Surrogate models for the analysis of friction induced vibrations under uncertainty

Jérémy Sadet

▶ **To cite this version:**

Jérémy Sadet. Surrogate models for the analysis of friction induced vibrations under uncertainty. Mechanics of materials [physics.class-ph]. Université Polytechnique Hauts-de-France; Institut national des sciences appliquées Hauts-de-France, 2022. English. ⟨NNT : 2022UPHF0014⟩. ⟨tel-03778236⟩

HAL Id: tel-03778236

https://uphf.hal.science/tel-03778236v1

Submitted on 15 Sep 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons CC BY-NC-ND 4.0 - Attribution - Non-commercial use - No Derivative Works - International License

Thèse de doctorat

Pour obtenir le grade de Docteur de

l'UNIVERSITE POLYTECHNIQUE HAUTS-DE-FRANCE

et de l'INSA HAUTS-DE-FRANCE

Sciences pour l'ingénieur - Mecanique des solides, des matériaux, des structures et des surfaces

Informatique et applications - Sciences et Technologies de l'Information et de la Communication

Présentée et soutenue par Jérémy Sadet.

Le 20/06/2022, à Valenciennes

Ecole doctorale :

Ecole Doctorale Polytechnique Hauts-de-France (ED PHF n°635)

Unités de recherche :

Laboratoire d'Automatique, de Mécanique et d'Informatique Industrielles et Humaines (LAMIH - UMR CNRS 8201)

Centre de Recherche en Informatique, Signal et Automatique de Lille (CRIStAL - UMR CNRS 9189)

Numéro d'ordre : 22/17

Modèles de substitution pour l'analyse de problèmes de Vibrations induites par le frottement sous incertitudes

JURY

Présidente du jury : Bartoli, Nathalie. Maître de Recherche HDR. ONERA DTIS/M2CI.

Rapporteur : Hanss, Michael. Professeur des universités. ITM, Université de Stuttgart.

Rapporteur : Néron, David. Professeur des universités. LMPS, ENS Paris Saclay.

Examinateur : Sinou, Jean-Jacques. Professeur des universités. LTDS, Ecole Centrale de Lyon.

Co-directeur de thèse : Talbi, El-Ghazali. Professeur des universités. CRIStAL-INRIA, Université de Lille.

Co-directeur de thèse : Tison, Thierry. Professeur des universités. LAMIH, UPHF.

Co-encadrant : Massa, Franck. Maître de conférences. LAMIH, UPHF.

Co-encadrante : Turpin, Isabelle. Maître de conférences. CERAMATHS, UPHF.

This work is licensed under the Creative Commons Attribution International License (CC BY NC ND 2.0 FR). Please find more informations on https://creativecommons.org/licenses/by-nc-nd/2.0/fr/deed.en.

Remerciements

Le travail que représente la thèse de doctorat est rarement le fruit du travail d'une seule personne. Il est plutôt le fruit d'un travail d'équipe, et sans une équipe incroyable, le résultat ne serait pas à la hauteur des espérances. Une des plus grandes difficultés des remerciements est donc de réussir à mettre en exergue les contributions de chacun en les couchant par écrit.

Je tiens tout d'abord à témoigner de toute ma gratitude envers les professeurs El-Ghazali Talbi et Thierry Tison qui ont accepté ma candidature pour ce travail de recherche. El-Ghazali, merci de m'avoir emmené en conférence aux quatre coins du monde. Ces conférences m'ont permis de confronter mes travaux de recherche à des chercheurs mondialement connus, de rencontrer d'autres doctorants et chercheurs travaillant sur les mêmes axes de recherche que les miens et de surtout prendre du recul dans cette période incertaine. Thiery, merci d'avoir été présent dès que le besoin s'en faisait sentir. Vous avez toujours su trouver le bon équilibre entre mon besoin d'indépendence et mon besoin d'être cadré. Désolé pour tous ces vendredis soirs où je descendais dans votre bureau à 17h pour discuter boulot et qu'on repartait du laboratoire à 20h passé. Même dans les moments difficiles, vous avez quand même sû garder un peu de temps quand j'en ai eu le besoin.

Je souhaite également transmettre mes remerciements et ma gratitude à mon équipe encadrante : le Docteur Isabelle Turpin et le Docteur Franck Massa. Isabelle, merci de m'avoir remis à niveau sur les probabilités pour que je puisse m'attaquer à la compréhension du processus gaussien profond. Les maths m'avaient abandonné en licence et vous avez su me faire replonger dedans et à me faire apprécier les probabilités. Franck, merci d'avoir été présent pendant ces quatre longues années. Malgré des victoires et des défaites, vous m'avez toujours poussé à me raccrocher sur mes travaux, même après de longues périodes de désillusion. Votre soutien inébranlable a été une source d'inspiration tout au long de ma thèse.

Je tiens à remercier sincèrement les membres du jury qui ont accepté de juger de la pertinence de mes travaux. Les points que le jury a mis en avant, les remarques et questions évoquées durant la soutenance m'ont permis de prendre du recul sur mes travaux et de mettre en avant des pistes d'amélioration à explorer pour des travaux futurs. Plus précisément, un grand merci aux professeurs David Néron et Michael Hanss d'avoir accepté d'affronter la lecture de ce mémoire, disons-le, assez épais. Un grand merci également au professeur Jean-Jacques Sinou qui, malgré un empêchement de dernière minute, a pu être présent à ma soutenance. Les échanges que nous avons eu sur la réorganisation de modes et l'application de mes travaux dans un cadre industriel étaient très enrichissants. Enfin, je souhaiterais témoigner toute ma gratitude envers la Maître de recherche Nathalie Bartoli qui a accepté de présider le jury.

Une pensée à l'équipe d'organisation de Giens – Yassine, Julien, Max et Tuan – qui m'ont lâchement abandonné sur une île déserte en me tirant trois balles dans le dos. J'aurais ma vengeance à Galèrapagos : surveillez bien vos arrières. Merci aux collègues anciens doctorants et doctorants (Ali, Robin, Julie, Clément, François, Oussama, Xavier). Un remerciement tout particulier à Yassine avec

qui j'ai partagé les galères professionnelles sur le crissement pendant quatre années. Merci également à Catherine qui a supporté toutes mes discussions du matin sans rechigner et qui m'a énormément aidé pendant le déroulé de ma thèse.

J'aimerais remercier la famille du Summer Space Festival : Rémi, Lucia, Justine, Antoine, Charlotte, Lucas, Paul F., Paul C., Davia, Dorian, Bertrand, Lou, Morgane, Anaïs, Gabriel, Bram, Jean Emmanuel. Organiser cet évènement avec vous m'a apporté beaucoup et m'a permis de prendre un bon bol d'air frais tout au long de cette dernière année.

Ces remerciements ne seraient pas complets sans mentionner le soutien inébranlable de Stessy tout au long de ces quatre années malgré tout le temps que j'ai pu investir et toutes les difficultés que j'ai pu rencontrer. Ce titre, je le dois aussi à toi et à tous les sacrifices que tu as faits pour moi. Merci !

Après ces longs remerciements, je vous invite, lecteur ou lectrice, jeune padawan ou grand maître Jedi à tourner cette page et à vous perdre dans le monde passionnant, quoi que parfois déroutant, des méthodes numériques utilisés pour le crissement.

Une pensée est adressée aux simulations numériques perdues dans les méandres du Sino et qui hantent toujours les serveurs du laboratoire.

Contents

List of Figures

List of Tables

Introduction

« *Suddenly uncertainty was inside the citadel of mathematics itself, and it appeared irreducible.* »

— Michael Smithson

« *The most merciful thing in the world, I think, is the inability of the human mind to correlate all its contents.* »

— H.P Lovecraft

« *Avant tout, ne te trouble point ; tout arrive, en effet, conformément à la nature universelle, et sous peu de temps, tu ne seras plus rien, comme ne sont rien Hadrien et Auguste. Ensuite, fixant les yeux sur ce que tu as à faire, considère-le bien ; et, te souvenant qu'il faut être homme de bien et de ce que réclame la nature de l'homme, accomplis-le sans te détourner et de la façon qui t'apparaît la plus juste, mais que ce soit seulement avec bonne humeur, modestie et sans faux-semblant.* »

— M. Aurèle

1 The squeal and the associated problematics.

From many years, the squeal phenomenon has been a major issue for the transportation manufacturers (automotive, railway or aeronautics), as long as for the scientific community. Indeed, this elusive phenomenon, which generates sound levels above 100dB on a large frequency audible bandwidth, as emphasised in Fig. 1, is perceived by the vehicle purchasers as a poor-quality indicator. This causes an increase of the cost for the car manufacturers, due to client's claims, despite that the braking capabilities are not modified at all.

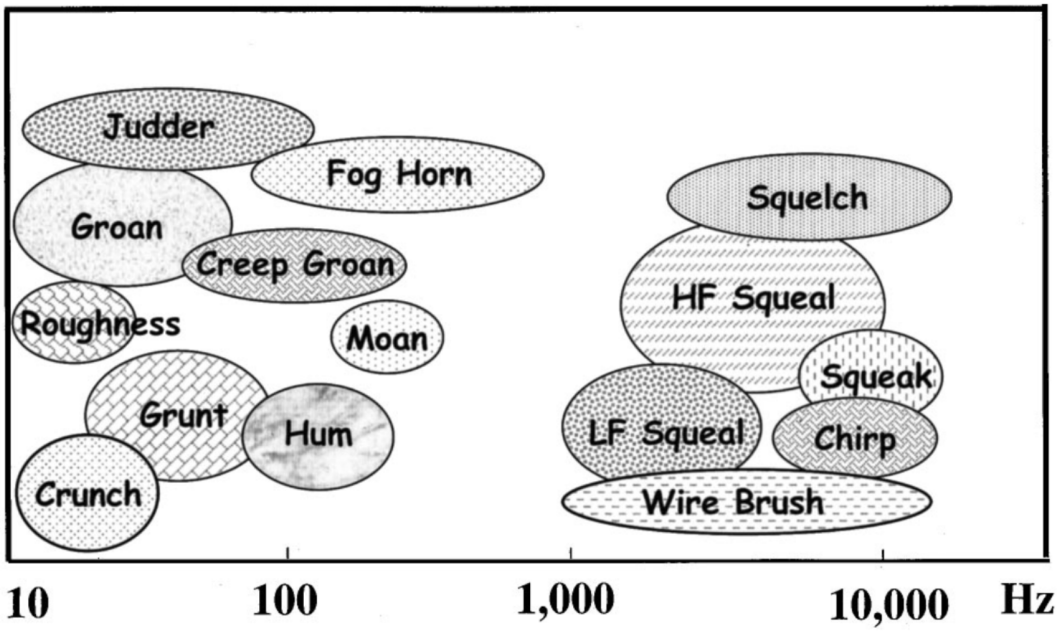


FIGURE 1: Classification of braking noises. [1]

This instability is characterised by oscillations of the system around an equilibrium position, which is unstable and whose vibration levels can reach a limit-cycle, generating the generation of an acoustic noise disturbance. The origin of these self-excited vibrations has been at the heart of the scientific literature on brake noise and friction-induced-vibration. The resulting scientific research highlighted that the emergence of the squeal noise is due to multi-level mechanisms – from the engineering to molecular scale (Fig. 2) – occurring at the contact interface. The researchers then linked the propensity of brake squeal with material, tribological and geometrical parameters.

In addition to these multiple origins, the squeal phenomenon is not deterministic and is difficult to reproduce experimentally. Conversely, it is highly fugitive, variable or even considered chaotic. Besides, this variability is existing and measurable within two braking systems of the same vehicle, as shown in Fig. 3. The quantification and propagation of the uncertainties is a key aspect of the current research to reproduce the phenomenon in numerical simulations.

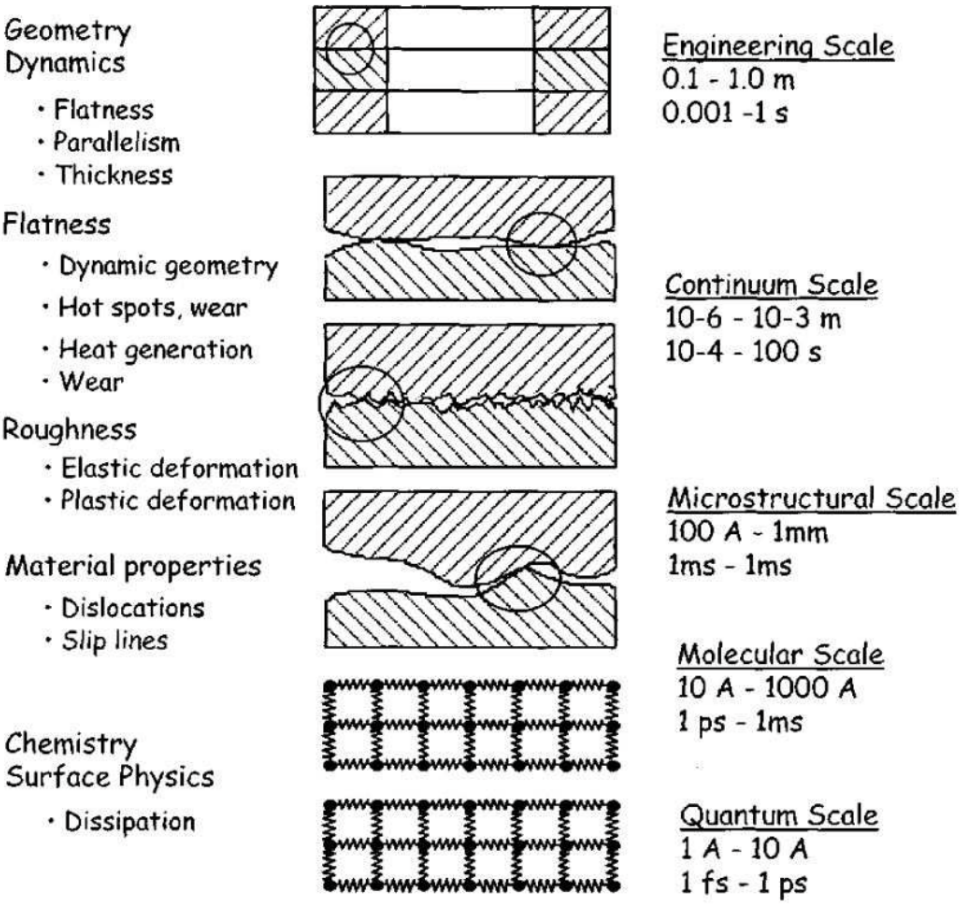


FIGURE 2: Phenomena emerging at the contact interface. [1]

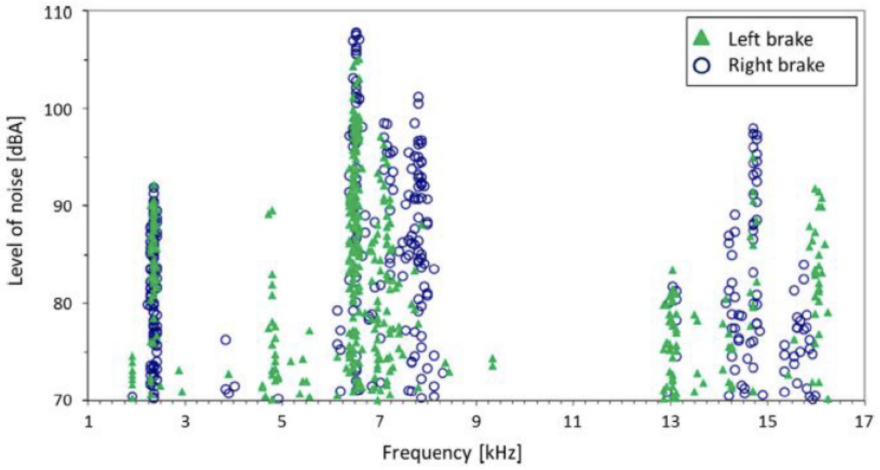


FIGURE 3: Acoustic level of a left and right brake systems within the same vehicle on a frequency bandwidth of 1 to 17kHz. [2]

The development of the computational power allows to numerically investigate the squeal phenomenon thanks to a digital twin, based on a finite element model of the studied system. To characterise the instabilities, two numerical methods are traditionally considered: the frequency and the temporal approaches. The first one consists in studying the system stability around an equilibrium

state by computing the eigensolutions of a quadratic eigenvalue problem. The analysis of the modes is carried out in the sens of Lyapunov. By examining the sign of the real part of the complex eigenvalue, the stability of the given mode can be determined. This approach is clearly interesting from a computational cost point of view, but was shown to be either under- or over- predictive. The second approach hinges on the time integration of the dynamics equation, while taking into account the material, geometrical and contact non-linearities. Unfortunately, the time integration is restricted to reduced brake systems because of its computational cost. It is thus difficult to use for industrial brake systems, with millions of degrees-of-freedom.

The frequency stability analysis was recently shown to produce promising results with the integration of the uncertainties into the simulation. Nonetheless, this increase of performance comes at a cost of more expensive simulations.

2 Thesis objectives

This thesis is a multidisciplinary work and is focusing on computer science and mechanics. In addition, the objective of this work is mainly dedicated to the development of new numerical strategies.

The **main objective** of this thesis is to suggest a new strategy of uncertainty propagation, for squeal numerical simulations, to maintain the numerical cost acceptable. To do so, three sub-objectives are pursued:

- The **precision of the solution** is a key factor to correctly estimate the uncertainties. The purpose is, here, to propose a new reduced order method to improve the computation of the eigensolutions of the system;
- The **low cost of the solution** is another key factor to propagate the uncertainties in a reasonable computation time. The aim is here to investigate surrogate modelling strategy, which allows reducing the computation cost by replacing the solver and approximating the solutions of the solver;
- The **efficiency of the uncertainty propagation step** is the final key factor to provide a good estimation of the uncertainties of the studied eigensolutions. The purpose is here to combine an optimisation strategy with the fuzzy set theory.

3 Structure of the manuscript

This research manuscript is decomposed into six chapters.

The **first chapter** presents a state-of-the-art of the brake squeal phenomenon, from the experimental investigations to study the emergence of the friction-induced-vibrations to the analytical or

numerical methods to compute the squeal phenomenon. This review allows positioning the proposed works in the squeal literature, especially for numerical methods, surrogate modelling, reduced order model, optimisation and uncertainty propagation.

The **second chapter** aims at increasing the quality of the solutions of a large quadratic eigenvalue problem. To do so, the traditional projection strategy is generalised with the addition of high-order homotopy modes. The proposed method is finally applied to solve a stability problem, considering a finite element model of a simplified brake system.

The **third chapter** assesses the performance of three surrogate models, namely a well-known Gaussian Process and two deep learning methods, currently used by the machine learning community. These approximation methods are used to replace the computation of the eigensolutions of a brake system by an expensive solver. First, the determination of the coalescence graph while only considering the friction coefficient allows accessing the capabilities of the different methods toward predictions of the coupling state. Second, a multiparametric analysis is performed to account for the performance of the Gaussian process models.

The **fourth chapter** is devoted to correct a deceptive prediction issue of the Gaussian Process, highlighted by the different assessments of the **third chapter**. The causes of this issue are determined, and a criterion is given to detect it independently of the dimensionality of the problem. A strategy to restore the prediction capabilities is also proposed.

The **fifth chapter** proposes a new optimisation strategy to overcome a potential issue of the traditional Bayesian Optimisation. A lack of exploration can arise in certain conditions, demonstrated in this chapter, which causes the workflow to be stuck in local optima. The new strategy hinges on a reformulation of the optimisation process into a multiobjective problem, which allows increasing the exploration capabilities.

The **sixth and final chapter** introduces a new combinatorial strategy to efficiently propagate the uncertainty by using a fuzzy formalism. Considering an α-cut discretisation of input fuzzy numbers, the Bayesian Optimisation process is integrated to compute the membership functions of the eigensolutions.

Chapter 1

State of the art for squeal phenomenon

Contents

1.1 The problematic of Brake

The invention of the wheel marks the beginning of the revolution of the objects motion. With this circular mechanical part, the terrestrial transportation of humans and goods really emerges. The basic idea behind this invention is to transfer the mechanical energy of a generator (originally, produced by humans or animals) to a given object by lowering the energy loss between those two entities: the mechanical energy is hence transformed into kinetic energy, while the energy loss corresponds to friction between the wheel and the road.

Quickly, the increase of velocity becomes an issue when the vehicle has to stop. The brake was invented to fulfill this purpose. The idea of a traditional brake system is to increase the friction to convert the kinetic energy into thermal energy. The transformation into thermal energy is the most common way to reduce velocity, but other types of energy are used, namely the electric energy (with the regenerative brake) and mechanical energy (with different mechanical systems which aims at rising the drag of the vehicle).

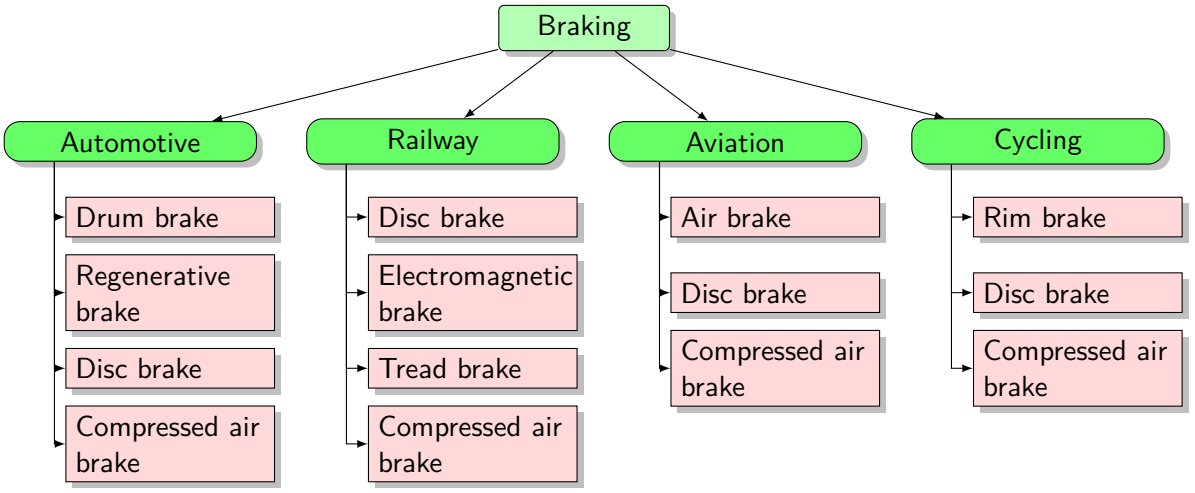
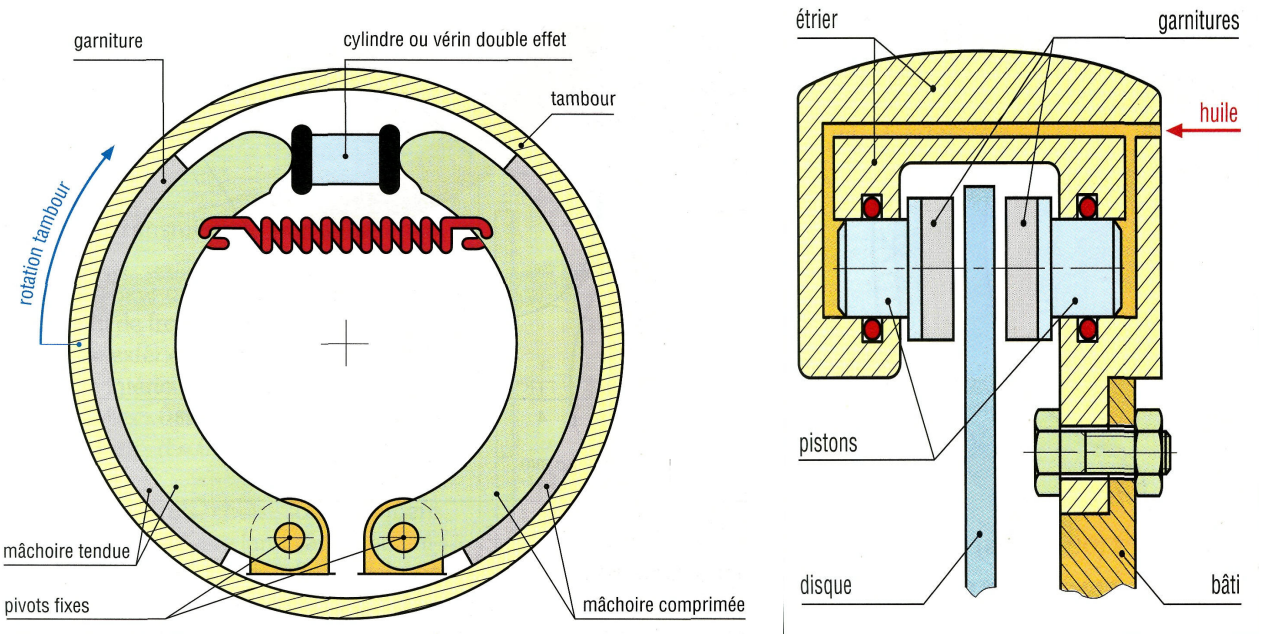


FIGURE 1.1: Braking technologies used for the four main types of transportation.

Fig. 1.1 suggests a classification of several types of brake systems, depending on the transportation field. This classification is not exhaustive, but, instead, it presents the main technologies used for braking. Usually, the compressed air brake is a mechanical system where the pressing of a pad or shoe is performed using compressed air. The air brake corresponds to all the elements that allow increasing the drag of the aircraft and spacecraft vehicles (parachute or spoilers).

Here, the focus is done on automotive vehicles. Commonly, two types of brakes are considered for the combustion engine: the drum and disc brakes. The regenerative brake is restricted to hybrid or electric cars. With regard to the drum brake, the mechanical system was used during the first part of the XX$^{\text{th}}$ century, in both aeronautics and automotive industry. This system, shown in Fig. 1.2(A), is composed of a drum (fixed to the wheel hub), two brake linings, lying on two lining tables, an actuator and a return spring. Typically, when the driver hits the brake pedal, the actuator pushes the lining tables and the brake linings come into contact with the drum, reducing the velocity of the vehicle.



(A) Drum brake (B) Disc brake

FIGURE 1.2: Diagram of a drum brake (A) and a disc brake (B).

This brake system is cheap to produce, which explains its success during the first part of the XX$^{\text{th}}$ century. Moreover, the system is closed, allowing a better tightness toward dust and water. This tightness induces a low resilience toward heat with high temperatures, which increases the wear of the brake lining. This issue, at first, caused little harm for the transportation fields but, quickly, with the velocity rising – first, in aeronautics, and then, in automotive industry – the need for a more sustainable mechanical systems was increasing.

The disc brake, shown in Fig. 1.2(B), was designed to improve the viability of the brake system. This system is composed of a caliper, a disc, two brake linings, fixed on a backplate, moved by pistons. Typically, when the driver hits the brake pedal, the pistons pushed the backplates, which moves the brake linings against the rotating disk, reducing the velocity of the vehicle.

This mechanical system has the main advantage to allow a better dissipation of the heat generated by the braking since the system is fully opened. Because of this lack of tightness, the disc brake is more fragile to water, reducing the efficiency of the braking during rainy days. Conversely, due to the rotation of the disk against the brake lining, the system is less sensitive to dust.

Nowadays, most of the cars are equipped with disc brakes due to its resistance toward heat. Nonetheless, a fair amount of small cars are still equipped with drum brakes.

1.2 Traditional methods to account for squeal in brake simulations

In practice, the kinetic energy is not fully converted into the thermal energy in all cases. In some peculiar conditions, some kinetic energy of the vehicle is transformed into vibrating energy: acoustic waves are then generated, resulting in a large variety of noises (denoted Friction-Induced Vibration noises, hereby denoted FIV). Comprehensive reviews of these phenomena can be found in [7–13].

To understand the FIV phenomenon, scientists focused their research on small lumped models [14]. The idea behind them is to simplify the associated physics to allow analytical computation of the solutions. Originally, they were the only way to numerically assess complex phenomena, but, with the increase of computer power, more and more computations were undergone through computers. Nowadays, these models are still used to allow fast evaluations of numerical methods.

With regard to squeal, these models have allowed to investigate three theories, trying to find explanations about the phenomenon. The first one, called stick-slip [15, 16], states that the squeal is associated to a decreasing friction coefficient depending on the sliding velocity. The model, composed of a mass in contact with a sliding band, switches between sticking phases and sliding phases. Spurr [17] introduced the sprag-slip, where the friction coefficient is not dependent on the sliding velocity. The causes of squeal is then explained by an accumulation of mechanical energy into the model. The given model is composed of a bar in contact with a sliding band. Due to the force applied on the bar, it bends and begins accumulating energy. Then, this energy goes beyond a threshold – which is dependent of the angle between the bar and the sliding band – and is released in a single moment. This alternance between those two types of events continues as the band is sliding. Finally, several authors [18–20] introduced new phenomenological models to describe a derivation of the sprag-slip theory, which is now the most favoured explanation for squeal phenomenon: the mode coupling.

To assess the squeal propensity of a brake system, several approaches exist and all consist in solving the dynamics equation, with or without some additional assumptions. The most commonly used strategy is the Complex Eigenvalue Analysis, hereby denoted CEA [21, 22]. It is a numerical method that relies on a linearisation of the nonlinear dynamics equation. Solving the resulting differential equation leads to the Quadratic Eigenvalue Problem (or QEP). This formulation of the linearised dynamics equation allows studying the stability of a given system by computing its eigensolutions. Then, these eigensolutions are analysed using the Lyapunov's theory, which states that the system is unstable if, at least, one eigenvalue has a positive real part.

The second way to address the squeal propensity of a brake system is to use direct time integration of the differential equations. The intuition behind this method is to discretise the dynamics equation along the time variable. To do so, several techniques are used [23–26], which falls within two categories: the explicit or implicit schemes [27]. The entire time history of the model dynamics with the transient and the stationary state can be determined by doing the time integration.

Blaschke et al. [28] emphasised that the CEA predicts more instable frequencies than really occurring on a brake system. AbuBakar and Ouyang [29] showed that there is a slight correlation between the predictions of both analyses, since the CEA succeeds in predicting one of the three unstable frequencies in the considered model. Massi et al. [30] compared both approaches with experimental results, and the experiment unstable modes are well-correlated with the transient and CEA results. Nonetheless, the CEA produces some additional unstable frequencies, which were not observed experimentally. Oberst and Lai [31] showed that CEA leads to an under-prediction of the unstable frequencies of the brake system. Finally, Sinou [32] highlighted that new harmonics can emerge during the transient analysis. These harmonics are not predictable with a CEA, due to the linearisation of the contact state during the first step of the computation. Hence, the CEA lacks in predictability efficiency. Conversely, the time integration is much more efficient, but it is not compatible with engineering time, whereas the CEA is.

1.3 Recent methods to account for squeal in brake simulations

1.3.1 Deterministic approaches

New methodologies have been developed to reduce the computational burden of the time integration of the dynamics equation, while providing a better analysis tool than the CEA. The first three methods hinge on the determination of only the stationary state of the time history of the brake system, while the latest improves the capabilities of the CEA.

The shooting method [33] is a time integration procedure, which supposes that the vibration response is periodic. From this assumption, the initial conditions and the periodic solutions (displacement and velocity) are iteratively updated until the convergence of the algorithm. As the studied phenomenon is subjected to self-excitated vibrations, the unstable frequency of the model is itself an unknown of the system. The determination of these solutions (periodic solutions and unstable frequency) method are highly expensive, especially for models with many degrees of freedom. Additionally, a good guess of the initial conditions has to be done to efficiently determine these solutions. Charroyer et al. [33] developed a new strategy to compute the initial conditions that showed good results for approximating the limit-cycles of a lumped model. Despite its computational attractiveness compared to time integration, the method remains highly expensive as pointed out by Denimal [34]. Moreover, another limitation of the method is that only the periodic vibrations (where one mode is unstable) is dealt with. The quasi-periodic vibrations scenario has not been assessed yet, and the experimentations carried out by Loyer [35] showed that the squeal phenomenon can be subjected to both periodic or quasi-periodic types of vibrations.

The Constrained Harmonic Balance Method (denoted CHBM) for periodic [36] and quasi-periodic [37] solutions also aims at determining the stationary regime of the considered model. To do so, the displacement and the forces are expressed as truncated Fourier series, by making the assumption

that the vibration state is periodic. As the vibrations are self-excited, the unstable frequency (or unstable frequencies, in the quasi-periodic case) is itself an unknown of the system and an additional assumption has to be carried out: the real part of the considered unstable frequency must be equal to 0. An algebraic system of equations is then solved with a Newton algorithm to determine the solutions of the system. One of the main limitations of the method is the dependency towards CEA. Indeed, the unstable modes are first determined with a CEA and fed in the algorithm. As highlighted previously, the CEA can be under- or over-predictive, meaning that some frequencies are detected unstable, whereas they are not. Injecting this information into the process yields a poor estimation of the limit-cycles. In addition, the complexity brought by the damping and friction generally causes the performance to be poorer.

The Modal Amplitude Stability Analysis, or MASA, [38], and its generalisation to quasi-periodic vibrations, the Generalised MASA [39], is a complementary tool to CEA. The idea is to assume that the contributions of an unstable mode on the vibration levels are determined by the first harmonics of its fundamental. In addition, the mode shape and angular frequency are constant, whatever the evolution of the dynamics response. When the first harmonics drives the dynamics response of the system, the method allows a good identification of the unstable modes, as well as the associated limit cycles. Nonetheless, the MASA is restricted to the analysis of periodic or quasi-periodic solutions [40]. Indeed, when the dynamic responses are driven by the other harmonics of the considered frequencies (meaning that the system dynamics is complex), the contribution of the unstable modes is badly estimated [34]. It is all the more problematic as the periodicity (or not) of the system dynamics cannot be known *a priori*.

Finally, the Improved Squeal Detection Methodology (or ISDM) hinges on the creation of mode families by considering the problem as non-probabilistic. In [41], the authors used a non-deterministic modelling of the contact interface to determine the most unstable frequencies. To do so, the random field theory is applied to model the height distribution of the surface nodes with a preliminary knowledge of two topographical parameters: the form and the waviness. Then, by using mathematical operations (KL expansion and Johnson transformation), the height of the surface nodes are obtained and integrated in the numerical models. As the surface nodes are probabilistic distribution, a given amount of realisations is computed to evaluate the uncertain eigensolutions. Nonetheless, the numerical cost associated with the use of non-deterministic methods can become prohibitive when the number of realisations used in ISDM is important.

Synthesis

Table 1.1 summarises the comparison between the different deterministic methods with respect to the cost, the predictability and the initialisation complexity of the methods.

As highlighted before, the CEA and time integration methods have important limitations in terms of respectively predictability and cost, justifying the needs for new methodologies. The CHBM and the (G-)MASA present interesting results, but a clear limitation of those methods is the initialisation

Table 1.1
Recent developments of deterministic methods.

		Response type	
Method	Ref.	Frequency	Stationary
Shooting method	[33]		✓
(C-)HBM	[36, 37]		✓
(G-)MASA	[38, 39]		✓
ISDM	[41]	✓	

of the methods. Conversely, the ISDM strategy presents an appealing compromise between cost and predictability, with an easier initialisation. In this following research, the aim is to suggest strategies to allow an improvement of both cost and predictability of the ISDM.

1.3.2 Approximation strategies

Reduced Order Models

The Reduced Order Model method (hereby denoted ROM) hinges on a projection on a smaller subspace. Traditionally, the projection on a subspace arises two times when solving a Friction-Induced-Vibration problem. The first is when the contact is taken into account in the equations, allowing expressing whether the nodes are in contact. The second corresponds to the reduction of the computational cost. Indeed, the numerical models used in Finite Element Method (or FEM) usually contain millions of degrees-of-freedom and this projection reduces the number of equations to roughly a hundred. In return, the solutions determined on this subspace are estimations of the true solutions. Several authors [42, 43] showed that in both cases if the reduced basis is a sufficiently good representation of the complete model, then the estimation would be efficient.

The Component Mode Synthesis strategy showed good performance to describe the full brake model, as shown in [43–45], but Loyer et al. [43] highlighted that the reduced basis must be selected with an extra care: the unstable modes, the stable modes near the unstable modes harmonics and combination of harmonics are not sufficient to provide a good reduction basis and other modes must be added. Consequently, the numerical cost of this strategy quickly expands. Recently, other authors [46, 47] introduced the Double Modal Synthesis method, which combines a traditional Modal Synthesis with a condensation of the contact interface. The results are in good agreement with the traditional strategy and the computational time is drastically reduced. Nonetheless, the convergence of the method has to be strictly monitored. Lai et al. [45] proposed a new method, the contact static approximation, which also allows a reduction of the computational time, while maintaining the prediction performance. However, the case application used by the authors does not allow a generalisation to other cases, since the nonlinear effects were directly integrated in the reduction basis. The performance of the method was not assessed if important nonlinear effects were present in the full model but not in the reduced one.

The reduction of model size with reduced basis in brake squeal is tackled by [44, 48]. Massa et al. [48] proposed a strategy to solve linear modal analysis, while Do et al. [44] suggested a method to carry out reanalysis problem. The main limitation of the first research is that it does not take into account the contribution of the damping in the system. The second research is especially suited for reducing the numerical cost of multi-parametric studies, but has too only been applied to linear modal analysis problem.

Surrogate modelling

The surrogate modelling principle hinges on replacing an expensive solver by a cheaper mathematical function, allowing faster evalutions with new parameter values. To do so, a set of parameter values, representative of the evolution of the considered solver, is evaluated to allow the surrogate model acquiring knowledge about the solver. Then, after training the model, predictions for non-evaluated sets of parameter values can be performed. Lü et al. [49] used regression to replace the solver and performed uncertainty propagation, similarly to Treimer et al. [50]. Similar studies were carried out with kriging surrogate model by [51–53]. Recently, Stender et al. [54] used Convolutional Neural Networks to automatically detect brake noises in experimental sound signals.

Nonetheless, one of the main issue of surrogate modelling is that the model is created on a training set which contains a limited amount of evaluated parameters values. Hence, if, for some parameters values, the approximated function present a different behaviour than for the already evaluated parameter values, then the surrogate model will obviously not be able to predict the behaviour of the solver.

Synthesis

Table 1.2 summarises the comparison between the different approximation methods with respect to the cost, the predictability and the application field of the methods.

Table 1.2
Summary of the approximation methods with regards to three criteria.

Method	References	Type	Application
Component Mode Synthesis	[43, 44]	ROM	Resolution
Double Modal Synthesis	[46, 47]	ROM	Resolution
Contact Static Approximation	[45]	ROM	Resolution
Homotopy perturbation	[48]	ROM	Resolution
CHOC	[44]	ROM	Reanalysis
Regression	[49, 50]	Surrogate	Resolution
Kriging	[51–53]	Surrogate	Resolution
Neural Network	[54]	Surrogate	Analysis

As highlighted previously, the contact reduction methods are interesting methods, but the complexity of construction of the reduction basis is important. Thus, no reduction of the contact interface is used in this research. With regard to the homotopy perturbation, the results provided by this

method are interesting but only available for modal analysis. Thus, this research is oriented towards a generalisation of this method. Finally, for the approximation methods, the reduction cost associated with the approximation of the solver is interesting. Nonetheless, a correct use of these methods can become tricky, depending on the parameter setting. Thus, this research focuses on the efficient use of this type of method.

1.3.3 Uncertainties

The uncertain nature of the parameters of brake systems was observed quite early in the literature [55, 56]. The origin of this nature is two-folds, as pointed out by Nobari et al. [53]. The first one is aleatory uncertainty and refers to variations caused by deviations from the nominal values of the studied properties. Typically, this type of uncertainty arises with material imperfections, component geometries tolerances and manufacturing processes. The second origin is labelled epistemic uncertainty, and is mostly caused by a lack of knowledge. This lack of knowledge can be alleviated by increasing the knowledge about the studied phenomenon, whilst the aleatory cannot. As pointed in several papers [57, 58], the friction and contact parameters are a major source of uncertainty and must necessarily be considered to represent their evolutions [59].

Recently, some authors [41, 58–60] highlighted that the accuracy of the CEA can be enhanced by considering the uncertainty into the squeal simulations. Ouyang et al. [61] described two types of uncertainty methods in their review: the stochastic and the deterministic ones. Nonetheless, in a broader sense, the methods are categorised as follows: probabilistic, non-probabilistic (or ensemblist) and hybrid uncertainty – the deterministic methods being a subset of the ensemblist ones.

Probabilistic methods

The probabilistic methods usually hinge on a description of the parameters of the brake system as random variables. The simplest approach is the crude Monte-Carlo Simulations (MCS) method [62], which involves many calls to the solver to use the central limit theorem. It consists of the evaluation of quantities of interest (mainly the complex eigenvalues calculated from the CEA) for a large sample of uncertain inputs and of the building of statistical estimators, such as the mean and the standard deviation. Indeed, Culla and Massi [63] used this approach to determine the probability density function of the output eigenvalues of a simplified brake model. In practice, the MCS approach is not usable since the squeal simulators are significantly expensive even though recent developments rely on parallel computations.

To allow an efficient computation of the output probability density function with an acceptable computational cost, alternative methods were developed, namely the perturbation methods, the Polynomial Chaos Expansions (PCE) and the non-parametric approach [64]. This last approach, based on a global uncertainty quantification and random matrices, was never used for squeal applications.

The perturbation methods are based on a first or second order Taylor series expansion on a set of zero mean random variables. This truncation is mainly due to difficulties associated to the high order derivatives evaluation. Thus, Butlin and Woodhouse [65] applied the 1st-order perturbation method to study the uncertainty of an FIV problem. However, the range of validity of this method is limited to applications with a small variation of random inputs.

The PCE, introduced in the field of structural dynamics by Ghanem and Spanos [66], is the best alternative to the MCS, and relies on the decomposition of quantities of interest on an orthogonal basis of polynomials. Several authors have investigated this way to calculate the random complex eigensolutions by taking two different types of approaches: the intrusive, which corresponds to a modification of the studied problem, and non-intrusive strategies, which considers the studied problem as a black-box function. Sarrouy et al [67] used an intrusive version by projecting the random eigenvalue problem associated to the brake squeal model onto the PC space. This Galerkin way gives rise to an augmented deterministic eigenvalue problem, allowing determining the unknown PCE coefficients. Moreover, to manage quantities whose variations with respect to the random parameters are not continuous, the authors proposed to divide the stochastic space and switch on the multi-element generalized polynomial chaos. The authors targeted in the analysis the evolution of random friction coefficient and the contact stiffness of the brake. An adaptation of this random strategy to CHBM was proposed by Sarrouy in [68]. Meanwhile, Zhang et al. [60, 69] used the PCE with a collocation method to determine the real parts of a brake system. Nechak et al. [70] investigated a non-intrusive version of the PCE to avoid the solving a large numerical problem associated to PCE coefficients identification. Indeed, the proposed solution hinges on a training set, as for surrogate modelling, to calculate the PCE coefficients. It allows taking into account uncertainty of the nonlinear static equilibrium when dealing with the stability analysis of nonlinear dynamical systems with random parameters. Recently, Nechak et al. [71] highlighted the robustness of GPC and Wiener-Haar stochastic expansions for the stability of uncertain nonlinear dynamical systems.

To model the corrected spatial variations, the use of independent random variables is not sufficient. Thus, Heussaff et al. [72], Tison et al. [41] and Renault et al. [2] successively investigated the random field theory for the modelling of the variability of brake linings. They progressively developed a complete method allowing building real pad topographies from an experimental database and analyse the effects on the frequency spectrum. The random field is the representation of the joint probability distribution for a set of random variables. To achieve this, a multiscale strategy is investigated and, each reconstructed topography relies on the adding of a random form and a waviness. The first scale is based on a polynomial form, which is defined with a set of random control points. For each of them, a probability density function of heights allows managing the whole of observed experimental behaviours. Next, the second scale is generated thanks to a collection of random fields generated using the Karhunen–Loève decomposition.

In the probabilistic methods, the uncertain parameters are treated as random variables whose probability distributions are defined unambiguously. To construct the precise probability distributions,

a large number of statistical information is necessary. To tackle this limitation, Lu et al. [73] proposed an imprecise probability approach based on evidence theory [74, 75] to model data imprecision, induced by a lack of or conflict information, observed in squeal applications. The proposed approach is established by integrating Taylor expansion, subinterval analysis and surrogate modelling. Thus, the uncertain parameters with imprecise data are treated as evidence variables, for which a belief measure and plausibility measure are employed to evaluate system squeal instability via the damping ratio.

Ensemblist methods

In the literature, the ensemblist methods refer to two main types of methods: the interval [76] and the fuzzy sets [77–79] theories. The first one describes the imperfection as an interval considering only the lower and upper bounds of variation for the quantity of interest. The latter one is an extension of the traditional set theory where any value of the non-deterministic parameters is linked to a membership degree, varying between 0 and 1, instead of either 0 or 1. Meanwhile, several methods allow propagating this variation of membership through the model to estimate the membership degree of the quantity of interest. This modelling allows a broader and much more precise description than the interval theory, while reducing the computational cost compared to a probabilistic method.

The reference method for fuzzy sets theory is the Zadeh's Extension Principle, or ZEP. Practically, this principle is not computationally acceptable since it produces an infinite number of combinations. To study the uncertainty of the quantities of interest, a common strategy is to discretise the membership degree into several values and to consider them as a collection of intervals. Hanss [80] proposed the Transformation Method, which consists in a specific Design Of Experiments (DOE). This approach is comparable to the MCS for the probabilistic formalism, and implies the same limitations in terms of computational time, especially when the number of parameters is important. Gauger et al. [81] used the fuzzy sets theory to compute the fuzzy eigenvalues of the complex eigenvalue problem, especially with a fuzzy friction coefficient and a fuzzy contact pressure. More recently, Hanselowski et al. [82] performed a comparison between frequency and transient analysis, whilst considering uncertain the stiffness, damping, friction coefficient and rotational velocity of the disk. By means of the Transformation Method, the fuzzy real part and maximum limit cycle were estimated, showing a strong correlation between the real part and growth rate of the temporal responses. Conversely, no correlation was identified with the maximum limit cycle, highlighting the complementarity of both strategies. To reduce computational cost for multiparametric studies, Massa et al. [83] used a fuzzy reanalysis technique, developed in [84] to estimate the fuzzy frequencies and growth rate of a friction-induced vibration problem as a function of fuzzy material, geometric and interface parameters. This method hinges on the fuzzy logic, fuzzy sets and interval theories. The proposed approach decomposes the fuzzy problem into interval problems and calculates interval output solutions by local min-max optimisations for each α-cut level. Each calculation of the problem output data, which is useful during the optimisation process, is reanalysed by integrating fuzzy logic controllers for the static step and homotopy development and projection techniques for the modal step.

Moreover, some authors highlighted that the fuzziness and interval uncertainty may exist in engineering problems simultaneously. An uncertainty quantification method of brake squeal instability involving two cases fuzzy–interval uncertainties was proposed by Lu et al. [85, 86]. The first fuzzy-interval case considers both fuzzy parameters and interval parameters exist simultaneously and independently in brake systems, whereas the second fuzzy-interval case is fuzzy-boundary interval uncertainty case. All uncertain parameters are expressed as interval variables, but their lower and upper bounds are treated as fuzzy variables rather than deterministic values due to subjective data. The uncertain response is computed with the aid of the combination of level-cut strategy, Taylor series expansion, subinterval analysis and Monte Carlo simulation to quantify the uncertainty in disc brake squeal analysis.

Hybrid methods

The last category is dedicated to the hybrid uncertainty approach, which consists in managing both probabilistic and non-probabilistic data. Indeed, Lu et al. [87] simultaneously considered interval and random variables. The parameters of brake pressure, densities of component materials, and thickness of backplate are treated as random variables; whereas the friction coefficient and Young's modulus of component materials are modelled as interval variables. A stability analysis of a brake system was then performed using the Response Surface Method (or RSM) combined with the CEA calculations to study the dominant unstable mode. This work was extended to random and fuzzy uncertainties by the same authors in [88]. Moreover, Lu et al. [89] extended the evidence theory to solve the hybrid uncertainty analysis problems with both random and interval variables. The effects of hybrid uncertainty on uncertain output are assessed by belief measure and plausibility measure, which is computed by subinterval perturbation technique. Finally, the proposed method is used to investigate the squeal analysis of automotive disc brakes involving both random (elastic modulus of friction lining) and interval properties (friction, contact stiffness).

Similarly, a recent work was proposed by Denimal et al. [90, 91], which consists in the combination of the PCE and kriging modelling. The idea is to associate both methods to give rise to a hybrid surrogate model. This hybrid surrogate model is able to take into account two sets of uncertain parameters and was used to predict the mode coupling instabilities for the Double Hulten phenomenological model. Recently, Denimal et al. [40] extended these works for investigating an uncertain brake squeal analysis. The authors considered a random friction coefficient managed by a PCE whereas the interval design parameters, namely some additional masses, are managed via kriging method. From the PCE coefficients, the evolution of the mean and the variance of each eigenvalue is studied, allowing obtaining a stochastic description of the system. Finally, the authors discussed the mode tracking procedure, which is a crucial step for the definition of an efficient training set necessary to build the surrogate models.

1.4 Chapter outcomes

The purpose of this chapter was to **describe the numerical simulation of the squeal phenomenon and the recent developments of new numerical methods available in the literature** to account for this phenomenon. As the squeal is highly fugitive, numerous experimentations were carried out to identify the sources of the phenomenon and allow better numerical simulations.

Traditionally, **two types of methods are usually considered accounting for the squeal of brake systems: the frequency and temporal analyses**. Some studies showed that the **frequency analysis can be under- or over-predictive**, while the **temporal analysis owns a high computational cost**, and cannot be considered by the industrialists.

Hence, the researchers investigated better strategies to either improve the representativity of the frequency analysis or reduce the computational burden of the simulation. The **most promising strategy is the ISDM, since it shows better performance with the prediction of unstable frequencies and is compatible with the industrialist's procedure of using frequency analysis** for computational reasons. Nonetheless, this method needs multiple calls to the solver to propagate the uncertainty in the brake model, which increases the computational burden (while still being cheaper than a temporal analysis).

To reduce this numerical burden, **some authors used surrogate modelling to replace the solver by a cheaper mathematical model**. Nonetheless, this method has rarely been used with the traditional uncertainty propagation methods. The following chapters will be aiming at the development of several numerical methods to perform efficient uncertainty propagation.

Chapter 2

Homotopy perturbation technique for improving solutions of large quadratic eigenvalue problem

Contents

2.1 Introduction

As highlighted by the previous chapter, to study the instabilities of a brake system, a well-used strategy in the industry is to determine the eigensolutions of the considered system by solving the QEP. Due to many degrees of freedom, the large discrete models require a projection onto a relevant subspace. Many strategies (sub-structuring [92], component mode synthesis [92], condensation [93], reanalysis techniques [94] and so on) are based on the construction of reduced models by means of projection on a basis of displacements. These methods are known to provide accurate results when the reduced subspace is a close approximation of the full-order model [95].

In modal analysis, the classical method relies on building a basis with a set of eigenvectors of the associated undamped model. For complex applications (vibroacoustics, fluid dynamics or FIV, for instance), this basis is not generally suitable and can lead to significant errors. For example, one of the reasons for potential errors in FIV problems is that the coupled normal modes are a poor representation of the model dynamics subjected to friction.

In addition, for the aforementioned techniques, increasing the subspace size with additional vectors of the same type slowly improves the solutions quality, making it not computationally attractive. Other subspace strategies were investigated in other contexts such as Taylor series [96], perturbation techniques [97–100], Padé approximants [101, 102] or Homotopy Perturbation Method (or HPM) [103].

At first, the HPM was used in the literature to calculate nonlinear solutions for different applications [104–107]. Duigou et al. [108] associated homotopy, asymptotic numerical techniques and Padé approximants to investigate the vibrations of damped sandwich structures, whereas Boumediene et al. [109] developed a reduction method for solving the complex nonlinear eigenvalue regarding viscoelastic structures. Sun et al. [110] provided numerical solutions for sound-propagation computations using the homotopy method. Next, Lee et al. [111] searched for asymptotic solutions of nonlinear problems with the HPM to yield natural frequencies.

Secondly, several authors considered the HPM to estimate modified solutions of a perturbed problem by reanalysis or by ROM. Indeed, Sliva et al. [112] suggested solving the new eigenvalue problem with the combination of a homotopy transformation and the perturbation method. Next, Li et al. [113] worked on the computation of complex eigenpairs of modified asymmetric systems as a way of creating a Campbell diagram of the modified rotor system. Massa et al. [114] integrated two-level homotopy techniques to estimate instabilities of a friction-induced vibration problem. Do et al. [115] used the HPM to reanalyse the modal bases of modified components of a mechanical system before using them in the component modal synthesis. More recently, Massa et al. [48] defined a ROM for a linear modal analysis by considering a projection basis made of high-order perturbed eigenvectors.

The purpose of this chapter is to show the limitations of the current subspace strategy and to describe a novel method to better approximate the eigensolutions of the QEP. To do so, the traditional

projection is enhanced with high-order modes, determined with the HPM. An algorithm is then suggested as a generalisation of the traditional method to deal with the QEP.

To highlight the capabilities of the new suggested method, this chapter is organised as follows. Section 2.2 directly jumps into the presentation of the common and the new subspace on which the system of equations is projected. In addition, the suggested algorithm is described, as well as its key parameters. Then, Section 2.3 proposes different assessments where the key parameters of the new method are studied. This section is ended by an application of this method for an approximation of a wide spectrum of the model. A simplified brake system is considered for the application of the different assessments of the proposal.

2.2 Subspace projection bases

2.2.1 Complex Eigenvalue Analysis

This approach aims at solving the dynamics equation (Eq. 2.1) and is divided into two steps: first, an equilibrium position is determined through an implicit solver; then, an eigenvalue analysis is computed on the linearised problem.

$$\mathbf{M}\ddot{\mathbf{U}} + \mathbf{C}\dot{\mathbf{U}} + \mathbf{K}_\Sigma\mathbf{U} = \mathbf{F}_{NL}(\mathbf{U}) + \mathbf{F} \tag{2.1}$$

where $\mathbf{M}$, $\mathbf{C}$ and $\mathbf{K}_\Sigma$ are respectively the mass, damping and stiffness matrices of the system. $\ddot{\mathbf{U}}$, $\dot{\mathbf{U}}$ and $\mathbf{U}$ are respectively the acceleration, the velocity and the displacements of the system nodes and $\mathbf{F}_{NL}$, the nonlinear forces of the system and $\mathbf{F}$ are the external forces.

The determination of the equilibrium position hinges on the decomposition of the displacement $\mathbf{U}$ into a mean value $\mathbf{u}_0$ and a small perturbation $\delta\mathbf{u}$, given by (Eq. 2.2).

$$\mathbf{U} = \mathbf{u}_0 + \delta\mathbf{u} \tag{2.2}$$

This mean position $\mathbf{u}_0$ is the static equilibrium position and can be computed from the static problem (Eq. 2.3).

$$\mathbf{K}_\Sigma\mathbf{u_0} + \mathbf{F}_{NL}(\mathbf{u_0}) = \mathbf{F} \tag{2.3}$$

In addition, by assuming that $\mathbf{F}_{NL}$ is of class C^1 over $\mathbb{R}$ and since the pertubation $\delta\mathbf{u}$ is considered small, the Taylor's theorem can be used to develop the nonlinear contact loads around the static displacement $\mathbf{u}_0$ into a first order polynomial.

$$\mathbf{F}_{\mathrm{NL}}\left(\mathbf{u}_0+\delta\mathbf{u}\right)=\mathbf{F}_{\mathrm{NL}}\left(\mathbf{u}_0\right)+\delta\mathbf{F}_{\mathrm{NL}}\left(\mathbf{u}_0\right)\delta\mathbf{u}+\mathrm{o}((\mathbf{u}_0-\delta\mathbf{u})^2) \tag{2.4}$$

Neglecting the rest of the Taylor series and substituting (Eq. 2.4) and (Eq. 2.2) in (Eq. 2.1), the governing equation is rewritten as follows.

$$\mathbf{M}\ddot{\delta\mathbf{u}}+\mathbf{C}\dot{\delta\mathbf{u}}+\mathbf{K}_\Sigma\delta\mathbf{u}+\mathbf{K}_\Sigma\mathbf{u_0}+\mathbf{F}_{NL}(\mathbf{u_0})+\delta\mathbf{F}_{\mathrm{NL}}\left(\mathbf{u}_0\right)\delta\mathbf{u}=\mathbf{F} \tag{2.5}$$

Finally, given (Eq. 2.3), the linearised form (Eq. 2.5) is simplified.

$$\mathbf{M}\ddot{\delta\mathbf{u}}+\mathbf{C}\dot{\delta\mathbf{u}}+\left(\mathbf{K}_\Sigma+\delta\mathbf{F}_{\mathrm{NL}}\left(\mathbf{u}_0\right)\right)\delta\mathbf{u}=0 \tag{2.6}$$

The dependence of (Eq. 2.6) over the contact state is explicit with the matrix $\delta\mathbf{F}_{\mathrm{NL}}\left(\mathbf{u}_0\right)$, which itself depends on the equilibrium position $\mathbf{u}_0$. A slight change of the contact surfaces of the brake pads and the disk might cause a completely different equilibrium position.

The solution of a homogeneous second order differential equation is given by (Eq. 2.7).

$$\delta\mathbf{u}=\mathbf{\Psi}\exp\left(\boldsymbol{\lambda}t\right) \tag{2.7}$$

Substituting (Eq. 2.7) in (Eq. 2.6) gives the Quadratic Eigenvalue Problem, where the quantity of interest are the eigenvalues λ_i and the eigenvectors $\mathbf{\Psi}_i$.

$$\left(\lambda_i^2\mathbf{M}+\lambda_i\mathbf{C}+\mathbf{K}\right)\mathbf{\Psi}_i=\mathbf{0} \tag{2.8}$$

where $\mathbf{K}=\left(\mathbf{K}_\Sigma+\delta\mathbf{F}_{\mathrm{NL}}\left(\mathbf{u}_0\right)\right)$

A wide variety of problems can be tackled using the QEP, given by (Eq. 2.8). When the three given matrices are Hermitian positive definite, or positive semidefinite for $\mathbf{C}$ and $\mathbf{K}$, (Eq. 2.8) corresponds to a structure modeled with viscous damping or a damped mechanical system where components are only coupled by the normal contact entries. The eigenvalues of such problems are real or complex conjugate and come in pairs. For gyroscopic systems, $\mathbf{M}$ and $\mathbf{K}$ are Hermitian, with $\mathbf{M}$ positive definite, and $\mathbf{C}$ skew-Hermitian ($\mathbf{C}=-\mathbf{C}^H$, with $\mathbf{C}^H$, the conjugate transpose of $\mathbf{C}$). In this case, the eigenvalues are purely imaginary. When the mechanical system is subjected to friction (such as FIV problems), the tangential loads induce a non-symmetric stiffness matrix, resulting from the tangential coupling. Even if the damping is not considered, complex solutions of the latter QEP can reveal positive real parts. Other types of problems, such as vibroacoustic, fluid mechanics, control theory and even numerical optimisation can also be reduced to a QEP. For a complete description of those applications, the interested readers can refer to the paper of Tisseur et al. [116].

The traditional strategy to solve the QEP is to linearise the problem by writing it in a 2n space. By doing so, the Generalised Eigenvalue Problem (hereby denoted GEP) is emphased (Eq. 2.9).

$$\mathbf{A}\mathbf{u}_i = \lambda_i \mathbf{B}\mathbf{u}_i \tag{2.9}$$

where $\mathbf{u}_i = \begin{pmatrix} \boldsymbol{\psi}_i \\ \lambda_i \boldsymbol{\psi}_i \end{pmatrix}$, λ_i and ψ_i being the eigensolutions of (Eq. 2.8).

The most common choice of linearisation is the first companion form, given by (Eq. 2.10) ; or the second companion form, given by (Eq. 2.11).

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{N} \\ -\mathbf{K} & -\mathbf{C} \end{bmatrix}; \mathbf{B} = \begin{bmatrix} \mathbf{N} & \mathbf{0} \\ \mathbf{0} & \mathbf{M} \end{bmatrix} \tag{2.10}$$

$$\mathbf{A} = \begin{bmatrix} -\mathbf{K} & \mathbf{0} \\ \mathbf{0} & \mathbf{N} \end{bmatrix}; \mathbf{B} = \begin{bmatrix} \mathbf{C} & \mathbf{M} \\ \mathbf{N} & \mathbf{0} \end{bmatrix} \tag{2.11}$$

where $\mathbf{N}$ can be any nonsingular matrix $n \times n$.

The choice between those two forms depends on the nonsingularity of the mass and stiffness matrices. Typically, in modal analysis, the second companion form (Eq. 2.11) is considered with a matrix $\mathbf{N}$ equal to the mass matrix $\mathbf{M}$. It allows conserving both the symmetry of $\mathbf{A}$ and $\mathbf{B}$ when the mass and stiffness are symmetric.

With this linearisation, the problem is written in a form that the algorithms are capable of dealing with (namely, the QZ algorithm [117]). Nonetheless, for large dimension problems, when matrix rank is greater than 1000, these algorithms cannot solve (Eq. 2.9) because of the numerical cost.

To sort out this issue, a common approach is to project the matrices of the GEP onto a subspace $\mathbf{T}$ to reduce the size of the problem. Considering this projection, the matrices $\mathbf{A}$, $\mathbf{B}$ and the vector $\mathbf{u}_i$ take the form of (Eq. 2.12).

$$\mathbf{A} = \begin{bmatrix} -\mathbf{T}^T\mathbf{K}\mathbf{T} & \mathbf{0} \\ \mathbf{0} & \mathbf{T}^T\mathbf{M}\mathbf{T} \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} \mathbf{T}^T\mathbf{C}\mathbf{T} & \mathbf{T}^T\mathbf{M}\mathbf{T} \\ \mathbf{T}^T\mathbf{M}\mathbf{T} & \mathbf{0} \end{bmatrix} \quad \mathbf{u}_i = \begin{pmatrix} \boldsymbol{q}_i \\ s_i^* \boldsymbol{q}_i \end{pmatrix} \tag{2.12}$$

where $\boldsymbol{\psi}_i^* = \mathbf{T}\boldsymbol{q}_i$.

Since the eigensolutions are determined in a subspace, $\mathbf{s}^*$ and $\boldsymbol{\psi}^*$ are the approximations of the true eigensolutions.

2.2.2 Normal mode subspace

In practice, the subspace $\mathbf{T}$, used for the projection, is composed of the first normal modes of the associated conservative system given by the Linear Eigenvalue Problem (Eq. 2.13), also denoted LEP.

These eigenvectors can be computed with a classical eigenvalue algorithm, for example, a Lanczos solver.

$$\left(\mathbf{K}_S - \omega_i^2 \mathbf{M}_S\right) \boldsymbol{\phi}_i = \mathbf{0} \tag{2.13}$$

Both matrices $\mathbf{M}_S$ and $\mathbf{K}_S$ are obtained by considering the mass and stiffness of the system components. Only the normal contributions of the contact interface are considered when solving the LEP.

The projection basis $\mathbf{T}$ is built with n_m normal modes of (Eq. 2.14).

$$\mathbf{T} = [\boldsymbol{\phi}_1 \ \ldots \ \boldsymbol{\phi}_{n_m}] \tag{2.14}$$

The above strategy is common and largely implemented in commercial finite element software. In most of the practical problems, particularly the modal analysis with viscous damping, the approximate solutions of (Eq. 2.12) are very satisfactory.

When studying the stability of rubbing systems, some discrepancies arise, due to the absence of asymmetric contributions in the projection basis $\mathbf{T}$. These errors clearly alter the judgment of the behaviour of the mechanical system and can be reduced by adding appropriate modes, calculated from the HPM presented in the next section.

2.2.3 Homotopy subspace

The main idea behind HPM is to combine the standard homotopy theory in topology and the perturbation technique. To do so, the linear and non-linear parts of the problem are separated and an embedding parameter ε is introduced to highlight the non-linearity. Assuming that this additional parameter is small, the approximate solutions of the considered equations can be written as a power series in ε.

Homotopy decomposition

Considering the general case of the QEP, the basic idea of the suggested methodology is to introduce the HPM in two ways. The first one concerns the symmetric and asymmetric contributions of the mass and stiffness matrices. The decomposition in (Eq. 2.15) represents the symmetric contribution – the S index – and the asymmetric contribution – the A index. The decomposition is here presented for both mass and stiffness matrices for the sake of generality. Nonetheless, the asymmetric contribution can easily be suppressed in the case of a purely symmetric matrix.

$$\begin{aligned}\mathbf{M} &= \mathbf{M}_S + \varepsilon\mathbf{M}_A \\ \mathbf{K} &= \mathbf{K}_S + \varepsilon\mathbf{K}_A\end{aligned} \tag{2.15}$$

The second decomposition is related to the damping, which is completely considered a non-linear part of (Eq. 2.16). The decomposition of the damping matrix $\mathbf{C}$ is then different as all entries are considered asymmetric contributions, such as:

$$\mathbf{C} = \varepsilon\mathbf{C}_A \tag{2.16}$$

According to the HPM, the i^{th} eigensolutions of (Eq. 2.8) are developed as a power series in ε with a nominal term – the (0) index – and higher order contributions – indexed by (n) – and (d), the order of truncation, such as:

$$\begin{aligned}\lambda_i &= \lambda_i^{(0)} + \varepsilon\lambda_i^{(1)} + \ldots + \varepsilon^n\lambda_i^{(n)} + \ldots + \varepsilon^d\lambda_i^{(d)} \\ \boldsymbol{\psi}_i &= \boldsymbol{\psi}_i^{(0)} + \varepsilon\boldsymbol{\psi}_i^{(1)} + \ldots + \varepsilon^n\boldsymbol{\psi}_i^{(n)} + \ldots + \varepsilon^d\boldsymbol{\psi}_i^{(d)}\end{aligned} \tag{2.17}$$

Substituting Eq.(2.17), Eq.(2.16) and Eq.(2.15) into Eq.(2.8), the QEP is rewritten as follows:

$$\begin{aligned}&\Big[(\mathbf{K}_S + \varepsilon\mathbf{K}_A) + \left(\lambda_i^{(0)} + \varepsilon\lambda_i^{(1)} + \ldots + \varepsilon^n\lambda_i^{(n)} + \ldots + \varepsilon^d\lambda_i^{(d)}\right)\varepsilon\mathbf{C}_A \\ &\quad + \left(\lambda_i^{(0)} + \varepsilon\lambda_i^{(1)} + \ldots + \varepsilon^n\lambda_i^{(n)} + \ldots + \varepsilon^d\lambda_i^{(d)}\right)^2(\mathbf{M}_S + \varepsilon\mathbf{M}_A)\Big] \\ &\qquad \left(\boldsymbol{\psi}_i^{(0)} + \varepsilon\boldsymbol{\psi}_i^{(1)} + \ldots + \varepsilon^n\boldsymbol{\psi}_i^{(n)} + \ldots + \varepsilon^d\boldsymbol{\psi}_i^{(d)}\right) = 0\end{aligned} \tag{2.18}$$

From (Eq. 2.18), it is possible to identify the n^{th} homotopy order and the high-order vectors associated with that order. The next subsections describe the form of the nominal (0^{th} order), the 1^{st} order and any n^{th} order of the high-order vectors.

Nominal terms of the homotopic decomposition

The terms of the 0^{th} order are simply obtained by setting $\varepsilon = 0$ in (Eq. 2.18), leading to a GEP (Eq. 2.19) of the form:

$$\left(\mathbf{K}_S + \lambda_i^{(0)^2}\mathbf{M}_S\right)\boldsymbol{\psi}_i^{(0)} = \mathbf{0} \tag{2.19}$$

(Eq. 2.19) has no asymmetric contributions. Moreover, substituting $\lambda_i^{(0)}$ by $j\omega_i^{(0)}$ (with j, the imaginary unit) leads to the LEP in (Eq. 2.13), with a normalisation condition over the eigenvectors (Eq. 2.20).

$$\boldsymbol{\psi}_i^{(0)T}\mathbf{M}_S\boldsymbol{\psi}_i^{(0)} = 1 \tag{2.20}$$

1^{st} order terms of the homotopic decomposition

Considering only the 1^{st} order terms, (Eq. 2.18) is rewritten as follows:

$$\left(\mathbf{K}_S + \lambda_i^{(0)}\mathbf{M}_S\right)\boldsymbol{\psi}_i^{(1)} + \left(2\lambda_i^{(0)}\mathbf{M}_S\boldsymbol{\psi}_i^{(0)}\right)\lambda_i^{(1)} + \mathbf{G}_i^{(1)} = 0 \tag{2.21}$$

where $\mathbf{G}_i^{(1)} = \left(\mathbf{K}_A + \lambda_i^{(0)}\mathbf{C}_A + \lambda_i^{(0)^2}\mathbf{M}_A\right)\boldsymbol{\psi}_i^{(0)}$

Premultiplying (Eq. 2.21) by $\boldsymbol{\psi}_i^{(0)^T}$ and considering (Eq. 2.19), the 1^{st} order homotopic eigenvalue is written in the form (Eq. 2.22).

$$\lambda_i^{(1)} = \frac{-\boldsymbol{\psi}_i^{(0)^T}\mathbf{G}_i^{(1)}}{2\lambda_i^{(0)}} \tag{2.22}$$

Then, (Eq. 2.21) is written in the same form as (Eq. 2.19), with a non-zero right member.

$$\left(\mathbf{K}_S + \lambda_i^{(0)}\mathbf{M}_S\right)\boldsymbol{\psi}_i^{(1)} = \mathbf{F}_i^{(1)} \tag{2.23}$$

where $\mathbf{F}_i^{(1)} = -\left(2\lambda_i^{(0)}\mathbf{M}_S\boldsymbol{\psi}_i^{(0)}\right)\lambda_i^{(1)} - \mathbf{G}_i^{(1)}$

Unfortunately, the left member of (Eq. 2.23) is not invertible due to the symmetry of the matrices $\mathbf{K}$ and $\mathbf{M}$. Hence, another equation is needed to determine $\boldsymbol{\psi}_i^{(1)}$.

The chosen solution follows Wang's proposal for the calculation of eigenvector derivatives in [118]. The idea is to express each perturbed eigenvector as a linear combination of the 0^{th} order eigenvectors $\boldsymbol{\psi}_i^{(0)}$, i.e. normal modes, of the system. For the 1^{st} order eigenvectors, the linear combination reads:

$$\boldsymbol{\psi}_i^{(1)} = \sum_{k=1}^{N} c_k^{(1)}\boldsymbol{\psi}_k^{(0)} \tag{2.24}$$

In practice, the number n_m of available normal modes is lower than N, causing an erroneous approximation of $\boldsymbol{\psi}_i^{(1)}$. A common solution is to complete the development with a static correction term $\boldsymbol{\Gamma}_i^{(1)}$. The linear combination (Eq. 2.24) is then rewritten as (Eq. 2.25).

$$\begin{aligned}\boldsymbol{\psi}_i^{(1)} &= \sum_{k=1}^{n_m} c_k^{(1)}\boldsymbol{\psi}_k^{(0)} + \boldsymbol{\Gamma}_i^{(1)} \\ \boldsymbol{\Gamma}_i^{(1)} &= \mathbf{K}_S^{-1}\mathbf{F}_i^{(1)} - \sum_{k=1}^{n_m}\frac{\boldsymbol{\psi}_k^{(0)}\boldsymbol{F}_i^{(1)}}{w_k^{(0)^2}}\boldsymbol{\psi}_i^{(0)}\end{aligned} \tag{2.25}$$

For $i \neq k$, the coefficients $c_k^{(1)}$ are calculated with (Eq. 2.26).

$$c_{k\neq i}^{(1)} = \frac{\boldsymbol{\psi}_i^{(0)^T} \boldsymbol{F}_i^{(1)}}{w_k^{(0)^2} - w_i^{(0)^2}} \tag{2.26}$$

If $i = k$, (Eq. 2.26) cannot be evaluated. The coefficients $c_{k=i}^{(1)}$ have to be determined by other means. A second convenient normalisation, given by (Eq. 2.27), is considered. Instead of directly considering a perturbed mass normalisation homogeneous to (Eq. 2.20), this equation links the 0^{th} order eigenvector $\boldsymbol{\psi}_i^{(0)}$ and the high-order eigenvector $\boldsymbol{\psi}_i^{(1)}$ to the full mass matrix. The aim is to limit the complexity of the following developments and use the orthonormal property of (Eq. 2.27).

$$\boldsymbol{\psi}_i^{(0)^T} \mathbf{M} \boldsymbol{\psi}_i^{(1)} = 1 \tag{2.27}$$

Introducing (Eq. 2.15) and (Eq. 2.17) in (Eq. 2.27) and keeping only the 1^{st} order terms, (Eq. 2.28) reads as:

$$\boldsymbol{\psi}_i^{(0)^T} \mathbf{M}_S \boldsymbol{\psi}_i^{(1)} + \boldsymbol{\psi}_i^{(0)^T} \mathbf{M}_A \boldsymbol{\psi}_i^{(0)} = 0 \tag{2.28}$$

Substituting the 1^{st} order eigenvector by the linear combination given by (Eq. 2.24) and considering that normal modes are $\mathbf{M}_S$-orthonormals, the $c_{k=i}^{(1)}$ coefficient can be calculated with (Eq. 2.29).

$$c_{k=i}^{(1)} = -\boldsymbol{\psi}_i^{(0)^T} \mathbf{M}_A \boldsymbol{\psi}_i^{(0)} \tag{2.29}$$

$\mathbf{n}^{th}$ order terms of the homotopic decomposition

Following the developments detailed in the previous section, the expressions to evaluate any n^{th} order of the i^{th} eigensolutions can be determined and are given by (Eq. 2.30) to (Eq. 2.35).

$$\left(\mathbf{K}_S + s_i^{(0)^2} \mathbf{M}_S\right) \boldsymbol{\psi}_i^{(n)} = -\left(2 s_i^{(0)} \mathbf{M}_S \boldsymbol{\psi}_i^{(0)}\right) s_i^{(n)} - \mathbf{G}_i^{(n)} \tag{2.30}$$

where

$$\begin{aligned}
\mathbf{G}_i^{(n)} = {} & \mathbf{K}_A \boldsymbol{\psi}_i^{(n-1)} + \mathbf{C}_A \sum_{k=0}^{n-1} \boldsymbol{\psi}_i^{(k)} s_i^{(n-k-1)} + \mathbf{M}_A \left(\sum_{k=0}^{n-1} \left(\sum_{l=0}^{n-1-k} s_i^{(l)} s_i^{(n-k-l-1)} \right) \boldsymbol{\psi}_i^{(k)} \right) \\
& + 2\mathbf{M}_S \left[\sum_{k=1}^{n-1} s^{(0)} s_i^{(k)} \boldsymbol{\psi}_i^{(n-k)} + \mathbb{1}_{\{n\geq 3\}} \sum_{l=1}^{\lfloor \frac{n}{2} \rfloor - mod((n+1),2)} \left(\sum_{k=l+1}^{n-l} \left(s_i^{(l)} s_i^{(k)} \boldsymbol{\psi}_i^{(n-k-l)} \right) \right) \right] \\
& + \mathbf{M}_S \left(\sum_{k=1}^{\lfloor \frac{n}{2} \rfloor} s_i^{(k)^2} \boldsymbol{\psi}_i^{(n-2k)} \right)
\end{aligned} \tag{2.31}$$

where $mod(x, 2)$ is the modulo 2 of x, $\lfloor x \rfloor$ corresponds to the greatest integer less than or equal to x, also denoted floor(x) and where the indicator function $\mathbb{1}_{\{n \geq p\}}$ equals 1 if n is superior to p and 0 otherwise.

The equation of the n^{th} homotopic eigenvalue is given by (Eq. 2.32).

$$\lambda_i^{(n)} = \frac{-\boldsymbol{\psi}_i^{(0)^T} \mathbf{G}_i^{(n)}}{2\lambda_i^{(0)}} \tag{2.32}$$

Using Wang's proposal to determine the n^{th} order homotopic eigenvectors, (Eq. 2.33) is derived.

$$\begin{aligned} \boldsymbol{\psi}_i^{(n)} &= \sum_{k=1}^{n_m} c_k^{(n)} \boldsymbol{\psi}_k^{(0)} + \boldsymbol{\Gamma}_i^{(n)} \\ \Gamma_i^{(n)} &= \mathbf{K}_S^{-1} \mathbf{F}_i^{(n)} - \sum_{k=1}^{n_m} \frac{\boldsymbol{\psi}_k^{(0)^T} \boldsymbol{F}_i^{(n)}}{\omega_k^{(0)^2}} \boldsymbol{\psi}_i^{(0)} \end{aligned} \tag{2.33}$$

Finally, (Eq. 2.34) and (Eq. 2.35) are determined following a similar reasoning than the one of the first order.

$$c_{k \neq i}^{(n)} = \frac{\boldsymbol{\psi}_i^{(0)^T} \boldsymbol{F}_i^{(j)}}{\omega_k^{(0)^2} - \omega_i^{(0)^2}} \tag{2.34}$$

$$c_{k=i}^{(n)} = -\boldsymbol{\psi}_i^{(0)^T} \mathbf{M}_A \boldsymbol{\psi}_i^{(n-1)} \tag{2.35}$$

The computational cost of determining the perturbed eigensolutions is relatively moderate. Indeed, in (Eq. 2.33), the first term of the sum is low-cost as the coefficients of the residue are determined through simple operations. Conversely, the static correction $\Gamma_i^{(n)}$ is much more expensive to solve. Indeed, the second term is of complexity $\mathcal{O}(n_m \times d^2)$, with n_m, the number of considered normal modes and d, the homotopic order of truncation. The first term is of complexity $\mathcal{O}(N^3)$, when using a LU decomposition algorithm (N being the number of degrees of freedom). However, the decomposition of the symmetric contribution of the stiffness matrix involved in these expressions is only needed once. Hence, for the computation of first homotopy order, the numerical complexity is $\mathcal{O}(N^3 + n_m \times d^2)$; for the other homotopy orders, the numerical complexity is $\mathcal{O}(n_m \times d^2)$.

Construction of the enhanced projection basis

As the homotopy eigenvectors of (Eq. 2.33) are computed using the damping and asymmetric contributions, the representativity of those phenomena is more important and increases with the number of homotopy order. Thus, the key aspect of this method is to enhance the projection basis with those high-order eigenvectors. In a way, the suggested method, denoted as HOmotopy PErturbation

Projection (or HOPEP), retains the same characteristics of the classical method for large problems: linearization of the QEP, projection to reduce the corresponding GEP and, then, resolution with a global algorithm such as the QZ algorithm. HOPEP can thus be considered a generalisation of the classical method.

The projection basis $\mathbf{T}$ in HOPEP takes the following form:

$$\mathbf{T} = \left[\boldsymbol{\phi}_1^{(0)} \ \dots \ \boldsymbol{\phi}_{n_m}^{(0)} \ \boldsymbol{\psi}_1^{(1)} \ \dots \ \boldsymbol{\psi}_m^{(d)}\right] \tag{2.36}$$

When dealing with complex values in numerical analysis, it is frequent to separate the real and imaginary parts in mathematical expressions. This solution allows preserving a maximum of information, but doubles the number of modes in the projection basis, and so, the time to solve the QEP. A more economic solution considers only the real parts of the perturbed eigenvectors. It is, however, important to keep sufficient information in the real part before the decomposition of the complex quantities. A common method [119] is to scale the complex matrix with the maximum value in each column and retains only the modified real part.

As the projection basis $\mathbf{T}$ in HOPEP is composed of a set of modes from different types or orders, it is mandatory to proceed to an orthonormalization phase. Here, two numerical methods have been identified. The first method is the so-called iterative Gram-Schmidt process [120]. This method is accurate and allows computing an orthonormal basis preserving the initial size. The second method, implemented in the Structural Dynamics Toolbox [121], orthonormalised the projection basis using a preconditioned Cholesky decomposition and eliminate eventual colinear vectors using a singular value decomposition. An advantage of this second solution is a lower numerical cost. For both methods, the vectors of the basis are normalised thanks to the identity matrix.

Once the different calculations detailed above are performed, the modified projection basis $\mathbf{T}$ is used in (Eq. 2.12) to solve the approximate solutions of the QEP.

Many tests showed that the performance of HOPEP relies on a good balance between the number of normal modes, the number of perturbed eigenvectors and the maximum homotopy order. The flowchart in Fig. 2.1 presents the strategy to efficiently build the projection basis. The algorithm mainly focuses on the sequence and criteria used to define the number of normal modes (n_m) and the maximum homotopy order (d). Indeed, the number of perturbed eigenvectors (m) is fixed to the number of mode of interest, since a stabilisation of all complex eigensolutions is desired in the entire frequency band. It must be noted that each modification of the projection basis is followed by a new evaluation of the QEP.

Starting with a number of normal modes equal to the number of modes in the frequency band of interest, additional normal modes or homotopy orders are iteratively added to the projection basis. The process is achieved by monitoring the evolution of the residual error and the convergence of

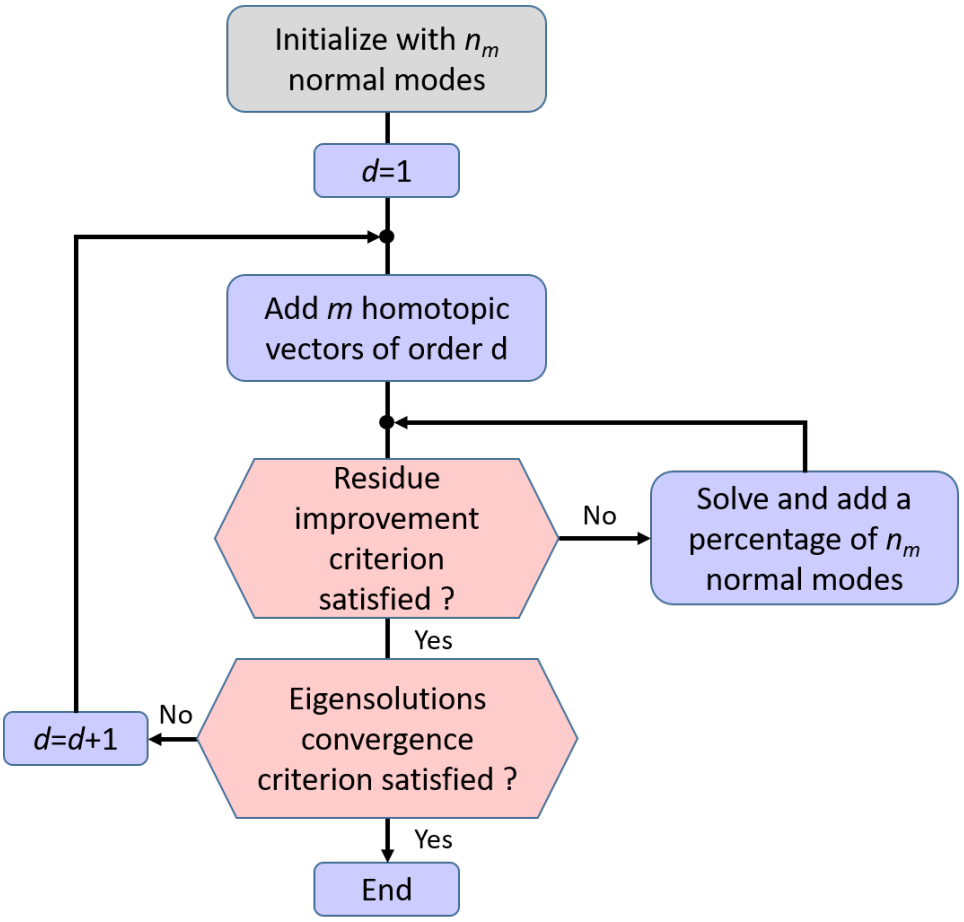


FIGURE 2.1: Construction of the projection basis.

eigensolutions. For each mode, the residual error is calculated with the Manhattan norm of the residual vector defined by (Eq. 2.37).

$$R_i = \| \left(\mathbf{K} + \lambda_i \mathbf{C} + \lambda_i^2 \mathbf{M} \right) \boldsymbol{\psi}_i \|_1 \tag{2.37}$$

It is expected that the perturbed eigenvectors of any order, added to the projection basis, reduce the residual error for all complex modes compared to those calculated at order 0. It is, however, more relevant to first increase the number of normal modes, rather than homotopy order, if the residual error is not reduced for all modes or at least many of them. The convergence criterion is evaluated with the relative errors between eigensolutions of two successive orders. In practice, it is sufficient to monitor only the real parts of eigensolutions. The algorithm stops when the relative errors for all modes of interest are inferior to a given threshold; otherwise, a new homotopy order is added to the projection basis.

Table 2.1 summarises all the parameters of the algorithm and highlights their main action.

Table 2.1
Hyperparameters of the HOPEP algorithm.

Parameter type	Symbol	Purpose
Normal mode basis size	n_m	Control the basis size used for the homotopy vectors
Homotopy vectors number	m	Control the added homotopy vectors number per order
Homotopy vectors order	d	Control the enhancement of the projection basis
Orthonormalisation scheme	\	Choice of the orthonormalisation scheme
Vector decomposition method	\	Control how the homotopy vectors are traited

2.3 Application of the HOmotopy PErturbation Projection method

2.3.1 Assessment procedure

To highlight the performance of the HOPEP method, the simplified disc-pad model (presented in Fig. 2.2) is studied. The stability of this system is, then, analysed in the Lyapunov sense, through the frequency method. This method involves a CEA, in which the system of equations is linearised around a stationary state, as thoroughly described in (Eq. 2.2). The positive real parts of the complex eigenvalues account for the unstable modes.

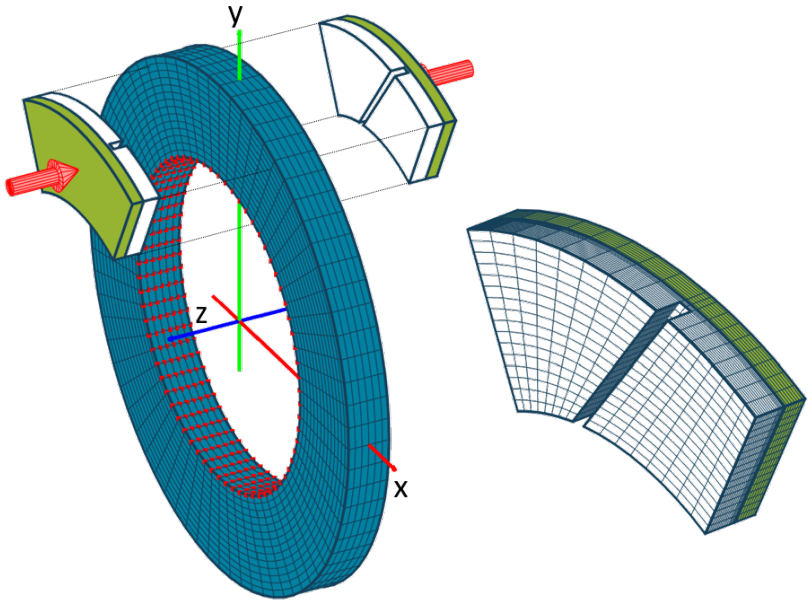


FIGURE 2.2: FE model of the simplified brake system.

Each pad is constituted of a lining with a centered diagonal slot and a backplate. The full mesh is composed of 23,302 nodes and 19,104 linear brick elements. The coupled model contains 66,490 DOF. The material properties correspond to cast iron for the disc, steel for the backplates and a sintered material for the linings.

Table 2.2
Geometrical and material parameters of the simplified brake model.

Component	**Parameter**	**Value**	**Unit**
Disc	Thickness	25.	mm
	Inner radius	95.5	mm
	Outer radius	152.5	mm
	Density	7200.	kg/m^3
	Young Modulus	130.	GPa
	Poisson's ratio (ν)	0.3	
Back plate	Thickness	7.	mm
	Inner radius	96.5	mm
	Outer radius	151.5	mm
	Angle	$2\pi/7$	rad
	Density	7300.	kg/m^3
	Young Modulus	170.	GPa
	Poisson's ratio (ν)	0.3	
Lining	Thickness	10.	mm
	Inner radius	96.5	mm
	Outer radius	151.5	mm
	Angle	$2\pi/7$	rad
	Slot width	5.	mm
	Density	2600.	kg/m^3
	Young Modulus (E_x)	6.5	GPa
	Young Modulus (E_y)	6.5	GPa
	Young Modulus (E_z)	3.5	GPa
	Poisson's ratio (ν_{xy})	0.2	
	Poisson's ratio (ν_{yz})	0.15	
	Poisson's ratio (ν_{xz})	0.20	
	Shear Modulus (G_{xy})	1.	GPa
	Shear Modulus (G_{xz})	1.	GPa

The material behaviour of the disc and backplates is supposed isotropic, but a transverse isotropic material is considered for the linings. The dimensions and material properties of each component are reported in Table 2.2.

The mass and stiffness matrices (respectively, $\mathbf{M}$ and $\mathbf{K}$) of the simplified disc-pad are computed using a commercial software, Abaqus. A Rayleigh damping (Eq. 2.38) is chosen to model the damping matrix $\mathbf{C}$ of the system.

$$\mathbf{C} = \alpha\mathbf{M} + \beta\mathbf{K} \tag{2.38}$$

The mass and stiffness damping coefficients, namely α and β, as well as friction coefficient μ between pads and disc are reported in Table 2.3.

Finally, the fixed boundary conditions for all directions are applied on the nodes of the interior diameter of the disc (the red markers in Fig. 2.2) and all translations along x and y are fixed for the pads. Moreover, a pressure of 25 bars, represented by the red arrows, is applied on the external pad

Table 2.3
Friction and Rayleigh coefficients for the brake system.

Parameter	**Value**	**Unit**
Friction coefficient μ	0.6	
Damping coefficient α	25.	s^{-1}
Damping coefficient β	5.10^{-8}	s

surfaces and a rotation velocity of 8.17 rad/s around the z axis is applied to the disc to initiate sliding friction.

All the tests performed on the considered brake model require a starting phase to construct the system matrices. Using the parameter values given in Section 2.2, the stiffness $\mathbf{K}$, damping $\mathbf{C}$ and mass $\mathbf{M}$ matrices of the uncoupled system (i.e., block diagonal matrices) are pre-computed in the commercial software Abaqus and exported in a dedicated software developed in Matlab.

The symmetric contributions are given by (Eq. 2.39), using the normal connectivity matrix $\mathbf{T}_N$.

$$\begin{aligned} \mathbf{M}_S &= \mathbf{T}_N^T \mathbf{M} \mathbf{T}_N \\ \mathbf{K}_S &= \mathbf{T}_N^T \mathbf{K} \mathbf{T}_N \end{aligned} \tag{2.39}$$

The asymmetric contributions are given by (Eq. 2.40), using the normal and tangential connectivity matrices $\mathbf{T}_N$ and $\mathbf{T}_f$.

$$\begin{aligned} \mathbf{M}_A &= \mathbf{T}_f^T(\mu) \mathbf{M}_0 \mathbf{T}_N - \mathbf{M}_S \\ \mathbf{C}_A &= \mathbf{T}_f^T(\mu) \mathbf{C}_0 \mathbf{T}_N \\ \mathbf{K}_A &= \mathbf{T}_f^T(\mu) \mathbf{K}_0 \mathbf{T}_N - \mathbf{K}_S \end{aligned} \tag{2.40}$$

Section 2.3.2 to 2.3.3 analyse the effects of the key parameters over the convergence of the complex eigensolutions. The algorithm presented in Fig. 2.1 is applied in Section 2.3.4. In each case, the frequency band of interest is inferior to 20 kHz, which corresponds to roughly 74 complex modes.

For all the tests, the homotopy developments are limited to the 6^{th} order, which represents a good balance between precision and numerical cost. To assess for the precision of the suggested method, the QEP Residual (hereby denoted QEPR), given by (Eq. 2.37) is monitored. The analysis of this quantity is rather simple: the lower, the better. For the convergence of the complex eigenvalues, two error criteria are monitored: (Eq. 2.41) for the real parts and (Eq. 2.42) for the circular frequencies. The chosen relative error for complex eigenvectors, defined by (Eq. 2.43), is of the most accurate since major errors in only one vector component are noticeable. It is relevant to consider that a value below 10% denotes a very good agreement. Finally, for the last assessment in Section 2.3.4, an eigenvector correlation is performed with the most widely used MAC, recalled in (Eq. 2.44) for complex quantities [122].

$$e_{Re(\lambda_i)} = \frac{Re(\lambda_i^{*(n+1)}) - Re(\lambda_i^{*(n)})}{Re(\lambda_i^{*(n+1)})} \tag{2.41}$$

$$e_{|\lambda_i|} = \frac{|\lambda_i^{*(n+1)}| - |\lambda_i^{*(n)}|}{|\lambda_i^{*(n+1)}|} \tag{2.42}$$

$$e_{\boldsymbol{\psi}_i} = \frac{\|\boldsymbol{\psi}_i^{*(n+1)} - \boldsymbol{\psi}_i^{*(n)}\|}{\|\boldsymbol{\psi}_i^{*(n+1)}\| + \|\boldsymbol{\psi}_i^{*(n)}\|} \tag{2.43}$$

$$\mathrm{MAC}_{ik} = \frac{\left\|\psi_i^{*(n+1)H}\psi_k^{*(n)}\right\|^2}{\left\|\psi_i^{*(n+1)H}\psi_i^{*(n+1)}\right\| \left\|\psi_k^{*(n)H}\psi_k^{*(n)}\right\|} \tag{2.44}$$

2.3.2 Impact of the projection basis definition

2.3.2.1 Convergence of the residue

The projection basis $\mathbf{T}$, enhanced with the homotopic basis, depends on two main parameters: the number of normal modes n_m and the homotopy order d. Here, the influence of these parameters are evaluated according to three configurations:

- A gradual increase of the sets of normal modes (from n_m to $7 \times n_m$);
- A set of n_m normal modes, completed with a set of m perturbed eigenvectors (from 1^{st} to 6^{th} order);
- A set of $2 \times n_m$ normal modes, completed with a set of m perturbed eigenvectors (from 1^{st} to 6^{th} order).

The first configuration is linked with the projection basis $\mathbf{T}_{norm}$; the second configuration, with $\mathbf{T}_{once}$; and the third configuration, with $\mathbf{T}_{twice}$.

At each order, the perturbed eigenvectors are split into real and imaginary parts and then merged with $\mathbf{T}$. Then, the complete projection basis is orthonormalised using the iterative Gram-Schmidt process in order to preserve all available information. The number of normal modes n_m and perturbed eigenvectors m is fixed to 74, which tallies with the mode number needed to stabilise the results of the Lanczos algorithm over the studied frequency spectrum (from 1 Hz to 20kHz). Finally, it is worth noting that the 0^{th} order stands for a set of normal modes, whatever the considered configuration.

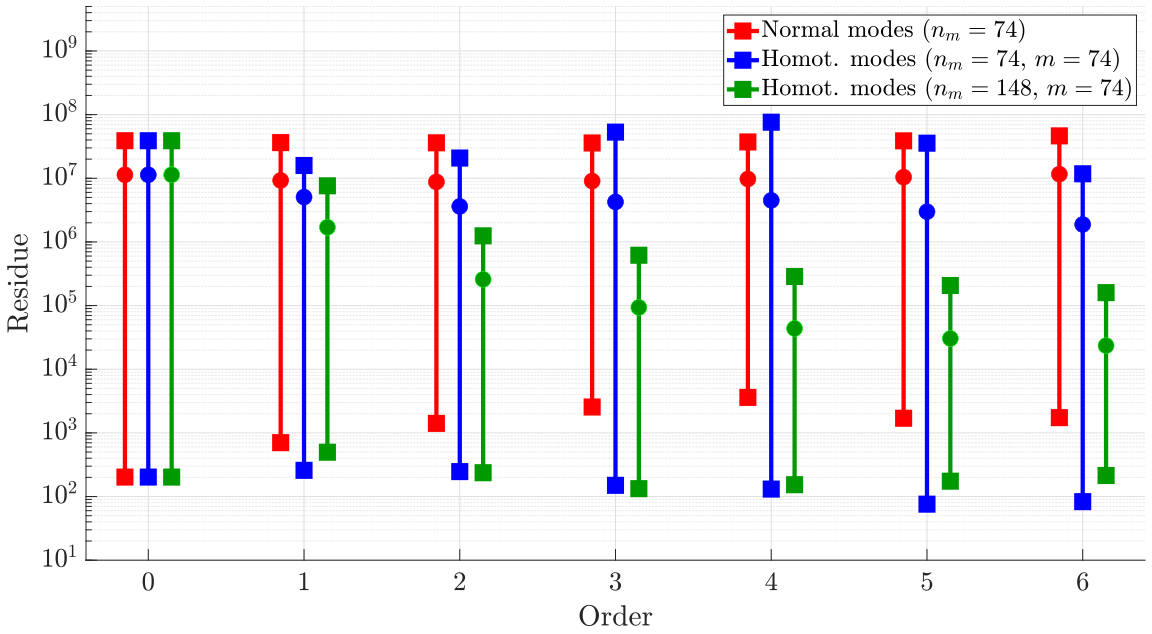


FIGURE 2.3: Evolution range of the QEP residual as a function of homotopy orders (● is the mean residue, ■ is the min-max residue).

Observations

Fig. 2.3 shows the convergence of the QEP residue with the increase of the projection basis size. The performance at each order is given as a variation range between the minimum and maximum residue. Finally, the mean residue, computed over all the eigensolutions, is also shown.

For the first configuration, the projection basis $\mathbf{T}_{norm}$ exhibits no clear improvements of the quality of the eigensolutions. The mean and maximum residue are almost constant (resp. 10^7 and 10^8), for all order. The minimum residue is even deteriorated with the order increase, whereas the number of studied modes remains identical.

The second configuration, linked with the projection basis $\mathbf{T}_{once}$, provides some interesting improvements. Indeed, the minimum residue is slightly lowered, whereas the mean residue is moderately improved from 10^7 to 10^6 decades. On the contrary, the maximum residue slightly worsens.

Finally, the third configuration exhibits the most interesting results. Indeed, the mean and maximum residues of the eigensolutions decrease fast (resp. from 10^7 to 10^4 and from 10^8 to 10^5). The projection basis $\mathbf{T}_{twice}$ causes the residues to be less widespread than the two other projection basis (six decades for $\mathbf{T}_{once}$ and four decades for $\mathbf{T}_{norm}$).

Discussions

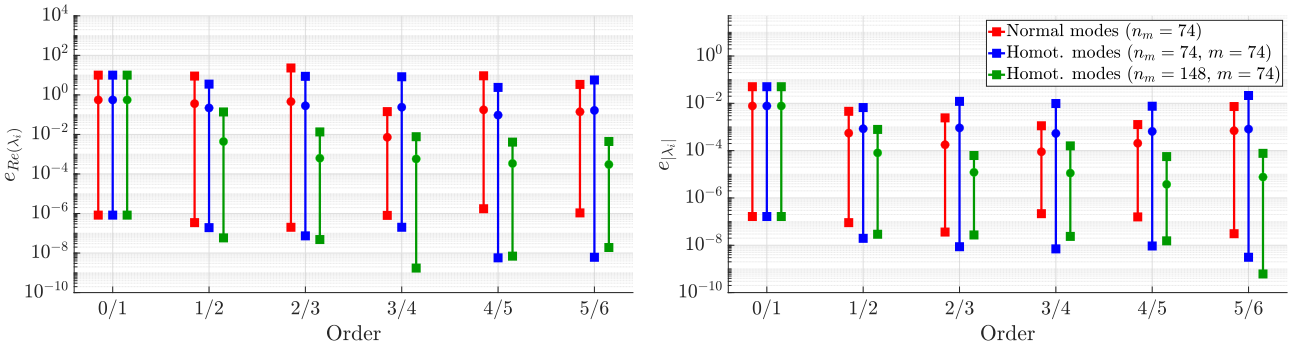
The previous results have strong implications, especially for the subspace $\mathbf{T}_{norm}$. The evolution of the residue over the order shows that widening the subspace with only normal modes is not necessarily a good solution. Indeed, the minimum residue is worsening, and the other quantities remain constant.

The additional vectors of $\mathbf{T}_{norm}$ act as discrepancies and provide no qualitative representations compensating for the absence of the asymmetric contributions in the system. Conversely, both homotopy projection bases $\mathbf{T}_{once}$ and $\mathbf{T}_{twice}$ improve the quality of the computed eigensolutions, whether this is slightly, moderately or greatly.

In a way, the results associated with $\mathbf{T}_{norm}$ are logical, since no information about the tangential contact forces and the damping are taken into account. For the projection bases $\mathbf{T}_{once}$ and $\mathbf{T}_{twice}$, the results are also logical. Indeed, the homotopy eigenvectors are approximated as a linear combination of normal eigenvectors and asymmetric contributions (Eq. 2.33), corrected with a static residue (also derived from the asymmetric contributions). Thus, widening the sum on which the homotopy eigenvectors are computed allows a better approximation, by providing additional information about the symmetric and asymmetric contributions.

2.3.2.2 Convergence of the eigensolutions

Fig. 2.4 shows the convergence of the eigenvalue errors with the increase of the projection basis size. The presentation of the results in the figure is similar to the previous plot.



(A) Real part (B) Circular frequency

FIGURE 2.4: Evolution range of the eigenvalue errors as a function of homotopy orders (● is the mean residue, ■ is the min-max residue).

Observations

The projection bases $\mathbf{T}_{norm}$ and $\mathbf{T}_{once}$ exhibit the same nature of errors for both the real and imaginary parts. The mean and max errors are moderately fluctuating around 10^{0} for the real part and 10^{-3} for the circular frequencies. Conversely, the projection on the subspace $\mathbf{T}_{twice}$ shows more interesting results. In fact, in all cases, the evolution of the two convergence criteria presents the same decrease, attesting the stabilisation of solutions. The main improvement is observed for real parts where maximum relative errors calculated between order 0 and order 1 are relatively high, denoting large discrepancies for some modes. Increasing the homotopy order significantly reduce these errors, which can reach very low values for the circular frequencies.

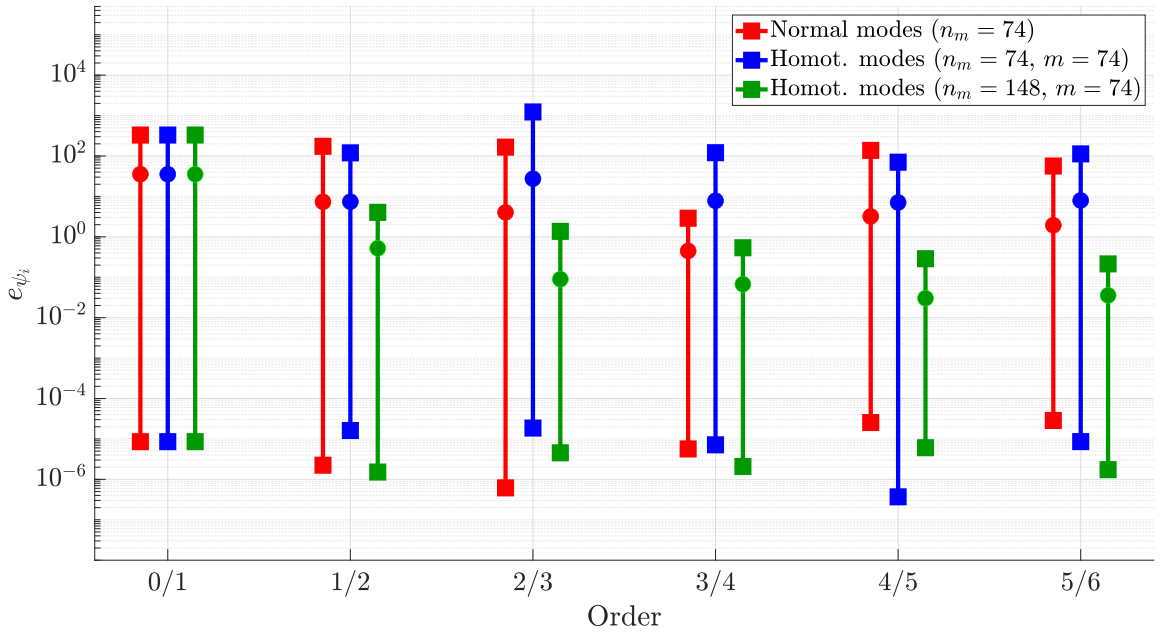


FIGURE 2.5: Evolution range of the eigenvector error as a function of homotopy orders (● is the mean residue, ■ is the min-max residue).

Fig. 2.5 shows the convergence of the eigenvector errors with the increase of the projection basis size. The presentation of the results in the figure is similar to the previous plot.

The results for the eigenvectors approximated with the three methods are similar to the previous observations. The projection basis $\mathbf{T}_{twice}$ ensures a convergence of the eigenvectors, whereas for the two other bases, the mean errors moderately fluctate around 10^1. The use of a wider normal basis for the computation of the homotopy vectors allows a reduction of two decades on the magnitude error.

Discussions

The study of the convergence of the eigensolution magnitude error rises confidence on the consistency of the quality of the computed solutions with the projection basis $\mathbf{T}_{twice}$. Indeed, as the magnitude error of the given solutions decreases, the QEPR also decreases, meaning that the determined eigensolutions are both converging and less erroneous.

Conclusions

In conclusion, these first results give the two main characteristics of the suggested method. Firstly, the use of perturbed eigenvectors in the projection basis $\mathbf{T}$ greatly improves the QEP residue, provided that the size of the normal mode basis is at least greater than the number of modes of interest. Secondly, the stabilisation of QEP residue can be monitored with only the mean residue, as the stability of the eigensolutions can be tracked with the mean real parts.

2.3.3 Impact of the normalisation method

2.3.3.1 Convergence of the residue

The second test aims to study the influence of the orthonormalisation scheme as well as the decomposition of complex perturbed eigenvectors, using either real or imaginary parts (labelled R+I) or modified real parts (labelled Rm). The projection basis can be orthonormalised either with the Gram-Schmidt process (labelled GS) or with the Cholesky decomposition (labelled Chol). In all four cases, the configuration of the projection bases follows the best out of the previous test, namely a normal mode basis of 148 eigenvectors completed with sets of 74 perturbed eigenvectors up to order 6.

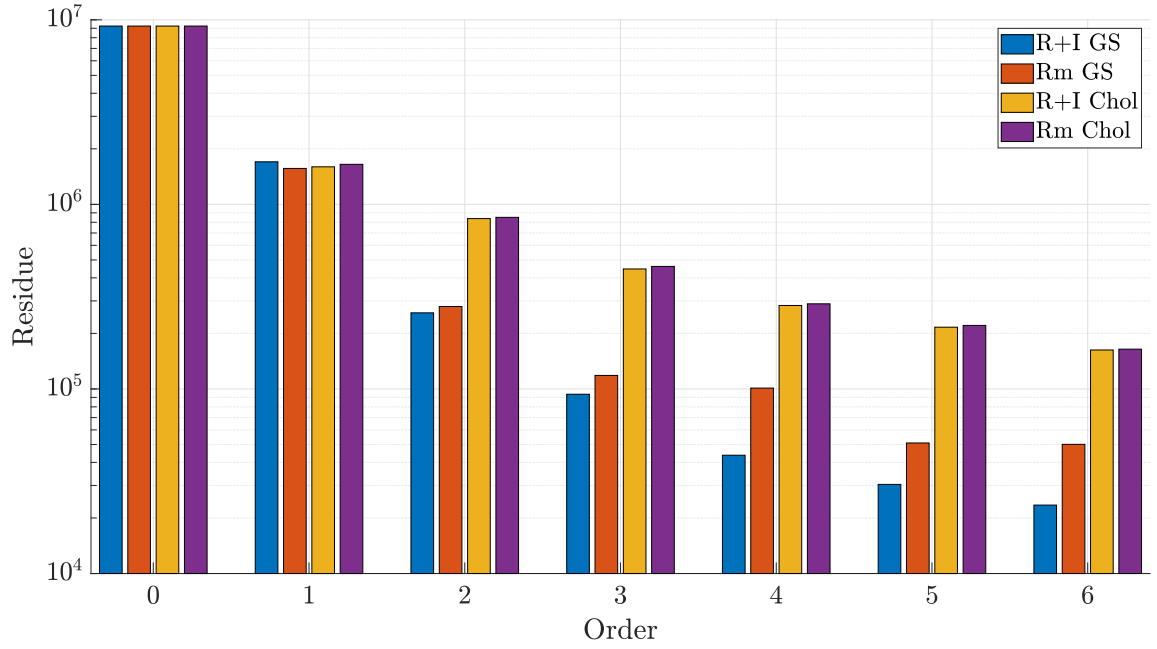


FIGURE 2.6: Influence of the projection basis orthonormalisation on the residue. Evolution of the mean residue as a function of homotopy orders.

Observations

Fig. 2.6 shows the evolution of the mean QEPR as a function of the homotopy order. First, all the configurations show a reduction and a convergence of the residue. For the first homotopy order, no difference is observed between the orthonormalisation techniques. Nonetheless, some differences appear with the increase of homotopy order: the Gram-Schmidt algorithm provides results from 0.5 to one decade lower than the Cholesky decomposition, when the homotopy order is greater than 2.

For the decomposition of the complex perturbed eigenvectors, the modified real part is moderately less efficient than the real and imaginary parts decomposition when orthonormalised with the Gram-Schmidt process. Conversely, no differences are observed with the Cholesky decomposition.

Discussions

A main feature of the suggested method, highlighted by those observations, is that the improvement brought by HPM is still relevant whatever the orthonormalisation schemes and the decomposition of the perturbed eigenvectors. Hence, it is possible to consider the modified real part decomposition for lowering the numerical cost, without a too important reduction of the precision. Furthermore, if the numerical cost becomes a problem, it may be once again reduced by considering the Cholesky decomposition method, which is a lot cheaper than the Gram-Schmidt algorithm.

2.3.3.2 Convergence of the real part

Fig. 2.7 shows the evolution of the mean real part as a function of the homotopy order. The results clearly show that the orthonormalisation schemes and the decomposition methods have no impact over the convergence of the real parts, since the errors are both lowering and stabilising.

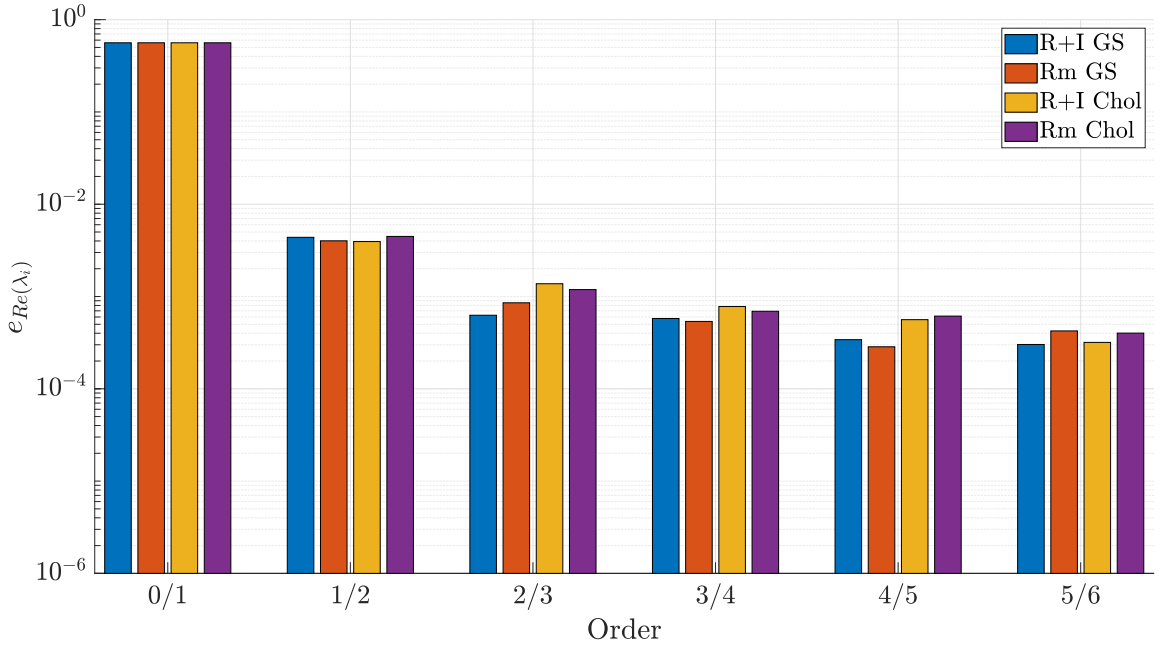


FIGURE 2.7: Influence of the projection basis orthonormalisation on the real part. Evolution of the mean real part error as a function of homotopy orders.

Conclusions

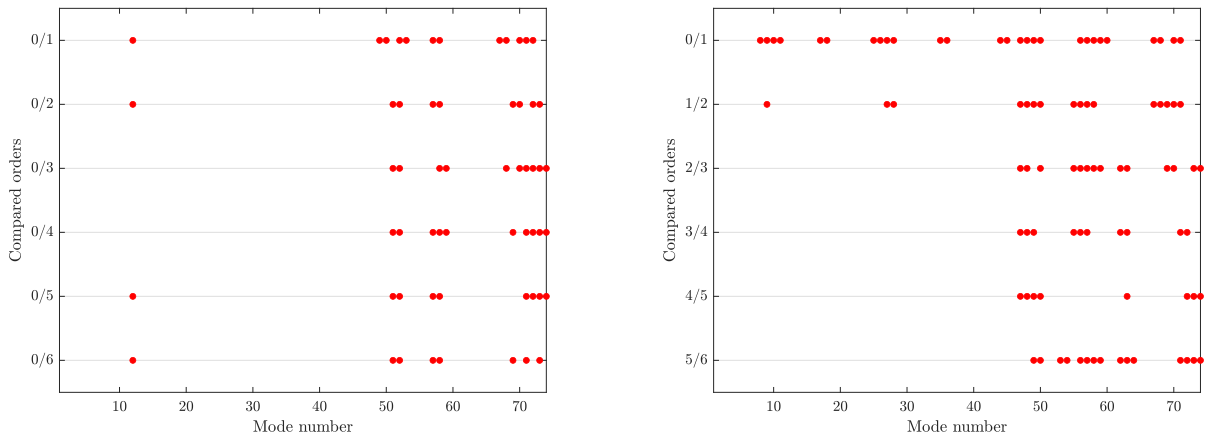
In conclusion, the best configuration remains the one which uses the Gram-Schmidt process on the real and imaginary parts of perturbed eigenvectors. Nonetheless, if the computational cost becomes prohibitive (dealing with many eigensolutions or many degrees of freedom), it may be reduced by considering less efficient configurations, which are faster to compute.

2.3.4 Analysis of the algorithm performance on a large frequency spectrum.

2.3.4.1 Convergence of the eigensolutions and residue monitoring

In the HOPEP algorithm, the iterative construction of the projection basis $\mathbf{T}$ relies on the residue criterion and the stabilisation criterion, which governs an increase of respectively the number of normal modes and of the homotopy order. The residue criterion at order d is validated if the difference between the residual error at that order and the residual error at order 0 is negative. Ideally, this criterion should be validated for all complex modes. In practice, a reasonable threshold can be fixed to 95% of the number of modes. With the second criterion, the solutions of the QEP are considered stabilised if the relative errors between real parts of complex eigenvalues for two successive orders are inferior to a given threshold, typically 5%.

In this section, the behaviour of these two criteria is studied through three configurations of the projection basis. The first configuration corresponds to the initialization phase of the algorithm, where the size of the normal mode basis is fixed to $n_m = 74$ complex modes of interest. For the second configuration, the normal mode basis is chosen in a frequency band increased by 10% (up to 22 kHz) and contains $n_m = 93$ modes. In the same way, the frequency band for the third configuration is increased by 20%, which provides a normal mode basis of $n_m = 113$ modes. For the three configurations, the projection bases are progressively completed by the modified real part of perturbed eigenvectors up to order 6 and are orthonormalised thanks to the Gram-Schmidt process. Only the 74 first complex modes of the QEP are still studied, and the threshold stabilisation criterion is fixed to 5%.



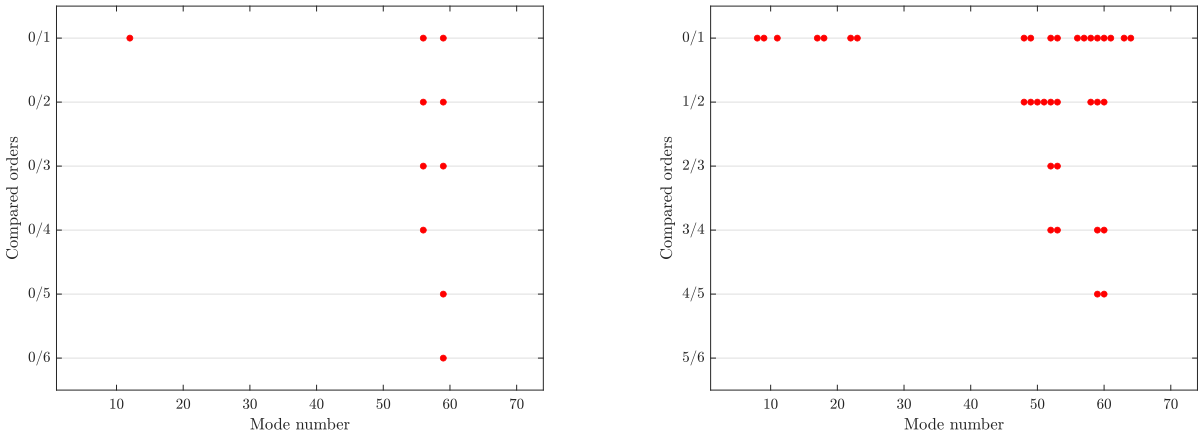
(A) **Residue checking** (B) **Stabilisation Checking**

FIGURE 2.8: (a) Residual criterion checking per mode and (b) stabilisation criterion checking per mode as a function of homotopy orders for the first configuration ($n_m = 74$). A red dot indicates an unsatisfactory check.

Observations

The results of the first configuration are reported in Fig. 2.8, with the progression of the homotopy order (from top to bottom). A red dot indicates that the residual improvement or the stabilisation is not valid for the related mode.

With this configuration, the normal mode basis is not sufficient to stabilise the complex modes in the higher frequency band and the residual error is not improved for 10% of modes, no matter what the homotopy order is.



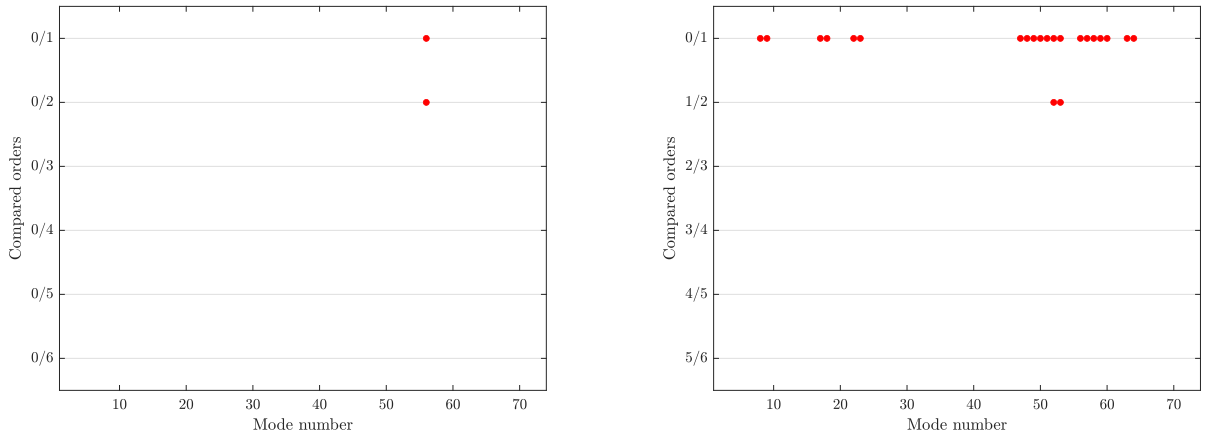
(A) **Residue checking** (B) **Stabilisation Checking**

FIGURE 2.9: (a) Residual criterion checking per mode and (b) stabilisation criterion checking per mode as a function of homotopy orders for the second configuration ($n_m = 93$). A red dot indicates an unsatisfactory check.

In the same way, the results of the second configuration are presented in Fig. 2.9 and for the third configuration, in Fig. 2.10. In both cases, the validation criteria show an improvement in both stability and precision of the eigensolutions. However, for the second configuration, six homotopy orders are needed to stabilise the modes in the entire frequency band. As expected, with the third configuration, the results improve faster since three homotopy orders are enough to stabilise all modes.

Table 2.4 presents a focus on the real parts and frequencies calculated with the three configurations from order 0 (normal modes only) up to order 3 in a way that shows the behaviour of the results according to the setting of the projection basis more specifically. In this table, the positive values of real parts, which correspond to the unstable modes, are highlighted in light gray.

From all the considered modes, only the mode 46 is not altered by the change of configurations (*i.e.*, the increase of normal modes). Conversely, all the other eigenvalues are affected, both in signs and in values. Important variations of the real part values are even observed for mode number 47, 51 and 52. Nonetheless, they are quickly stabilised with homotopy order in the third configuration ($n_m = 113$).



(A) **Residue checking** (B) **Stabilisation Checking**

FIGURE 2.10: (a) Residual criterion checking per mode and (b) stabilisation criterion checking per mode as a function of homotopy orders for the third configuration ($n_m = 113$). A red dot indicates an unsatisfactory check.

For mode number 53, the real part is always negative for the first configuration ($n_m = 74$), indicating a stable mode. For the second ($n_m = 93$) and the third ($n_m = 113$) configurations, this real part becomes positive as soon as one homotopy order is taken into account, but remains negative if only normal modes are considered. In addition, as high fluctuations of the real part arise for the second configuration when using homotopy order, a decrease of the difference between two orders is occurring with the third configurations.

With regard to the modes 48 to 50, important fluctuations of real parts are observed depending on the number of normal modes used in the projection basis. Regarding the first configuration, starting with three stable modes, two unstable modes are finally detected if homotopy orders are added. Nevertheless, the large variations of the values of the real parts lead to a doubtful perception of this result. Conversely, for the second and the third configurations, the state of the modes is completely different since only the mode number 49 is detected as unstable.

Discussions

The latter observation clearly shows that the stability of the eigensolutions is governed by a good balance of normal modes and homotopy orders. Indeed, with the highest number of normal modes without homotopy order, the modes number 48 and 50 would be considered unstable. However, the amount of normal modes remains essential to improve the convergence of the stabilisation as it can be observed for the third configuration compared to the second one.

All observations given above are also strengthened by the correlations and relative eigenvector errors reported in Table 2.5. The comparison between eigenvectors of the same number can be arguable because some MAC values, for modes 47 to 49, do not correspond to any agreement between

Table 2.4
Real parts and natural frequencies for a selection of modes. Values are given for the three configurations of the projection basis from homotopy order 0 (H0) up to order 3 (H3).

Modes No	**Re**(s_i)	$\lvert s_i\rvert/2\pi$	**Re**(s_i)	$\lvert s_i\rvert/2\pi$	**Re**(s_i)	$\lvert s_i\rvert/2\pi$	**Re**(s_i)	$\lvert s_i\rvert/2\pi$
	n_m**=74 H0**		n_m**=74 H1**		n_m**=74 H2**		n_m**=74 H3**	
46	-297.78	17001.38	-298.71	17029.26	-299.34	17047.79	-299.19	17043.26
47	-319.65	17641.22	-1427.77	17924.99	-1731.21	17996.67	-1441.13	18077.99
48	-321.24	17686.83	768.63	17925.94	1067.05	17997.88	771.11	18078.95
49	-331.27	17971.80	-1246.17	18003.58	-1825.61	18089.39	-1929.98	18101.83
50	-333.28	18028.20	581.42	18004.37	1154.87	18090.68	1258.38	18103.22
51	-338.68	18179.48	-338.65	18178.50	-338.61	18177.42	-338.51	18174.75
52	-338.91	18185.70	-338.99	18188.03	-339.06	18189.92	-339.44	18200.59
53	-359.81	18758.83	-355.64	18646.06	-351.56	18534.85	-350.79	18513.63
	n_m**=93 H0**		n_m**=93 H1**		n_m**=93 H2**		n_m**=93 H3**	
46	-298.17	17012.97	-298.78	17031.24	-299.09	17040.34	-299.06	17039.61
47	-322.16	17712.98	-327.70	17870.91	-333.02	18020.99	-333.46	18033.21
48	-1431.20	17995.62	-1646.37	18077.69	-1792.70	18099.79	-1846.34	18110.00
49	767.05	17996.57	976.42	18078.83	1121.21	18101.06	1174.13	18111.31
50	-398.73	18180.62	-400.09	18179.11	-427.55	18169.97	-421.89	18170.66
51	-278.72	18180.67	-277.25	18179.16	-249.14	18170.05	-254.85	18170.73
52	-347.09	18412.20	-1852.22	18437.30	-1375.01	18385.26	-1759.76	18380.07
53	-350.84	18515.19	1156.40	18438.63	682.86	18386.17	1068.07	18381.32
	n_m**=113 H0**		n_m**=113 H1**		n_m**=113 H2**		n_m**=113 H3**	
46	-298.51	17023.16	-298.79	17031.55	-299.06	17039.46	-299.05	17039.33
47	-1712.34	18024.67	-335.06	18078.15	-334.78	18070.33	-334.70	18068.11
48	1046.19	18025.87	-1882.10	18092.98	-1862.89	18104.09	-1879.42	18109.32
49	-972.06	18070.48	1211.12	18094.33	1191.11	18105.42	1207.27	18110.66
50	302.51	18071.03	-429.02	18166.02	-427.42	18168.34	-424.84	18168.38
51	-338.37	18170.82	-247.38	18166.10	-249.15	18168.42	-251.73	18168.46
52	-339.79	18210.32	-1763.65	18354.98	-1900.46	18358.83	-1913.93	18357.78
53	-353.23	18580.28	1073.78	18356.23	1210.35	18360.20	1223.90	18359.17

mode shapes. As no reference is available, the analysis of the eigensolutions is carried in the sense of the convergence of the solutions.

First, one can observe a very good improvement in correlations and a drastic decrease in errors as soon as both normal mode number and homotopy orders increase. This is particularly true for the third configuration at order 3, where the relative eigenvector errors drop below 1%, denoting quasi identical vectors.

Secondly, an initial increase of the normal mode basis brings better results for the second homotopy order. Indeed, the first configuration is associated with a mean error of roughly 10% for the eigenvectors (and a correlation between the eigenforms of 98%), whereas the third configuration exhibits a mean error of roughly 2% and a perfect correlation between the eigenforms. With regard to the second configuration, the results are also better than the first configuration with a perfect correlation, but the mean error is higher (about 15%).

Table 2.5
MAC and relative eigenvector errors for a selection of modes. Comparison results are given for three following homotopy order (H0-H1) to (H2-H3) of the three projection basis configurations.

Modes No	**MAC (%)**	e_ψ(%)	**MAC (%)**	e_ψ(%)	**MAC (%)**	e_ψ(%)
i-k	n_m**=74 H0-H1**		n_m**=74 H1-H2**		n_m**=74 H2-H3**	
46-46	91.	15.34	98.	6.90	100.	1.34
47-47	85.	20.20	97.	8.81	99.	6.20
48-48	0.	70.31	97.	8.81	99.	6.20
49-49	4.	63.59	97.	9.36	100.	3.22
50-50	91.	15.23	97.	9.36	100.	3.22
51-51	99.	5.33	98.	8.57	97.	17.05
52-52	95.	16.26	97.	14.84	94.	27.45
53-53	94.	12.40	97.	9.31	100.	1.80
	n_m**=93 H0-H1**		n_m**=93 H1-H2**		n_m**=93 H2-H3**	
46-46	97.	11.04	100.	4.02	100.	0.69
47-47	1.	11.32	100.	9.30	100.	2.61
48-48	72.	8.41	100.	8.14	100.	3.03
49-49	0.	8.41	100.	8.14	100.	3.03
50-50	97.	27.63	100.	42.86	100.	4.82
51-51	97.	27.63	100.	42.86	100.	4.82
52-52	95.	14.20	100.	5.89	100.	6.16
53-53	92.	16.74	100.	5.89	100.	6.16
	n_m**=113 H0-H1**		n_m**=113 H1-H2**		n_m**=113 H2-H3**	
46-46	97.	8.31	100.	2.27	100.	0.51
47-47	1.	67.79	100.	1.03	100.	0.38
48-48	72.	27.41	100.	1.48	100.	0.61
49-49	0.	69.88	100.	1.48	100.	0.61
50-50	97.	14.85	100.	3.15	100.	0.72
51-51	97.	24.93	100.	3.15	100.	0.72
52-52	95.	12.54	100.	1.51	100.	0.39
53-53	92.	14.14	100.	1.51	100.	0.39

Conclusions

In conclusion, these results have highlighted the efficiency of the suggested algorithm to deal with the computation of the eigensolutions of a brake system. One key aspect of the method is that the precision of the solutions can be enhanced by initially widening the normal modes basis on which the homotopy orders are computed. Hence, the computational cost of the suggested method can be reduced.

2.4 Chapter outcomes

The purpose of this chapter was to **highlight the limitations of the traditional strategy to solve the Quadratic Eigenvalue Problem**. As the FE models become more and more precise, the number of equations (associated with the number of degrees of freedom) to solve exponentially grow. Hence, a direct computation of the eigensolutions of the system is not achievable, since the

QZ algorithm, used to solve the QEP, is too expensive (typically for N>1000). A projection over a subspace is then used to reduce the size of the mathematical problem. This subspace is determined by solving a traditional LEP, which encompasses neither the damping contributions nor the asymmetrical contributions. This method was successfully used in modal analysis with viscous damping, but the prediction of the eigensolutions of systems subjected to the FIV was systematically challenging.

Hence, the major proposal of this chapter is to **introduce high-order eigenvectors to the projection basis, which are computed from HPM**. These traditional eigensolutions are written as a series development, coupled with perturbation techniques. The classical QEP is then re-written in several high-order perturbed eigenvalue problems. However, these problems were shown to be ill-conditioned and do not allow the computation of the high-order eigenvectors. Thus, the complex perturbed eigenvectors has to be expressed as a linear combination of the normal eigenvectors corrected with a static residue, following Wang's proposal [118]. Finally, the determined homotopy eigenvectors are iteratively merged with the projection basis. A final step relies on decomposing the complex eigenvectors according to several methods and to normalise the projection basis once again. **This method was proven to be a generalisation of the traditional projection approach.**

The first experiment showed that the **high-order eigenvectors greatly improve the precision and the stability of the eigensolutions compared to the traditional projection method**. Two configurations of the given strategy were tested: for the first one, the number of normal modes is equal to the number of homotopy eigenvectors; for the second one, the number of normal modes is twice the number of homotopy eigenvectors. Both configurations perform better than the traditional method, from one to four decades of improvement for the residue. For the eigensolutions stabilisation, the first configuration similarly performs compared to the normal modes projection, whereas the second configuration allows a better stabilisation. This experiment also showed that the stabilisation can be monitored with the mean real part and the precision with the mean residue.

The second experiment investigated the effects of the orthonormalisation schemes (Gram-Schmidt and Cholesky) and the decomposition methods (modified real parts and real+imag) over the homotopy projection basis performance. As shown by the results, a modification of these methods does not influence the stabilisation of the eigensolutions, whereas the residue is impacted. **The Gram-Schmidt, linked with the Real+Imag decomposition, was found to be the most efficient scheme to predict the eigensolutions**. Nonetheless, the difference between this method and the worst one is only one decade. Hence, if the numerical cost is of main interest, it is possible to downgrade the precision of the results to speed up the computation of the eigensolutions.

The third experiment proposed an assessment of the HOPEP algorithm where, at each order, m homotopy vectors are added, computed from n_m normal modes. Besides, during each order, this normal modes basis is widened until the residue criterion is fullfilled. Considering this algorithm, three configurations were evaluated. The results showed that the **precision of the solutions can be enhanced by a widening of the normal basis completed by homotopy vectors**. By doing so, the correlation between the first and second homotopy order was the highest for the second

and third configurations. Similarly, the errors for the eigenvectors computation dropped to 2%, on average, for the third configuration.

All of these tests have risen confidence about the precision enhancement brought by the HOPEP algorithm. This improvement comes at an additional numerical cost, which is quantified by the numerical complexity: $\mathcal{O}(N^3 + n_m \times d^2)$ for the first homotopy order; $\mathcal{O}(n_m \times d^2)$ for the other homotopy order. For the analysis of different set of parameters of a brake system, this cost is clearly acceptable if the number of homotopy vectors m and the number of normal modes n_m are relatively low.

Chapter 3

Surrogate modelling of complex eigenvalues of brake systems

Contents

3.1 Introduction

Nowadays, the current trend is to consider uncertainty of the model parameters in numerical analyses [50, 59] and optimisation processes [123] to improve the predictivity of simulations [41]. The aim is to increase the reliability and robustness of designs by taking uncertainty of various kinds into account at the earliest stage of design. Indeed, Sarrouy et al. [67] proposed solving stochastic quadratic eigenvalue problems using intrusive polynomial chaos expansions and analyse the stability of a simplified brake squeal model as a function of variable contact stiffness and friction coefficient. Massa et al. [83] developed a numerical strategy to perform a FIV analysis, while considering model uncertainties using a fuzzy formalism. In a more targeted way, Lu et al. [124] as well as Renault et al. [2] highlighted the significant sensitivity of pad surface topographies through specific numerical simulations and experimental correlations.

In parallel of improving the quality of the results of the solver, all these strategies generate an additional computational cost. To decrease this computational burden while maintaining the numerical precision, several authors proposed to use Reduced Order Methods or Surrogate modelling. For instance, homotopy developments in a projection basis [125], the double modal synthesis method [126] or the generalised modal synthesis with complex modes [46] can be used to improve the quality and reduce the common burden of additional computations. Other researchers focused their developments on surrogate modelling, mainly on kriging [51–53, 127]. These works highlight the capabilities of the Gaussian Process to approximate frequencies and damping rates and to perform accurate sensitivity analyses.

The deep learning methods, such as Deep Neural Networks [128–130] and Deep Gaussian Processes [131, 132] were developed and successfully used to model complex behaviours, considering large training sets and managing high dimensional problems [133–135]. Recently, Stender et al. [54] proposed to use Deep Neural Network to detect, identify vibrations and predict brake squeal. Kong et al. [136] optimised Deep Neural Network architecture to detect with high accuracy the fatigue life of automotive coil springs. The Deep Gaussian Process was used in nuclear simulations [137] to perform uncertainty propagation and parameter screening, whereas Tagade et al. [138] employed it as regression for lithium-ion battery health prognosis and degradation mode diagnosis. Next, Hebbal et al. [135] integrated a Deep Gaussian Process in Efficient Global Optimisation to study different mathematical problems and an aeronautic application.

Albeit optimal training sets allow efficient surrogate model predictions, approximation errors naturally arise for suboptimal training sets. Traditionally, in engineering applications, these training sets are generated with space-filling strategies, which might be optimal if the number of samples is sufficient. The curse of dimensionality causes the number of samples needed to construct optimal training sets to grow exponentially with the number of dimensions. Unfortunately, due to the expensiveness of each sample, the number of solver evaluations has to be restricted, inducing small suboptimal training sets.

The purpose of this chapter is to study the effects of these randomly generated training sets over the performance of non-linear evolutions of FIV problems, while considering the three aforementioned surrogate models, that is to say, Gaussian Process (GP), Deep Neural Network (DNN) and Deep Gaussian Process (DGP). Conclusions about the most relevant hyperparameter setting are drawn for these surrogate models, and the pertinence of the different models with respect to the sampling density is addressed.

To highlight the capabilities of the three considered surrogate models, this chapter is organised as follows. Section 3.2 reminds the testing model, the Double Hulten, and presents the mode pairing strategy. Then, the surrogate model theory is introduced for the GP, DGP and DNN in Section 3.3. Section 3.4 assesses the performance of the surrogate model with respect to the friction coefficient, the driving parameters of the different coupling states. The best and worst hyperparameters are then studied to highlight the possible downfalls of the given surrogate models. Finally, Section 3.5 studies several uncertain scenarios, where three, five and seven parameters are successively considered to draw conclusions about the hyperparameter setting and surrogate model performance toward multiparametric scenarios.

3.2 Dynamics theory

3.2.1 Considered model

The assessment of the capabilities and limitations of the surrogate models presented in Section 3.3 is done with the Double Hulten [34, 139], shown in Fig. 3.1.

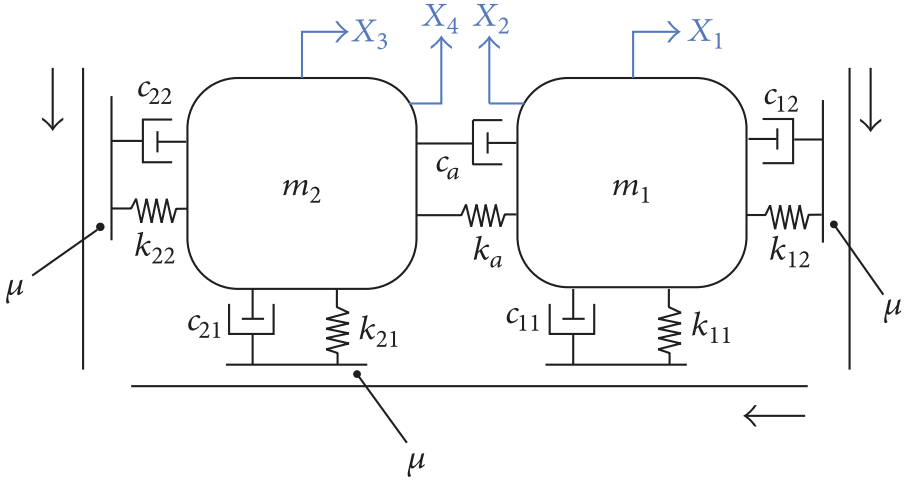


FIGURE 3.1: Double Hulten model.

The displacement of each mass is given along the X-axis and Y-axis. The contact between the masses and the bands is supposed to be permanent; the relation between the tangential friction force $\mathbf{F}_T$ and the normal friction force $\mathbf{F}_N$ is linear and given by the Coulomb law (Eq. 3.1).

$$\mathbf{F}_T = \mu \mathbf{F}_N \tag{3.1}$$

Applying the classical Lagrangian formalism, the mass, damping, and stiffness matrices are determined in (Eq. 3.2).

$$\mathbf{M} = \begin{bmatrix} m_1 & 0 & 0 & 0 \\ 0 & m_1 & 0 & 0 \\ 0 & 0 & m_2 & 0 \\ 0 & 0 & 0 & m_2 \end{bmatrix} \qquad \mathbf{C} = \begin{bmatrix} c_{12}+c_a & 0 & -c_a & 0 \\ 0 & c_{11} & 0 & 0 \\ -c_a & 0 & c_{22}+c_a & 0 \\ 0 & 0 & 0 & c_{21} \end{bmatrix}$$

$$\mathbf{K}_\Sigma = \begin{bmatrix} k_{12}+k_a & 0 & -k_a & 0 \\ 0 & k_{11} & 0 & 0 \\ -k_a & 0 & k_{22}+k_a & 0 \\ 0 & 0 & 0 & k_{21} \end{bmatrix} \qquad \mathbf{K}_{\mathrm{NL}} = \begin{bmatrix} 0 & \mu k_{11} & 0 & 0 \\ -\mu k_{12} & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu k_{21} \\ 0 & 0 & -\mu k_{22} & 0 \end{bmatrix} \tag{3.2}$$

The parameters of the Double Hulten are split into two categories: the system and the contact parameters; their reference values are recalled in Table 3.1, Table 3.2 and Table 3.3.

Table 3.1
Nominal value of the system parameters of the Double Hulten.

	m_1 (kg)	m_2 (kg)	k_a ($\mathrm{N \cdot m^{-1}}$)	c_a ($\mathrm{N \cdot s \cdot m^{-1}}$)
Nominal	1	1	1000	1

Table 3.2
Nominal value of the contact friction and stiffness parameters of the Double Hulten.

	μ	k_{11} ($\mathrm{N \cdot m^{-1}}$)	k_{12} ($\mathrm{N \cdot m^{-1}}$)	k_{21} ($\mathrm{N \cdot m^{-1}}$)	k_{22} ($\mathrm{N \cdot m^{-1}}$)
Nominal	0	3000	6000	1000	3000

Table 3.3
Nominal value of the contact damping parameters of the Double Hulten.

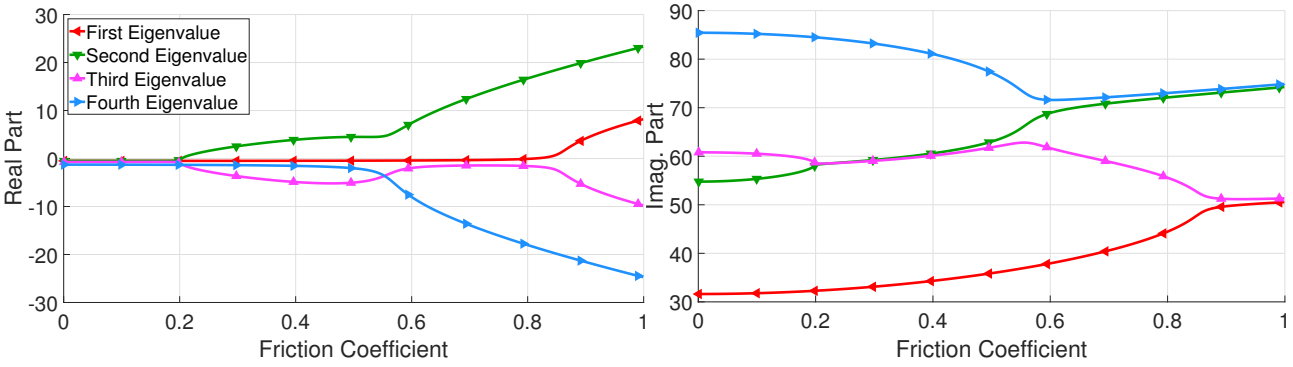
	c_{11} ($\mathrm{N \cdot s \cdot m^{-1}}$)	c_{12} ($\mathrm{N \cdot s \cdot m^{-1}}$)	c_{21} ($\mathrm{N \cdot s \cdot m^{-1}}$)	c_{22} ($\mathrm{N \cdot s \cdot m^{-1}}$)
Nominal	1	1	1	1

To determine the eigensolutions $(\lambda_i, \boldsymbol{\psi}_i)$ of the system, the QEP (Eq. 3.3) is solved with the well-known QZ algorithm, where the mass $\mathbf{M}$, damping $\mathbf{C}$ and stiffness $\mathbf{K}$ matrices are given in (Eq. 3.2).

$$\left(\lambda_i^2 \mathbf{M} + \lambda_i \mathbf{C} + \mathbf{K}\right) \boldsymbol{\psi}_i = \mathbf{0} \tag{3.3}$$

where $\mathbf{K} = \mathbf{K}_{\Sigma} + \mathbf{K}_{NL}$.

The coalescence graph of this model when considering the nominal parameters presented in Table 3.1, Table 3.2 and Table 3.3 is given in Fig. 3.2. Three mode couplings are respectively detected for a friction coefficient of 0.2, 0.57 and 0.84.



(A) Real Part (B) Imaginary Part

FIGURE 3.2: Evolution of the eigenvalue components with respect to the friction coefficient.

The imaginary part evolutions of the model eigenvalues are relatively smooth and present small nonlinearities. Conversely, the real part evolutions of the model eigenvalues are more complex and exhibit non-stationary behaviour. Indeed, for instance, the first real part eigenvalue is constant on the majority of the design space, except when the friction coefficient μ is greater than 0.8.

From all eight functions (real and imaginary parts), the second and the third eigenvalues represent the highest level of complexity. Nonetheless, to efficiently compare the capabilities and limitations of the considered surrogate model techniques, all of these functions are considered in this chapter.

3.2.2 Mode pairing strategy

Traditionally, the mode pairing strategy hinges on the MAC, given by (Eq. 2.44). When considering a multiparametric analysis, the strategy is to compare all eigenvectors $\boldsymbol{\psi}_j$ with a reference set of eigenvectors (for instance, the nominal values of the parameters of the multiparametric analysis, $\widehat{P}_0$) and construct a family $\mathcal{F}_i$ for each mode of the system.

The first limitation comes from the reference set of parameter values $\widehat{P}_0$. To be fully efficient, $\widehat{P}_0$ must be quite close, in the MAC sense, to any set of parameter values $\widehat{P}$ of the design space $\mathcal{D}$.

Fig. 3.3 shows a comparison between two families (circled in gray), generated with two different reference sets of parameter values $\widehat{P}_0$, depicted by a red star. The blue dots represent the evaluated configurations ψ_k and the black arrows represent a pairing with MAC values greater than the threshold. Fig. 3.3(A) shows a good choice for the reference set $\widehat{P}_0$ since the given set is close to any member of the eigenforms ψ_k. Conversely, Fig. 3.3(B) exhibits a poor choice since four configurations are not included into the family $\mathcal{F}$.

(A) Good reference choice (B) Bad reference choice

FIGURE 3.3: Reference choice impact over the family construction.

The FIV problems are highly subjected to mode veering [3], which appears when modes are coupling with each other. The veering is characterised by modes which approach each other and suddenly veer away. Before the veering, the shape of the mode 1 and the mode 2 are respectively given by ϕ_1 and ϕ_2. The modal form at the loci is a linear combination of ϕ_1 and ϕ_2. After the veering, the mode 1 is characterised by the shape ϕ_2 and the mode 2 by ϕ_1. Fig. 3.4 shows an example of veering.

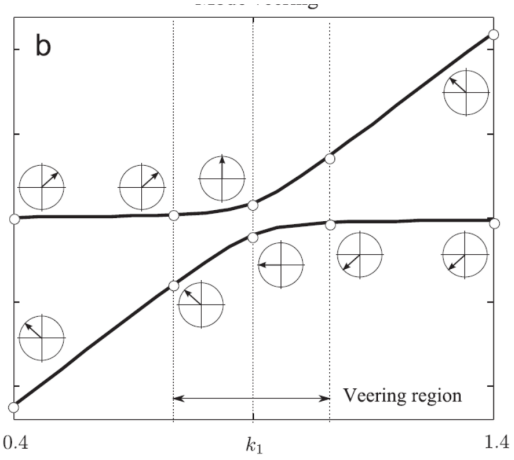


FIGURE 3.4: Veering phenomenon [3]

In practice, choosing a veering shape to initialise the pairing strategy creates only one family and not two. To prevent this issue, the reference has to be a modal form which is not in a superposed state (either before or after the veering). For squeal problems, only the normal contact state (μ=0) has the highest probability to be veering-free. Finally, the other parameters are set to the nominal values, since the uncertainties are always defined around these values.

However, these conditions place the strategy in the state of Fig. 3.3(B), where some configurations are not paired. To counterbalance this issue, the pairing process is executed iteratively: first, the process is computed with the red star reference set $\widehat{P}_0$. Then, the configurations ψ_k, which have just been paired, are added to the reference set $\widehat{P}_0$. This set is thus composed of four reference

configurations for the given example of Fig. 3.3(B). The pairing process is started again until all configurations have successfully been added to the reference set $\widehat{P}_0$ to create the family $\mathcal{F}$.

This strategy is used throughout this thesis for the pairing of the eigenvectors.

3.3 Surrogate modelling theory

3.3.1 Surrogate principle

The main idea behind surrogate modelling is to approximate an expensive function $f : \mathcal{D} \to \mathcal{E}$, where $\mathcal{D} = \prod_{i=1}^{D} [L_i; U_i]$ and $\mathcal{E}$ is the image of $\mathcal{D}$ through the mapping function f ; with a cheaper model. The variable D represents the number of input parameters, whereas L_i and U_i are respectively the lower and upper bounds of the i^{th} dimension. $\mathcal{D}$ and $\mathcal{E}$ are respectively subsets of $\mathbb{R}^D$ and $\mathbb{R}$. To construct an approximation of the considered function, the surrogate model hinges on the acquisition of *a priori* knowledge. This *a priori* knowledge is given by a training set, from which the statistical relationship between the inputs and the outputs is deduced.

Let $\mathcal{T}$ be a training set such as $\mathcal{T} = \{(x_i, y_i) | i \in [\![1, n]\!]\}$, where $\mathbf{x}_i$ denotes an input vector of $(1 \times D)$ dimension and y_i is a scalar corresponding to the image of $\mathbf{x}_i$ through f. In addition, the matrix $\mathbf{X} \in \mathcal{M}_{n,D}$ and the vector $\mathcal{Y} \in \mathbb{R}^n$ contain all the inputs and outputs of the training set $\mathcal{T}$.

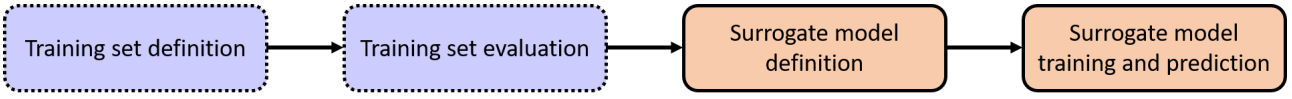


FIGURE 3.5: Surrogate modelling workflow.

Fig. 3.5 shows a flowchart of the surrogate modelling computation. The dotted rectangles represent the definition of the training set using a space-filling strategy, such as the LHS [140], and the evaluation of this training set through a simulator. Then, the surrogate model is chosen, and its parameters are defined. They are commonly divided into two categories: the fixed ones and the optimised ones. The distinction is done through the consideration of several hypotheses about the training set and the given simulator. Finally, the surrogate model is fed with the information from the training set to carry out the predictions.

For a fixed training set, the prediction efficiency and time computation of a surrogate model is clearly dependent on the setting of its parameters. Consequently, the techniques used to tune these data are presented in the following subsections for both surrogate models. Throughout this chapter, the fixed parameters are denoted as hyperparameters, as it can be denoted in the literature, and the optimised parameters are simply denoted as parameters.

3.3.2 Gaussian Process

A GP [141] is a stochastic process $\mathbf{G}$, indexed by a set $\mathbb{X} \subset \mathbb{R}^D$, such as any finite number of random variables is jointly Gaussian: $\forall p \in \mathbb{N}^*, \forall \mathbf{X} = [\mathbf{X}_1, \cdots, \mathbf{X}_p]^T$, $\mathbf{G}(\mathbf{X}) \sim \mathcal{N}(\mu(\mathbf{X}), \mathbf{C}(\mathbf{X}, \mathbf{X}))$ with $\mathbf{G}(\mathbf{X}) = [\mathbf{G}(\mathbf{X}_1), \cdots \mathbf{G}(\mathbf{X}_p)]^{\mathrm{T}}$, where $\mu(\cdot)$ and $\mathbf{C}(\cdot, \cdot)$ respectively stand for the mean and covariance function of the GP. Using a GP as a surrogate model consists in considering the output prediction, denoted by the random variable $\widehat{\mathbf{Y}}$, as a realisation of this stochastic process.

Since $\widehat{\mathbf{Y}}$ is also Gaussian, its density function is described by its first two momenta $\widehat{\mu}$ and $\widehat{s}$, given respectively by (Eq. 3.4) and (Eq. 3.5).

$$\widehat{\mu}(\mathbf{x}_*) = \mu(\mathbf{x}_*) + \mathbf{C}(\mathbf{x}_*, \mathbf{X}) \left[\mathbf{C}(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I}\right]^{-1} (\mathbf{y}(\mathbf{X}) - \mu(\mathbf{X})) \tag{3.4}$$

$$\widehat{s}(\mathbf{x}_*) = \mathbf{C}(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{C}(\mathbf{x}_*, \mathbf{X}) \left[\mathbf{C}(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I}\right]^{-1} \mathbf{C}(\mathbf{X}, \mathbf{x}_*) \tag{3.5}$$

where $\mathbf{x}_*$ is any vector defined in $\mathcal{D}$, μ the mean of the GP, $\mathbf{C}$ its covariance function, σ_n^2 the noise of the data and $\mathbf{I}$, the identity matrix.

Next, considering the covariance function defined by (Eq. 3.6).

$$\mathbf{C}(\mathbf{x}_r, \mathbf{x}_s) = \sigma_k^2 \mathbf{K}(\mathbf{x}_r, \mathbf{x}_s) \tag{3.6}$$

where $(r, s) \in [\![1; n]\!]^2$, σ_k^2 is the signal variance (also called the noise of the kernel) and $\mathbf{K}$ the correlation matrix (also called kernel function or kernel matrix).

Table 3.4 presents the most common choice for the kernel function. They are presented from the least smooth to the smoothest. This property has a direct impact on the approximation of functions. Indeed, the smoothness drives the GP prediction: the more the kernel is non-smooth, the greater the non-linearity of the prediction will be.

Table 3.4
Kernel functions.

	$\mathbf{K}(\mathbf{x}_r, \mathbf{x}_s)$
Matern 3/2	$\left(1 + \sqrt{3}\,\lvert\mathbf{x}_r, \mathbf{x}_s\rvert\right) \exp\left(-\sqrt{3}\,\lvert\mathbf{x}_r, \mathbf{x}_s\rvert\right)$
Matern 5/2	$\left(1 + \sqrt{5}\,\lvert\mathbf{x}_r, \mathbf{x}_s\rvert + \frac{5}{3}\lvert\mathbf{x}_r, \mathbf{x}_s\rvert\right)^2 \exp\left(-\sqrt{5}\,\lvert\mathbf{x}_r, \mathbf{x}_s\rvert\right)$
Exponential quadratic	$\exp\left(-\lvert\mathbf{x}_r, \mathbf{x}_s\rvert^2\right)$

$|\mathbf{x}_r, \mathbf{x}_s| := \sqrt{\left(\sum_{i=1}^{D} \frac{(x_{(r,i)} - x_{(s,i)})^2}{\theta_i^2}\right)}$ represents the weighted Euclidian distance with θ_i being a weight coefficient, called the lengthscale, allowing increasing or reducing the importance of a dimension.

The common way to optimise the GP parameters is through optimising the model evidence $f_{(\mathbf{Y}|\mathbf{X})}$, which is Gaussian, and determining the Maximum Likelihood Estimator (hereby denoted MLE). The optimisation is carried out considering the loss function given by (Eq. 3.7). Typically, the MLE corresponds to parameter values that maximise the probability of the random variable $\widehat{\mathbf{Y}}$.

$$\text{Loss}(\Theta) = -\frac{n}{2}\log(2\pi) - \frac{1}{2}\log\Big(\det(\mathbf{C} + \sigma_n^2\mathbf{I})\Big) - \frac{1}{2}\left(\mathbf{y}^T\left(\mathbf{C} + \sigma_n^2\mathbf{I}\right)^{-1}\mathbf{y}\right) \tag{3.7}$$

where Θ is the vector of lengthscale.

The second term of the sum is the regularisation coefficient that prevents the GP from overfitting. The third term of the sum corresponds to the terms that try to fit the data.

Table 3.5
GP parameter and hyperparameters.

Name of the quantity	Nature of the quantity	Associated variable	Equation
Data noise	Hyperparameter	σ_n^2	(Eq. 3.4)
Mean function	Hyperparameter	μ	(Eq. 3.4)
Signal variance	Hyperparameter	σ_k^2	(Eq. 3.6)
Kernel function	Hyperparameter	$\mathbf{K}$	(Table 3.4)
Lengthscale	Parameter	Θ	(Eq. 3.7)

The different surrogate parameters (hyperparameters and parameters) considered for the GP are summarised in Table 3.5.

3.3.3 Deep Neural Network

A DNN [142] is a deterministic surrogate model composed of artificial neurons. An artificial neuron corresponds to a coarse and over-simplified version of a brain neuron, called a perceptron [143], and aims at mimicking the biological phenomenon that generates a signal between two neurons. A DNN is then obtained by connecting perceptrons with each other and organising them in layers.

Fig. 3.6 shows an example of a neural network architecture. In this case, the input dimension d is 2 and the output dimension is 1. There are two intermediate layers, each composed of four neurons. The input neurons on the left are associated with the input data (labelled X_s^d), whereas the output neuron on the right is associated with the output data (labelled $\widehat{y}_s^1$). The value of a neuron on the intermediate layers is determined by carrying the linear combinations of the connected neurons from the previous layer and then applying an activation function ((Eq. 3.8)).

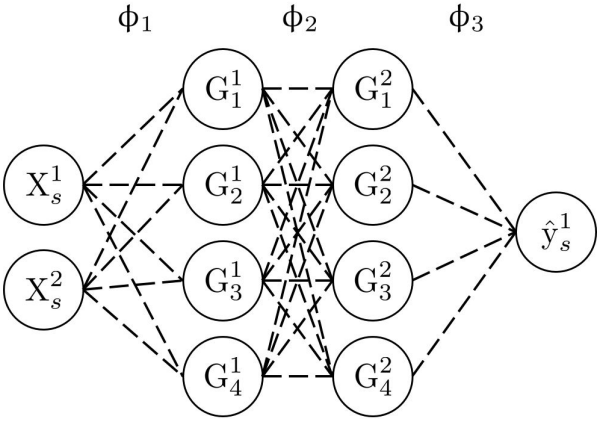


FIGURE 3.6: Structure of a neural network.

$$G_i^{D^{(i)}} = \phi_i(\mathbf{w}_{i-1}^T \mathbf{G}^{i-1}) \tag{3.8}$$

where $i \in [\![1; p]\!]$, p is the number of hidden layers, $D^{(i)}$, the number of neurons per hidden layer, ϕ_i, the activation function of the i^{th} hidden layer, $\mathbf{w}_i = [w_i^1 \ \cdots \ w_i^j \ \cdots \ w_i^{D^{(i)}}]^T$, the weight vector associated with the i^{th} hidden layer, and $\mathbf{G} = [G_{i-1}^1 \ \cdots \ G_{i-1}^j \ \cdots \ G_{i-1}^{D^{(i-1)}}]^T$, the neuron vector associated with the previous hidden layer.

Table 3.6
Common activation functions.

	$\phi(x)$
ReLU	$\max(0, x)$
Hyperbolic tangent	$\tanh(x)$
Logistic	$\frac{1}{1+\exp(-x)}$

The most common activation functions used in DNN are summarised in Table 3.6. These functions coupled with the architecture of the neural network allow capturing the full complexity of the considered functions. Nielsen [144] proved that one characteristic of this surrogate model is that it is a universal approximator. Nevertheless, this strength is also one of its biggest weaknesses. Indeed, this capacity to increase the complexity of the learning via the increase of neurons in hidden layers can lead to overfitting [145]. Typically, this overfitting phenomenon is associated with weight values which are too high. One way to try to remove this flaw is penalising the weight values with a regularisation coefficient, such as in (Eq. 3.9).

$$\text{Loss}(\mathbf{W}) = \frac{1}{2n}\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2 + \frac{\alpha}{2}\sqrt{\left(\sum_{k=1}^{p}\sum_{j=1}^{D^{(k)}} W_{j,k}^2\right)} \tag{3.9}$$

where $\mathbf{W} = [\mathbf{w}_1 \ \cdots \ \mathbf{w}_k \ \cdots \ \mathbf{w}_p]$ is the coefficient matrix for which each column corresponds to the coefficient vector from the i^{th} hidden layer and α the regularisation coefficient.

(Eq. 3.9) gives the loss function used for the neural network. This equation is equivalent to the least squares method with an additional contribution which allows controlling overfitting. This control is done by adding all the weight values together and multiplying them with a coefficient which modifies the impact of the sum on the Loss function. Typically, from the observations made on these considered functions, a value of α lower than 0.005 corresponds to a small regularisation, between 0.005 and 0.05, to a medium regularisation and a value above 0.05 corresponds to a strong regularisation. This hyperparameter has to be tuned cautiously since a high value prevents the surrogate model from going through the training set points, inducing important error values, and a small value increases the chance of overfitting.

The optimisation of the weight values linking the different neurons is done via backpropagation procedure. First, the weight values are initialised randomly. Then, the Loss function is computed and minimised using a classical gradient descent algorithm. In this research, since the data is scarce, the optimisation is ensured by an L-BFGS (Limited-memory Broyden-Fletcher-Goldfard-Shanno) algorithm [146–148].

Once the weight matrix is optimised, the prediction is carried with (Eq. 3.10).

$$\widehat{y}(\mathbf{x}_*) = \phi_p(\phi_{p-1}(\dots(\phi_1(\mathbf{w}_1\mathbf{x}_*))\dots)) \tag{3.10}$$

Table 3.7 summarises the different quantities considered in the DNN surrogate model.

Table 3.7
DNN parameters and hyperparameters.

Name of the quantity	**Nature of the quantity**	**Associated variable**	**Equation**
Activation function	Hyperparameter	ϕ	(Table 3.6)
Weight matrix	Parameter	$\mathbf{W}$	(Eq. 3.8)
Number of hidden layers	Hyperparameter	p	(Eq. 3.10)
Number of neurons per hidden layer	Hyperparameter	$D^{(i)}$	(Eq. 3.8)
Regularisation coefficient	Hyperparameter	α	(Eq. 3.9)

3.3.4 Deep Gaussian Process

The DGP is a recent class of surrogate models [131, 132, 149] and is inspired by the Deep Learning theory. The main idea is to capture complex variations of the underlying function by decomposing the information embedded in the training set, *i.e* through nested structures. Thus, the DGP hinges on neurons and layers, being a stack of GPs.

Hence, a random vector $\mathbf{H}^{\ell}$ (Eq. 3.12) is introduced for each layer of the DGP. This random vector is defined by a GP $\mathbf{G}^{\ell}$ given by (Eq. 3.11). This prior hinges on the fundamental assumption that the current layer is only conditionally dependent on the previous layer random vector $\mathbf{H}^{\ell-1}$.

$$\mathbf{G}^{\ell}(\mathbf{H}^{\ell-1}) \sim \mathcal{N}(\mu(\mathbf{H}^{\ell-1}), C(\mathbf{H}^{\ell-1}, \mathbf{H}^{\ell-1})) \tag{3.11}$$

for $\ell \in [\![1, L]\!]$, $\mathbf{H}^0 = \mathbf{X}$ and L is the number of hidden layers. μ and $\mathbf{C}$ are respectively the mean and the covariance function of the ℓ^{th} layer.

$$\mathbf{H}^{\ell} = \begin{bmatrix} H^{\ell}_{1,1} & \cdots & H^{\ell}_{1,j} & \cdots & H^{\ell}_{1,D^{(\ell)}} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ H^{\ell}_{i,1} & \cdots & H^{\ell}_{i,j} & \cdots & H^{\ell}_{i,D^{(\ell)}} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ H^{\ell}_{n,1} & \cdots & H^{\ell}_{n,j} & \cdots & H^{\ell}_{n,D^{(\ell)}} \end{bmatrix} \tag{3.12}$$

where $D^{(\ell)}$ is the number of neurons per hidden layer.

These priors are quite expensive to compute since they involve an inversion of the covariance matrix $\mathbf{C}(\mathbf{H}^{\ell-1}, \mathbf{H}^{\ell-1}))$ which is $\mathcal{O}(n^3)$ for each layer of the surrogate model. To reduce the computational cost of the matrix inversion, the pseudo-inputs [150] (also called inducing points in the literature) are introduced. The main idea behind this trick is to increase the probability space with non-observed points which will, to a certain extent, summarise the data and reduce the cost of the inversion to $\mathcal{O}(m^3)$, where m is the number of pseudo-inputs. For significant effects, m has to be a lot smaller than n when considering large training sets. If the number of samples is small, this number can be chosen with more flexibility.

For each layer, a random vector $\mathbf{U}^{\ell}$ and a set of pseudo-inputs, denoted $\mathbf{Z}^{\ell}$, are introduced. The random vector $\mathbf{U}^{\ell}$ is defined using the aforementioned GP ((Eq. 3.13)).

$$\mathbf{G}^{\ell}(\mathbf{Z}^{\ell-1}) \sim \mathcal{N}(\mu(\mathbf{Z}^{\ell-1}), C(\mathbf{Z}^{\ell-1}, \mathbf{Z}^{\ell-1})) \tag{3.13}$$

where $\ell \in [\![1, L]\!]$ and $\mathbf{Z}^{\ell} = [z^{\ell}_{1,1}, ..., z^{\ell}_{m,D^{(\ell)}}]^T$.

Fig. 3.7 graphically exhibits how the different random variables interact with each other. The straight green circles correspond to variables which are strictly Gaussian. The dashed orange circles correspond to variables that are not conditionally Gaussian, and the blue rectangles symbolise the conditioning process. In addition, each layer is conditionally independent to the others.

From this additional assumption, the model evidence $f_{(\mathbf{Y}|\mathbf{H}^0)}$ can be written as (Eq. 3.14) and similarly to the GP, it is optimised to obtain optimal values for the parameters.

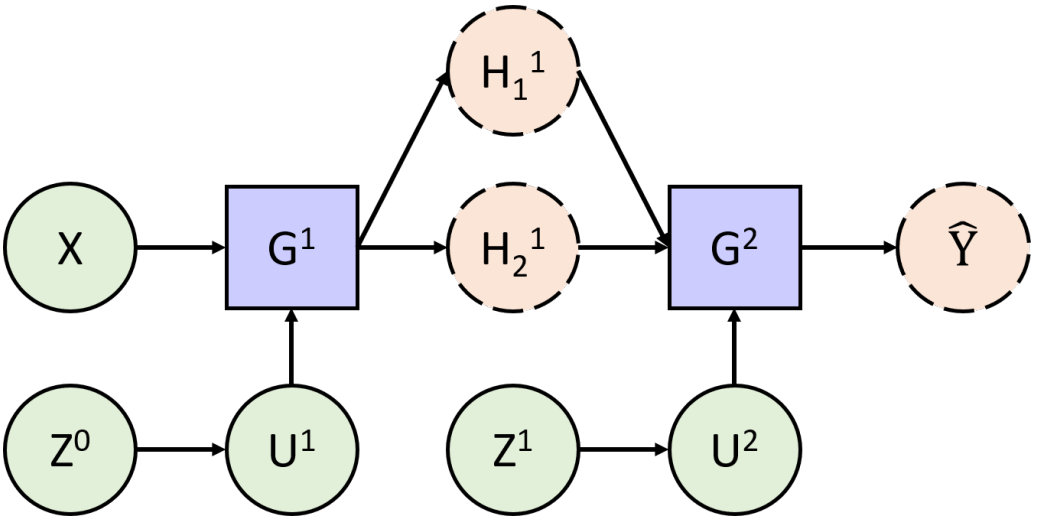


FIGURE 3.7: Structure of a DGP.

$$\begin{aligned}\log\left(f_{(\mathbf{Y}|\mathbf{H}^0)}(y)\right) &= \log\left(\int_{\mathbf{U}^\ell}\int_{\mathbf{H}^\ell} f_{(\mathbf{Y},\mathbf{H}^\ell,\mathbf{U}^\ell|\mathbf{H}^0)}(y,h^\ell,u^\ell|h^0)d\mathbf{H}^\ell d\mathbf{U}^\ell\right)\\ &= \log\left(\int_{\mathbf{U}^\ell}\int_{\mathbf{H}^\ell} f_{(\mathbf{Y}|\mathbf{H}^L)}f_{(\mathbf{H}^\ell|\mathbf{H}^{\ell-1},\mathbf{U}^\ell)}f_{(\mathbf{U}^\ell)}d\mathbf{H}^\ell d\mathbf{U}^\ell\right)\end{aligned} \tag{3.14}$$

where $\mathbf{H}^\ell := \{\mathbf{H}_\ell\}_{\ell=1}^L$, $\mathbf{U}^\ell = \{\mathbf{U}_\ell\}_{\ell=1}^L$, $\int_{\mathbf{H}^\ell} := \int_{\mathbf{H}^L}\cdots\int_{\mathbf{H}^1}$, $\int_{\mathbf{U}^\ell} := \int_{\mathbf{U}^L}\cdots\int_{\mathbf{U}^1}$ and $f_{(\mathbf{H}^\ell|\mathbf{H}^{\ell-1},\mathbf{U}^\ell)}f_{(\mathbf{U}^\ell)} := \prod_{\ell=1}^{L} f_{(\mathbf{H}_\ell|\mathbf{H}_{\ell-1},\mathbf{U}_\ell)}f_{(\mathbf{U}_\ell)}$.

This model evidence can be decomposed into two parts. The first member of the integrand $f_{(\mathbf{Y}|\mathbf{H}^L)}$ corresponds to the likelihood of the data. The second member $f_{(\mathbf{H}^\ell|\mathbf{H}^{\ell-1},\mathbf{U}^\ell)}f_{(\mathbf{U}^\ell)}$ is the DGP prior. Unfortunately, this prior is not analytically tractable due to the successive inversion of the covariance matrix $\mathbf{C}(\mathbf{H}^{\ell-1},\mathbf{H}^{\ell-1})$ which prevents the computation of the conditional density function $f_{(\mathbf{H}^\ell|\mathbf{H}^{\ell-1},\mathbf{U}^\ell)}$.

Following a variational inference approach, the variational distribution $q_{(\mathbf{H}^\ell,\mathbf{U}^\ell)}$, reminded in (Eq. 3.15), is introduced to remove this issue of tractability. The representativeness of the distribution is maintained over the conditional density function $f_{(\mathbf{H}^\ell|\mathbf{H}^{\ell-1},\mathbf{U}^\ell)}$ and the approximation is carried out on the inducing variables.

$$q_{(\mathbf{H}^\ell,\mathbf{U}^\ell)} = \prod_{\ell=1}^{L} f_{(\mathbf{H}^\ell|\mathbf{U}^\ell,\mathbf{H}^{\ell-1})}q_{(\mathbf{U}^\ell)} \tag{3.15}$$

where $q_{(\mathbf{U}^\ell)}$ is the variational distribution of $\mathbf{U}^\ell$ and is Gaussian distributed with $(\mathbf{U}^\ell)_q \sim \mathcal{N}(\mathbf{m}^\ell,\mathbf{S}^\ell)$. The subscript q refers to the variational nature of the distribution describing the random variable. $\mathbf{m}^\ell$ and $\mathbf{S}^\ell$ are the mean and the variance of $\mathbf{U}^\ell$.

The inducing variables can be marginalised to obtain the variational distribution of $\mathbf{H}^{\ell}$ which is still Gaussian distributed with the mean given by (Eq. 3.16) and the variance given by (Eq. 3.17).

$$\tilde{\boldsymbol{\mu}}^{\boldsymbol{\ell}} = \mu(\mathbf{H}^{\ell-1}) + \mathbf{A}(\mathbf{m}^{\ell} - \mu(\mathbf{Z}^{\ell})) \tag{3.16}$$

$$\tilde{\boldsymbol{\Sigma}}^{\boldsymbol{\ell}} = \mathbf{C}_{HH} - \mathbf{A}(\mathbf{C}_{ZZ} - \mathbf{S}^{\ell})\mathbf{A}^{T} \tag{3.17}$$

where $\mathbf{A} = \mathbf{C}_{HZ}\mathbf{C}_{ZZ}^{-1}$, $\mathbf{C}_{HZ} = \mathbf{C}(\mathbf{H}^{\ell-1}, \mathbf{Z}^{\ell-1})$, $\mathbf{C}_{ZZ} = \mathbf{C}(\mathbf{Z}^{\ell-1}, \mathbf{Z}^{\ell-1})$ and $\mathbf{C}_{HH} = \mathbf{C}(\mathbf{H}^{\ell-1}, \mathbf{H}^{\ell-1})$

Thus, due to the conditional dependence of each layer given the previous layer, the computation of a realisation of the density function $(\mathbf{H}^{L})$ is carried out by propagating the given input $\mathbf{x}$ through each layer.

Combining (Eq. 3.15) with (Eq. 3.14) gives the logarithmic mathematical expectation of the model evidence over the variational distribution of inducing variables ((Eq. 3.18)).

$$\begin{aligned}
\log\left(f_{(\mathbf{Y}|\mathbf{H}^{0})}(y)\right) &= \log\left(\int_{\mathbf{U}^{\ell}}\int_{\mathbf{H}^{\ell}} f_{(\mathbf{Y}|\mathbf{H}^{L})} f_{(\mathbf{H}^{\ell}|\mathbf{H}^{\ell-1},\mathbf{U}^{\ell})} f_{(\mathbf{U}^{\ell})} \frac{q_{(\mathbf{H}^{\ell},\mathbf{U}^{\ell})}}{q_{(\mathbf{H}^{\ell},\mathbf{U}^{\ell})}} d\mathbf{H}^{\ell} d\mathbf{U}^{\ell}\right) \\
&= \log\left(\int_{\mathbf{U}^{\ell}}\int_{\mathbf{H}^{\ell}} q_{(\mathbf{H}^{\ell},\mathbf{U}^{\ell})} f_{(\mathbf{Y}|\mathbf{H}^{L})} \frac{f_{(\mathbf{U}^{\ell})}}{q_{(\mathbf{U}^{\ell})}} d\mathbf{H}^{\ell} d\mathbf{U}^{\ell}\right) \\
&= \log\left(\mathbb{E}_{(\mathbf{H}^{\ell},\mathbf{U}^{\ell})_q}\left[f_{(\mathbf{Y}|\mathbf{H}^{L})} \frac{f_{(\mathbf{U}^{\ell})}}{q_{(\mathbf{U}^{\ell})}}\right]\right)
\end{aligned} \tag{3.18}$$

Since the logarithmic function is a concave function, the Jensen's inequality [151] can be used to obtain a lower bound of the model evidence.

$$\begin{aligned}
\log\left(f_{(\mathbf{Y}|\mathbf{H}^{0})}(y)\right) &\geq \mathbb{E}_{(\mathbf{H}^{\ell},\mathbf{U}^{\ell})_q}\left[\log\left(f_{(\mathbf{Y}|\mathbf{H}^{L})} \frac{f_{(\mathbf{U}^{\ell})}}{q_{(\mathbf{U}^{\ell})}}\right)\right] \\
&\geq \mathbb{E}_{(\mathbf{H}^{\ell},\mathbf{U}^{\ell})_q}\left[\log\left(f_{(\mathbf{Y}|\mathbf{H}^{L})}\right)\right] + \mathbb{E}_{(\mathbf{H}^{\ell},\mathbf{U}^{\ell})_q}\left[\log\left(\frac{f_{(\mathbf{U}^{\ell})}}{q_{(\mathbf{U}^{\ell})}}\right)\right] \\
&\geq \mathbb{E}_{(\mathbf{H}^{L})_q}\left[\log\left(f_{(\mathbf{Y}|\mathbf{H}^{L})}\right)\right] + \mathbb{E}_{(\mathbf{U}^{\ell})_q}\left[\log\left(\frac{f_{(\mathbf{U}^{\ell})}}{q_{(\mathbf{U}^{\ell})}}\right)\right] \\
&\geq \mathbb{E}_{(\mathbf{H}^{L})_q}\left[\log\left(f_{(\mathbf{Y}|\mathbf{H}^{L})}\right)\right] - \sum_{\ell=1}^{p} D_{KL}\left(q_{(\mathbf{U}^{\ell})} || f_{(\mathbf{U}^{\ell})}\right)
\end{aligned} \tag{3.19}$$

where D_{KL} is the Kullback-Leibler divergence between the variational and the true distribution of $\mathbf{U}^{\ell}$.

The lower bound given by (Eq. 3.19) is called the Efficient Lower BOund (hereby denoted ELBO) and provides a tight bound for the model evidence. The Kullback-Leibler divergence is in closed form, since both distributions are Gaussian distributed. Nevertheless, the tractability of the first term is dependent of the type of likelihood chosen. Indeed, for the Gaussian and Poisson likelihood, this term can be determined analytically. Otherwise, a Gaussian quadrature or a Monte Carlo sampling can be used. For either case, the computation of the first term is done by propagating each instance into the surrogate model.

Considering the expectation of the likelihood given the variational distribution, the first ELBO term becomes (Eq. 3.20), since each sample from the training set $\mathcal{T}$ is independent.

$$\mathbb{E}_{(\mathbf{H}^L)_q}\left[\log\left(f_{(\mathbf{Y}|\mathbf{H}^L)}\right)\right] = \mathbb{E}_{(\mathbf{U}_s^L)_q}\left[\log\left(f_{(y_s|\mathbf{H}_s^L)}\right)\right] \tag{3.20}$$

where $(\mathbf{U}_s^L)_q$ is a realisation of the variational distribution of $\mathbf{U}^L$ for the sample $\mathbf{x}_s$, y_s is the image of $\mathbf{x}_s$ through the function f and $(\mathbf{H}_s^L)_q$ is a realisation of the distribution of $\mathbf{H}^L$ for the sample $\mathbf{x}_s$.

In addition, the re-parameterisation trick, introduced by Rezende et al. [152] and Kingma et al. [153], in the context of Bayesian Inference, is considered and given by (Eq. 3.21). This trick allows a better optimisation of variational distributions.

$$\mathbf{H}_s^\ell = \tilde{\mu}(\mathbf{H}_s^{\ell-1}) + \epsilon_s^\ell \sqrt{\tilde{\Sigma}(\mathbf{H}_s^{\ell-1}, \mathbf{H}_s^{\ell-1})} \tag{3.21}$$

where ϵ_s^ℓ is a Gaussian-distributed random variable given by $(\epsilon_s^\ell) \sim \mathcal{N}(0, \mathbf{I})$ with $\mathbf{I}$ being the identity matrix.

Considering this trick and (Eq. 3.20), the ELBO reduces to (Eq. 3.22).

$$\text{ELBO} = \mathbb{E}_{(\epsilon_s^L)}\left[\log\left(f_{(y_s|\mathbf{H}_s^L)}\right)\right] - \sum_{\ell=1}^{L} D_{KL}\left(q_{(\mathbf{U}^\ell)} || f_{(\mathbf{U}^\ell)}\right) \tag{3.22}$$

where $\mathbf{H}_s^L$ is determined with (Eq. 3.21). The analytical forms for both terms are given in (Eq. B.1) and (Eq. B.2).

Once the parameters are optimised, the prediction is done by propagating vectors $\mathbf{x}_*$ through the surrogate model with (Eq. 3.23). Since $(\mathbf{H}^L)$ is not Gaussian distributed and depends on the ϵ_s^ℓ random variable, T samples are drawn from its distribution and then averaged.

$$q_{(\mathbf{H}_*^L)} = \frac{1}{T}\sum_{t=1}^{T} q_{(\mathbf{H}_t^L)} \tag{3.23}$$

where $q_{(\mathbf{H}_t^L)}$ is the t^{th} realisation of the variational distribution of $(\mathbf{H}^L)$.

Table 3.8
DGP parameters and hyperparameters.

Name of the quantity	**Nature of the quantity**	**Associated variable**	**Equation**
Hidden layer number	Hyperparameter	p	(Eq. 3.11)
Neuron number	Hyperparameter	$D^{(\ell)}$	(Eq. 3.6)
Pseudo-inputs number	Hyperparameter	m	(Eq. 3.13)
Kernel	Hyperparameter	$\mathbf{K}$	(Table 3.4)
Likelihood	Hyperparameter	$f_{(\mathbf{Y}\|\mathbf{H}^L)}$	(Eq. 3.14)
Lengthscale	*Parameter (Deterministic)*	Θ^ℓ	(Eq. 3.6)
Pseudo-inputs locations	*Parameter (Deterministic)*	$\mathbf{Z}^\ell$	(Eq. 3.13)
Signal variance	*Hyperparameter*	$\sigma_k^{2,\ell}$	(Eq. 3.6)
Data noise	*Hyperparameter*	σ_n^2	(Eq. B.1)
Mean function	*Hyperparameter*	μ^ℓ	(Eq. 3.16)
Variational mean	*Parameter (Variational)*	$\mathbf{m}^\ell$	(Eq. 3.15)
Variational covariance matrix	*Parameter (Variational)*	$\mathbf{S}^\ell$	(Eq. 3.15)

Table 3.8 summarises the different quantities considered in the DGP surrogate model. The hyperparameters and parameters in italics are repeated for each layer. The signal variance and data noise are fixed, invoking the same reasons as for the GP.

3.4 Analysis of the performance of the surrogate for a one-dimensional problem

3.4.1 Preamble

The aim of this section is to account for the performance of the surrogate models presented in Section 3.3, when considering the approximation of the four complex eigenvalues of the Double Hulten. To do so, the effects of the hyperparameter setting are thoroughly studied.

The most challenging one-dimensional configuration is considered by taking the friction coefficient as an input variable. This parameter generates non-linear behaviours and asymmetries in the contact matrices, and is, thus, the most complex parameter to manage.

Finally, for engineering applications, small training sets (about 200 to 300 samples max.) are mandatory to maintain the computational cost compliant with industrial process. Hence, the number of samples used to train the surrogate models is restrained following this limitation. The samples number is parameterised with regard to the number of dimension to give a reference for the comparison of scenarios with different number of dimensions.

3.4.1.1 Toolboxes and hyperparameter setting

The current applications are based on different Python toolboxes. The GP model is constructed with the GPFlow toolbox [154] and its parameters are optimised using the LBFGS algorithm [146–148]. Next, the DGP model is generated with the Doubly Stochastic toolbox [132] and its deterministic parameters, namely the location of the pseudo-points and the lengthscales of the kernel of each layer, are optimised using Adam [155]. With regard to the variational parameters, namely the mean and variance matrix of each layer, they are optimised using the Natural Gradient descent method, introduced by Salimbeni et al. [156]. For the DNN, the Scikit-Learn toolbox [157] is used, and the optimisation is carried out with the LBFGS algorithm.

Some of the hyperparameters, reminded in Table 3.5, Table 3.7 and Table 3.8, are put aside in this chapter and are not studied based on the following reasons. Since the solver is exact and the data is standardised, there is no need to optimise the data noise and signal variance with respect to the loss function. Following Ginsbourger's conclusions [158], for the GP and the DGP, the mean function is not considered and taken to be Zero function. As for the number of DGP pseudo-inputs, Salimbeni's formalism [132] does not allow inferring an optimal number, but rather to impose a user selection of this number. Concerning the likelihood, it is also user defined and carries the hypothesis on the relationship between the output and the realisation of the DGP. Finally, for the DNN, the regularisation is traditionally set to 0.0001 [157].

Table 3.9 summarises the fixed hyperparameters of the study with their associated values.

Table 3.9
Fixed hyperparameters for the GP, the DNN and the DGP.

Surrogate model	Hyperparameter name	Symbol	Equations	Value
GP	Signal variance	σ_{k}^2	(Eq. 3.6)	1
GP	Data noise	σ_{n}^2	(Eq. 3.4)	10^{-6}
GP	Mean function	μ	(Eq. 3.4)	Zero ()
DNN	Regularisation	α	(Eq. 3.9)	0.0001
DGP	Pseudo-inputs number	m	(Eq. 3.13)	n
DGP	Likelihood	$f_{(Y\|\mathbf{H}^L)}$	(Eq. 3.14)	Gaussian
DGP	Signal variance	σ_k^ℓ	(Eq. 4.3)	1
DGP	Data noise	σ_n^2	(Eq. B.1)	10^{-3}
DGP	Mean function	μ^ℓ	(Eq. 3.16)	Zero ()

Similarly, the parameters of the optimisers are given in Table 3.10.

3.4.1.2 Experimental protocol

The approximated functions are given in Section 3.2.1 and correspond to the real and imaginary parts of the four eigenvalues of the Double Hulten. As mentioned earlier, the friction coefficient is considered variable, while the other parameters are fixed, following the values of Table 3.9. Ten

Table 3.10
Optimisation parameters for the DGP.

Optimizer	**Name**	**Value**
Adam	Step	10^{-3}
	Learning rate (Alpha)	0.8
	First moment decay rate (Beta)	0.8
	Epsilon	10^{-8}
Natural Gradient	Step	10^{-3}

randomly distributed training sets are generated for three different sizes: 6, 10 and 15 samples per dimension, using an LHS.

Considering the surrogate model definition and the remaining hyperparameters (Table 3.11), several values are tested to evaluate the performance of each surrogate model depending on its setting and the number of samples used for training.

Table 3.11
Studied hyperparameters for the three surrogate models.

Surrogate	Kernel/Activation function	Hidden layers	Neurons
GP	RBF, Matern 3/2 (M32), Matern 5/2 (M52)	-	-
DNN	ReLU, Tanh, Logistique	$[\![1, 4]\!]$	{50,100,150,200,250,300}
DGP	Matern 3/2 (M32), Matern 5/2 (M52)	$[\![1, 4]\!]$	1-3-7-10

For the GP, only the kernel is taken into account here. For the DGP and DNN, the deep nature of these surrogate models requires a more complex architecture than the one of the GP. Thus, a parametric study of the definition of hidden layers and associated neurons per layer is carried out in addition to the analysis of the kernel/activation functions. The RBF function is put aside for the DGP due to non-convincing results. The DGP architecture values were chosen with regard to the literature [131, 132, 135] and the DNN was set according to many tests.

For the GP and DGP, the lengthscale is initialised at a value of 1, while the weight of the DNN are randomly initialised. These quantities are then optimised, considering the optimisation algorithms, presented in Section 3.4.1.1.

As the purpose is here to assess the surrogate performance over the entire model, the whole spectrum is considered. Nonetheless, each real and imaginary parts of the eigenvalue is assumed to be independent, inducing one surrogate training per quantity of interest. A CRMSE criterion, given by (Eq. 3.24), is used to account for the whole spectrum approximation capabilities. To validate our different predictions, a validation set of 10,000 values is defined.

$$CRMSE = \sqrt{\frac{\sum_{j=1}^{t}\sum_{i=1}^{n_{\text{test}}}(y_{ij} - \widehat{y}_{ij})^2}{t \times n_{\text{test}}}} \tag{3.24}$$

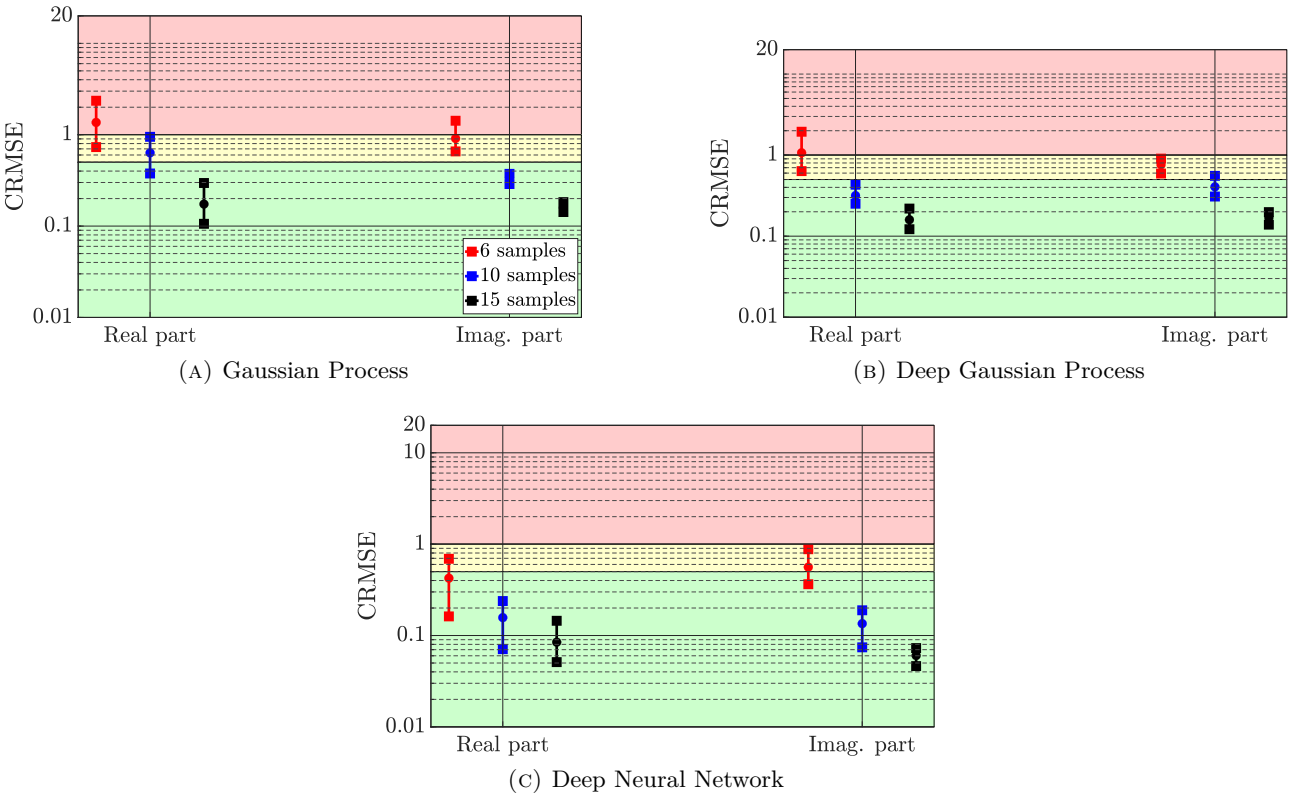
where n_{test} is the number of test values and t, the number of considered eigenvalues to approximate. Then, y_{ij} and $\widehat{y}_{ij}$ are the reference and the surrogate approximation values for the i^{th} sample of the j^{th} eigenvalue.

3.4.2 Best hyperparameter setting for small training sets

3.4.2.1 Performance overview

Fig. 3.8 presents the CRMSE range of variation for the ten randomly distributed training sets – the min-max bounds are given with squares, while the mean performance over the training sets is displayed with a circle. In addition, the results considering the sample density (6, 10 and 15 samples) are respectively shown in red, blue and black interval performance. For all the tested hyperparameter setting, the CRMSE value is compared, and the configurations associated with the lowest CRMSE value is denoted the best configuration. This analysis is carried out on the averaged CRMSE over the training set.

The real and imaginary approximation performance are displayed on the left and right side of each figure. Finally, a color map is added in the background to denote the interpretation of the CRMSE: a CRMSE value greater than 1 is associated with a bad performance, while a CRMSE value lower than 0.5 is linked to a good performance. Between these two bounds, the results are mitigated and cannot be considered efficient nor bad.



(A) Gaussian Process

(B) Deep Gaussian Process

(C) Deep Neural Network

FIGURE 3.8: CRMSE range of variations for the best hyperparameter setting of the three considered surrogate models.

Observations

First, considering the best hyperparameter setting, the best surrogate model is clearly the DNN: whatever the training set size, the mean performance over the ten training sets is associated with a good CRMSE for the real part approximation, or at least a mitigated CRMSE (6 samples) and then a good CRMSE (10 to 15 samples) for the imaginary part.

With regard to the DGP, the overall performance is slightly better than for the GP. Indeed, for 6 samples, both surrogate models almost perform identically. For the approximation of the real parts with 10 samples, the DGP provides a better approximation since both mean and maximum CRMSE are linked to a good CRMSE value (resp. 0.35 and 0.48). Conversely, the GP slightly performs worse by providing a mean CRMSE of 0.61 and a maximum CRMSE of roughly 1. For other scenarios, the results differ few.

Finally, the real part approximations are more sensitive to the sample distribution, especially for the GP and the DNN. Indeed, the min-max bounds of the first surrogate roughly spread about 0.3 CRMSE points, while the latter bounds spread about 0.6. Conversely, the DGP allows more stable approximations since the bounds spread are roughly 0.1 CRMSE points.

Table 3.12 exhibits the hyperparameter settings that led to the mean CRMSE, shown in Fig. 3.8. These best settings are determined by taking the mean trend of the surrogate model performance over all the considered training sets. Typically, a hyphen is put for surrogate models where no particular mean trend can be emphasised, highlighting the instability of the given surrogate model toward its hyperparameter setting. Moreover, for the DNN and the DGP, the values in the brackets give the number of hidden layers and neurons. The bold text of Table 3.12 highlights the following observations about the best hyperparameter setting.

Table 3.12
Best hyperparameter setting of GP, DGP and DNN.

Surrogate model	Training set size	6 points		10 points		15 points	
	Approximated function	Real	Imag.	Real	Imag.	Real	Imag.
GP	Kernel	**M32**	**M32**	M52	**M32**	**M32**	**M32**
DGP	Kernel	**M32**	**M32**	**M32**	**M32**	**M32**	**M32**
	Architecture	(4,10)	**(2,10)**	**(2,7)**	**(3,7)**	**(2,10)**	(1,10)
DNN	Activation function	ReLU	ReLU	ReLU	Tanh	Tanh	Tanh
	Architecture	(4,150)	-	(4,300)	-	-	-

The GP and DGP exhibits the highest performance when the **Matern 3/2** is used, whatever the considered approximated functions. For the architecture of the DGP, the best hyperparameter setting is almost always associated with a **medium deep structure**. Finally, for the DNN, the results are rather mitigated. Indeed, the performance for the small density sample distribution (real and imaginary approximation) and the medium density sample distribution is maximised when the **ReLU activation function** is considered; however, for the other considered scenarios, the **Tanh** is

needed. In addition, no particular trend has been highlighted for the imaginary approximation with regard to the hidden layer and neurons settings of the DNN.

Discussions

The results of the kernel functions are in a way relevant with the mathematical properties of these functions. Indeed, the Matern 3/2 is only once differentiable, making it highly rough. The roughness of this function allows the surrogate model to deal with higher non-linearities than the RBF, which is infinitely differentiable and therefore highly smooth. This property is, thus, highly interesting while dealing with friction-induced vibration problems due to the non-linearities involved in the models.

For the DNN, the hyperparameter setting instability is one of the main flaw of this surrogate model. Indeed, this issue enforces the user to evaluate numerous configurations to find the best one. This causes that no conclusion can be drawn towards the hyperparameter setting of this surrogate. Some authors [159, 160] tried to tackle this issue, but it is still an open research.

3.4.2.2 Sample distribution density effect

Fig. 3.9 proposes an assessment of the approximation quality with respect to the sample distribution density for a given training set. The first row of Fig. 3.9 (from (A) to (C)) corresponds to the approximation with 6 samples; the second row of Fig. 3.9 (from (D) to (F)), with 10 samples; and the final row of Fig. 3.9 (from (G) to (I)), with 15 samples. The hyperparameters used for training the surrogate models are given in Table 3.12.

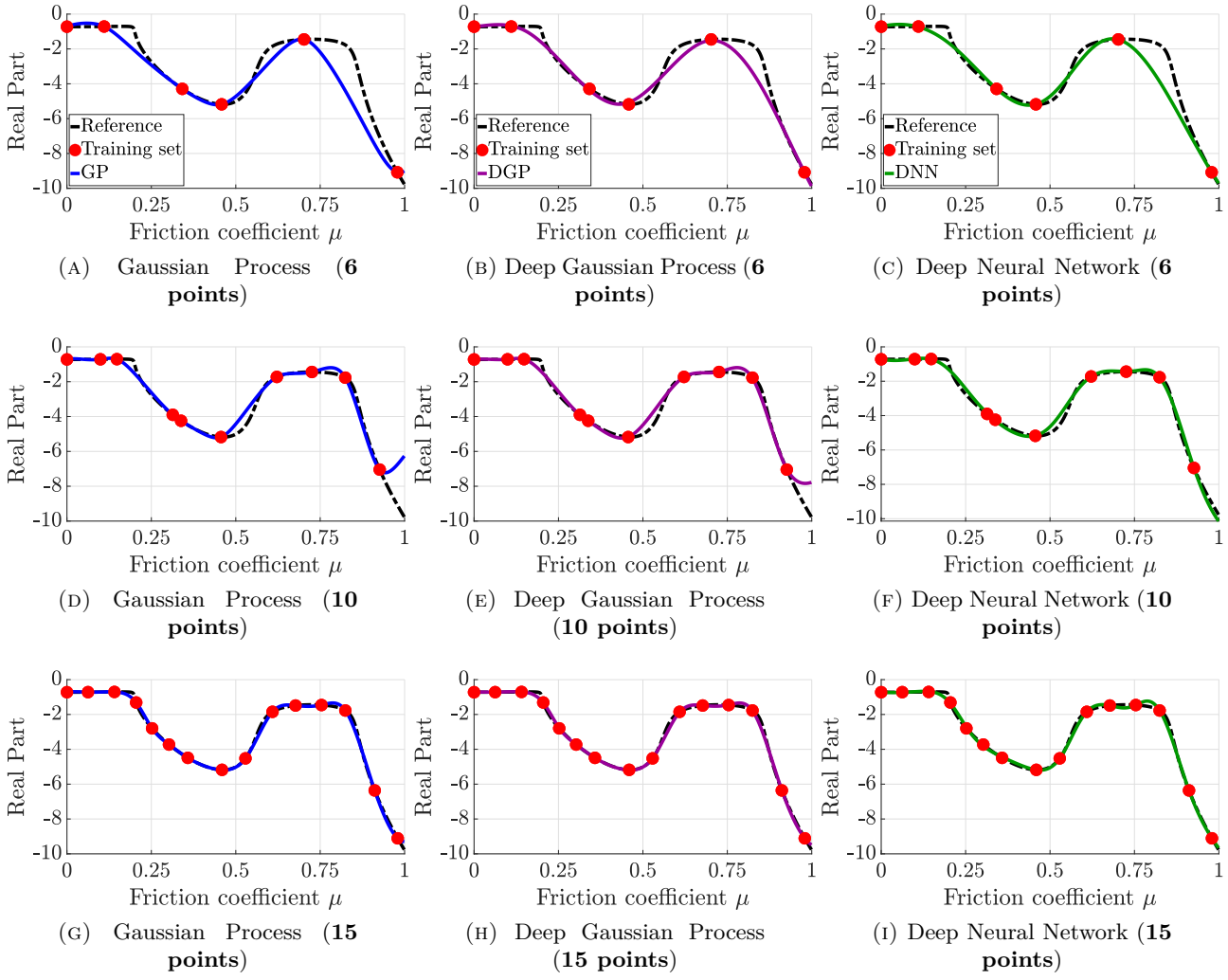
First, whatever the training set size, the prediction of each surrogate model is merely identical, except for the 10 samples (Fig. 3.9 (D) to (F)) where the approximations are slightly different for a friction coefficient between 0.8 and 1.

Moreover, as the use of a 6 samples training set allows a coarse prediction of the behaviour of the considered function, a 10 samples training set gives a rather good approximation. The 15 samples training set generates a perfect prediction, but the additional computation cost for computing the 5 additional samples is not justified, considering the given increase of precision.

Table 3.13
CRMSE values with respect to the sampling density for each surrogate model.

	Gaussian Process	Deep Gaussian Process	Deep Neural Network
6 samples	0.91	0.518	0.666
10 samples	0.57	0.357	0.206
15 samples	0.119	0.113	0.128

Table 3.13 shows the CRMSE values for the configurations considered in Fig. 3.9. The previous observations are confirmed by the numerical values of the CRMSE, namely that, for 15 samples, each surrogate model provides the same level of performance; and that, for 10 samples, the approximation is fair enough for any surrogate model.



(A) Gaussian Process (**6 points**) (B) Deep Gaussian Process (**6 points**) (C) Deep Neural Network (**6 points**)

(D) Gaussian Process (**10 points**) (E) Deep Gaussian Process (**10 points**) (F) Deep Neural Network (**10 points**)

(G) Gaussian Process (**15 points**) (H) Deep Gaussian Process (**15 points**) (I) Deep Neural Network (**15 points**)

FIGURE 3.9: Comparison of each surrogate model approximation with respect to the sampling density.

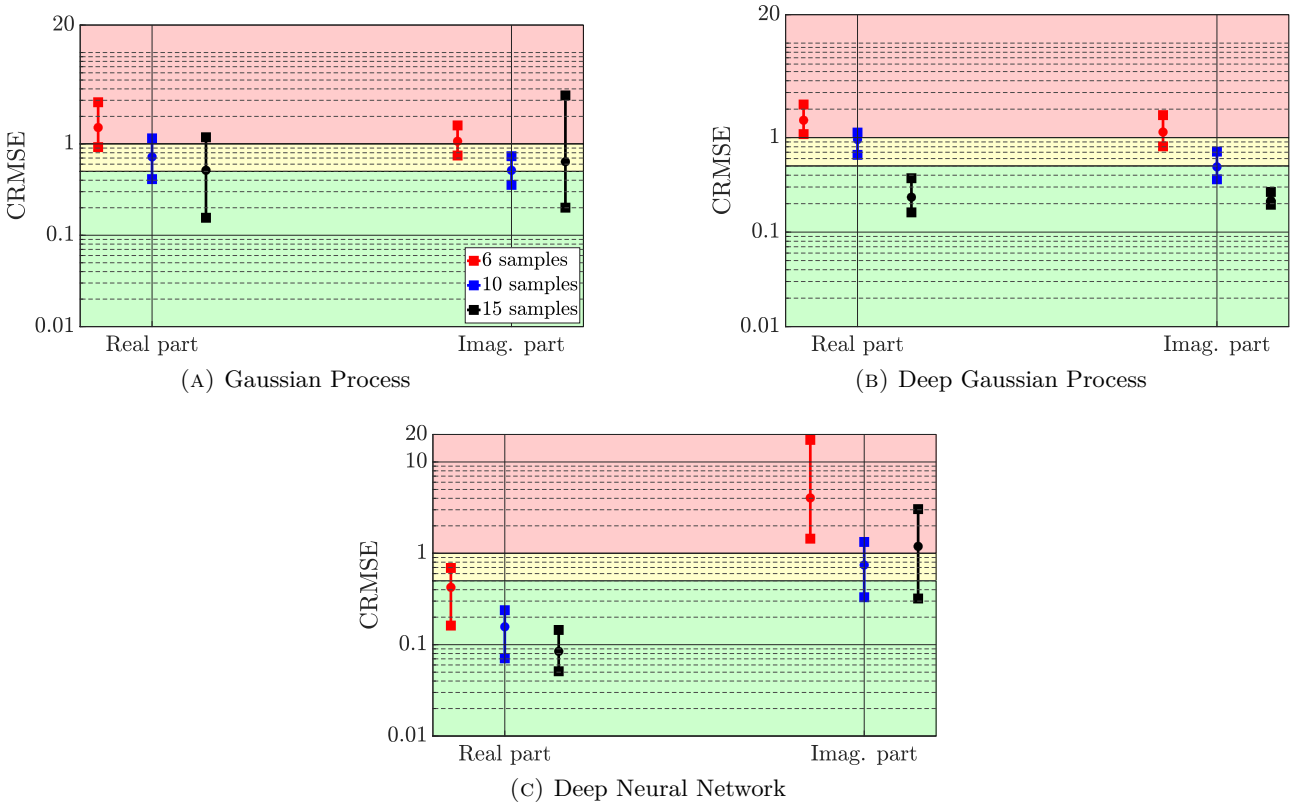
3.4.3 Worst hyperparameter setting for small training sets

3.4.3.1 Performance overview

Fig. 3.10 presents the CRMSE range of variation for the ten randomly distributed training sets in a similar way than for the best hyperparameter setting. For all the tested hyperparameter setting, the CRMSE value is compared and the configurations associated with the highest CRMSE value is denoted the worst configuration. This analysis is carried out on the averaged CRMSE over the training set.

Observations

First, for the DGP and the DNN, the real part approximations are clearly acceptable when the number of samples is greater than 10 points, even with a bad hyperparameter setting. Indeed, for the DNN, the worst performance of the mean approximation of the real part is always associated with a good CRMSE value. Conversely, for the DGP, it is at least linked to a mitigated CRMSE value.



(A) Gaussian Process

(B) Deep Gaussian Process

(C) Deep Neural Network

FIGURE 3.10: CRMSE range of variations for the worst hyperparameter setting of Gaussian Process (A), Deep Gaussian Process (B) and Deep Neural Network (C).

However, for the GP approximation, the results are more problematic. Indeed, for the first surrogate model, the CRMSE range of variation is wider for a sample number of 15 than for the smaller training set sizes. For instance, the range of variations are respectively about 1 CRMSE point and 3 CRMSE points, for the approximations of the real and imaginary parts. With regard to the DNN imaginary part approximations, the results are also problematic, whatever the training set size. The CRMSE is even greater than 10 for 6 training samples.

Table 3.14 shows the hyperparameter settings that led to the mean CRMSE, shown in Fig. 3.10. The procedure is similar to the one explained for Table 3.12.

Table 3.14
Worst hyperparameter setting of GP, DGP and DNN.

Surrogate model	Training set size	6 points		10 points		15 points	
	Approximated function	Real	Imag.	Real	Imag.	Real	Imag.
GP	Kernel	**RBF**	**RBF**	**RBF**	**RBF**	**RBF**	**RBF**
DGP	Kernel function	**M52**	**M52**	**M52**	**M52**	**M52**	**M52**
	Architecture	(3,10)	(4,10)	(3,10)	(4,10)	(4,10)	(4,10)
DNN	Activation function	TanH	ReLU	Logistique	ReLU	Logistique	ReLU
	Architecture	(4,200)	(4,300)	-	(4,200)	-	(4,200)

The worst GP and DGP settings are stable with regard to the kernel function. Whatever the considered approximated functions, the **RBF** (resp. the **Matern 5/2**) gives the worst results. For

the architecture of the DGP and the DNN, **deeper structures** cause the worst approximations, since the worst hyperparameter setting is associated with 4 hidden layers for both surrogate models. The DNN produces unstable approximation results, especially for the real prediction with 10 and 15 samples, where no particular hyperparameter setting has clearly been identified.

Discussions

The instability of the hyperparameter setting of the DNN is highlighted for the worst setting. Here, the ReLU activation function generates bad approximations, especially for the imaginary functions. This can be easily explained by the characteristics of the activation function. Indeed, the ReLU cannot be differentiated due to the use of the maximum operator. Consequently, it is more adapted to highly non-linear approximations. Here, the imaginary functions are the frequencies of the studied problem, which remain relatively smooth, despite the (large) variation of the friction coefficient.

Conversely, the DGP exhibits more stable results than the DNN. The performance for the worst hyperparameter setting is pretty close to the performance with the best setting. For instance, the approximation of the imaginary functions with 10 samples generates an error of 0.42 with the best setting, while an error of 0.5 is generated with the worst setting.

Finally, the observations about the roughness of the kernel functions are still valid here since the smoothest functions (the RBF and the Matern 52) generate the worst approximations, respectively, for the GP and the DGP.

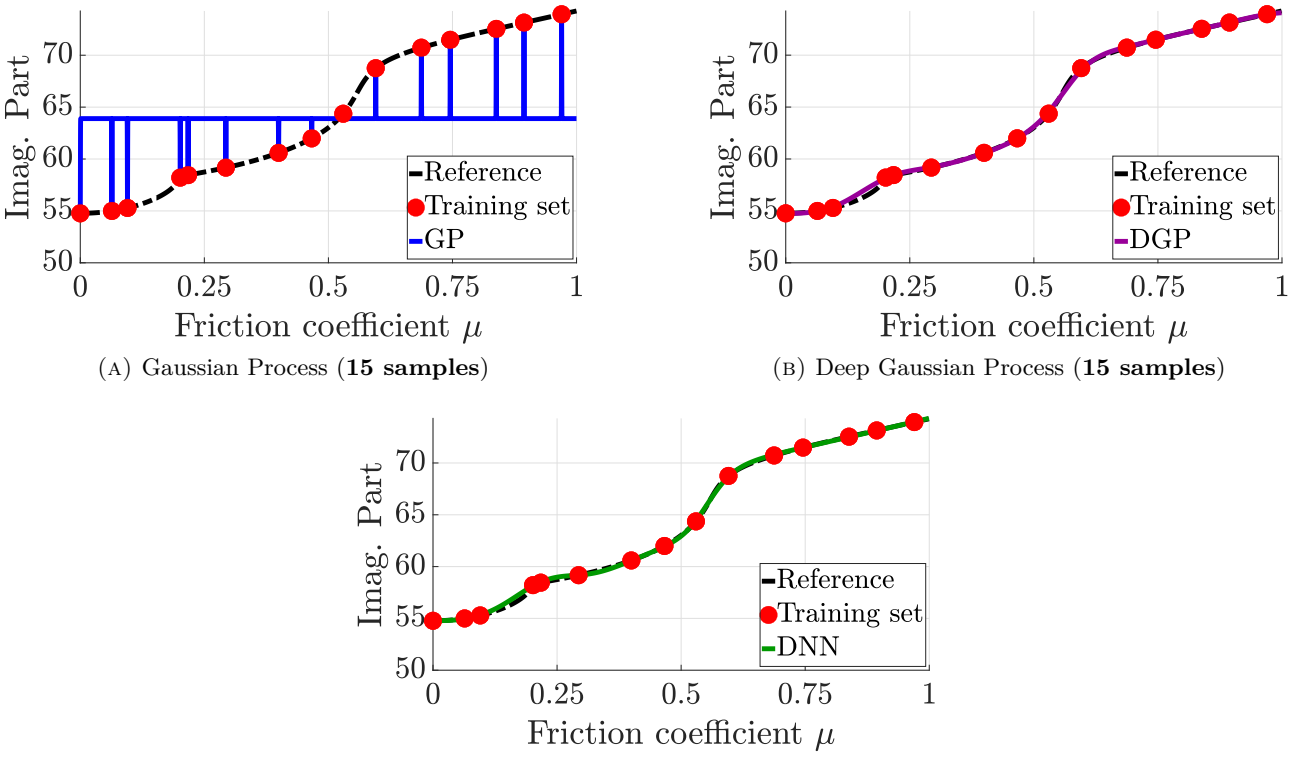
The next section focuses on the two scenarios where the GP and the DNN produce erroneous predictions. The purpose is to highlight the consequence of the issue of both surrogate models over their approximation capabilities and to compare with the DGP approximation.

3.4.3.2 Erroneous predictions for GP and DNN

Fig. 3.11 focuses on a limitation of GP surrogate model for specific training sets. The GP prediction Fig. 3.11(A) is constant on the majority of the design space, except at the vicinity of the samples of the training sets. Hence, this deceptive approximation clearly affects the predictivity of the surrogate model, yielding it unusable in this case. However, when a deep architecture is taken into account, this problem of deceptive approximation does not appear. For both other surrogate models, the prediction is almost perfect.

Similarly, Fig. 3.12 highlights a limitation of the DNN surrogate model that appears with 6 samples, but can appear whatever the considered training set. Whereas GP and DGP approximation are almost similar and quite efficient with regard to the training set information, the DNN does not give a good approximation at all. Indeed, between each sample, the prediction is highly fractured.

This phenomenon is called overfitting and has first been emphasised by Runge [145]. In a nutshell, the overfitting corresponds to the case when the surrogate model becomes very efficient to explain



(A) Gaussian Process (**15 samples**)

(B) Deep Gaussian Process (**15 samples**)

(C) Deep Neural Network (**15 samples**)

FIGURE 3.11: Comparison of each surrogate model prediction for a faulty GP setting.

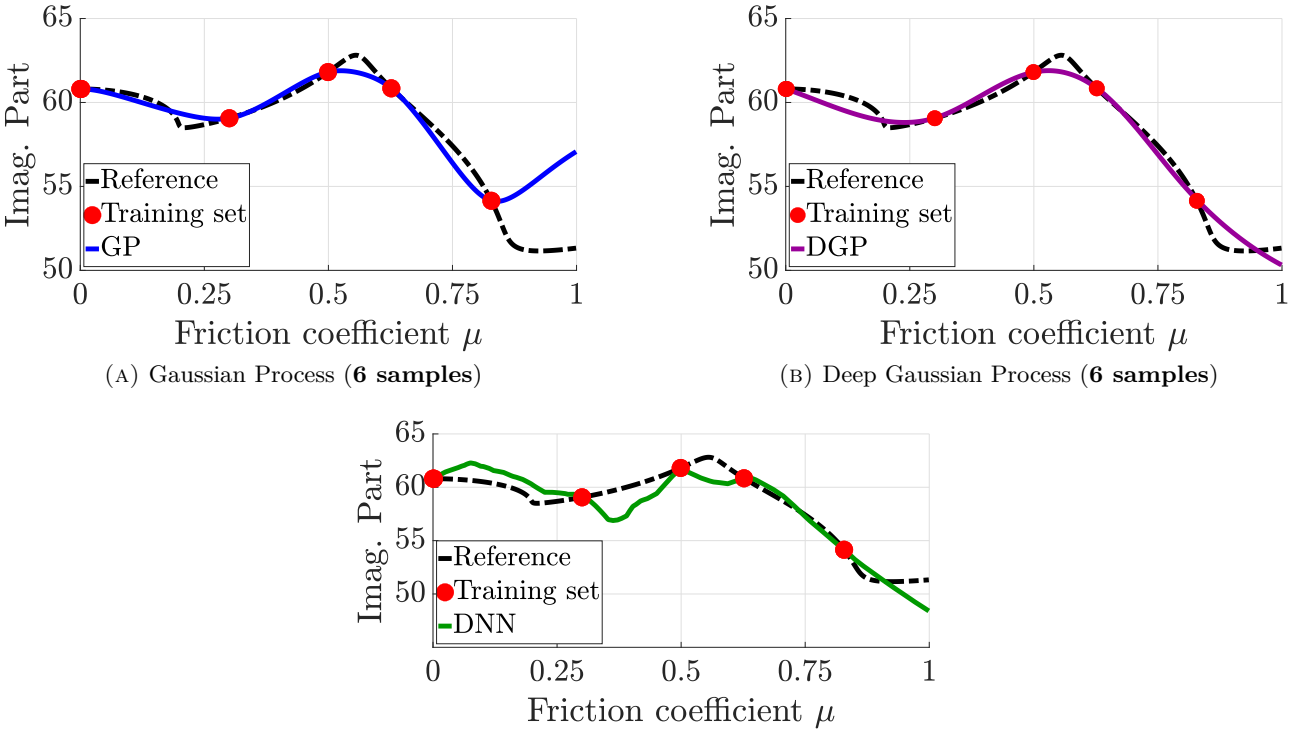
the training set, but cannot explain other set of samples (namely the test set). Consequently, the prediction oscillates quickly, and the surrogate model is unusable. This issue is known to be frequent with neural network [142, 161]. This is still an open research field, although many methods have been proposed to overcome this phenomenon [159, 160].

3.4.4 Conclusions

This section showed that, considering an optimal hyperparameter setting, the approximation provided by the surrogate models are interchangeable and any of them can be used. In addition, the optimal sample density is showed to be 10 samples per dimension since it allows a good representation of the design space, while limiting the computational cost.

For the hyperparameter setting, the GP and the DGP were showed to be pretty stable. Indeed, for both surrogate models, the Matern 3/2 corresponds to the best kernel function. For the DGP, a mid-deep structure gave the most efficient approximations. For the DNN, the results were more mitigated. Indeed, no clear conclusions were drawn from these results, and a good hyperparameter setting for a given training set can become a bad setting for another sample distribution. This enforces the user to evaluate a lot of configurations to find the most efficient one.

Consequently, the DNN is set aside due to its instability with respect to the hyperparameter setting, whatever the number of samples in the training set and the quantity to approximate. In



(A) Gaussian Process (**6 samples**)

(B) Deep Gaussian Process (**6 samples**)

(C) Deep Neural Network (**6 samples**)

FIGURE 3.12: Comparison of each surrogate model prediction for a faulty DNN setting.

addition, the DNN does not provide a direct measure of the modelling uncertainty (the variance of GP and DGP), which will be important for the use of Bayesian Optimisation in the following chapters. It is possible to obtain a substitute to the variance when using the DNN, but the implementation is more complex.

3.5 Applications of Gaussian Processes and Deep Gaussian Processes in a multiparametric analysis

3.5.1 Hyperparameter setting of the Deep Gaussian Process in a multiparameteric analysis

3.5.1.1 Preamble

This section aims at highlighting the evolution of the DGP performance with respect to the dimensionality of the problem. Many tests showed that, unlike the GP, the hyperparameter setting conclusions on the 1-dimensional problem do not extend to n-dimensional problems. Thus, a discussion on the optimal architecture of the DGP is done to emphasise the relation between the system parameters and the capacity of the DGP to efficiently approximate the whole frequency spectrum.

To do so, the analysis is decomposed into two steps: first, the approximation of the eigenvalues of the Double Hulten model is considered with five variable parameters. Then, three scenarios are studied with three, five and seven variable parameters.

3.5.1.2 Five-dimensional study

The following parameters of the Double Hulten are considered variable: m_1, m_2, k_a, c_a and μ, with a variation of $\pm$ 10 %, except for the friction coefficient, which varies between 0 and 1. The first four parameters are similar to the structural parameters of a real brake system. With regard to the training set $\mathcal{T}$, used to train the surrogate model, ten sets, composed of 50 samples (10 per parameter) are randomly generated using an LHS. A DGP is built for different hidden layers and neurons setting.

Observations

Table 3.15 and Table 3.16 shows the CRMSE values for the DGP prediction over an MC sampling of 10,000 reference samples. The values for the hidden layers and neurons setting are given in these tables.

Table 3.15
CRMSE for real part approximations.

Neurons	Hidden layers			
	1	2	3	4
1	6.29	6.13	6.23	6.29
3	4.35	4.33	4.32	4.32
5	0.468	0.469	0.47	0.47
7	0.467	0.469	0.478	0.478
10	0.467	0.468	0.491	0.489

Table 3.16
CRMSE for imaginary part approximations.

Neurons	Hidden layers			
	1	2	3	4
1	6.159	6.033	6.098	6.003
3	4.266	4.242	4.241	4.269
5	0.448	0.454	0.46	0.459
7	0.448	0.454	0.458	0.457
10	0.448	0.454	0.455	0.459

First, two levels of performance are highlighted in these tables. The DGP provides bad approximations when the number of neurons is lower or equal to 3. Conversely, when the number of neurons is at least equal to 5, the approximations are efficient for both real and imaginary parts of the eigenvalues.

Secondly, the performance is quite stable over the number of neurons and the number of hidden layers, with a CRMSE around 0.47 for the real part and around 0.45 for the imaginary part. Thus, one hidden layer is sufficient to provide efficient approximations.

From the analysis of the previous results, the number of neurons has to be at least set to 5, which corresponds to the number of parameters taken into account, for either the real or the imaginary part approximations. To validate this assertion, the following section focuses on the study of scenarios which involve several variable parameters.

3.5.1.3 N-dimensional study.

The three considered scenarios are the following ones:

- 3 input parameters: m_1, m_2 and μ ;
- 5 input parameters: m_1, m_2, k_a, c_a and μ ;
- 7 input parameters: k_{11}, k_{12}, k_{21}, k_{22}, k_a, c_a, and μ.

Similarly to the previous test, for the performance evaluation, 10 training sets are randomly generated with 10 samples per input parameter (30, 50 and 70 samples on overall for each scenario).

Table 3.17
CRMSE for real part approximations with 3D, 5D and 7D scenarios.

3D		5D		7D	
Neurons	CRMSE	Neurons	CRMSE	Neurons	CRMSE
1	6.355	1	6.29	1	6.219
2	4.349	3	4.35	3	5.319
3	0.491	5	0.468	5	4.722
4	0.491	7	0.467	7	0.541
5	0.491	10	0.467	10	0.541

Table 3.18
CRMSE for imaginary part approximations with 3D, 5D and 7D scenarios.

3D		5D		7D	
Neurons	CRMSE	Neurons	CRMSE	Neurons	CRMSE
1	6.079	1	6.159	1	5.876
2	3.915	3	4.266	3	4.969
3	0.471	5	0.448	5	4.101
4	0.471	7	0.448	7	0.537
5	0.471	10	0.448	10	0.538

Observations

The CRMSE values for each configuration and each scenario are given in Tables 3.17 and 3.18. The assertion of the previous section is validated by those results since, for the 3D scenario, the number of neurons has to be at least 3, and, for the 7D scenario, it should at least be 7.

Discussions

This observation has a strong implication: the formalism of the DGP used in this research is not capable of doing dimension reduction. This limitation of the considered DGP allows stating a rule of thumb about the setting of the DGP: the number of neurons has to be at least equal to the number of parameters.

Moreover, taking a number of neurons greater than the number of parameters also yield efficient results, but, due to the stability of the DGP, its performance is not improved. In the studied applications, one hidden layer provides good performance for the surrogate model approximations. It limits the computational cost induced by the inference of deeper and deeper models.

3.5.2 Comparison between GP and DGP performance

This section compares the prediction of the GP using the best hyperparameter setting, highlighted in Section 3.4.2 and 3.5.1. To do so, the same experimental procedure than the one of the previous section is used, meaning 10 randomly generated training sets with 10 points per dimension, for three scenarios: 3, 5 and 7 parameters.

Table 3.19
Comparison between the GP and DGP CRMSE for real and imaginary part approximations with 3D, 5D and 7D scenarios.

	GP			DGP		
	3D	5D	7D	3D	5D	7D
Real	0.53	0.479	0.56	0.491	0.468	0.541
Imag.	0.51	0.464	0.55	0.471	0.448	0.537

Observations

Table 3.19 shows the CRMSE value for each surrogate approximation. The DGP performs better than the GP, whatever the considered scenario or the approximated quantity (real or imaginary). Nonetheless, the improvements brought by the DGP are small, roughly 0.2 points of CRMSE value.

Discussions

This observation arises some concerns about the relevance of the DGP, used for the approximation of the eigenvalues of an FIV problem. Indeed, all the tests performed in Section 3.4 and 3.5 proved the efficiency of the DGP; however, the computational cost of training this surrogate model is high. For instance, a GP shows a training time of less than 1 second. The DGP training time is highly dependent on its architecture and the dimensionality of the problem. For these tests, in a one-dimensional problem, the training lasted 2 minutes to 30 minutes, on average; for a 5-dimensional problem, the training lasted 2 hours to 6 hours. Obviously, this training cost can be leveraged by using heavy parallelism, but, because of the precision improvement, using the DGP may not be relevant for FIV problems.

Two main reasons can be invoked to justify this weak improvement of precision. The first one is that, in the literature, the majority of works on DGP [131, 133, 149] use a great amount of samples (from 39 to 5081 samples per dimension in [132], for instance). These training set sizes are not reachable for our kind of application, especially with industrial models, and the DGP may not have sufficient data to be trained.

A second explanation is that the considered functions (the eigenvalues of the squeal problem) are not sufficiently non-linear to highlight a large difference between GP and DGP approximations. For instance, Damianou et al. [131] show the efficiency of the DGP with a Heaviside function, which exhibits highly abrupt variations. Similarly, Hebbal et al. [135] use the DGP to approximate the constraints of an optimisation problem. Typically, Fig. 3.2(A) and (B) clearly show that the evolution of the real or imaginary part eigenvalues is non-abrupt, even in the coupling areas.

3.6 Chapter outcomes

The purpose of this chapter was to address the **challenge of the computational burden of the CEA in a multiparametric analysis**. As the size of FE models become larger and larger, the evaluation time of one set of parameter is increasing. In addition, the integration of uncertainties in the simulations was proven to be challenging due to the additional evaluations of the solver. Finally, the optimisation of the studied structure usually invokes multiple calls to the solver, burdening once again the budget allocated for the study of a given physical phenomenon.

Hence, to tackle this issue, this chapter assesses the performance of three mainly used surrogate models: the GP, the DGP and the DNN. The study aims at illustrating the impact of surrogate models parameterisation for an FIV problem. A focus is made on suboptimal training sets. This suboptimality is a consequence of the use of common random training sets, which need a high sampling density to be fully efficient, non-achievable for traditional engineering applications.

The first experiment studied the effects of the most complex parameter of a friction-induced-vibration problem: the friction coefficient. Three different training set density were considered, namely 6, 10 and 15 samples, for 10 randomly distributed training sets. The results showed that, given that the hyperparameter setting is optimal, all the surrogates are interchangeable. In addition, from a computational cost aspect, the 10 samples per parameter density produces relevant results. The 15 samples density is better, but the small improvement in precision does not counterbalance the increase in the computational cost.

With regard to the hyperparameter setting of the surrogate models, the **GP and DGP gave the most efficient results when using the Matern 3/2 kernel function**. **The performance of the DNN was highly unstable, and no trend was drawn from the results**. Many architecture configurations must be evaluated to determine the most suitable one. Hence, the DNN was set aside for the second experiment.

The second experiment focused on the performance of GP and DGP with a higher number of uncertain parameters, namely three, five and seven parameters. First, the results showed that **the number of hidden layer and neurons can be controlled to lower the computational cost of the DGP**. Indeed, the number of neurons can be set to be equal to the number of uncertain parameters. As for the hidden layer number, it can be restrained to one. These rules of thumb

are important since the computational cost of training the DGP is important. For instance, for a 5-dimensional problem, the inference step lasted two to six hours.

Finally, a comparison between the GP and the DGP for these multidimensional scenarios was performed. The results showed that, even if the latter one outperforms the first one, **the differences between those two surrogates are small and the additional computational cost is not justified**. Two main reasons are discussed to explain this weak improvement. First, the studied functions may be not sufficiently nonlinear. Then, the number of samples is too small, and the DGP needs a higher sample density.

Considering these reasons, **the Deep learning surrogate models are set aside**. Following the observations about the interchangeability of the surrogate models when optimally set, **the GP represents the best compromise for the FIV problem**. Its training is indeed cheap and simple. Nonetheless, it is also shown that **some training sets may lead to incorrect predictions of this surrogate model**. This issue is addressed in the following chapter.

Chapter 4

Deceptive predictions using Gaussian Processes: A restoration of the predictivity of the surrogate model

Contents

4.1 Introduction

The previous chapter highlighted the performance of different surrogate models with regard to the approximation of the complex eigenvalues of the classical QEP. The GP exhibits interesting performance, while limiting the computational cost of inferring over the training set. Besides, from a design point of view, the GP is a powerful tool for the designer. Indeed, compared to other surrogate models, the modelling uncertainty, *i.e.* the variance, is estimated. The strength of this estimation is that this surrogate model indicates areas of the design space which may be interesting to explore to improve the representativity of the approximation. Hence, this information is of main interest in the pre-design phase to improve the knowledge of the mechanical behaviour of the product at study. Moreover, the use of the GP, and its first two statistical moments, is also relevant when optimising mechanical parts. Indeed, the Bayesian Optimisation (or BO) [162] takes advantage of these moments to identify potential interesting areas of the design space and increment the training set to converge towards the global optimum.

Nonetheless, the previous chapter showed that some predictions are completely deceptive. Indeed, the mean of the GP is constant for the majority of the design space for this case. Thus, this pathology of the GP is problematic. Considering a complete design process, it can greatly affect both pre-design and optimisation of mechanical parts, making this surrogate unusable.

This deceptiveness issue has been fewly tackled in the literature. Jones [163] was the first to coin out the term "deceptiveness" to characterise the samples used to construct the prediction of the GP, which produces erroneous predictions. Then, Forrester et al. [164] introduced the notion of

deceptive functions to characterise the same phenomenon. It is worth-noting that, for both authors, the deceptiveness (of the samples or the functions) illustrate a misprediction of the surrogate model and not a pathology of the surrogate parameter optimisation. More recently, Benassi et al. [165] followed a similar definition of the deceptiveness and suggested a Bayesian method to determine surrogate parameters in an optimisation context. On the opposite side, Li et al. [166] illustrate a similar pathology than the one emphasised in the previous chapter (without linking it to the deceptiveness of the GP). The authors attributed this pathology to a loss of correlation and suggested a penalisation method over the lengthscale vector to overcome it. Hebbal et al. [167] discarded the GP and highlight the performance of the DGP to overcome this issue. Nonetheless, the issue remains rarely tackled in the literature and most authors [168, 169] simply acknowledged or suffered the issue, without looking for tackling it.

Thus, two main reasons can be summoned to characterise the deceptiveness of a GP prediction. The first one corresponds to the sample location, which produces erroneous predictions, but the GP is not pathological. The prediction misrepresentation is mostly caused by a lack of exploration in a certain area of the design space. The second one is a problem arising from the classical way of training the GP. Indeed, a loss of correlation is observed between samples, inducing flat predictions all over the design space.

Since the behaviour of the function to approximate is not *a priori* known, there is no way to characterise the deceptiveness of a function (or samples). Thus, a restriction of the term "deceptiveness" to pathological predictions of GP, as highlighted by Li et al. [166] or Hebbal et al. [167], seems quite natural.

Table 4.1 summarises the authors acknowledging the deceptiveness of the GP predictions and whether they address it.

Table 4.1
Summary of the different approaches toward the deceptiveness of GP.

		Cause		Solution	
		Samples/Functions	Surrogate model	Yes	No
Jones	[163]	✓			✓
Forrester	[164]	✓			✓
Benassi	[165]	✓		✓	
Li	[166]		✓	✓	
Hebbal	[167]		✓	✓	
Lizotte	[168]	✓			✓
Chakraborty	[169]		✓		✓
Mohammadi	[170]		✓		✓

The penalisation method [166] introduces a penalty in the form of a sum of the lengthscale components, which is a macroscopic approach. The main flaw of this strategy is that the lengthscale values are all aggregated in one criterion, and the strategy cannot account for the effect of each component. Besides, the deceptiveness issue is generally caused by only one ill-conditioned lengthscale value, which

is hardly detected by the suggested method. As for the strategy proposed by Hebbal et al. [167], the use of DGP was shown in the previous chapter not to be compatible with the friction-induced vibration applications. Finally, the strategy of Benassi et al. [165] needs more data to describe the probability distribution of the parameters of the GP. This data is, in practice, hard to obtain.

The purpose of this chapter is, first, to highlight the consequences of the deceptiveness of the GP with respect to the surrogate model driving equations. Second, the traditional strategy is shown not to be suited for the approximation of training sets that causes the deceptiveness and a strategy is proposed to overcome the issue. This strategy aims at restoring the approximation capabilities of the surrogate model. To do so, a detection criterion is derived from the likelihood equation and an algorithm is then suggested.

To highlight the capabilities of the new suggested method, this chapter is organised as follows. Section 4.3 presents and identifies the deceptiveness issue with the Xiong's function, which allows stating a detection criterion and a new algorithm in Section 4.4. Finally, Section 4.5 assesses the performance of the new method for different application cases, namely the Xiong's function, a benchmark of non-linear functions and a friction-induced vibration problem.

4.2 Gaussian Process theory reminder

The GP was presented in Chapter 3 and is recalled here for practical reasons. A traditional GP hinges on a primary assumption that any set of random variables defined from this process is jointly Gaussian-distributed. Commonly, an additional assumption is added with regard to the likelihood, which is taken to be Gaussian. Using those hypotheses and basic algebra, the first two moments $\widehat{y}$ and $\widehat{s}$ of the output random variable $\widehat{\mathbf{Y}}$ are given by (Eq. 4.1) and (Eq. 4.2).

$$\widehat{\mu}(\mathbf{x}_*) = \mu(\mathbf{x}_*) + \mathbf{C}(\mathbf{x}_*, \mathbf{X}) \left[\mathbf{C}(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I}\right]^{-1} (\mathbf{y}(\mathbf{X}) - \mu(\mathbf{X})) \tag{4.1}$$

$$\widehat{s}(\mathbf{x}_*) = \mathbf{C}(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{C}(\mathbf{x}_*, \mathbf{X}) \left[\mathbf{C}(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I}\right]^{-1} \mathbf{C}(\mathbf{X}, \mathbf{x}_*) \tag{4.2}$$

where $\mathbf{x}_*$ is any vector defined in the design space, μ the mean of the GP, $\mathbf{C}$ its covariance matrix and σ_n^2 the noise of the data.

The covariance matrix $\mathbf{C}$ is given by (Eq. 4.3):

$$\mathbf{C}(\mathbf{x}_p, \mathbf{x}_q) = \sigma_k^2 \mathbf{K}(\mathbf{x}_p, \mathbf{x}_q) \tag{4.3}$$

where (p,q) $\in [\![1; n]\!]^2$, σ_k^2 is the signal variance and $\mathbf{K}$, the correlation matrix (also called kernel function or kernel matrix), $\mathbf{x}_p$ and $\mathbf{x}_q$ being any sample from $\mathbf{X}$.

In the previous chapter, the kernel functions were studied and the Matern 3/2 showed the best performance for the approximation of the system eigenvalues. Hence, this kernel function (Eq. 4.4) is considered in the following of this chapter to highlight the deceptiveness issue. Nonetheless, it is worth-noting that the conclusions drawn for this kernel function extend to other traditional class of functions.

$$\mathbf{K}(\mathbf{x}_p, \mathbf{x}_q) = \left(1 + \sqrt{3}\,|\mathbf{x}_p, \mathbf{x}_q|\right) \exp\left(-\sqrt{3}\,|\mathbf{x}_p, \mathbf{x}_q|\right) \tag{4.4}$$

where $|\mathbf{x}_p, \mathbf{x}_q| := \sqrt{\left(\sum_{i=1}^{D} \frac{(x_{(p,i)} - x_{(q,i)})^2}{\theta_i^2}\right)}$ is the weighted squared Euclidian distance and θ_i corresponds to the lengthscale of the ith dimension.

Table 4.2 summarises the different parameters of the GP, enumerated in the previous chapter and equations, namely the data noise σ_n^2, the signal variance σ_k^2 and the lengthscale vector $\mathbf{\Theta}$. In addition to the parameter name, complementary information is given about whether the parameter is optimised or selected, as well as its initial value.

Table 4.2
Parameters of the GP.

Parameter name	Computation	Initial Value
σ_n^2	Fixed	10^{-6}
σ_k^2	Fixed	1
$\mathbf{\Theta}$	Optimized by MLE	**1**

Since the data from the training set are not noisy (the solver is exact), there is no need to consider the data noise σ_n^2, which is fixed at 10^{-6}. Moreover, as the data are normalised in [0;1] range, the signal variance, σ_k^2, is not optimised for the proposed applications and is equal to 1. Nevertheless, it is important to mention that the deceptive predictions can arise whatever the value of these parameters.

Regarding the lengthscale, its value is determined by computing the MLE (Eq. 4.5). The quantity to maximise is rarely unimodal and, thus, complex to optimise.

$$\underset{\mathbf{\Theta}}{\text{maximise}}\,(\log(L)) \tag{4.5}$$

where L is the likelihood of the GP, namely its probability density function, and its log-form is given by (Eq. 4.6).

$$\log(L) = -\frac{n}{2}\log(2\pi) - \frac{1}{2}\log\left(\det\left(\mathbf{C} + \sigma_n^2\mathbf{I}\right)\right) - \frac{1}{2}\left(\mathbf{y}^T\left(\mathbf{C} + \sigma_n^2\mathbf{I}\right)^{-1}\mathbf{y}\right) \tag{4.6}$$

Overall, this estimator is quite efficient to determine the parameter values, which gives suitable predictions. Nevertheless, in some cases, more or less frequently, depending on the multi-modality of

the approximated function and its non-stationarity, finding the MLE can lead to parameter values inducing ill predictions. The following section assesses the consequences of this issue, from a graphical and mathematical points of view.

4.3 Identification of the deceptiveness issue

4.3.1 Graphical illustration

To highlight the studied problem, the prediction of Xiong's function [171] is investigated with a GP for two training sets. These are composed of 17 samples and present two different distributions along the design space, as shown in Fig. 4.1 and Fig. 4.2 with the blue dots. In these figures, the black curve corresponds to the target curve, whereas the red curve corresponds to the prediction of the surrogate. The prediction variance is represented by the green dashed line.

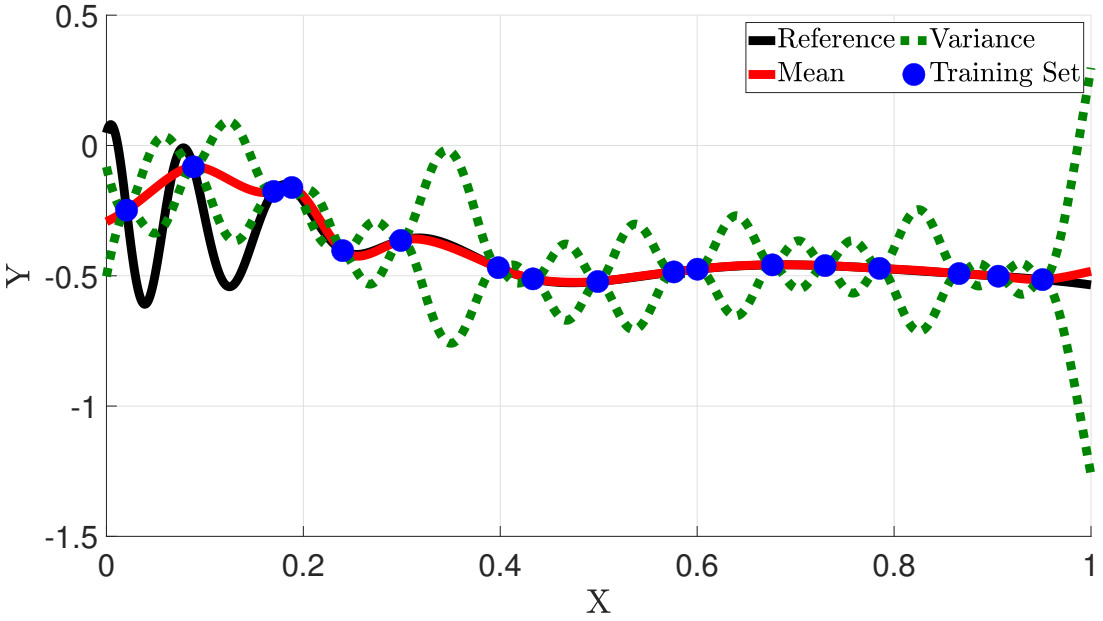


FIGURE 4.1: Prediction of the GP with the satisfying training set.

Fig. 4.1 shows a traditional prediction of the surrogate model. The poorly explored areas tend to have worse approximations than well-explored areas and present important levels of uncertainty (here, in the sense of surrogate modelling, *i.e.* variance). On one hand, between 0 and 0.2 abscissa, the approximation does not capture the oscillations well. On the other hand, between 0.2 and 1 abscissa, the target curve is well approximated.

A second prediction, given by Fig. 4.2, is more problematic. Indeed, the samples are evenly distributed, which implies that the design space is well explored. The mean and variance predictions of the GP are respectively flat and maximal in the majority of the design space, except in the vicinity of the instances of the training set. The samples have no correlation with each other, causing the

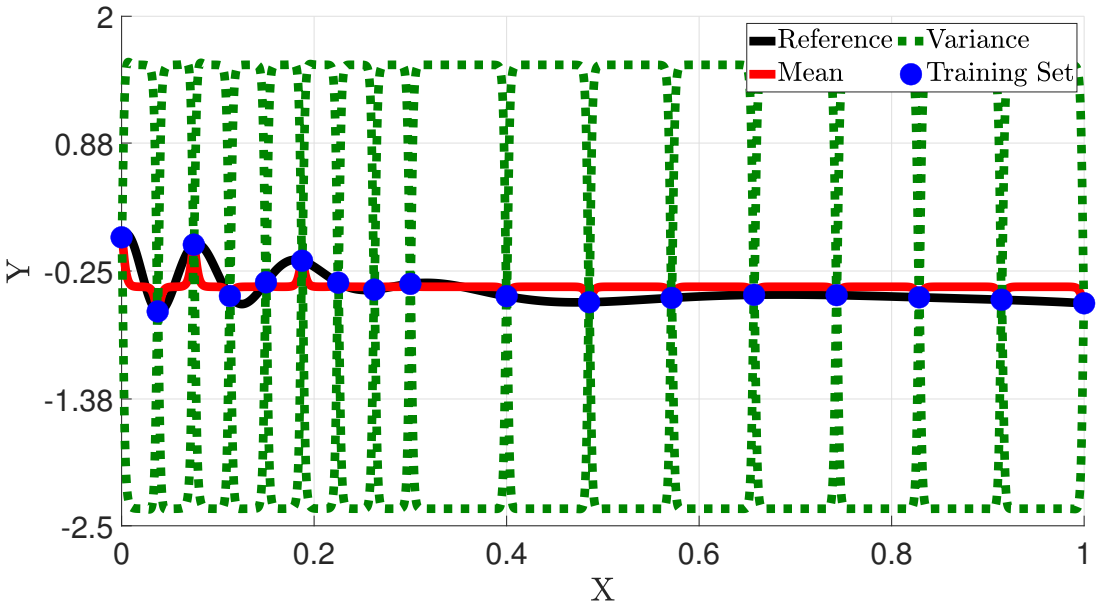


FIGURE 4.2: Prediction of the GP with the deceptive training set (b).

mean prediction to be equal to 0 in the standardised space, *i.e* -0.389 in the design space, and the variance prediction to be equal to 1.

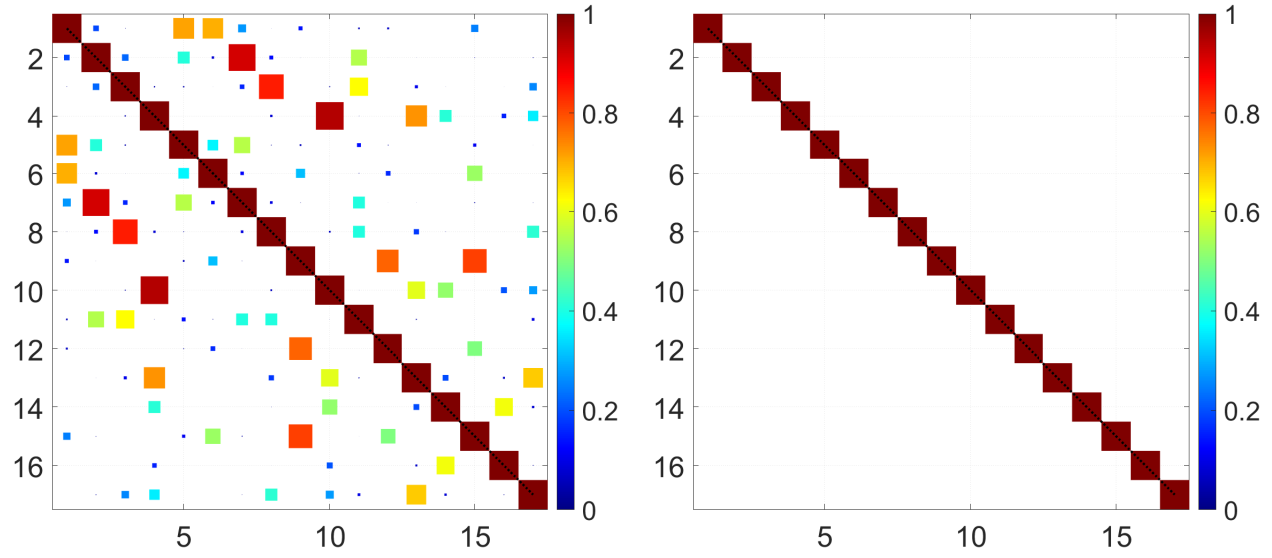
These observations validate the following statement: a GP is deceptive if $\widehat{y}$ is constant on all the design space and $\widehat{s}$ is equal to 1, except for the elements of $\mathbf{X}$ where $\widehat{y}$ equals to the elements of $\mathbf{Y}$ and $\widehat{s}$ equals to 0.

In the following sections of this chapter, the prediction presented in Fig. 4.1(A) is denoted as a satisfying prediction, whereas the one shown in Fig. 4.2(B) is denoted as deceptive. The related training sets are similarly denoted.

To analyse the difference between both predictions, a focus is done on the correlation matrix $\mathbf{K}$, which quantifies the correlation between each sample of the training set (Fig. 6.4). The higher the correlation between samples, the greater the value is. A value of 1 corresponds to a full dependence between instances (large and dark square). On the other hand, a value of 0 implies a full independence between instances (vanishing blue square).

Fig. 6.4(A) shows the correlation matrix for the satisfying prediction, which exhibits different levels of correlation. Some instances are strongly correlated, such as samples 2 and 7 (resp. X = 0.576 and X = 0.6). The kernel value of these locations is 0.917 and is quite logical since these samples are close to each other (distance of 0.024). Other samples present a mid-range correlation, such as samples 2 and 11 (resp. X = 0.576 and X = 0.449) with a kernel value of 0.547 and a distance of 0.077. Finally, a majority of samples exhibit a low-range correlation, such as samples 2 and 10 (resp. X = 0.576 and X = 0.169) with a kernel value of 0.029 and a distance of 0.407.

Fig. 6.4(B) shows the correlation matrix of the deceptive prediction, which, unlike the satisfying training set, shows no correlation at all between samples. All the extradiagonal terms are equal to 0.



(A) Satisfying training set

(B) Deceptive training set

FIGURE 4.3: Correlation matrices of the GP for both training sets.

However, the distance between each sample is quite small (0.038 on average). Comparing it to the previous covariance matrix values, the correlation would have been at least greater than 0.5.

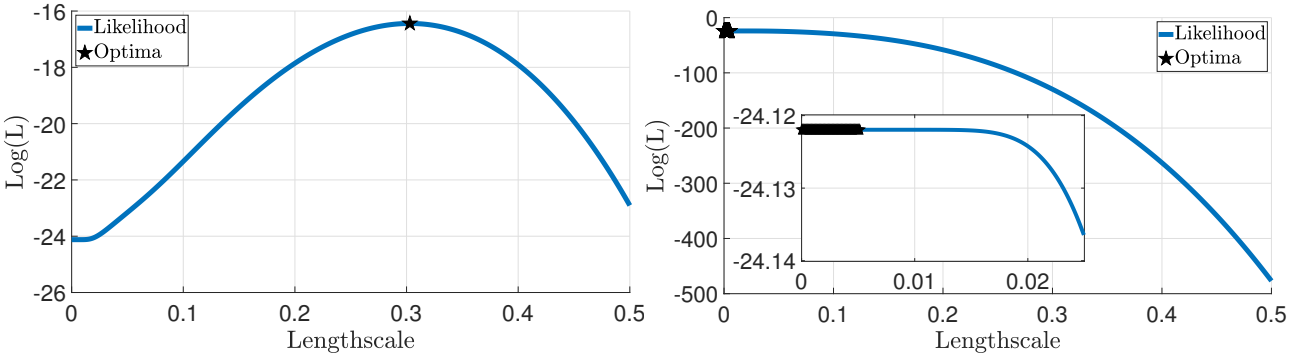
Table 4.3
Quartiles for both training sets.

Quartile	1^{st}	2^{nd}	3^{rd}	4^{th}
Satisfying training set	0.0004	0.0152	0.17	0.947
Deceptive training set	0	0	0	0

Table 4.3 shows a statistical analysis of the quartiles of the extra-diagonal terms of the correlation matrix for both training sets. For the satisfying training set, the 3^{rd} and 4^{th} quartiles are respectively equal to 0.17 and 0.947 (approximately 30 values), exhibiting low-range to high-range correlations, whereas, for the deceptive training set, all the quartiles equal to 0.

Considering that the kernel function depends not only on the distance between samples but also on the lengthscale, the behaviour of the likelihood, which allows setting the lengthscale, is assessed in Fig. 4.4. The optimum (or optima) is indicated with a black pentagram. These curves are typical of satisfying and deceptive training sets.

For the satisfying predictions, the global optimum is clearly unique. On the other hand, when the predictions are deceptive, the global optimum is not unique and is associated with a lengthscale close to 0. The non-uniqueness is a consequence of the constant behaviour of the likelihood function when the lengthscale is small. Here, when using an optimiser to find the MLE, the optimum lengthscale is about 0.01 (the first value which induces no improvement of the objective function), whereas, for the satisfying training set, this lengthscale is about 0.301.



(A) Satisfying training set (B) Deceptive training set

FIGURE 4.4: Log likelihood of the GP for both training sets.

Thus, a satisfying prediction is characterised here by an optimum, which is not contained in the plateau located close to 0. Conversely, a deceptive prediction is defined by an optimum located in this plateau.

4.3.2 Investigations about the deceptive configurations

In the previous section, different observations showed that the deceptiveness of the prediction is the result of a too small lengthscale, which is itself induced by a peculiar behaviour of the likelihood function when setting the GP parameters. Thus, a mathematical analysis is carried out to emphasise the implications of this observation on the driving equations of the GP, expressed in Section 4.2. The equations are derived in their multidimensional forms.

Let i $\in [\![1; D]\!]$, (p,q) $\in [\![1; n]\!]^2$ and let me consider the right limit of the covariance matrix $\mathbf{C}$ (Eq. 4.7), when θ_i goes to 0:

$$\begin{aligned}\lim_{\theta_i \to 0^+} \mathbf{C}(\mathbf{x}_p, \mathbf{x}_q) &= \sigma_k^2 \lim_{\theta_i \to 0^+} \mathbf{K}(\mathbf{x}_p, \mathbf{x}_q) \\ &= \sigma_k^2 \lim_{\theta_i \to 0^+} \left[\left(1 + \sqrt{3}|\mathbf{x}_p, \mathbf{x}_q| \right) \exp\left(-\sqrt{3}|\mathbf{x}_p, \mathbf{x}_q| \right) \right] \end{aligned} \tag{4.7}$$

where

$$\begin{aligned}|\mathbf{x}_p, \mathbf{x}_q|^2 &= \sum_{j=1}^{D} \frac{(x_{(p,j)} - x_{(q,j)})^2}{\theta_j^2} \\ &= \frac{(x_{(p,i)} - x_{(q,i)})^2}{\theta_i^2} + \sum_{\substack{j=1 \\ j \neq i}}^{D} \frac{(x_{(p,j)} - x_{(q,j)})^2}{\theta_j^2} \end{aligned} \tag{4.8}$$

First case: $\mathbf{x}_p = \mathbf{x}_q$

$$\text{then} \quad |\mathbf{x}_p, \mathbf{x}_q| = 0 \quad \text{and} \quad \lim_{\theta_i \to 0^+} \mathbf{C}(\mathbf{x}_p, \mathbf{x}_q) = \sigma_k^2 \tag{4.9}$$

Second case: $\mathbf{x}_p \neq \mathbf{x}_q$

(i) $x_{(p,i)} = x_{(q,i)}$ then

$$\lim_{\theta_i \to 0^+} |\mathbf{x}_p, \mathbf{x}_q| = \sqrt{\sum_{\substack{j=1 \\ j \neq i}}^{D} \frac{(x_{(p,j)} - x_{(q,j)})^2}{\theta_j^2}} := l \neq 0$$

and then

$$\mathbf{C}(\mathbf{x}_p, \mathbf{x}_q) = \sigma_k^2 \ \left(1 + \sqrt{3}l\right) \exp\left(-\sqrt{3}l\right) \tag{4.10}$$

(ii) $x_{(p,i)} \neq x_{(q,i)}$ then as

$$\lim_{\mathbf{X} \to +\infty} \mathbf{X} \exp\left(-\mathbf{X}\right) = 0$$

the following limit is deduced

$$\lim_{\theta_i \to 0^+} \mathbf{C}(\mathbf{x}_p, \mathbf{x}_q) = 0 \tag{4.11}$$

Regarding (Eq. 4.10), it cannot be defined as deceptive due to the observations made in Fig. 6.4(B). In a more practical way, this case does not arise frequently, even at all in numerical applications. Indeed, since most experimentations are based on an LHS sampling plan, the samples from $\mathcal{T}$ cannot have identical values for a dimension i. In addition to this, due to the numerical precision of computers, two values are not strictly identical, but rather equal within a constant ϵ, which corresponds to the computer precision. This case can be dropped without loss of generality.

Thanks to (Eq. 4.9), (Eq. 4.11) and the above remark, the limit of C is given by (Eq. 4.12).

$$\lim_{\theta_i \to 0^+} \mathbf{C}(\mathbf{X}, \mathbf{X}) = \sigma_k^2 \mathbf{I} \tag{4.12}$$

With regard to the prediction equations, the mean $\widehat{y}$ and the variance $\widehat{s}$ of the output random variable are also impacted and become (Eq. 4.13) and (Eq. 4.14), when $\mathbf{x}_*$ differs from the samples of the training set.

$$\lim_{\theta_i \to 0^+} \widehat{y}(\mathbf{x}_*) = \mu(\mathbf{x}_*) \tag{4.13}$$

$$\lim_{\theta_i \to 0^+} \widehat{s}(\mathbf{x}_*) = \sigma_k^2 \tag{4.14}$$

If $\exists \ell \in [\![1; n]\!]$ such that $\mathbf{x}_* = \mathbf{x}_l \in \mathbf{X}$,

$$\lim_{\theta_i \to 0^+} \widehat{y}(\mathbf{x}_l) = \mu(\mathbf{x}_l) + \sigma_k^2 \, \frac{y_l - \mu(\mathbf{x}_l)}{\sigma_k^2 + \sigma_n^2} \tag{4.15}$$

$$\lim_{\theta_i \to 0^+} \widehat{s}(\mathbf{x}_l) = \sigma_k^2 - \frac{\sigma_k^4}{\sigma_k^2 + \sigma_n^2} \tag{4.16}$$

Thus, the following property can be stated without loss of generality: if θ_i goes to positive zero and $\mathbf{x}_{(p,i)} \neq \mathbf{x}_{(q,i)}$, then the GP is deceptive, and reciprocally.

All the latter observations emphasised the consequences of a deceptive GP. Thus, the detection of this issue is of great importance since, contrary to the function shown in Section 4.3 which is one dimensional, most of the functions approximated with GP are not directly displayable due to the high dimensionality of the studied systems. One has to use indirect techniques, such as projection onto a space with fewer dimensions. Consequently, the deceptiveness issue cannot be efficiently assessed.

The following section suggests a method for the detection of this issue and a way to correct it to approximate the underlying functions with GP efficiently.

4.4 Deceptiveness criterion and new strategy

The previous section showed that one of the properties of deceptiveness is that one element of the lengthscale vector goes to positive zero. Since the behaviour of $\log(L)$ is not known, it is impossible to determine *a priori* if the location of the samples in $\mathcal{T}$ will generate a near-zero lengthscale component, and, thus, lead to a deceptive prediction. Consequently, the detection of whether the prediction is deceptive has to be done after the conventional optimisation of the GP.

As emphasised in Fig. 4.4, when the lengthscale is small enough, the $\log(L)$ exhibits a plateau, which holds the global optimum for which the prediction is deceptive. Thus, the main proposal of this chapter is to use this particular behaviour as an automatic way of detection. It is formalised as follows (Eq. 4.17), and a detection criterion can be stated from it, denoted Deceptive Upper Bound (DUB):

$$\lim_{\theta_i \to 0^+} \log(L) = -\frac{n}{2}\log(2\pi) - \frac{n}{2}\log\left(\sigma_k^2 + \sigma_n^2\right) - \frac{1}{2(\sigma_k^2 + \sigma_n^2)} \sum_{j=1}^{n} y_j^2 \tag{4.17}$$

Consequently, a GP of which the MLE is equal to the DUB is deceptive by definition.

One of the main advantages of this criterion is that it is independent of the dimensionality of the problem, and can therefore be applied to any case. Moreover, its complexity is quite low since it is of $\mathcal{O}(n)$, with n, the number of samples of the training set. The DUB can be viewed as a property of the training set $\mathcal{T}$ in its general form, since it depends only on the set $\mathbf{Y}$.

Finally, the DUB can again be simplified when the samples included in $\mathcal{T}$ are standardised. The DUB takes the form of (Eq. 4.18) with a constant complexity.

$$\text{DUB} = -\frac{n}{2}\log(2\pi) - \frac{n}{2}\log\left(\sigma_k^2 + \sigma_n^2\right) - \frac{n}{2(\sigma_k^2 + \sigma_n^2)} \tag{4.18}$$

The algorithm for dealing with a deceptive GP is depicted in Fig. 4.5.

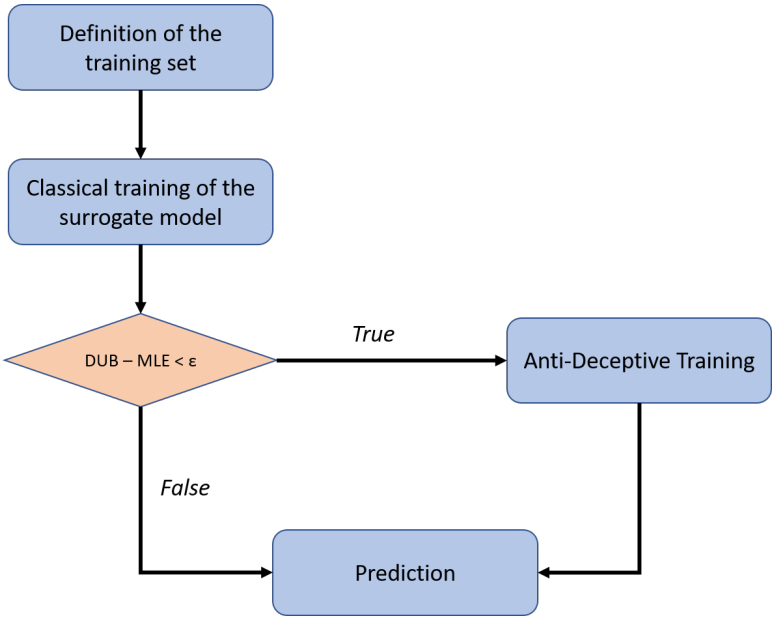


FIGURE 4.5: Workflow for the detection (and correction) of deceptive GP.

At first, the algorithm computes the GP considering (Eq. 4.6), where the log form of the likelihood is optimised. Then, a comparison between the values of the DUB and the MLE is performed. If both values are different, then the prediction can be carried out and the surrogate can be labelled as satisfying. On the other hand, if both values are equal within a precision value, the surrogate is considered deceptive.

In that case, a new training procedure, called Anti-Deceptive training (or ADT) is performed. In this new problem, a constraint driven by the DUB is introduced, as expressed by (Eq. 4.19). To introduce a distinction between both optimisation problems, the optimum found with this method is denoted as Constraint Maximum Likelihood Estimator, or CMLE.

$$\begin{aligned} &\underset{\theta_i}{\text{maximise}}\ \log(L) \\ &\text{subject to}\ |\log(L) - (1+k)\,\text{DUB}| \leq 0 \end{aligned} \tag{4.19}$$

where k is a hyperparameter, which controls the difference between the CMLE and the DUB.

This parameter can be directly fixed at a value within the range [0-1] or optimised by studying the convergence evolution of the mean and variance predictions for two successive values of k. These two strategies are assessed in Section 4.5.1.

In the next section, several numerical applications are proposed to highlight the performance of the suggested method. First, the behaviour of the method is emphasised with regard to the hyperparameter k on Xiong's function, which allowed describing the issue. Next, the method performance is evaluated on multidimensional mathematical functions. Finally, an engineering problem is considered, namely an FIV problem.

4.5 Applications of the suggested strategy

4.5.1 One dimensional problem

In this section, the suggested DUB detection criterion and the ADT procedure are investigated for the prediction of Xiong's function by considering the deceptive training set shown in Section 4.3.

Table 4.4
Quantities of interest for several hyperparameter values.

Hyperparameter value (k)	CMLE	θ	Quartiles			
			1^{st}	2^{nd}	3^{rd}	4^{th}
0.05	-24.12	0.01	0	0	0	0
0.1	-25.37	0.062	0	0	0	0.153
0.5	-36.79	0.139	0	0	0.018	0.562
1	-47.39	0.174	0	0.0003	0.049	0.667

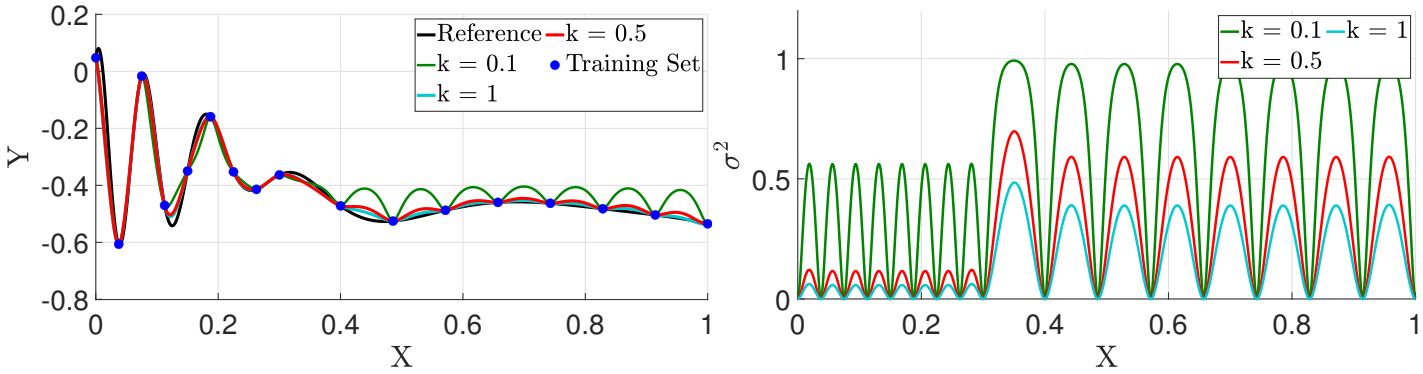
Observations

Table 4.4 gives the CMLE and the lengthscale for different values of the hyperparameter k. In addition, a statistical analysis on the extradiagonal values of the covariance matrix $\mathbf{K}$ is done with the four quartiles of these values. In the present case, the DUB is equal to -24.12.

First, a value of 0.05 for the hyperparameter k has no impact on the CMLE. Indeed, the CMLE is still equal to the DUB and the different quartiles equal to 0. When the hyperparameter value is equal to 0.1, the CMLE is greater than the DUB in absolute value and the lengthscale begins to move away from the boundary value at 0.01. Moreover, a correlation exists in $\mathbf{K}$ since the fourth quartile is equal to 0.153.

Interesting results arise for the hyperparameter k of respectively 0.5 and 1. Indeed, the CMLE is largely greater than the DUB in absolute value here, and the lengthscale is also greater than 0.01. Moreover, the third quartile is equal to 0.018 for k=0.5 and to 0.049 for k=1. In a similar way, some samples show strong correlations, since the fourth quartile is respectively about 0.562 and 0.667. These last observations are very close to the satisfying correlation values found in Section 4.3.

Fig. 4.6 compares the mean and variance predictions of the GP, while Fig. 4.7 shows the correlation matrices associated to the hyperparameter k=0.1, k=0.5 and k=1.



(A) Mean (B) Variance

FIGURE 4.6: Output prediction of the GP.

Considering the ADT, the mean prediction of the two GP matches the reference curve well (Fig. 4.6(A)) when the hyperparameter is set at a value greater than 0.1 and the associated output variance (Fig. 4.6(B)) has a more conventional behaviour. With a hyperparameter value of 0.1, the mean prediction has still a deceptive behaviour for an input value greater than 0.4 where the variance is almost equal to 1 between the samples of the training set. When using a hyperparameter value of 1, the variance is much smaller, which can be problematic in applications where this quantity is of main interest (BO [162], for example). Finally, for a hyperparameter value of 0.5, the mean prediction is quite similar to the mean prediction associated with $k = 1$, but the variance is greater.

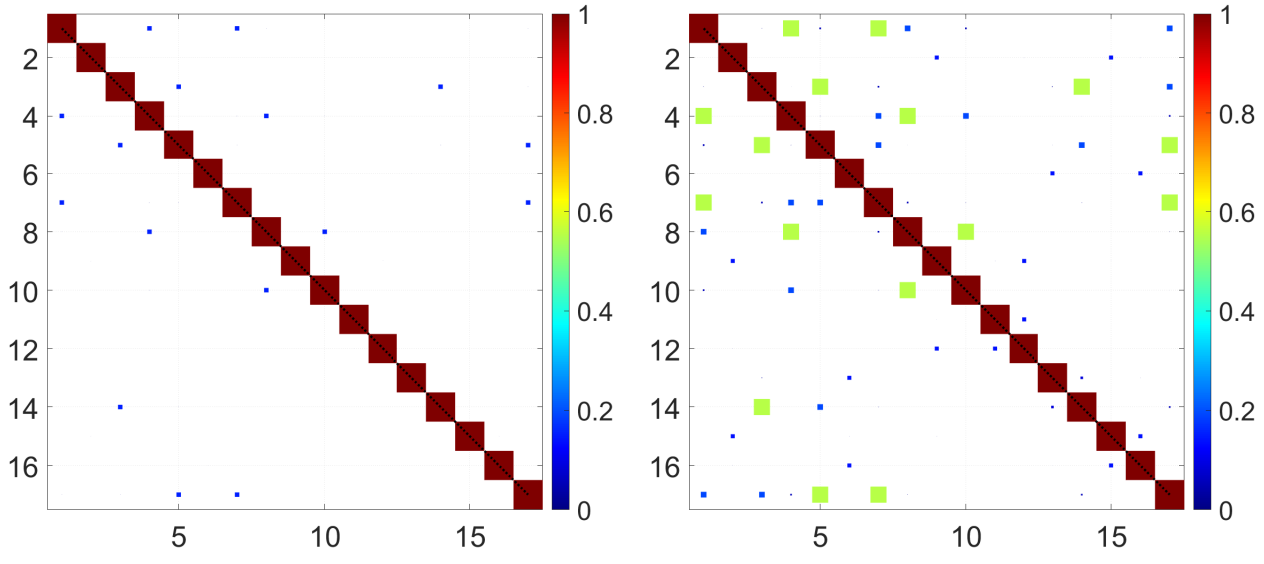
In addition, the correlation matrices, shown in Figs. 4.7 (A) to (C), confirm the observed trend, by presenting a non-null correlation between the different samples. The correlation matrix associated with a hyperparameter value of 0.1 clearly shows the weakest correlation between the samples, whereas the one associated with $k = 1$ exhibits the highest correlation.

In complement to the previous analysis, Fig. 4.8 presents the convergence evolution of the mean and variance predictions as a function of the hyperparameter k. The residue criterion (Eq. 4.20) is calculated as follows:

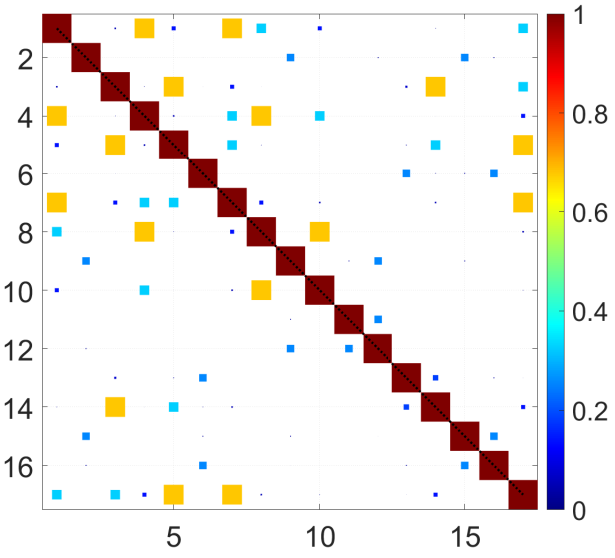
$$\text{Residue} = \sqrt{\frac{\sum_{j=1}^{n_{val}} |\hat{y}_j^{i+1} - \hat{y}_j^i|}{n_{val} \times \overline{\hat{y}^i}}} \tag{4.20}$$

where n_{val} represents the number of values used for the prediction. The indexes i and i + 1 traduce the iterative calculation for two successive values of the hyperparameter k, whereas $\hat{y}_j^i$ and $\hat{y}_j^{i+1}$ respectively describe the j^{th} predicted responses at each iteration. Finally, $\overline{\hat{y}^i}$ denotes the mean of predictions at the i^{th} iteration.

Considering a convergence threshold equal to 10^{-2}, which traduces a good stabilisation of the mean and variance predictions, a hyperparameter k of 0.6 is a good choice. This procedure allows supplying a more precise value of k but is slightly more time-consuming due to the different calculations of the CMLE.



(A) k=0.1 (B) k=0.5



(C) k=1

FIGURE 4.7: Correlation matrices of the GP prediction.

In conclusion, the observations carried out in this section have increased our confidence in the pertinence of the suggested strategy, since the correlation has been restored between samples. Indeed, the deceptiveness of the surrogate has been alleviated and the impact on the prediction and the correlation matrix have been well established.

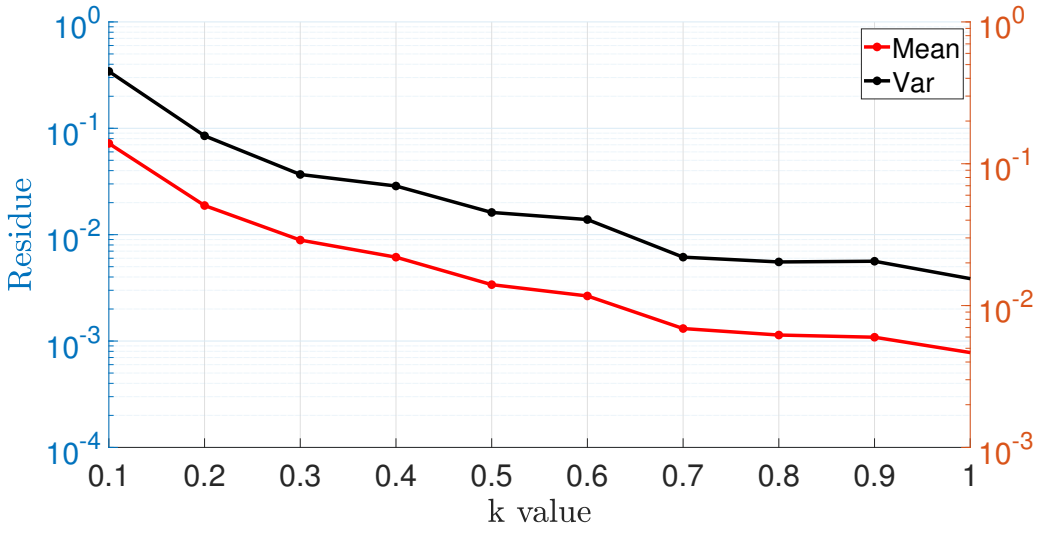


FIGURE 4.8: Convergence evolution of the predictions as a function of different values of the hyper-parameter k.

4.5.2 Benchmark of functions

The purpose of this section is to highlight the performance of the suggested strategy on a benchmark of mathematical functions described in Table 4.5 for the detection and correction of the deceptiveness issue. These functions can be found in the literature [171–173] except for the functions n°1 to 3, which are described in Appendix A and yield similar properties than those of FIV problems.

The following table shows the different properties of the functions, namely their dimension, their degree of non-stationarity and non-linearity and if they are multi-modal or not. In addition to this, the number of points considered for this benchmark is added. The majority of functions are non-stationary and non-linear, but some mid-range or low-range non-stationary or non-linear functions were added to the benchmark to emphasise that the problem is not restricted to the aforementioned functions.

In this benchmark, 10 satisfying and 10 deceptive training sets are considered to ensure the repeatability of the performance of the suggested strategy.

Table 4.5
Properties of the benchmark functions.

Function	Dimension	Non-stationarity	Non-linearity	Multi-modal	Number of points
Xiong	1D	High	High	Yes	10
Function n°1	2D	High	High	Yes	20
Heaviside	2D	Mid	High	No	20
Function n°2	2D	High	Mid	Yes	20
Six Hump	2D	Low	Low	No	10
Alpine n°2	5D	-	-	-	50
Schwefel	5D	-	-	-	50

Observations

At first, the efficiency of the detection criterion is evaluated. Then, the performance is investigated by comparing the quartiles of the satisfying training sets, the deceptive training sets with computation of the MLE and the deceptive training sets with computation of the CMLE.

Table 4.6
Detection criterion applied on satisfying training sets.

Function	DUB	MLE	Percentage of detection
Xiong	-14.19	-11.77	0%
Function n°1	-28.38	-26.51	0%
Heaviside	-28.38	-19.43	0%
Function n°2	-28.38	-23.81	0%
Six Hump	-14.19	-11.9	0%
Alpine n°2	-70.95	-67.64	0%
Schwefel	-70.95	-65.98	0%

Table 4.6 shows the results concerning the detection of satisfying training sets. The MLE and the DUB values are given in this table. In addition, the percentage of detection are added, which corresponds to the number of training sets detected as deceptive. For all functions, no deceptive training sets have been detected, which confirms the initial assumption, since the MLE is greater than the DUB.

Table 4.7
Detection criterion applied on deceptive training sets.

Function	DUB	MLE	k*	CMLE	Percentage of detection
Xiong	-14.19	-14.19	0.5	-21.22	100%
Function n°1	-28.38	-28.38	0.6	-45.32	100%
Heaviside	-28.38	-28.38	0.8	-51.34	100%
Function n°2	-28.38	-28.38	0.6	-46.74	100%
Six Hump	-14.19	-14.19	0.4	-19.55	100%
Alpine n°2	-70.95	-70.95	0.4	-99.31	100%
Schwefel	-70.95	-70.95	0.4	-100.66	100%

Table 4.7 is similar to the previous one, except that it addresses the deceptive training sets. In addition, the optimal value of k (convergence threshold equal to 10^{-2}) and the CMLE are added. Since the MLE equals the DUB, 100% of the sets are deceptive, which is graphically validated. Moreover, when the Anti-Deceptive algorithm is used to correct the deceptiveness of the surrogate, the CMLE is indeed greater than the DUB in absolute value, emphasising the efficiency of the algorithm.

To highlight the impact of the algorithm on the correlation matrix, the quartiles of this matrix are studied for each case in Table 4.8.

Table 4.8
Quartiles for the three considered cases and for all functions of the benchmark.

Quartile	Satisfying training set				Deceptive training set (MLE)				Deceptive training set (CMLE)			
	1^{st}	2^{nd}	3^{rd}	4^{th}	1^{st}	2^{nd}	3^{rd}	4^{th}	1^{st}	2^{nd}	3^{rd}	4^{th}
Xiong	0.023.	0.094	0.269	0.733	0	0	0	0.013	0.005	0.026	0.115	0.704
Function n°1	0.0004	0.004	0.044	0.77	0	0	0	$3.9e^{-4}$	0.012	0.042	0.149	0.884
Heaviside	0.07	0.158	0.339	0.923	0	0	0	0.003	0.009	0.036	0.137	0.895
Function n°2	0.018	0.065	0.171	0.859	0	0	0	0.006	0.003	0.016	0.064	0.756
Six Hump	0.018	0.065	0.171	0.859	0	0	0	$2.3e^{-6}$	0.163	0.29	0.461	0.856
Alpine n°2	0	0.002	0.004	0.781	0	0	0	0.002	0.053	0.092	0.163	0.763
Schewefel	0	0.001	0.014	0.801	0	0	0	$2.6e^{-6}$	0.12	0.186	0.289	0.861

At first, for the satisfying training sets, the first two quartiles are small (at best 0.06), except for the Heaviside function where 50% of the extradiagonal values of the correlation matrix are lower than 0.158. Finally, for all functions of the benchmark, the 4^{th} quartile is quite high and at least greater than 0.7.

Considering the deceptive training sets, where the GP is optimised with the traditional MLE value, for the 1^{st}, 2^{nd} and 3^{rd} quartiles, all values equal to 0. It means that there is no low-range or mid-range correlation between the samples of the training set. Finally, the 4^{th} quartile is also small with regard to the satisfying training sets, for any benchmark function.

When the GP is optimised with the suggested CMLE value, the results are more interesting. Indeed, for all functions of the benchmark, 50% of the extradiagonal values of the correlation matrix are around 0.02, except for the Six Hump with a 2^{nd} quartile around 0.29. With regard to the 4^{th} quartile, the values are quite high and at least greater than 0.7. Finally, the correlation values are quite similar to those from the satisfying training sets, meaning that the suggested method succeeds in restoring a mid-range and a high-range correlation between the different training set samples.

Conclusions

The numerical experimentations on a benchmark of mathematical functions have confirmed that the suggested method is efficient in restoring the correlation between samples. In addition, the detection criterion works well for both satisfying and deceptive training sets.

4.5.3 Application to a Friction-Induced-Vibration problem

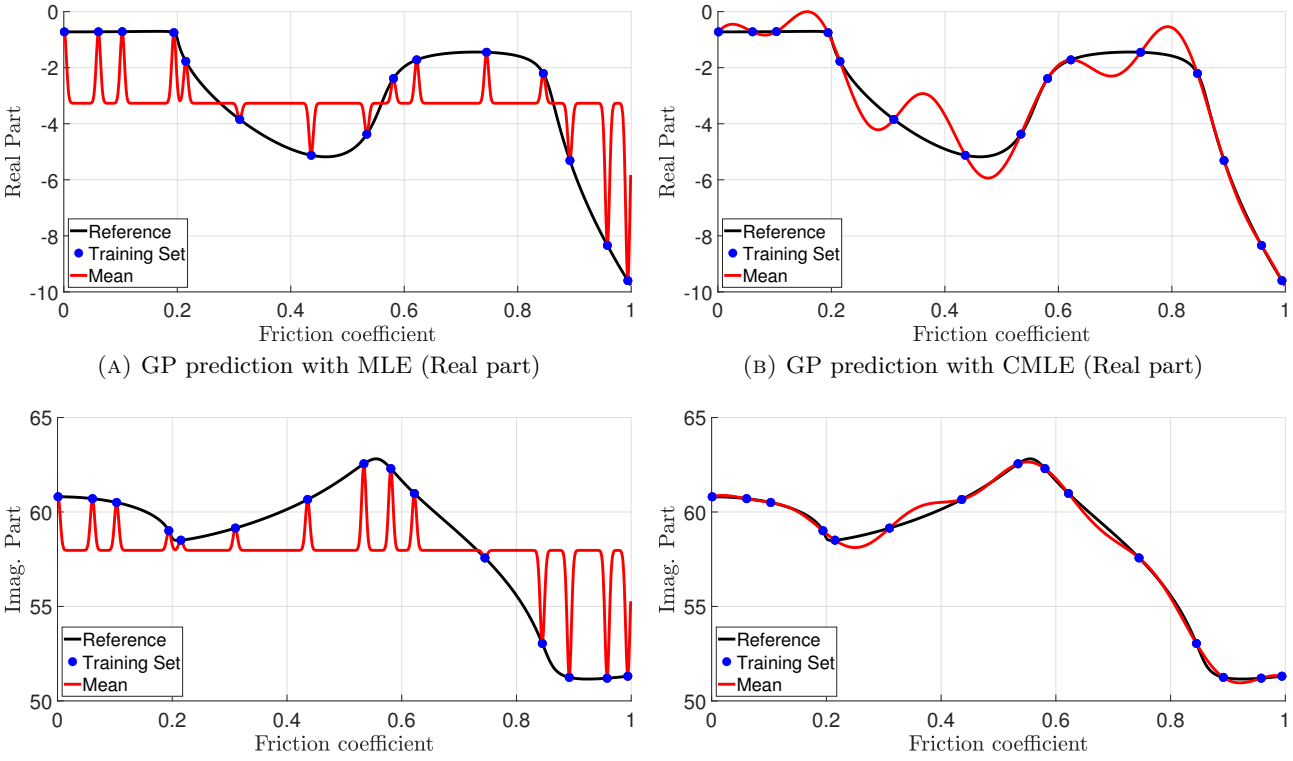
The purpose of this section is to assess the contributions of the given method, in an engineering context, namely the approximation of the eigenvalues of brake systems. The considered model is the Double Hulten, presented in Section 3.2.1. The aim of using this phenomenological model is to simulate squeal with a cheap model, allowing performing multiple assessments, without bothering about the computational cost. The approximated quantities are the eigenvalues λ_i of the system.

4.5.3.1 1-dimensional application

For this first test, the friction coefficient μ is considered. This parameter, which implies a nonlinear behaviour of the complex eigenvalues, is allowed to evolve between 0 and 1. Fig. 4.9 highlights the evolution of the real and imaginary parts of the third eigenvalue as a function of this parameter. The reference and GP predictions are drawn before and after correction of the lengthscales. In the present case, the hyperparameter k is equal to 0.5.

Here, the surrogate model result of Fig. 4.9(A) clearly does not succeed in producing a coherent approximation of the complex evolution of the considered function. Consequently, the GP is not able to assess for the different coupling states of the given frequency, causing it to be useless.

Considering the CMLE, the GP predictions are quite acceptable given the non-linearity of the studied functions and the number of samples.



(A) GP prediction with MLE (Real part)

(B) GP prediction with CMLE (Real part)

(C) GP prediction with MLE (Imag. part)

(D) GP prediction with CMLE (Imag. part)

FIGURE 4.9: Evolution of real and imaginary parts of the third eigenvalue as a function of the friction coefficient. Blue dots correspond to samples from the training set.

4.5.3.2 5-dimensional application

For this second test, five parameters are variable. Both mass parameters m_1 and m_2, k_a and c_a are added to the friction coefficient. A variation of $\pm 10\%$ around the nominal value is taken into account. Similarly to the previous test, 10 training sets are considered to highlight the efficiency of the suggested strategy for both satisfying and deceptive cases. The training sets are composed of 30 points, corresponding to a density of 6 points per dimension, which is a common practice in mechanical problems (in a pre-design phase, for example). The DUB is equal to -42.568 in this case, where the hyperparameter k is always equal to 0.5.

Table 4.9 exhibits the results of the considered squeal problem for the MLE/CMLE and the four quartiles. The results are given for the satisfying and deceptive training sets. Both results with or without the constraint for the deceptive training sets are considered.

The optimum for both satisfying and deceptive training sets with constraint is different from the DUB. On the other hand, the deceptive training sets without constraint always give an MLE equal to the DUB. Considering the quartiles, they are still equal to 0 for the deceptive surrogate optimised

Table 4.9
Quantities of interest for the three considered type of training sets.

Training set type	CMLE/MLE	Quartiles			
		1^{st}	2^{nd}	3^{rd}	4^{th}
Satisfying training set	-20.214	0.026	0.145	0.46	0.97
Deceptive training set (MLE)	-42.568	0	0	0	$1.73e^{-6}$
Deceptive training set (CMLE)	-64.019	0.197	0.277	0.393	0.85

with the traditional method, contrary to the surrogate optimised with the suggested method where the mid-range and high-range correlation values have been restored.

In addition, to highlight that a deceptive scenario can appear several times for a training set whose size is increasing, three training set sizes are respectively defined with 20, 30 and 40 samples. The training sets composed of 30 samples is built from that with 20 samples and 10 new samples. Moreover, the training sets of 40 samples is built from the training of 30 samples, where 10 new samples are added again.

Table 4.10
Quartiles for the three considered type of training sets.

Strategy	CMLE/MLE	Quartiles			
		1^{st}	2^{nd}	3^{rd}	4^{th}
MLE – 20 samples	-28.38	0	0	0	$2.85e^{-7}$
CMLE – 20 samples	-42.71	0.284	0.383	0.491	0.867
MLE – 30 samples	-42.568	0	0	0	$1.73e^{-6}$
CMLE – 30 samples	-64.019	0.197	0.277	0.393	0.85
MLE – 40 samples	-56.76	0	0	0	$4.7e^{-5}$
CMLE – 40 samples	-85.84	0.183	0.269	0.394	0.884

Table 4.10 shows the quartiles for each considered training set size. The first observation is that the issue of deceptiveness still appears, whatever the training set size, even when the number of samples is increased. The randomly distributed samples clearly do not allow a correction of the deceptiveness issue. Moreover, the deceptiveness issue is well-corrected as the quartiles indicate, when the suggested strategy is considered.

4.6 Chapter outcomes

The purpose of this chapter was to **investigate the deceptive prediction of the GP**. This issue is already known in the literature, but fewly tackled. Indeed, Jones et al. [163] was the first to coin out this terminology. However, they attempt to describe the potentially misleading prediction of the surrogate model by non-explored areas of the design space. Conversely, some strategies [166, 167] are suggested in the literature to overcome the faulty prediction of the GP, due to the loss of correlation in the covariance matrix of the random process. Nonetheless, these strategies are not relevant in the context of friction-induced vibration approximations.

Hence, the first proposal of this chapter is to **propose a recalibration of the terminology to describe this phenomenon**. Then, this phenomenon is comprehensively studied with regard to the evolution of the likehood and the covariance function with respect to the lengthscale. Following the conclusion of this study, **a detection criterion, the Deceptive Upper Bound, and a numerical strategy to overcome the issue are proposed**. The first one hinges on the mathematical analysis of the phenomenon and is computed from a limit analysis. The latter one primarily derives from the aforementioned criterion. The main idea is to train the surrogate model and to automatically analyse the likelihood value. Then, depending on the criterion value, the GP is trained once again with the ADT strategy. This strategy corresponds to the traditional optimisation of the likelihood, but with an additional constraint to prevent the likelihood from reaching the DUB. **This constraint is only parameterised by one hyperparameter, one key aspect which shows the simplicity of the ADT method.**

The analysis of the phenomenon showed that **the main cause of this issue is that a component of the lengthscale vector goes to 0^+**. Using limit operations, this simple fact causes the driving equations of the GP to become ill-defined. Indeed, the covariance matrix takes the unit form and the mean and variance of the GP respectively become constant and equal to one, on the majority of the design space. From these observations, the DUB criterion is stated by simply taking the limit of the likelihood, when any lengthscale component goes to 0^+.

The first experiment mainly focused on providing insights on how the suggested strategy is behaving. To do so, the Xiong's function was considered and the performance of the strategy was assessed by monitoring the evolution of the correlation values in the covariance matrix. In addition, a discussion on the setting of the hyperparameter of the strategy was performed. The different results showed that **values higher than 0.5 for this hyperparameter allows a restoration of the correlation between the samples of the training sets**. Additionally, an analysis of the convergence of both mean and variance of the GP showed that, with a convergence threshold of 10^{-2}, these two quantities exhibit a good stabilisation for a hyperparameter value of 0.6.

The second experiment showed the performance of the suggested method for a large benchmark of functions, which exhibit different levels of non-stationary and non-linearity. Two sets of 10 randomly generated training sets were studied: the first one was specially designed not to produce deceptive predictions, while the latter one was designed to produce deceptive predictions. The different results showed that **the suggested strategy has a 100% rate of detection with regard to the phenomenon**. In addition, the correlation between the samples of the training set was successfully restored for all the considered functions.

Finally, the third experiment proposed an application of the suggested strategy for the approximation of the eigenvalue of a brake squeal problem. First, a one-dimensional application is considered to show the consequence of approximating the different coupling states with a deceptive GP prediction. Then, a 5-dimensional application is performed to study the effects of increasing the training set on the propensity of the surrogate model to be deceptive. The different results showed that, **when the**

prediction is deceptive, the coupling state of the system cannot be efficiently identified. This observation is quite problematic, especially when considering the use of the surrogate modelling strategy in an industrial context. Additionally, **a simple increase of the training set size cannot be considered an efficient strategy to deal with this issue**, as shown by the last test.

Chapter 5

Multi-Infill Bayesian Optimisation: a better way to determine the optimum of an objective function

Contents

5.1 Introduction

The optimisation of functions is a complex task to tackle. A wide variety of problems (*e.g.* engineering design, machine learning) is concerned by extremely expensive objective functions. These problems are generally characterised by black-box objective functions, whose algebraic definitions are unknown or computationally expensive. Indeed, in engineering design, the objective function, evaluated through a solver, may take several hours, several days or even months to be computed.

To tackle these optimisation problems, many algorithms were developed. For continuous optimisation, Talbi [174] distinguishes the iterative methods from the metaheuristics. The iterative methods correspond to the methods that usually get trapped in local minimum: stochastic gradient descent [175], Adam [155] or AdaGrad [176]. Conversely, the metaheuristics have the ability to escape from local minima but with an extra-computational cost for optimising the function.

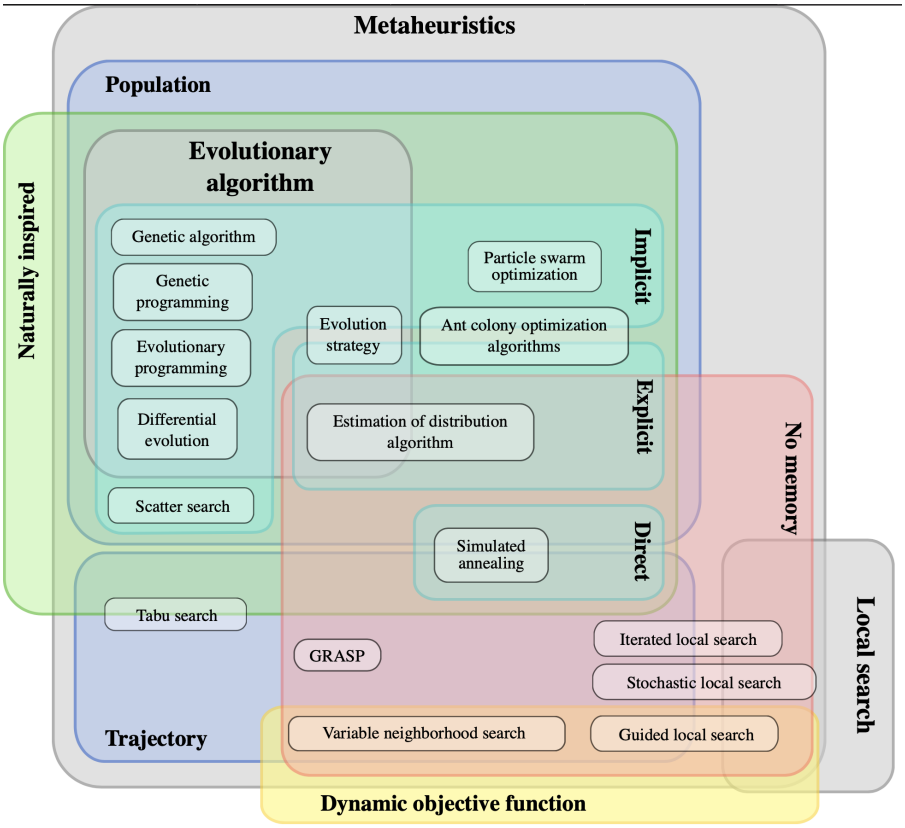


FIGURE 5.1: Classification of metaheuristics [4]

Fig. 5.1 presents a non-exhaustive classification of the different metaheuristics. Despite their better efficiency, these algorithms typically involve multiple calls to the objective function, making it hardly usable with engineering problems.

For such expensive optimisation problems, a popular strategy is to use BO [177]. This algorithm hinges on approximating the objective function with a surrogate model and then exploring the most promising locations of the design space. The surrogate model is then refined and aims at the problem

optimum. This approach was used in hyperparameter tuning [178], engineering design [162, 164], robotics [179], drug design [180] or reinforcement learning [181, 182].

The determination of the most promising areas of the design space is carried out by using an acquisition function (or infill criterion). A lot of acquisition functions are available in the literature, mainly because no infill criterion dominates the other. Besides, the best choice can change during the execution of the process, as stated by Shahriari et al. [183]. Traditionally, the acquisition function provides new samples aiming for exploration, exploitation or both.

The Probability of Improvement [184] generates samples that aims for the optimum of the objective function, but is known to focus too aggressively on exploration [163]. The Expected Improvement is the most widely used criterion [162, 181] and the convergence rate of this criterion was proven in [185]. The Upper Confidence Bound [186, 187] aggregates directly the two momenta of the GP surrogate model with a coefficient to control the balance between exploration and exploitation. Besides, Srinivas [187] showed that a good tuning of this coefficient can offer significant performance improvements. Other infill criteria can be used: Entropy Search [188], Thomson sampling [189] or portfolio of infill criteria [190].

Recently, new approaches [191, 192] considering multiples samples were introduced as q-EI strategy. This intuition behind q-EI is to divide the design space into q-areas and add one point per area. Other authors [193, 194] suggested strategies relying on multiple samples. Bisch et al. [193] adressed BO as a multiobjective infill problem by using directly the mean and the variance of the GP. Desautels et al. [194] proposed a multipoint strategy relying on the Upper Confidence Bound for Bayesian Optimization.

Following the works of these authors, this chapter proposes a new strategy, hinging on the definition of a multiobjective infill criterion Bayesian optimisation. Two ways to perform this new strategy are defined from the consideration of the EI and the variance.

To describe and assess the performance of the suggested strategy, this chapter is organised as follows. Section 5.2 presents the Bayesian Optimisation process, by detailing the considered acquisition functions in this chapter and proposing an application of the algorithm. Then, Section 5.3 discussed the limitation of Bayesian Optimisation with regard to a certain type of functions. A new method is proposed to improve the overall performance of the Bayesian Optimisation algorithm. Finally, Section 5.4 assesses the performance of the new strategy against the traditional Bayesian Optimisation process.

5.2 Bayesian Optimisation

5.2.1 Bayesian optimisation presentation

The Bayesian Optimisation process [195] is a strategy hinged on the surrogate modelling approach. Here, the process is given for the GP surrogate model, but can be used with DGP [135] or DNN

[196]. The main intuition behind this strategy is to increment the training set on which the surrogate approximation is built to increase the available knowledge of the surrogate model. This acquisition of knowledge is traditionally performed to aim at the global optimum of the approximated function.

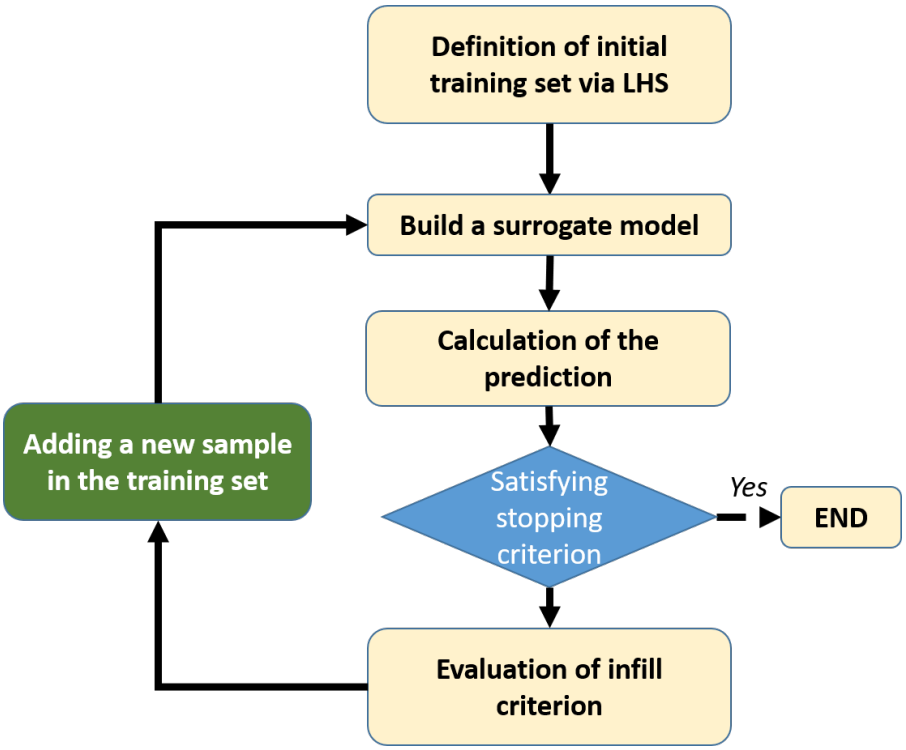


FIGURE 5.2: Bayesian Optimisation workflow.

Fig. 5.2 exhibits a typical workflow of a Bayesian Optimisation process. First, an initial training set is generated to allow the surrogate model making a primary approximation. Second, the prediction of the mean and variance of the GP is performed. Then, an infill criterion (or acquisition function) is evaluated to determine the new sample to add to the training set $\mathcal{T}$. Finally, this procedure is repeated until a stopping criterion is satisfied.

The basic idea behind the infill criterion is to highlight the potential behaviour of the approximated function. The following reviews [181, 183, 197] give a comprehensive overview of the nature and applications of these different acquisition functions. As pointed out in the introduction, the infill criterion is a compromise between the exploration and exploitation.

First, the exploration aims at giving insights of the areas of the design which have been sub-explored and where the surrogate model approximation is highly uncertain. Conversely, the exploitation allows scanning the areas of the design already explored and which present a high probability of holding the true global optimum of the approximated functions.

Traditionally, the Expected Improvement is the most thoroughly used for the optimisation, while the variance is considered for the exploration of the design space [183].

Finally, with regard to the end computation of the algorithm, no stopping criterion is efficient to work in all cases. Indeed, the first approach is to consider that the algorithm converges when the

acquisition function level is under a certain threshold. In practice, this method does not succeed in providing a good convergence, as it is emphasised in Section 5.3.1. Another method is to consider a computational budget, where the user has a certain amount of available computations. In this case, the convergence of the algorithm is not assessed. In the following applications of the Bayesian Optimisation process, the end computation of the algorithm is performed considering the computational budget.

5.2.2 Acquisition functions

Expected Improvement (EI)

The Expected Improvement [162] is an infill criterion that combines both exploration and exploitation by aggregating them together (Eq. 5.1). The first term of the equation corresponds to the exploitation term, while the second term to the exploration term.

$$C_{EI}(x) = (y_{min} - \widehat{\mu}(x_*)) \left[\frac{1}{2} + \frac{1}{2}\,\mathrm{erf}\left(\frac{y_{min} - \widehat{\mu}(x_*)}{\sqrt{2}\,\widehat{s}(x_*)}\right)\right] + \frac{\widehat{s}(x_*)}{\sqrt{2\pi}} \exp\left[-\left(\frac{y_{min} - \widehat{\mu}(x_*)}{(\sqrt{2}\,\widehat{s}(x_*))}\right)^2\right] \tag{5.1}$$

where y_{min} is the minimum of the evaluated solutions of the current training set at an iteration i.

Variance (Var)

The Variance corresponds to the second moment of the GP output random variable. It only carries the exploration term and does not provide exploration capabilities.

$$C_{var}(x_*) = \widehat{s}(x_*) \tag{5.2}$$

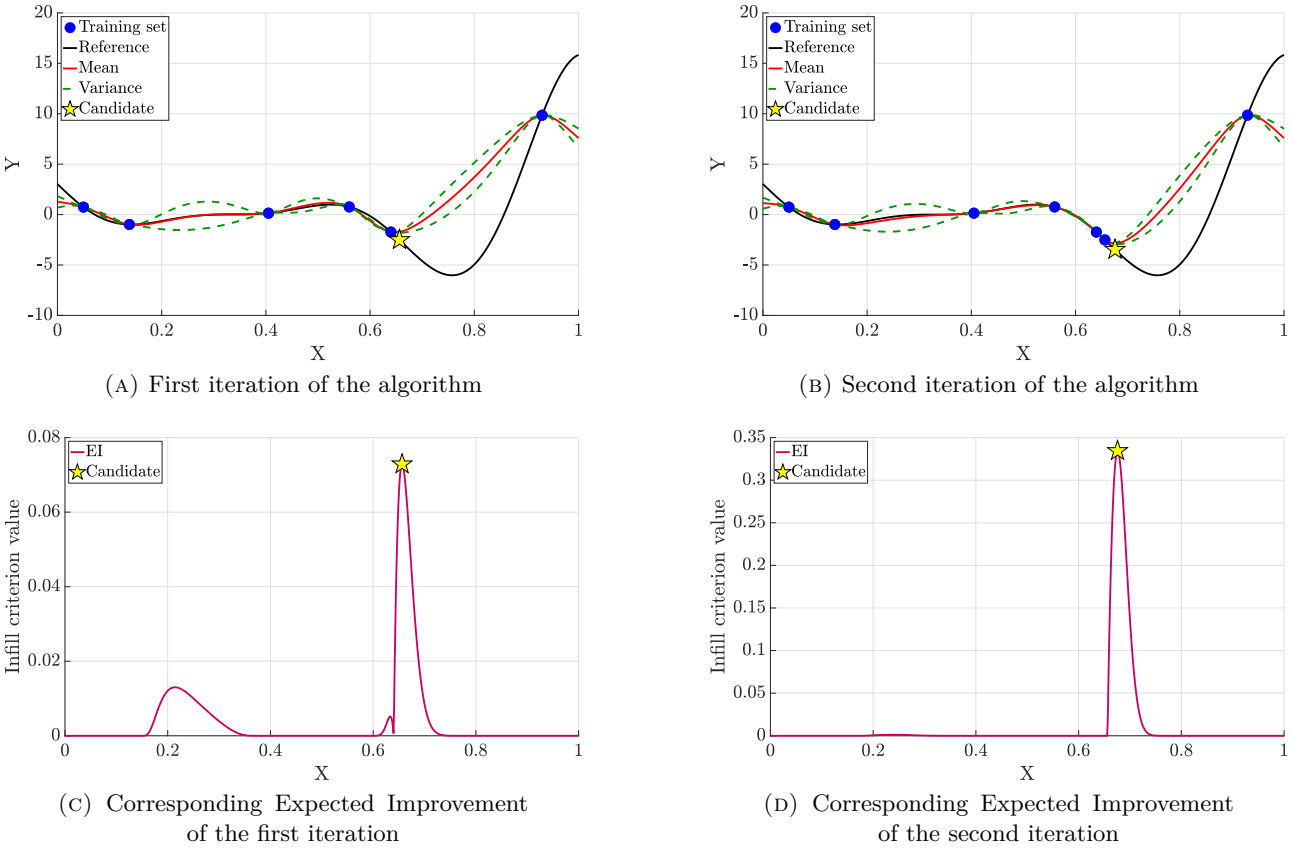
5.2.3 Application of the Bayesian Optimisation algorithm

The figures 5.3(A) and (B) give an example for two successive iterations of the application of the BO algorithm to optimise a one-dimensional function, given by (Eq. 5.3).

$$y(x) = (6x - 2)^2 \times \sin(12x - 4) \tag{5.3}$$

The Expected Improvement expresses the interest of two areas of the design space for the first iteration of the algorithm (Fig. 5.3(C)): the area located between 0.17 and 0.35 and the area located between 0.61 and 0.67. The second area is the one that maximise the EI and, consequently, a new sample is evaluated at this location. Then, for the second iteration (Fig. 5.3(D)), the surrogate model

prediction is updated. The BO algorithm has increased its confidence in finding the area that yields the global optimum by focusing on the second location.



(A) First iteration of the algorithm

(B) Second iteration of the algorithm

(C) Corresponding Expected Improvement of the first iteration

(D) Corresponding Expected Improvement of the second iteration

FIGURE 5.3: Two successive iterations of the Bayesian Optimisation process.

Fig. 5.4 shows the convergence of the algorithmn. In that case, the slow convergence is surely slow, but the algorithm converges toward the global optimum anyway.

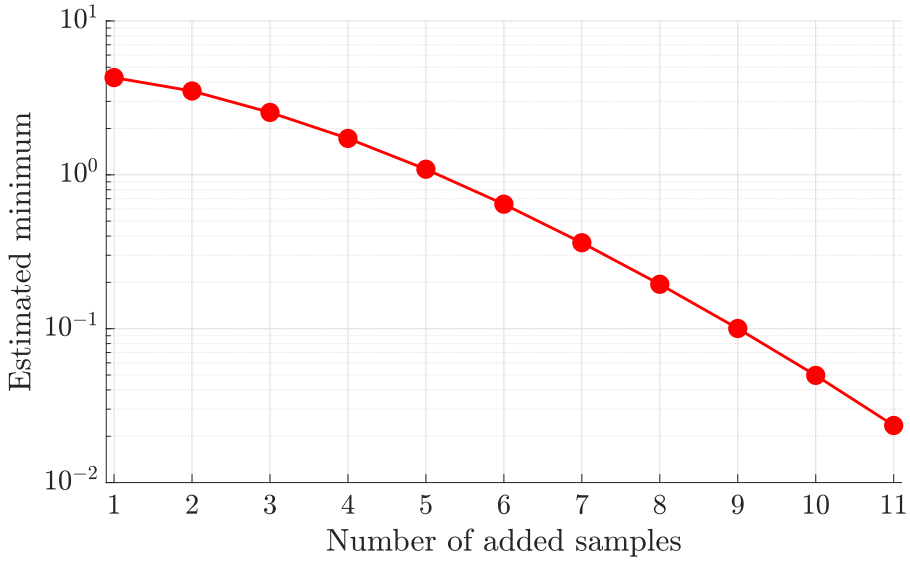


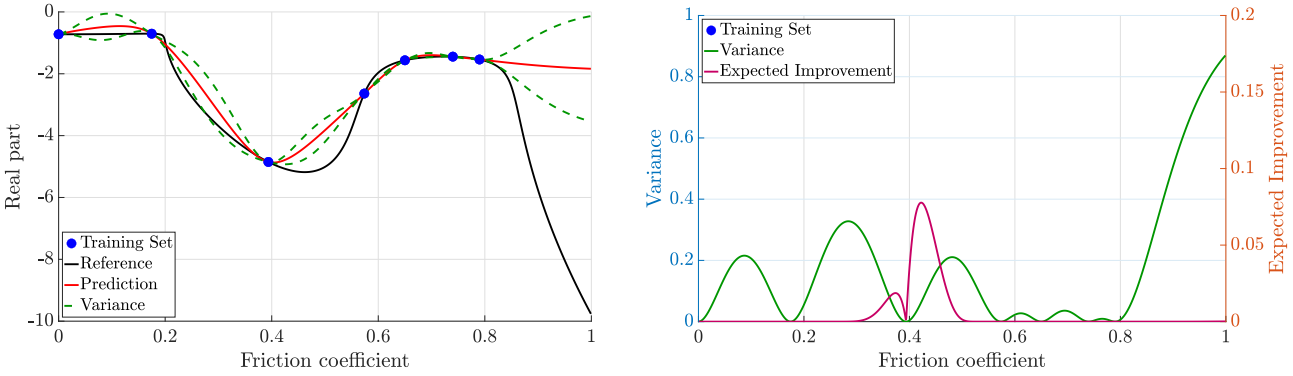
FIGURE 5.4: Convergence of the algorithm.

5.3 Proposal

5.3.1 Issue with the traditional Bayesian Optimisation workflow

The following application highlights a limitation of the traditional Expected Improvement. The function to optimise is the third real part of the eigenvalue of the Double Hulten (presented in Chapter 3). The infill criteria (EI and Variance) are evaluated for the first iteration of the BO algorithm. Then, the overall convergence process is assessed to highlight the issue of using the traditional EI.

The samples of the training set are generated with an LHS. The GP used to approximate the evolution of the considered function is set with a Matern 3/2 kernel function and the variance and kernel noises are fixed, following the reasons enumerated in Section 3.4.1.1. Fig. 5.5(A) shows the mean and variance prediction of the GP and the corresponding evaluation of the considered acquisition functions. The global minimum is located at a friction coefficient of 1, and the considered function exhibits a local minimum for a friction coefficient of 0.475.



(A) GP prediction (B) Infill criteria evaluation

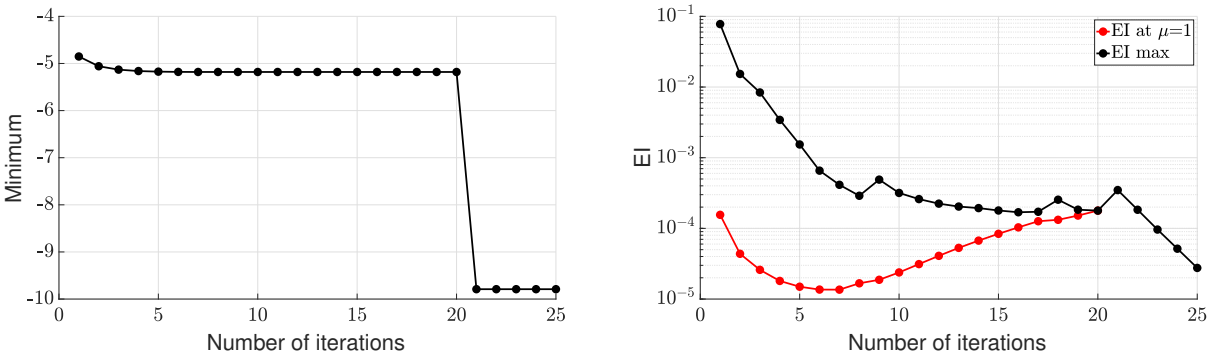
FIGURE 5.5: Prediction of the GP and the associated infill criteria evaluation.

Considering the locations of the samples in the design space, the GP prediction is quite logical. Indeed, for a friction coefficient between 0.8 and 1, the prediction is erroneous as it would be expected from the lack of sampling in this location. Consequently, the variance in this location is important.

The variance infill criterion, given in Fig. 5.5(B), exhibits a maximum for a friction coefficient value of 1, inducing a new sample for this value, if this acquisition function would be used. Conversely, the EI (Fig. 5.5(B)) shows a maximum for a friction coefficient value of roughly 0.42.

Observations

Fig. 5.6(A) shows the convergence of the BO algorithm for the considered function. The convergence is very slow since the algorithm need 21 iterations to reach the global optimum of the function. Roughly 30 samples have to be evaluated for a one-dimensional problem.



(A) Evolution of the minimum value

(B) Evolution of the two location EI values

FIGURE 5.6: Convergence of the BO algorithm and assessment of the maximum EI value and the EI value associated with the friction coefficient value of 1.

Fig. 5.6(B) exhibits the evolution of two locations EI values, namely the maximum EI value and the EI associated with the friction coefficient value of 1. The latter one is always associated with a small value of EI (between 10^{-4} and 10^{-5}), indicating a small potential of holding the true global optimum of the considered function, with regard to this infill criterion. Moreover, at the 20^{th} iteration, the maximum EI value has decreased until reaching the same value as the one associated with the friction coefficient value of 1.

Discussions

From the following observations, two main remarks arise. First, the algorithm is obviously stuck in the local optimum, located at a friction coefficient value of 0.475. Besides, all new samples between the first and the twentieth iteration are added in this area. This local minimum has thus a large influence area due to the bowl shape of the function, which has proven to be challenging for the EI infill criterion.

Second, the EI acquisition function does not succeed in describing the potential of all the design space areas, resulting in a lack of exploration. Here, the area associated with a friction coefficient of 0.8 to 1 always exhibits an important difference $y_{min} - \widehat{\mu}(x_*)$, causing the second term of (Eq. 5.1) to be very small. Moreover, this area is not interesting with regard to the exploitation term, since the difference $y_{min} - \widehat{\mu}(x_*)$ is obviously significant.

5.3.2 Multi-Infill Bayesian Optimisation process

As the exploration term of the EI can become useless in certain conditions, another strategy is needed. Moreover, the Variance infill criterion cannot be considered as it owns any exploitation component. Instead of proposing a complex strategy (portfolio of infill [190] or Entropy Search [188]), the main proposal is to combine the two considered acquisition functions: Variance and Expected Improvement.

To do so, the Bayesian Optimisation is stated as a bi-objective problem, following Bisch's proposal [193], where the first objective corresponds to the exploration and the second to the exploitation. Typically, two main approaches exist in the literature to deal with multi-objective problem [198]. The first strategy transforms the multi-objective problem into a mono-objective (aggregating or scalarising) and is already used with EI [162] or UCB [187]. The second strategy is to construct a Pareto front.

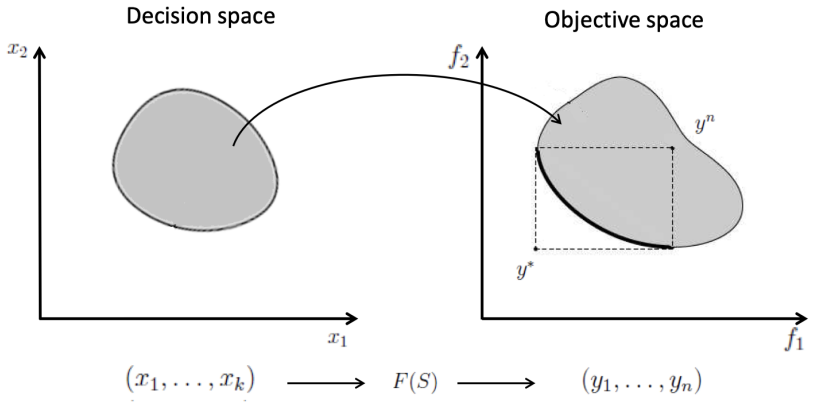


FIGURE 5.7: Design space and objective space [5].

In Fig. 5.7, the design space shows the possible values that two variables x_1 and x_2 can take. Evaluating this design space with the functions f_1 and f_2 results in the gray shape, which is expressed in the objective space, with f_1 in abscissa and f_2 in ordinate. The Pareto front corresponds to the bold black line, which is the set Y^* of solutions that dominates all the solution y^n [199, 200].

In this case, the two objectives are conflicting. Indeed, the set of solution that minimises f_1 (far left of the Pareto front) is maximising f_2, and vice-versa for the minimisation of f_2. This conflict induces difficulties in decision-making. Several approaches exist to determine the most efficient candidate and usually rely on the decision-maker [201, 202].

Hence, the proposed strategy is to define a Pareto front of solutions, while considering the Expected Improvement and the Variance as the two objectives of this Pareto front. This new strategy is called Multi-Infill Bayesian Optimisation (or MIBO) and is available in two different versions.

Fig. 5.8 gives an overview of the suggested new strategy.

Multi-Infill Bayesian Optimisation - Independent

This version of MIBO is the simplest one: the idea is to add at each iteration a sample determined by maximising the Expected Improvement and one by maximising the Variance. This strategy is denoted MIBO-Independent (or MIBO-I).

Fig. 5.9 shows the solutions y^n in the objective space in red, while the Pareto front is given in black. In addition, the maximum EI value and the maximum Variance value are located in the Pareto front. This front is determined with an NSGA-II algorithm [203], implemented in the jMetal toolbox [204].

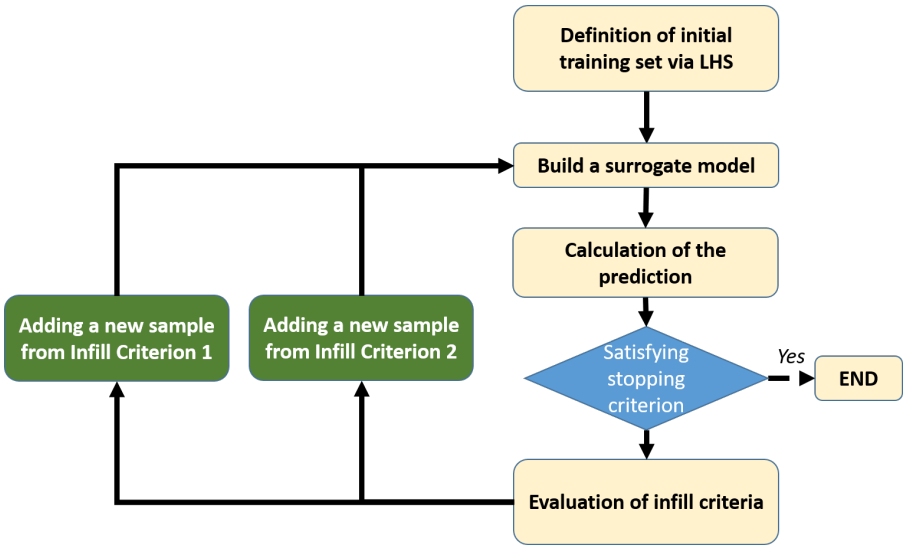


FIGURE 5.8: Multi-Infill Bayesian Optimisation workflow.

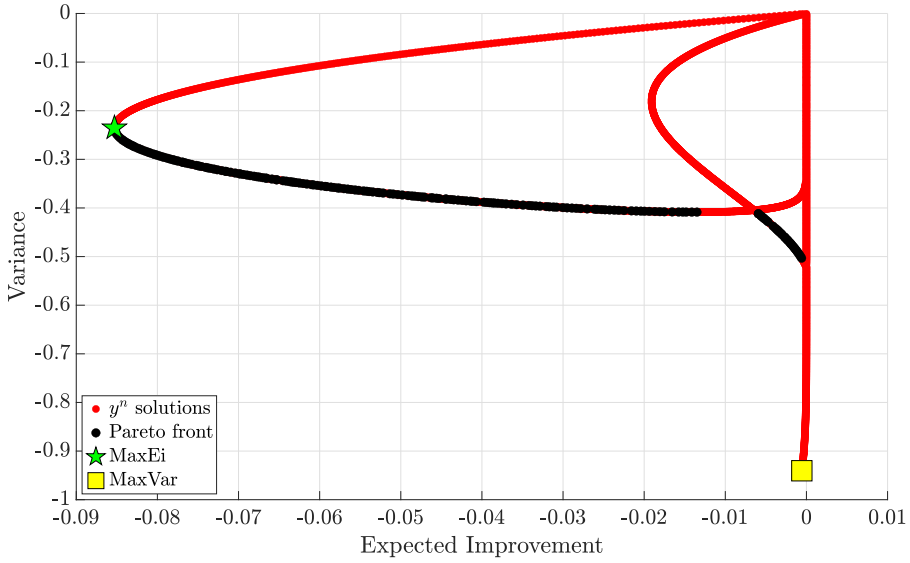


FIGURE 5.9: Pareto front for the Double Hulten application with the MIBO-I strategy.

The location on the Pareto front of the new candidates gives insights on the nature of these samples. For instance, the MaxVar sample is a full exploration candidate, while the MaxEI is a trade-off candidate as it maximises the EI and generates a variance value of roughly 0.2.

Multi-Infill Bayesian Optimisation - Pareto

This version of MIBO consists in taking advantage of the diversity of candidates in the Pareto front. The idea is still to add two samples per iteration of the algorithm, but these new candidates are not on the extremities of the Pareto front, rather in the middle of it. This strategy is denoted MIBO-Pareto (or MIBO-P).

To solve the issue of decision-making, the proposal is to use clustering methods [205, 206]. The idea of these methods is to create a certain amount of groups depending on the similarity between them.

Three major classes of clustering can be used to generate these groups: the K-Means [207], DBScan [208] and the hierarchical methods [209]. The K-Means usually creates groups with a measure of the distance between the instances and by pre-setting the number of cluster. The DBScan is a hierarchical method that determines the number of clusters with a measure of the density between the instances. Finally, the hierarchical methods create a tree of instances by merging or dividing the instances into groups. In the sequel, the K-Means is chosen for its simplicity of implementation.

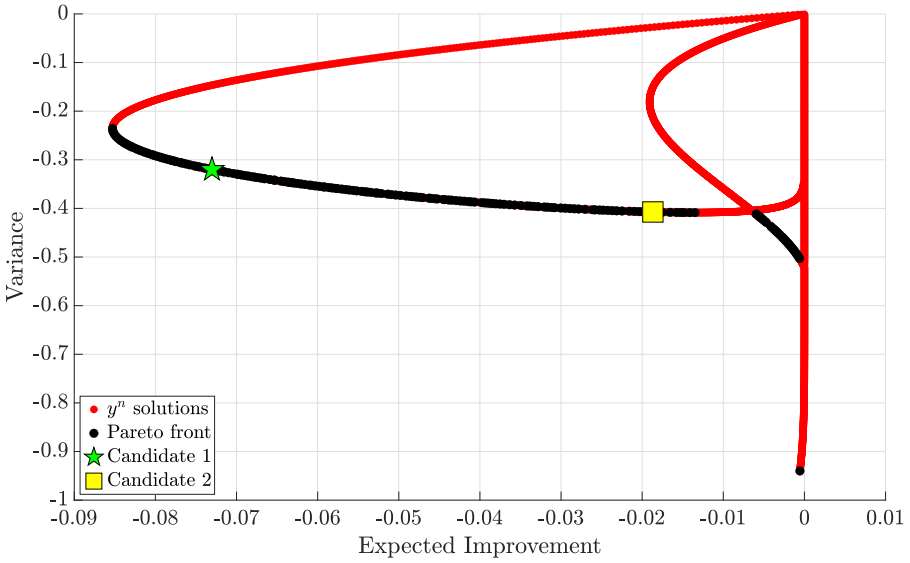


FIGURE 5.10: Pareto front for the application of Section 5.3.1 with the MIBO-P strategy.

Fig. 5.10 shows the Pareto front on which both candidates are located in the Pareto front and are determined with the K-Means implementation of [157]. Here, the new candidates are neither full exploitation samples nor exploration. They are rather a combination of both objectives.

5.4 Numerical experimentations

5.4.1 One dimensional problem

In this section, the suggested method is tested on the Xiong's function, already used in Section 4.5.1 to study the deceptiveness of GP predictions. Ten random initial training set are generated to assess the reproducibility of the suggested method. The traditional EI is denoted TradBO and is set for 60 iterations. The two suggested methods (MIBO-I and MIBO-P) are set for 30 iterations. The MIBO-P is set according to the default parameters of the Scikit-Learn K-Means [157]. The global optimum of the Xiong's function is -0.609.

Observations

First, the convergence of the TradBo strategy is troublesome. Indeed, at the 60^{th} iteration, the absolute gap between the global minimum and the determined minimum is roughly 0.4 (*i.e.* an estimated

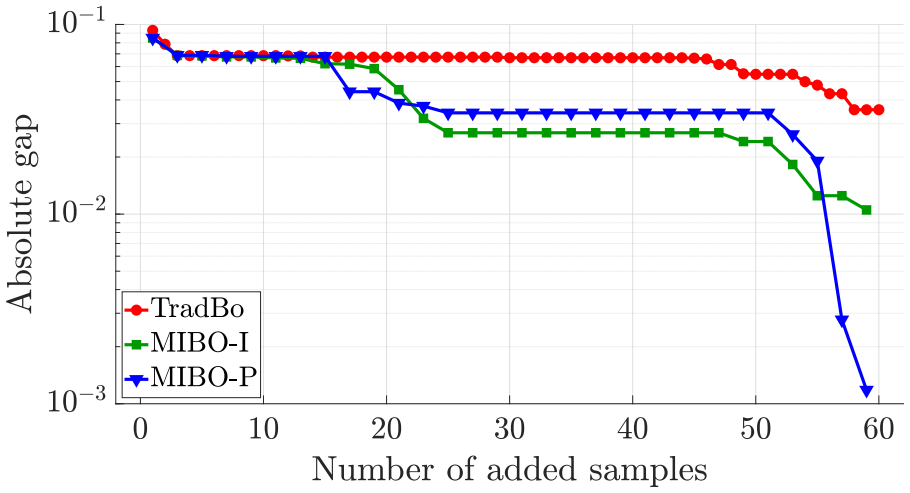


FIGURE 5.11: Convergence rates of the three considered Bayesian Optimisation strategies for the Xiong's function.

minimum of 0.569), indicating the convergence in a local minimum. Moreover, the convergence pattern shows that the algorithm remains stuck in a local minimum during roughly 50 iterations. Conversely, both suggested methods escape from this local minimum quite early during the execution of the algorithm. They visit a second local minimum during 15 iterations (*i.e* 30 added samples) and, then, find the global optimum of the considered functions. The MIBO-I produces a final value of -0.608, against -0.6089 for the MIBO-P – both values being fairly acceptable.

Discussions

Here, the suggested methods (MIBO-I and MIBO-P) produces better results than the traditional strategy for Bayesian Optimisation. It shows that the issue of concentrating in a local minimum can be overcome by adding samples determined from other infill criteria. Besides, the performance of the MIBO-I strategy is interesting due to its simplicity. Indeed, the method only needs the definition of the two considered infill criteria, without having to set other hyperparameters.

5.4.2 Benchmark of functions

This section aims at evaluating the performance of the proposed strategy on a benchmark of mathematical functions described in Table 5.1. The functions benchmark is available in the literature [171–173] except for the functions n°1 to 3, which are presented in Appendix A. All of these functons exhibit high levels of non-stationarity and non-linearity, except for the function n°3. This function is smooth and nearly quadratic, but the global minimum of the function is hard to determine because of highly attractive local minima.

The suggested algorithm is fed with initial knowledge acquired by ten randomly generated training sets. These training sets are determined by a traditional LHS method. Three initial training sets are assessed for the test : four, five and six points per dimension. Preliminary tests on the suggested strategy showed that using too many initial samples tends to downgrade the performance of MIBO.

Table 5.1
Properties of the benchmark functions.

Function	Dimension	Non-stationarity	Non-linearity	Multi-modal
Xiong	1D	High	High	Mid
Function n°1	2D	High	High	Mid
Function n°2	2D	High	Mid	High
Function n°3	2D	Low	Mid	Low
Powell	4D	-	-	High
Ackley	10D	-	-	High
Powell	12D	-	-	High

To optimise the acquisition function, several preliminary tests emphasised that determining the global optimum of this function can become prohibitive while using metaheuristics such as CMAES or DE [197]. Instead of depending on these methods, the strategy suggested by [210] is considered. A large MC sampling is evaluated and the optimum of this sampling is determined. Then, a local optimiser (namely, the LBFGS) is initialised with this optimum. Finally, this operation is performed 10 times.

The performance of the algorithm is evaluated by computing the absolute gap between the global minimum of the studied function and the estimated minimum. The suggested method is compared to the traditional Expected Improvement strategy.

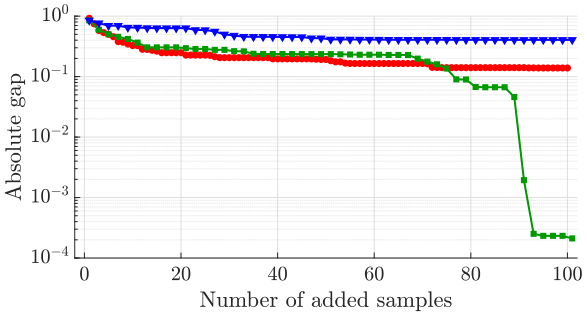
The following results are presented in three sections, highlighting the different performance levels. The first section shows the benchmark functions where the Independent version of the suggested algorithm is the most efficient to determine the global minimum of the studied function, while the second section considers the function where both methods outperform the traditional bayesian strategy. Finally, the third section emphasises the benchmark functions where the Pareto version of the suggested algorithm is the most efficient.

5.4.2.1 Case 1: MIBO-Independent outperforms the other methods

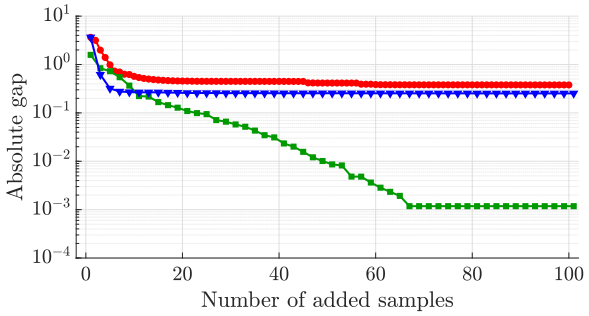
Fig. 5.12 shows the results of the benchmark functions n°2 (left column figures) and n°3 (right column figures). The global optima of function n°2 and n°3 are respectively -1.3525 and -1.1895.

For both functions, the MIBO-I outperforms the two other methods, namely the MIBO-P and the traditional EI, whatever the considered number of initial samples. The MIBO-P does not perform well for the function n°2 while it provides slightly better performance for the function n°3. Nonetheless, the difference is quite small between the last two methods.

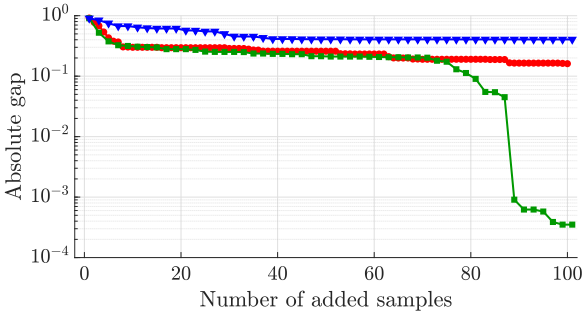
For instance, with 4 initial samples per dimension, the MIBO-I provides an absolute gap of roughly 10^{-4} for the function n°2 and 10^{-3} for the function n°3. Conversely, the MIBO-P respectively gives an absolute gap of 0.4 and 0.2, while the traditional EI performs better with an absolute gap of roughly



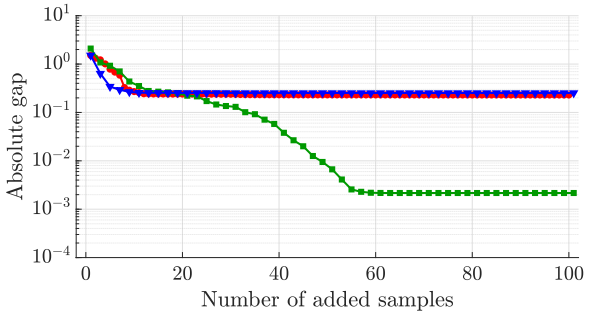
(A) Function n°2 - 4 initial points per dimension



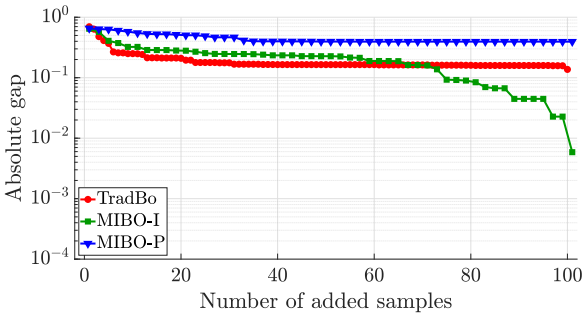
(B) Function n°3 - 4 initial points per dimension



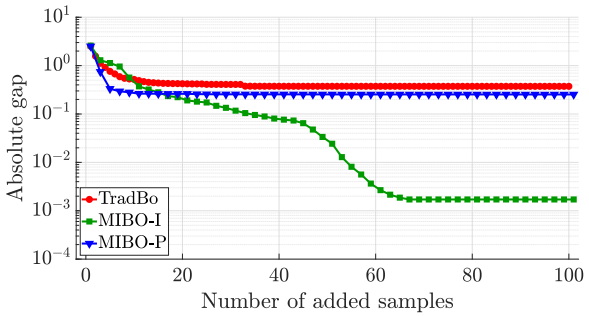
(C) Function n°2 - 5 initial points per dimension



(D) Function n°3 - 5 initial points per dimension



(E) Function n°2 - 6 initial points per dimension



(F) Function n°3 - 6 initial points per dimension

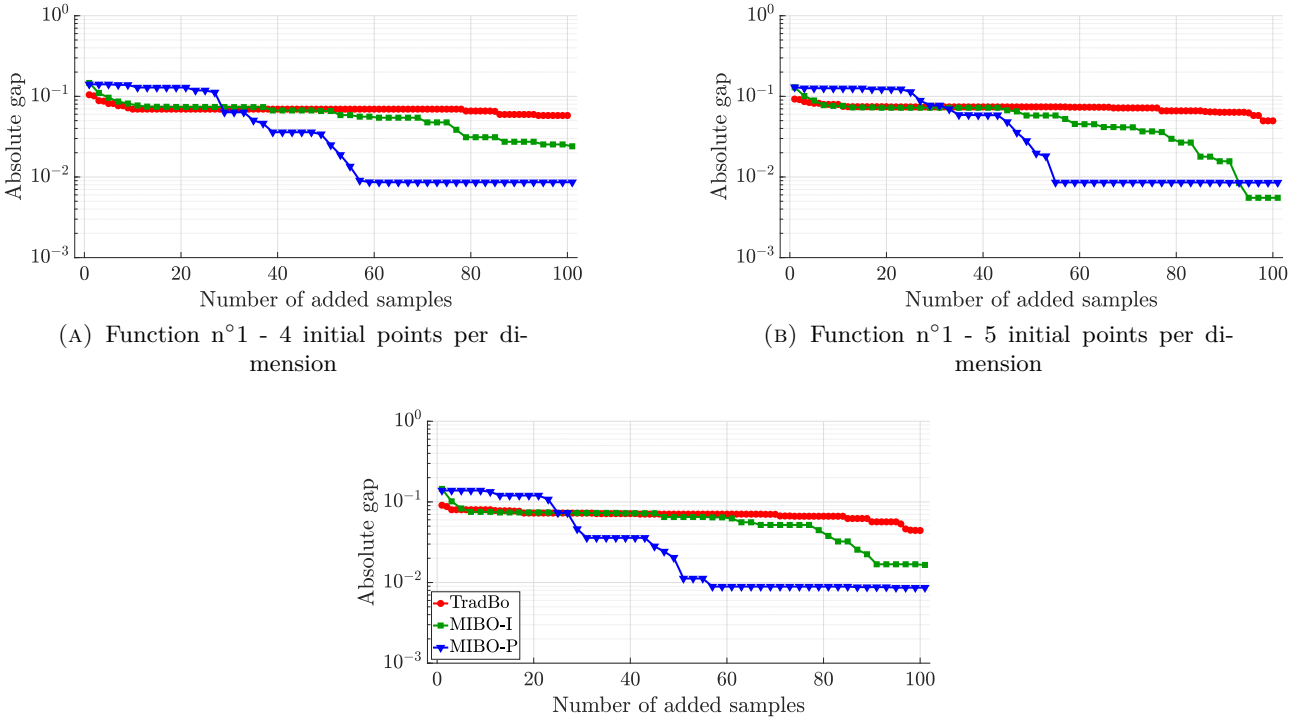
FIGURE 5.12: Convergence rates of the three considered Bayesian Optimisation strategies for the functions n°2 and n°3.

0.1 and 0.3. Hence, the absolute gap observed for the MIBO-P and EI are not good with regard to the global optima of both considered functions, highlight the convergence in a local minimum.

For the function n°2, the increase of the number of initial samples slows the convergence toward the global optimum of the function. Indeed, the final absolute gap is respectively equal to 10^{-4} and 10^{-2} with 5 and 6 initial points per dimension. Conversely, no significative difference is noticed for the function n°3.

5.4.2.2 Case 2: MIBO-Independent and MIBO-Pareto outperform the traditional method

Fig. 5.13 shows the results of the benchmark function n°1. The global optimum of this function is -0.833.



(A) Function n°1 - 4 initial points per dimension

(B) Function n°1 - 5 initial points per dimension

(C) Function n°1 - 6 initial points per dimension

FIGURE 5.13: Convergence rates of the three considered Bayesian Optimisation strategies for the function n°1.

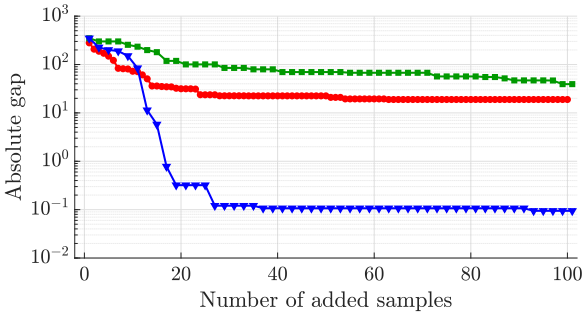
Whatever the initial training set size, both MIBO-I and MIBO-P outperform the traditional EI. Nonetheless, two levels of performance are distinguished from the results: first, for 4 and 6 initial samples per dimension, the MIBO-P performs better than the other suggested algorithm; second, for 5 initial samples per dimension, the MIBO-I is slightly better than the MIBO-P.

The difference between the two suggested methods remains quite small, since both methods give an absolute gap of approximately 10^{-2}. Conversely, the traditional EI does not provide good results since the absolute gap is about 10^{-1}, suggesting a local minimum.

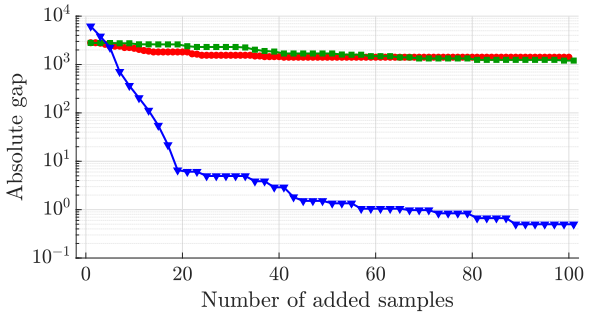
5.4.2.3 Case 3: MIBO-Pareto outperforms the other methods

Fig. 5.14 shows the results of the Powell 4D and 12D. The global optimum of both function is 0.

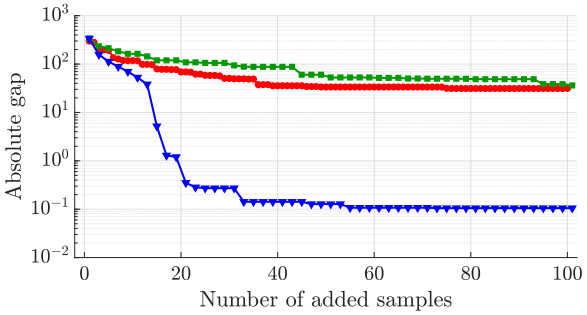
For these functions, the results of MIBO-I are quite troublesome, since the given strategy is less efficient than the traditional method and generates an important absolute gap (around 10^2 for the



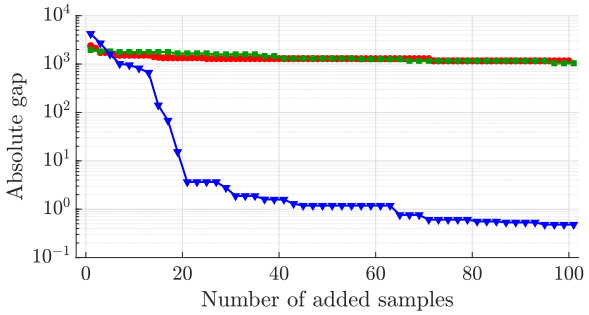
(A) Powell 4D - 4 initial points per dimension



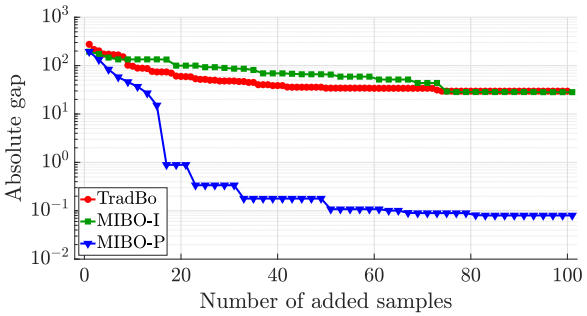
(B) Powell 12D - 4 initial points per dimension



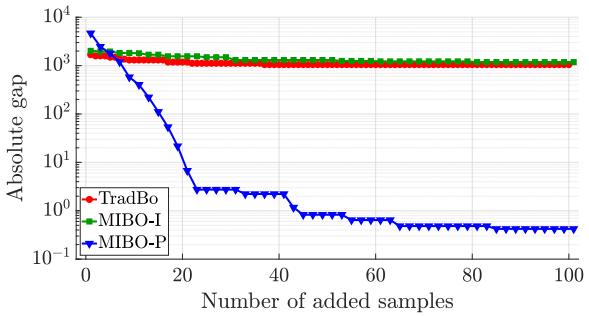
(C) Powell 4D - 5 initial points per dimension



(D) Powell 12D - 5 initial points per dimension



(E) Powell 4D - 6 initial points per dimension



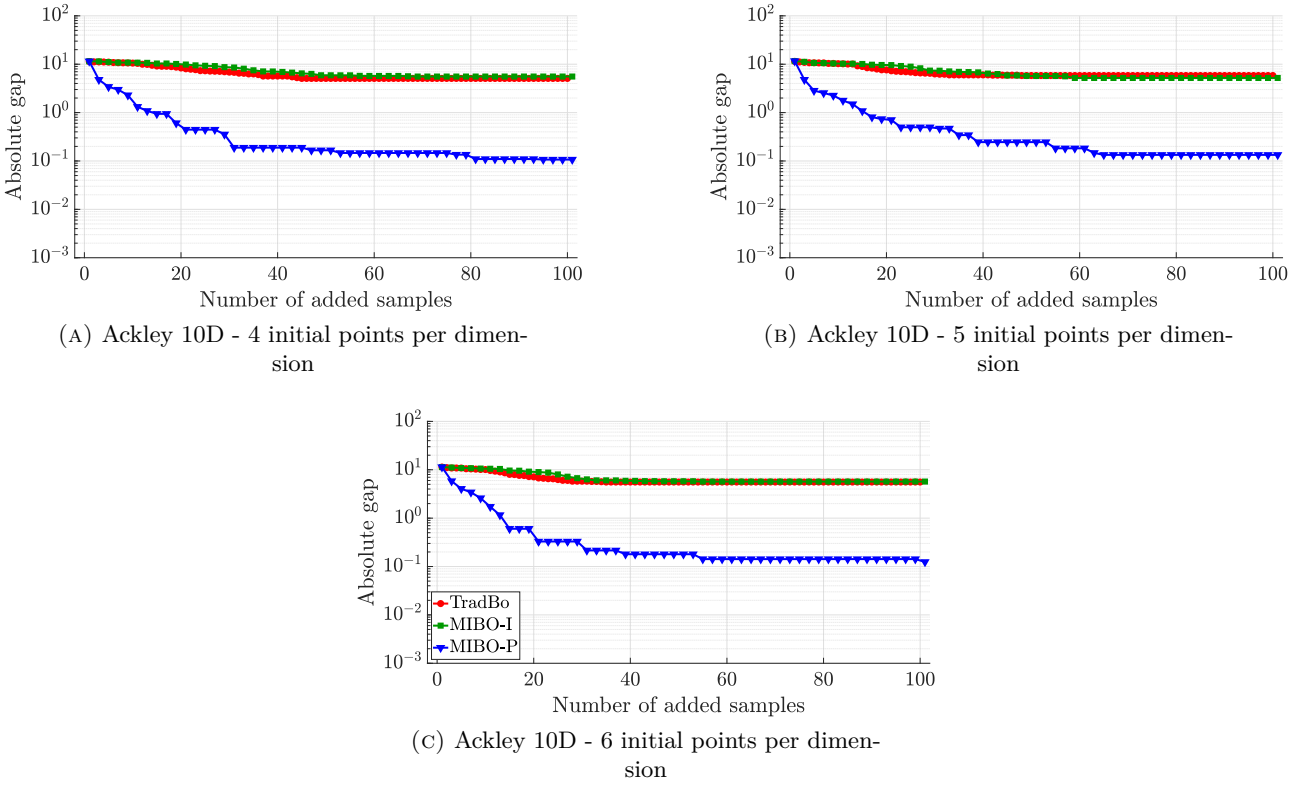
(F) Powell 12D - 6 initial points per dimension

FIGURE 5.14: Convergence rates of the three considered Bayesian Optimisation strategies for the Powell functions.

Powell 4D and 10^3 for the Powell 12D). Similarly, the traditional method provides a bad approximation of the global optimum of the studied functions. For instance, the level of error is equivalent with the Powell 12D, but is lower with an absolute gap of 10^1 with the Powell 4D. Finally, the MIBO-P clearly outperforms the two other methods by almost providing the global optimum of the considered functions: the gap is equal to 10^{-1} for both functions.

With regard to the impact of the number of initial points per dimension, the effect is almost inexistant for the traditional EI and MIBO-P. Conversely, for the Powell 4D, the convergence of the MIBO-I algorithm is slightly faster with a higher amount of initial samples.

Fig. 5.15 shows the results of the Ackley 10D. The global optimum of this function is 0.



(A) Ackley 10D - 4 initial points per dimension

(B) Ackley 10D - 5 initial points per dimension

(C) Ackley 10D - 6 initial points per dimension

FIGURE 5.15: Convergence rates of the three considered Bayesian Optimisation strategies for the Ackley function.

For this function, the results are of the same nature as the Powell 4D and 12D. The MIBO-I and the EI do not perform well: both find an optimum associated with a gap of 10^1. Conversely, the MIBO-P provides the global optimum of the Ackley 10D function, since the associated gap is around 10^{-1}.

5.4.2.4 Discussions

The suggested strategies show different levels of performance depending on the nature of the studied functions. The function n°2 is highly multimodal (7 local minima + 1 global minimum) with a highly attractive minimum – this minimum forms a large bowl, similar to a two-dimensional quadratic function. The function n°3 has a low number of local minima, but the global minimum is located on the extremities of the design space after a small bump. Consequently, this minimum is difficult to reach. On the contrary, the function n°1 owns a high number of local minima but without highly attractive ones. For the multidimensional functions, the number of minima cannot be estimated, but the design space is large.

As shown by the previous results, on one hand, the MIBO-I is efficient on the functions n°2 and 3. On the other hand, the MIBO-P is not very efficient for these functions. It shows that the MIBO-I can easily escape from local optima, while the MIBO-P has a higher propensity to be trapped in local

optima (but lesser than the propensity of the EI). Hence, the MIBO-I tend to prioritise the exploration over the exploitation.

With regard to the other functions (Powell 4D and 12D, Ackley 10D), the MIBO-P is highly efficient, while the MIBO-I does not provide good solutions. In this case, the exploration property of MIBO-I is problematic since this method can over-explore the design space, which is larger and larger with the increase of dimensions. On the contrary, the MIBO-P allows really nice patterns of convergence, highlighting a propensity of prioritising the exploitation over the exploration.

These observations can be linked with the way both strategies are constructed. Indeed, the MIBO-I generates a trade-off sample (the one determined with EI) and a full exploratory sample (the one determined with Variance). Thus, this MIBO-I has a higher probability to prioritise the exploration over the exploitation. Conversely, the MIBO-P provides new samples that are on the Pareto front of both acquisition functions. Hence, depending on the location on the Pareto front, the method can generate samples that are more likely to prioritise the exploitation over the exploration.

5.5 Chapter outcomes

The purpose of this chapter was to **address an issue of the most traditionally used acquisition function for Bayesian Optimisation: the Expected Improvement**. As this infill criterion represents a good balance between exploration and exploitation, it might, under some circumstances, focus on local minima with bowl-shape areas. These minima have a high attractor potential and generally enforces the BO process to remain in it until a full exploration of this area. Then, the algorithm can explore other areas and aim for better solutions. Hence, multiple unnecessary calls to the objective functions are done, resulting in longer convergence. When the simulation budget is small, these calls can cripple the process efficiency and induce a minimum that is not the most interesting one.

The major proposal of this chapter is to **propose a re-statement of the Bayesian Optimisation algorithm from a mono-objective to a multi-objective process**. First, the consequence of the localisation of the EI into local minima is highlighted: showing that the exploration term of the EI does not succeed in making the algorithm escape from the local minimum. Then, a new strategy, the Multi-Infill Bayesian Optimisation, is suggested. This new method consists in a reformulation of the traditional mono-objective aspect of the BO algorithm into a multi-objective problem. In a nutshell, the idea is to construct a Pareto front of exploration and exploitation. Then, two new candidates are determined, instead of only one, allowing the injection of a denser information into the process.

The **first version of the suggested algorithm (MIBO-I) is the simplest one and corresponds to the maximisation of two acquisition functions** (the extremities of the Pareto front). Conversely, the **second version of the new strategy (MIBO-P) takes advantage of the diversity of candidates provided by the Pareto front**. The new candidates are generated

with a clustering algorithm, the K-means method. Both methods allow the generation of samples with different properties. For instance, the **MIBO-I allows a generation of one full exploratory sample (determined by maximising the Variance) and a trade-off sample (determined by maximising the EI)**. On the contrary, the **MIBO-P produces two trade-off samples between exploration and exploitation.**

The first experiment considers the optimisation of a simple one-dimensional problem. The results showed that the **two versions of MIBO produce better results than the traditional BO process**. In addition, the MIBO algorithm exhibited a lesser propensity to be stuck in a local minimum than the traditional strategy.

The second experiment involves a benchmark of mathematical functions to assess the capabilities of the suggested strategy for the optimisation of functions with different properties. The results showed that the **two versions of MIBO exhibit different levels of performance depending on the nature of the considered functions**. For instance, the **MIBO-I performs better with functions with large bowl-shape attractors that trap the BO process**. Conversely, when the design space is large, the MIBO-I tends to explore all the design space, making it useless to determine the global optimum of the function, especially when the explored areas have no interest for the EI. With regard to the **MIBO-P, since both added samples are a trade-off between exploration and exploitation, it produces excellent performance with large design spaces**, while generating poor results with functions that have large bowl-shape attractors.

Consequently, this chapter proposed a strategy that can perform better than the traditional BO process. Unfortunately, **an *a priori* knowledge is necessary to choose the most suitable method**. Thus, it would be interesting to pursue this research to explore other clustering methods to generate other types of candidates.

Chapter 6

Fuzzy modelling for uncertain friction-induced-vibration problems

Contents

6.1 Introduction

As highlighted in Chapter 1, the integration of the uncertainties in the FIV simulations is an emerging promising field of research to numerically predict the propensity of instabilities. Nowadays, the current methods rely either on probabilistic strategies or ensemblist approaches. The probabilistic methods allow a precise definition of the output uncertainties, but need a large amount of information, obtained through numerous calls to the solver. Even if some strategies as polynomial chaos can reduce the computational cost, these methods are not viable when the solver is highly expensive. From the ensemblist approaches, the interval analysis is more affordable since only interval problem is solved. Conversely, the fuzzy set theory provides more information (several interval problems are solved), but is less cheap.

The computation of the output membership function, which is a tool similar to the probability density function, can be complex. An early attempt to compute this function was fuzzy arithmetics [211]. The intuition is to redefine the traditional algebraic operators in a fuzzy form. Nonetheless, this approach leads to the overestimation of the output membership functions, as pointed out in [6]. Klir et al. [212] suggested a contrainted formalism to avoid the overestimation issue. More recently, Ngan [211] proposed a new approach to fuzzy arithmetics. Nonetheless, this type of methods is not really suited for engineering problems as shown by Buckley et al.[213].

The combinatorial methods [6] are a reference in the literature to compute the output fuzzy sets. Indeed, they provide an automatic way to sample the different α-cuts of the output membership function. Nonetheless, they fail to perform efficient sampling when the function is nonlinear [214] and to handle high dimensionality.

Other authors hence suggested using perturbation techniques and optimisation to determine the output fuzzy set. The quantity of interest is expended with a Taylor's expansion series [215] or with Padé approximants [216]. Then, a sensitivity analysis is performed on the first derivative of the expansion to determine the monotonicity of the considered quantity. If the function is monotonous, then a local optimiser is used to determine the value of the optimum. Conversely, if the function is not monotonous, the optimum is not determined. Finally, when the function is expensive, it can be replaced by the expansion series to speed up the evaluation of the output fuzzy set. The main limitation of these strategies is that the estimated expansion series is valid only if the perturbation is small. In addition, when the functions are highly nonlinear and multimodal, the approaches suggested by the authors lack in efficiency.

To reduce the computational cost generated by multiple calls to the considered function when determining the min-max bounds of each level of the fuzzy set, some authors used surrogate modelling. Akpan et al. [217] approximated the simulator with a response surface model and carry out optimisation to determine the min-max of the α-cut to, finally, reconstruct the output fuzzy set membership. Recently, a similar approach was suggested by Lu et al. [218] to determine the output fuzzy set of

the eigenvalues of a brake system. As studied in Chapter 3 and Chapter 5, the training set on which the surrogate model is trained needs to be the most informative as possible.

The purpose of this chapter is to propose a new combinatorial strategy which hinges on surrogate modelling to reduce the computational cost and BO to produce efficient adaptative sampling of the considered nonlinear functions.

To assess for the capabilities of the suggested algorithm, this chapter is organised as follows. Section 6.2 reviews the fuzzy set theory and presents the traditional combinatorial strategies available in the literature. Section 6.3 describes the proposed strategy by introducing the sampling strategy and the algorithm associated with the new strategy. Then, Section 6.4 assesses the performance of the suggested strategy. This method is compared with the literature strategies and the hyperparameter setting is thoroughly studied. Finally, Section 6.5 highlights the performance of the given strategy to estimate the coalescence graph of the eigenvalues of an FIV problem.

6.2 Fuzzy set theory

6.2.1 Some definitions

Fuzzy number

The fuzzy set theory, proposed by Zadeh [77–79], is a generalisation of the classical crisp set theory. Both types of set correspond to a collection of elements which fulfills a condition $\mathcal{A}$. The only difference between those two types of set lies on the membership function $\mu_A(x)$, which corresponds to the mathematical expression of the completion of $\mathcal{A}$. On one hand, the fuzzy membership function is defined on the continuous set $[0;1]$, while, on the other hand, the crisp membership function is defined on the discrete set $\{0;1\}$.

Additional properties are usually considered for the definition of fuzzy sets. The fuzzy set has to be convex and only one element is associated with a membership value of 1. Finally, the fuzzy membership function has to be at least piecewise monotonous, in the inclusion sense [219]. Given these properties, the fuzzy set $\tilde{A}$ is called a fuzzy number and is defined by (Eq. 6.1).

$$\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) \mid x \in \mathbf{R}\} \tag{6.1}$$

where, $\mu_{\tilde{A}}(x) : \mathbf{R} \longrightarrow [0;1]$.

This modification of the definition of the membership function (from the crisp to the fuzzy) allows the modelling of a wider range of applications. The vagueness of a material or geometrical parameter can then be described using a fuzzy set: a value of 1 would traduce an absolute confidence on the parameter value, while a value of 0.5 would traduce a mid-confidence on the parameter value.

The generalisation to n-dimensions is quite simple since the fuzzy number is not defined over $\mathbb{R}$ but over $\mathbb{R}^n$. In practice, a membership function is defined for each dimension.

Choice of the membership function

Several choices are available in the literature as, for instance, Triangular, Gaussian, and Quadratic fuzzy numbers [6]. Nonetheless, this choice is highly arguable in the literature. Indeed, some authors [83, 218, 220] prefer to keep the fuzzy representation as simple as possible. Other authors [221–224] suggested enhancing the fuzzy representation with more complex fuzzy membership function, by using either convex or non-convex functions.

The first approach owns the advantage of simplicity, but may be under predictive depending on the complexity of the phenomenon. If the convexity is ensured, the second approach only needs sufficient prior knowledge to describe the membership function. Otherwise, other methods to solve the non-convex problem are needed (see aforementioned references about non-convexity of the fuzzy sets).

Nonetheless, De Lima et al. [225, 226] showed that the fuzzy methodology with simple membership functions is a good first approximation to quantify the uncertainty of a numerical model by comparing fuzzy sets to several approaches (Monte Carlo and stochastic FEM). Hammell et al. [227] concluded that triangular fuzzy numbers induce smoother results than trapezoidal fuzzy sets. Similarly, Adil et al. [228] emphasised that triangular and trapezoidal fuzzy sets give better results than gaussian fuzzy sets.

Hence, to ease the definition of the fuzzy numbers, each fuzzy number of this research is taken to be triangular. In addition, the input $\tilde{P}$ is always normalised between 0 and 1 to work at the same variation scale between the model parameters. The form taken into account in the following applications is given by (Eq. 6.2) and shown in Fig. 6.1.

$$\mu_{\tilde{p}}(x) = \begin{cases} x/\bar{x} & \text{for} \quad \bar{x} \in]0;1[,\ x \in [0;\bar{x}[\\ (x-1)/(\bar{x}-1) & \text{for} \quad \bar{x} \in]0;1[,\ x \in]\bar{x};1] \\ 1 & \text{for} \quad \bar{x} =]0;1[,\ x = \bar{x} \\ [0;1] & \text{for} \quad \bar{x} = \{0;1\},\ x = \bar{x} \end{cases} \tag{6.2}$$

where $\bar{x}$ is the modal crisp value of the fuzzy set $\tilde{P}$.

The Zadeh's Extension Principle (ZEP) was first introduced in [229] as a way to compute the solution of an input fuzzy set $\tilde{A}$ through a functional f. This principle is given in its continuous form by (Def. 6.2.1).

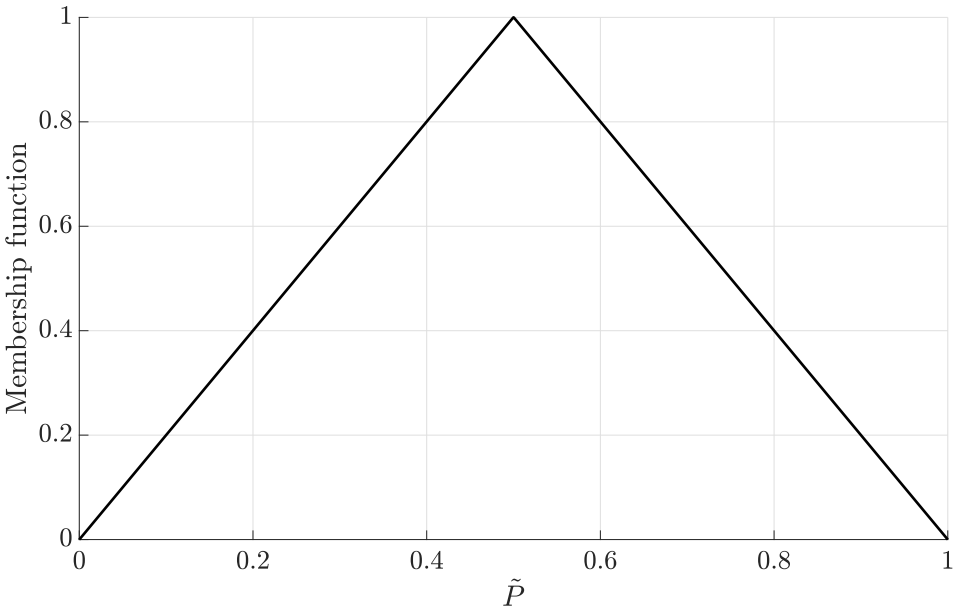


FIGURE 6.1: A triangular membership function ($\bar{x} = 0.5$).

Definition 6.2.1

Let f be a functional mapping from a set $\mathbf{X}$ to a set $\mathbf{Y}$, $\tilde{A}$, a fuzzy set defined by the membership function $\mu_{\tilde{A}}$ on $\mathbf{X}$, $\tilde{B}$, a fuzzy set defined by the membership function $\mu_{\tilde{B}}$ on $\mathbf{Y}$, x, an element of $\mathbf{X}$ and y, an element of $\mathbf{Y}$.

The membership function $\mu_{\tilde{B}}(y)$ is given by:

$$\mu_{\tilde{B}}(y) = \begin{cases} \sup_{y=f(x)} \min\left\{\mu_{\tilde{A}}(x)\right\} & \text{if} \quad f^{-1}(y) \neq \varnothing \\ 0 & \text{if} \quad f^{-1}(y) = \varnothing \end{cases} \tag{6.3}$$

If $\mathcal{X}$ is the cartesian product of the sets $\mathcal{X}_1$ to $\mathcal{X}_d$, with d, the number of dimension, then (Def. 6.2.1) becomes (Def. 6.2.2).

Definition 6.2.2

Let f be a functional mapping from a set $\mathbf{X}$ to a set $\mathbf{Y}$, $\tilde{A}$, a fuzzy set defined by the membership function $\mu_{\tilde{A}}$ on $\mathbf{X}$, $\tilde{B}$, a fuzzy set defined by the membership function $\mu_{\tilde{B}}$ on $\mathbf{Y}$, x, an element of $\mathbf{X}$ and y, an element of $\mathbf{Y}$.

The membership function $\mu_{\tilde{B}}(y)$ is given by:

$$\mu_{\tilde{B}}(y) = \begin{cases} \sup_{y=f(x)} \min\left\{\mu_{\tilde{A}}(x_1), \cdots, \mu_{\tilde{A}}(x_d)\right\} & \text{if} \quad f^{-1}(y) \neq \varnothing \\ 0 & \text{if} \quad f^{-1}(y) = \varnothing \end{cases} \tag{6.4}$$

In practice, the evaluation of this principle is not possible. As shown by Hanss [6], the use of the ZEP for a simple addition between two fuzzy numbers leads to the enumeration of all the possible

combinations to obtain a given real (which is obviously infinite). Hence, the input fuzzy set $\tilde{A}$ and output fuzzy set $\tilde{B}$ have to be discretised.

The most used discretisation scheme is the discretisation along μ-axis. The idea is to create m cuts of the membership function, called α-cut, and to iteratively solve each α-cut problem as an interval problem (Fig. 6.2). This strategy allows maintaining the independence between the discretisation $\Delta\mu$ and the parameter evolution. In addition, $\delta\mu$ is also invariant to arithmetical operations, meaning that $(\forall i \in [\![1, n]\!],\ (\text{cut}_i)_{\text{input}} = (\text{cut}_i)_{\text{output}})$.

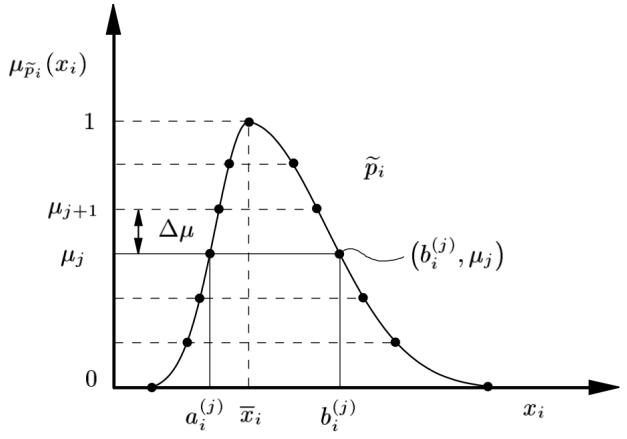


FIGURE 6.2: Discretisation along μ-axis [6]

.

Other methods of discretisation exist as shown in [6], but these methods necessit the use of fuzzy algebra, which is not suited for engineering problems.

6.2.2 Combinatorial methods and limitations

6.2.2.1 Presentation of the transformation methods

Principle

For all the following methods, the fuzzy vector $\tilde{\mathbf{A}}$ is considered, where each element of $\tilde{\mathbf{A}}$ is an independent fuzzy number $\tilde{A}_i$, associated with the membership function $\mu_{\tilde{A}_i}$.

Each element $\tilde{A}_i$ is then decomposed along the μ-axis and described by (Eq. 6.5), where X_i^j is a crisp interval and m is the number of α-cuts.

$$\tilde{P}_i = \{X_i^0, X_i^1, \ldots, X_i^m\} \tag{6.5}$$

where $\tilde{P}_i$ is the decomposed version of $\tilde{A}_i$, $X_i^j = \text{cut}_j(\tilde{A}_i)$, for $i \in [\![1, m]\!]$.

Reduced Transformation Method

The Vertex method, also called the RTM, was first proposed by [214] to reduce numerical costs of computing the ZEP. Note that this strategy was later formulated as a special case of the GTM [80].

All combinations between the minimum and maximum of each parameter α-cut is enumerated and evaluated. The number of calls of the considered function f is $n_T = 1 + (m-1) \times 2^n$, where n is the number of fuzzy parameters and m, the number of α-cuts.

Fig. 6.3 shows the distribution of the different combinations in a grid for two input fuzzy sets. Each black dot corresponds to a combination to evaluate. Here, 17 samples have to be computed ($m = 5$ and $n = 2$).

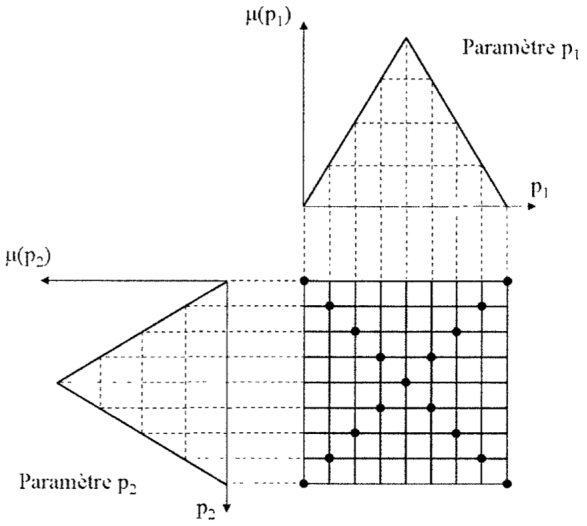


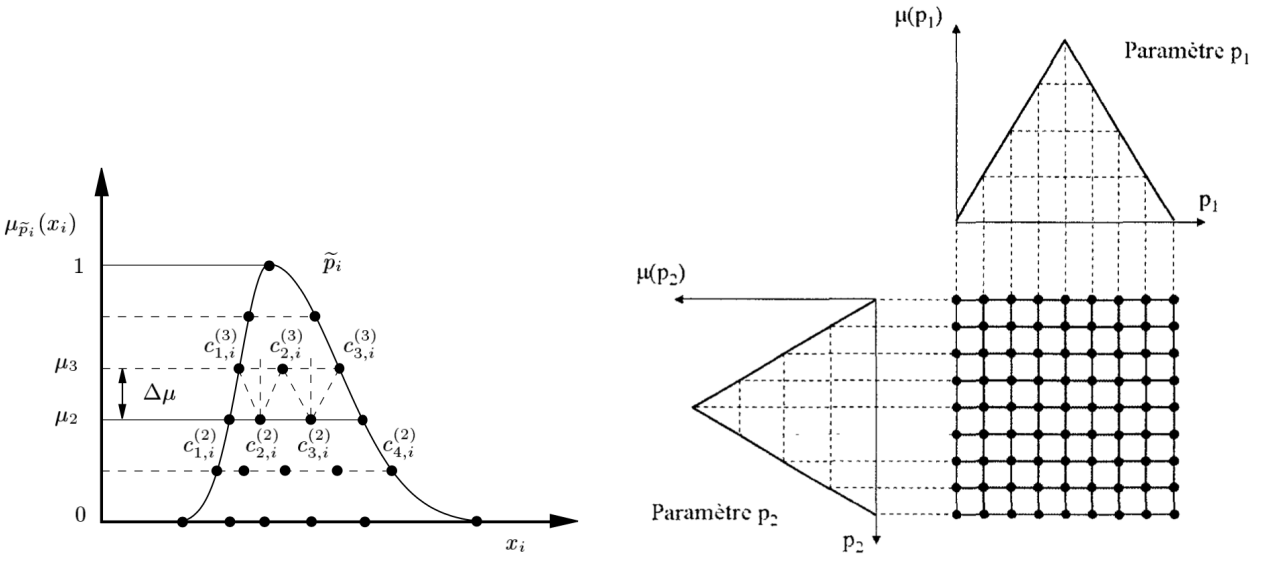
FIGURE 6.3: Localisation of the combination samples for the reduced transformation method for two input fuzzy sets [230].

.

For monotonous functions (where the extrema are located on the vertices of the input fuzzy set), Dong et al. [214] showed that the Vertex method determines the exact output fuzzy set $\tilde{Q}$. Nonetheless, other authors [231, 232] highlighted that, when the studied functions are non-monotonous, the Vertex method cannot be considered an efficient strategy to determine the bounds of $\tilde{Q}$. Hence, for complex real-life applications, better strategies have to be considered.

Generalised Transformation Method

To overcome the drawback of Vertex method, Hanss [80] suggests enhancing the array $\hat{X}_i^j$ with additional samples. These samples are constructed as shown in Fig. 6.4(A). For instance, the sample $c_{2,i}^{(2)}$ is built as the average of $c_{1,i}^{(3)}$ and $c_{2,i}^{(3)}$, and so on. Another feature of the GTM is that the number of additional samples decreases as the α-cut increases: the additional samples number is $m-2-i$, where i the α-cut level. Fig. 6.4(B) shows the corresponding mapping for a two input fuzzy sets.



(A) Example for one input fuzzy set [6] (B) Example for two input fuzzy sets [230]

FIGURE 6.4: Localisation of the combination samples for the generalised transformation method.

Nonetheless, this better mapping of the design space can quickly become a burden. Indeed, the number of calls of the considered function f is $n_T = \sum_{i=1}^{m} i^n$, where n is the number of fuzzy parameters and m, the number of α-cuts. For instance, for a 5-cuts one-dimensional fuzzy set, this method needs 15 samples. For a 5-cuts five-dimensional fuzzy set, 4,425 samples must be computed (for a comparison, only 129 configurations are needed with the reduced transformation method).

Other transformation methods

To reduce this computational burden, but still benefit from the better sampling of the input fuzzy sets, Hanss [233] suggested discriminating the fuzzy parameters depending on their impact over the studied functions. These parameters are classified either as generalised-parameters or reduced-parameters. Then, the Extended Transformation Method (or ETM) is used to generate the different samples combinations.

This method works as a compromise between the reduced transformation and the generalised transformation method. It benefits from the advantages of both methods and tries to reduce the impact of the drawbacks of each method. The number of calls of the considered function f is given by $n_T = 1 + \sum_{i=2}^{m} 2^{p_r} i^{p_g}$, where p_r stands for a reduced-parameter and p_g, for a generalised-parameter and $n = p_r + p_g$.

Finally, the Short Transformation Method (or STM) was suggested by [234] to take advantage of the monotonic behaviour of eigenvalue shifts in modal analysis and harmonic excitation. The idea is to determine the most influential diagonal in the parameter space. This detection is done by carrying out a sensitivity analysis over an initial set of samples (computed at α=0). The number of call of the considered function is $n_T = 1 + 2^n + 2 \times (m - 2)$.

6.2.2.2 Limitations of these methods

The limitations of these transformation methods are two-folds. First, these methods do not cope well with multi-dimensional problems.

Table 6.1 presents an evaluation of the number of samples n_T generated using the techniques presented above. For each method, the number of α-cuts is 5. The ETM is evaluated for two configurations: the first one is obtained considering 2/3 of reduced-parameters and 1/3 of generalised-parameters; the second one with 1/3 of reduced-parameters and 2/3 of generalised-parameters. In industry, the computation budget is roughly about 200 to 300 evaluations of f, depending on the specificities of the project. The green values correspond to an acceptable cost and the red values, an unacceptable cost.

Table 6.1
Comparison of the number of samples needed for each technique (5-cuts fuzzy set)

Technique \ Parameter		1	3	6	12
STM	[234]	9	15	71	4,103
RTM	[80]	9	33	257	16,385
GTM	[80]	15	225	20,515	261,453,379
ETM (Config. 1)	[233]	\	61	881	250,625
ETM (Config. 2)	[233]	\	111	3,917	7,407,665

.

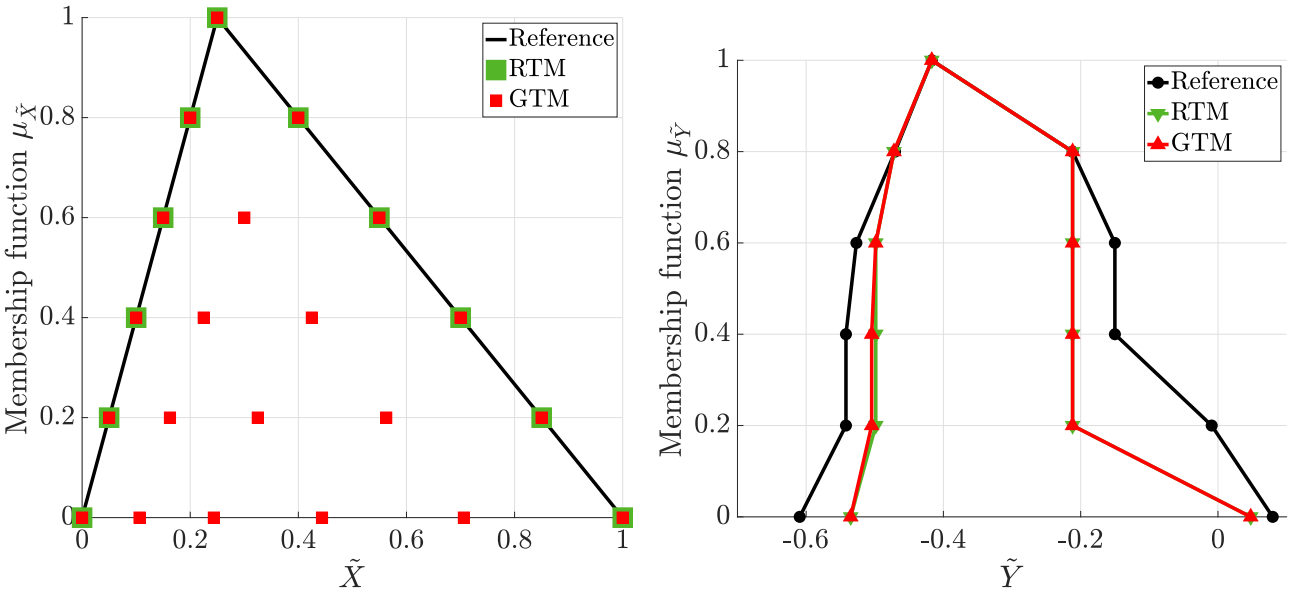
Only the short and reduced transformation methods are still usable in an industrial context for 6 fuzzy parameters, but when the number of fuzzy parameters is increased to 12, no techniques may be used. Besides, the GTM even generate unrealistic evaluation procedure with more than 260 millions samples.

Second, these methods do not deal well with highly nonlinear functions. The illustration of this statement is done with the Xiong's function. The input parameter is represented by a triangular fuzzy set (Eq. 6.2), with a modal value $\bar{x}$ of 0.25 and a membership discretisation m of 6. The output fuzzy set $\tilde{Y}$ is evaluated with the RTM and the GTM. The first strategy generates 11 samples while the latter induces 21 samples.

Observations

Fig. 6.5(A) shows an asymmetrical triangular set $\tilde{X}$, with a modal value $\bar{x}$ of 0.25. The reference is given in black, the samples associated with RTM in dark green squares and those of GTM in red squares. Fig. 6.5(B) exhibits the output fuzzy set $\tilde{Y}$, using both techniques.

Only the first two α-cuts are well approximated with both methods, since no significant difference is observed; the maximum error is about 0.32% (for the estimation of the lower bound associated with $\mu_{\tilde{Y}}(x) = 0.8$, $\forall x \in [\![0.2; 0.4]\!]$).



(A) Input fuzzy set

(B) Output fuzzy set

FIGURE 6.5: Approximation of the output fuzzy set using the Reduced and Generalised Transformation Method for a non-symmetric input fuzzy set.

For the other α-cuts, the results are quite disappointing. Large errors are observed, especially for the upper bound estimation, which lead to an underestimation of the output fuzzy set membership function. For instance, the maximum error is about 2255% (for the estimation of the upper bound associated with $\mu_{\tilde{Y}}(x) = 0.2, \forall x \in [\![0.05; 0.85]\!]$)[1].

Discussions

For the example presented in Fig. 6.5(B), both methods have largely presented suboptimal performance. This is mainly a consequence of the high non-linearity nature of the problem to solve.

Considering the input fuzzy parameter $\tilde{X}$, defined with a modal value $\bar{x}$ of 0.25 and an α-cut associated with $\mu_{\tilde{X}}(x) = 0.2$ ($x \in [\![0.05; 0.85]\!]$).

Fig. 6.6 compares the parametric space associated with an α-cut$_1$ with the parametric space associated with an α-cut$_2$. The samples created with the RTM and the GTM are respectively given in green and red squares. The extrema of the current α-cut are reminded in yellow stars.

As pointed out earlier, this problem is highly non-linear. Indeed, the variations of the left-side parametric space contain both extrema of the current α-cut whereas the right-side parametric space is almost flat.

Here, these sampling strategies are clearly not suited for this kind of problem. As the RTM evaluates the bounds of the parametric space, the GTM generates computations in the next α-cut, without creating samples in the α-cut$_1$, which contains one of the extrema. In addition, the samples

[1]The error is high because of the true value of the upper bound, which is 0.009, making the error value hazardous

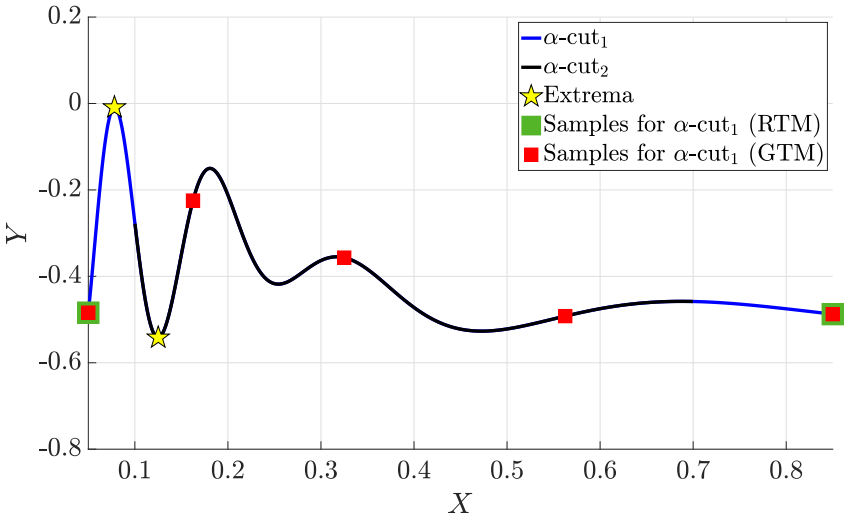


FIGURE 6.6: Xiong function for the α-cut associated with $\mu_{\tilde{X}}(x) = 0.2$.

generated in the α-cut$_2$ are of no use at that moment of the algorithm, since the algorithm will generate samples in the next α-cuts.

Then, to limit the computational cost, all the samples have to be included in the new parametric space. Nonetheless, even if the GTM is restricted to these regions of the design space, it is still unusable with non-linear problems. Indeed, as shown in Fig. 6.6, the left and right sides exhibit different behaviours (resp. quadratic and almost linear). The static procedures, as RTM or GTM, cannot adapt to such non-stationary evolution. Thus, it is not possible to rely on these methods, especially with non-linear functions.

6.3 Proposed method

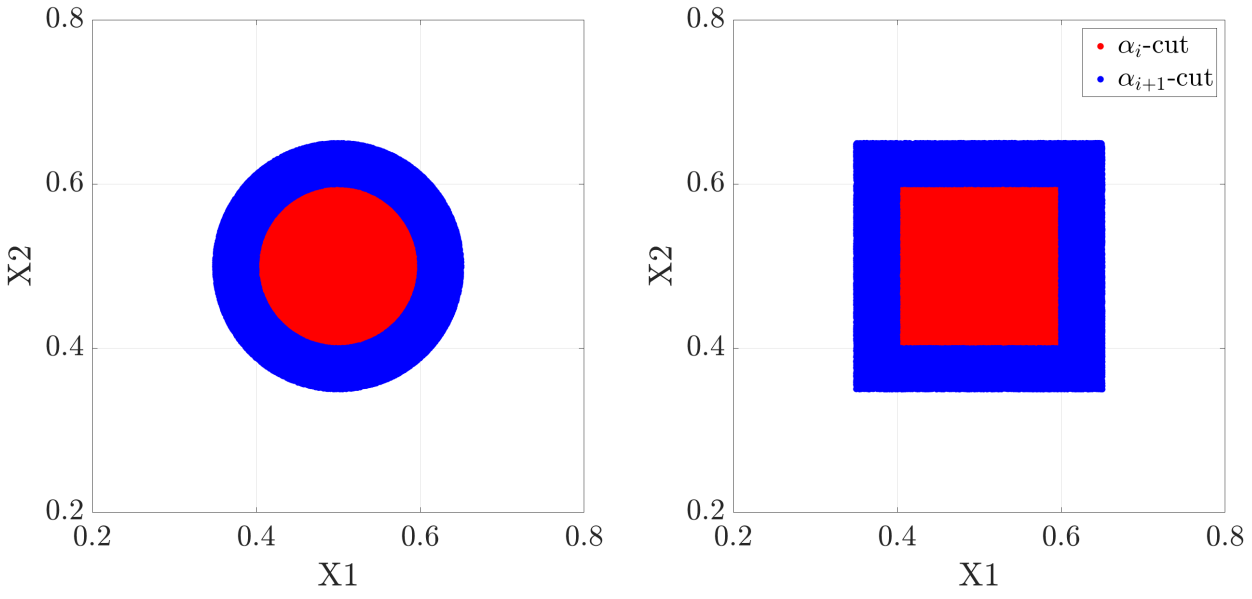
6.3.1 Snapshot generation

As highlighted in the previous section, at a given α-cut$_i$, the combinatorial method can provide samples in α-cut$_i + 1$. Thus, for obvious reasons of computational efficiency, the snapshots generated by the BO algorithm, presented in the following section, are sampled in the given α-cut$_i$. The algorithm is not allowed sampling in the other α-cuts.

To do so, a specific random sampling is, here, proposed by considering the following homeomorphism. To map a circle $C = \{(x, y) \in \mathbb{R}^2 | x^2 + y^2\}$ to a square $S = \{(x, y) \in \mathbb{R}^2 || x| \leq 1, |y| \leq 1\}$, the transformation is given by (Eq. 6.6).

$$\begin{aligned} (x, y) &\mapsto \frac{\sqrt{x^2 + y^2}}{\max(|x|, |y|)}(x, y), \quad \forall (x, y) \in \mathbb{R}^2 \backslash \{(0, 0)\} \\ (0, 0) &\mapsto (0, 0) \end{aligned} \tag{6.6}$$

This transformation can easily be generalised for $\mathbb{R}^n$ and is here used, for each α-cut, to build the training set, shown in (Fig. 6.7(B)), compatible with the suggested BO algorithm, from an initial random training set (Fig. 6.7(A)). Fig. 6.7 illustrates the proposed implementation for two successive α-cuts and 20,000 samples per cut.



(A) Traditional random sampling

(B) Transformed random sampling

FIGURE 6.7: Transformation of the traditional random sampling with the considered homeomorphism.

Those transformations allow preventing the Bayesian strategy from sampling in another α-cut than the one which is currently dealing with.

6.3.2 Bayesian algorithm

Two limitations of the current combinatorial methods are pointed out, namely the computational cost and the numerical efficiency, while dealing with nonlinear functions. The proposal of this chapter is to benefit from the surrogate modelling strategy (reviewed in Chapter 3) and the Bayesian Optimisation strategy (reviewed in Chapter 5). The first method allows a reduction of the computation cost, while the second allows an adaptability of the sample strategy.

The new algorithm, called FUZZy Bayesian Optimisation (denoted FuzzBO), is presented in the flowchart (Fig. 6.8). First, an essential modification of the fuzzy paradigm is proposed here. The traditional combinatorial strategies compute a given training set and then associate the evaluations to their corresponding α-cut. Conversely, the suggested approach is an iterative process that evolves from the top ($\mu_{\tilde{X}}(\bar{x}) = 1$) to the bottom ($\mu_{\tilde{X}}(x) = 0$).

At first, the core element ($\mu_{\tilde{X}}(\bar{x}) = 1$) is evaluated. Then, the algorithm goes to the first α-cut interval. A modified random sampling (with the transformation described in Section 6.3.1) is

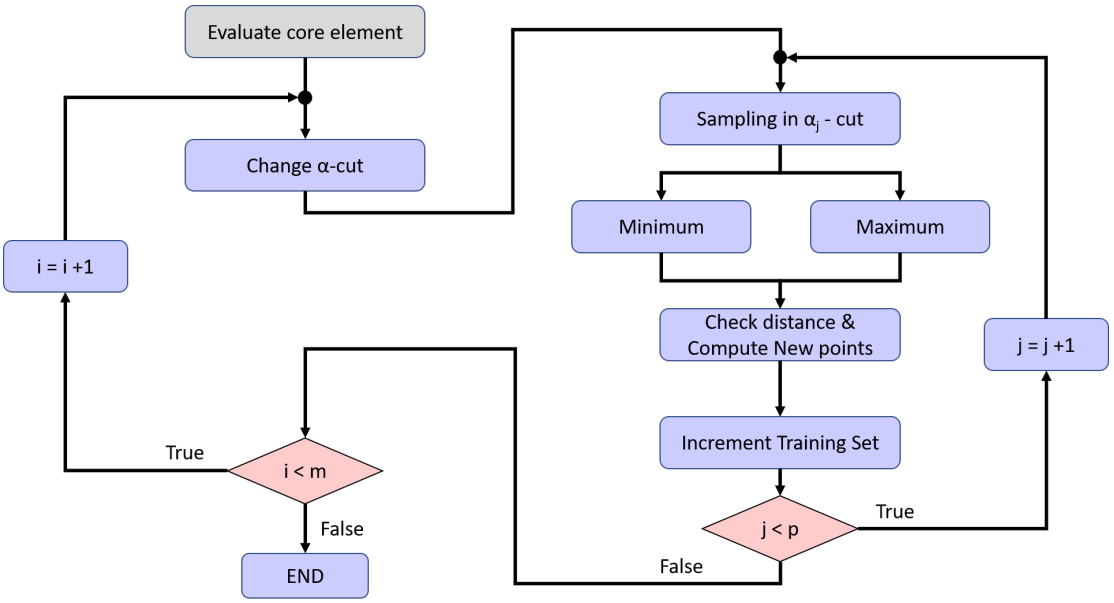


FIGURE 6.8: FuzzBO workflow.

evaluated to acquire knowledge about the behaviour of the solver in this cut. This training set is, then, used to construct the GP to approximate the behaviour of the considered function for all the design space.

After this step, the evaluation of the bounds of the current α-cut is performed. Two independent optimisation problems are solved: one for estimating the optimal sample with regard to the minimisation of the solver; one with regard to the maximisation of the solver. These optimisation problems are solved by maximising the Expected Improvement, given for either the minimisation (Eq. 6.7) or the maximisation (Eq. 6.8) of the studied function. The solution of those two optimisation problems corresponds to the sample which has the highest probability to be the optimum of the considered solver.

$$\mathrm{EI}_{min}(x) = \alpha_{min}\left(\frac{1}{2} + \frac{1}{2}\mathrm{erf}\left(\frac{\alpha_{min}}{\sqrt{2}\sigma}\right)\right) + \frac{\sigma}{\sqrt{2\pi}}\exp\left(-\frac{\alpha_{min}^2}{2\sigma^2}\right) \tag{6.7}$$

$$\mathrm{EI}_{max}(x) = \alpha_{max}\left(-\frac{1}{2} + \frac{1}{2}\mathrm{erf}\left(\frac{\alpha_{max}}{\sqrt{2}\sigma}\right)\right) + \frac{\sigma}{\sqrt{2\pi}}\exp\left(-\frac{\alpha_{max}^2}{2\sigma^2}\right) \tag{6.8}$$

where $\alpha_{min} = y_{min} - \widehat{f}$, $\alpha_{max} = y_{max} - \widehat{f}$, y_{min} and y_{max} are the extrema of the training set and $\widehat{f}$, the prediction of the GP.

Before incrementing the training set, the distance between the two new candidates and the current training has to be checked. This problem is responsible for causing the inversion of Cholesky decomposition of the GP covariance matrix to fail. The chosen approach is to limit the proximity of the samples by computing the Euclidean distance and ensuring that it is under a threshold s_{Fuzzbo}.

Many tests showed that a practical value is $5 \cdot 10^{-3}$. The new samples can, then, be added to the training set to update the GP approximation. A new evaluation of the bounds is performed.

When the maximum number of iterations p is reached, the algorithm goes to the next α-cut and a new initial training set is computed to acquire knowledge about this cut. The training set of the previous cut is kept and merged with this new training set.

The number of iterations p is constant for each α-cut. The final number of samples, obtained after the execution of the FuzzBO algorithm, is typically $n_T = m \times (q + 2 \times p)$ but, due to the proximity issue, the previous formula is only an upper bound of the total number of samples.

The Bayesian algorithm is set for p iterations and, when the number of added samples has been met, the α-cut is modified and the iterative process begins again for the next α-cut. Table 6.2 summarises all the hyperparameters of the algorithm and highlights their main action. The first hyperparameter manage the fuzzy parameters, while the two others control the Bayesian Optimisation; highlighting the two main components of the suggested algorithm.

Table 6.2
Hyperparameters of the FuzzBO algorithm.

Parameter type	Symbol	Purpose
α-cut number	m	Control the discretisation of the membership function
Initial training set size	q	Control the initial diversification
Iteration number	p	Control the added samples number

6.4 Approximation of the output fuzzy sets

6.4.1 Assessment procedure

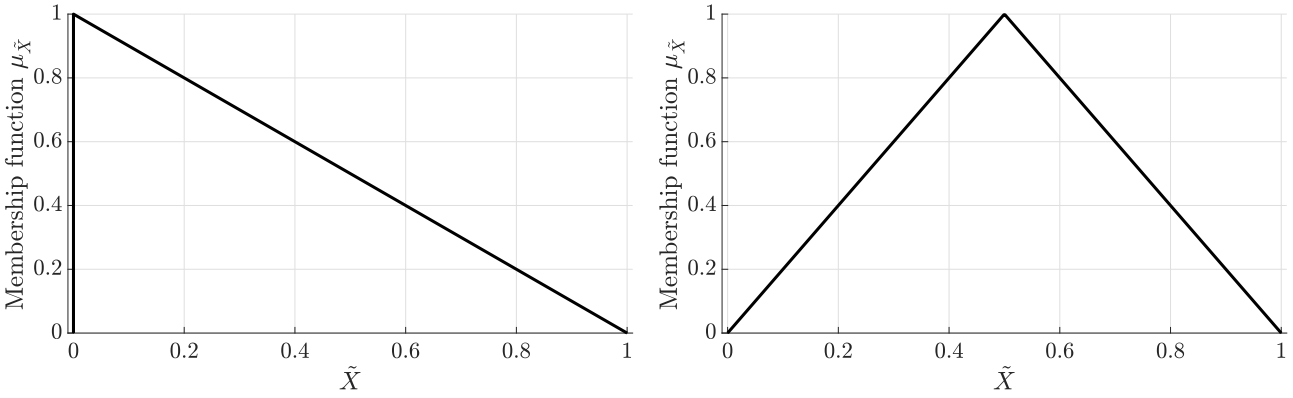
The case study is the Double Hulten, presented in Chapter 3. The purpose of using this phenomenological model is to validate the proposed methodology with a cheap and practical model, which simulates the brake squeal phenomenon. Hence, the benchmark is easier to compute compared to a classical FE model - which may take a longer time for the completion of a simulation.

The parameters, as well as their range of variation (about 10% around the nominal value), are shown in Table 6.3. They are normalised to 0-1 range, and then considered fuzzy inputs with a triangular membership function. The fuzzy friction coefficient is constructed with a modal value $\bar{x}_\mu = 0$ and the fuzzy stiffnesses with a modal value $\bar{x}_k = 0.5$.

Table 6.3
Range of variation of the considered system parameters of the Double Hulten.

	μ	k_{11} ($N \cdot m^{-1}$)	k_{12} ($N \cdot m^{-1}$)	k_{21} ($N \cdot m^{-1}$)	k_{22} ($N \cdot m^{-1}$)	k_a ($N \cdot m^{-1}$)
Range	[0;1]	[2700;3300]	[5400;6600]	[900;1100]	[2700;3300]	[900;1100]

Fig. 6.9 exhibits the form of the two considered types of fuzzy numbers. The reference output fuzzy set for the following applications is determined by computing roughly 100,000 configurations for each α-cut to find the extremums of these cuts.



(A) Friction coefficient fuzzy number

(B) Other parameter fuzzy number

FIGURE 6.9: Forms of the considered fuzzy numbers.

The sections 6.4.2 to 6.4.4 analyse the capabilities and effects of key parameters of FuzzBO over the approximation of the output fuzzy set. First, the suggested algorithm is compared to the state-of-the-art methods, that are the RTM [80] and the surrogate modelling approach [141]. Secondly, the impact of the Bayesian hyperparameter setting is emphasised. Then, following the conclusions about the setting, the effect of the fuzzy discretisation is considered to explicit the behaviour of the algorithm with coarser and refined fuzzy sets. The third complex eigenvalue is chosen because it presents the most complex behaviour of the Double Hulten. Finally, the Section 6.4.5 studies a pairing issue that was identified after execution of the FuzzBO algorithm.

Two metrics are used to assess the capabilities of the FuzzBO algorithm: the support and the area of the fuzzy set, which is the Riemann integral of the membership function (Eq. 6.9).

$$\text{Area} = \int_0^1 \mu_{\tilde{A}}(x)dx \tag{6.9}$$

6.4.2 Influence of the Bayesian formalism

Three methods are here compared to assess the optimal way of generating the samples used to evaluate the output fuzzy set: the deterministic strategy with the RTM, the fully Random strategy with GP approximation and the incremental strategy with the FuzzBO algorithm.

A rather refined discretisation of the membership function is considered, with 7 α-cuts. The RTM needs 385 samples to be computed, about 64 samples per α-cut. The same number of samples is used to generate the LHS on which the GP is trained. For FuzzBO, the initial training set size q is fixed to 30 and 17 iterations are used at each α-cut to solve the optimisation problems. Hence, the number of samples is similar for all methods, *i.e.* 385 samples.

The bounds of each α-cut are determined with the training set for the RTM and the FuzzBO algorithm. For the GP, a prediction over a significant amount of samples is carried out, and the bounds are computed with these predictions.

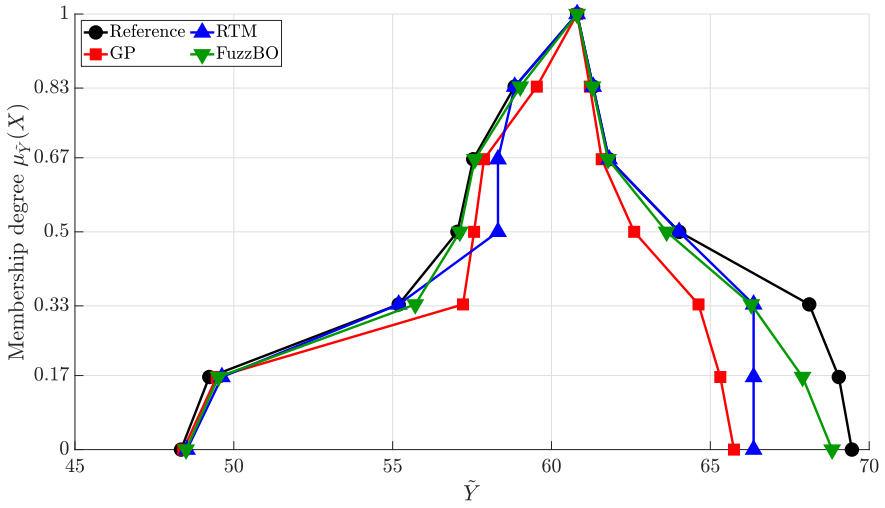


FIGURE 6.10: Comparison of the output fuzzy set approximation using RTM, GP or FuzzBO.

Observations

Fig. 6.10 compares the three considered methods for the estimation of the output fuzzy set. For the RTM and GP, since both strategies do not ensure the convexity of the fuzzy set, the results were modified to keep the convexity property.

For the RTM, the results are quite interesting. On the overall, the output fuzzy set is quite well approximated, more especially for the minimum bounds of the last three α-cuts or the maximum bounds of the first four α-cut. Conversely, the approximation of the maximum bounds of the last three α-cut is moderately erroneous.

For the GP, the results are disappointing. Indeed, except for the approximation of the last two minimum bounds, the approximation is highly inaccurate. Nonetheless, the trend of the output fuzzy set is well approximated since its shape is quite similar from the reference output fuzzy set.

For the FuzzBO algorithm, the results are very interesting. This strategy allows an almost perfect approximation of the minimum bounds and a pretty good estimation of the maximum bounds of the output fuzzy set. In addition, the shape of the approximation is also good, despite the underestimation of the maximum bounds.

Table 6.4 provides the values of the two considered metrics for the three methods. From the different metrics values, it is obvious that FuzzBO outperforms the two other methods, since the estimation error of the support is about 3.61% and the area is roughly 6%.

Table 6.4
Comparison of the area and support approximation with RTM, GP or FuzzBO.

Method	Support		Area	
	Value	Error	Value	Error
Reference	21.098	\	9.435	\
FuzzBO	20.336	-3.61%	8.684	-6.05%
RTM	17.835	-15.47%	8.082	-14.34%
GP	16.814	-20.31%	7.232	-23.35%

Discussions

As emphasised by Fig. 6.10, the considered frequency has a non-linear behaviour since the width of the output fuzzy set greatly varies depending on the α-cut level. This observation can be linked to the propensity of some eigensolutions of a brake system to become unstable. This instability generally arises from a coupling with other eigensolutions, making it more difficult for conventional strategies (RTM or GP) to correctly estimate the output fuzzy set.

Conclusions

These first results are very encouraging with regard to the capabilities of the suggested algorithm for the estimation of the membership function of fuzzy sets. The iterative training sets strongly improve the precision of the estimation and better handle the non-monotonicity of the considered functions. The FuzzBO algorithm is less time-consuming and provides better results than the two other methods, since, for each iteration of the algorithm, a GP and two optimisation of the infill criteria are carried out. Nonetheless, the most time-consuming operation remains the evaluation of the solver, which is more expensive than the execution of the suggested algorithm, especially in engineering applications.

6.4.3 Influence of the Bayesian hyperparameter setting of the FuzzBO algorithm

6.4.3.1 Area approximation

This first test aims at highlighting the effect of the Bayesian hyperparameter setting over the performance of the suggested method. These parameters are the initial training set size q and the iteration number p. The first one controls the initial amount of information available for the algorithm to explore the design space, and the latter handles the amount of samples that the algorithm can add to the training set $\mathcal{T}$.

Here, the purpose is to link the setting of these hyperparameters to the characteristics of the considered problem, *i.e.* the number of α-cut m and the dimensionality of the problem d. Thus, the considered values of both hyperparameters are given as follows: $\{6, 7, 12, 14, 18, 21\}$, corresponding to $[\![1; 3]\!] \times \mathbf{d}$ and $[\![1; 3]\!] \times \mathbf{m}$ values (with $d = 6$ and $m = 7$).

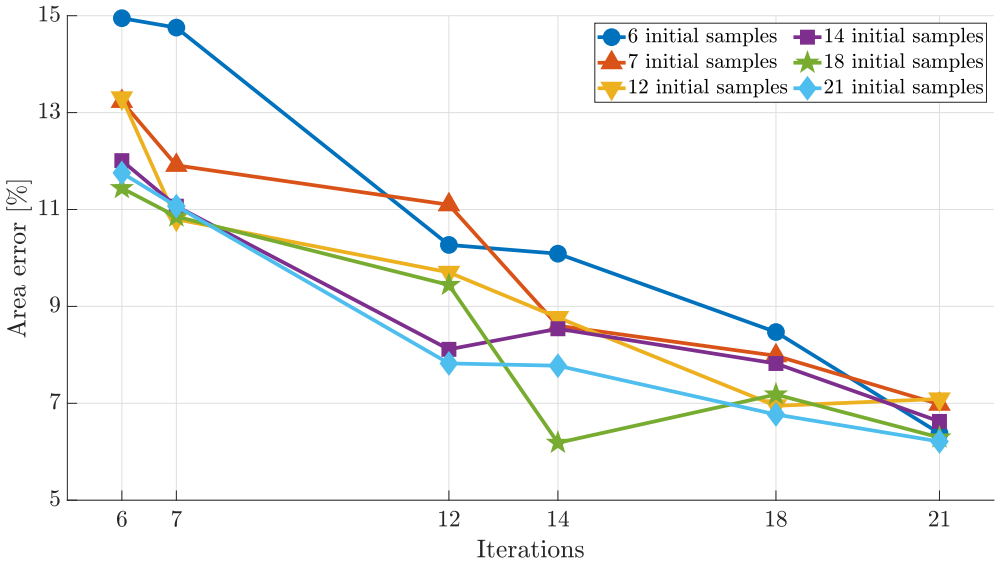


FIGURE 6.11: Evolution of the area error with respect to the Bayesian hyperparameters of FuzzBO.

Observations

Fig. 6.11 shows the evolution of the area error with regard to the iteration number of the Bayesian algorithm and to the number of samples of the initial training set.

First, for all configurations, the increase of the iterations implies a decrease of the area error. Indeed, for 6 iterations, the area error associated with the 6 samples training set is roughly 15% and decreases to 7%. Similarly, the area error associated with the 21 samples training set is about 12% and is reduced to 6%, *i.e.* a reduction of 50% of the area error in all cases.

Secondly, the initial training set size slightly influences the performance of the algorithm, when the number of iterations is high. Indeed, for 21 iterations, the area error spread is 1 percentage point (from 6.2 to 7.3%). From 6 to 14 iterations, the initial training set size has a greater impact, since the area error spread is about 3 to 4 percentage points.

To highlight the difference between the best configuration (21 initial samples and 21 iterations) and the worst configuration (6 initial samples and 6 iterations), the approximation of the output fuzzy set for both methods are drawn in Fig. 6.12. These configurations respectively generate an area error of 6.2% and 14.94%.

For the first four α-cut, the difference between the two approximations are not very important. Conversely, for the last three α-cuts, the difference is moderately important. The approximations are more erroneous and the shape of the output fuzzy set is less well-approximated (the slope is the same for the last three maximum bounds) for the worst configuration. For the best configuration, the slope and the maximum bounds are better approximated.

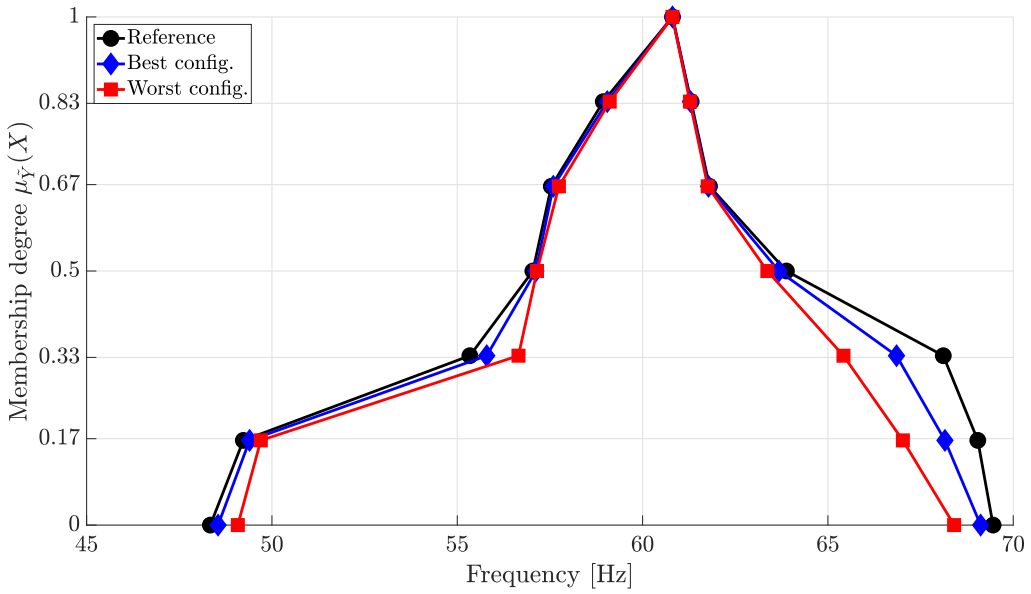


FIGURE 6.12: Comparison between the worst and best configurations for the area approximation (● area is 9.435; ♦ area is 8.849; ■ area is 8.025).

Discussions

The results showed that an increase of the iterations of the FuzzBO algorithm allows a more precise approximation of the output fuzzy set area. The sample number of the initial training set can also help the algorithm to provide better results, and its effect is quite simple: the greater, the better. Nonetheless, the effect of this parameter tends to fade away as the number of iterations increases.

In a way, these results show that the diversification brought by the initial training set has a lower effect on the convergence of the suggested strategy, when the number of iterations is important. As emphasised in Chapter 5, the infill criterion used in FuzzBO encompasses both exploration and exploitation.

6.4.3.2 Support approximation

Observations

Fig. 6.13 shows the evolution of the support error with regard to the iteration number of the Bayesian algorithm and to the number of samples of the initial training set.

With regard to the support, the effects of the iterations are more mixed. For instance, the performance of FuzzBO with 14 initial samples fluctuates around 6%, even with the maximum number of iterations. The other configurations are improved with the increase of the number of iterations, despite some fluctuations. For instance, the 21 initial samples configuration generates a support error of roughly 8%, with 6 iterations, which decreases to 2%, for 21 iterations.

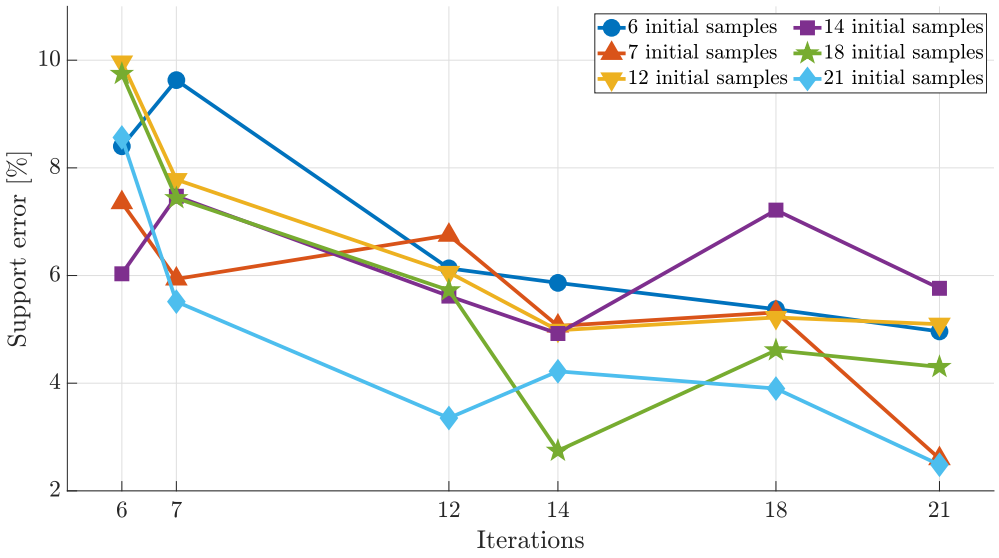


FIGURE 6.13: Evolution of the support error with respect to the Bayesian hyperparameters of FuzzBO.

Considering the effect of the training set size, the increase of the initial samples number has a great impact on the performance of the algorithm. Contrary to the area approximation, the support error spread is almost constant whatever the iteration number. Whatever the number of iterations, the difference of errors between the best and the worst configurations is the same, around 4 percentage points.

Moreover, the effect of the initial samples size is more random: the 7 initial samples configuration provides better results than the 18 initial samples configuration; the 14 initial samples configuration gives worse results than the 6 initial samples configuration. Nonetheless, the greatest initial sizes (18 and 21 samples) are less affected by the reduction of performance (fewer fluctuations).

Similarly to the area approximation, the difference between the best configuration (21 initial samples and 21 iterations) and the worst configuration (12 initial samples and 6 iterations) is highlighted in Fig. 6.14. These configurations respectively generate an area error of 2.48% and 9.96%.

The target value for the lower bound is 48.35 Hz and the upper bound is 69.45 Hz. The gap between the two configurations is quite small, since the worst (resp. best) setting gives a lower bound of 49.07 Hz (resp. 48.55 Hz) and an upper bound of 67.96 Hz (resp. 69.12 Hz).

Discussions

The results showed that the number of iterations is also a key parameter to efficiently estimate the support of the output fuzzy set. This observation rises confident on the behaviour of the suggested method since, with more iterations, the Bayesian algorithm performs better.

Contrary to the area, the initial sample size effect is more random if the number of initial samples is kept under a certain value. This observation is troublesome as it were. Indeed, it implies that the infill criterion used for FuzzBO, the Expected Improvement, performs worse if the number of samples

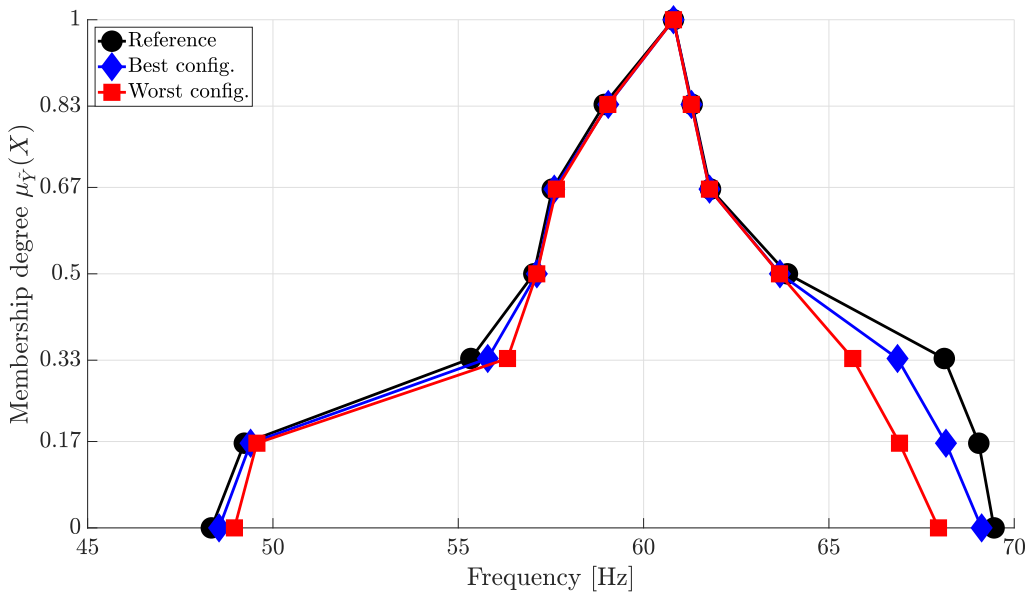


FIGURE 6.14: Comparison between the worst and best configurations for the support approximation (● support is 21.1 Hz; ♦ support is 20.57 Hz; ■ support is 18.99 Hz).

is low. This effect has already been identified in Chapter 5 and indicates a limitation of the Expected Improvement. Nonetheless, an increase of the iteration number fades this effect.

6.4.3.3 How to fix efficiently the FuzzBO parameters?

From the previous subsection, the results highlighted some influence of the different parameters of the bayesian algorithm: the initial sample number and the iteration number. This section aims at comparing several couples of parameter values, which, after the execution of the algorithm, produce training sets of the same size.

Area error

Table 6.5
Comparison between the best and worst area approximation.

	Best configuration			Worst configuration		
Final training set size	Initial sample	Iterations	Perf.	Initial sample	Iterations	Perf.
157 points	12 [2 × **d**]	7 [1 × **m**]	10.8%	14 [2 × **m**]	6 [1 × **d**]	12%
181 points	6 [1 × **d**]	12 [2 × **d**]	10.3%	18 [3 × **d**]	6 [1 × **d**]	11.4%
211 points	7 [1 × **m**]	14 [2 × **m**]	8.6%	21 [3 × **m**]	7 [1 × **m**]	11.1%
253 points	6 [1 × **d**]	18 [3 × **d**]	8.5%	14 [2 × **m**]	14 [2 × **m**]	8.5%
289 points	6 [1 × **d**]	21 [3 × **m**]	6.4%	12 [2 × **d**]	18 [3 × **d**]	7%
295 points	7 [1 × **m**]	21 [3 × **m**]	7%	21 [3 × **m**]	14 [2 × **m**]	7.8%
325 points	12 [2 × **d**]	21 [3 × **m**]	7.1%	18 [3 × **d**]	18 [3 × **d**]	7.2%

Table 6.5 summarises the different couples of parameter values identified here. The results shown in this table come from the previous study on the area and support approximation. For a given final training set size, a best and worst configurations are compared. The value of each parameter (initial

sample number and iterations) is reminded as long as their dependence toward the characteristics of the studied function. Finally, the area error is given in the performance column.

First, the best configuration is almost always associated with a high iteration number and a low initial training set size (results for 181 to 325 points). Conversely, the worst configuration is always linked to a high initial training set size. Nonetheless, the difference between the two configurations are relatively low (between 0 and 2 percentage points).

No particular setting arises from the results. Indeed, whether it is the setting with respect to the dimensionality or the α-cut, the results are either good or bad. For instance, for the training set of 181 samples, the best configuration is associated with the dimensionality – $1 \times \mathbf{d}$, for the initial sample number; $2 \times \mathbf{d}$, for the iterations – and the worst configuration is also linked to the dimensionality – $3 \times \mathbf{d}$, for the initial sample number; $1 \times \mathbf{d}$, for the iterations. Similarly, for the training set of 211 points, the best configuration is associated with the α-cut discretisation – $1 \times \mathbf{m}$, for the initial sample number; $2 \times \mathbf{m}$, for the iterations – and the worst configuration is also linked to the α-cut discretisation – $3 \times \mathbf{m}$, for the initial sample number; $1 \times \mathbf{m}$, for the iterations.

Support error

Table 6.6
Comparison between the best and worst support approximation.

	Best configuration			Worst configuration		
Final training set size	Initial sample	Iterations	Perf.	Initial sample	Iterations	Perf.
157 points	14 [$2 \times \mathbf{m}$]	6 [$1 \times \mathbf{d}$]	6%	12 [$2 \times \mathbf{d}$]	7 [$1 \times \mathbf{m}$]	7.8%
181 points	6 [$1 \times \mathbf{d}$]	12 [$2 \times \mathbf{d}$]	6.1%	18 [$3 \times \mathbf{d}$]	6 [$1 \times \mathbf{d}$]	9.7%
211 points	7 [$1 \times \mathbf{m}$]	14 [$2 \times \mathbf{m}$]	5.1%	21 [$3 \times \mathbf{m}$]	7 [$1 \times \mathbf{m}$]	5.5%
253 points	14 [$2 \times \mathbf{m}$]	14 [$2 \times \mathbf{m}$]	4.9%	6 [$1 \times \mathbf{d}$]	18 [$3 \times \mathbf{d}$]	5.4%
289 points	6 [$1 \times \mathbf{d}$]	21 [$3 \times \mathbf{m}$]	5%	12 [$2 \times \mathbf{d}$]	18 [$3 \times \mathbf{d}$]	5.2%
295 points	7 [$1 \times \mathbf{m}$]	21 [$3 \times \mathbf{m}$]	2.6%	21 [$3 \times \mathbf{m}$]	14 [$2 \times \mathbf{m}$]	4.2%
325 points	18 [$3 \times \mathbf{d}$]	18 [$3 \times \mathbf{m}$]	4.6%	12 [$2 \times \mathbf{d}$]	21 [$3 \times \mathbf{m}$]	5.1%

Table 6.6 summarises the different couples of parameter values identified here. The presentation format is similar to the previous table.

The best configuration is here associated with only a high number of iterations. The initial sample number is either low (for four final training set size) or high (for three final training set size). Conversely, the worst configuration is almost always linked to a high number of initial samples.

As for the setting with respect to the dimensionality or the α-cut discretisation, the observations are similar to those from the area approximation.

6.4.3.4 Conclusions

To conclude, these studies showed that the iteration number is the key parameter to set for the Bayesian algorithm: the greater, the better. The effect of the initial sample number is more arbitrary,

except if it is sufficiently high (in this case, greater than 18 initial samples) where the performance is stabilising. Nonetheless, the performance of FuzzBO was higher if the initial sample number is kept low.

This observation allows the user of this algorithm to manage the computational cost of the sample evaluations by simply controlling the number of iterations. Hence, the computational budget has to be mainly spent with this key parameter. The Bayesian algorithm is, thus, set with a low number of initial samples (about $[1 \times \mathbf{d}]$, *i.e.* 6 points per α-cut) and a number of iterations depending on the total number of samples allowed for the study.

6.4.4 Influence of the α-cut decomposition

This third test aims at highlighting the effects of the α-cut discretisation m over the performance of the Bayesian algorithm. Typically, this parameter is also a key factor, since it controls the representativity of the output fuzzy set.

Following the conclusions of the previous section, the initial samples number is given to 6 points per α-cut and the number of iterations is parameterised according to a computational budget. Four targeted computational budgets are considered, namely 100, 150, 250 and 350. However, some samples provided by the Bayesian algorithm may be redundant (see Section 6.3.2 for some explanations). These samples are removed to ensure the correct execution of the algorithm and thus, the true computational budget is lower than the targeted one.

Table 6.7 recalls the true computational budgets and the associated number of iterations. For the 10 α-cut case, the true computational budget of 75 samples is too low (the number of iterations would be 1) and has not been considered here.

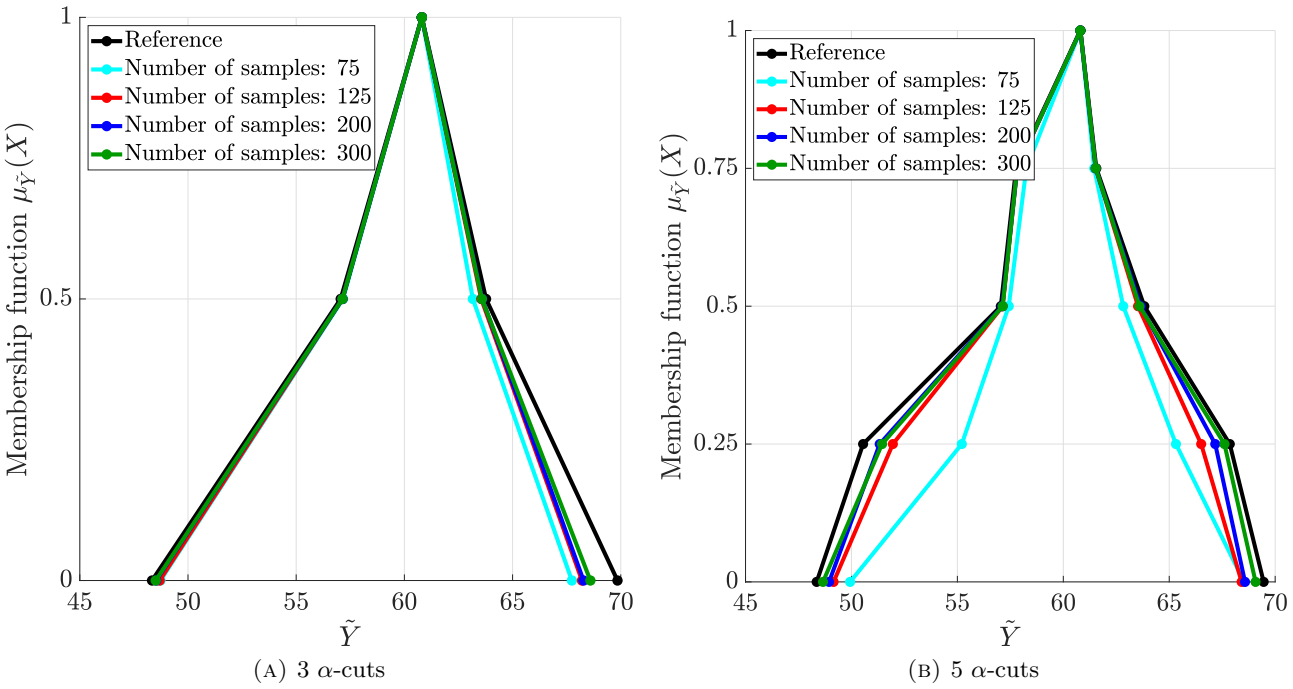
Table 6.7
Iteration number with respect to the true computational budget.

True computational budget	$m = 3$	$m = 5$	$m = 7$	$m = 10$
75	12	4	4	\
125	22	12	8	4
200	38	22	15	9
300	55	32	22	14

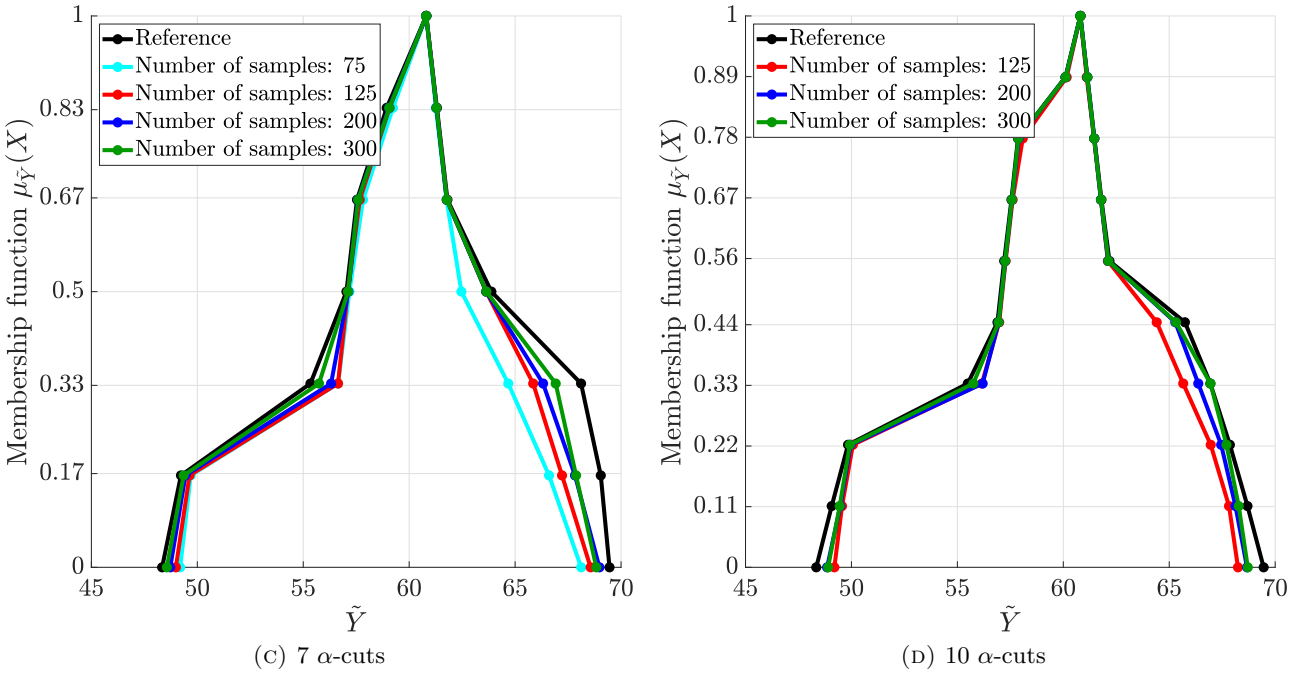
6.4.4.1 Fuzzy set output approximation

Fig. 6.15 shows the effects of the α-cut decomposition over the estimation performance of the output fuzzy set for each considered discretisation.

In all cases, the output fuzzy set is well approximated, with a final training set size of 125 to 300 samples. For 75 samples, if the number of α-cut is 3, the approximation is fair enough, but for either 5 or 7, the approximation performance is not good enough. Indeed, as the configurations with a higher



(A) 3 α-cuts (B) 5 α-cuts



(C) 7 α-cuts (D) 10 α-cuts

FIGURE 6.15: Output fuzzy set approximation with respect to the computational budget and the α-cut number.

number of samples allow determining the overall shape of the output fuzzy set, the approximation for $m = 5$ is not good (especially, for a membership value of 0.25). For the approximation for $m = 7$, the output fuzzy set is better predicted than for $m = 5$, but still the shape is not fully correct.

In addition, the improvement brought by the increase of iterations between 75 and 125 samples is moderately important compared to the other training set size increases. The precision improvement brought from 125 samples to 200 or 300 is low. The effect of the iteration increase is rather important

for the $m = 5$ and $m = 7$ approximations, whereas for the two other discretisation values, the effect is quite minor.

This test confirms some previous observations, namely that the convergence of the FuzzBO algorithm is greater if the number of iterations is important. In addition, even if the increase of α-cut lowers the computational budget allocated per membership value, the suggested method still succeeds in correctly approximating the output fuzzy set.

6.4.4.2 Area and support approximations

Table 6.8 shows the error associated with the area approximation for each final training set size and α-cut number. It is worth-noting that the increase of the error approximation with the increase of the α-cut number is completely natural, since the fuzzy output set shape is more and more complex as the discretisation of the sets are more important.

First, the results show that an increase of the final training set size improves the quality of the area approximation. Indeed, for all number of α-cuts, the error goes from 20%, on average, to roughly 5%. Then, the ratio of computational cost becomes acceptable when the number of final samples reaches 125. Indeed, the area approximation error is about 10% with a low budget (125 samples) and quickly drops to 7% with a greater budget (200 samples).

Table 6.8
Area error approximation with respect to the computational budget and the α-cut number.

	$m = 3$	$m = 5$	$m = 7$	$m = 10$
$n_T = 75$	11.23%	27.09%	19.36%	\
$n_T = 125$	7.42%	10.75%	12.69%	9.03%
$n_T = 200$	6.53%	6.59%	9.11%	5.05%
$n_T = 300$	5.64%	4.9%	6.81%	3.17%

Table 6.9 shows the error associated with the support approximation for each final training set size and α-cut number. Here, the observations are simpler since the support is similar for any discretisation values m.

Table 6.9
Support error approximation with respect to the computational budget and the α-cut number.

	$m = 3$	$m = 5$	$m = 7$	$m = 10$
$n_T = 75$	11.61%	11.86%	10.38%	\
$n_T = 125$	9.26%	8.71%	7.31%	9.78%
$n_T = 200$	8.19%	6.95%	4.08%	6.14%
$n_T = 300$	6.62%	3.22%	4.02%	6.19%

The results show that, for a small final number of samples ($n_T = 75$), the support approximation error is equivalent, whatever the α-cut number chosen. Conversely, when the final number of samples increase, some discrepancies appear. From $m = 3$ to $m = 7$, the error associated with the support

approximation clearly reduces: for $n_T = 125$, the error goes from 9% to 7% while for $n_T = 300$, the error drops from 7% to 4%. However, for a discretisation value of $m = 10$, the support area approximation is higher and may be comparable to either the error of $m = 3$ or $m = 5$, questionning the efficiency of increasing the number of α-cut m, for the support approximation.

On overall, the error value remains acceptable for any α-cut value, when the final number of samples is at least equal to 125. The ratio between the computational cost and the precision is also acceptable since the support approximation error is roughly 7%, when $n_T = 125$. Increasing the final training set size is not optimal with regard to this ratio, since the support error drops to 5% for $n_T = 300$ (more than twice the number of samples).

Conclusions

Those observations show that the discretisation value of the input and output fuzzy sets m has a low effect on the performance of the FuzzBO algorithm. Indeed, with regard to the shape of the output fuzzy set, the approximation efficiently determines the shape, as long as the final number of samples, *i.e.* the number of iterations of the Bayesian algorithm, is sufficiently high. For the area and support approximation, the results are similar: the key aspect is to have a sufficient number of iterations.

6.4.5 Pairing issue

6.4.5.1 Highlight of the issue

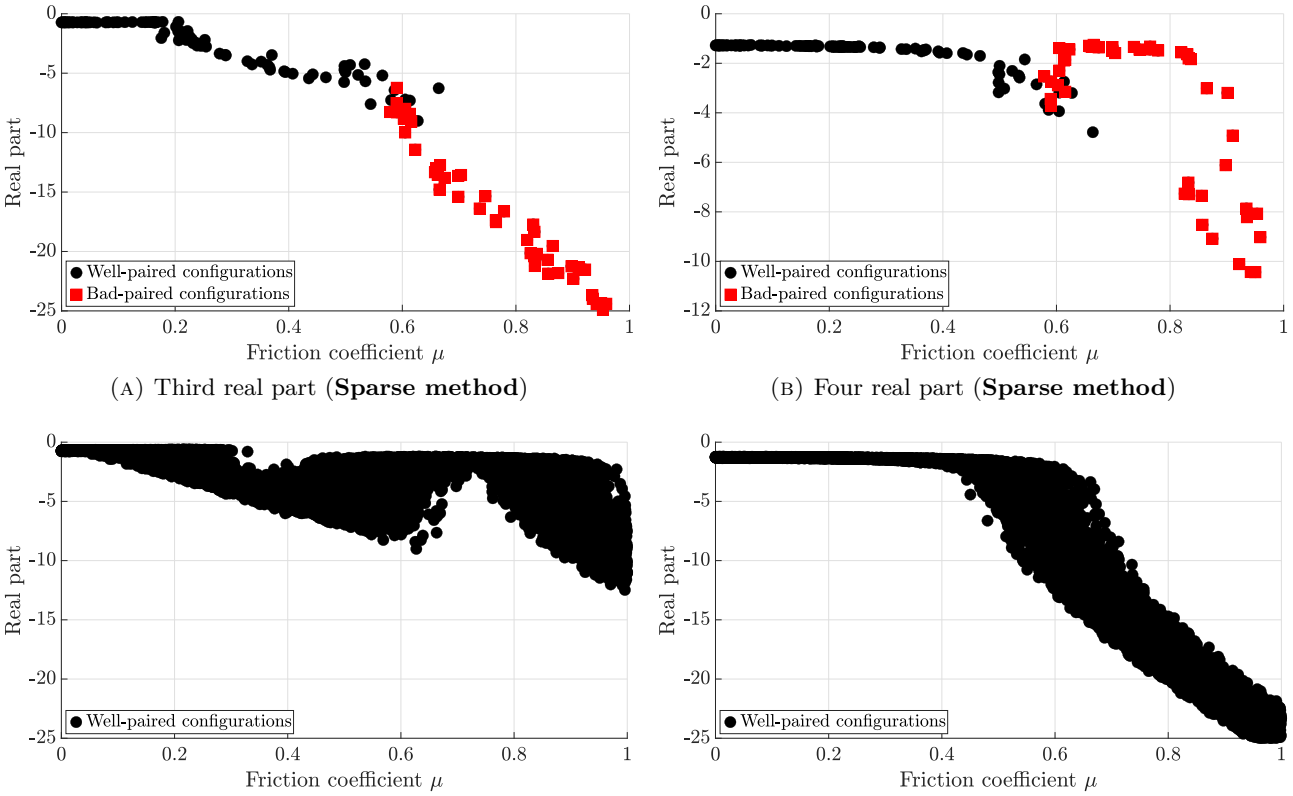
After the execution of the FuzzBO algorithm, about 5% of the final training sets exhibit pairing issues. To investigate the failure of the pairing procedure, the results presented in Section 6.4.2 to Section 6.4.4 were paired once again using a greater reference set of families (about 10,000 already evaluated configurations).

Fig. 6.16 compares the real part of the third (first column) and fourth (second column) eigenvalues of the Double Hulten model, where the eigenforms of the configurations were paired with two different procedures. The first row corresponds to the pairing resulting from the common procedure presented in Chapter 3, denoted sparse method, while the second row gives the pairing results obtained with 10,000 already evaluated configurations, referred as full method.

For the first row, the well-paired configurations are shown in black circles, whereas the bad-paired in red squares. For the second row, the well-paired configurations are displayed in black circles.

One of the main properties of the eigenvalues of a system (even for squeal models) is that if the variations of parameters between two configurations is sufficiently small, the variations of the outputs can be considered small too.

In the light of the sample density of the full method and following the previous assertion, if the red squared configurations are reported in the second row figures, they would have been nested in



(A) Third real part (**Sparse method**)

(B) Four real part (**Sparse method**)

(C) Third real part (**Full method**)

(D) Four real part (**Full method**)

FIGURE 6.16: Comparison between the two pairing strategies [Projected on the μ-dimension].

an envelope made of the 10,000 evaluations. If the bad-paired configurations of the first and second columns are inverted, these configurations fit well in the given envelopes.

The considered pairing strategy strongly hinges on the representativity of the families set. When the evolution between some eigenforms are small, there is no need to over-sampling the associated area of the design space. Conversely, in complex areas of the design space where the eigenforms of two different eigenvalues are close to one another, an extra care has to be made on the quality of these eigenforms. For instance, in veering areas, the MAC values may be high between coupled modes, since their eigenform is a linear combination of the eigenform of the two modes before the critical coupling point, inducing potential difficulties in the pairing process.

For the configurations of Fig. 6.16(A) and (B), the computation of the AUTOMAC for the eigenvector basis $\mathcal{X}$ reveals that roughly 4% (6 config. over 157) of the configurations exhibits high-coupling state. Fig. 6.17 shows the AUTOMAC of the eigenvector basis for one of these configurations. The big red (resp. blue vanishing) squares indicate high (resp. low) correlation between two modes.

Here, the eigenvector ψ_1 and ψ_2 exhibits a correlation of 97%. This high value obviously represents a problem when using the MAC between two sets of parameters.

Table 6.10 presents a comparison when the modal forms ψ_1 and ψ_2 of the configuration 133 are compared to the families $\mathcal{F}_1$ and $\mathcal{F}_2$ of the final training set, paired with the full method. With this

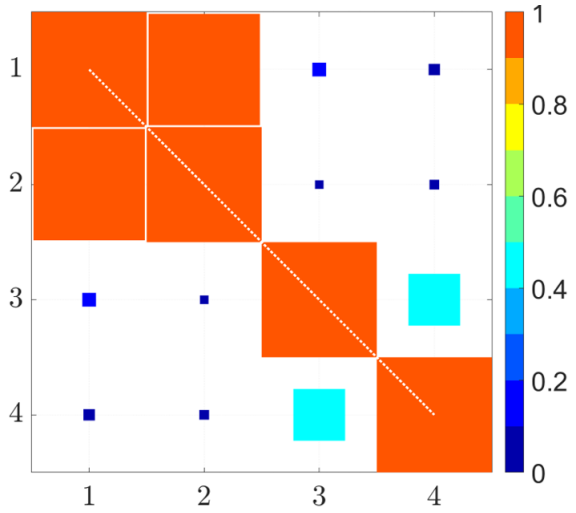


FIGURE 6.17: AUTOMAC of the configuration 133.

strategy, the eigenvector ψ_1 is associated with $\mathcal{F}_1$ and ψ_2 with $\mathcal{F}_2$. The last two rows of the tables correspond to the number of values where the corresponding eigenshape gives the highest correlation value over 95%.

Table 6.10
Comparison of the MAC values of ψ_1 and ϕ_2.

	Family $\mathcal{F}_1$		Family $\mathcal{F}_2$	
	ψ_1	ψ_2	ψ_1	ψ_2
Mean MAC	80,5	85,3	35.2	32.6
Max MAC	99.84	98,8	97.3	99.9
Values when MAC$(\psi_1, \mathcal{F}_i)$ is higher	24	\	28	\
Values when MAC$(\psi_2, \mathcal{F}_i)$ is higher	\	10	\	27

The results for both families $\mathcal{F}_1$ and $\mathcal{F}_2$ clearly show that there is some ambiguity about the pairing of the eigenforms ψ_1 and ψ_2. Indeed, even if the maximum of MAC is given for ψ_1 (99.84%), the maximum of the MAC for ψ_2 is very close to the latter one (98.8%). For the mean MAC, ψ_2 shows the highest correlation with a value of 85.3%, meaning that, on average, this eigenform ψ_2 is closer to the family $\mathcal{F}_1$ than ψ_1. The comparison with the family $\mathcal{F}_2$ exhibits similar conclusions.

Since the full method uses a huge database of configurations, this ambiguity is worked around efficiently. In FE applications, it is impossible to create a set of reference configurations as used in Fig. 6.16 to overcome the pairing limitation. Consequently, the following section presents a detection criterion and a strategy to deal with this issue.

6.4.5.2 Detection criterion

The proposed criterion is very simple and directly hinges on the AUTOMAC, with real vectors. (Eq. 6.10) is evaluated for the whole eigenbasis $\mathcal{X}$, and, most of the time, ψ_i is only correlated with itself (meaning only one 100% per mode).

$$C(\mathcal{X}) = \text{AUTOMAC}(|\mathcal{X}|) \tag{6.10}$$

where $\mathcal{X}$ is the eigenbasis composed of all ψ_j. Hence, the element $C_{i,j}$ of the matrix C gives the correlation between the eigenvector ψ_i and ψ_j.

For the eigenvectors which are correlated with other eigenvectors of the basis $\mathcal{X}$, the strategy mainly depends on the number of correlated eigenvectors. If only one vector ψ_j is fully correlated with ψ_i, then the common mode coupling is identified: the eigenbasis $\mathcal{X}$ is said to be 1-coupled. If n additional vector are fully correlated with ψ_i, then the given eigenbasis is said to be n-coupled.

In practice, it is not possible to use a threshold k equal to 100%. Indeed, many tests showed that a detection threshold k of 98% allows a correct detection of these pathological configurations. Hence, it would mean that a configuration with a vector which holds two $C_{i,j}$ greater than the threshold k would be classified as pathological.

Fig. 6.18 shows an application of the statement about coupled modes with the final training set considered for the display of the pairing issue. The 0-coupled configurations are given in green squares; the 1-coupled, in red squares and the 2-coupled, in blue squares. Finally, the bad-paired configurations are depicted with a black dot.

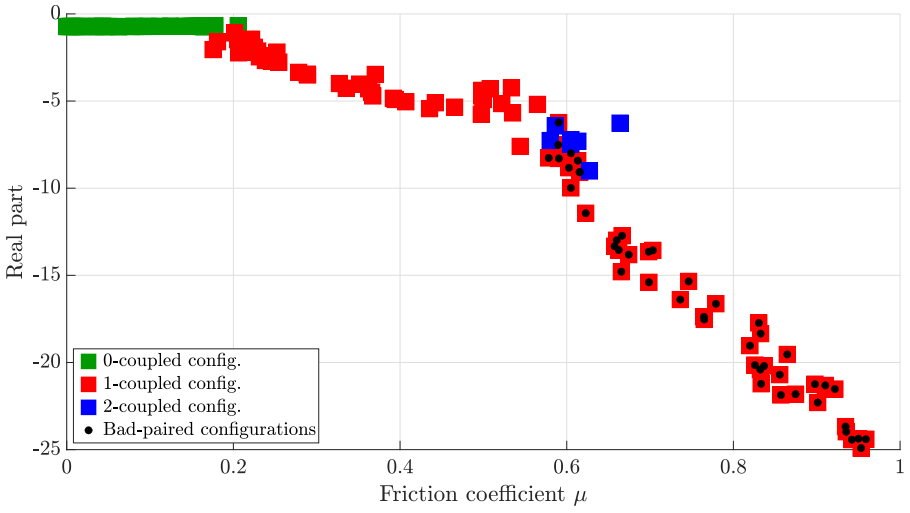


FIGURE 6.18: Superposition of the n-coupled and the bad-paired configuration.

First, three types of coupled configurations are here distinguished. A comparison with the bad-paired configurations determined in Fig. 6.16 shows that only the 1-coupled configurations are concerned by the pairing issue, whereas, interestingly, the 2-coupled configurations are well-paired.

An analysis of the pairing process highlighted that the blue configurations were the cause of the pairing issue. Indeed, when the reference family $\mathcal{F}_i$ is incremented with these configurations, the black-dotted configurations are paired with those configurations and the pairing strategy failed.

Nonetheless, the full method of Fig. 6.16 does not exhibit this issue, since the sample density is sufficiently high to absorb it. For the sparse method, these configurations introduce ambiguities in the pairing process, resulting in the aforementioned pairing issue.

In practice, only one strategy almost succeeds in dealing with those n-coupled configurations and is described in the following section.

6.4.5.3 Proposed strategy

As the 2-coupled configurations were identified to be the cause of the failing of the pairing strategy, a rather naïve strategy is to remove them from the configuration set to pair. With regard to the 1-coupled configurations, several tests showed that the pairing strategy succeeds in dealing with them.

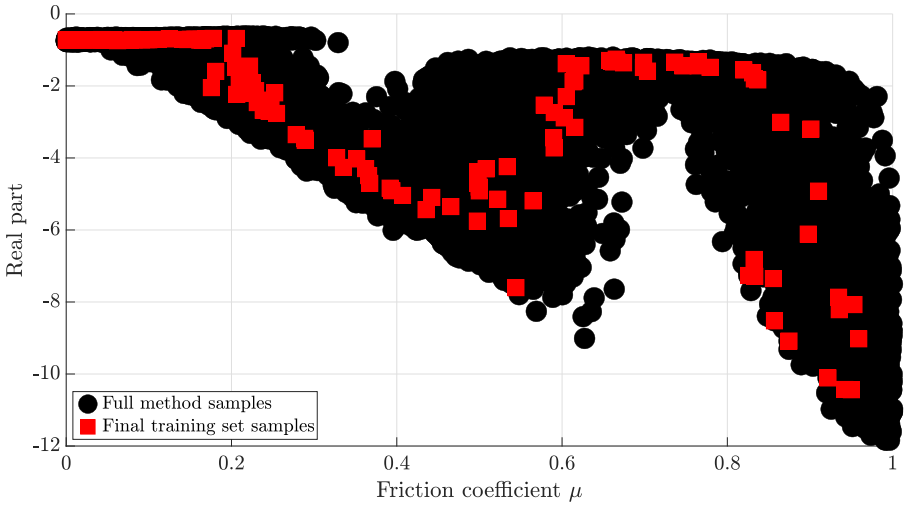


FIGURE 6.19: Comparison of the removing strategy and the full method.

Observations

Fig. 6.19 shows the results of the pairing strategy without the 2-coupled configurations. Here, the samples of the full method are only used to analyse the results of the new pairing strategy and are not considered in this pairing strategy. The results show that, with this new strategy, the final training set is nested inside the full method samples, which rises confidence about the performance of the suggested strategy.

To assess for the performance of the suggested method, all the training sets obtained after executing the FuzzBO algorithm are here considered. These tests allow studying the method over a large number of training sets. Some detection threshold values k were tested to highlight the performance of the strategy with respect to the number of removed modes.

Table 6.11 summarises the results of the suggested strategy. The best, mean and worst performance assess the number of bad-paired configurations by comparing the final pairing to a pairing with the full

Table 6.11
Number of badly paired configurations.

k value	85	90	95	97	98	None
Best performance	0	0	0	0	0	120
Mean performance	0.15	0.25	0.59	0.68	0.78	26.31
Worst performance	1	4	12	13	13	80

method. Finally, the performance of the classical strategy (sparse method) is showed by the column 'None'.

First, the mean performance shows great results since almost all the configurations are well-paired: less than 1 configuration is badly paired, whatever the threshold value k. Conversely, the sparse method exhibits poor results since, on average, 26 configurations are badly paired.

Secondly, the worst performance shows that the suggested strategy is not perfect. Indeed, depending on the threshold value k, some configurations are still badly paired. The greater the threshold value is, the higher the badly paired configurations will be. Nonetheless, compared to the traditional strategy, the suggested strategy still perform better and succeeds in pairing 95% of the previously failed paired final training sets.

Table 6.12
Removed configurations.

k value	85	90	95	97	98	None
Mean removed configurations	18%	14%	7%	5%	3%	0%

Table 6.12 shows the percentage of removed points for each threshold value. Only the mean performance values are emphasised since the best and worst performance values give similar percentage.

This table highlights the main drawback of this method, *i.e.* the removing of evaluated configurations. For a threshold of $k = 85\%$, about 18% of the total number of configurations are set aside to allow a correct pairing of the other 82% of the configurations. For a threshold of $k = 98\%$, about 3% of the total number of configurations are set aside to allow a correct pairing of the other 97% of the configurations.

These configurations correspond to a computational budget which has been spent to evaluate them. For FE models, this computational budget is about hours per configuration. Thus, relying on a strategy which set aside some evaluated configuration is not optimal.

Discussions

The removal strategy shows interesting results since it manages to counterbalance the effects of pathological configurations, as highlighted by the comparison with the full strategy in Table 6.11. Nonetheless, after the execution of the FuzzBO algorithm, the final training set cannot be labeled well- or badly paired since no reference is available (as with the full strategy).

Thus, with regard to this issue, an additional step has to be considered when the algorithm has ended. This step consists in using the removal strategy to pair the final training set. Then, a comparison between the traditional pairing strategy (presented in Chapter 3) and the removal strategy is done. If the pairing is different, the result of the removal strategy is kept – a different result would mean that one of the configurations is pathological. If the pairing is similar, then the result of the traditional pairing strategy is kept.

Finally, a natural questioning arises about the removed points. Several tests showed that it is not possible to incorporate them after the first pairing, which would remain fixed. Indeed, the pairing issue with these points is related to the lack of compatibility with the other computed configurations of the training set. A possible solution would be to densify the number of computed configurations at the vicinity of those removed points, allowing a better pairing.

6.5 Approximation of the coalescence graph

6.5.1 Assessment procedure

The aim of this section is to use the proposed algorithm to approximate the coalescence graph for a given eigenvalue. The determination of this graph is done by determining the inferior and superior bounds for each α-cut of the fuzzy friction coefficient. The other parameters are modelled as interval fuzzy numbers. The case study used for this application is the same as the one used in Section 6.4.

The sections 6.5.2 to 6.5.4 assess the performance of the suggested method with respect to its key parameters. First, the method is compared to a common strategy, which is the surrogate modelling approach. Then, the effects of the Bayesian formalism are highlighted. Finally, a study over the impact of the decomposition with the friction coefficient is carried out. For all these applications, the most complex function is studied: the real part of the third eigenvalue of the Double Hulten.

One metric is used to measure the performance of the suggested algorithm: the CRMSB, which is given by (Eq. 6.11). The analysis of this metric is rather simple: the lower, the better. Besides, the interpretation is similar to the CRMSE, meaning that a CRMSB greater than 1 is associated with bad results; a CRMSB lower than 0.5 is associated with good results. Between these two values, the results are mitigated.

$$\text{CRMSB} = \sqrt{\frac{\sum_{i=1}^{n_d} \left(y_{min}^i - \widehat{y}_{min}^i\right)^2}{n_d} + \frac{\sum_{i=1}^{n_d} \left(y_{max}^i - \widehat{y}_{max}^i\right)^2}{n_d}} \tag{6.11}$$

6.5.2 Influence of the Bayesian formalism

Two strategies are here compared to account for the best way of generating the samples used to construct a coalescence graph: the first is the fully Random strategy with the approximation of the real part with the GP; the second is the incremental strategy with the suggested algorithm.

To do so, a rather refined discretisation of the friction coefficient is considered, with 11 values (which corresponds to an increment of 0.1). The computational budget is set to 220 samples, which corresponds to $p = 7$ iterations and $q = 6$ initial samples per friction coefficient values.

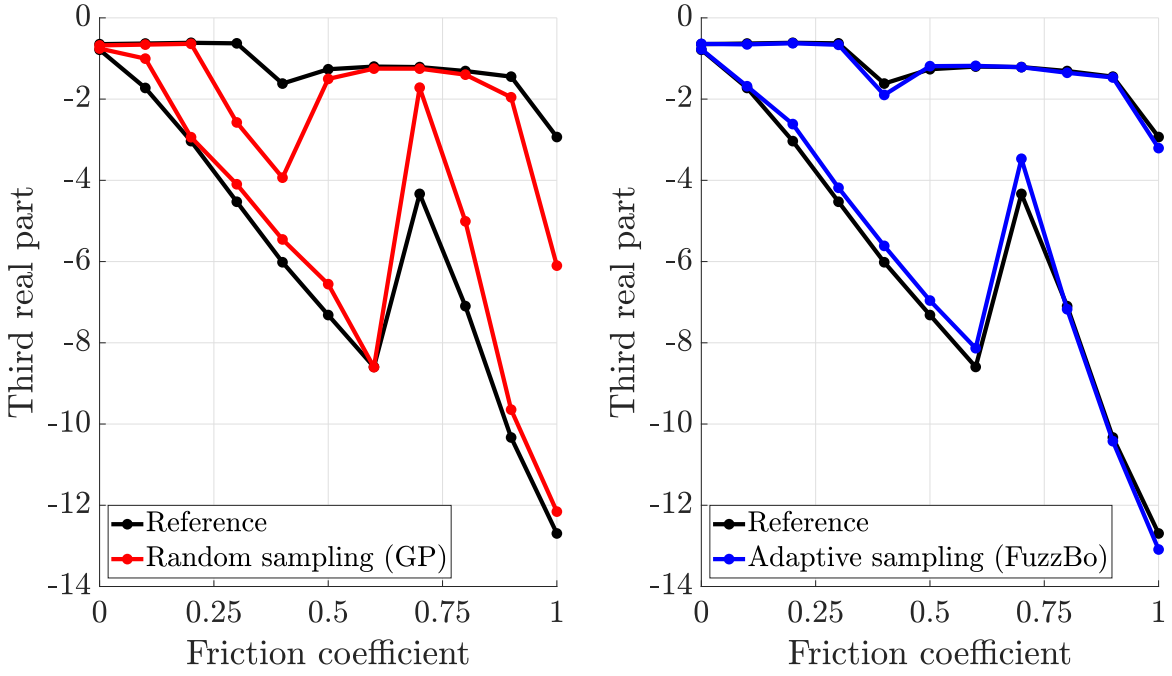


FIGURE 6.20: Comparison of the output fuzzy set approximation using GP or FuzzBO.

Fig. 6.20 compares the two considered methods. The reference min-max bounds are given in black line. Fig. 6.20(A) gives the extremum bounds associated with the GP approximation, while Fig. 6.20(B) those associated with the FuzzBO algorithm.

First, the results with the fully random sampling strategy (GP) are quite interesting. Indeed, even if some bounds are badly approximated (namely, the maximum bounds for $\mu = \{0.3, 0.4, 1\}$ and the minimum bounds for $\mu = \{0.7, 0.8\}$, the approximation of the envelope of the coalescence graph is pretty fair. The maximum error is, besides, located at a friction coefficient of 1, with a value of 3.17.

Conversely, the results with the adaptive sampling strategy (FuzzBO) are almost identical to the reference min-max bounds. Indeed, the maximum gap is about 0.86. In addition, the friction coefficient values where the fully random sampling underestimates the extrema bounds are clearly well-approximated with the suggested method.

With regard to the CRMSB values, the FuzzBO algorithm gives a CRMSB value of 0.413 and the GP approximation generates a CRMSB value of 1.734, showing a clear benefit of using an adaptive training set for the approximation of the coalescence graph.

6.5.3 Influence of the Bayesian hyperparameter setting

The first test aims at studying the performance of the suggested algorithm with respect to its Bayesian hyperparameters. The first one is the initial samples size, which controls the initial amount of information available for the method to initialise Bayesian optimisation; the second one is the number of iterations, which controls the amount of samples that the suggested algorithm can add to the training set $\mathcal{T}$.

Here, four initial samples size are tested: $[\![3;6]\!]$; and four iteration number values are also tested: $[\![4;7]\!]$. Ten different initialisations are evaluated for the randomly generated training set, to assess the robustness of the suggested method.

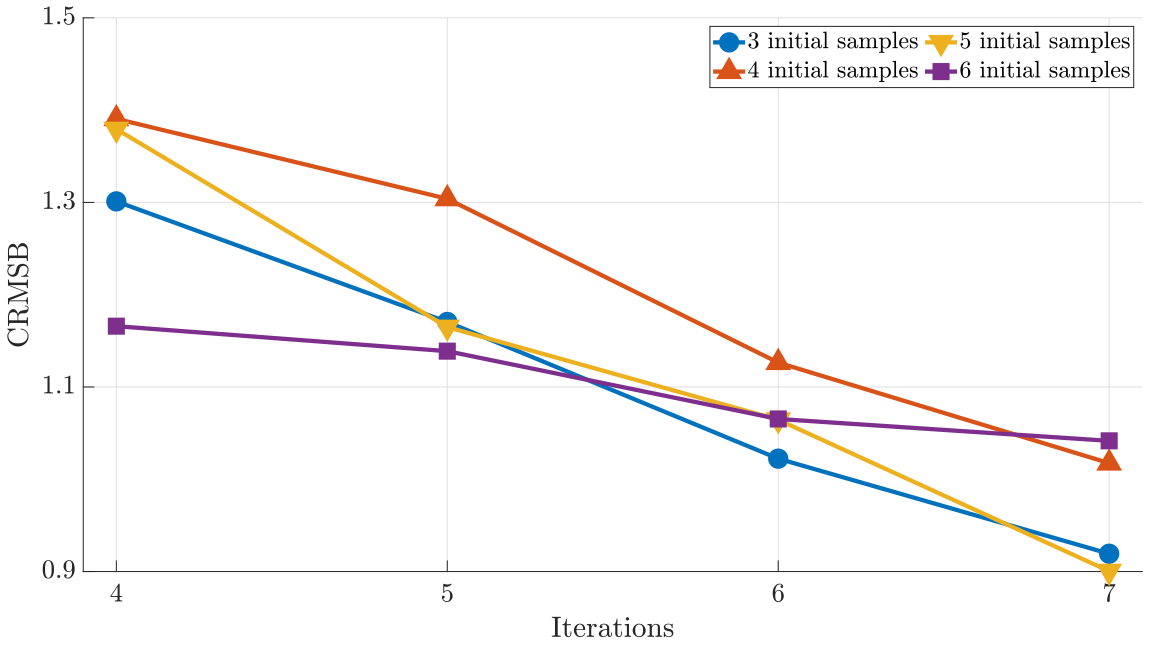


FIGURE 6.21: Evolution of the CRMSB with respect to the iteration number for different initial samples number.

Observations

Fig. 6.21 shows the evolution of the CRMSB with respect to the initial samples size and the number of iterations.

First, the results show that, whatever the setting of the algorithm, the CRMSB is decreasing when the number of iterations increases. For instance, for 4 initial samples per friction coefficient value, the CRMSB goes from 1.3 to 0.92. Similarly, for 6 initial samples, the CRMSB goes from 1.38 to roughly 0.9.

Secondly, as the effect of the number of iterations is clear and simple (the higher, the better), the effect of the number of initial samples is more mitigated. Indeed, an increase of the number of initial samples is not correlated with a better approximation of the bounds. For instance, for 4 iterations, the 3 initial samples configuration generates a greater CRMSB than the 4 and 5 initial samples. Conversely, for 7 iterations, the 6 initial samples configuration provides a worse CRMSB than the 3 initial samples.

Table 6.13
Mean computational budget spent by the FuzzBO algorithm.

Iterations \ Initial samples	3	4	5	6
4	120.9	142.9	164.9	186.9
5	131.9	153.9	175.9	197.9
6	143	165	187	209
7	154	176	198	220

Table 6.13 shows the mean computational budget for each considered configuration. The aim of this table is to highlight the contribution of the increase of samples with respect to the bounds approximation precision.

For instance, the CRMSB value associated with (3,7) – where the first digit corresponds to the initial samples size and the second digit with the iteration number – is roughly 0.92, whereas the CRMSB of (6,7) is 1.03. Here, in this case, the number of samples generated with the first hyperparameter values is 154 against roughly 220. On the contrary, the CRMSB value, linked to (6,4), is 1.17 whereas the CRMSB value of (4,4) is 1.34. The final number of samples generated by the FuzzBO algorithm is respectively 186.9 and 142.9 on average.

Discussions

These observations show that, as with the approximation of output fuzzy sets, an initial increase of the training set $\mathcal{T}$ is needed only if the number of iterations is kept low. Conversely, if the number of iterations is high, then the initial training set size has to be low to allow better performance of the algorithm.

In a way, it means that the computational budget has to be spent on increasing the number of iterations for the suggested algorithm. This result allows a simple control of the computational cost of the algorithm and the performance of the final approximation of the bounds.

6.5.4 Influence of the interval decomposition

Following the previous observations, the initial number of samples is 3 and the iterations number is set according to the allocated computational budget. Ten initial training sets are then randomly generated to assess the performance of the algorithm.

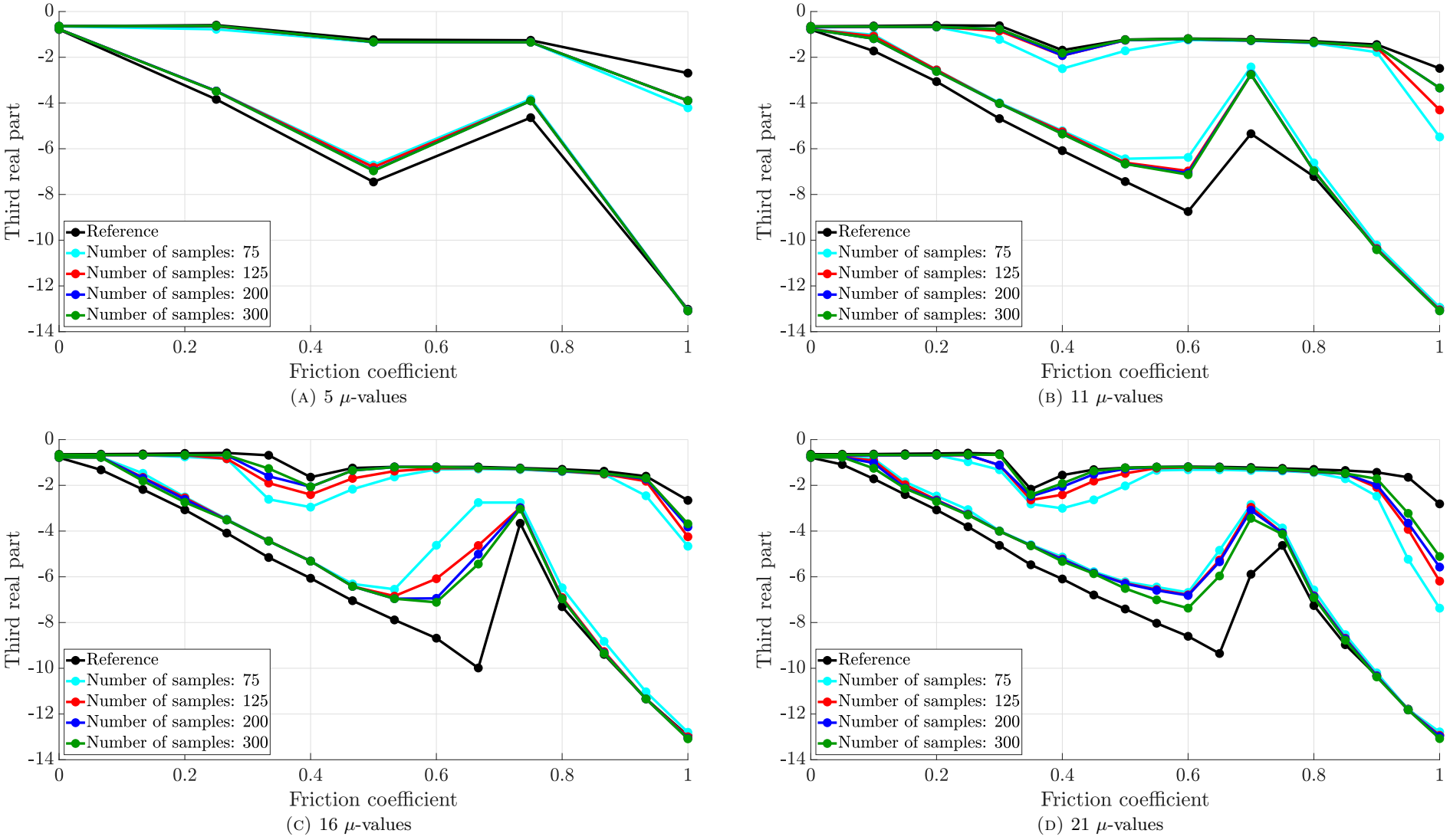
Table 6.14 gives the number of iterations of the FuzzBO algorithm with respect to the considered computational budget and the number of μ-values. For the comparison, a focus is particularly made on three coupling areas: between $\mu = 0.3$ and $\mu = 0.45$, between $\mu = 0.6$ and $\mu = 0.7$ and above $\mu = 0.95$.

Table 6.14
Iteration number with respect to the true computational budget.

True computational budget	$n_\mu = 5$	$n_\mu = 11$	$n_\mu = 16$	$n_\mu = 21$
75	8	3	2	1
125	13	5	3	2
200	18	8	5	3
300	28	12	8	6

Observations

Fig. 6.22 highlights the results for each considered scenario. The resulting bounds are projected on the μ-parameter plan to allow a one-dimensional display of the bounds evolution.



(A) 5 μ-values (B) 11 μ-values

(C) 16 μ-values (D) 21 μ-values

FIGURE 6.22: Coalescence graph approximation with respect to the computational budget and the number of μ values.

On overall, the evolution of the coalescence graph is well approximated, except for two areas: the minimum bounds of the coupling area located between $\mu = 0.6$ and $\mu = 0.7$; the maximum bounds of the coupling area located above $\mu = 0.9$. This under performance is especially highlighted for Fig. 6.22(B) to (D), since they sample more precisely these areas. Conversely, the only remaining coupling area is well approximated, namely the area located between $\mu = 0.3$ and $\mu = 0.45$.

With regard to the performance of the algorithm, the increase of allocated samples with higher iteration number provides small improvements of the coalescence graph approximation. Only two cases show noticeable enhancements: the maximum bounds of $\mu = 1$ and the minimum bounds of $\mu = 0.6$ for respectively 11 and 16 μ-values.

When comparing the discretisation of the friction coefficient n_μ, the results clearly show that this parameter affects the detection of the coupling areas. For instance, only one coupling area is detected with five values of μ, while 11 values (or above) cases allow a fine detection of the coupling areas.

Discussions

First, the performance of the algorithm in non-coupling areas are interesting since it allows a good approximation of the extremities of the coalescence graph. This shows that the Bayesian framework is clearly able to deal with these problems when the non-linearities are not too important.

Second, with regard to the coupling areas, the performance is different depending on the complexity of the coupling areas. The first coupling is associated with a rather easy configuration: the third real part eigenvalue has not been coupled with another frequency yet. Conversely, the other couplings are more complex, since the considered eigenvalue switches of coupling frequency.

This increase of complexity proves challenging for the FuzzBO algorithm and is linked to a flaw of the Expected Improvement, already studied in Chapter 5. The Expected Improvement gets stuck in a local optima and wastes iterations of the algorithm trying to escape from this area of the design space.

Nonetheless, Fig. 6.20 clearly shows that the suggested algorithm can determine the full bounds evolution of the coalescence graph. The main difference is that, in this case, the initial training set is more informative, allowing a better convergence. Conversely, Fig. 6.22 presents the mean bounds estimation, showing that, on average, the initial training set has a great impact over the performance of the bounds if compared to Fig. 6.20.

6.6 Chapter outcomes

The purpose of this chapter was to **highlight the limitations of the state-of-the-art methods, used to propagate the uncertainty with the fuzzy set theory**. As shown throughout this

thesis, the evolution of the eigenvalues of a FIV problem are mainly nonlinear. The traditional methods, namely the RTM and GTM, are known in the literature to perform well when the considered functions are monotonous. Conversely, these strategies are quite irrelevant for non-monotonous problems. Moreover, the Transformation Methods (Reduced and Generalised) do not cope well with a high number of dimensions, since they need many samples to be executed.

To reduce the computational cost of evaluating the considered functions, some authors suggested approximating the solver with a surrogate model, and then propagate uncertainty with the fuzzy formalism. Whereas this strategy was shown to be relevant for lightening the computational cost of the determination of the output fuzzy set, the precision of the approximation may reduce the precision of the output fuzzy set representation.

Hence, the major proposal of this chapter is to **combine the Bayesian Optimisation algorithm with the fuzzy formalism**. To do so, the fuzzy parameters are decomposed along the μ axis in α-cuts. For each α-cut, two optimisation problems are solved: one for the minimisation and one for the maximisation; to determine the samples which have the highest probability of being the minimum (resp. maximum) of the considered solver. Then, the Euclidian distance between those samples and the training set is checked before incrementing it. After a certain amount of iterations, the FuzzBO algorithm changes the α-cut level and begins once the incrementing of the training set.

The first experiment proposes a comparison between the RTM, the GP modelling and the suggested FuzzBO algorithm. The results showed that **the FuzzBO algorithm improve the approximation quality of the output fuzzy set**. Indeed, the iterative training set succeeds in lowering the area and support errors to roughly 5%. The static training set, generated by the RTM, performs better than the fully random training set, with a performance of roughly 15% against 20% for the surrogate modelling approach. Table 6.15 summarises the different characteristics of the considered methods.

Table 6.15
Performances of the three tested algorithms.

Method	Flexibility	Computational time	Performance
RTM	+	++++	++
GP	++	++	+
FuzzBO	++++	++	+++

The second experiment assesses the effect of the Bayesian parameters over the performance of the FuzzBO algorithm. Two parameters are studied: the number of iterations and the initial training set size. The results showed that the **estimation of the support and area is better if the number of iterations is high and the initial samples number is kept low** by setting it to $1 \times \mathbf{d}$, d, being the dimensionality of the problem. This observation allows controlling the computational budget of the Bayesian algorithm.

The third experiment tackles the effect of the fuzzy parameter, namely the number of α-cut, to study the performance of the algorithm for coarser and refined fuzzy sets. **The FuzzBO algorithm provides a good estimation of the support and area for all considered α-cut discretisation.**

On this application, these observations show that the FuzzBO algorithm is clearly compatible with industrial workflows, where the acceptable computational budget is around 200 to 300 samples, since it allows a good representativity of the system uncertainties.

Finally, **a pairing issue, which may arises when using the FuzzBO algorithm (or any other parametric study), was studied**. This problem was associated with the sparsity of the samples in coupling areas. **A detection criterion was introduced, relying on the definition of n-coupled modes.** Then, one strategy, hinging on this criterion, was proposed. On average, this **new method was successful, but some limitations have been identified.** Even if the removing strategy works well in the majority of scenarios, some final training sets remain challenging to pair, emphasising that the detection criterion needs to be enhanced. Secondly, **the model used for this application only allowed to study the case of 2-coupled configurations.** Thus, a study on a model with much more degrees of freedom needs to be performed to explore the effects of n-coupled configurations, when n is greater than 2. Since no reference is available, it is not directly possible to determine whether the final training set is well-paired. To compensate for the absence of such a reference, an additional step is considered: namely, a comparison between the traditional pairing and the removal strategies. If this comparison shows that the pairing results are different, then the removal strategy has to be considered; however, if the results are similar, then the pairing results of the traditional strategy is kept. Other solutions involving iterative pairing – first, the 0-coupled configurations are paired; then, the 1-coupled configurations and so on – were investigated, but have not produced relevant results at the time. Nonetheless, it would be interesting to pursue this research to better tackle this pairing issue and integrate the removed samples.

Then, a particular application of the algorithm to approximate the coalescence graph is proposed. The aim is to **assess the efficiency of the suggested algorithm with regard to the coalescence graph.** The first experiment associated with this application showed that the FuzzBO algorithm is a relevant alternative to random sampling strategy with GP. Then, a sensitivity analysis toward the hyperparameter setting was investigated, and the results emphasised that an initial increase of the training set is only needed when the number of iterations is low, as with the approximation of the output fuzzy sets. Finally, the last experiment has brought to light that **considering only the friction coefficient as a fuzzy parameter inevitably leads to difficulty to approximate the coupling areas.** These results showed that the initial training set information has a great impact on the convergence rate.

Conclusions and outlooks

Conclusions

All the works which were carried out throughout this research focused on the developpement of new numerical methods to provide a panel of strategies to handle uncertainty and optimisation of non-linear and non-stationary functions, such as those encountered in the FIV problems. Thus, three objectives were also pursued.

- The **precision of the solution**;
- The **low cost of the solution**;
- The **efficiency of the uncertainty propagation step**.

The first chapter focused on positioning this research with respect to the literature. The traditional and more recent deterministic methods to determine the eigenvalues of a brake system were reviewed. In addition, a focus about the approximation strategies (ROM and surrogate), which are nowadays integrated in uncertainty propagation step, is proposed.

The second chapter considered the QEP which is used to determine the eigenvalues in the FIV ones problem. The most commonly used strategy hinges on a projection on a subspace composed of the normal modes of the conservative problem. Whereas this strategy provides good results in practice for most problems, this strategy is not optimal in FIV ones since the asymmetric contributions (due to the frictional contact) are not integrated in the subspace. Hence, a new strategy, based on homotopy subspace, incremented with modes computed from the asymmetric contributions, is suggested. The results showed an increase of precision on the computation of the eigensolutions of the system. In addition, the suggested method allowed a stabilisation of the eigensolutions values of the system.

The third chapter investigated the performance of the surrogate modelling strategy to reduce the computational cost of the solver. Three surrogate models – the GP, the DNN and the DGP – were studied on non-linear functions. The results emphasised that the GP performs well, but exhibits deceptive predictions for peculiar sample distributions. Interestingly, the use of deep structures can help capturing the complex evolution of these functions and overcoming the deceptive GP predictions.

Nonetheless, for the DNN, many architecture configurations have to be tested to find the most suitable one. Similarly, the precision improvement given by the use of the DGP may not be computationaly interesting, since the time cost of inference process is currently important. Nonetheless, the DGP can be helpful in handling abrupt evolutions, due to the decomposition of the complexity in the deep structure.

The fourth chapter adressed the predictivity issue of the GP emphasised in the previous chapter. This issue, denoted as deceptive, was clearly formulated and showed to be dependent on the sample distribution. Moreover, it was emphasised that an increase (or a decrease) of the samples on which the surrogate is built does not necessarily correct the issue. Then, a detection criterion was derived. From this criterion, a new strategy to overcome the deceptiveness issue was proposed. The results showed that this strategy was efficient to correct and restore the predictivity of the GP on a large benchmark of non-linear and non-stationary functions.

The fifth chapter highlighted an issue with the traditional BO strategy, which hinges on the EI. This issue typically occurs when the studied function owns large attractive local minima and causes the algorithm to be stuck in these local minima. Thus, multiple new samples are needed to escape from these optima. Consequently, the convergence of the algorithm is slowed down and the computational budget to determine the global optimum increases. A new strategy was then suggested to overcome this issue by using multiple acquisition functions, namely the EI and the Variance in this research. The results showed that a multi-infill process and an extraction of the new promising samples from the Pareto front allow reducing the number of samples needed to converge toward the global optimum greatly.

The sixth chapter considered the problem of uncertainty propagation with expensive non-linear functions, described with a fuzzy formalism. The most common way to deal with these problems is to use combinatorial methods, especially the Transformation Method family. Whereas this type of strategies is particularly suited for weak non-linear functions, it was shown to produce non-optimal evaluations of the membership functions due to static combinatorial training set. Hence, the strategy proposed in this chapter relies on an adaptive combinatorial training set, created with the BO strategy, to allow handling the uncertainty propagation for non-linear functions. The results guarantee a building of the fuzzy output membership functions with a higher precision. In addition, the suggested method is more computationally attractive than the traditional combinatorial methods.

This research has thus allowed to mitigate some scientific locks related to the objectives fixed at the beginning of these works. The HOPEP method, suggested in the chapter 2, increased the **precision** on the computation of the eigensolution of the system, as well as the confidence about the results. For the ADT strategy, described in the chapter 4, it counterbalanced the deceptive predictions of the GP, used to approximate the eigensolutions of the problem, allowing a better estimation of these quantities.

With regard to the **low cost of the solution**, the comparison of the surrogate models, carried in the chapter 3, allowed to deduce the most fitting method to reduce the cost of the solver. The MIBO strategy, presented in the chapter 5, also proved to reduce the estimation cost of the solution by providing a faster way to determine the global optimum of the considered solver.

Finally, the FuzzBO algorithm, introduced in the chapter 6, allows a clear reduction of the computational cost to estimate the solutions and **an efficient way to propagate uncertainty.**

Outlooks

Direct improvements of these research works

Some extensions to the different studies developed in this research can obviously be carried out. First of all, the HOPEP algorithm, which was here applied to a single parametric scenario, can be generalised to a multi-parametric scenario as shown in [84]. To reduce the computational cost of carrying out multiple evaluations of different parameter values, the authors suggested updating the projection basis with homotopy eigenvectors computed from specific parameter values. This method has proven to be efficient in estimating the eigensolutions of a FE model with a great number of degrees of freedom for the resolution of the LEP. Hence, an extension of these works to a multi-parametric FIV study would be interesting to investigate the performance of the suggested method.

Recently, Tu et al. [235] used the homotopy perturbation method with a kriging surrogate model to compute the eigensolutions uncertainties of an additively manufactured lattice structure model subjected to linear buckling loads. The surrogate model was used to approximate the participation factors associated to the reduction technique, computed using the Homotopy Perturbation Method. Hence, a coupling of the GP with the HOPEP algorithm would be interesting to investigate. Two strategies can be identified: for the first one, the eigensolutions, computed with the HOPEP algorithm, could simply be approximated with a GP to reduce the computional cost of the multi-parametric study. The second strategy, more intrusive, would rely on the approximation of participation factors for each complex modes. This way is comparable to recent works performed by Denimal et al. [90] on the integration of GP in Polynomial Chaos Expansion (PCE).

For the BO algorithm, two main improvements could be brought to this research. First of all, Sadet et al. [236] showed that the deceptiveness of the GP can affect the performance of the algorithm convergence by increasing the number of iterations needed to find the global optimum. This observation has only been made on the Xiong's function, and an extension of this communication would be interested to pursue for other non-linear functions. With regard to the MIBO strategy, only one type of clustering strategies was investigated, namely the K-Means. The Pareto front of such problems was shown to be piecewise continuous, and the K-Means method produced candidates in the same areas of the objective space. Hence, other strategies as density-based [208] or hierarchical clustering methods [209] could be investigated to generate samples in other areas of the piecewise Pareto front.

Finally, for the FuzzBO algorithm, it would be interested to investigate the performance of the Multi-Infill Bayesian Optimisation strategy for the uncertainty propagation step. In this research, only the traditional method was investigated as preliminary tests with EI and segmentation of the design space, not shown in this sequel, showed interesting results in preventing the EI from being stuck in local minima. Nonetheless, the suggested algorithm in the last chapter still seemed to struggle with the estimation of the last α-cut bounds, by being stuck in a local minimum. By a lack of time, the MIBO strategy has not been applied for this problem, but as this method showed interesting results at escaping local minima, applying it could improve the efficiency of the FuzzBO algorithm. In addition, other strategies for the pairing of the eigensolutions in the case of fuzzy α-cut would be interesting to investigate to allow keeping the pathological configurations. As the pairing procedure does not succeed in pairing them correctly, these configurations cannot be directly taken into account. For instance, evaluations of new configurations at the vicinity of these configurations could allow pairing them correctly. The main complexity would be to define a criterion which allows selecting only configurations that are sufficiently close to those pathological ones, close to well-paired configurations and also not twice-coupled.

Future extensions of this thesis works

This last section focuses on the future extensions of these thesis works. Indeed, the new suggested methods allow exploring new scientific opportunities. First, the suggested methods could be applied to a squeal problem where the instabilities of an industrial FE model of a brake system is considered. This would allow proving the efficiency of this thesis works in reducing the computation cost of the multi-parametric study. In addition, these methods could be applied to the work of Tison et al. [41], where the contact interface would be modelised by random fields. Indeed, the topographical uncertainty seems to be a key parameter to understand the evolution of instabilities in FIV problems.

For the uncertainty propagation strategy, a coupling of the BO algorithm with the PCE could be investigated to improve the organisation of the training sets and certainly reduce the number of samples needed to perform a good approximation. Indeed, the issue of the training set definition, discussed in the sixth chapter for the FuzzBO algorithm, is also present for the PCE.

Finally, a complete study involving a reliability analysis or an optimisation under uncertainty could be carried out to highlight the performance of the suggested FuzzBO strategy.

Appendix A

New benchmark functions suggested in this research

This appendix gives some information (general aspect of the function, location of the minimum, non-linearity, equation) about the new benchmark functions used throughout this thesis.

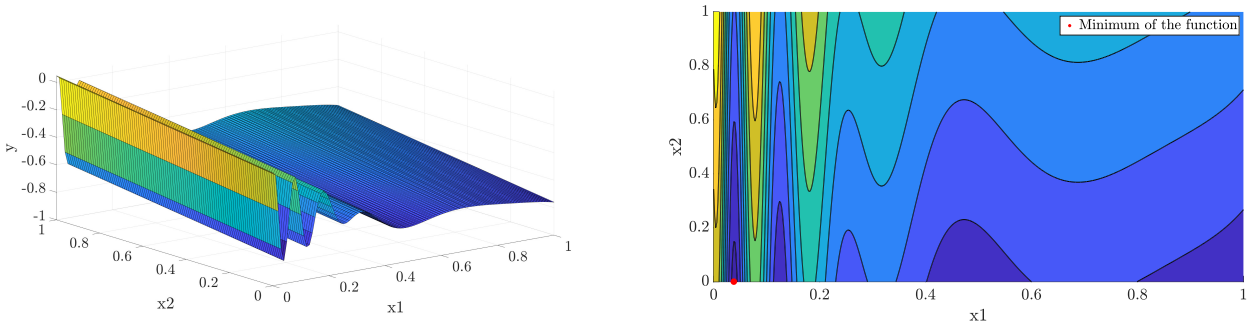
1.1 Function n° 1

This function is based on the Xiong's function and is defined by (Eq. A.1) for $x \in [0;1] \times [0;1]$. The function was constructed to introduce a linear decreasing dimension while keeping the evolution of the Xiong's function on the first dimension. This function aim is to study the performance of the suggested methods in the case of the weakest non-linear behaviour in the second dimension. Indeed, a decreasing dimension usually helps the optimisation algorithm to converge.

$$f(\mathrm{x}) = -0.5\left(\sin\left(40\left(x_1 - 0.85\right)^4\right)\cos\left(2.5\left(x_1 - 0.95\right)\right) + 0.5\left(x_1 - 0.9x_2\right) + 1\right) \tag{A.1}$$

The figures A.1(A) and (B) respectively show the overall behaviour of the function and a contour plot where the minimum of the function is positionned.

The minimum is positionned in the same $x1$ as for the Xiong's function and is equal to -0.833. Nonetheless, two large attractive minimums are located around $(x1, x2) = (0.5, 0)$ and $(x1, x2) = (1, 0)$, which causes the traditional optimisation algorithms to struggle to find the global minimum. In addition, the minimum is located in a narrow valley of the function, complexifying the problem.



(A) Overall evolution (B) Contour plot

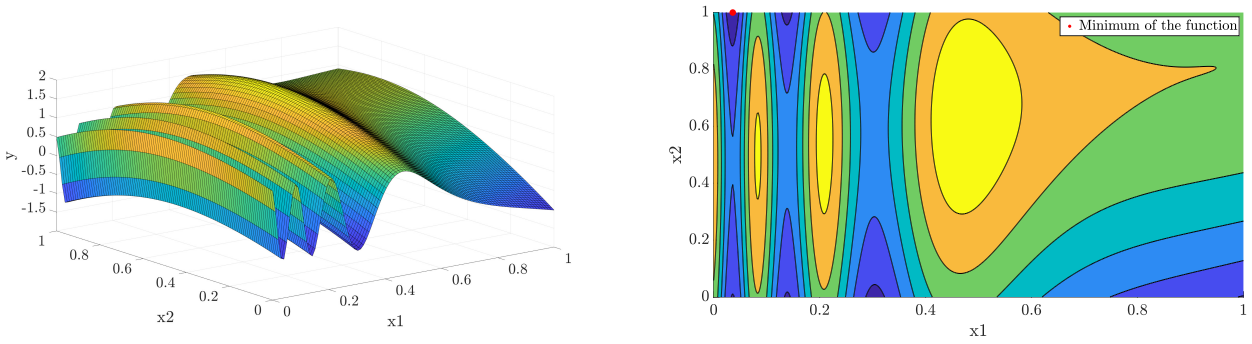
FIGURE A.1: Characteristics of the function n° 1

1.2 Function n° 2

This function is also based on the Xiong's function and is defined by (Eq. A.2) for $x \in [0;1] \times [0;1]$. The function was constructed to introduce a quadratic dimension while keeping the evolution of the Xiong's function on the first dimension. This function aim is to study the performance of the suggested methods in the case of a higher non-linear behaviour in the second dimension. The quadratic dimension greatly complexifies the optimisation problem to solve.

$$f(\mathbf{x}) = \sin\left(20\,(x_1 - 1)^4\right) + \cos^4(x_2 - 0.8) - \left(x_1 - \frac{7}{6}x_2\right)^2 + \frac{1}{2}(x_2 - 0.8)^3 \tag{A.2}$$

The figures A.2(A) and (B) respectively show the overall behaviour of the function and a contour plot where the minimum of the function is positionned.



(A) Overall evolution (B) Contour plot

FIGURE A.2: Characteristics of the function n° 2

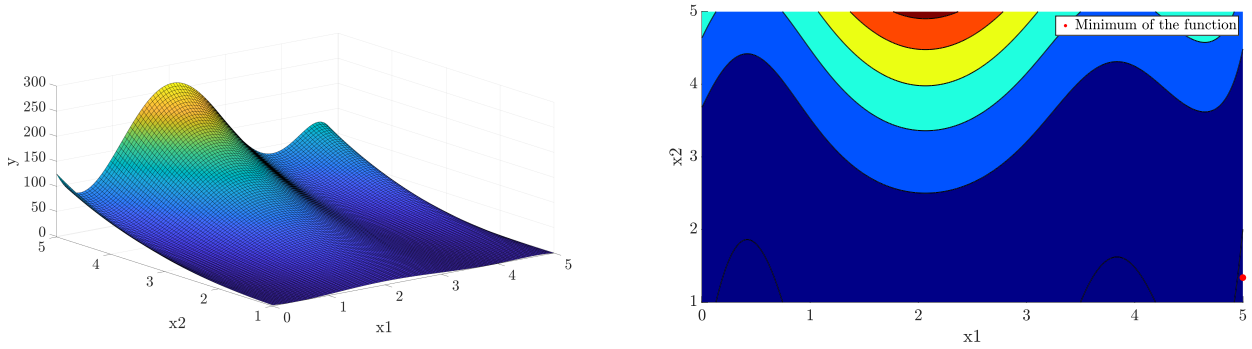
The minimum is equal to -1.352. As for the previous function, two large attractive minimums are located around $(x1, x2) = (0.5, 0)$ and $(x1, x2) = (1, 0)$, which causes the traditional optimisation algorithms to struggle to find the global minimum. In addition, the minimum is also located in a narrow valley of the function, causing a higher complexity of the optimisation problem.

1.3 Function n° 3

This function is a polynomial and is defined by (Eq. A.3) for $x \in [0;5] \times [1;5]$. The function is rather simple compared to the two other functions. This function aim is to highlight the effect of the influence of the large attractive minimums.

$$f(\mathbf{x}) = \left(x_2 - \frac{3107}{333}x_1 + \frac{2839}{230}x_1^2 - \frac{569}{859}x_1^3 - \frac{2850}{1097}x_1^4 + \frac{659}{805}x_1^5 - \frac{177}{2479}x_1^6\right)x_2^2 \tag{A.3}$$

The figures A.2(A) and (B) respectively show the overall behaviour of the function and a contour plot where the minimum of the function is positionned.



(A) Overall evolution (B) Contour plot

FIGURE A.3: Characteristics of the function n° 3

The minimum is equal to -1.189. As for the previous functions, two large attractive minimums are located around $(x1, x2) = (0.12, 0)$ and $(x1, x2) = (0.79, 0)$, which causes the traditional optimisation algorithms to struggle to find the global minimum. In addition, the minimum is also located in the edge of the design space, which has shown to be quite difficult to handle with the traditional BO strategies.

Appendix B

Analytical forms of the ELBO equation (Eq. 3.22)

2.1 Derivation of the expectation of the likelihood in ELBO for a Gaussian likelihood

The expectation of the evidence given the distribution of ϵ^{L_s} is given by for a Gaussian likelihood.

$$\mathbb{E}_{(\epsilon_s^L)}\left[\log\left(f_{(y_s|\mathbf{H}_s^L)}\right)\right] = -\frac{1}{2}\log(2\pi) - \frac{1}{2}\log\left(\sigma_n^2\right) - \frac{1}{2}\left(\frac{(y_s - \tilde{\mu}_s) + \tilde{\Sigma}_s}{\sigma_n^2}\right) \tag{B.1}$$

where μ_s and Σ_s correspond to propagation of a vector $\mathbf{x} \in \mathcal{D}$ through the layers of the DGP.

2.2 Derivation of the Kullback-Leibler divergence in ELBO

For the ℓ^{th} layer, the Kullback-Leibler divergence is given by (Eq. B.2).

$$\begin{aligned} D_{KL}\left(q_{(\mathbf{U}^\ell)}||f_{(\mathbf{U}^\ell)}\right) = \frac{1}{2}\sum_{i=1}^{n}\log\Big(\det\Big(C_{H^\ell,H^\ell}\Big)\Big)) - \frac{1}{2}\sum_{i=1}^{n}\log\Big(\det\Big(\tilde{\Sigma}^\ell\Big)\Big)) \\ - \frac{n}{2} + \frac{1}{2}\text{trace}(C_{H^\ell,H^\ell}^{-1}\tilde{\Sigma}^\ell) + \mathbf{m}^{\ell T} C_{H^\ell,H^\ell}\mathbf{m}^\ell \end{aligned} \tag{B.2}$$

Bibliography

[1] Akay, A.. Acoustics of friction. *The Journal of the Acoustical Society of America* 2002;111(4):1525–1548. doi:10.1121/1.1456514.

[2] Renault, A., Massa, F., Lallemand, B., Tison, T.. Experimental investigations for uncertainty quantification in brake squeal analysis. *Journal of Sound and Vibration* 2016;367:37 – 55. doi:http://dx.doi.org/10.1016/j.jsv.2015.12.049.

[3] Gallina, A., Pichler, L., Uhl, T.. Enhanced meta-modelling technique for analysis of mode crossing, mode veering and mode coalescence in structural dynamics. *Mechanical Systems and Signal Processing* 2011;25(7):2297–2312. doi:10.1016/j.ymssp.2011.02.020.

[4] Ezugwu, A.E., Shukla, A.K., Nath, R., Akinyelu, A.A., Agushaka, J.O., Chiroma, H., *et al.* Metaheuristics: a comprehensive overview and classification along with bibliometric analysis. *Artificial Intelligence Review* 2021;54(6):4237–4316. doi:10.1007/s10462-020-09952-0.

[5] Bahri, O.. A fuzzy framework for multi-objective optimisation under uncertainty. Thèse de doctorat; University of Lille 1; 2017.

[6] Hanss, M.. Applied Fuzzy Arithmetic. Springer-Verlag; 2005. doi:10.1007/b138914.

[7] Cantoni, C., Cesarini, R., Mastinu, G., Rocca, G., Sicigliano, R.. Brake comfort – a review. *Vehicle System Dynamics* 2009;47(8):901–947. doi:10.1080/00423110903100432.

[8] Crolla, D., Lang, A.. Paper VII (i) brake noise and vibration - the state of the art. Dans: *Tribology series*. Elsevier; 1991, p. 165–174. doi:10.1016/s0167-8922(08)70132-9.

[9] Ibrahim, R.A.. Friction-induced vibration, chatter, squeal, and chaos—part II: Dynamics and modeling. *Appl Mech Rev* 1994;47(7):227–253. doi:10.1115/1.3111080.

[10] Kinkaid, N., O'Reilly, O., Papadopoulos, P.. Automotive disc brake squeal. *Journal of Sound and Vibration* 2003;267(1):105 – 166. doi:http://dx.doi.org/10.1016/S0022-460X(02)01573-0.

[11] Nishiwaki, M.. Review of study on brake squeal. *Jsae Review* 1990;11.

[12] Papinniemi, A., Lai, J.C., Zhao, J., Loader, L.. Brake squeal: a literature review. *Applied Acoustics* 2002;63(4):391–400. doi:10.1016/s0003-682x(01)00043-3.

[13] Rashid, A.. Overview of disc brakes and related phenomena - a review. *International Journal of Vehicle Noise and Vibration* 2014;10(4):257. doi:10.1504/ijvnv.2014.065634.

[14] von Wagner, U., Hochlenert, D., Hagedorn, P.. Minimal models for disk brake squeal. *Journal of Sound and Vibration* 2007;302(3):527–539. doi:10.1016/j.jsv.2006.11.023.

[15] Mills, H.. Brake squeal. Dans: *Report No. 9162 B*. Institution of Automotive Engineers; 1938,.

[16] Sinclair, D.. Frictional vibrations. *Journal of Applied Mechanics* 1955;.

[17] Spurr, R.T.. A theory of brake squeal. *Proceedings of the Institution of Mechanical Engineers: Automobile Division* 1961;15(1):33–52. doi:10.1243/pime_auto_1961_000_009_02.

[18] North, M.. Disc brake squeal. Institute of Mechanical Engineers; 1976,.

[19] Millner, N.. An analysis of disc brake squeal. SAE International; 1978,doi:10.4271/780332.

[20] Hultén, J.. Brake squeal - a self-exciting mechanism with constant friction. Dans: *SAE Technical Paper*. SAE International; 1993,doi:10.4271/932965.

[21] Abubakar, A., Ouyang, H.. A prediction methodology of disk brake squeal using complex eigenvalue analysis. *International Journal of Vehicle Design* 2008;46:416–435.

[22] Liu, P., Zheng, H., Cai, C., Wang, Y., Lu, C., Ang, K., *et al.* Analysis of disc brake squeal using the complex eigenvalue method. *Applied Acoustics* 2007;68(6):603–615. doi:10.1016/j.apacoust.2006.03.012.

[23] Belytschko, T., Kennedy, J.. A fluid-structure finite element method for the analysis of reactor safety problems. *Nucl Eng Des* 1976;38(1):71–81. doi:10.1016/0029-5493(76)90087-x.

[24] Dokainish, M., Subbaraj, K.. A survey of direct time-integration methods in computational structural dynamics—i. explicit methods. *Computers & Structures* 1989;32(6):1371–1386. doi:10.1016/0045-7949(89)90314-3.

[25] Hull, T.E., Enright, W.H., Fellen, B.M., Sedgwick, A.E.. Comparing numerical methods for ordinary differential equations. *SIAM J Numer Anal* 1972;9(4):603–637. doi:10.1137/0709052.

[26] Subbaraj, K., Dokainish, M.. A survey of direct time-integration methods in computational structural dynamics—II. implicit methods. *Computers & Structures* 1989;32(6):1387–1401. doi:10.1016/0045-7949(89)90315-5.

[27] des Roches, G.V.. Frequency and time simulation of squeal instabilities. application to the design of industrial automotive brakes. Thèse de doctorat; Ecole Centrale Paris; 2011.

[28] Blaschke, P., Tan, M., Wang, A.. On the analysis of brake squeal propensity using finite element method. Dans: *SAE Technical Paper Series*. SAE International; 2000,doi:10.4271/2000-01-2765.

[29] AbuBakar, A.R., Ouyang, H.. Complex eigenvalue analysis and dynamic transient analysis in predicting disc brake squeal. *International Journal of Vehicle Noise and Vibration* 2006;2(2):143 – 155. doi:http://dx.doi.org/10.1504/IJVNV.2006.011051.

[30] Massi, F., Baillet, L., Giannini, O., Sestieri, A.. Brake squeal: Linear and nonlinear numerical approaches. *Mechanical Systems and Signal Processing* 2007;21(6):2374–2393. doi:10.1016/j.ymssp.2006.12.008.

[31] Oberst, S., Lai, J.. Nonlinear transient and chaotic interactions in disc brake squeal. *Journal of Sound and Vibration* 2015;342:272–289. doi:10.1016/j.jsv.2015.01.005.

[32] Sinou, J.J.. Transient non-linear dynamic analysis of automotive disc brake squeal – on the need to consider both stability and non-linear analysis. *Mechanics Research Communications* 2010;37(1):96–105. doi:https://doi.org/10.1016/j.mechrescom.2009.09.002.

[33] Charroyer, L., Chiello, O., Sinou, J.J.. Self-excited vibrations of a non-smooth contact dynamical system with planar friction based on the shooting method. *International Journal of Mechanical Sciences* 2018;144:90–101. doi:10.1016/j.ijmecsci.2018.05.045.

[34] Denimal, E.. Prédiction des instabilités de frottement par méta-modélisation et approches fréquentielles: Application au crissement de frein automobile. Thèse de doctorat; Ecole Centrale de Lyon; 2018.

[35] Loyer, A.. Etude numerique et experimentale du crissement des systèmes de freinage ferroviaires. Thèse de doctorat; Ecole Centrale Lyon; 2012.

[36] Coudeyras, N., Sinou, J.J., Nacivet, S.. A new treatment for predicting the self-excited vibrations of nonlinear systems with frictional interfaces: The constrained harmonic balance method, with application to disc brake squeal. *Journal of Sound and Vibration* 2009;319(3-5):1175–1199. doi:10.1016/j.jsv.2008.06.050.

[37] Coudeyras, N., Nacivet, S., Sinou, J.J.. Periodic and quasi-periodic solutions for multi-instabilities involved in brake squeal. *Journal of Sound and Vibration* 2009;328(4-5):520–540. doi:10.1016/j.jsv.2009.08.017.

[38] Nacivet, S., Sinou, J.J.. Modal amplitude stability analysis and its application to brake squeal. *Applied Acoustics* 2017;116:127–138. doi:https://doi.org/10.1016/j.apacoust.2016.09.010.

[39] Denimal, E., Sinou, J.J., Nacivet, S.. Generalized modal amplitude stability analysis for the prediction of the nonlinear dynamic response of mechanical systems subjected

to friction-induced vibrations. *Nonlinear Dynamics* 2020;100(4):3121–3144. doi:10.1007/s11071-020-05627-1.

[40] Denimal, E., Sinou, J.J., Nacivet, S.. Prediction and analysis of quasi-periodic solution for friction-induced vibration of an industrial brake system with the generalized modal amplitude stability analysis. *Journal of Sound and Vibration* 2021;506:116164. doi:10.1016/j.jsv.2021.116164.

[41] Tison, T., Heussaff, A., Massa, F., Turpin, I., Nunes, R.. Improvement in the predictivity of squeal simulations: Uncertainty and robustness. *Journal of Sound and Vibration* 2014;333(15):3394 – 3412.

[42] Balmès, E.. Parametric families of reduced finite element models. theory and applications. *Mechanical Systems and Signal Processing* 1996;10(4):381 – 394. doi:https://doi.org/10.1006/mssp.1996.0027.

[43] Loyer, A., Sinou, J.J., Chiello, O., Lorang, X.. Study of nonlinear behaviors and modal reductions for friction destabilized systems. application to an elastic layer. *Journal of Sound and Vibration* 2012;331(5):1011–1041. doi:10.1016/j.jsv.2011.10.018.

[44] Do, H., Massa, F., Tison, T., Lallemand, B.. A global strategy for the stability analysis of friction induced vibration problem with parameter variations. *Mechanical Systems and Signal Processing* 2017;84:346–364. doi:10.1016/j.ymssp.2016.07.029.

[45] Lai, V.V., Chiello, O., Brunel, J.F., Dufrénoy, P.. Full finite element models and reduction strategies for the simulation of friction-induced vibrations of rolling contact systems. *Journal of Sound and Vibration* 2019;444:197–215. doi:10.1016/j.jsv.2018.12.024.

[46] Besset, S., Sinou, J.J.. Modal reduction of brake squeal systems using complex interface modes. *Mechanical Systems and Signal Processing* 2017;85:896–911. doi:10.1016/j.ymssp.2016.09.006.

[47] Monteil, M., Besset, S., Sinou, J.J.. A double modal synthesis approach for brake squeal prediction. *Mechanical Systems and Signal Processing* 2016;70-71:1073–1084. doi:10.1016/j.ymssp.2015.07.023.

[48] Massa, F., Turpin, I., Tison, T.. From homotopy perturbation technique to reduced order model for multiparametric modal analysis of large finite element models. *Mechanical Systems and Signal Processing* 2017;96:291 – 302. doi:https://doi.org/10.1016/j.ymssp.2017.04.025.

[49] Lü, H., Yu, D.. Brake squeal reduction of vehicle disc brake system with interval parameters by uncertain optimization. *Journal of Sound and Vibration* 2014;333(26):7313–7325. doi:https://doi.org/10.1016/j.jsv.2014.08.027.

[50] Treimer, M., Allert, B., Dylla, K., Müller, G.. Uncertainty quantification applied to the mode coupling phenomenon. *Journal of Sound and Vibration* 2017;388:171–187. doi:10.1016/j.jsv.2016.10.019.

[51] Denimal, E., Nechak, L., Sinou, J., Nacivet, S.. Kriging surrogate models for predicting the complex eigenvalues of mechanical systems subjected to friction-induced vibration. *Shock and Vibration* 2016;2016.

[52] Nechak, L., Gillot, F., Besset, S., Sinou, J.J.. Sensitivity analysis and kriging based models for robust stability analysis of brake systems. *Mechanics Research Communications* 2015;69:136 – 145. doi:http://dx.doi.org/10.1016/j.mechrescom.2015.08.001.

[53] Nobari, A., Ouyang, H., Bannister, P.. Uncertainty quantification of squeal instability via surrogate modelling. *Mechanical Systems and Signal Processing* 2015;60-61:887–908. doi:10.1016/j.ymssp.2015.01.022.

[54] Stender, M., Tiedemann, M., Spieler, D., Schoepflin, D., Hoffmann, N., Oberst, S.. Deep learning for brake squeal: vibration detection, characterization and prediction. *arXiv:200101596 [cs, eess]* 2020;ArXiv: 2001.01596.

[55] Qiao, S., Ibrahim, R.. Stochastic dynamics of systems with friction-induced vibration. *Journal of Sound and Vibration* 1999;223(1):115–140. doi:10.1006/jsvi.1998.2099.

[56] Ibrahim, R., Madhavan, S., Qiao, S., Chang, W.. Experimental investigation of friction-induced noise in disc brake systems. *International Journal of Vehicle Design* 2000;23(3/4):218. doi:10.1504/ijvd.2000.001893.

[57] Oberst, S., Lai, J.. Statistical analysis of brake squeal noise. *Journal of Sound and Vibration* 2011;330(12):2978–2994. doi:10.1016/j.jsv.2010.12.021.

[58] Oberst, S., Lai, J.. A statistical approach to estimate the lyapunov spectrum in disc brake squeal. *Journal of Sound and Vibration* 2015;334:120–135. doi:10.1016/j.jsv.2014.06.025.

[59] Oberst, S., Lai, J.. Chaos in brake squeal noise. *Journal of Sound and Vibration* 2011;330(5):955–975. doi:10.1016/j.jsv.2010.09.009.

[60] Zhang, Z., Oberst, S., Lai, J.C.. On the potential of uncertainty analysis for prediction of brake squeal propensity. *Journal of Sound and Vibration* 2016;377:123–132. doi:10.1016/j.jsv.2016.05.023.

[61] Ouyang, H., Nack, W., Yuan, Y., Chen, F.. Numerical analysis of automotive disc brake squeal: a review. *International Journal of Vehicle Noise and Vibration* 2005;1(3/4):207. doi:10.1504/ijvnv.2005.007524.

[62] Metropolis, N., Ulam, S.. The monte carlo method. *Journal of the American Statistical Association* 1949;44(247):335–341. doi:10.1080/01621459.1949.10483310.

[63] Culla, A., Massi, F.. Uncertainty model for contact instability prediction. *The Journal of the Acoustical Society of America* 2009;126(3):1111–1119. doi:10.1121/1.3183376.

[64] Soize, C.. A nonparametric model of random uncertainties for reduced matrix models in structural dynamics. *Probabilistic Engineering Mechanics* 2000;15(3):277–294. doi:10.1016/s0266-8920(99)00028-4.

[65] Butlin, T., Woodhouse, J.. Friction-induced vibration: Should low-order models be believed? *Journal of Sound and Vibration* 2009;328(1-2):92–108. doi:10.1016/j.jsv.2009.08.001.

[66] Ghanem, R., P., S.. Stochastic Finite Elements: A Spectral Approach. Springer; 1991.

[67] Sarrouy, E., Dessombz, O., Sinou, J.J.. Piecewise polynomial chaos expansion with an application to brake squeal of a linear brake system. *Journal of Sound and Vibration* 2013;332(3):577–594. doi:10.1016/j.jsv.2012.09.009.

[68] Sarrouy, E., Dessombz, O., Sinou, J.J.. Stochastic study of a non-linear self-excited system with friction. *European Journal of Mechanics - A/Solids* 2013;40:1–10. doi:10.1016/j.euromechsol.2012.12.003.

[69] Zhang, Z., Oberst, S., Lai, J.C.S.. Instability prediction of brake squeal by nonlinear stability analysis. Dans: *Inter.noise 2014*. Inter.noise 2014; 2014,.

[70] Nechak, L., Berger, S., Aubry, E.. Non-intrusive generalized polynomial chaos for the robust stability analysis of uncertain nonlinear dynamic friction systems. *Journal of Sound and Vibration* 2013;332(5):1204–1215. doi:10.1016/j.jsv.2012.09.046.

[71] Nechak, L., Besset, S., Sinou, J.J.. Robustness of stochastic expansions for the stability of uncertain nonlinear dynamical systems – application to brake squeal. *Mechanical Systems and Signal Processing* 2018;111:194–209. doi:10.1016/j.ymssp.2018.01.021.

[72] Heussaff, A., Dubar, L., Tison, T., Watremez, M., Nunes, R.. A methodology for the modelling of the variability of brake lining surfaces. *Wear* 2012;289:145–159. doi:10.1016/j.wear.2012.04.002.

[73] Lü, H., Shangguan, W.B., Yu, D.. An imprecise probability approach for squeal instability analysis based on evidence theory. *Journal of Sound and Vibration* 2017;387:96–113. doi:10.1016/j.jsv.2016.10.001.

[74] Dempster, A.P.. Upper and lower probabilities induced by a multivalued mapping. *The Annals of Mathematical Statistics* 1967;38(2):325–339. doi:10.1214/aoms/1177698950.

[75] Shafer, G.. A Mathematical Theory of Evidence. Princeton University Press; 1976.

[76] Moore, R.. Interval Analysis. Prentice Hall, Englewood Cliff; 1966.

[77] Zadeh, L.. The concept of a linguistic variable and its application to approximate reasoning—I. *Information Sciences* 1975;8(3):199–249. doi:10.1016/0020-0255(75)90036-5.

[78] Zadeh, L.. The concept of a linguistic variable and its application to approximate reasoning—II. *Information Sciences* 1975;8(4):301–357. doi:10.1016/0020-0255(75)90046-8.

[79] Zadeh, L.. The concept of a linguistic variable and its application to approximate reasoning-III. *Information Sciences* 1975;9(1):43–80. doi:10.1016/0020-0255(75)90017-1.

[80] Hanss, M.. The transformation method for the simulation and analysis of systems with uncertain parameters. *Fuzzy Sets and Systems* 2002;130(3):277–289. doi:10.1016/s0165-0114(02)00045-3.

[81] Gauger, U., Hanss, M., Gaul, L.. On the inclusion of uncertain parameters in brake squeal analysis. Dans: *IMAC-XXIV: Conference & Exposition on Structural Dynamics, St. Louis, MO, January*. 2006,.

[82] Hanselowski, A., Hanss, M.. Uncertainty and sensitivity analysis of different models of brake squeal. *PAMM* 2014;14(1):277–278. doi:10.1002/pamm.201410126.

[83] Massa, F., Do, H.Q., Tison, T., Cazier, O.. Uncertain friction-induced vibration study: Coupling of fuzzy logic, fuzzy sets, and interval theories. *ASCE-ASME J Risk and Uncert in Engrg Sys Part B Mech Engrg* 2015;2(1). doi:10.1115/1.4030469.

[84] Massa, F., Lallemand, B., Tison, T.. Multi-level homotopy perturbation and projection techniques for the reanalysis of quadratic eigenvalue problems: The application of stability analysis. *Mechanical Systems and Signal Processing* 2015;52-53:88 – 104.

[85] Lü, H., Shangguan, W.B., Yu, D.. Uncertainty quantification of squeal instability under two fuzzy-interval cases. *Fuzzy Sets and Systems* 2017;328:70–82. doi:10.1016/j.fss.2017.07.006.

[86] Lü, H., Cai, Z., Feng, Q., Shangguan, W.B., Yu, D.. An improved method for fuzzy–interval uncertainty analysis and its application in brake instability study. *Computer Methods in Applied Mechanics and Engineering* 2018;342:142–160. doi:10.1016/j.cma.2018.07.028.

[87] Lü, H., Yu, D.. Stability Analysis and Improvement of Uncertain Disk Brake Systems With Random and Interval Parameters for Squeal Reduction. *Journal of Vibration and Acoustics* 2015;137(5). doi:10.1115/1.4030044. arXiv:https://asmedigitalcollection.asme.org/vibrationacoustics/article-pdf/137/5/051003/ 051003.

[88] Lü, H., Shangguan, W.B., Yu, D.. A unified approach for squeal instability analysis of disc brakes with two types of random-fuzzy uncertainties. *Mechanical Systems and Signal Processing* 2017;93:281–298. doi:10.1016/j.ymssp.2017.02.012.

[89] Lü, H., Shangguan, W.B., Yu, D.. A new hybrid uncertainty analysis method and its application to squeal analysis with random and interval variables. *Probabilistic Engineering Mechanics* 2018;51:1–10. doi:10.1016/j.probengmech.2017.11.001.

[90] Denimal, E., Nechak, L., Sinou, J.J., Nacivet, S.. A new surrogate modeling method associating generalized polynomial chaos expansion and kriging for mechanical systems subjected to friction-induced vibration. Dans: *Special Topics in Structural Dynamics, Volume 6*. Springer International Publishing; 2017, p. 17–23. doi:10.1007/978-3-319-53841-9_2.

[91] Denimal, E., Nechak, L., Sinou, J.J., Nacivet, S.. A novel hybrid surrogate model and its application on a mechanical system subjected to friction-induced vibration. *Journal of Sound and Vibration* 2018;434:456–474. doi:10.1016/j.jsv.2017.08.005.

[92] Seshu, P.. Substructuring and component mode synthesis. *Shock and Vibration* 1997;4(3):199–210. doi:10.1155/1997/147513.

[93] Hou, Z., Chen, S.. Iterated dynamic condensation technique and its applications in modal testing. *Shock and Vibration* 1997;4(3):143–151. doi:10.1155/1997/738784.

[94] Level, P., Gallo, Y., Tison, T., Ravalard, Y.. On an extension of classical modal reanalysis algorithms: the improvement of initial models. *Journal of Sound and Vibration* 1995;186(4):551–560. doi:10.1006/jsvi.1995.0467.

[95] Balmès, E.. Parametric families of reduced finite element models. theory and applications. *Mechanical Systems and Signal Processing* 1996;10(4):381 – 394. doi:https://doi.org/10.1006/mssp.1996.0027.

[96] Massa, F., Tison, T., Lallemand, B., Cazier, O.. Structural modal reanalysis methods using homotopy perturbation and projection techniques. *Computer Methods in Applied Mechanics and Engineering* 2011;200(45):2971 – 2982. doi:https://doi.org/10.1016/j.cma.2011.06.016.

[97] Chen, S., Yang, X., Lian, H.. Comparison of several eigenvalue reanalysis methods for modified structures. *Structural and Multidisciplinary Optimization* 2000;20:253 – 259.

[98] Kirsch, U., Bogomolni, M., Sheinman, I.. Nonlinear dynamic reanalysis of structures by combined approximations. *Computer Methods in Applied Mechanics and Engineering* 2006;195(33):4420 – 4432. doi:https://doi.org/10.1016/j.cma.2005.09.013.

[99] Ma, L., Chen, S.H., Meng, G.W.. Combined approximation for reanalysis of complex eigenvalues. *Computers and Structures* 2009;87(7):502 – 506. doi:https://doi.org/10.1016/j.compstruc.2009.01.009.

[100] Jian-jun, H., Xiang-zi, C., Bin, X.. Structural modal reanalysis for large, simultaneous and multiple type modifications. *Mechanical Systems and Signal Processing* 2015;62-63:207 – 217. doi:https://doi.org/10.1016/j.ymssp.2015.03.019.

[101] Damil, N., Potier-Ferry, M., Najah, A., Chari, R., Lahmam, H.. An iterative method based upon Padé approximants. *Communications in Numerical Methods in Engineering* 1999;15(10):701–708. doi:10.1002/(SICI)1099-0887(199910)15:10<701::AID-CNM283>3.0.CO;2-L.

[102] Elhage-Hussein, A., Potier-Ferry, M., Damil, N.. A numerical continuation method based on Padé approximants. *International Journal of Solids and Structures* 2000;37(46):6981 – 7001. doi:https://doi.org/10.1016/S0020-7683(99)00323-6.

[103] He, J.H.. Homotopy perturbation technique. *Computer Methods in Applied Mechanics and Engineering* 1999;178(3):257 – 262. doi:https://doi.org/10.1016/S0045-7825(99)00018-3.

[104] He, J.H.. A coupling method of a homotopy technique and a perturbation technique for non-linear problems. *International Journal of Non-linear Mechanics* 2000;35:37–43.

[105] He, J.H.. New interpretation of homotopy perturbation method. *International Journal of Modern Physics B* 2006;20.

[106] He, J.H.. Homotopy perturbation method with two expanding parameters. *Indian Journal of Physics* 2014;88(2):193–196.

[107] Yu, D.N., He, J.H., Garcia, A.G.. Homotopy perturbation method with an auxiliary parameter for nonlinear oscillators. *Journal of Low Frequency Noise, Vibration and Active Control* 2019;38(3-4):1540–1554.

[108] Duigou, L., Daya, E.M., Potier-Ferry, M.. Iterative algorithms for non-linear eigenvalue problems. application to vibrations of viscoelastic shells. *Computer Methods in Applied Mechanics and Engineering* 2003;192(11):1323 – 1335. doi:https://doi.org/10.1016/S0045-7825(02)00641-2.

[109] Boumediene, F., Cadou, J.M., Duigou, L., Daya, E.M.. A reduction model for eigensolutions of damped viscoelastic sandwich structures. *Mechanics Research Communications* 2014;57:74 – 81. doi:https://doi.org/10.1016/j.mechrescom.2014.03.001.

[110] Sun, X., Du, L., Yang, V.. A homotopy method for determining the eigenvalues of locally or non-locally reacting acoustic liners in flow ducts. *Journal of Sound and Vibration* 2007;303(1):277 – 286. doi:https://doi.org/10.1016/j.jsv.2007.01.020.

[111] Lee, M.K., Fouladi, M.H., Namasivayam, S.N.. Natural frequencies of thin rectangular plates using homotopy-perturbation method. *Applied Mathematical Modelling* 2017;50:524 – 543. doi:https://doi.org/10.1016/j.apm.2017.05.050.

[112] Sliva, G., Brezillon, A., Cadou, J., Duigou, L.. A study of the eigenvalue sensitivity by homotopy and perturbation methods. *Journal of Computational and Applied Mathematics* 2010;234(7):2297 – 2302. doi:https://doi.org/10.1016/j.cam.2009.08.086; fourth International Conference on Advanced COmputational Methods in ENgineering (ACOMEN 2008).

[113] Li, L., Hu, Y.J., Wang, X.L.. Modal modification of damped asymmetric systems without using the left eigenvectors. Dans: *Mechanical Design and Power Engineering*; vol. 490 of *Applied Mechanics and Materials*. Trans Tech Publications Ltd; 2014, p. 331–335. doi:10.4028/www.scientific.net/AMM.490-491.331.

[114] Massa, F., Lallemand, B., Tison, T.. Multi-level homotopy perturbation and projection techniques for the reanalysis of quadratic eigenvalue problems: The application of stability analysis. *Mechanical Systems and Signal Processing* 2015;52-53:88 – 104. doi:https://doi.org/10.1016/j.ymssp.2014.07.013.

[115] Do, H., Massa, F., Tison, T., Lallemand, B.. A global strategy for the stability analysis of friction induced vibration problem with parameter variations. *Mechanical Systems and Signal Processing* 2017;84:346 – 364. doi:https://doi.org/10.1016/j.ymssp.2016.07.029.

[116] Tisseur, F., Meerbergen, K.. The quadratic eigenvalue problem. *SIAM Review* 2001;43(2):235–286. doi:10.1137/s0036144500381988.

[117] Golub, G., Van Loan, C.. Matrix Computations. JHU Press; 1996.

[118] Wang, B.P.. Improved approximate methods for computing eigenvector derivatives in structural dynamics. *AIAA Journal* 1991;29(6):1018–1020. doi:10.2514/3.59945. arXiv:https://doi.org/10.2514/3.59945.

[119] Fuellekrug, U.. Computation of real normal modes from complex eigenvectors. *Mechanical Systems and Signal Processing* 2008;22(1):57 – 65. doi:https://doi.org/10.1016/j.ymssp.2007.07.009.

[120] Jamai, R., Damil, N.. Influence of iterated Gram–Schmidt orthonormalization in the asymptotic numerical method. *Comptes Rendus Mécanique* 2003;331(5):351 – 356. doi:https://doi.org/10.1016/S1631-0721(03)00072-X.

[121] Balmès, E.. Structural Dynamics Toolbox (for use with MATLAB). February 24, 2019.

[122] Pastor, M., Binda, M., Harčarik, T.. Modal assurance criterion. *Procedia Engineering* 2012;48:543–548. doi:10.1016/j.proeng.2012.09.551.

[123] Lü, H., Yu, D.. Optimization design of a disc brake system with hybrid uncertainties. *Advances in Engineering Software* 2016;98:112 – 122. doi:https://doi.org/10.1016/j.advengsoft.2016.04.009.

[124] Lü, H., Cai, Z., Feng, Q., Shangguan, W.B., Yu, D.. An improved method for fuzzy–interval uncertainty analysis and its application in brake instability study. *Computer Methods in Applied Mechanics and Engineering* 2018;342:142 – 160. doi:https://doi.org/10.1016/j.cma.2018.07.028.

[125] Do, H., Massa, F., Tison, T., Lallemand, B.. A global strategy for the stability analysis of friction induced vibration problem with parameter variations. *Mechanical Systems and Signal Processing* 2017;84:346 – 364. doi:https://doi.org/10.1016/j.ymssp.2016.07.029.

[126] Monteil, M., Besset, S., Sinou, J.J.. A double modal synthesis approach for brake squeal prediction. *Mechanical Systems and Signal Processing* 2016;70–71:1073 – 1084. doi:http://dx.doi.org/10.1016/j.ymssp.2015.07.023.

[127] Denimal, E., Sinou, J.J., Nacivet, S.. Influence of structural modifications of automotive brake systems for squeal events with kriging meta-modelling method. *Journal of Sound and Vibration* 2019;463:114938. doi:https://doi.org/10.1016/j.jsv.2019.114938.

[128] Schmidhuber, J.. Deep learning in neural networks: An overview. *Neural Networks* 2015;61:85–117. doi:https://doi.org/10.1016/j.neunet.2014.09.003.

[129] Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y., Alsaadi, F.E.. A survey of deep neural network architectures and their applications. *Neurocomputing* 2017;234:11–26. doi:https://doi.org/10.1016/j.neucom.2016.12.038.

[130] Talbi, E.G.. Automated design of deep neural networks: A survey and unified taxonomy. *ACM Comput Surv* 2021;54(2). doi:10.1145/3439730.

[131] Damianou, A., Lawrence, N.. Deep Gaussian processes. Dans: Carvalho, C., Ravikumar, P., *éditeurs*. Proceedings of the Sixteenth International Workshop on Artificial Intelligence and Statistics (AISTATS). AISTATS '13; JMLR W&CP 31; 2013, p. 207–215.

[132] Salimbeni, H., Deisenroth, M.. Doubly stochastic variational inference for deep gaussian processes. Dans: *Advances in Neural Information Processing Systems*. 2017,.

[133] Tripathy, R.K., Bilionis, I.. Deep uq: Learning deep neural network surrogate models for high dimensional uncertainty quantification. *Journal of Computational Physics* 2018;375:565–588. doi:https://doi.org/10.1016/j.jcp.2018.08.036.

[134] Zhu, Y., Zabaras, N.. Bayesian deep convolutional encoder–decoder networks for surrogate modeling and uncertainty quantification. *Journal of Computational Physics* 2018;366:415–447. doi:https://doi.org/10.1016/j.jcp.2018.04.018.

[135] Hebbal, A., Brevault, L., Balesdent, M., Talbi, E.G., Melab, N.. Bayesian Optimization using Deep Gaussian Processes. *arXiv:190503350 [cs, stat]* 2019;ArXiv: 1905.03350.

[136] Kong, Y., Abdullah, S., Schramm, D., Omar, M., Haris, S.. Optimization of spring fatigue life prediction model for vehicle ride using hybrid multi-layer perceptron artificial neural networks. *Mechanical Systems and Signal Processing* 2019;122:597–621. doi:10.1016/j.ymssp.2018.12.046.

[137] Radaideh, M.I., Kozlowski, T.. Surrogate modeling of advanced computer simulations using deep Gaussian processes. *Reliability Engineering & System Safety* 2020;195:106731. doi:10.1016/j.ress.2019.106731.

[138] Tagade, P., Hariharan, K.S., Ramachandran, S., Khandelwal, A., Naha, A., Kolake, S.M., *et al.* Deep Gaussian process regression for lithium-ion battery health prognosis and degradation mode diagnosis. *Journal of Power Sources* 2020;445:227281. doi:10.1016/j.jpowsour.2019.227281.

[139] Hultén, J.. Some drum brake squeal mechanisms. SAE International; 1995,doi:10.4271/951280.

[140] Mckay, M.D., Beckman, R.J., Conover, W.J.. A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output From a Computer Code. *Technometrics* 2000;42(1):55–61. doi:10.1080/00401706.2000.10485979.

[141] Rasmussen, C.E., Williams, C.K.I.. Gaussian processes for machine learning. Adaptive computation and machine learning; 3. print ed.; Cambridge, Mass.: MIT Press; 2008. ISBN 978-0-262-18253-9. OCLC: 552376743.

[142] Basheer, I., Hajmeer, M.. Artificial neural networks: fundamentals, computing, design, and application. *Journal of Microbiological Methods* 2000;43(1):3–31. doi:10.1016/S0167-7012(00)00201-3.

[143] Rosenblatt, F.. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review* 1958;65(6):386–408. doi:10.1037/h0042519; place: US Publisher: American Psychological Association.

[144] Hecht-Nielsen, . Theory of the backpropagation neural network. Dans: *International 1989 Joint Conference on Neural Networks*. 1989, p. 593–605 vol.1.

[145] Runge, V.C.. Über empirische funktionen und die interpolation zwischen äquidistanten ordinaten. *Zeitschrift für Mathematik und Physik* 1901;:224 – 243.

[146] Byrdt, R.H., Lut, P., Nocedalt, J.. A Limited Memory Algorithm for Bound Constrained Optimization. *SIAM Journal on Scientific Computing* 1995;:19.

[147] Zhu, C., Byrd, R.H., Lu, P., Nocedal, J.. Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization. *ACM Trans Math Softw* 1997;23(4):550–560. doi:10.1145/279232.279236.

[148] Morales, J.L., Nocedal, J.. Remark on "algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound constrained optimization". *ACM Trans Math Softw* 2011;38(1):1–4. doi:10.1145/2049662.2049669.

[149] Dai, Z., Damianou, A., González, J., Lawrence, N.. Variational Auto-encoded Deep Gaussian Processes. *arXiv:151106455 [cs, stat]* 2016;ArXiv: 1511.06455.

[150] Quinonero-Candela, J., Rasmussen, C.E.. A unifying view of Sparse Approximate Gaussian Process Regression. *Journal of Machine Learning Research* 2005;Vol. 6:21.

[151] Jensen, J.L.W.V.. Sur les fonctions convexes et les inégualités entre les valeurs Moyennes. *Acta Mathematica* 1906;doi:10.1007/bf02418571.

[152] Rezende, D.J., Mohamed, S., Wierstra, D.. Stochastic backpropagation and approximate inference in deep generative models. Dans: *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*. ICML'14; JMLR.org; 2014, p. II–1278–II–1286.

[153] Kingma, D.P., Salimans, T., Welling, M.. Variational dropout and the local reparameterization trick. Dans: *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*. NIPS'15; Cambridge, MA, USA: MIT Press; 2015, p. 2575–2583.

[154] Matthews, A.G.d.G., van der Wilk, M., Nickson, T., Fujii, K., Boukouvalas, A., León-Villagrá, P., *et al.* GPflow: A Gaussian process library using TensorFlow. *Journal of Machine Learning Research* 2017;18.

[155] Kingma, D.P., Ba, J.. Adam: A method for stochastic optimization. Dans: Bengio, Y., LeCun, Y., *éditeurs*. 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings. 2015,.

[156] Salimbeni, H., Eleftheriadis, S., Hensman, J.. Natural gradients in practice: Non-conjugate variational inference in gaussian process models. Dans: *Artificial Intelligence and Statistics*. 2018,.

[157] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., *et al.* Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 2011;12:2825–2830.

[158] Ginsbourger, D., Dupuy, D., Badea, A., Carraro, L., Roustant, O.. A note on the choice and the estimation of Kriging models for the analysis of deterministic computer experiments. *Appl Stochastic Models Bus Ind* 2009;25(2):115–131. doi:10.1002/asmb.741.

[159] Piotrowski, A.P., Napiorkowski, J.J.. A comparison of methods to avoid overfitting in neural networks training in the case of catchment runoff modelling. *Journal of Hydrology* 2013;476:97 – 111. doi:https://doi.org/10.1016/j.jhydrol.2012.10.019.

[160] Zur, R.M., Jiang, Y., Pesce, L.L., Drukker, K.. Noise injection for training artificial neural networks: A comparison with weight decay and early stopping. *Medical Physics* 2009;36(10):4810–4818. doi:https://doi.org/10.1118/1.3213517. arXiv:https://aapm.onlinelibrary.wiley.com/doi/pdf/10.1118/1.3213517.

[161] Sietsma, J., Dow, R.J.. Creating artificial neural networks that generalize. *Neural Networks* 1991;4(1):67 – 79. doi:https://doi.org/10.1016/0893-6080(91)90033-2.

[162] Jones, D.R., Schonlau, M., Welch, W.J.. Efficient global optimization of expensive black-box functions. *Journal of Global optimization* 1998;13(4):455–492.

[163] Jones, D.R.. A Taxonomy of Global Optimization Methods Based on Response Surfaces. *Journal of Global Optimization* 2001;21(4):345–383. doi:10.1023/A:1012771025575.

[164] Forrester, A., Jones, D.. Global optimization of deceptive functions with sparse sampling. Dans: *AIAA 2008*. ISBN 978-1-60086-982-2; 2008,.

[165] Benassi, R., Bect, J., Vazquez, E.. Robust Gaussian process-based global optimization using a fully Bayesian expected improvement criterion. Dans: *5th International Conference on Learning and Intelligent Optimization (LION 5)*; vol. 6683 of *Lecture Notes in Computer Science*. Rome, Italy: Springer; 2011, p. 176–190. doi:10.1007/978-3-642-25566-3_13.

[166] Li, R., Sudjianto, A.. Analysis of computer experiments using penalized likelihood in gaussian kriging models. *Technometrics* 2005;47(2):111–120.

[167] Hebbal, A., Brevault, L., Balesdent, M., Talbi, E.G., Melab, N.. Bayesian optimization using deep gaussian processes with applications to aerospace system design. *Optimization and Engineering* 2021;.

[168] Lizotte, D.J., Greiner, R., Schuurmans, D.. An experimental methodology for response surface optimization methods. *Journal of Global Optimization* 2012;53(4):699–736.

[169] Chakraborty, S., Adhikari, S., Ganguli, R.. The role of surrogate models in the development of digital twins of dynamic systems. *Applied Mathematical Modelling* 2021;90:662 – 681.

[170] Mohammadi, H.,, , Riche, R.L., Bay, X., and, . An analysis of covariance parameters in gaussian process-based optimization. *Croatian Operational Research Review* 2018;9(1):1–10. doi:10.17535/crorr.2018.0001.

[171] Xiong, Y., Chen, W., Apley, D., Ding, X.. A nonstationary covariance based kriging method for metamodeling in engineering design. *International Journal for Numerical Methods in Engineering* 2007;71:733 – 756. doi:10.1002/nme.1969.

[172] Laguna, M., Martí, R.. Experimental testing of advanced scatter search designs for global optimization of multimodal functions. *Journal of Global Optimization* 2005;33(2):235–255. doi:10.1007/s10898-004-1936-z.

[173] Adorio, E.P., January, R.. Mvf - multivariate test functions library in c for unconstrained global optimization. 2005,.

[174] Talbi, E.G.. Metaheuristics - From Design to Implementation. John Wiley & Sons; 2009.

[175] Robbins, H.E.. A stochastic approximation method. *Annals of Mathematical Statistics* 2007;22:400–407.

[176] Duchi, J., Hazan, E., Singer, Y.. Adaptive subgradient methods for online learning and stochastic optimization. *J Mach Learn Res* 2011;12(null):2121–2159.

[177] Tenne, Y., Goh, C.K., éditeurs. Computational Intelligence in Optimization. Springer Berlin Heidelberg; 2010. doi:10.1007/978-3-642-12775-5.

[178] Snoek, J., Larochelle, H., Adams, R.P.. Practical bayesian optimization of machine learning algorithms. Dans: *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 2*. NIPS'12; 2012, p. 2951–2959.

[179] Martinez-Cantin, R., de Freitas, N., Brochu, E., Castellanos, J., Doucet, A.. A bayesian exploration-exploitation approach for optimal online sensing and planning with a visually guided mobile robot. *Autonomous Robots* 2009;27(2):93–103. doi:10.1007/s10514-009-9130-2.

[180] Negoescu, D.M., Frazier, P., Powell, W.B.. The knowledge-gradient algorithm for sequencing experiments in drug discovery. *INFORMS J Comput* 2011;23:346–363.

[181] Brochu, E., Cora, V.M., de Freitas, N.. A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *ArXiv* 2010;abs/1012.2599.

[182] Lizotte, D., Wang, T., Bowling, M., Schuurmans, D.. Automatic gait optimization with gaussian process regression. Dans: *Proceedings of the 20th International Joint Conference on Artifical Intelligence*. IJCAI'07; 2007, p. 944–949.

[183] Shahriari, B., Swersky, K., Wang, Z., Adams, R.P., de Freitas, N.. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE* 2016;104(1):148–175.

[184] Kushner, H.J.. A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise. *Journal of Basic Engineering* 1964;86(1):97–106. doi:10.1115/1.3653121.

[185] Bull, A.D.. Convergence rates of efficient global optimization algorithms. *J Mach Learn Res* 2011;12(null):2879–2904.

[186] Priem, R., Bartoli, N., Diouane, Y.. On the use of upper trust bounds in constrained bayesian optimization infill criteria. Dans: *AIAA Aviation 2019 Forum*. American Institute of Aeronautics and Astronautics; 2019,doi:10.2514/6.2019-2986.

[187] Srinivas, N., Krause, A., Kakade, S., Seeger, M.. Gaussian process optimization in the bandit setting: No regret and experimental design. Dans: *Proceedings of the 27th International Conference on International Conference on Machine Learning*. ICML'10. ISBN 9781605589077; 2010, p. 1015–1022.

[188] Villemonteix, J., Vazquez, E., Walter, E.. An informational approach to the global optimization of expensive-to-evaluate functions. *Journal of Global Optimization* 2008;44(4):509–534. doi:10.1007/s10898-008-9354-2.

[189] Thompson, W.R.. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika* 1933;25:285–294.

[190] Shahriari, B., Wang, Z., Hoffman, M.W., Bouchard-Côté, A., de Freitas, N.. An entropy search portfolio for bayesian optimization. *ArXiv* 2014;abs/1406.4625.

[191] Briffoteaux, G., Gobert, M., Ragonnet, R., Gmys, J., Mezmaz, M., Melab, N., *et al.* Parallel surrogate-assisted optimization: Batched bayesian neural network-assisted GA versus q-EGO. *Swarm and Evolutionary Computation* 2020;57:100717. doi:10.1016/j.swevo.2020.100717.

[192] Ginsbourger, D., Riche, R.L., Carraro, L.. Kriging is well-suited to parallelize optimization. Dans: *Computational Intelligence in Expensive Optimization Problems*. Springer Berlin Heidelberg; 2010, p. 131–162. doi:10.1007/978-3-642-10701-6_6.

[193] Bischl, B., Wessing, S., Bauer, N., Friedrichs, K., Weihs, C.. Moi-mbo: Multiobjective infill for parallel model-based optimization. Dans: *LION*. 2014,.

[194] Desautels, T., Krause, A., Burdick, J.W.. Parallelizing exploration-exploitation tradeoffs in gaussian process bandit optimization. *J Mach Learn Res* 2014;15(1):3873–3923.

[195] Mockus, J.. On Bayesian Methods for seeking the extremum. *Optimization Techniques* 1974;.

[196] Snoek, J., Rippel, O., Swersky, K., Kiros, R., Satish, N., Sundaram, N., *et al.* Scalable bayesian optimization using deep neural networks. Dans: *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*. ICML'15; 2015, p. 2171–2180.

[197] Frazier, P.. A tutorial on bayesian optimization. *ArXiv* 2018;abs/1807.02811.

[198] Hwang, C.L., Masud, A.S.M.. Multiple Objective Decision Making — Methods and Applications. Springer Berlin Heidelberg; 1979. doi:10.1007/978-3-642-45511-7.

[199] Edgeworth, F.. Mathematical Physics. P. Keagan, Londres; 1881.

[200] Pareto, V.. Manual of Political Economy. MacMillan, Londres; 1971.

[201] Bechikh, S., Kessentini, M., Said, L.B., Ghédira, K.. Preference incorporation in evolutionary multiobjective optimization. Dans: *Advances in Computers*. Elsevier; 2015, p. 141–207.

[202] Rachmawati, L., Srinivasan, D.. Preference incorporation in multi-objective evolutionary algorithms: A survey. Dans: *2006 IEEE International Conference on Evolutionary Computation*. 2006, p. 962–968. doi:10.1109/CEC.2006.1688414.

[203] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation* 2002;6(2):182–197. doi:10.1109/4235.996017.

[204] Benítez-Hidalgo, A., Nebro, A.J., García-Nieto, J., Oregi, I., Ser, J.D.. jmetalpy: A python framework for multi-objective optimization with metaheuristics. *Swarm and Evolutionary Computation* 2019;:100598doi:https://doi.org/10.1016/j.swevo.2019.100598.

[205] Saxena, A., Prasad, M., Gupta, A., Bharill, N., Patel, O.P., Tiwari, A., *et al.* A review of clustering techniques and developments. *Neurocomputing* 2017;267:664–681. doi:10.1016/j.neucom.2017.06.053.

[206] Xu, D., Tian, Y.. A comprehensive survey of clustering algorithms. *Annals of Data Science* 2015;2(2):165–193. doi:10.1007/s40745-015-0040-1.

[207] Lloyd, S.. Least squares quantization in pcm. *IEEE Transactions on Information Theory* 1982;28(2):129–137. doi:10.1109/TIT.1982.1056489.

[208] Ester, M., Kriegel, H.P., Sander, J., Xiaowei, X.. A density-based algorithm for discovering clusters in large spatial databases with noise. *2nd International Conference on Knowledge Discovery and Data Mining* 1996;.

[209] Quinlan, J.R.. Induction of decision trees. *Machine Learning* 1986;1(1):81–106. doi:10.1007/bf00116251.

[210] Riche, R.L., Picheny, V.. Revisiting bayesian optimization in the light of the COCO benchmark. *Structural and Multidisciplinary Optimization* 2021;64(5):3063–3087. doi:10.1007/s00158-021-02977-1.

[211] Ngan, S.C.. A concrete reformulation of fuzzy arithmetic. *Expert Systems with Applications* 2021;167:113818. doi:10.1016/j.eswa.2020.113818.

[212] Klir, G.J., Pan, Y.. Constrained fuzzy arithmetic: Basic questions and some answers. *Soft Computing - A Fusion of Foundations, Methodologies and Applications* 1998;2(2):100–108. doi:10.1007/s005000050038.

[213] Buckley, J., Qu, Y.. On using α-cuts to evaluate fuzzy equations. *Fuzzy Sets and Systems* 1990;38(3):309–312. doi:10.1016/0165-0114(90)90204-j.

[214] Dong, W., Shah, H.C.. Vertex method for computing functions of fuzzy variables. *Fuzzy Sets and Systems* 1987;24(1):65–78. doi:10.1016/0165-0114(87)90114-x.

[215] Massa, F., Tison, T., Lallemand, B.. A fuzzy procedure for the static design of imprecise structures. *Computer Methods in Applied Mechanics and Engineering* 2006;195(9-12):925–941. doi:10.1016/j.cma.2005.02.015.

[216] Massa, F., Ruffin, K., Tison, T., Lallemand, B.. A complete method for efficient fuzzy modal analysis. *Journal of Sound and Vibration* 2008;309(1-2):63–85. doi:10.1016/j.jsv.2007.06.004.

[217] Akpan, U., Koko, T., Orisamolu, I., Gallant, B.. Practical fuzzy finite element analysis of structures. *Finite Elements in Analysis and Design* 2001;38(2):93–111. doi:10.1016/s0168-874x(01)00052-x.

[218] Lü, H., Shangguan, W., Yu, D.. Squeal reduction of a disc brake system with fuzzy uncertainties. *Journal of Vibroengineering* 2016;18(6):3981–4001. doi:10.21595/jve.2016.17055.

[219] Mesiar, R.. Fuzzy measures and integrals. *Fuzzy Sets and Systems* 2005;156(3):365–370. doi:10.1016/j.fss.2005.05.033.

[220] Awruch, M.D., Gomes, H.M.. A fuzzy α-cut optimization analysis for vibration control of laminated composite smart structures under uncertainties. *Applied Mathematical Modelling* 2018;54:551–566. doi:10.1016/j.apm.2017.10.002.

[221] Calcagnì, A., Lombardi, L., Pascali, E.. Non-convex fuzzy data and fuzzy statistics: a first descriptive approach to data analysis. *Soft Computing* 2013;18(8):1575–1588. doi:10.1007/s00500-013-1164-x.

[222] Garibaldi, J., Musikasuwan, S., Ozen, T., John, R.. A case study to illustrate the use of non-convex membership functions for linguistic terms. *IEEE International Conference on Fuzzy Systems* 2004;3:1403 – 1408. doi:10.1109/FUZZY.2004.1375377.

[223] Götz, M., Graf, W., Kaliske, M.. Enhanced uncertain structural analysis with time- and spatial-dependent (functional) fuzzy results. *Mechanical Systems and Signal Processing* 2019;119:23–38. doi:10.1016/j.ymssp.2018.08.041.

[224] Valdebenito, M., Pérez, C., Jensen, H., Beer, M.. Approximate fuzzy analysis of linear structural systems applying intervening variables. *Computers & Structures* 2016;162:116–129. doi:10.1016/j.compstruc.2015.08.020.

[225] de Lima, B., Teixeira, E., Ebecken, N.. Probabilistic and possibilistic methods for the elastoplastic analysis of soils. *Advances in Engineering Software* 2001;32(7):569–585. doi:10.1016/s0965-9978(00)00102-2.

[226] de Lima, B.S., Ebecken, N.F.. A comparison of models for uncertainty analysis by the finite element method. *Finite Elements in Analysis and Design* 2000;34(2):211–232. doi:10.1016/s0168-874x(99)00039-6.

[227] Hammell II, R., Miao, S., Hanratty, T., Tang, Z.. Comparison of fuzzy membership functions for value of information determination. *CEUR Workshop Proceedings* 2014;1144.

[228] Adil, O., Ali, A., Sumait, B.. Comparison between the effects of different types of membership functions on fuzzy logic controller performance. *International Journal of Emerging Engineering Research and Technology* 2015;3:76–83.

[229] Zadeh, L.. Fuzzy sets. *Information and Control* 1965;8(3):338–353. doi:10.1016/s0019-9958(65)90241-x.

[230] Mourier, K.. Reponse dynamique de structures mecaniques à parametres imprecis. Thèse de doctorat; University of Valenciennes; 2008.

[231] Otto, K.N., Lewis, A.D., Antonsson, E.K.. Approximating α-cuts with the vertex method. *Fuzzy Sets and Systems* 1993;55(1):43–50. doi:10.1016/0165-0114(93)90300-7.

[232] Yang, H.Q., Yao, H., Jones, J.D.. Calculating functions of fuzzy numbers. *Fuzzy Sets and Systems* 1993;55(3):273–283. doi:10.1016/0165-0114(93)90253-e.

[233] Hanss, M.. The extended transformation method for the simulation and analysis of fuzzy-parameterized models. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 2003;11(06):711–727. doi:10.1142/s0218488503002491.

[234] Donders, S., Vandepitte, D., de Peer, J.V., Desmet, W.. Assessment of uncertainty on structural dynamic responses with the short transformation method. *Journal of Sound and Vibration* 2005;288(3):523–549. doi:10.1016/j.jsv.2005.07.003.

[235] Doan, V., Massa, F., Tison, T., Naceur, H.. Coupling of homotopy perturbation method and kriging surrogate model for an efficient fuzzy linear buckling analysis: Application to additively manufactured lattice structures. *Applied Mathematical Modelling* 2021;97:602–618. doi:10.1016/j.apm.2021.04.005.

[236] J.Sadet, , Talbi, E.G., Massa, F., Tison, T., Turpin, I.. Bayesian optimization with deceptive gaussian processes: How to maintain efficiency in the optimization process ? International Conference on Optimization and Learning; 2021,.

Abstract

The automotive squeal is a noise disturbance, which has won the interest of the research and industrialists over the year. This elusive phenomenon, perceived by the vehicle purchasers as a poor-quality indicator, causes a cost which becomes more and more important for the car manufacturers, due to client's claims. Thus, it is all the more important to propose and develop methods allowing predicting the occurring of this noise disturbance with efficiency, thanks to numerical simulations.

Hence, this thesis proposes to pursue the recent works that showed the certain contributions of an integration of uncertainties into the squeal numerical simulations. The objective is to suggest a strategy of uncertainty propagation, for squeal simulations, maintaining numerical cost acceptable (especially, for pre-design phases). Several numerical methods are evaluated and improved to allow precise computations and with computational time compatible with the constraints of the industry.

After positioning this thesis work with respect to the progress of the researchers working on the squeal subject, a new numerical method is proposed to improve the computation of the eigensolutions of a large quadratic eigenvalue problem. To reduce the numerical cost of such studies, three surrogate models (gaussian process, deep gaussian process and deep neural network) are studied and compared to suggest the optimal strategy in terms of methodology or model setting. The construction of the training set is a key aspect to insure the predictions of these surrogate models. A new optimisation strategy, hinging on bayesian optimisation, is proposed to efficiently target the samples of the training set, samples which are probably expensive to compute from a numerical point of view. These optimisation methods are then used to present a new uncertainty propagation method, relying on a fuzzy set modelisation.

Keywords : Structural dynamics, Squeal, Surrogate modelling, Reduced Order models, Optimisation, Uncertainty

Résumé

Le crissement automobile est une nuisance sonore qui intéresse à la fois les chercheurs et les industriels. Ce phénomène fugace, perçu par les acquéreurs de véhicule comme gage de piètre qualité, induit un coût de plus en plus important pour les équipementiers automobiles dû aux réclamations-client. Par conséquent, il est primordial de proposer et développer des méthodes permettant de prédire avec efficacité l'occurrence de cette nuisance sonore grâce à des modèles de simulation numérique.

Ainsi, cette thèse se propose de poursuivre les récents travaux montrant l'apport certain d'une intégration des incertitudes au sein des simulations numériques de crissement. L'objectif est de proposer une stratégie de propagation d'incertitudes pour des simulations de crissement en maintenant des coûts numériques acceptables (en phase d'avant-projet). Plusieurs méthodes numériques sont évaluées et améliorées pour permettre des calculs à la fois précis et dans des temps de calcul compatibles avec les contraintes de l'industrie.

Après avoir positionné ce travail de thèse par rapport aux avancées des chercheurs travaillant sur la thématique du crissement, une nouvelle méthode d'amélioration des solutions propres d'un problème aux valeurs propres complexes est proposée. Pour réduire les coûts numériques de telles études, trois modèles de substitution (processus gaussien, processus gaussien profond et réseau de neurones profonds) sont étudiés et comparés pour proposer les stratégies optimales que ce soit en termes de méthode ou de paramétrage. La construction de l'ensemble d'apprentissage est un élément clé pour assurer les futures prédictions des modèles de substitution. Une nouvelle stratégie/méthode d'optimisation exploitant l'optimisation bayésienne est présentée pour cibler idéalement le choix des données de l'ensemble d'apprentissage, données potentiellement coûteuses d'un point de vue numérique. Ces méthodes sont ensuite exploitées pour proposer une technique de propagation des incertitudes selon une modélisation par sous-ensembles flous.

Mots clés : Dynamique des structures, Crissement, Modèles de substitution, Réduction de modèles, Optimisation, Incertitudes.