capitalize on this behavior, High-Dimensional Model-Reduction techniques can be applied, see the references of Rabitz and Aliş [24, 25]. Our QoI is represented by a n_ξ-dimensional function $f(\boldsymbol{\xi},\mathbf{x})$ defined on the hypercube K_{n_ξ}, where $\mathbf{x}$ is a possible physical coordinate which we will again omit from the notation for the sake of brevity. Then, the HDMR expansion is an exact and finite hierarchical expansion of component functions of increasing dimension, given by

$$f(\boldsymbol{\xi}) = f_0 + \sum_i f_i(\xi_i) + \sum_{i_1<i_2} f_{i_1 i_2}(\xi_{i_1},\xi_{i_2}) + \cdots + \sum_{i_1<\cdots<i_l} f_{i_1\cdots i_l}(\xi_{i_1},\cdots,\xi_{i_l}) + \cdots + f_{1\cdots n_\xi}(\xi_{i_1},\cdots,\xi_{i_{n_\xi}}). \tag{4.28}$$

Here, the $i_1,\cdots,i_{n_\xi}$ are integers satisfying $1 \le i_1 < i_2 < \cdots < i_{n_\xi} \le n_\xi$. The zero-th order component function f_0 is a constant and represents the mean effect. The first-order function $f_i(\xi_i)$ is a univariate function, generally nonlinear, which represents the effect of independently varying input parameter ξ_i. Higher order functions represent the cooperative effects of increasing number of variables acting together on the output. If high order correlations are weak, the physical system $f(\boldsymbol{\xi})$ can be efficiently represented by a truncated L-th order expansion, where $L < n_\xi$. This a called a problem with low *effective dimension*, which occurs frequently in problems of physical nature [10]. Thus, the general idea is to solve multiple low-dimensional subproblems in place of a single high-dimensional one. The resultant computational effort to determine the component functions will scale polynomially, rather than the traditional exponential increase with n_ξ [24].

A measure μ for the measure space $(K^{n_\xi},\mathscr{B}(K^n),\mu)$, where $\mathscr{B}$ is the Borel σ-algebra on K^{n_ξ}, is defined as

$$\begin{aligned} \mathrm{d}\mu(\boldsymbol{\xi}) := \mathrm{d}\mu(\xi_1,\cdots,\xi_{n_\xi}) = \prod_{i=1}^{n_\xi} \mathrm{d}\mu_i(\xi_i), \quad \int_{K_1} \mathrm{d}\mu_i(\xi_i) = 1, \\ \mathrm{d}\mu(\boldsymbol{\xi}) = g(\boldsymbol{\xi})\mathrm{d}\boldsymbol{\xi} = \prod_{i=1}^{n_\xi} g_i(\xi_i)\mathrm{d}\xi_i. \end{aligned} \tag{4.29}$$

Here, $g(\xi_i)$ is the marginal density of the input ξ_i. It is the particular form chosen for the $g_i(\xi_i)$ that will determine the form of the component functions. In order to compute these functions, let us also define unconditional and conditional mean with respect to a

group of input variables as

$$\mathbf{M}f(\boldsymbol{\xi}) = \int_{K_{n_\xi}} f(\boldsymbol{\xi}) d\mu, \qquad \mathbf{M}^{(i_1\cdots i_l)} f(\boldsymbol{\xi}) = \int_{K_{n_\xi - l}} f(\boldsymbol{\xi}) \left[\prod_{j\notin\{i_1\cdots i_l\}} d\mu_j(\xi_j) \right] \tag{4.30}$$

Then, via a family of projection operators $P_{i_1\cdots i_l} : K^{n_\xi} \to K^l$, the component functions are recursively defined as follows [24]:

$$\begin{aligned}
f_0 &:= P_0 f(\boldsymbol{\xi}) = \mathbf{M} f(\boldsymbol{\xi}) \\
f_i(\xi_i) &:= P_i f(\boldsymbol{\xi}) = \mathbf{M}^{(i)} f(\boldsymbol{\xi}) - P_0 f(\boldsymbol{\xi}) \\
f_{ij}(\xi_i,\xi_j) &:= P_{ij} f(\boldsymbol{\xi}) = \mathbf{M}^{(ij)} f(\boldsymbol{\xi}) - P_i f(\boldsymbol{\xi}) - P_j f(\boldsymbol{\xi}) - P_0 f(\boldsymbol{\xi}) \\
&\vdots \\
f_{i_1\cdots i_l}(\boldsymbol{\xi}) &:= P_{i_1\cdots i_l} f(\boldsymbol{\xi}) = \mathbf{M}^{(i_1\cdots i_l)} f(\boldsymbol{\xi}) - \sum_{j_1<\cdots<j_{l-1}\subset\{i_1\cdots i_l\}} P_{j_1\cdots j_{l-1}} f(\boldsymbol{\xi}) - \cdots - P_0 f(\boldsymbol{\xi})
\end{aligned} \tag{4.31}$$

The component functions $f_{i_1,\cdots i_l}$ and $f_{j_1\cdots j_k}$ are independent and orthogonal, thus as long as one index between $\{i_1,\cdots i_l\}$ and $\{j_1\cdots j_k\}$ differs we have

$$\int_{K_{n_\xi}} f_{i_1,\cdots i_l}(\xi_{i_1},\cdots,\xi_{i_l}) f_{j_1\cdots j_k}(\xi_{j_1},\cdots,\xi_{j_k}) \mathrm{d}\mu = 0 \tag{4.32}$$

The correlation interpretation of $f_{i_1\cdots i_l}$ is associated with the chosen form of the measure μ. If $g_i = 1$, $i = 1,\cdots,n_\xi$, the Lebesgue measure ($\mathrm{d}\mu = \mathrm{d}\xi_1\mathrm{d}\xi_2\cdots\mathrm{d}\xi_{n_\xi}$) is retrieved and (4.28) together with (4.31) becomes the well-know Analysis Of Variance (ANOVA) decomposition. Computing the component functions in the ANOVA decomposition involves evaluating multi-dimensional integrals, which can be done by for instance MC techniques [28]. An alternative which is more computationally tractable is the cut-HDMR decomposition proposed in [24, 25]. In this case the measure is defined as

$$\mathrm{d}\mu = \prod_{i=1}^{n_\xi} \delta(\xi_i - \eta_i)\mathrm{d}\xi_i, \tag{4.33}$$

i.e. $g_i(\xi_i) = \delta(\xi_i - \eta_i)$, a Dirac measure located at the 'cut center' $\boldsymbol{\eta} = (\eta_1,\eta_2,\cdots,\eta_{n_\xi})$. This choice removes the need for evaluating multi-dimensional integrals, and it expresses $f(\boldsymbol{\xi})$ as a superposition of its values along lines, planes and hyperplanes passing through the cut center $\boldsymbol{\eta}$. The component functions (4.31) now become

$$\begin{aligned}
f_0 &:= P_0 f(\boldsymbol{\xi}) = f(\boldsymbol{\eta}) \\
f_i(\xi_i) &:= P_i f(\boldsymbol{\xi}) = f^{(i)}(\xi_i) - P_0 f(\boldsymbol{\xi}) \\
f_{ij}(\xi_i,\xi_j) &:= P_{ij} f(\boldsymbol{\xi}) = f^{(ij)}(\xi_i,\xi_j) - P_i f(\boldsymbol{\xi}) - P_j f(\boldsymbol{\xi}) - P_0 f(\boldsymbol{\xi}) \\
&\vdots \\
f_{i_1\cdots i_l}(\boldsymbol{\xi}) &:= P_{i_1\cdots i_l} f(\boldsymbol{\xi}) = f^{(i_1\cdots i_l)}(\xi_{i_1},\cdots,\xi_{i_l}) - \sum_{j_1<\cdots<j_{l-1}\subset\{i_1\cdots i_l\}} P_{j_1\cdots j_{l-1}} f(\boldsymbol{\xi}) - \cdots - P_0 f(\boldsymbol{\xi}).
\end{aligned} \tag{4.34}$$

Here, $f^{(i_1\cdots i_l)}(\xi_{i_1},\cdots,\xi_{i_l})$ is the conditional mean (4.30) taken with respect to measure (4.33), and thus it equals f with its inputs ξ_i set to η_i, except inputs $\xi_{i_1},\cdots,\xi_{i_l}$. As an example, consider the univariate function $f^{(i)}(\xi_i) = f(\eta_1,\cdots,\eta_{i-1},\xi_i,\eta_{i+1},\cdots,\eta_{n_\xi})$.

The authors of [18] used the cut-HDMR framework coupled with their Adaptive Sparse-Grid (ASG) collocation method [17], where they chose $\boldsymbol{\eta}$ as the mean of the random input vector. Besides truncating (4.28) at a certain order, they also made their approach dimension adaptive based on weights which identify the important dimensions. Although their ASG method uses only a linear finite-element basis, interpolation overshoots can still occur. Thus, motivated by their work in [18] we will also employ a dimension adaptive cut-HDMR approach, except we will couple it with the SSC method utilizing the SCENO stencils to avoid the mentioned downsides of ASG.

If we define $\mathcal{K} := \{1,2,\cdots,n_\xi\}$, the HDMR expansion (4.28) can be written in shorthand notation as [18]

$$f(\boldsymbol{\xi}) = \sum_{\mathbf{u}\subseteq\mathcal{K}} f_{\mathbf{u}}(\boldsymbol{\xi}_{\mathbf{u}}) = \sum_{\mathbf{u}\subseteq\mathcal{K}}\sum_{\mathbf{v}\subseteq\mathbf{u}} (-1)^{|\mathbf{u}|-|\mathbf{v}|} f^{(\mathbf{v})}(\boldsymbol{\xi}_{\mathbf{v}}), \tag{4.35}$$

where in the first equality we sum over the powerset of $\mathcal{K}$, i.e. over all possible subsets $\mathbf{u}\subseteq\mathcal{K}$. We furthermore set $f_\emptyset = f_0$. The second equality is obtained by expanding each component function $f_{\mathbf{u}}(\boldsymbol{\xi}_{\mathbf{u}})$ as indicated in (4.34). Notation wise, if for instance $\mathbf{v}=\{1,4,6\}$, then $f^{(\mathbf{v})}(\boldsymbol{\xi}_{\mathbf{v}}) = f^{(146)}(\xi_1,\xi_4,\xi_6)$. Each individual $|\mathbf{v}|$-dimensional subproblem $f^{(\mathbf{v})}(\boldsymbol{\xi}_{\mathbf{v}})$ can be approximated by a SSC surrogate (4.6). In that case (4.35) becomes

$$f(\boldsymbol{\xi}) \approx w(\boldsymbol{\xi}) = \sum_{\mathbf{u}\subseteq\mathcal{K}}\sum_{\mathbf{v}\subseteq\mathbf{u}} (-1)^{|\mathbf{u}|-|\mathbf{v}|} \sum_{j=1}^{n_e}\sum_{l=0}^{N_j} c_{jl}\boldsymbol{\Psi}_{jl}(\boldsymbol{\xi}_{\mathbf{v}}). \tag{4.36}$$

In order to assess the convergence of each individual $f^{(\mathbf{v})}(\boldsymbol{\xi}_{\mathbf{v}})$, the authors of [18] use the hierarchical surplus. This is also possible in the case of the SSC method, see (4.15). Alternatively, the RMS error estimate (4.17) can used for this purpose. Since (4.17) is a global error estimate and it also includes information from the distribution of the input parameters, we use the RMS error to assess the convergence.

Furthermore, the mean of each component function, defined as $J_{\mathbf{u}}$, can also be computed from the surrogate model

$$J_{\mathbf{u}} = \sum_{\mathbf{v}\subseteq\mathbf{u}} (-1)^{|\mathbf{u}|-|\mathbf{v}|} \sum_{j=1}^{n_e}\sum_{l=0}^{N_j} c_{jl}\mathbb{E}\left[\boldsymbol{\Psi}_{jl}(\boldsymbol{\xi}_{\mathbf{v}})\right]. \tag{4.37}$$

We compute (4.37) via random sampling, which can be performed quickly since it requires only sampling the surrogate model.

In order to identify the important dimensions, all first order component functions $f_i(\xi_i)$ are computed. Again, these are one-dimensional functions which measure the impact of a single independent input parameter on the output. Next, a weight is defined

$$\alpha_i = \frac{\|J_i\|_2}{\|f_0\|_2}, \tag{4.38}$$

which measures the contribution of each individual ξ_i on the mean of all first order component functions [18]. We always take the L_2 norm $\|\cdot\|_2$ over the spatial domain.

Equation (4.38) can be considered as a sensitivity index, and only those dimensions for which (4.38) is larger than a user-prescribed error threshold ϵ_1 are considered important. All higher order $f_{\mathbf{v}}(\boldsymbol{\xi}_{\mathbf{v}})$ where $\mathbf{v}$ contains indices of dimensions which did not make the cut will not be computed. Consider e.g. a n_ξ-dimensional problem on K_{n_ξ}, where only $\mathbf{v} = \{1\}$ and $\mathbf{v} = \{2\}$ satisfy $\alpha_i > \epsilon_1$. The only higher-order component function that will be computed in this case is $f_{12}(\xi_1, \xi_2)$, regardless of the value of n_ξ.

The downside of (4.38) is that it is hard to choose ϵ_1 beforehand. One should first create the first-order HDMR expansion and decide on an appropriate value *a postiori*. An alternative is to use a weight measuring the relative contribution of J_i with respect to the sum of all first-order means, i.e.

$$\alpha_i = \frac{\|J_i\|_2}{\sum_{k=1}^{n_\xi} \|J_k\|_2}. \tag{4.39}$$

Now one can *a priori* choose a $\epsilon_1 \in [0,1]$, and select the smallest set of important dimensions for which the sum of their α_i is greater than ϵ_1.

Dimension adaptivity is extended to higher dimensions as well by defining a weight for $|\mathbf{u}| > 1$ as [18]

$$\alpha_{\mathbf{u}} = \frac{\|J_{\mathbf{u}}\|_2}{\|\sum_{\mathbf{v}\in\mathcal{V}_{comp}, |\mathbf{v}|<|\mathbf{u}-1|} J_{\mathbf{v}}\|_2}. \tag{4.40}$$

Here, the set $\mathcal{V}_{comp}$ simply holds all the indices $\mathbf{v}$ that were computed. Furthermore, all subsets $\mathbf{v}$ of component functions which are important are added to a set $\mathcal{V}_{imp}$. That way, a higher-order important $\mathbf{u}$ is admissible if all $\mathbf{v} \subset \mathbf{u}$ required to compute (4.35) are also in $\mathcal{V}_{imp}$. This is the so-called admissibility condition, which is given by

$$\mathbf{u} \in \mathcal{V}_{imp} \text{ and } \mathbf{v} \subset \mathbf{u} \Rightarrow \mathbf{v} \in \mathcal{V}_{imp}. \tag{4.41}$$

Similar to the first-order case, we can define a relative counterpart of (4.40) as

$$\alpha_{\mathbf{u}} = \frac{\|J_{\mathbf{u}}\|_2}{\sum_{\mathbf{v}\in\mathcal{V}_{comp}, |\mathbf{v}|=|\mathbf{u}|} \|J_{\mathbf{v}}\|_2}, \tag{4.42}$$

such that the $\alpha_{\mathbf{u}}$ sum to one and we can choose a $\epsilon_1 \in [0,1]$ *a priori*.

Finally, a relative error measure between two HDMR expansions of consecutive orders $p-1$ and p is defined as

$$\alpha_p = \frac{\|\sum_{|\mathbf{u}|\le p} J_{\mathbf{u}} - \sum_{|\mathbf{u}|\le p-1} J_{\mathbf{u}}\|_2}{\|\sum_{|\mathbf{u}|\le p-1} J_{\mathbf{u}}\|_2}. \tag{4.43}$$

The algorithm stops when α_p becomes smaller than another used-defined threshold ϵ_2. An overview of the HDMR algorithm is depicted in D.

4.5. RESULTS AND DISCUSSION

4.5.1. SSC METHOD

We present the results obtained with the baseline SSC method with nearest-neighbour or ENO stencils, versus the SSC method with the SCENO stencils. As a test case we use a quasi-1D nozzle case, described in the next section.

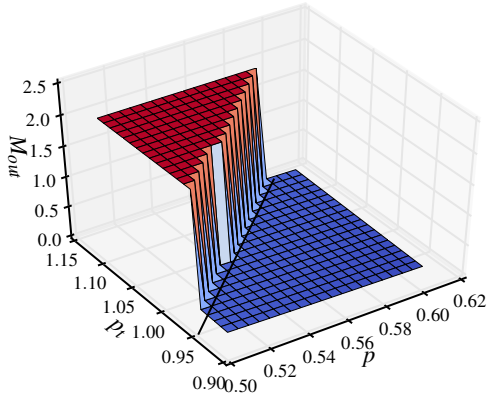


Figure 4.9: M_{out} as function of p and p_t obtained by MC sampling, with the geometrical constants fixed to their nominal value.

NOZZLE FLOW

As a test case we use the solver from [23], which computes the flow through a quasi-1D diverging nozzle. We prescribe the flow to be sonic at the nozzle inlet, i.e. $M_{in} = 1$. From fluid mechanics we know that the flow is driven by the pressure gradient, i.e. by the difference between the total pressure p_t at the inlet and the static pressure p of the surroundings at the nozzle exit. Depending on the value of p_t/p, the flow can show very different behavior. If p_t/p exactly equals the adaptation value, the flow reaches the static pressure of the surroundings at the nozzle exit and the jet exhausts smoothly into the atmosphere. A stronger p_t/p will result in smooth flow through the nozzle, which is supersonic at the nozzle exit. In order to match the outside pressure p, the flow undergoes a supersonic expansion attached to the nozzle exit (under-expanded nozzle). A smaller p_t/p, but still above a threshold that depends on the ratio of the exit to the throat area, still results in smooth flow through the nozzle, but this is now over-expanded and is compressed to the outside pressure through an oblique shock attached to the nozzle exit. When p_t/p is equal to the threshold value, the flow is characterized by a normal shock located at the nozzle exit: upstream of the shock, the flow is smooth, and verifies adaptation conditions in the exit section; immediately downstream of it, the flow is subsonic and matches the outside pressure. Finally, when p_t/p is below the threshold value, a normal shock wave is formed somewhere inside the nozzle. This results in subsonic flow at the exit, and an exit pressure that is equal to p [2].

Given the pressure gradient, the flow is completely characterized by the shape of the nozzle [2]. As in [23], we consider the following hyperbolic tangent for the nozzle shape

$$f(x) = a + b\tanh(cx - d). \tag{4.44}$$

To test the SSC method, we specify two different ranges for the uncertain parameters

such that two radically different response surfaces have to be created. First, we prescribe a wider range for p such that the QoI is highly discontinuous, see Figure 4.9. In the second case we restrict p to a more narrow interval such that the QoI is smooth. More specifically, we prescribe the uniform input distributions for the 6 uncertain parameters described in Table 4.2. Furthermore, we choose M_{out} (the Mach numbers at the nozzle

Table 4.2: Uncertain input parameters of the discontinuous (D) and smooth (S) case.

n_ξ	Parameter	Mean (D)	Range (D)	Mean (S)	Range (S)
1	p [bar]	0.55	[0.5, 0.6]	0.625	[0.60, 0.65]
2	p_t [bar]	1.0	[0.9, 1.1]	1.0	[0.9, 1.1]
3	a [-]	1.75	[1.575, 1.925]	1.75	[1.575, 1.925]
4	b [-]	0.7	[0.63, 0.77]	0.7	[0.63, 0.77]
5	c [-]	0.8	[0.72, 0.88]	0.8	[0.72, 0.88]
6	d [-]	4.0	[3.6, 4.4]	4.0	[3.6, 4.4]

exit) as our quantity of interest, as it allows us to easy calculate other flow quantities via the isentropic relations once M_{out} is known [2]. When constructing the surrogate models, we will use a linear transformation for each input to map points from $[0,1]$ in the stochastic domain to points in the physical domain with the range as specified in Table 4.2. This simplifies the construction of the surrogate models as it allows us to always work in the standard n_ξ-dimensional hypercube K_{n_ξ}.

In Table 4.3 we show the computation time T in minutes versus the dimension n_ξ, in case of the discontinuous QoI for both the baseline and the method based on SCENO stencils. This is of course dependent upon the available computational resources, in our case a 24 core workstation. We can see that T rises very quickly as n_ξ increases in the case of the baseline method. In the case of the 6-dimensional baseline method with the discontinuous QoI, we restricted the maximum polynomial interpolation order to 2 and we used nearest neighbour stencils instead of ENO stencil in an attempt to reduce the computational time. Still, the surrogate model construction required 1000 minutes (16.7 hours) to be completed. Due to this high cost we did not attempt to construct a 6-dimensional surrogate model for the smooth QoI, which is expected to take even longer.

To explain which element of the SSC method is responsible for the high computation time, we also show the percentage of T in Table 4.3 that is spent on the LEC condition, construction of the stencils S_j, and QoI calculation. The results for the continuous QoI are depicted in Table 4.4. Due to the absence of discontinuities, the construction of ENO stencils was not necessary for the baseline method, and nearest-neighbor stencils were used. Note that the number of samples n_s for the baseline and the SSC-SC method differs for the same n_ξ. This is due to the refinement of simplices at the hypercube boundaries. As briefly explained in Section 4.2.1, we refine the edges of simplices located at the hypercube boundary, rather than the simplex interior. If two of such neighbouring simplices are both selected for refinement, we make sure that we do not refine the same edge twice. This approach results in a different n_s for the same number of iterations.

Since the nozzle code is just a cheap test problem, Table 4.3 shows that computing the QoI samples $\mathbf{v}$ only takes up a significant portion of T for low n_ξ. For the baseline SSC

Table 4.3: The computational cost of the discontinuous QoI.

type	n_ξ [-]	n_s [-]	T [min]	LEC [%T]	S_j [%T]	**v** [%T]
baseline	2	49	1.66	17.48	11.21	70.17
	3	140	5.02	55.08	15.28	22.7
	4	184	12.66	70.97	17.73	5.87
	5	210	156.09	73.85	23.44	0.51
	6	186	1000.63	75.6	N/A	0.07
SCC-SC	2	57	0.42	0.67	0.66	83.72
	3	141	0.7	1.6	2.28	86.07
	4	219	1.0	7.06	6.98	74.17
	5	243	1.99	27.86	12.38	41.29
	6	307	22.15	61.56	11.51	11.0

Table 4.4: The computational cost of the smooth QoI.

type	n_ξ [-]	n_s [-]	T [min]	LEC [%T]	S_j [%T]	**v** [%T]
baseline	2	41	0.65	1.35	N/A	83.69
	3	95	1.0	12.34	N/A	54.73
	4	104	2.15	65.87	N/A	19.17
	5	129	47.13	95.18	N/A	0.94
SCC-SC	2	57	0.59	0.53	0.51	83.85
	3	112	0.74	1.3	3.26	84.57
	4	108	0.64	5.44	5.89	71.7
	5	161	1.37	16.88	26.95	37.41
	6	193	40.87	0.92	94.77	1.05

method the construction of ENO-type stencils makes up a significant part of the computational cost, but Table 4.3 clearly indicates that the enforcement of the LEC condition is by far the most expensive component in higher dimensions. From Table 4.4 we can infer that this is especially true for a smooth QoI, in which case roughly 95% is spent on the LEC condition. Thus, for the baseline method, most of the the computational effort is put into enforcing the LEC condition. For that reason the computational cost of the LEC condition is investigated in more detail.

As explained in Section 4.2.1, the LEC condition (4.14) is enforced by a MC approach, for all simplices in S_j at all $j = 1, \cdots, n_e$. Thus, for the baseline SSC method the number of times the surrogate model is evaluated in each iteration i is bounded by

$$n_{w_i} = n_e \times n_{e,S_j}, \quad i = 1, \cdots, I \tag{4.45}$$

where n_{e,S_j} is the number of simplices in a single stencil S_j with $p = p_{max}$, and I is the total number or of iterations of the SSC algorithm. Here we assumed that per S_j, one sample is placed in each simplex using (4.25). The number of points in the Delaunay grid is given simply by (4.5), but estimating n_e for arbitrary n_ξ is not trivial. The worst-case number of simplices in a Delaunay triangulation is bounded by the so-called Upper-Bound theorem, which states that n_e is at most of $\mathcal{O}(n_s^{n_\xi/2})$. In the best-case scenario (points distributed uniformly at random inside the unit sphere) n_e scales as $\mathcal{O}(n_s)$ for any n_ξ, with a constant factor that is exponential with the dimension [1]. To find out where in between these two bounds our specific problem resides, we plot n_e versus n_s in Figure 4.10 for $n_\xi \in \{2, \cdots, 6\}$. These results indicate that we are close to the $\mathcal{O}(n_s)$ bound, since the $n_e(n_s)$ are described quite well by the linear regression also shown in Figure 4.10. However, the exponential increase of $\mathrm{d}n_e/\mathrm{d}n_s$ means that for a moderate number of samples we can still have a large number of simplices if n_ξ is high enough. Note that other than limiting the number of samples n_s, we have no means of controlling the magnitude of n_e.

The term n_{e,S_j} in (4.45) grows exponentially with p_j for a given n_ξ. This can be seen in Figure 4.11, where we plot n_{e,S_j} versus the local polynomial order p_j for $n_\xi = 5, 6$. Unlike n_e, we obviously have some control over the magnitude of n_{e,S_j} through the inclusion of a maximum allowable cutoff value for p_j. The upper bound (4.45), added over iterations i is plotted as a function of n_s in Figure 4.12 for $n_\xi = 2, \cdots, 6$. It shows a rapid increase in this bound with both n_ξ and p_j.

By comparing the computational time T of the SSC method with that of the SSC-SC method (Tables 4.3-4.4), it is clear that the SSC-SC method is several orders of magnitude more efficient for high n_ξ. As an example consider the discontinuous QoI for $n_\xi = 6$. In this case, with 186 samples in $\mathbf{v}$, the baseline SSC method needed $T = 1000$ minutes to construct the surrogate model. The SSC-SC method only required roughly 22 minutes, even with a higher number of samples ($n_s = 307$). The distribution of computational time is also different than for the baseline method. For the smooth QoI, almost all the computational effort in put into creating the SC stencils. In case of the discontinuous QoI, a lot of simplices were in $\mathscr{D}$, and thus the number of uncovered simplices is smaller at the start of the SC algorithm. As a result, creating the SC stencils takes less time than for the smooth QoI at the same n_ξ.

To clearly explain why the SSC-SC method is more computationally efficient than the

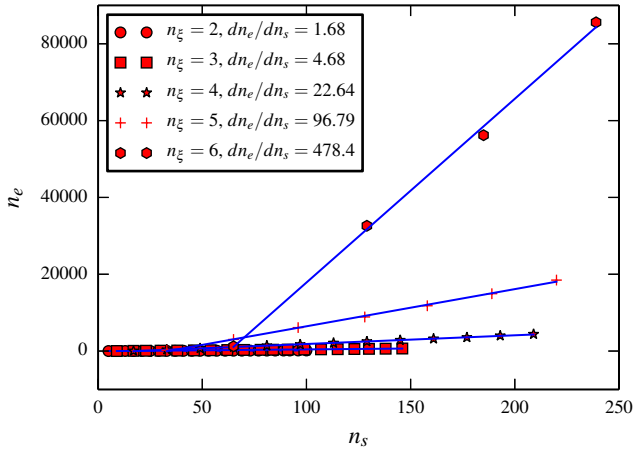
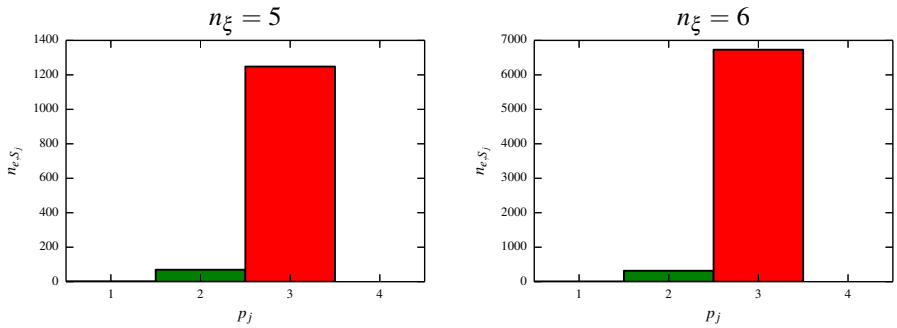


Figure 4.10: n_e as function of n_s for the smooth QoI. The discontinuous QoI gives a similar figure. The slope dn_e/dn_s is computed via a least-square regression line.



(a) $n_s = 73$, $p_{max} = 3$ (b) $n_s = 105$, $p_{max} = 3$

Figure 4.11: n_{e,S_j} versus p_j for $n_\xi \in \{5,6\}$.

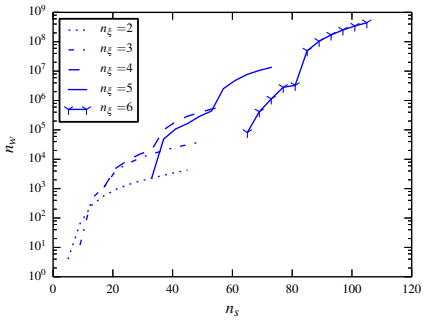


Figure 4.12: The maximum number of surrogate model evaluations vs n_s. Here, $p_{max} = 4$ for $n_\xi = 2,3$ and $p_{max} = 3$ for $n_\xi = 4,5,6$. The jumps that can be observed occur when n_s is large enough to allow a higher p_j.

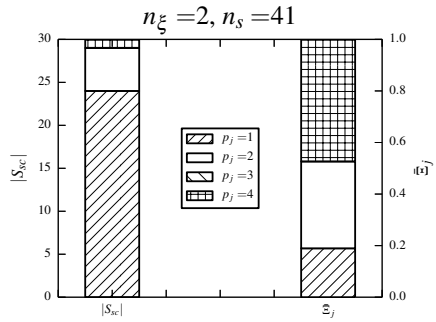
baseline method, we depict the number of stencils in $\mathscr{S}_{sc}$ ($|\mathscr{S}_{sc}|$) and their volume $\bar{\Xi}_{S_j}$ as a function of p_j in Figure 4.13, for a 2 and 6-dimensional discontinuous QoI. Note that although by far the most $S_j \in \mathscr{S}_{sc}$ are linear, they occupy as relatively small part of the probabilistic domain considering their share in $|\mathscr{S}_{sc}|$. Especially note the case for $n_\xi = 6$, where all $S_j \in \mathscr{S}_{sc}$ except 10 are linear. However, these 10 high-order stencils do occupy 40.7% of the total probabilistic domain. Since linear S_j are guaranteed to be LEC, we only have to check the LEC condition for the 10 high-order stencils. Contrast this with the baseline SSC method, where the LEC condition is checked for all n_e (initially high-order) stencils. In the particular case of Figure 4.13b this would amount to 70.884 LEC iterations during 1 iteration of the SSC method.

The results for the smooth QoI are shown in Figure 4.14. Due to the lack of discontinuous simplices there are dramatically fewer stencils. Only those simplices that are 'left over' without free neighbours at the end of the SCENO stencil construction are given a linear stencil. Still, for $n_\xi = 6$ they make up the majority of stencils in $\mathscr{S}_{sc}$. Unlike the discontinuous case however, they cover a small fraction of the probabilistic domain, i.e. only 4.2 %. Thus again we have only a few number of high-order S_j for which we have to invest computational effort, while at the same time these stencils cover 95.8% of the domain.

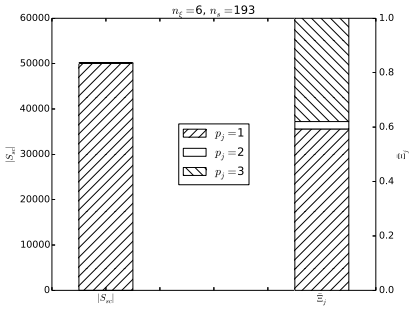
As stated in Section 4.2.1, our primary interest is computing the statistical moments of the QoI, in particular the mean and standard deviation. To assess the accuracy of the SSC method we used a reference solution for each considered dimension n_ξ. To compute the errors we define the following relative L_2 error measures for the mean, standard deviation and interpolation surface

$$\epsilon_\mu = \frac{\|\mu_w - \mu_{ref}\|_2}{\|\mu_{ref}\|_2}, \quad \epsilon_\sigma = \frac{\|\sigma_w - \sigma_{ref}\|_2}{\|\mu_{ref}\|_2}, \quad \epsilon_w = \frac{\|\mathbf{w}(\boldsymbol{\xi}_{ref}) - \mathbf{v}_{ref}\|_2}{\|\mathbf{v}_{ref}\|_2}. \tag{4.46}$$

Here, the subscript w denotes a quantity computed with the surrogate model, and ref is the exact value computed via random sampling. In the interpolation surface error, $\mathbf{v}_{ref}$ is a vector containing 10^4 MC code samples and $\mathbf{w}(\boldsymbol{\xi}_{ref})$ are the surrogate model outputs evaluated at the same MC locations $\boldsymbol{\xi}_{ref}$. The values of the error measures (4.46) for both

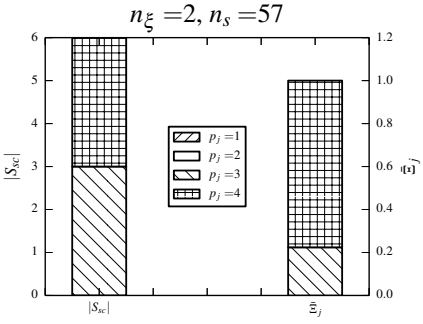


(a) Results for $n_\xi = 2$ with the discontinuous QoI. There are 24 $S_j \in \mathscr{S}_{sc}$ with $p_j = 1$, responsible for 19.0% of $\bar{\Xi}_{S_j}$, 5 $S_j \in \mathscr{S}_{sc}$ that takes up 33.7% of the volume, 0 $S_j \in \mathscr{S}_{sc}$ with $p_j = 3$, and 1 $S_j \in \mathscr{S}_{sc}$ with $p_j = 4$ which makes up 47.3% of $\bar{\Xi}_{S_j}$. The total number of simplices is 67.

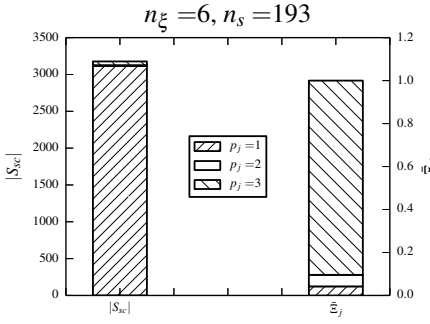


(b) Results for $n_\xi = 6$ with the discontinuous QoI. There are 50.135 $S_j \in \mathscr{S}_{sc}$ with $p_j = 1$, responsible for 59.3% of $\bar{\Xi}_{S_j}$, 1 $S_j \in \mathscr{S}_{sc}$ with $p_j = 2$ that covers 2,7 %, and 9 $S_j \in \mathscr{S}_{sc}$ with $p_j = 3$ which make up 38.0% of $\bar{\Xi}_{S_j}$. The total number of simplices is 70.884.

Figure 4.13: The number of stencils in $\mathscr{S}_{sc}$ (left bars) and the volume $\bar{\Xi}_{S_j}$ (right bars) per polynomial order for $n_\xi = 2$ and $n_\xi = 6$.



(a) Results for $n_\xi = 2$ with the smooth QoI. There are no stencils for $p_j = 1,2$, and there are 3 $S_j \in \mathscr{S}_{sc}$ for both $p_j = 3$ and $p_j = 4$, responsible for 22.4% and 77.6% of $\bar{\theta}_j$ respectively. The total number of simplices is 93.



(b) Results for $n_\xi = 6$ with the smooth QoI. There are 3116 $S_j \in \mathscr{S}_{sc}$ with $p_j = 1$, responsible for 4.2% of $\bar{\Xi}_{S_j}$, 10 $S_j \in \mathscr{S}_{sc}$ with $p_j = 2$ which cover 5.4 %, and 51 $S_j \in \mathscr{S}_{sc}$ with $p_j = 3$ that take up 90.4% of $\bar{\Xi}_{S_j}$. The total number of simplices is 69.088.

Figure 4.14: The number of stencils in $\mathscr{S}_{sc}$ (left bars) and the volume $\bar{\Xi}_{S_j}$ (right bars) per polynomial order for $n_\xi = 2$ and $n_\xi = 6$.

QoIs and both surrogate models can be found in Tables 4.5-4.6. Note that the error levels are roughly the same for both surrogate models.

Table 4.5: The relative errors (4.46) of the discontinuous QoI for the baseline SSC and SSC-SC method.

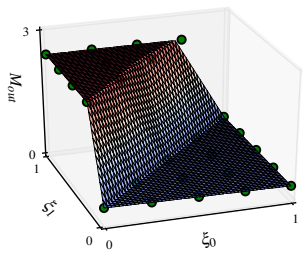
type	n_ξ	n_s	ϵ_μ	ϵ_σ	ϵ_w
baseline	2	49	4.603e-03	6.318e-02	2.035e-01
	3	140	1.619e-02	7.620e-02	2.439e-01
	4	184	1.131e-02	1.450e-01	2.699e-01
	5	210	3.966e-02	2.516e-01	3.473e-01
SCC-SC	2	57	6.952e-03	3.499e-02	1.852e-01
	3	141	3.855e-03	6.501e-02	1.984e-01
	4	219	1.548e-02	1.273e-01	2.457e-01
	5	243	3.787e-02	1.845e-01	2.820e-01
	6	307	7.518e-02	2.833e-01	3.552e-01

Table 4.6: The relative errors (4.46) of the continuous QoI for the baseline SSC and SSC-SC method.

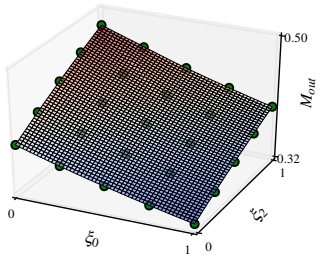
type	n_ξ	n_s	ϵ_μ	ϵ_σ	ϵ_w
baseline	2	41	2.752e-07	2.140e-06	1.842e-06
	3	95	2.156e-05	5.955e-04	4.486e-04
	4	104	2.891e-05	2.131e-04	6.282e-04
	5	129	7.534e-05	8.174e-04	2.669e-03
SCC-SC	2	57	1.483e-05	7.760e-05	1.536e-04
	3	112	1.390e-05	1.868e-04	1.503e-04
	4	108	1.572e-06	1.189e-05	1.441e-04
	5	161	5.060e-05	5.486e-04	2.467e-03
	6	193	1.921e-04	7.687e-03	8.496e-03

From Tables 4.5-4.6 we note that the errors of the discontinuous case are considerable higher than for the smooth case. This can be attributed to the smearing of discontinuities, i.e. the linear interpolation of a discontinutity over a simplex, which especially contributes to the error of the surrogate model in higher dimensions. See for instance Figure 4.15, which depicts 2D projections of a 3D surrogate model along with reference data on an ordered uniform grid. Especially in Figure 4.15(c) we can clearly identify regions where the smearing of the discontinuity contributes to the error. For this particular case, we plotted the difference between the surrogate model and the reference data in Figure 4.15(d), which also identifies sharp regions of high error. This situation gets progressively worse as n_ξ increases.

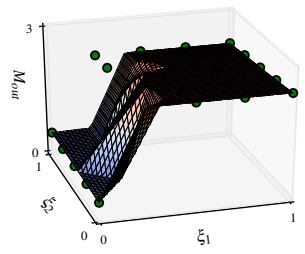
In [33, 37] Witteveen et. al apply a subcell-resolution approach to the SSC method for the case when the discontinuity in the probabilistic space is a function of a physical discontinuity with random location. Our results indicate that for high n_ξ subcell resolution could prove to be benificial, even if the physical location of the QoI is not random.



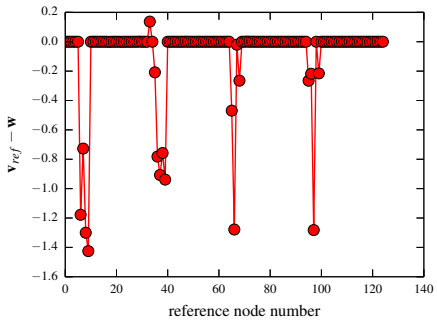
(a) Fixed ξ_2.



(b) Fixed ξ_1.

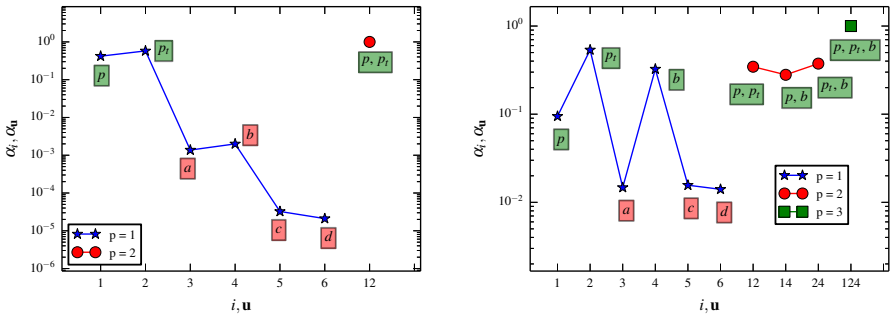


(c) Fixed ξ_0.



(d) The difference $\mathbf{v}_{ref} - \mathbf{w}$ vs the reference node number for fixed ξ_0.

Figure 4.15: 3D surrogate model displayed in 2 dimensions by fixing 1 dimension to a particular value. The green dots are reference data on an ordered uniform grid.



(a) Discontinuous QoI. (b) Smooth QoI

Figure 4.16: The relative weights (4.39) and (4.42) for the discontinuous and smooth nozzle QoI. The green boxes are the dimensions which are added to V_{imp}.

4.5.2. CUT-HMDR APPLIED TO NOZZLE FLOW

We now show the results obtained from using the cut-HDMR approach coupled with the SSC-SC method, applied to the nozzle flow case.

The first order weights α_i (4.39) are used to determine the important dimensions. Figure 4.16 shows α_i with $i = 1, \cdots, 6$ corresponding to the 6 parameters of Table 4.2, as well as the subsequent higher order $\alpha_{\mathbf{u}}$. For the discontinuous QoI only the first 2 parameters are significant (p and p_t). Together they are responsible for 99.7 % of the total first order mean. We have set $\epsilon_1 = 0.9$, such that those dimensions which make up 90 % or more of the p-th order mean are added to $\mathcal{V}_{imp}$. For the smooth QoI we need p and p_t as well, but also the coefficient b in order to meet this constraint. All admissible subsequent dimensions are important as well.

The computational time for both QoIs, due to the fact that the nozzle code is quickly evaluated, is in the order of several minutes. In order to create a baseline SSC surrogate model which can also be evaluated in the full six-dimensional space, the computational time is much greater, see Tables 4.3-4.4.

The values of error measures (4.46) for the discontinuous and smooth case are given in Tables 4.7-4.8. Notice that a first-order HDMR expansion is not sufficient for the discontinuous QoI, but already for $p = 2$ the relative errors in the statistical moments are of $\mathcal{O}(10^{-2})$. Table 4.8 shows the results for the smooth QoI. The errors, even for a first-order expansion, are of $\mathcal{O}(10^{-3})$ or below. Again, the higher errors in the discontinuous case can be attributed to the linear smearing of discontinuities in the response surface.

Table 4.7: Relative error values vs the HDMR order p and n_s in the case of the discontinuous QoI.

HDMR order p	ϵ_1	n_s	ϵ_μ	ϵ_σ	ϵ_w
1	0.9	57	4.514e-01	3.601e-01	7.434e-01
2		89	3.926e-02	8.830e-02	2.196e-01

Table 4.8: Relative error values vs the HDMR order p and n_s in the case of the smooth QoI.

HDMR order p	ϵ_1	n_s	ϵ_μ	ϵ_σ	ϵ_w
1	0.9	37	4.933e-05	2.885e-04	5.035e-03
2		67	3.603e-05	6.457e-04	3.608e-03

Compare the relative errors from Table 4.5 with those of Table 4.7, and likewise for Tables 4.6 and 4.8. The errors are of a similar order of magnitude, even though the HDMR method doesn't sample the full six-dimensional space. At the same time we gain information about the correlation between the input parameters, in the sense of their combined impact on the code output. Also, the cut-HDMR method could be applied to even higher dimensional spaces, provided that the effective dimension is low.

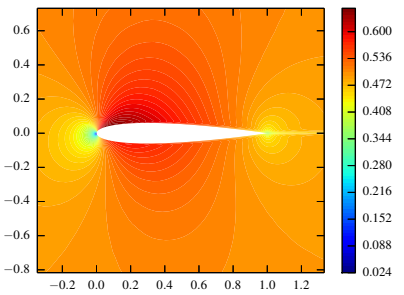
These results demonstrate the power of the cut-HDMR technique in terms of computational efficiency for problems with low effective dimension. The cost of computing 6 one-dimensional and 1-3 two-dimensional surrogate models is significantly less than computing 1 six-dimensional problem. Also, in the case of the SSC method, it can avoid problems with the bad scalability of the Delaunay triangulation with increasing n_ξ. And although the SSC-SC method is computationally more efficient than the baseline method, Tables 4.3-4.4 show that the computational cost also rises with n_ξ. Thus, given certain computational resources, there will be a maximum n_ξ beyond which this method will no longer be computationally tractable. Instead, the cut-HDMR method could be applied to even higher dimensional spaces, provided that the effective dimension is low.

4.5.3. CUT-HDMR APPLID TO AIRFOIL FLOW

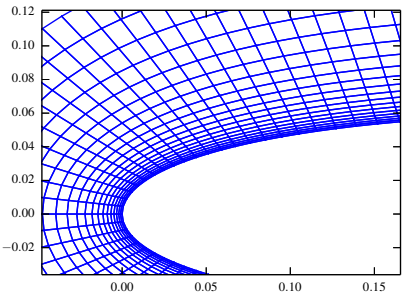
In this Section we present the results of the cut-HDMR approach (again coupled to the SSC-SC method), when applied to a computationally expensive problem, i.e. the turbulent flow over a NACA0012 airfoil, see Figure 4.17a. The freestream Mach number is 0.5 and the angle of attack is set to 5°. The Reynolds number based on the chord length is $1.2 \cdot 10^7$. The grid is a C-grid with 70.085 nodes and the first node is located at a distance of 10^{-6} from the wall in order to provide sufficient resolution, see Figure 4.17b.

The governing equations are the Reynolds-Averaged Navier-Stokes (RANS) equations, coupled to the Spalart-Allmaras (SA) turbulence model [29]. This model contains 7 empirically determined closure coefficients [31], whose best-fit values are unknown *a-priori* [9]. Therefore, we treat all 7 inputs as uniformly distributed variables with the end points set at ±30% of their nominal values, see Table 4.9. The chosen QoI is the lift coefficient, defined as $c_l = L'/(\rho_\infty V_\infty^2/2)$, where ρ_∞ and V_∞ are the freestream density and velocity respectively. The term L' is the two-dimensional lift force.

The values of the relative weights α_i and $\alpha_{\mathbf{u}}$, i.e. equations (4.39) and (4.42), are depicted in Figure 4.18. The value of ϵ_1 was again set to 0.9. For the first-order HDMR expansion we need three coefficients, namely κ, C_{b1} and C_b2, to capture more than 90 % of the total first-order mean. Further note that the constants C_{w2} and C_{w3} are completely unimportant for the computation of our QoI, as their weights are of $\mathcal{O}(10^{-14})$. For the second-order expansion, we only need the interactions of (C_{b1}, κ) and (C_{b2}, κ)



(a) A deterministic solution showing the distribution of the Mach number. The freestream Mach number is 0.5 and the angle of attack is 5°.



(b) A zoom into the leading edge of the discretized airfoil. The first grid points are set at a distance of 10^{-6} from the wall.

Figure 4.17: The symmetrical NACA0012 airfoil.

Table 4.9: Uniformly distributed closure coefficients of the SA model.

Parameter	Mean	Range
C_{b1}	0.1355	[0.0949, 0.1762]
C_{b2}	0.622	[0.435, 0.809]
C_{v1}	7.1	[4.97, 9.10]
σ	2/3	[0.467, 0.867]
C_{w2}	0.3	[0.210, 0.390]
C_{w3}	2	[1.40, 2.60]
κ	0.41	[0.287, 0.455]

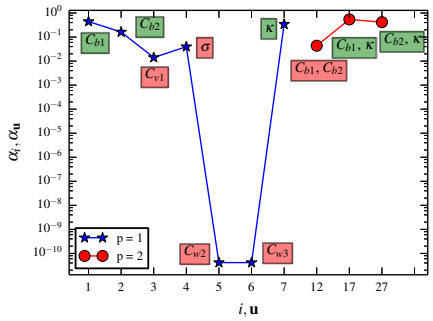


Figure 4.18: The relative weights (4.39) and (4.42) for the NACA 0012 test case. The value of ϵ_1 was set to 0.9. The green boxes indicate the dimensions which are important.

4

to represent more than 90 % of the second-order mean component functions. Although (C_{b1}, C_{b2}) is not added to $\mathcal{V}_{imp}$, the third-order interaction (C_{b1}, C_{b2}, κ) is still admissible according to the admissibility condition (4.41). However, in this particular simulation a second order expansion was enough to satisfy the error measure between two HDMR expansions of consecutive orders (4.43), which was set to $\epsilon_2 = 10^{-3}$. Hence, no third order interaction was computed.

As no reference solution is available for this test case, we plot the convergence of the mean and standard deviation of c_l in Table 4.10. As can be expected from the weights in Figure 4.18, there is little difference between the statistics of first and second order HDMR expansion. In this particular case even a first order expansion could be considered as converged.

Table 4.10: The convergence of the statistical moments of c_l as a function of the HDMR order.

HDMR order p	n_s	μ	σ
1	49	6.385509e-01	3.729915e-03
2	62	6.381699e-01	3.746544e-03

Note that if we would have applied either the baseline or the SSC-SC approach to a full 7-dimensional space, the initial Delaunay grid alone would be comprised of 129 samples. As can be seen from Table 4.10, the HDMR approach requires significantly less samples. This difference can be expected to increase as the dimensionality increases, provided that the problem is one of low effective dimension. It should be noted however, that Witteveen and Iaccarino suggested a method in which the initial 2^{n_ξ} samples can be avoided [35], but this approach requires extrapolation towards the hypercube boundaries. In this case there is no guarantee that the LEC limiter is respected in the simplices where the extrapolation takes place, and only in the limit $n_e \to \infty$ full extremum-diminishing robustness is recovered for the entire domain Ξ.

Finally, for each code run we saved the results for a range of different physical quantities. Thus, we can *a posteriori* construct a surrogate model for each of these quantities.

However, the samples were adaptively placed based on our chosen QoI c_l, and therefore might not be optimally distributed for another QoI. Moreover, the dimension adaptivity might cut out dimensions that are important for a QoI other than c_l. To perform a qualitative investigation, we therefore constructed a first-order HDMR expansion for the pressure coefficient c_p, skin-friction coefficient c_f, Mach number M and the turbulent kinetic energy k, all defined below.

$$c_p := \frac{p - p_\infty}{\frac{1}{2}\rho_\infty V_\infty^2}, \quad c_f := \frac{\tau_w}{\frac{1}{2}\rho_\infty V_\infty^2}, \quad M := \frac{u}{a}, \quad k := \frac{1}{2}\left(\overline{u_1'^2} + \overline{u_2'^2} + \overline{u_3'^2}\right) \tag{4.47}$$

Here, p and p_∞ are the static and freestream pressure respectively. Furthermore, ρ_∞ and V_∞ are defined as before in c_l. The quantity τ_w is the wall shear stress, and in the expression for M, u is the local velocity, and a is the speed of sound. Finally, $\overline{u_i'^2}$ is the mean normal Reynolds stress in x_i direction [31].

In Table 4.11 we show the coefficients of the SA model sorted according to their value of the (non-relative) weight α_i (4.38) for all the QoIs of (4.47). Note that which coefficients are most influential does not change much from one QoI to another. For c_p, c_f and M the three most important coefficients are κ, C_{b1} and C_{b1}. In the case of the turbulent kinetic energy k, σ has taken third place, with again C_{b1} and κ as the most influential. Still, σ's value of α_i is close to the weight corresponding to C_{b2}. The ranking of the bottom three coefficients remains unchanged for all considered QoIs.

Table 4.11: The rankings from largest (1^{st}) to smallest (7^{th}) α_i for QoIs other than c_l. Below each coefficient its corresponding value of α_i is written.

QoI	1^{st}	2^{nd}	3^{rd}	4^{th}	5^{th}	6^{th}	7^{th}
c_p	C_{b1}	κ	C_{b2}	σ	C_{v1}	C_{w2}	C_{w3}
	2.604e-03	1.339e-03	8.784e-04	5.510e-04	6.866e-05	3.674e-08	3.655e-08
c_f	κ	C_{b1}	C_{b2}	σ	C_{v1}	C_{w2}	C_{w3}
	1.764e-01	8.835e-02	8.676e-02	2.072e-02	6.819e-03	9.976e-07	9.875e-07
M	κ	C_{b1}	C_{b2}	σ	C_{v1}	C_{w2}	C_{w3}
	3.658e-03	1.951e-03	1.498e-03	6.591e-04	1.233e-04	3.958e-08	3.951e-08
k	C_{b1}	κ	σ	C_{b2}	C_{v1}	C_{w2}	C_{w3}
	8.708e-02	2.982e-02	2.726e-02	2.529e-02	5.735e-04	2.650e-05	2.591e-05

4.6. CONCLUSION

We have examined means to improve upon the performance of the Simplex-Stochastic Collocation (SSC) method [36] for uncertainty quantification problems involving computationally expensive computer codes. We found that in order to make the SSC method work for our considered range of n_ξ, we needed to add some new features. First, for high n_ξ we run the risk of obtaining a singular sample matrix. This can be circumvented by a method similar to the method used to construct the ENO stencils. If we encounter a stencil S_j for which the sample matrix is singular, we collect a set of candidate nearest-neighbour stencils which all contain the simplex Ξ_j associated to the j-th stencil S_j. We then select the stencil which has the highest polynomial order and which is non-singular.

In the worst-case scenario we get a linear stencil, which is guaranteed to be non-singular irrespective of the dimension n_ξ.

Due to the exponential increase in the number of simplex elements with increasing dimension, enforcing the LEC condition becomes quickly very expensive for $n_\xi \geq 5$. As a first measure to combat this sharp increase in the computational burden, we have proposed an alternative technique for the stencil construction, based on the Set-Covering (SC) problem [14]. Unlike in the SSC method we do not construct a stencil for every simplex. Since n_e increases exponentially fast with n_ξ, we want to find a minimal set of stencils that cover all simplices in the probability domain. Due to the fact that the stencil size rises also exponentially, only a few high-order stencils are required to achieve this. We furthermore use the information contained in the code samples about the location of a possible discontinuity in the construction of the stencils. Using a simple measure based upon the maximum jump in code samples, we mark those simplices which contain a discontinuity and manually set their respective stencils to first order. Using only the remaining simplices as admissible candidates, we grow stencils by adding neighbouring simplices. Once a simplex is covered by a stencil, it is no longer an admissible candidate. This approach assures that the number of stencils is significantly lower than n_e, and hence the number of times the LEC condition must be checked is reduced equally. Also, since the discontinuous simplices are removed as admissible candidates *a priori*, no stencil crosses a discontinuity. As a consequence the SC stencils resemble the ENO stencils in shape.

For dimensions 5 and 6, our SSC-SC method is significantly more computationally efficient as the baseline SSC method. However, like the original SC problem which is shown to be NP-complete in [14], the construction of our SC stencils also becomes expensive for high n_ξ. Therefore we examined another alternative, where we adapted the cut-HDMR method of [18] to the SSC method. Given a problem with low effective dimension, this approach circumvents the bad scalability of the Delaunay triangulation, while at the same time obtaining error estimates of similar order of magnitude compared to the full-dimensional baseline or SSC-SC method. Thus, this method is adaptive in both the stochastic domain as well as in the dimensions themselves, while retaining the advantages of the SSC method such as Runge-phenomenon free interpolation. We applied it to a computationally expensive flow case, i.e. the turbulent flow over an airfoil modelled with the Spalart-Allmaras eddy-viscosity model, which contains 7 imperfectly known closure coefficients. Instead of fully sampling a 7 dimensional space, and enforcing the LEC condition in this space, we could obtain a converged surrogate model with a second order HDMR expansion and 62 code samples.

Acknowledgements: The present work was supported by the French 'Agence Nationale de la Recherche' (ANR) under contract ANR-11-MONU-008-002.

REFERENCES

[1] N. Amenta, D. Attali, and O. Devillers. Complexity of delaunay triangulation for points on lower-dimensional polyhedra. In *Proceedings of the eighteenth annual*

ACM-SIAM symposium on Discrete algorithms, pages 1106–1113. Society for Industrial and Applied Mathematics, 2007.

[2] J.D. Anderson. *Fundamentals of aerodynamics*, volume 2. McGraw-Hill New York, 2001.

[3] J Axerio-Cilies, G Petrone, V Sellappan, and G Iaccarino. Extreme ensemble computation for optimization under uncertainty. *Evolutionary and Deterministic Methods for Design, Optimization and Control*, 2011.

[4] I. Babuška, F. Nobile, and R. Tempone. A stochastic collocation method for elliptic partial differential equations with random input data. *Siam J. Numer. Anal*, 45(3):1005–1034, 2007.

[5] K.C. Chung and T.H. Yao. On lattices admitting unique lagrange interpolations. *SIAM Journal on Numerical Analysis*, 14(4):735–743, 1977.

[6] Paola Cinnella, Pietro Marco Congedo, Valentino Pediroda, and Lucia Parussini. Sensitivity analysis of dense gas flow simulations to thermodynamic uncertainties. *Physics of Fluids (1994-present)*, 23(11):116101, 2011.

[7] Pietro Marco Congedo, Jeroen Witteveen, and Gianluca Iaccarino. A simplex-based numerical framework for simple and efficient robust design optimization. *Computational Optimization and Applications*, 56(1):231–251, 2013.

[8] Pietro Marco Congedo, Jeroen Witteveen, Gianluca Iaccarino, et al. Simplex-simplex approach for robust design optimization. In *EUROGEN 2011, International Conferences on Evolutionary Computing for Industrial Applications.-ECCOMAS Thematic Conference*, 2011.

[9] WN Edeling, P Cinnella, and RP Dwight. Predictive rans simulations via bayesian model-scenario averaging. *Journal of Computational Physics*, 275:65–91, 2014.

[10] Jasmine Foo and George Em Karniadakis. Multi-element probabilistic collocation method in high dimensions. *Journal of Computational Physics*, 229(5):1536–1557, 2010.

[11] Jasmine Foo, Xiaoliang Wan, and George Em Karniadakis. The multi-element probabilistic collocation method (me-pcm): Error analysis and applications. *Journal of Computational Physics*, 227(22):9572–9595, 2008.

[12] B. Ganapathysubramanian and N. Zabaras. Sparse grid collocation schemes for stochastic natural convection problems. *Journal of Computational Physics*, 225(1):652–685, 2007.

[13] Serge Guillas, Nina Glover, and Liora Malki-Epshtein. Bayesian calibration of the constants of the k-ε turbulence model for a cfd model of street canyon flow. *Computer Methods in Applied Mechanics and Engineering*, 2014.

[14] R.M. Karp. *Reducibility among combinatorial problems*. Springer, 1972.

[15] K.J. Kozaczek, S.K. Kurtzl, and D.S. Kurtz. A direct algorithm for solving ill-conditioned linear algebraic systems. *Advances in X-ray Analysis*, 42, 2000.

[16] G.J.A. Loeven, J.A.S. Witteveen, and H. Bijl. Probabilistic collocation: an efficient non-intrusive approach for arbitrarily distributed parametric uncertainties. In *Proceedings of the 45th AIAA Aerospace Sciences Meeting*, volume 6, pages 3845–3858, 2007.

[17] X. Ma and N. Zabaras. An adaptive hierarchical sparse grid collocation algorithm for the solution of stochastic differential equations. *Journal of Computational Physics*, 228(8):3084–3113, 2009.

[18] Xiang Ma and Nicholas Zabaras. An adaptive high-dimensional stochastic model representation technique for the solution of stochastic partial differential equations. *Journal of Computational Physics*, 229(10):3884–3915, 2010.

[19] X Merle and P Cinnella. Bayesian quantification of thermodynamic uncertainties in dense gas flows. *Reliability Engineering & System Safety*, 2014.

[20] F. Nobile, R. Tempone, and C.G. Webster. A sparse grid stochastic collocation method for partial differential equations with random input data. *SIAM Journal on Numerical Analysis*, 46(5):2309–2345, 2008.

[21] P.J. Olver. On multivariate interpolation. *Studies in Applied Mathematics*, 116(2):201–240, 2006.

[22] R. Pecnik, J.A.S. Witteveen, and G. Iaccarino. Uncertainty quantification for laminar-turbulent transition prediction in rans turbomachinery applications. *AIAA Paper 2011*, pages 1–14, 2011.

[23] M. Pini and P. Cinnella. Hybrid adjoint-based robust optimization approach for fluid-dynamics problems. In *15th Non-Deterministic Approaches Conference*. American Institute of Aeronautics and Astronautics, 2013.

[24] Herschel Rabitz and Ömer F Aliş. General foundations of high-dimensional model representations. *Journal of Mathematical Chemistry*, 25(2-3):197–233, 1999.

[25] Herschel Rabitz, Ömer F Aliş, Jeffrey Shorter, and Kyurhee Shim. Efficient input—output model representations. *Computer Physics Communications*, 117(1):11–20, 1999.

[26] R. Rojas, Sa. Kahunde, L. Peeters, O. Batelaan, L. Feyen, and A. Dassargues. Application of a multimodel approach to account for conceptual model and scenario uncertainties in groundwater modelling. *Journal of Hydrology*, 394(3):416–435, 2010.

[27] S. A. Smolyak. Quadrature and interpolation formulas for tensor products of certain classes of functions. In *Dokl. Akad. Nauk SSSR*, volume 4, pages 240–243, 1963.

[28] I.M. Sobol. Global sensitivity indices for nonlinear mathematical models and their monte carlo estimates. *Mathematics and computers in simulation*, 55(1-3):271–280, 2001.

[29] P.R. Spalart and S.R. Allmaras. A one-equation turbulence model for aerodynamic flows. *La recherche aérospatiale*, 1(1):5–21, 1994.

[30] G. Turk. Graphics gems. chapter Generating Random Points in Triangles, pages 24–28. Academic Press Professional, Inc., San Diego, CA, USA, 1990.

[31] D.C. Wilcox, American Institute of Aeronautics, and Astronautics. *Turbulence modeling for CFD*, volume 3. DCW industries La Canada, CA, 2006.

[32] J.A.S. Witteveen and G. Iaccarino. Simplex elements stochastic collocation in higher-dimensional probability spaces. In *12th AIAA Non-Deterministic Approaches Conference, Orlando, Florida, AIAA*, volume 2924, 2010.

[33] JA.S. Witteveen and G. Iaccarino. Introducing essentially non-oscillatory stencil selection with subcell resolution into uncertainty quantification. *Annual Research Briefs, Center for Turbulence Research*, pages 169–180, 2011.

[34] J.A.S. Witteveen and G. Iaccarino. Refinement criteria for simplex stochastic collocation with local extremum diminishing robustness. *SIAM Journal on Scientific Computing*, 34(3):A1522–A1543, 2012.

[35] J.A.S. Witteveen and G. Iaccarino. Simplex stochastic collocation with random sampling and extrapolation for nonhypercube probability spaces. *SIAM Journal on Scientific Computing*, 34(2):A814–A838, 2012.

[36] J.A.S. Witteveen and G. Iaccarino. Simplex stochastic collocation with eno-type stencil selection for robust uncertainty quantification. *Journal of Computational Physics*, 239:1–21, 2013.

[37] J.A.S. Witteveen and G. Iaccarino. Subcell resolution in simplex stochastic collocation for spatial discontinuities. *Journal of Computational Physics*, 251:17–52, 2013.

[38] J.A.S. Witteveen, A. Loeven, and H. Bijl. An adaptive stochastic finite elements approach based on newton–cotes quadrature in simplex elements. *Computers & Fluids*, 38(6):1270–1288, 2009.

5

PREDICTIVE RANS SIMULATIONS OF AN EXPENSIVE COMPUTER CODE

5.1. INTRODUCTION

In this chapter we will merge the developed methodologies from Chapter 3 and 4. In other words, we will use the improved SSC method to create a surrogate model that can be used in the predictive phase of the BMSA framework. To avoid sampling high-dimensional spaces, we will use the cut-HDMR approach. Furthermore, we will make use of the topology-independent nature of the closure coefficients, and apply the posterior distributions from the turbulent boundary layers to the (transonic) flow over the symmetric NACA64A010 airfoil (see Figure 5.1 for an example flow field).

Due to the number of required propagations, a full BMSA is computationally expensive in the case of airfoil flow. This is true even with the use of the improved SSC method. Therefore, to reduce the computational cost, we will propagate the scenario-averaged posteriors as described in Section 3.5.5 instead. The flow over an airfoil experiences different kinds of pressure gradients. For instance, the flow over the top may be subject to a favourable gradient, but in the transonic case a strongly adverse gradient is experienced as well due to the presence of a shock. We incorporate this expectation in our scenario-averaged posteriors by combining the posteriors from two favourable, one mildly adverse and one strongly adverse scenario. The results for the $k-\varepsilon$ and Baldwin-Lomax model can be found in Figure 5.2. Note that for the $k-\varepsilon$ model the inclusion of the strongly-adverse scenario resulted in a distribution for κ that is bi-modal, since low κ values are favoured in these scenarios (see Figure 3.7). The same behaviour can be observed in the distribution of $C_{cp}\alpha$ in the Baldwin-Lomax model.

The outline of this chapter is as follows. In the first section we give a brief overview of the computational framework that is used. In Section 5.3 we present the results, consisting of first the obtained spatial variability of the uncertainty estimates. Secondly, we compare the predictions with validation data. In the final section the conclusion is given.

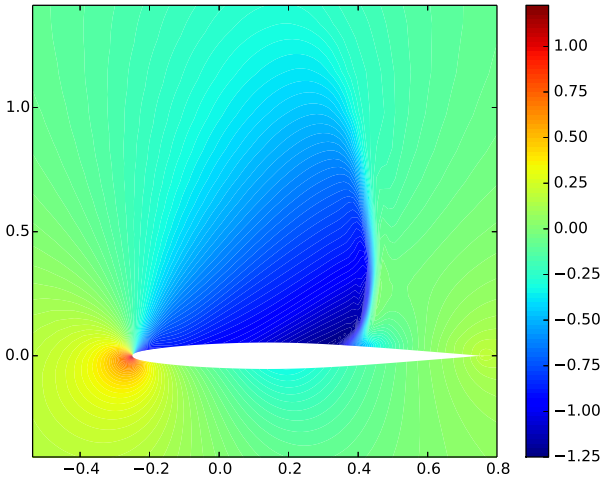


Figure 5.1: Contour plot of the pressure coefficient c_p, computed with the Baldwin Lomax model at an angle of attack of 3.5 degrees and a Mach number of 0.8.

5.2. COMPUTATIONAL FRAMEWORK

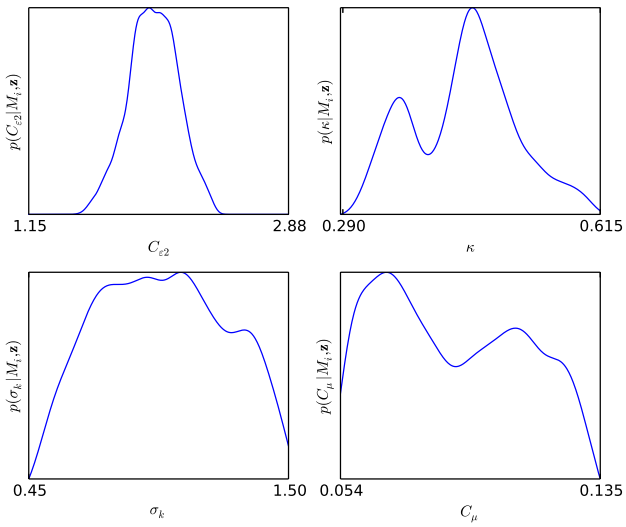
Because we combine multiple elements from different chapters, we will begin by giving a clear and concise overview of the computational framework used in this chapter. First, the BMSA framework with scenario-averaged posteriors is described by:

$$p(\tilde{\boldsymbol{\theta}}_i) := p(\boldsymbol{\theta}|M_i,\mathbf{z}) = \sum_{k=1}^{K} p(\boldsymbol{\theta}|M_i,\mathbf{S}_k,\mathbf{z})\,\mathbb{P}(\mathbf{S}_k), \quad i = 1,\cdots,M, \tag{5.1}$$

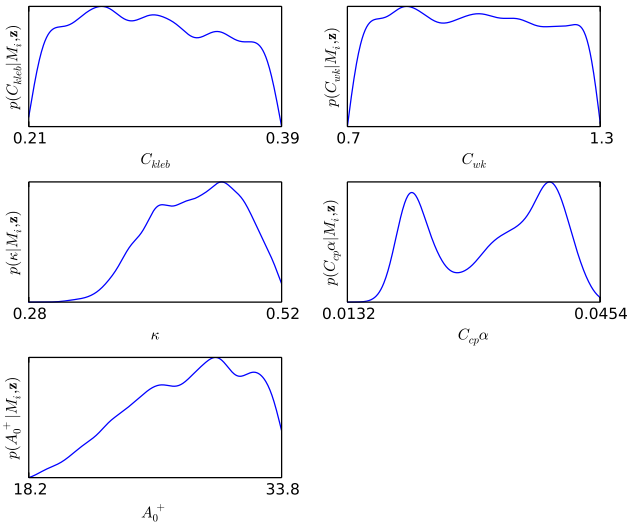
$$\mathbb{E}[\Delta|\mathbf{z}] = \sum_{i=1}^{I} \mathbb{E}[\tilde{\Delta}_i]\mathbb{P}(M_i|\mathbf{z}), \quad \mathbb{E}[\tilde{\Delta}_i] := \mathbb{E}[\Delta|M_i,\mathbf{z}], \quad \mathbb{P}(M_i|\mathbf{z}) = \sum_{k=1}^{K} \mathbb{P}(M_i|\mathbf{S}_k,\mathbf{z})\,\mathbb{P}(\mathbf{S}_k), \tag{5.2}$$

$$\mathrm{Var}[\Delta|\mathbf{z}] = \sum_{i=1}^{I} \mathrm{Var}[\tilde{\Delta}_i]\mathbb{P}(M_i|\mathbf{z}) + \sum_{i=1}^{I} \left(E[\tilde{\Delta}_i] - \mathbb{E}[\Delta|\mathbf{z}]\right)^2 \mathbb{P}(M_i|\mathbf{z}), \quad \mathrm{Var}[\tilde{\Delta}_i] := \mathrm{Var}[\Delta|M_i,\mathbf{z}]. \tag{5.3}$$

Here, (5.1) are the scenario averaged posteriors which are computed for each turbulence model in the set. Equation (5.2) describes the total BMSA prediction $\mathbb{E}[\Delta|\mathbf{z}]$. It also gives the prediction of the i^{th} turbulence model, obtained by propagating $p(\tilde{\boldsymbol{\theta}}_i)$ through the



(a) The distributions of the $k-\varepsilon$ model



(b) The distributions of the Baldwin-Lomax model

Figure 5.2: The scenario-averaged posteriors of the coefficients for two turbulence models, both created by assigning a weight of 0.25 to the distributions of the 1100, 6300, 1100 and 4400 scenarios.

RANS code. The third quantity of (5.2) denotes the weight applied to each model. Finally, the measure of uncertainty in the prediction is computed with (5.3). The right-hand side of this expression shows first the in-model variance, which is a measure for the uncertainty due to imperfectly known coefficients. Secondly, the between-model variance is displayed, i.e. our measure of model inadequacy. Through the term $\mathbb{P}(M_i|\mathbf{z})$, the uncertainty due to the variance of posterior distributions over different scenarios is implicitly present in both terms.

Unlike Chapter 3, the use of an expensive simulation code prevents us to use direct Monte-Carlo sampling to compute $\mathbb{E}[\tilde{\Delta}_i]$ and $\mathrm{Var}[\tilde{\Delta}_i]$. Therefore we use our improved SSC method coupled to the HDMR framework, which (when written in BMSA notation), is described by:

$$\tilde{\Delta}_i(\tilde{\boldsymbol{\theta}}_i) = \tilde{\Delta}(\tilde{\boldsymbol{\theta}}) = \sum_{\mathbf{u}\subseteq\mathscr{K}} \tilde{\Delta}_{\mathbf{u}}(\tilde{\boldsymbol{\theta}}_{\mathbf{u}}) = \sum_{\mathbf{u}\subseteq\mathscr{K}} \sum_{\mathbf{v}\subseteq\mathbf{u}} (-1)^{|\mathbf{u}|-|\mathbf{v}|} \tilde{\Delta}^{(\mathbf{v})}(\tilde{\boldsymbol{\theta}}_{\mathbf{v}}), \quad \mathscr{K} := \{1,\cdots,n_\xi\}, \tag{5.4}$$

$$\tilde{\Delta}^{(\mathbf{v})}(\tilde{\boldsymbol{\theta}}_{\mathbf{v}}) \approx \sum_{j=1}^{n_e} \sum_{l=0}^{N_j} c_{jl} \boldsymbol{\Psi}_{jl}(\tilde{\boldsymbol{\theta}}_{\mathbf{v}}). \tag{5.5}$$

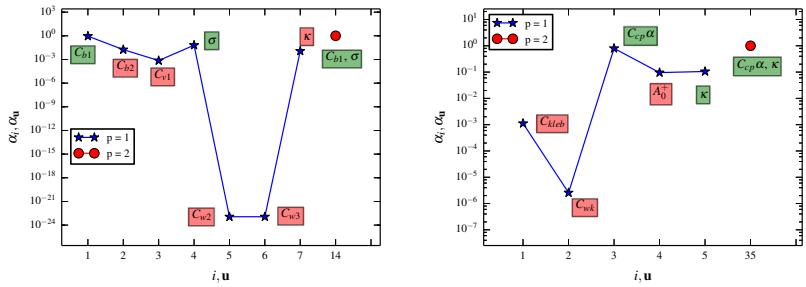
Note that we suppressed the model subscript i for brevity of notation. Equation (5.4) is the full HDMR expansion of the QoI Δ. Each component function $\Delta_{\mathbf{u}}(\boldsymbol{\theta}_{\mathbf{u}})$ is a linear combination of $|\mathbf{v}|$-dimensional functions $\Delta^{(\mathbf{v})}(\boldsymbol{\theta}_{\mathbf{v}})$, defined as the code output as function of $\boldsymbol{\theta}_{\mathbf{v}} \subseteq \boldsymbol{\theta}$. The remaining coefficients $\boldsymbol{\theta} \setminus \boldsymbol{\theta}_{\mathbf{v}}$ are set equal to their respective reference values. Therefore, each individual $\Delta^{(\mathbf{v})}$ can be approximated by an SSC expansion as described by (5.5). In practice, we do not compute the full HDMR expansion (5.4). Instead, we rely on weights such as (4.42) in order to identify the important multi indices $\mathbf{v}$. Once we have a surrogate model for $\tilde{\Delta}_i$, we can compute $\mathbb{E}[\tilde{\Delta}_i]$ and $\mathrm{Var}[\tilde{\Delta}_i]$ for use in (5.1)-(5.3).

5.3. RESULTS

5.3.1. SPATIAL VARIABILITY OF UNCERTAINTY

We will now investigate the effect of injecting uncertainty via the approach described in the preceding section. In particular we will look at the spatial variability of the uncertainty estimate for several QoIs, first in the case of a NACA64A010 airfoil at an angle of attack of 6.2 degrees and a Mach number of 0.61. In this section we will combine just the predictions of the Spalart-Allmaras and Baldwin-Lomax model. Thus, for both models a surrogate model is created via (5.4)-(5.5). The QoI during the surrogate construction was the pressure coefficient, and in Figure 5.3 we show the obtained HDMR weights. Note that both models have a low effective dimension, since a second order expansion is sufficient.

The first and second order moments of the c_p distribution around the airfoil, computed by sampling the BMSA-HDMR model, are shown in Figure 5.4. From the mean field we observe that a shock is present near the leading edge of the airfoil, see Figure 5.4(a). We furthermore show the in-model, between-model and a total standard devi-



(a) HDMR weights α_i and $\alpha_{\mathbf{u}}$ for the Spalart-Allmaras model.

(b) HDMR weights α_i and $\alpha_{\mathbf{u}}$ for the Baldwin-Lomax model.

Figure 5.3: HDMR weights for the pressure coefficient c_p of the NACA64A010 airfoil at an angle of attack of 6.2 degrees and a Mach number of 0.61. For both cases a relative cut-off value of $\epsilon_1 = 0.9$ is chosen, such that the smallest combination of first-order weights α_i which make up more that 90 % is chosen for a second-order component function.

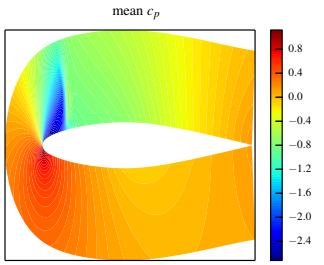
5

ation of (5.3) in Figures 5.4(b)-5.4(d). From these plots we can observe that by far the most dominant contribution to the uncertainty in the pressure coefficient comes from the in-model variance, indicating that coefficient uncertainty plays the most important role here.

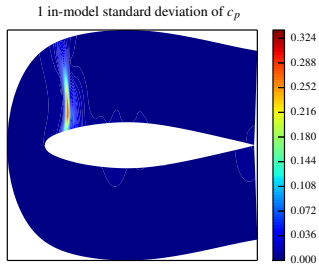
As already mentioned in Section 4.5.3, we can also use the computed code samples to create surrogate models for quantities other than the chosen QoI. Besides replacing the code samples with the other QoI, this also requires us to recompute the stencils S_j, the polynomial order p_j, and the LEC condition must be enforced once more. Afterwards we can again sample the BMSA-HDMR expansion (5.1) and compute the moments. We have done so for the Mach number M, the results of which are shown in Figure 5.5. Notice that in this case the between-model variance is a significant fraction of the total uncertainty. It acts in a different location than the in-model variance, mainly in the boundary layer just before the shock. The in-model variance is most significant directly at the shock location, see Figure 5.6 for a zoom-in on the two regions.

What becomes apparent from Figures 5.4 and 5.5, is that the injected uncertainty is highly localised in the spatial domain. As expected, at the shock location we find the most significant disagreement between the different models and coefficient values. This is in contrast to our boundary-layer predictions of Chapters 2 and 3, where the coefficients were influential throughout the entire considered domain. Thus the posterior distributions, when propagated through the models, act as automatic markers which identify regions where the models cannot be trusted.

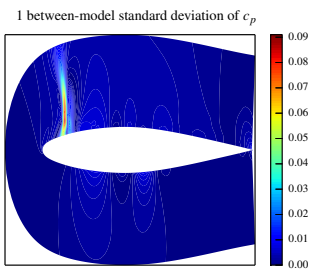
We will also show the results for a more extreme flow case, namely the transonic flow at a Mach number of 0.8 and an angle of attack of 3.5 degrees. The mean predictions of c_p and M with the same three uncertainty estimates are displayed in Figures 5.7 and 5.8. The main difference in the results compared to the previous flow case is the severity of the uncertainty. A stronger shock is formed, which increases the magnitude of both the coefficient and model uncertainty. The shock location also becomes more uncertain,



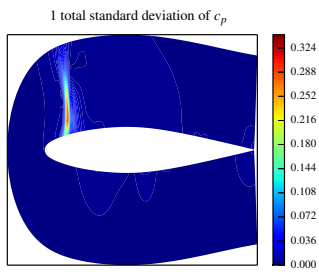
(a) The mean field



(b) One in-model standard deviation

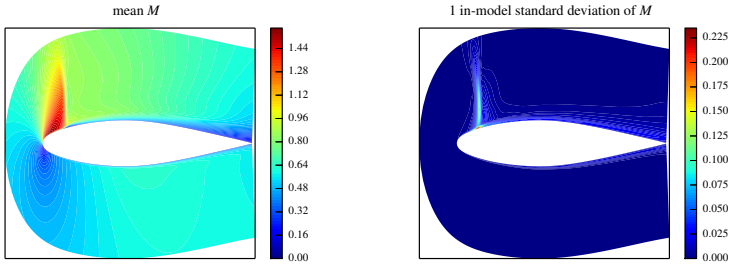


(c) One between-model standard deviation.



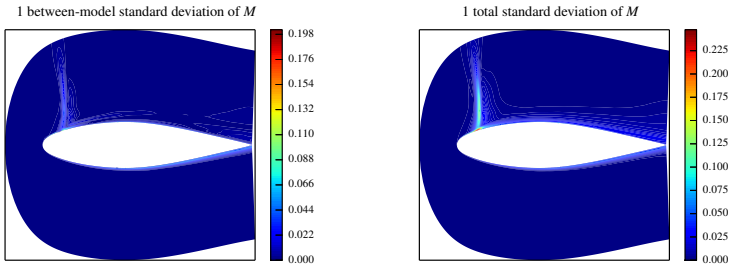
(d) One total standard deviation, defined by (5.3)

Figure 5.4: The statistics for c_p computed from a BMSA-HDMR expansion with Spalart-Allmaras and Baldwin-Lomax. The angle of attack is 6.2 degrees and the free-stream Mach number is 0.61.



(a) The mean field.

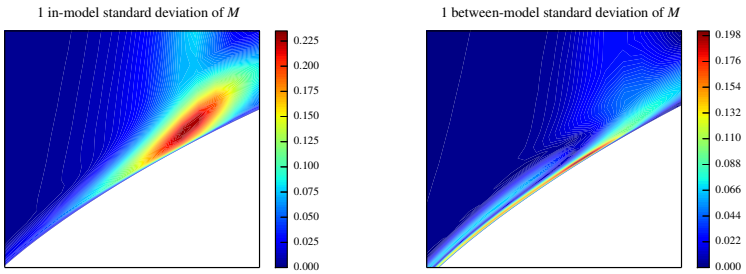
(b) One in-model standard deviation.



(c) One between-model standard deviation.

(d) One total standard deviation, defined by (5.3)

Figure 5.5: The statistics for the Mach number M computed from a BMSA-HDMR expansion with Spalart-Allmaras and Baldwin-Lomax. The angle of attack is 6.2 degrees and the free-stream Mach number is 0.61.

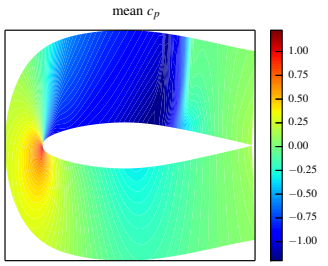


(a) One in-model standard deviation.

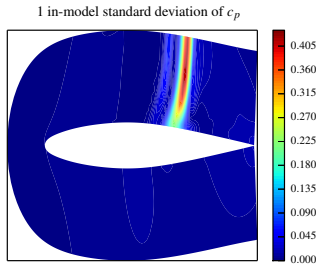
(b) One between-model standard deviation.

Figure 5.6: A zoom-in on the regions of most significant uncertainty due to the in-model (left) and between model (right) variance.

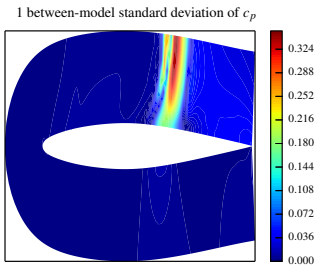
and thus the uncertainty is injected into a larger spatial domain. For both QoIs, the between-model variance plays a significant role. Also the in-model and between model variance have a larger overlap, which results in a larger total uncertainty estimate. Like the previous flow case, the variance of c_p in Figures 5.7(b)-5.7(d) is most severe inside the shock and away from the wall. The variance of the Mach number shows a slightly different spatial distribution, see Figures 5.8(b)-5.8(d). Still at the shock location, but close to the wall is the prediction most uncertain.



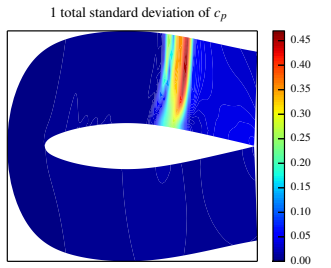
(a) The mean field



(b) One in-model standard deviation



(c) One between-model standard deviation.

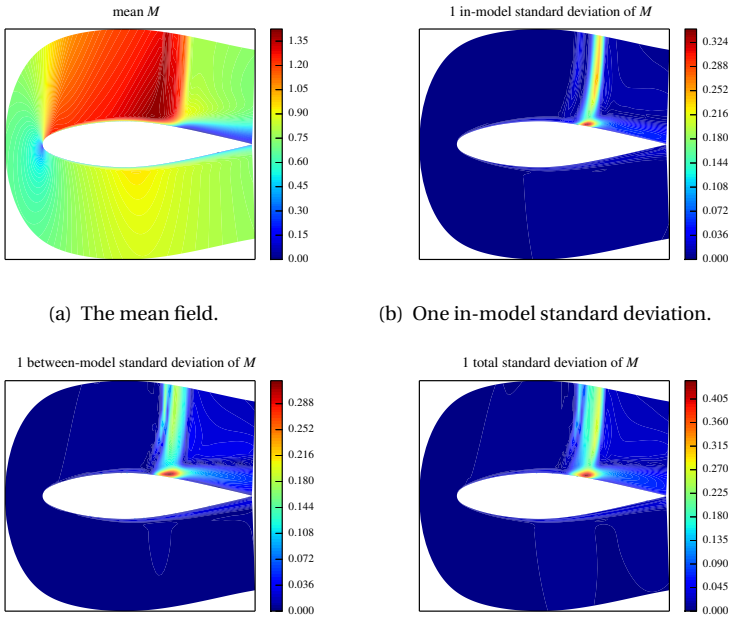


(d) One total standard deviation, defined by (5.3)

Figure 5.7: The statistics for c_p computed from a BMSA-HDMR expansion with Spalart-Allmaras and Baldwin-Lomax. The angle of attack is 3.5 degrees and the free-stream Mach number is 0.80.

5.3.2. COMPARISON TO VALIDATION DATA

We have experimental validation data for the NACA64A010 airfoil. From [2] we have the c_p distributions for a wide variety of angles of attack and Mach numbers. The authors of this reference report errors in c_p up to ± 0.012 due to wind tunnel boundaries. Furthermore, the maximum c_p errors due to misalignment in the angle of attack is reported to be ± 0.01. Also errors of $\pm 0.1 \times 10^6$ in the Reynolds number and small errors in M (± 0.005) are mentioned, but how these errors affect c_p is not specified. However, small errors in the Mach number can have a large effect on the pressure distribution, especially in the suction peak. See for instance Figure 5.9, which shows experimentally determined pres-



(a) The mean field.

(b) One in-model standard deviation.

(c) One between-model standard deviation.

(d) One total standard deviation, defined by (5.3).

Figure 5.8: The statistics for the Mach number M computed from a BMSA-HDMR expansion with Spalart-Allmaras and Baldwin-Lomax. The angle of attack is 3.5 degrees and the free-stream Mach number is 0.80.

sure profiles for the NACA64A010 airfoil, at a variety of closely grouped Mach numbers. In this figure we see that a change of 0.01 in the Mach number can shift the suction peak by roughly 0.1. We therefore take a conservative experimental error estimate of ±0.1.

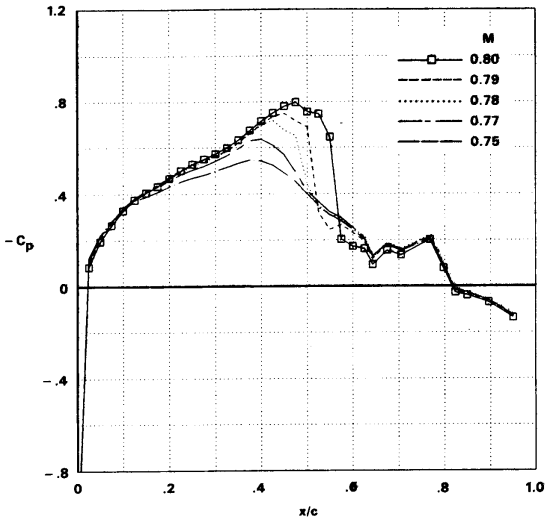


Figure 5.9: Experimentally determined upper-surface c_p distributions for several closely grouped Mach numbers. Notice that the small changes in M have a significant effect on the peak suction. The considered airfoil was the NACA64A010 at a zero angle of attack and a Reynolds number of $6.5 \cdot 10^6$. Source [1].

In Figure 5.10 we show c_p at the airfoil surface computed with BMSA-HDMR expansion at a free-stream Mach number of 0.61 and $\alpha = 6.2°$. This expansion contains predictions from the Launder-Sharma, Spalart-Allmaras and Baldwin-Lomax models. The experimental data from [2] is plotted as well. We again see that the only region where significant uncertainty is predicted is at the shock location. In other parts there is little variance and the prediction is consistent with the experimental data.

5.4. CONCLUSION

To conclude, with an efficient surrogate modelling technique, our initial results indicate that BMSA can be used to measure the effect of different sources of uncertainty on a predictive flow case. However, more validation flow cases will be performed to confirm this. Also, future work will investigate the effect of expert opinion in the construction of BSA posteriors, and means to eliminate expert opinion altogether, see Section 6.2 for a discussion.

REFERENCES

[1] D. Buell and G. Malcolm. An experimental investigation of the aerodynamics of a naca 64 a 010 airfoil-flap combination with and without flap oscillations. part 1:

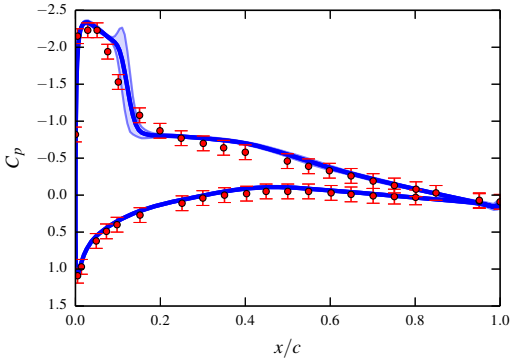


Figure 5.10: The BMSA-HDMR prediction with $\pm 3\sqrt{\text{Var}[\Delta|\mathbf{z}]}$ uncertainty bounds of c_p at the wall, and the validation data.

Steady-state characteristics. *NASA Technical memorandum 88217*, 1986.

[2] L.S. Stivers, H. Press, and C.C. Shufflebarger. *Effects of Subsonic Mach Number on the Forces and Pressure Distributions on Four NACA 64A-series Airfoil Sections at Angles of Attack as High as 28 Degrees.* National Advisory Committee for Aeronautics, 1954.

6

CONCLUSIONS AND RECOMMENDATIONS

In Section 6.1 we will give the conclusions regarding the uncertainty quantification methodologies developed in this thesis. In the following section we give our recommendations for future research directions.

6.1. CONCLUSIONS

6.1.1. PARAMETER AND MODEL VARIABILITY

Our method of uncertainty quantification consists of two main phases, i.e. the calibration and a prediction phase. In the former, we have performed multiple Bayesian calibrations on five turbulence models from different classifications, i.e.

- Algebraic models: Baldwin-Lomax
- One-equation models: Spalart-Allmaras
- Two-equation models: Launder-Sharma $k-\varepsilon$ and Wilcox (2006) $k-\omega$
- Reynolds Stress models: Willcox Stress-ω model

We found that the coefficients show a large degree of variability when calibrated against different flow scenarios, in our case different turbulent boundary layers. Each calibration was successful, since the calibrated fit moved close to the data, when the prediction with standard coefficient values left room for improvement. The model with the least observed variability in the coefficients, and thus the most 'flow-independent coefficient values', was the Spalart Allmaras model. However, this does not necessarily mean that the Spalart-Allmaras model is the best model in our set, since its accuracy is still dependent on the flow scenario to which it is applied. These results challenge the traditional approach in turbulence modelling, where the closure coefficients are calibrated once on elementary flow problems in a deterministic fashion. Afterwards the

obtained point estimates are often assumed to be universal. The fact that we observe such a large variability in posterior coefficient values clearly indicates that no universal closure coefficient values exist.

The same type of variability is also observed in the posterior model probabilities, which can be interpreted as a measure of consistency that a model has with the experimental data. For all five models, applied to all 14 different boundary-layer flow scenarios, we computed these probabilities. Again, the observed variability was large, and which model is 'most-likely' according to the posterior model probability metric can change significantly from one flow scenario to the next. Hence, this underlines the notion that also no universal 'best' turbulence model exists. Although the scenario dependent performance of turbulence models is already a known fact, computing posterior model probabilities for several flow scenarios quantifies the extend of this variability. Furthermore, these probabilities can be used in the predictive phase to weight different model predictions.

The fact that both the closure coefficients and the turbulence model itself are uncertain is a clear indication for the need of uncertainty quantification for turbulent flow simulations.

6.1.2. STATISTICAL MODEL-INADEQUACY TERM

A well-known way to account for the uncertainty introduced due to imperfect (physical) models is to include a statistical model-inadequacy term to the output of the simulation code. We have done so for all the Bayesian calibrations that were performed. These model inadequacy terms contain hyper parameters that must be calibrated alongside the coefficients from the simulation code. Examining the posterior values of the hyper parameters reveals information on the severity of the model inadequacy in the flow scenario under consideration. Even so, there are drawbacks to model inadequacy terms applied to the code output, i.e.

1. Model inadequacy terms must be calibrated, and so the posterior hyper parameters are only an indication of the model inadequacy in the calibration scenario. And since we have already seen a large variability of performance over different scenarios, the added value of this information is questionable.

2. Model inadequacy terms are topology dependent. The term used in this thesis is designed for use in boundary layers, and so its posterior distributions can only be used to predict other boundary layer flows.

To move around these drawbacks, we only use the posterior distributions of the closure coefficients in the predictive phase. Although they are calibrated as well, they are topology independent. Furthermore, because we used a fast boundary-layer code, posterior distributions could be obtained quickly. By combining posterior closure coefficients from multiple models, a measure of model inadequacy in the predictive phase can be obtained (see Section 6.1.4). The effect of the model inadequacy term is still implicitly present, since it has an influence in the shape of the posterior closure coefficient distributions.

6.1.3. P-BOXES

One of the methods we used to combine multiple posterior distributions in the predictive phase was through probability boxes. Although we showed that in our case the p-boxes were consistent with the experimental validation data, they over predict the amount of uncertainty. This is due to the fact that the p-boxes constitute an unweighted prediction, essentially saying that all posterior distributions are equally applicable to the predictive flow scenario at hand. This can lead to excessively large error bars. Hence, p-boxes are useful, but only in the case when a uniform distribution of the scenario probabilities is an acceptable choice.

6.1.4. BAYESIAN MODEL-SCENARIO AVERAGING

In our Bayesian Model-Scenario Averaging (BMSA) framework, we made a weighted prediction by propagating 65 individual posterior distributions from 5×13 model - scenario combinations through the computational model. This model was configured to predict some unmeasured quantity of interest. Afterwards, the distributions of the output were combined into a single predictive probability density function (pdf). Each individual output pdf must be weighted by the product a model and a scenario probability. The model weights were the posterior model probabilities, which were determined from the data in a separate Bayesian inference procedure. The scenario probabilities come into play during the predictive phase, and therefore they are not informed from the calibration data. In fact, during this phase there might be no experimental data at all. We therefore created an automatic scenario sensor based upon model agreement in each flow scenario. If within a given scenario there is a high level of agreement between the different turbulence models, this scenario is given a high probability and *vice versa*. We first tested this approach on 14 different predictive boundary layer configurations, and found that in each case the averaged prediction was significantly more robust than the individual model predictions. By 'robust' we mean a methodology that does not suffer from scenario dependent performance. Due to the automatic sensor, the predictions formed a good fit with the available validation data. This sensor also assured that, unlike the p-box predictions, the amount of uncertainty was not excessive.

Our posterior coefficient database is completely informed using boundary layer flows. To test the predictive capability of this database on another flow type we also made predictions for the skin friction. In this case, the automatic sensor lost some of its optimality, resulting in a prediction that was sightly off the validation data. The robust nature of the framework, i.e. a flow-scenario independent capability to provide good predictions, was still preserved.

Performing 65 individual propagations is not a problem when dealing with a cheap boundary layer code. However, with a more expensive RANS code this is unlikely to be feasible, even with the use of surrogate models. We therefore proposed new scenario-averaged posterior distributions. In this case the scenario probabilities are specified using expert-knowledge, after which we could combine all distributions belong to the same model. This led to just 5 distributions which had to be propagated, there by reducing the computational cost by a factor equal to the number of flow scenarios. However, even with these distributions, an efficient surrogate modelling technique is required in order to propagate the 5 remaining distributions through a complex and expensive simulation

code.

6.1.5. IMPROVED SIMPLEX-STOCHASTIC COLLOCATION METHOD

A state-of-the-art surrogate modelling technique is the Simplex Stochastic Collocation (SSC) method. It discritizes the stochastic space into disjoint elements, and it assigns a (high-order) interpolation stencil to each element. Furthermore, it suppresses the Runge phenomenon through the enforcement of the Local Extremum Conserving (LEC) limiter, giving it the ability to create surrogate models of discontinuous quantities of interest. However, the computational cost of the LEC limiter scales poorly with dimension. For a cheap test problem with 6 uncertain parameters (which runs in a couple of seconds), it took us 1000 minutes (16.7 hours) to construct a surrogate model, 75 % of which was spent on enforcing the LEC limiter. Since most turbulence models have anywhere from 5 to 10 uncertain parameters, we developed two new techniques for improving the computational cost, i.e.

1. Remove the overlap of interpolations stencils. Since each element has its own stencil, the overlap of stencils becomes very large in higher dimensions. We proposed the use of set-covering stencils, which is a relatively small set of stencils with the property that every element is covered just once. The result is that the number of times the LEC condition needs to be checked is reduced significantly.

2. Incorporate the SSC method into the High-Dimensional Model Reduction (HDMR) framework. In this framework a function is decomposed into an expansion of so-called component functions of increasing dimensions. Each component function represents the impact that a certain subset of parameters has on the computed output. In physical problems we are often confronted with a low effective dimension, meaning that only a few input parameters have a significant influence. By assigning appropriate weights the component functions containing only the influential parameters can be identified, after which only these functions are computed. Thus, the idea is to use the SSC method to create several low-dimensional surrogate models of influential component functions, instead of computing one high-dimensional problem. This avoids the bad scalability of the SSC method, and results in an efficient surrogate modelling technique where the dominating computational cost is sampling the simulation code, and not surrogate modelling technique itself.

Both methods results in a significant computational speed up. With the use of set-covering stencils, the same 6-dimensional problem mentioned earlier takes 23 minutes instead of 1000. However, the construction of the SC stencils still requires us to construct the SC stencils in a high dimensional space. This process will still scale badly with dimension, and therefore the curse of dimensionality is postponed rather than eliminated. The HDMR technique allows us to avoid sampling high-dimensional spaces all together, provided that the effective dimension of the problem is low.

6.1.6. BMSA APPLIED TO AN EXPENSIVE SIMULATION CODE

With the improved SSC method and scenario-averaged posteriors it is computationally feasible to make predictions with quantified uncertainty for an expensive simulation

code. We applied this methodology to transonic flows over the NACA64A010 airfoil. This results in a localised injection of uncertainty into the spatial flow domain. Mainly in the region of the shock we obtain significant measures of the uncertainty, where the magnitude of this uncertainty estimate increases with increasing shock strength. Outside the shock domain the uncertainty estimate is close to zero, and thus the posteriors, when propagated, act an automatic marker that identifies regions where the turbulence models cannot be trusted. We finally computed a BMSA prediction of the pressure distribution on the bottom and top of the airfoil, which we found to be consistent with available validation data.

6.2. RECOMMENDATIONS

As it currently stands, our BMSA framework has three disadvantages which require further research:

1. The database of posterior distributions is informed using experimental boundary layer data only. This leaves some coefficients uninformed in the majority of flow cases. Furthermore, one might reasonably expect that our posterior distributions will be suitable to predict for instance airfoil flows, but flow topologies which are radically different from flat plate boundary layers might require posterior distributions calibrated on other flow types.

2. To reduce the computational effort using scenario-averaged posteriors, expert knowledge is required to assign probabilities to the similarity of calibration scenarios to the predictive scenario. However, the assigned distribution will differ from one expert to the next. Moreover, if the prediction scenario is of a very different type than any of the calibration scenarios, assigning the probabilities might prove difficult.

3. The Boussinesq hypothesis is only (indirectly) challenged if stress-transport models are included in the chosen model set.

Regarding the first mentioned downside, we recommend to grow the database with posterior distributions coming from a larger range of flow topologies. It is expected that other coefficients will be informed when calibrating for instance on free-shear layer data. Also, this will give a wider range of posterior distributions to choose from in the predictive phase. The calibration can once again be performed using cheap simulation codes. However, by replacing the simulation code with a surrogate model, posterior distributions from more expensive codes can be added as well.

Secondly, to remove the need of relying on expert opinion, we recommend to investigate the use of Maximum A-Posteriori (MAP) estimates in place of full posterior distributions. In this case there is no need for uncertainty propagation. Instead, given I models and K scenarios, $I \times K$ deterministic codes solves have to be performed. Afterwards, one can again use model agreement as a measure for which scenarios to favour. Although this will not give predictive bounds due to parametric uncertainty, estimates of model inadequacy and scenario uncertainty can still be computed.

Also note that further data could possibly be used to improve the predictions. If scenario averaged posteriors are used, we have M surrogate models weighted by the product of scenario and posterior model probabilities. However, the model probabilities were

informed using the data from the calibration phase, and might not be optimal. If any experimental data becomes available, this can be used to update the posterior model probabilities using Bayesian inference and the M surrogate models. This experimental data does not have to be of the same type as the quantity of interest in order to do so. For instance, if some force coefficients are measured, these can be used to update the model probabilities, which might result in an improved predicted pressure distribution. Investigating the effectiveness of such an approach is a possible research topic.

The most complex case considered in this thesis is the turbulent flow over an airfoil. While of engineering interest, we also recommend to investigate means to extend the BMSA approach to more complex flow cases. A logical next step would be performing uncertainty quantification on the three-dimensional flow over a wing. The described approach with MAP estimates can be a option that is computationally feasible.

Finally, in order to challenge the Boussinesq hypothesis, we recommend to perform research into direct probabilistic perturbations of the Reynolds stresses. A starting point can be the work done by the authors of [1]. They applied deterministic perturbations directly into the magnitude, shape and orientation of the Reynolds stresses. For each perturbed Reynold stress tensor, the mean flow field was solved. The resulting envelope of model outputs form bounds on the prediction based upon physical possible states of turbulence. Although this approach goes beyond the Boussinesq hypothesis, it does have some downsides. Due to its deterministic nature, only bounds are computed, and there is no most-like prediction within those bounds. Furthermore, no use is made of experimental data, and the predictions cannot be updated if data becomes available. Therefore, we recommend to perform research into parameterising Reynolds-stress perturbation functions using random variables. Once a perturbation approach in the probabilistic domain is available, all the statistical tools developed in this thesis become available again.

REFERENCES

[1] M. Emory, J. Larsson, and G. Iaccarino. Modeling of structural uncertainties in reynolds-averaged navier-stokes closures. *Physics of Fluids (1994-present)*, 25(11):–, 2013.

A

FLOWCHART

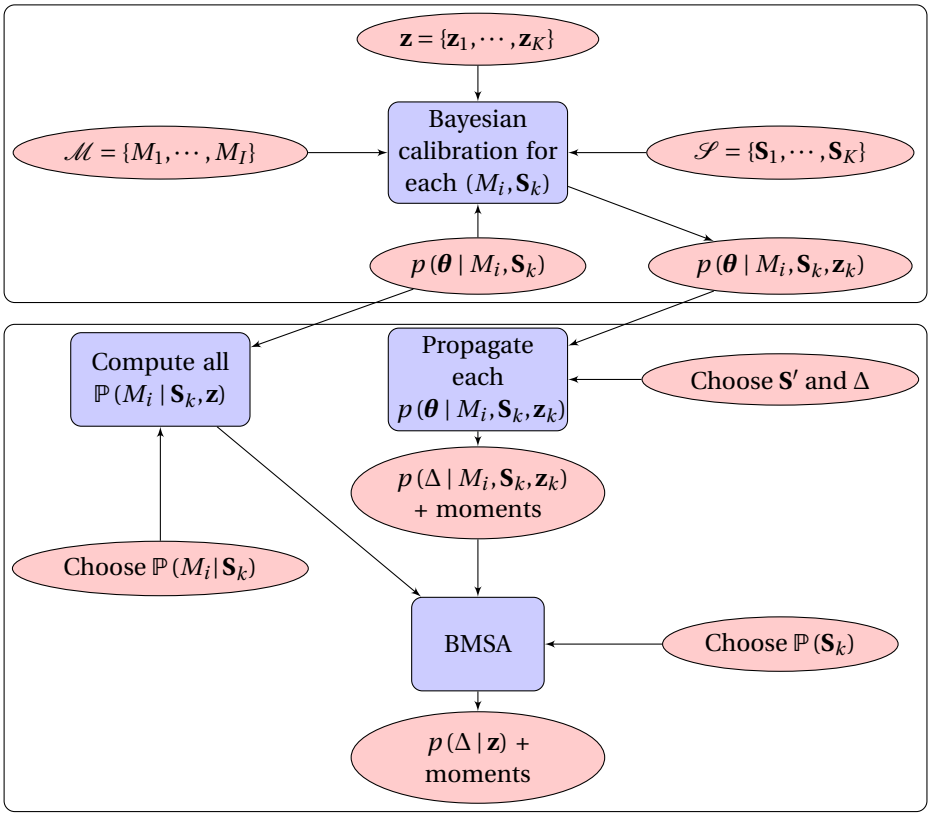


Figure A.1: Flowchart of the BMSA procedure.

B

BASELINE SSC ALGORITHM

This appendix provides the general pseudo code of the baseline SSC algorithm with ENO stencils.

Compute initial n_ξ-dimensional Delaunay grid
Compute initial code samples $\mathbf{v} = (v_0, \cdots, v_k, \cdots, v_{n_\xi})$ at the $2^{n_\xi} + 1$ grid points
Set initial hierarchical surplus errors to $\epsilon_k = -v_k$
$i \leftarrow 0$, choose max iterations I
while $i \leq I$ and $\hat{\epsilon}_{rms}$ > user-defined threshold **do**
    Determine p_{max} via (4.5)
    Compute nearest-neighbour stencils via (4.7)
    % Check (in parallel) the LEC condition:
    for $j = 1 \cdots, n_e$ **do**
        † Sample all $\Xi_j \in S_j$ via (4.25)
        At sample locations: compute w_j
        if (4.14) is violated in one or more sample locations **then**
            $p_j \leftarrow p_j - 1$
            Update S_j and goto †
        end if
    end for
    % Compute (in parallel) ENO stencils
    for $j = 1 \cdots, n_e$ **do**
        Collect the r nearest neighbour S_j that contain Ξ_j
        From these r stencils, select S_j^* := the S_j with the largest p_j
        if more than 1 S_j^* exists **then**
            Select the S_j^* with the smallest average distance to cell center of Ξ_j
        end if
    end for

Compute probabilities Ω_j and volumes $\bar{\Xi}_j$ via (4.10)
Compute refinement measures $\bar{e}_j$ via (4.9)
Sample the sub-simplices (4.13) of the N simplices with the largest $\bar{e}_j$ via (4.25)
Evaluate computer code at the N new locations, add code samples to $\mathbf{v}$
Compute the N new hierarchical surplus values via (4.15)
Refine the Delaunay grid by adding the N new sample locations
end while
Compute ENO stencils and check LEC condition from final iteration

C

SSC-SC ALGORITHM

Below we give the detailed construction of SCENO stencils in pseudo code, and we repeat some definitions for convenience.

- $\mathscr{C}$ is the set of simplices which are currently covered by a stencil S_j.
- $\mathscr{C}_j$ is the set of simplices currently under construction which will be added to $\mathscr{C}$.
- $\mathscr{N}_j$ is the set of all neighbouring simplices Ξ_k of simplex Ξ_j.
- $\mathscr{N}_{\mathscr{C}_j}$ is the set of all neighbouring simplices Ξ_k of all simplices in $\mathscr{C}_j$.
- $\{\boldsymbol{\xi}_k\}$ is the set of $n_\xi + 1$ vertices $\boldsymbol{\xi}_{k_{j,l}}$ that make up simplex Ξ_k.

% *for each iteration of the SSC-SC algorithm do*
compute $\mathscr{D}$ via (4.26)
$\mathscr{C} \leftarrow \mathscr{C} \cup \mathscr{D}$
%*select the simplex with the largest number of uncovered neighbours*
$\mathscr{I} \leftarrow \text{argmax}_{\Xi_j} = \{\Xi_j \mid |\mathscr{N}_j \cap \{\mathscr{U} \setminus \mathscr{C}\}| = \max|\mathscr{N}_j \cap \{\mathscr{U} \setminus \mathscr{C}\}|; j = 1, \cdots, n_e\}$
%*if there are multiple candidates, select the one with the largest volume*
if $|\mathscr{I}| > 1$ **then**
    $\Xi_j^* \leftarrow \text{argmax}_{\Xi_j} \bar{\Xi}_j = \{\Xi_j \mid \Xi_j \in \mathscr{I} \wedge \bar{\Xi}_j = \max \bar{\Xi}_j\}$
end if

⊛% *loop while* $\mathscr{C}$ *does not cover* $\mathscr{U}$
while $\mathscr{U} \nsubseteq \bigcup_{X_j \in \mathscr{C}} X_j$ **do**
    % *initialize all sets*
    $\mathscr{C}_j \leftarrow \{\Xi_j^*\}$

$S_j \leftarrow \{\boldsymbol{\xi}_{k_{j,l}} \mid \Xi_j \in \mathscr{C}_j\}$
$\mathscr{N}_{\mathscr{C}_j} \leftarrow \{\Xi_k \mid |\{\boldsymbol{\xi}_k\} \cap \{\boldsymbol{\xi}_j\}| = n_\xi + 1;\ \forall\, \Xi_j \in \mathscr{C}_j \wedge k = \{1, 2, \cdots, n_e\} \setminus \{j\}\}$

% *loop until no more uncovered neighbours are available or* S_j *is full*
while $\mathscr{N}_{\mathscr{C}_j} \setminus \{\mathscr{C} \cup \mathscr{C}_j\} \neq \emptyset$ and $|S_j| < N_j + 1$ **do**
% *update all sets*
$\mathscr{C}_j \leftarrow \mathscr{C}_j \cup \{\mathscr{N}_{\mathscr{C}_j} \setminus \{\mathscr{C} \cup \mathscr{C}_j\}\}$
$S_j \leftarrow \{\boldsymbol{\xi}_{k_{j,l}} \mid \Xi_j \in \mathscr{C}_j\}$
$\mathscr{N}_{\mathscr{C}_j} \leftarrow \{\Xi_k \mid |\{\boldsymbol{\xi}_k\} \cap \{\boldsymbol{\xi}_j\}| = n_\xi + 1;\ \forall\, \Xi_j \in \mathscr{C}_j \wedge k = \{1, 2, \cdots, n_e\} \setminus \{j\}\}$
end while

sort $\mathscr{C}_j$ and S_j according to $\|\boldsymbol{\xi}_{center_k} - \boldsymbol{\xi}_{center_j}\|_2$

if $|S_j| < N_j + 1$ **then**
reduce p_j to new maximum as allowed by (4.5)
end if

$\mathscr{C} \leftarrow \mathscr{C} \cup \mathscr{C}_j$
$\mathscr{S}_{sc} \leftarrow \mathscr{S}_{sc} \cup S_j$

%*if* $p_j = 1$ *is returned, all subsequent* $\mathscr{C}_j$ *will be linear as well* $\rightarrow$ *do not compute the rest*
if $p_j = 1$ **then**
%*manually set linear stencils for all remaining uncovered simplices*
for $\forall\, \Xi_j \in \mathscr{U} \setminus \mathscr{C}$ **do**
$\mathscr{C}_j \leftarrow \{\Xi_j\}$
$S_j \leftarrow \{\boldsymbol{\xi}_{k_{j,l}} \mid \Xi_j \in \mathscr{C}_j\}$
$p_j \leftarrow 1$
end for
end if
end while

%*check LEC condition for all* $S_j \in \mathscr{S}_{sc}$
if LEC is violated in any S_j **then**
$p_j \leftarrow p_j - 1$
update $\mathscr{C}_j$, S_j
update $\mathscr{C}$, $\mathscr{S}_{sc}$
goto $\circledast$
end if

D

HDMR ALGORITHM

The dimension-adaptive cut-HDMR of [1] coupled with the SSC method.

%*Initialize sets*
$\mathcal{V}_{imp} = \{\emptyset\}$, $\mathcal{V}_{comp} = \{\emptyset\}$, $\mathscr{R} = \{\emptyset\}$, $p = 1$

%*Compute the component functions of order* $p = 0$ *and* $p = 1$
Compute all sub problems $f^{(\mathbf{v})}$ in (4.36) using the SSC method. Stop when global RMS error measure (4.17) $< \epsilon$. Add all computed $\mathbf{u}$ to $\mathcal{V}_{comp}$.

Compute all first-order weights (4.38) α_i

if $\alpha_i > \epsilon_1$ **then**
    $\mathcal{V}_{imp} \leftarrow \mathcal{V}_{imp} \cup \{i\}$
end if

Add $\mathbf{u}$ with $|\mathbf{u}| = p$ and which satisfy admissibility condition to $\mathscr{R}$

while $\mathscr{R} \neq \{\emptyset\}$ and $\alpha_p > \epsilon_2$ **do**
    $p \leftarrow p + 1$
    Add $\mathbf{u}$ with $|\mathbf{u}| = p$ and which satisfy admissibility condition (4.41) to $\mathscr{R}$
    $\forall \mathbf{u} \in \mathscr{R}$, compute $f^{(\mathbf{v})}$ in (4.36) using SSC, stop when (4.17) $< \epsilon$, add all computed $\mathbf{u}$ to $\mathcal{V}_{comp}$.
    Compute weights $\alpha_{\mathbf{u}}$ (4.40)
    if $\alpha_{\mathbf{u}} > \epsilon_1$ **then**
        $\mathcal{V}_{imp} \leftarrow \mathcal{V}_{imp} \cup \{\mathbf{u}\}$
    end if
    $\mathscr{R} = \{\emptyset\}$, add $\mathbf{u}$ with $|\mathbf{u}| = p$ and which satisfy admissibility condition to $\mathscr{R}$
    Compute relative error measure α_p (4.43)
end while

REFERENCES

[1] Ma, Xiang and Zabaras, Nicholas, "An adaptive high-dimensional stochastic model representation technique for the solution of stochastic partial differential equations" Journal of Computational Physics, **229**, 3884–3915, 2010

E

PROOF OF UNIFORM DISTRIBUTION

Equation (4.25), repeated below for convenience, is used to map points from the hypercube $K_{n_\xi} := [0,1]^{n_\xi}$ to an arbitrary simplex Ξ described by the (unique) points $\boldsymbol{\xi}_i \in \mathbb{R}^{n_\xi}$, $i = 1, \cdots, n_\xi + 1$. The $R_j \in \mathbb{R}$ are n_ξ scalar i.i.d. random variables (r.v.'s) distributed uniformly as $\mathscr{U}[0,1]$, and describe a randomly picked point in K_{n_ξ}. This appendix contains the proof that (E.1) is distributed uniformly as well.

$$M_{n_\xi} = \boldsymbol{\xi}_0 + \sum_{i=1}^{n_\xi} \prod_{j=1}^{i} r_{n_\xi - j + 1}^{\frac{1}{n_\xi - j + 1}} (\boldsymbol{\xi}_i - \boldsymbol{\xi}_{i-1}), \tag{E.1}$$

From [1] we have the following theorem regarding the distribution of a transformation of random variables:

Theorem 3. *Consider the r.v.'s $R_1, \cdots, R_{n_\xi}$ with joint pdf $f_{R_1 \cdots R_{n_\xi}}$ positive and continuous on the set $A \subseteq \mathbb{R}^{n_\xi}$, and let $h_1, \cdots, h_{n_\xi}$ be real-valued transformations defined on A; that is, $h_1, \cdots, h_{n_\xi} \to \mathbb{R}$, and let B be the image of A under transformations $(h_1, \cdots, h_{n_\xi})$. Suppose that $(h_1, \cdots, h_{n_\xi})$ is one-to-one from A onto B. Thus, if we set $y_i = h_i(r_1, \cdots, r_{n_\xi})$, we can uniquely solve for r_i, $i = 1, \cdots, n_\xi : r_i = h_i^{-1}(y_1, \cdots, y_{n_\xi}), i = 1, \cdots, n_\xi$. Suppose further that the partial derivatives $\frac{\partial}{\partial y_j} h_i^{-1}$, $i, j = 1, \cdots, n_\xi$ exist and are continuous for $(y_1, \cdots, y_{n_\xi}) \in B$. Finally, suppose that the Jacobian*

$$J(y_1, \cdots, y_{n_\xi}) = \begin{bmatrix} \frac{\partial h_1^{-1}}{\partial y_1} & \cdots & \frac{\partial h_{n_\xi}^{-1}}{\partial y_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_1^{-1}}{\partial y_{n_\xi}} & \cdots & \frac{\partial h_{n_\xi}^{-1}}{\partial y_{n_\xi}} \end{bmatrix} \tag{E.2}$$

is $\neq 0$ on B. Then the joint pdf of the r.v.'s $Y_i = h_i(R_1, \cdots, R_{n_\xi}), i = 1, \cdots, n_\xi$, $f_{Y_1 \cdots Y_{n_\xi}}$, is given

by:

$$f_{Y_1\cdots Y_{n_\xi}} = \begin{cases} |\det J(y_1,\cdots,y_{n_\xi})| \cdot f_{R_1\cdots R_{n_\xi}}\left(h_1^{-1}(y_1,\cdots,y_{n_\xi}),\cdots,h_{n_\xi}^{-1}(y_1,\cdots,y_{n_\xi})\right), & (y_1,\cdots,y_{n_\xi}) \in B \\ 0 & \text{otherwise} \end{cases} \tag{E.3}$$

In our case A is the hypercube K_{n_ξ} and B is the target simplex Ξ. Also, we have uniform i.i.d. R_j such that $f_{R_1\cdots R_{n_\xi}} = f_{R_1} f_{R_2} \cdots f_{R_{n_\xi}} = 1$. Thus, in order for $f_{Y_1\cdots Y_{n_\xi}}$ to be uniform we need to show that $|J(y_1,\cdots,y_{n_\xi})|$ is a constant. Furthermore, since (E.1) maps to a simplex this constant must be equal to the reciprocal of the volume of the simplex.

To simplify the analysis we will consider the standard simplex with nodes $\boldsymbol{\xi}_0 = (0,0,\cdots,0)$, $\boldsymbol{\xi}_1 = (1,0,\cdots,0)$, $\boldsymbol{\xi}_2 = (0,1,\cdots,0)$ etc. To demonstrate the structure of (E.1), we provide a four-dimensional example here:

$$M_4 = \begin{bmatrix} \sqrt[4]{r_4} - \sqrt[4]{r_4}\sqrt[3]{r_3} \\ \sqrt[4]{r_4}\sqrt[3]{r_3} - \sqrt[4]{r_4}\sqrt[3]{r_3}\sqrt{r_2} \\ \sqrt[4]{r_4}\sqrt[3]{r_3}\sqrt{r_2} - \sqrt[4]{r_4}\sqrt[3]{r_3}\sqrt{r_2}r_1 \\ \sqrt[4]{r_4}\sqrt[3]{r_3}\sqrt{r_2}r_1 \end{bmatrix} \tag{E.4}$$

Each individual row of M_4 is a transformation function $y_i = h_i(r_1,\cdots,r_4)$, for which we can find the following inverse functions:

$$r_1 = \frac{y_4}{y_4+y_3}, r_2 = \frac{(y_4+y_3)^2}{(y_4+y_3+y_2)^2}, r_3 = \frac{(y_4+y_3+y_2)^3}{(y_1+y_4+y_3+y_2)^3}, r_4 = (y_1+y_4+y_3+y_2)^4. \tag{E.5}$$

Now we can compute the Jacobian matrix (E.2) as

$$J = \begin{bmatrix} 0 & 0 & -\frac{y_4}{(y_4+y_3)^2} & \frac{y_3}{(y_4+y_3)^2} \\ 0 & -2\frac{(y_4+y_3)^2}{(y_4+y_3+y_2)^3} & 2\frac{(y_4+y_3)y_2}{(y_4+y_3+y_2)^3} & 2\frac{(y_4+y_3)y_2}{(y_4+y_3+y_2)^3} \\ -3\frac{(y_4+y_3+y_2)^3}{(y_1+y_4+y_3+y_2)^4} & 3\frac{(y_4+y_3+y_2)^2 y_1}{(y_1+y_4+y_3+y_2)^4} & 3\frac{(y_4+y_3+y_2)^2 y_1}{(y_1+y_4+y_3+y_2)^4} & 3\frac{(y_4+y_3+y_2)^2 y_1}{(y_1+y_4+y_3+y_2)^4} \\ 4(y_1+y_4+y_3+y_2)^3 & 4(y_1+y_4+y_3+y_2)^3 & 4(y_1+y_4+y_3+y_2)^3 & 4(y_1+y_4+y_3+y_2)^3 \end{bmatrix}. \tag{E.6}$$

When we compute the determinant of J all the y_i terms drop out and we end up with $|\det J| = 24$. The values of $|\det J|$ for $n_\xi = 1,\cdots,8$ can be found in Table E.1. From these results it becomes clear that $f_{Y_1\cdots Y_{n_\xi}} = |\det J| = n_\xi!$. The volume $\bar{\Xi}$ of a simplex Ξ can be computed by

$$\bar{\Xi} = \frac{1}{n_\xi!}|\det(D)|, \quad D = \begin{bmatrix} \boldsymbol{\xi}_1 - \boldsymbol{\xi}_0 & \boldsymbol{\xi}_2 - \boldsymbol{\xi}_0 & \cdots & \boldsymbol{\xi}_{n_\xi+1} - \boldsymbol{\xi}_0 \end{bmatrix}, \tag{E.7}$$

which for the standard simplex reduces to $\bar{\Xi} = \frac{1}{n_\xi!}$. And thus we have

$$\int \cdots \int_{\Xi} f_{Y_1 \cdots Y_{n_\xi}} \, \mathrm{d}Y_1 \cdots \mathrm{d}Y_{n_\xi} = f_{Y_1 \cdots Y_{n_\xi}} \int \cdots \int_{\Xi} \mathrm{d}Y_1 \cdots \mathrm{d}Y_{n_\xi} = n_\xi! \cdot \frac{1}{n_\xi!} = 1, \tag{E.8}$$

which completes the proof.

Table E.1: The absolute value of the determinant of J as a function of n_ξ.

n_ξ	1	2	3	4	5	6	7	8
$\lvert \det J \rvert$	1	2	6	24	120	720	5040	40320

REFERENCES

[1] Roussas, G. G. "An introduction to probability and statistical inference" Academic Press, 2003

CURRICULUM VITÆ

Wouter Edeling was born on November 14, 1981 in Leiderdorp, Holland. After attending high school, he joined the royal Dutch army where he served as a private first class in the 106-th signal battalion. He was deployed on the SFOR-9 peace-keeping mission to Bosnia, and left the army after roughly 3 years of service. Afterwards, he finished his B.Eng from the applied university of INHOLLAND in 2009, doing the pre-master year of the TU Delft in parallel. In 2011 he graduated with distinction from Delft University of Technology on the topic 'Improved representations of unresolved scales using optimization techniques'.

On the 1st of December 2011, Wouter started a joint-PhD between the TU Delft and Arts et Métiers ParisTech, under the supervision of Richard Dwight and Hester Bijl in Delft and Paola Cinnella in Paris. The research was funded by the French Agence Nationale de la Recherche. The topic was the quantification of modelling uncertainties in turbulent flow simulations using the Bayesian statistical framework. During this period a novel probabilistic method capable of providing predictions of turbulent flow with quantified uncertainty was developed.

As of December 2014, Wouter is working as a post-doc at the TU Delft, again in collaboration with, and funded by the French partners. He can be contacted by email at wouteredeling@gmail.com.

List of Publications

Journal papers

- W.N. Edeling, P. Cinnella, R.P. Dwight and H. Bijl, *Bayesian estimates of the parameter variability in the $k-\varepsilon$ model*, Journal of Computational Physics, 258, 73–94, 2014.
- W.N. Edeling, P. Cinnella, R.P. Dwight, *Predictive RANS simulations via Bayesian Model-Scenario Averaging*, Journal of Computational Physics, 275, 65–91, 2014.
- L.Chen, S.J. Hulshoff, W.N. Edeling, *POD enriched boundary models and their optimal stabilisation*, International Journal for Numerical Methods in Fluids, 77, 92–107, 2015.
- W.N. Edeling, P. Cinnella, R.P. Dwight, *Improved Simplex-Stochastic Collocation Method For Higher Dimensional Uncertainty Quantification Problems.*, (submitted), 2014.

Lecture series

- W.N. Edeling, R.P. Dwight, P. Cinnella *Bayesian analysis of RANS closure models*, Lecture series on Uncertainty Quantification in Computational Fluid Dynamics STO-AVT-VKI Lecture Series 2013/14 – AVT 235, von Karman Institute for Fluid Dynamics

Conferences

- *Adaptive stochastic-collocation methods for high-dimensional uncertainty quantification problems*, Workshop on uncertainty quantification in computational fluid dynamics, May 26-27, Pisa, 2014 Italy.
- *Bayesian Model Average Estimates of Turbulence Closure Error*, SIAM Conference on Uncertainty Quantification, March 30 - April 3, 2014, Savannah, USA.
- *Capturing model error in RANS turbulence models*, International Workshop on Uncertainty Quantification in fluids simulation, 16-18 December, 2103, Bordeaux, France.
- *Bayesian Error Estimates for RANS Turbulence Models*, Euromech Colloquium 543: Quantification of uncertainties in modeling and predictive simulation of fluids, October 10-11, 2013, München, Germany.

- *Bayesian estimates of the parameter variability in turbulence models for the RANS equations*, 7th International Conference on Sensitivity Analysis of Model Output, July 1-4, 2013, Nice, France.
- *Towards estimating turbulence modelling discrepancy in RANS with Bayesian inference*, Uncertainty in Computer Models Conference, July 2-4, 2012, Sheffield, England.

QUANTIFICATION OF MODELLING UNCERTAINTIES IN TURBULENT FLOW SIMULATIONS

RÉSUMÉ : Le but de cette thèse est de faire des simulations prédictives à partir de modèles de turbulence de type RANS (Reynolds-Averaged Navier-Stokes). Ces simulations font l'objet d'un traitement systématique du modèle, de son incertitude et de leur propagation par le biais d'un modèle de calcul prédictif aux incertitudes quantifiées. Pour faire cela, nous utilisons le cadre robuste de la statistique Bayesienne. La première étape vers ce but a été d'obtenir une estimation de l'erreur de simulations RANS, pour une catégorie limitée de ecoulements. Nous avons recherché en particulier à estimer des incertitudes pour les coefficients du modele, pour des écoulements de parois en gradients favorable et défavorable. Dans le but d'estimer la propagation des coefficients qui reproduisent le plus précisemment ces types d'écoulements, nous avons étudié 13 configurations différentes de calibrations Bayesienne. Chaque calibration était associée à un gradient de pression spécifique grâce à un modèle statistique.

La variabilité est estimée par le recours à la calibration Bayesienne et confrontée aux mesures expérimentales de chaque scénario. Cependant, un scénario-modèle Bayesien moyen (BMSA) est ici utilisé pour faire correspondre les distributions a posteriori à un scénario (prédictif) non mesuré. Cette approche est une approche pondérée faisant appel aux probabilités des modèles de turbulence, déterminée par les données de calibration. Pour tous les scénarios de prédiction considérés, la déviation standard de l'estimation stochastique est consistante avec les mesures effectuées.

Finalement, nous avons appliqué le cadre BMSA à un écoulement transsonique autour d'un profil d'aile. Avec cet outil nous sommes maintenant capable de faire des simulations prédictives d'écoulements auparavant trop coûteux et offrant des incertitudes quantifiées selon les imperfections des différents modèles de turbulence.

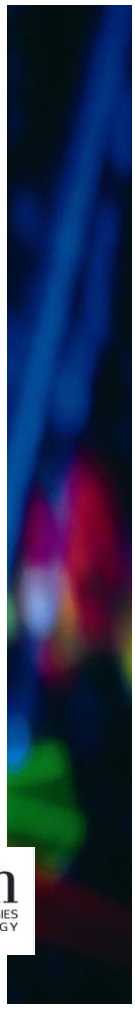
Mots clés : Reynolds-Averaged Navier-Stokes, statistique Bayesienne, estimation de l'erreur

ABSTRACT : The goal of this thesis is to make predictive simulations with Reynolds-Averaged Navier-Stokes (RANS) turbulence models, i.e. simulations with a systematic treatment of model and data uncertainties and their propagation through a computational model to produce predictions of quantities of interest with quantified uncertainty. To do so, we make use of the robust Bayesian statistical framework. The first step toward our goal concerned obtaining estimates for the error in RANS simulations, for a limited class of flows. In particular we searched for estimates grounded in uncertainties in the space of model closure coefficients, for wall-bounded flows at a variety of favourable and adverse pressure gradients. In order to estimate the spread of closure coefficients which reproduces these flows accurately, we performed 13 separate Bayesian calibrations. Each calibration was at a different pressure gradient, using measured boundary-layer velocity profiles, and a statistical model containing a multiplicative model inadequacy term in the solution space. The results are 13 joint posterior distributions over coefficients and hyper-parameters.

The variability is estimated using Bayesian calibration against experimental data for each scenario, and Bayesian Model-Scenario Averaging (BMSA) is used to collate the resulting posteriors in an unmeasured (prediction) scenario. This is a weighted approach involving turbulence model probabilities which are determined from the calibration data. The methodology was applied to the class of turbulent boundary-layers subject to various pressure gradients. For all considered prediction scenarios the standard-deviation of the stochastic estimate is consistent with the measurement ground truth.

Finally, we applied the BMSA framework to the transonic flow over an airfoil. With this we are able to make predictive simulations of computationally expensive flow problems with quantified uncertainty due to various imperfections in the turbulence models.

Keywords : Reynolds-Averaged Navier-Stokes, Bayesian Statistics, error estimation



ARTS ET MÉTIERS ParisTech

ParisTech INSTITUT DES SCIENCES ET TECHNOLOGIES PARIS INSTITUTE OF TECHNOLOGY