Study of neutron correlations measurements, model creation and supervised learning, bayesian inference of nuclear parameters.

Paul Lartaud

INSTITUT POLYTECHNIQUE DE PARIS

ÉCOLE POLYTECHNIQUE

IP PARIS

NNT : 2024IPPAX084

Thèse de doctorat

Étude des mesures de corrélations neutroniques, modélisation et apprentissage automatique, estimation bayésienne de paramètres nucléaires

Thèse de doctorat de l'Institut Polytechnique de Paris
préparée à l'École polytechnique

École doctorale n°574 École doctorale de mathématiques Hadamard (EDMH)
Spécialité de doctorat : Mathématiques appliquées

Thèse présentée et soutenue à Palaiseau, le 03/10/2024, par

M. PAUL LARTAUD

Composition du Jury :

Matthew Eaton Reader, Imperial College (Nuclear engineering group)	Rapporteur
Rodolphe Le Riche Directeur de recherche, École des Mines de Saint-Étienne (LIMOS)	Rapporteur
Pietro Congedo Directeur de recherche, INRIA	Président du jury
Julien Bect Maître de conférences, Centrale-Supélec (L2S)	Examinateur
Nicolas Bousquet Ingénieur-chercheur expert, EDF R&D, Sorbonne Université	Examinateur
Clémentine Prieur Professeure des universités, Université Grenoble Alpes (LJK)	Examinatrice
Josselin Garnier Professeur des universités, École polytechnique (CMAP)	Directeur de thèse
Philippe Humbert Ingénieur-chercheur, CEA DAM DIF	Co-directeur de thèse
Marie Ducauze-Philippe Ingénieure-chercheuse, CEA DAM DIF	Invitée

DOCTORAL THESIS

Study of neutron correlations, modeling, supervised learning and Bayesian estimation of nuclear parameters

Author:
Paul LARTAUD

Supervisors:
Josselin GARNIER
Philippe HUMBERT

Institut Polytechnique de Paris
École polytechnique
Centre de Mathématiques Appliquées
Route de Saclay, 91128 Palaiseau Cedex, France

CEA DAM Île-de-France
Bruyères-le-Châtel, 91297 Arpajon Cedex, France

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy in Applied Mathematics and publicly defended on October 3, 2024, in front of a committee composed of:

Matthew EATON	Imperial College	Reviewer
Rodolphe LE RICHE	École des Mines de Saint-Étienne	Reviewer
Pietro CONGEDO	INRIA	President
Julien BECT	Centrale-Supélec	Examiner
Nicolas BOUSQUET	EDF R&D, Sorbonne Université	Examiner
Clémentine PRIEUR	Université Grenoble Alpes	Examiner
Josselin GARNIER	École polytechnique	Director
Philippe HUMBERT	CEA DAM Île-de-France	Advisor
Marie DUCAUZE-PHILIPPE	CEA DAM Île-de-France	Invited

Remerciements

Cette thèse aura été le point culminant de trois années de travail et marque la fin de mon parcours académique. Elle n'aurait pas vu le jour sans l'aide et le soutien de nombreuses personnes que je souhaiterais donc remercier chaleureusement.

Tout d'abord, j'exprime mes plus profonds remerciements à l'ensemble des membres de mon jury. Je remercie Messieurs Rodolphe Le Riche et Matthew Eaton d'avoir accepté d'être rapporteurs de cette thèse, en sacrifiant ainsi leur mois d'août. *I express my deepest thanks to Matthew Eaton for his reviewing work and for giving up the end of his summer break.* Mes plus sincères remerciements vont également à Julien Bect, Nicolas Bousquet, Pietro Congedo ainsi que Clémentine Prieur qui ont examiné cette thèse et dont les remarques et commentaires sont un apport précieux.

Je tiens à exprimer toute ma gratitude à Marie Ducauze-Philippe pour m'avoir accueilli au sein du laboratoire au CEA, tout en veillant à ma bonne intégration en tant qu'unique doctorant du service. Le repas de Noël fondue/raclette était particulièrement mémorable !

Mes plus sincères remerciements vont à Philippe Humbert pour son encadrement tout au long de ces trois années de thèse. Tu as été mon interlocuteur privilégié dans toute la première partie de la thèse. Ton soutien sans faille et ton expertise ont été d'une grande aide au cours de ce projet.

Enfin, je remercie profondément Josselin Garnier pour avoir dirigé cette thèse et m'avoir accompagné dans ce projet. Tu as su me guider dans un domaine que je découvrais au début de ma thèse. Ta disponibilité constante (dimanche, jours fériés et nuits inclus) et ta bienveillance m'ont placé sur les bons rails pour mener à bien ce projet. J'ai beaucoup appris de ces trois années passées avec toi, et j'espère que nous garderons contact.

Par ailleurs, cette thèse n'aurait pas eu la même saveur sans les très nombreuses rencontres et échanges avec tous les collègues doctorants du CMAP et du CEA. Je remercie l'ensemble des doctorants du CMAP, et je regrette de n'avoir pas pu me dédoubler pour plus m'investir dans la vie de labo. Une mention toute particulière aux alumnis de la JGDS (Josselin Garnier Doctoral School) : Baptiste, Clément, Corentin, Guillaume, Ali, Antoine, Raphaël, Amine, Naoufal. Je remercie aussi l'ensemble des doctorants et nouveaux docteurs du CEA pour m'avoir fait découvrir une passion pour les mots fléchés, et m'avoir confirmé dans mon amour des viennoiseries. Vous êtes trop nombreux pour que je vous cite tous nommément, mais vous avez très

largement contribué à maintenir ma bonne humeur malgré les périodes de stress caractéristiques d'une thèse !

Je souhaite également remercier ma future équipe au CEA pour m'avoir donné l'opportunité de prolonger mon expérience de la recherche dans un environnement que je sais déjà de grande qualité, humaine et scientifique. Plus généralement, je remercie la communauté Mascot-Num / RT-UQ pour les échanges passionnants (et les karaokés !).

Un très grand merci à tous mes amis et anciens camarades de promo pour tous les bons moments passés ensemble, et tous ceux à venir. Je vais dorénavant avoir plus de temps pour vous spammer d'invitations à des après-midi jeux de société, ou pour me faire traquenarder dans des marches de 50 km (prêt Léonard ?).

Il va sans dire que mon parcours académique et le travail que j'ai accompli n'aurait jamais été possible sans le soutien indéfectible de toute ma famille. Je remercie donc infiniment mes parents. Vous répétez à qui veut l'entendre que vos enfants sont les meilleurs, mais ingrat que je suis, je ne vous dis pas assez à quel point vous êtes des parents exceptionnels ! Et votre "fausse" thèse n'y change rien ! Je remercie aussi mes deux grands frères, Antoine et Pierre-Jean, ma belle-soeur Audrey, ma tante Jacqueline, mon oncle Laurent et mes grand-parents. Une mention particulière pour Claire, la nouvelle venue de la famille, et future enfant la plus gâtée de France !

Enfin, comment ne pas remercier la personne qui me soutient (me supporte ?) au quotidien depuis plus de quatre années ? Clara, ta présence au quotidien rend tout plus facile, tu me fais rire, sourire, rêver et aimer tous les instants et je ne te remercierai jamais assez d'être venue me rejoindre en région parisienne. Je t'aime. Une dernière mention spéciale à notre petit chat, qui égaye nos journées, et ce malgré ses multiples modifications intempestives du code source de *numpy* !

Résumé

La sécurité nucléaire consiste à protéger la population et l'environnement contre les actes de malveillance impliquant des substances radioactives. L'Agence Internationale de l'Énergie Atomique (AIEA) exhorte les états à présenter une stratégie de sécurité nucléaire rapide, robuste et fiable, en réponse aux tensions géopolitiques croissantes. En particulier, l'identification des matières nucléaires fissiles est un élément fondamental de toute stratégie de sécurité nucléaire. Pour garantir la fiabilité de la réponse à toute menace, la quantification et le contrôle des incertitudes présentes dans les méthodes mathématiques sous-jacentes sont obligatoires. Cette thèse se place à l'intersection de l'identification des matières fissiles et de la quantification des incertitudes. L'objectif général est de développer des méthodes mathématiques et numériques adaptées à l'analyse du bruit neutronique dans les systèmes sous-critiques de puissance nulle. Cette technique de mesure passive est, avec la spectroscopie gamma, un aspect capital pour l'identification de matière fissile.
La méthodologie présentée dans ce manuscrit se base sur une résolution bayésienne d'un problème inverse, dont les observations proviennent de l'étude des corrélations temporelles entre les neutrons induits par fission. La résolution standard de ce problème est irréalisable telle quelle, en raison du coût du code Monte-Carlo utilisé pour le transport des neutrons. Cette thèse présente un cadre dans lequel le code Monte-Carlo est remplacé par divers méta-modèles, dont les incertitudes intrinsèques sont introduites dans le problème inverse. La procédure de quantification des incertitudes englobe les incertitudes épistémiques et aléatoriques dans un cadre commun. Cette stratégie peut être améliorée à l'aide de plans d'expériences séquentiels conçus spécifiquement pour le problème inverse, ou avec l'introduction de corrélations gamma qui aident à réduire les incertitudes résiduelles. Enfin, nous abordons une approche connexe qui contourne la résolution du problème inverse par une paramétrisation de la classe de distributions a posteriori à l'aide de distributions lambda généralisées.

Abstract

Nuclear security is the task of protecting the population and the environment against malicious acts involving radioactive substances. The International Atomic Energy Agency (IAEA) exhorts states to present a fast, robust, and reliable nuclear security strategy, in response to the increasing geopolitical tensions. In particular, the identification of nuclear fissile matter is a foundational element of any nuclear security strategy. To ensure the reliability of the response to any threat, the quantification and control of the uncertainties embedded in the underlying mathematical methods are mandatory. This thesis is located at the crossroads of fissile matter identification and uncertainty quantification. The general objective is to develop mathematical and numerical methods adapted to the neutron noise analysis in zero-power subcritical systems. This passive measurement technique is, along with gamma spectroscopy, a focal point of fissile matter identification.
The methodology presented in this manuscript is based on a Bayesian resolution of an inverse problem, whose observations come from the study of temporal correlations between fission-induced neutrons. The standard resolution of this problem is intractable due to the cost of the Monte Carlo code for neutron transport. This thesis presents a framework in which the computer model is replaced by various surrogates, whose intrinsic uncertainties are fed into the inverse problem. The uncertainty quantification procedure encompasses both epistemic and aleatoric uncertainties into a common framework. This strategy can be improved with the help of sequential design strategies built specifically for the inverse problem, or with the introduction of gamma correlations which helps in reducing the residual uncertainties. Finally, we discuss a connex approach that circumvents the resolution of the inverse problem with a parametrization of the class of posterior distributions with the help of generalized lambda distributions.

Contents

List of Figures

List of Tables

List of Abbreviations

iid	independent and identically distributed
w.r.t.	with respect to
s.t.	such that
a.s.	almost surely
GP	Gaussian Process
LMC	Linear Model of Coregionalization
BNN	Bayesian Neural Network
MCMC	Monte Carlo Markov Chain
MH	Metropolis-Hastings
AM	Adaptive Metropolis
HMC	Hamiltonian Monte Carlo
NUTS	No-U-Turn Sampler
PCE	Polynomial Chaos Expansion
SUR	Stepwise Uncertainty Reduction

Résumé

Contexte général

Dans le cadre de la sécurité nucléaire à l'échelle d'un territoire, l'état se doit de mettre en place des méthodes efficaces et robustes pour détecter et prévenir des menaces éventuelles. Avec l'émergence et le développement des technologies nucléaires, les dangers potentiels que représentent les matières nucléaires ont conduit au développement de stratégies globales de sécurité nucléaire. Dans la définition donnée par l'Agence Internationale de l'Énergie Atomique (AIEA), la sécurité nucléaire regroupe l'ensemble des moyens et méthodes mis en œuvre afin de détecter, prévenir et répondre aux menaces intentionnelles impliquant des matières nucléaires. Bien que distincte de la sûreté nucléaire, la sécurité nucléaire a également pour but final la protection de la population, de l'environnement et de la société en général.

Une stratégie de sécurité nucléaire repose sur un ensemble de méthodes logistiques, expérimentales et méthodologiques. Cette thèse s'inscrit plus particulièrement dans le cadre des méthodes de détection des matières nucléaires, qui représentent un des piliers amont de la sécurité nucléaire. Les techniques mises en application pour détecter une matière nucléaire sont nombreuses et peuvent êtres classées en deux catégories. On distingue les mesures destructives, pour lesquelles l'échantillon de matière est altéré, des mesures non-destructives. Parmi les mesures non-destructives, on sépare les méthodes actives, où l'on envoie des radiations dans le matériau, des mesures passives pour lesquelles on s'intéresse exclusivement aux émissions spontanées de l'échantillon. Parmi les méthodes les plus utilisées, on peut citer par exemple la spectroscopie gamma qui s'inscrit dans les mesures passives non-destructives ou la radiographie qui est une mesure non-destructive active.

Dans cette thèse, nous nous concentrons spécifiquement sur les mesures de corrélations neutroniques, qui font partie des méthodes de détection passive et sont généralement complémentaires de la spectroscopie gamma. Le terme de corrélation neutronique fait référence à la corrélation temporelle entre les instants de détection de deux neutrons appartenant à une même chaîne de fission. Les mesures de corrélations neutroniques reposent sur l'analyse de la distribution temporelle des neutrons captés par un détecteur externe. En particulier pour les matériaux fissiles, la multiplicité des neutrons produits lors des fissions entraîne un excès de variance dans les statistiques de détection. Ces méthodes, dites de bruit neutronique, ont été abondamment étudiées et permettent d'identifier des caractéristiques d'un matériau comme sa multiplication ou sa constante de décroissance prompte. Elles sont appliquées aussi bien à des réacteurs de puissance, pour identifier des anomalies de fonctionnement, qu'à des systèmes de faible puissance. On s'intéresse ici à ce second cas.

Objectifs et plan

Bien que connues et utilisées depuis les années soixante, les méthodes de bruit neutronique souffrent d'un manque de quantification d'incertitudes. Pourtant, afin de garantir l'efficacité d'une stratégie nationale de sécurité nucléaire, la maîtrise des incertitudes doit faire partie intégrante du cadre analytique. Ce projet de thèse a donc pour but de combler, au moins en partie, cette lacune méthodologique.

L'objectif global de ce projet de thèse est le suivant. On souhaite développer des méthodes mathématiques et numériques robustes et fiables afin de quantifier les incertitudes lors de l'identification de matières fissiles à l'aide de méthodes de bruit neutronique. Plus particulièrement, le chemin choisi se base sur l'estimation bayésienne de certains paramètres nucléaires caractéristiques d'un matériau fissile lors de la résolution d'un problème inverse. Afin d'accélérer la résolution de ce problème inverse, des méthodes d'apprentissage supervisé sont utilisées afin de fournir des modèles prédictifs rapides et fiables, et dont les incertitudes sont connues et maîtrisées.

Ce manuscrit se décompose en sept chapitres et est organisé comme suit. Tout d'abord, le chapitre 1 présente un rappel méthodologique général sur les processus markoviens et leurs applications à la neutronique aléatoire, la résolution bayésienne de problèmes inverses et l'échantillonnage Monte-Carlo par chaînes de Markov.
Par la suite, le chapitre 2 se focalise sur le développement de méta-modèles à l'aide de processus gaussiens, pouvant servir d'émulateurs fiables du modèle direct lors de la résolution du problème inverse.
Dans le chapitre 3, nous étendons cette méthodologie à une nouvelle classe de méta-modèles que sont les réseaux de neurones bayésiens.
Par la suite, dans le chapitre 4, nous développons des stratégies d'apprentissage actif spécifiquement conçues dans le but d'améliorer nos méta-modèles dans le cadre de la résolution d'un problème inverse donné.
Le chapitre 5 aborde l'utilisation de mesures de corrélations gamma qui sont l'analogue des corrélations neutroniques pour les rayonnements gamma. On montre que ces mesures peuvent servir à apporter une information supplémentaire et donc à réduire les incertitudes en jeu.
Nous évoquons ensuite dans le chapitre 6 une stratégie alternative pour la résolution du problème inverse qui se base sur l'apprentissage de la distribution a posteriori à l'aide de distributions paramétriques.
Enfin, nous achevons ce manuscrit par une conclusion générale dans le chapitre 7 faisant un aperçu succinct des contributions de ce travail et des perspectives futures.

Dans la suite de ce résumé, nous discuterons plus en détail des contributions réalisées pendant ce projet de thèse.

Quantification d'incertitudes par méta-modélisation

Problème inverse

L'objectif de ce projet de thèse est l'estimation de paramètres nucléaires et de leurs incertitudes. D'un point de vue mathématique, cet objectif se traduit par la résolution d'un problème inverse. À partir d'observations bruitées de corrélations neutroniques, qui peuvent être obtenues par mesures expérimentales ou par simulations, on cherche

à remonter aux paramètres nucléaires caractéristiques du matériau. Afin de quantifier les incertitudes sous-jacentes, l'approche traditionnelle consiste à résoudre le problème inverse par une approche bayésienne. Cependant, cette voie présente un obstacle majeur : le modèle direct, qui fournit le lien entre paramètres nucléaires et mesures de corrélations, est représenté par un code de calcul extrêmement coûteux. Comme la résolution du problème inverse est soumise à des contraintes en temps, le modèle direct ne peut pas être utilisé tel quel dans la résolution du problème inverse.

Pour cette raison, on cherche à remplacer le modèle direct par un émulateur, autrement appelé méta-modèle. On souhaite que le méta-modèle fournisse des prédictions proches du vrai modèle direct ainsi qu'une quantification des incertitudes intrinsèques.

Bases de données

La première étape de ce projet consiste en la création de méta-modèles pour le modèle direct à l'aide de processus gaussiens (voir chapitre 2) ou de réseaux de neurones bayésiens (voir chapitre 3).
Afin de créer ces émulateurs par apprentissage supervisé, on construit d'abord les données à l'aide de simulations neutroniques Monte-Carlo avec le code MCNP6. Partant d'une configuration de référence, nous modifions la géométrie et la source aléatoirement afin de créer une nouvelle simulation. Le processus est itéré un grand nombre de fois pour construire nos données numériques d'entraînement. Les deux cas de référence sont respectivement l'étude d'une sphère de plutonium métallique avec réflecteur en cuivre, issue de la littérature scientifique, et une modélisation du réacteur SILENE, un réacteur cylindrique à nitrate d'uranyle.

Les quantités d'intérêt caractérisant les corrélations neutroniques sont le taux de comptage moyen R, et les moments de Feynman asymptotiques d'ordre deux et trois notés Y_∞ et X_∞, qui représentent respectivement les nombres moyens de doubles et triples détections corrélées. Nous entendons par détection corrélée la détection simultanée de plusieurs neutrons issus d'une même chaine de fission. Pour décrire le matériau, nous nous basons sur le modèle ponctuel qui utilise le facteur de multiplication prompt k_p, l'efficacité de détection (de Feynman) ε_F, l'intensité de la source S et le type de source x_s, qui quantifie le ratio de fissions spontanées et de réactions nucléaires (α, n) dans la source. Nous ajoutons à ces entrées des paramètres additionnels qui caractérisent notamment le spectre en énergie, la réflexion en bord de géométrie et les absorptions parasitiques.

Pour déterminer les entrées de notre base de données, nous nous basons sur les fonctionnalités classiques de MCNP comme le calcul de valeur propre pour k_p et les estimateurs de comptage pour les autres quantités. Les paramètres liés au terme source (S et x_s) peuvent être fixés manuellement dans les fichiers entrées de MCNP et n'ont donc pas à être estimés.
Pour estimer les sorties (R, Y_∞, X_∞), nous avons à notre disposition deux estimateurs possibles. L'estimateur de binning séquentiel est analogue à un traitement expérimental du fichier d'instants de détection, alors que l'estimateur par déclenchement utilise les informations contenues dans la simulation MCNP pour filtrer les corrélations accidentelles entre neutrons détectés. En effet, on sait lors d'une simulation à quelle chaine de fission appartient un neutron et on peut donc utiliser cette information pour filtrer les corrélations accidentelles. Cette seconde approche est utilisée pour la création des bases de données d'entraînement. En revanche, pour créer les données

d'observations expérimentales utilisées pour résoudre les problèmes inverses, nous utilisons les estimateurs par binning séquentiel afin de reproduire des vraies données expérimentales.

Avec cette méthode nous construisons donc deux bases de données numériques. La première, dénommée **BERP dataset** se base sur le cas de la sphère de plutonium métallique et contient 2132 données, et la seconde **SILENE dataset** se base sur la géométrie du réacteur SILENE et contient 318 cas. Nous insistons sur la distinction entre ces données d'entraînement, et les observations d'un problème inverse qui peuvent également être obtenues par simulation, mais qui sont liés à une configuration spécifique d'une matière nucléaire.

Méta-modèles

Nous construisons par la suite des méta-modèles à l'aide de ces bases de données numériques. Les premiers modèles développés sont centrés sur la régression par processus gaussiens avec le modèle de corégionalisation. Ces modèles permettent de tenir compte des corrélations entre sorties qui sont attendues dans notre cas. Nous évaluons la performance de ces modèles à l'aide de plusieurs métriques quantifiant la fiabilité des prédictions moyennes mais aussi de la quantification des incertitudes. Les modèles sont améliorés ensuite par le biais de modélisations multi-fidélité et hétéroscédastique.

Par ailleurs, nous développons des méta-modèles à l'aide de réseaux de neurones bayésiens. Partant d'une architecture fixée, nous testons différentes méthodes d'inférence telles que l'inférence variationnelle, le *dropout* Monte-Carlo et un échantillonnage MCMC. Nous étudions également l'influence des différentes distributions a priori dans le réseau et tout particulièrement l'a priori sur les covariances de sortie. Enfin nous modifions aussi l'architecture des réseaux afin d'inclure une modélisation hétéroscédastique et multi-fidélité permettant d'améliorer les performances des modèles.

Avec de ces méta-modèles, on peut résoudre le problème inverse bayésien à l'aide d'une vraisemblance incluant à la fois les erreurs épistémiques et aléatoriques. Dans le cas des réseaux de neurones bayésiens, dont la distribution de sortie n'est pas gaussienne, deux vraisemblances différentes sont étudiées. Dans la première, on combine les moyennes et covariances prédictives fournies par chaque échantillon des paramètres du réseau, pour revenir à une distribution gaussienne qu'on peut inclure facilement dans la vraisemblance. Dans la seconde en revanche, on considère une vraisemblance sous la forme d'une mixture de gaussiennes, qui est issue de l'approximation Monte-Carlo de la loi prédictive du réseau de neurones bayésien. Dans tous les cas, le problème inverse est résolu à l'aide d'échantillonnage MCMC qui fournit une chaîne de Markov ergodique dont la distribution invariante est la distribution a posteriori dans la vision bayésienne du problème inverse. Les diverses méthodes présentées ici sont appliquées sur deux cas tests de référence. Le premier représente une sphère de plutonium métallique entourée de réflecteurs en cuivre. Pour ce cas test, les observations du problème inverse sont obtenues par simulation numérique en utilisant les estimateurs par binning séquentiel de manière à reproduire des données expérimentales. Le second cas test est tiré d'une campagne de mesures sur le réacteur expérimental SILENE. Nous montrons que notre méthodologie permet une quantification fiable des incertitudes liées aux observations et aux modèles numériques utilisés.

Design séquentiel

Dans le chapitre 4, nous nous penchons sur la création de plans d'expériences adaptés à la résolution bayésienne d'un problème inverse. Les méthodes proposées dans ce chapitre sont construites pour un problème inverse spécifique et ne cherchent donc pas à fournir le meilleur modèle possible. Elles vont plutôt chercher à sélectionner de nouveaux points d'entraînement de sorte à améliorer le modèle dans les régions les plus pertinentes, c'est-à-dire dans les régions de grande densité a posteriori. Ces stratégies peuvent être comprises comme un réglage fin d'un méta-modèle en vue de la résolution d'un problème inverse spécifique.

Design CSQ

En partant des designs dits D-optimaux, nous développons une stratégie de design séquentiel pour les problèmes inverses bayésiens. Cette méthode, dénommée CSQ (pour *Constraint Set Query*), cherche un nouveau point d'entrainement de manière à maximiser le déterminant de la covariance prédictive du méta-modèle, tout en se restreignant à une boule centrée (pour la distance de Mahalanobis) autour du maximum-a-posteriori. Cette première approche est facile à mettre en œuvre et garantit que les nouveaux points d'entraînement sont dans des régions de haute densité a posteriori.

Stratégie IP-SUR

Une deuxième méthode, intitulée IP-SUR (*Inverse Problem SUR*) et se fondant sur la méthodologie SUR (*Stepwise Uncertainty Reduction*) est également présentée. Par rapport à la stratégie CSQ, la stratégie IP-SUR présente des garanties théoriques de convergence. En effet, on prouve que la métrique d'intérêt, qui est ici le déterminant de la covariance prédictive, intégré sur la distribution a posteriori, converge presque sûrement vers zéro quand le nombre de simulations tend vers l'infini. Par ailleurs, nous proposons une méthode alternative se fondant sur l'utilisation d'une vraisemblance tempérée par un exposant compris entre 0 et 1, de manière à lisser la distribution a posteriori lors de la recherche du nouveau point d'entrainement. Cette méthode présente les mêmes garanties théoriques que la méthode IP-SUR pour un coût similaire.

Développements supplémentaires

Corrélations gamma

Dans le chapitre 5, nous étudions la possibilité d'utiliser l'information supplémentaire apportée par les corrélations gamma dans le problème inverse pour réduire les incertitudes et l'étalement de la distribution a posteriori. Cette approche peut se faire de deux manières. On peut utiliser des méta-modèles de plus grande dimension pour traiter conjointement les corrélations neutron et gamma. Il est également possible de résoudre successivement deux problèmes inverses, d'abord pour les corrélations neutron, puis pour les corrélations gamma. La première approche est théoriquement plus informative car elle tient compte de la structure de covariance complète des corrélations neutron et gamma. Néanmoins elle ne produit pas initialement de bons

résultats du fait du trop faible nombre de données numériques utilisées pour entraîner le méta-modèle. C'est pourquoi nous cherchons à enrichir le méta-modèle à l'aide des méthodes de design séquentiel précédemment introduites. Ces méthodes permettent d'améliorer significativement le méta-modèle neutron/gamma mais les incertitudes obtenues par cette approche conjointe restent plus importantes que dans le cas où les corrélations sont traitées séquentiellement. Cela s'explique par une sous-estimation des incertitudes lors de l'approche séquentielle, liée à l'absence de corrélations entre observations neutron et gamma dans cette méthode. Dans ce chapitre, nous démontrons donc le gain potentiel que peuvent apporter les corrélations gamma pour réduire les incertitudes dans l'estimation des paramètres nucléaires.

Apprentissage des distributions a posteriori

Enfin, dans le chapitre 6, nous étudions une approche détournée pour résoudre le problème inverse, en s'efforçant de modéliser la distribution a posteriori des paramètres nucléaires du matériau étudié par une distribution paramétrique. Les distributions lambda généralisées, corrélées par une copule gaussienne, sont utilisées pour fournir une classe de distributions paramétriques suffisamment générales. Une base de données est créée de manière à pouvoir entraîner un prédicteur, capable pour un ensemble d'observations et une covariance de mesure, de fournir les paramètres de la distribution a posteriori associée. Cette méthode alternative est largement plus rapide, et permet de fournir instantanément une distribution a posteriori complète. Les incertitudes fournies par ces distributions sont moins fiables que celles obtenues par résolution bayésienne du problème inverse. Néanmoins, les exemples étudiés semblent indiquer que les distributions paramétriques fournissent une surestimation systématique des incertitudes, ce qui rend leur utilisation envisageable pour fournir une estimation conservatrice des incertitudes en temps très court.

Conclusion

Cette thèse visait à quantifier les incertitudes associées à l'estimation de paramètres nucléaires dans un matériau inconnu à partir de mesures de corrélations neutroniques. Pour répondre à cet objectif, nous avons développé des méta-modèles basés sur des processus gaussiens et des réseaux de neurones bayésiens, utilisant des bases de données numériques générées par simulation. Ces méta-modèles agissent comme des émulateurs du modèle direct dans le cadre de la résolution bayésienne de problèmes inverses, en tenant compte de l'observation bruitée des mesures de corrélations neutroniques. Nous avons démontré la possibilité d'améliorer ces modèles pour des problèmes inverses spécifiques en utilisant des plans d'expériences adaptés pour la sélection optimale de nouveaux points d'entraînement. Par ailleurs, l'intégration de mesures de corrélations gamma constitue une autre approche prometteuse pour réduire les incertitudes dans l'estimation des paramètres nucléaires. Enfin, nous avons illustré la capacité à fournir une estimation instantanée et conservatrice des incertitudes à partir de données d'observation, en utilisant des distributions paramétriques.

Les travaux théoriques et appliqués présentés dans ce manuscrit ont abordé la problématique posée tout en identifiant de nouvelles opportunités de recherche dans des domaines connexes. Une voie d'amélioration de cette étude réside dans une paramétrisation optimale du problème inverse. Le cadre général de ce problème

repose sur l'utilisation d'un modèle ponctuel auquel quelques paramètres supplémentaires sont ajoutés pour une description plus précise des phénomènes physiques. Toutefois, ces paramètres ont été choisis de manière empirique et il est possible qu'une paramétrisation plus pertinente existe. De plus, d'autres approches alternatives pour résoudre le problème inverse n'ont pas été explorées ici, comme par exemple l'utilisation de modèles génératifs conditionnels. Enfin, bien que nous ayons démontré le potentiel de l'intégration des corrélations gamma dans l'estimation des paramètres nucléaires, ces méthodes n'ont pas encore été appliquées à des données réelles et se base sur une modélisation simpliste des processus physiques sous-jacents. Une modélisation plus réaliste de ces phénomènes et l'application à des données expérimentales serait nécessaire pour garantir la fiabilité de cette approche.

Chapter 1

Introduction

1.1 General introduction

The International Atomic Energy Agency defines nuclear security as all the experimental and theoretical methods seeking to safeguard people from malicious acts involving radioactive substances. In a context of geopolitical instabilities, nuclear security emerges as an increasingly pivotal aspect of national defense. The scope of nuclear security is broad and diverse, though this work focuses on fissile matter identification, using passive neutron and gamma correlation measurements.

A state must guarantee an efficient and fast response methodology for detecting and identifying nuclear and radioactive materials. Robust mathematical and numerical methods for the analysis of field measurements are one of the foundations that cement the reliability of the national nuclear security program. The research presented in this thesis is integral to the development and enhancement of these methods.

In a robust nuclear security architecture, field measurements are crucial in shaping responses to potential threats. The specific context requires fast and reliable decision-making, which is only possible if the field measurements and their conclusions on the run are reliable. Additionally, the continuous improvement of the nuclear security strategy may require a more complete analysis and benchmarking of the previously encountered threats and exercises. For all these reasons, nuclear security must rely on well-developed uncertainty quantification methods, spanning the entire spectrum of the speed-accuracy trade-off inherent to computational methods.

This thesis addresses the problem of uncertainty quantification in passive neutron multiplicity counting. This technique aims at identifying a fissile material based on the temporal correlations of detected neutrons. Serving as a passive assay method, it is complementary to gamma spectroscopy.

Neutron multiplicity counting allows for the determination of key parameters that describe fissile material by solving an inverse problem. However, the uncertainty quantification associated with these estimates is often overlooked. Given the ill-posed nature of the underlying inverse problem, estimations may exhibit strong dependency on observations, leading to significant discrepancies between estimates derived from independent observation sets. In this context, robust uncertainty quantification methods appear as an essential component of nuclear security.

The goal of this thesis is to develop a mathematical framework that ensures reliable uncertainty quantification for the inverse problem associated with fissile material identification. These methods are tailored to operate within constrained time budgets to facilitate on-field decision-making. The limited computational budget may be relaxed for post-analysis of previous measurement campaigns. Though the general

focus is on nuclear security, the mathematical developments presented herein hold potential applicability beyond nuclear security, extending to domains such as nuclear safety, especially for criticality accidents, and nuclear safeguards, whose goal is to prevent the diversion of nuclear material for hostile purposes.

This thesis is organized as follows: in Chapter 1, we begin with a general introduction to neutron multiplicity counting and the mathematical concepts recurrent in this work. In Chapter 2, we focus on the development of Gaussian process surrogate models which help solve the inverse problem reliably and promptly. Then, the focus is shifted to Bayesian Neural Networks as surrogate models which may provide a more costly yet more reliable uncertainty quantification in Chapter 3. Then, in Chapter 4 we develop a sequential design strategy to enrich the surrogate models for a given inverse problem. Our next step is to introduce new information in the form of gamma correlations, to further reduce the uncertainties in the estimation of nuclear parameters in Chapter 5. Finally in Chapter 6, we develop an alternative strategy that circumvents the need to solve the inverse problem by building a predictor able to solve the inverse problem with a constrained time budget using parametric probability distributions. A general conclusion in Chapter 7 summarizes the various contributions of this thesis.

1.2 Identification of nuclear fissile material

Various experimental techniques have been used to identify a fissile material containing a mixture of uranium and plutonium, and traces of other actinides or fission products. They are usually split into two categories, destructive and non-destructive methods. This thesis focuses on non-destructive assay techniques and specifically on neutron multiplicity counting. Still, we provide here a brief overview of the different methods used in nuclear material identification.

1.2.1 Destructive assay

The destructive assay techniques refer to experimental approaches in which the material or a fraction of it, is altered, destroyed, or meddled with, instead of having a passive surveillance of its characteristics. Among these techniques, one can think of mass-spectrometric analysis, spectrophotometry, and other chemical analysis methods.

Spectrometric analysis can be used to identify the contents of uranium and plutonium in a sample by the isotopic dilution technique. The idea is to introduce a reference volume of the sample in a solution of known isotopic compositions. The newly obtained solutions have updated isotopic compositions which can be obtained by mass spectrometry. The changes in composition are then used to estimate the isotopic composition of the unknown sample [HPL08; Ing54].

When focusing solely on the plutonium or uranium content, the sample can be directly dissolved in nitric acid (or a mixture of nitric and hydrofluoric acids for oxides) and then analyzed using mass spectrometry.

Spectrophotometry studies the interaction between the material and external radiations. For example, it is possible to measure the absorbance of a coordination complex formed by uranium at specific wavelengths [KRA+09]. For high concentrations, it is

also possible to study directly the absorbance of the dissolved uranium or plutonium [DP16].

Numerous other methods have been used over the years based on the actinide to be identified, like gravimetric methods [YHK22], or potentiometric titration [HWP78; Kar+86]. However, since destructive assay techniques are outside the scope of this thesis, no further details are provided and the author refers to the cited works for interested readers.

1.2.2 Non-destructive assay by active methods

On the other hand, non-destructive techniques avoid invasion or alteration of the material of interest. One can distinguish passive methods, where the information is obtained from spontaneously emitted radiation from the object, and active methods, in which the object is subject to external radiations, and the induced radiations it produces are analyzed. Active methods complement the passive methods introduced afterward, but they tend to be more difficult to set up for on-field measurements.

When a material is excited by incident X-ray (or gamma) radiations, ionization of the atoms can produce an induced X-ray emission. This fluorescence is characteristic of the element and can be used to analyze the content of the sample [MRH15; MDM06]. This method is known as X-ray fluorescence analysis (XRF).

It is also possible to use an external gamma source to measure the absorbance of an unknown sample in a solution. The attenuation of the gamma radiation through the solution can be linked to the concentration in uranium or plutonium [Con+58].

Uranium and plutonium can induce fission when hit by an incoming neutron, which in turn produces high-energy neutrons and gamma rays. Thus, one can deliberately subject an unknown sample to neutron irradiation and measure the neutron and gamma response.

Identification of fissile material can also be done by X-ray radiography. If the material is well-shielded, the X-ray may not have enough penetrating power, and in that case, gamma radiography may provide more information (see for example [OI+16; Hen+18; Kat+07])

1.2.3 Non-destructive assay by passive methods

1.2.3.1 Calorimetric measurements

Radioactive materials continuously emit radiation, among which α and β particles, that have a much lower penetrating power than gamma rays. These particles deposit their energy by heating the material. From this consideration, calorimetric measurements have been developed to monitor the heat production in radioactive materials. Of course, one cannot hope to find an isotopic composition of the material with calorimetric measurements, but it can provide information on the global radioactive content of the sample. These measurements are most notably applied to plutonium characterization since it has a much higher heat generation rate than uranium [Bra+02; Bat15]. For this reason, this method is mostly linked to nuclear safeguards considerations.

1.2.3.2 Gamma spectroscopy

Passive gamma-ray spectroscopy is a frequently used experimental technique whose goal is to identify spontaneous gamma-ray emissions caused by the radioactive decay of nuclides. The energy of the gamma rays is specific to each nuclide. The intensity of the peak in the detected gamma-ray spectrum can thus be used to identify nuclides and their respective abundances [CK18; Wil06; Dew+16]. Gamma spectroscopy can also provide information on the shielding by measuring the inherent gamma attenuation in the object. The inference can be made difficult in cases where the sample has a high content of fission products. Indeed, in this case, most of the emitted gamma signals come from the fission products and the identification of the specific uranium or plutonium can become tedious, though some methods have been developed to overcome this difficulty [BC98]. Gamma spectroscopy is one of the most commonly used techniques for nuclear material identification [Dew+16; Toh+16]. It is a complementary approach to neutron coincidence counting.

1.2.3.3 Neutron coincidence counting

In a radioactive material, neutrons can be emitted by spontaneous fissions of the nuclides (in ^{240}Pu or ^{238}U mainly), or they can be produced by nuclear reactions between low atomic number atoms and the α particles emitted during radioactive decay. These latter reactions often occur in the form of (α, n) reactions. Moreover, neutrons can be multiplied by neutron-induced fission reactions, in which an incoming neutron triggers the fission of a heavy nucleus, releasing more neutrons. Thus, one could try to measure directly the spontaneous neutron rate to obtain information on the content of the unknown sample. However, this cannot provide knowledge of the isotopic composition. Besides, the (α, n) reactions are also strongly dependent on the content of lighter atoms. As such, it is difficult to extract useful knowledge from global neutron counting since most of the detected neutrons would come from the (α, n) reactions.

However, one can filter out the neutrons created by the (α, n) source term. The principle is the following. Neutrons are produced by batches in spontaneous and induced fissions. They are correlated in time while the (α, n)-induced neutrons are not. Assume a neutron is detected at time t, and a detector system counts neutrons in the time interval $[t, t + T]$. In this window, one can distinguish two cases when a new neutron is detected. Either it is correlated to the neutron detected at time t, meaning that the two neutrons belong to the same fission chain, in which case we say we have a true correlation, or it is not and we have an accidental correlation. True correlations have a limited lifetime linked to the average lifetime of fission chains in the material. To obtain only the true correlations, neutron coincidence counters generally open two detection gates: the first one in the interval $[t, t + T]$ counts both accidental and true correlations, and the second one in the interval $[t + t', t + t' + T]$, with t' being a few seconds, counts only the accidental correlations since most of the true correlations have died out. Then, the number of true correlations can be obtained by subtracting the second gate from the first.

Neutron coincidence counting may also refer to other methods such as the Rossi-α and Feynman-α methods which will be extensively discussed in this thesis. These experimental techniques are at the heart of passive non-destructive assay and are complementary to gamma spectroscopy. They have been widely used for a whole range of applications, and in particular for nuclear safeguards (see for example

[Bru+96; BBC99; Cro+12; DF+17; PBJ19]). Coincidence counting is the main technique of interest in this work. It is also worth mentioning that efforts have been made to extend the framework of coincidence counting to gamma detections as well [Che+15; DYL18]. This approach will also be further discussed in this thesis.

1.2.4 Neutron noise analysis

Neutron noise analysis studies the fluctuations of the neutron population, otherwise known as neutron noise, inside a fissile material. More specifically, we distinguish between neutron noise in power reactors where the noise can be traced back to defects or vibrations in the core, and noise in zero-power systems which is a consequence of the stochastic nuclear reactions in the system.

1.2.4.1 Neutron noise in power reactors

Power reactors are subject to random modifications of their properties during operations, such as variations in the coolant density, phase changes, temperature variations, or mechanical vibrations. These perturbations induce changes in the macroscopic cross-sections which then impact the neutron flux and population. The temporal fluctuations of the neutron population are known as power reactor noise.

In power reactor noise, the underlying processes are very complex since they involve stochastic processes whose evolutions are themselves governed by randomly fluctuating quantities, which can be space and time-dependent [PA80]. Yet, the spectral analysis of the reactor transfer function can be used to identify locations of vibrating faulty control rods [Fry71; DP09]. Power noise can also be used to identify coolant flow blockage [OV+06] or local changes in the void fraction in Boiling Water Reactors (BWR) [And+75].

1.2.4.2 Neutron noise in zero-power systems

On the other hand, neutron noise in zero-power systems deals with fluctuations caused by the inherently stochastic nature of the physical processes involved. Such systems are at the core of this thesis. They can be for example experimental reactors with low output power or spent fuel drums. The focus is more specifically on sub-critical systems (with a source term). Neutron noise analysis in zero-power systems has also been applied to supercritical systems during the start-up phase, at which point the neutron population is still low enough for the stochasticity to play a decisive role [Har65]. Other works focused on the probability of extinction of the branching process describing the neutron population for such systems [Wil79; Tan24; WP15]. In [Coo+16], the focus is on the uncertainty quantification of the estimated probability of extinction. For pulsed experiments without external sources, the intrinsic source term (either from ^{238}U or ^{240}Pu) is low and the burst time of the pulse can vary between two successive experiments because of the predominant stochastic fluctuations at the start-up. Numerous studies were conducted to estimate the behavior of the random burst times [Wil16; ARH14; HM04; Han60].

In sub-critical systems, the Feynman-α and Rossi-α methods are the most standard approaches for extracting information on the multiplicative system based on the study of neutron fluctuations. These methods are more thoroughly presented afterward. Other techniques, which will not be further discussed in this thesis have also been

developed. In [Bab67], the distribution of the time intervals between two successive neutron detections is studied. This distribution differs from a memoryless exponential distribution in a multiplying medium, which occurs for standard non-multiplying nuclear reactions. Another method focuses on the auto and cross power spectral density, which are obtained by Fourier transform of the corresponding correlation function of the detector signals [Coh60; Pak+18].

For the rest of the work presented here, the focus is specifically on neutron noise analysis in zero-power subcritical systems. The theory of neutron fluctuations, described in terms of branching processes is further discussed in Section 1.4.

1.2.5 Neutron and gamma detection

We now turn our attention to the experimental means of neutron and gamma detections. This section serves only as a brief introduction to the subject. For a more expansive review of past and current methods, the author refers to [Peu00; CB91; CAD13].

1.2.5.1 Proportional counters

Proportional counters are a type of detector, suitable for neutron and gamma detections. Their main characteristic is that their output signal is proportional to the energy of the incident particle, which helps to probe the energy spectrum of the incoming particles. They generally consist of cylindrical tubes containing an inert gas at a given pressure (usually argon). When ionizing radiation enters the detector, it can ionize the atoms in the gas, which produces pairs of electrons and ions. A voltage is applied in the gas counter which separates the pairs. When too high voltage is applied, the accelerated charged particles (electrons or α particles) can generate Townsend avalanches, ionizing more atoms to produce additional particles and perpetuating the process. This phenomenon amplifies the output signal significantly. However, the voltage must be kept sufficiently low to produce only one avalanche per event otherwise the output signal is not proportional to the incident particle energy. This key calibration is at the basis of proportional counters.

The choice of the filler gas will govern the possible applications. For neutron detection, proportional counters usually contain boron-trifluoride BF_3 enriched in ^{10}B or ^{3}He [BAS55]. Incident neutrons in the thermal range (energy below 0.025 eV) can react with these fill gases respectively by (n, α) or (n, p) reactions. Both counters display low sensitivity to gamma making them specific to neutron detection, but ^{3}He detectors tend to have higher efficiency while being more expensive.

Although they are widely used even today for neutron detection and counting, the two aforementioned detectors are specific to thermal neutrons. Thus, if used to detect high-energy neutrons, they need to be encapsulated in a moderator matrix (like polyethylene or graphite) to slow down the neutrons before they can be detected, otherwise, the detector efficiency is significantly lower. In this case, the knowledge of the incident neutron spectrum is lost in the moderator and cannot be recovered. For fast neutron detection, scintillation detectors are most often used.

1.2.5.2 Scintillation detectors

Scintillation detectors are a class of detectors whose working principle is based on the detection of a visible light pulse obtained after interaction between the incoming radiation and a scintillation material. This pulse goes through a photomultiplier and a photodiode to produce an output signal.

Scintillation detectors are more and more common as they have the great benefit of being able to detect both neutrons and gammas. They can also detect both thermal and fast neutrons [Lee+14; BA60]. The choice of the scintillation material impacts the efficiency of the detection of the particle of interest. For example, liquid scintillation detectors have been commonly used for β-decaying radionuclides [Hou18]. For simultaneous neutron and gamma detections, anthracene and stilbene are commonly chosen scintillation crystals. The pulse signals corresponding to neutron and gamma detections can be distinguished by mean of Pulse Shape Discrimination (PSD) (see for example[YWF15; Ber+15; KE05; Zai+12]). In the more recent advances in this particular domain, learning methods have been applied to the PSD of the scintillation signal [Dur+21; Gri+20].

1.2.5.3 Semiconductor detectors

Though scintillation detectors are suitable for detecting gamma radiations, the state-of-the-art, in terms of gamma detection capabilities, are the semiconductor detectors such as high-purity germanium (HPGe) detectors. When ionizing radiation reaches the detector, it produces pairs of electrons and holes in the semiconductor, which are translated into an electric signal. The number of pairs is proportional to the incident energy. Since the production of electron-hole pairs requires less energy than the ionization of atoms in a gas counter, many more pairs are produced from a single radiation. Thus semiconductor detectors have a much higher resolution than gas counters or scintillation counters.

The most commonly used detector is the HPGe detector, though some other detectors have been used for more specific tasks such as cadmium telluride detectors [AMI17; Sif+75]. Semiconductor detectors are among the most commonly used tools for gamma spectroscopy. For a more detailed review, the author points to [DH88; Lut+07].

1.3 Stochastic processes in random neutronics

This section serves as an introduction to stochastic processes in countable state space. We deal mainly with continuous-time stochastic processes, with a specific focus on Markov processes, branching processes, and Poisson processes. The subjects tackled in this section are then applied in the next section to study neutron and gamma fluctuations in zero-power systems. This preliminary is based on comprehensive works on the subject such as [MT93; AH+83; And12].

1.3.1 Continuous-time Markov processes

1.3.1.1 Introductory concepts

In everything that follows, we define a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ where Ω is the sample space, $\mathcal{F}$ is a σ-algebra and $\mathbb{P}$ is a probability measure.

Definition 1.3.1 (Stochastic process)**.** *Let $(\mathcal{S}, \Sigma)$ be a measurable space. A stochastic process is a collection of random variables $(X_t)_{t\in\mathcal{T}}$ indexed by a set $\mathcal{T}$ with values in $\mathcal{S}$.*

Remark. A stochastic process $(X_t)_{t\in\mathcal{T}}$ can be understood as a measurable function $X\colon \mathcal{T}\times\Omega \to \mathcal{S}$.

Definition 1.3.2 (Continuous-time stochastic process)**.** *A stochastic process is said to be a continuous-time stochastic process if the index set $\mathcal{T}$ takes continuous values.*

Definition 1.3.3. *Let $(X_t)_{t\in\mathcal{T}}$ be a continuous-time stochastic process. For $\omega \in \Omega$, the function $t\colon X_t(\omega)$ is known as a realization or a sample path of this stochastic process.*

Definition 1.3.4. *The stochastic process is said to be continuous if the sample paths are continuous, for all $\omega \in \Omega$. The stochastic process is said to be almost surely continuous if, for almost all $\omega \in \Omega$, the sample paths are continuous. Similar definitions apply to left and right continuous stochastic processes.*

In most of this work, the index set is the set of positive real numbers $\mathbb{R}^+$.

Definition 1.3.5 (Markov processes)**.** *A continuous-time Markov process on a countable state space $\mathcal{S}$ is a right-continuous stochastic process $(X_t)_{t\geq 0}$ such that $X_t \in \mathcal{S}$ for any $t \geq 0$, and which verifies the Markov property, meaning that for $n \geq 1$ and for any finite collection of states $(x_1, ..., x_{n+1}) \in \mathcal{S}^{n+1}$ and times $0 \leq t_1 < ... < t_{n+1}$, it verifies:*

$$\mathbb{P}\left(X_{t_{n+1}} = x_{n+1} | X_{t_1} = x_1, ..., X_{t_n} = x_n\right) = \mathbb{P}\left(X_{t_{n+1}} = x_{n+1} | X_{t_n} = x_n\right). \tag{1.1}$$

Definition 1.3.6. *For a continuous-time Markov process $(X_t)_{t\geq 0}$, any given $t, s \geq 0$ and $i, j \in \mathcal{S}$, the transition probability $p_{i,j}(t,s)$ is defined by:*

$$p_{i,j}(t,s) = \mathbb{P}\left(X_{t+s} = j | X_s = i\right). \tag{1.2}$$

If $p_{i,j}(t,s)$ does not depend on s for any $i, j \in \mathcal{S}$ and $t \geq 0$, then the Markov process is said to have a homogeneous transition probability and we write $p_{i,j}(t) = \mathbb{P}\left(X_{t+s} = j | X_s = i\right) = \mathbb{P}\left(X_t = j | X_0 = i\right)$. In this case, we say that the process is homogeneous in time.

A continuous-time Markov process with homogeneous transition probability is completely specified by the knowledge of $p_{i,j}(t)$ for every $t \geq 0$, and $i, j \in \mathbb{N}$.

In what follows, we consider a continuous-time Markov process $(N_t)_{t\geq 0}$ on the state space $\mathbb{N}$ with homogeneous transition probabilities.

Theorem 1.3.1. *Consider a set of transition probabilities $p_{i,j}(t)$ for $t \geq 0$ and $i, j \in \mathbb{N}$. This set can be the transition probabilities of a continuous-time Markov process homogeneous in time if and only if it verifies the three following conditions for all $t \geq 0$ and $i, j \in \mathbb{N}$:*

- $p_{i,j}(t) \geq 0$
- $\sum_{i=0}^{+\infty} p_{i,j}(t) = 1$
- $p_{i,j}(t+s) = \sum_{k=0}^{+\infty} p_{i,k}(s) p_{k,j}(t)$.

Proof. We will not detail the full proof here but rather give some brief insights. The first two conditions are obvious, probabilities must be positive and sum to 1.

Consider a time-homogeneous Markov process $(N_t)_{t\geq 0}$. Let $t, s \geq 0$ and $i, j \in \mathbb{N}$. The probability $\mathbb{P}(N_{t+s} = i | N_0 = j)$ can be written as a sum using the law of total probability. Indeed, at time t, the events $\{N_t = k\}_{k\in\mathbb{N}}$ form a partition of the sample space and then:

$$\begin{aligned}\mathbb{P}(N_{t+s} = i | N_0 = j) &= \sum_{k=0}^{+\infty} \mathbb{P}(N_{t+s} = i | N_t = k, N_0 = j)\, \mathbb{P}(N_t = k | N_0 = j) \\ &= \sum_{k=0}^{+\infty} \mathbb{P}(N_{t+s} = i | N_t = k)\, \mathbb{P}(N_t = k | N_0 = j) \end{aligned} \tag{1.3}$$

since $\mathbb{P}(N_{t+s} = i | N_t = k, N_0 = j) = \mathbb{P}(N_{t+s} = i | N_t = k)$.

Now if we consider such probabilities, and define a stochastic process $(N_t)_{t\geq 0}$ with a probability distribution $\mathbb{P}(N_{t+s} = j | N_s = i) = p_{i,j}(t)$ for all $i, j \in \mathbb{N}$ and $t, s \geq 0$, one can check that the process is well-defined and verifies the Markov property (1.1). □

Remark. The third condition is known as the **Chapman-Kolmogorov equation**.

Remark. Usually, we add a fourth condition $\lim_{t\to 0^+} p_{i,i}(t) = 1$ which guarantees the process cannot jump an infinite number of times in a finite time interval. This is especially important to describe physical processes.

Definition 1.3.7 (Transition rate matrix)**.** *The transition rate matrix of the continuous-time Markov process is the function*

$$\begin{aligned} Q\colon \mathbb{N}\times\mathbb{N} &\longrightarrow \mathbb{R} \\ (i,j) &\longmapsto q_{i,j} \end{aligned}$$

which is defined for $i \neq j \in \mathbb{N}$ as:

$$q_{i,j} = \lim_{t\to 0^+} \frac{p_{i,j}(t)}{t} = \left(\frac{dp_{i,j}}{dt}\right)_{|t=0^+} \tag{1.4}$$

where we assume that the limit exists.

The transition rate matrix defines the rate at which the process transitions from a state i to a state j.

1.3.1.2 Kolmogorov equations

Theorem 1.3.2 (Forward Kolmogorov equation)**.** *If $(N_t)_{t\geq 0}$ is a continuous-time Markov process homogeneous in time on the state space $\mathbb{N}$ with transition rate matrix Q, and if we denote the transition probabilities by $p_{i,j}(t) = \mathbb{P}(N_t = j | N_0 = i)$ for $i, j \in \mathbb{N}$ and $t \geq 0$, then the evolution of the transition probabilities is governed by the following differential equation:*

$$\frac{dp_{i,j}(t)}{dt} = \sum_{k\neq j} q_{k,j} p_{i,k}(t) - p_{i,j}(t) \sum_{k\neq j} q_{k,j}. \tag{1.5}$$

Proof. To prove this equation we set $t \geq 0$ and $i, j \in \mathbb{N}$ and consider what is happening at $t + s$ where $s \geq 0$. From the Chapman-Kolmogorov equation:

$$p_{i,j}(t+s) = \sum_{k=0}^{+\infty} p_{i,k}(t) p_{k,j}(s). \tag{1.6}$$

Subtracting $p_{i,j}(t)$ gives us:

$$p_{i,j}(t+s) - p_{i,j}(t) = p_{i,j}(t)(p_{j,j}(s) - 1) + \sum_{k \neq j} p_{i,k}(t) p_{k,j}(s). \tag{1.7}$$

Rewriting $p_{j,j}(s) = 1 - \sum_{k \neq j} p_{j,k}(s)$:

$$p_{i,j}(t+s) - p_{i,j}(s) = -p_{i,j}(t) \sum_{k \neq j} p_{j,k}(s) + \sum_{k \neq j} p_{i,k}(t) p_{k,j}(s). \tag{1.8}$$

Dividing by s and going to the limit $s \to 0^+$, we can conclude that:

$$\frac{dp_{i,j}(t)}{dt} = \sum_{k \neq j} q_{k,j} p_{i,k}(t) - p_{i,j}(t) \sum_{k \neq j} q_{k,j}. \tag{1.9}$$

□

Similarly, one can derive a backward equation.

Theorem 1.3.3 (Backward Kolmogorov equation)**.** *With the same notations, the transition probabilities verify the Kolmogorov backward equation given by:*

$$\frac{dp_{i,j}(t)}{dt} = \sum_{k \neq i} q_{i,k} p_{k,j}(t) - p_{i,j}(t) \sum_{k \neq i} q_{i,k}. \tag{1.10}$$

Proof. The derivation is very similar to the previous equations. The only difference is that $p_{i,j}(t+s)$ is expressed as:

$$p_{i,j}(t+s) = \sum_{k=0}^{+\infty} p_{i,k}(s) p_{k,j}(t). \tag{1.11}$$

The rest of the proof is straightforward. □

Now, let us focus more specifically on Poisson processes, a particular example of continuous-time Markov processes, often used to describe memoryless nuclear reactions.

1.3.2 Poisson processes

Definition 1.3.8 (Counting process)**.** *A counting process is a non-decreasing continuous-time stochastic process $(N_t)_{t \geq 0}$ taking positive integer values. More formally, it verifies the two following conditions:*

- *For any $t \geq 0$, $N_t \in \mathbb{N}$.*
- *For any $0 \leq s \leq t$, $N_s \leq N_t$.*

Definition 1.3.9 (Homogeneous Poisson point process)**.** *A Poisson point process with rate $\lambda \in \mathbb{R}^+$ is a counting process $(N_t)_{t \geq 0}$ which verifies:*

- $N_0 = 0$.

- *It has independent increments, meaning that for any $n \in \mathbb{N}$ and for any collection $0 \leq t_1 < ... < t_n$, the random variables $\left(N_{t_{k+1}} - N_{t_k}\right)_{1 \leq k \leq n-1}$ are mutually independent.*
- *For any $t, s > 0$, the random variable $(N_{t+s} - N_s)$ follows a Poisson distribution with mean λt.*

In that case, we introduce the notation $(N_t)_{t \geq 0} \sim \mathcal{PP}(\lambda)$.

Proposition 1.3.1. *For homogeneous Poisson point processes, the jump instants are given by exponential distributions.*

Proof. Consider $(N_t)_{t \geq 0} \sim \mathcal{PP}(\lambda)$ and for $t \geq 0$, let X_t be the random variable defining the time of a new event occurring, assuming one occurred at time t. Then for $s \geq 0$:

$$\mathbb{P}\left(X_t \leq s\right) = 1 - \mathbb{P}\left(X_t > s\right) = 1 - \mathbb{P}\left(N_{t+s} - N_t = 0\right). \tag{1.12}$$

By definition, $N_{t+s} - N_t$ has the same law as N_s since $N_0 = 0$ and N_s follows a Poisson distribution with parameter λs meaning that:

$$\mathbb{P}\left(X_t \leq s\right) = 1 - \mathbb{P}\left(N_{t+s} - N_t = 0\right) = 1 - e^{-\lambda s} \tag{1.13}$$

which is the expression of the CDF of the exponential distribution with parameter λ. One can then conclude that $X_t \sim \mathcal{E}(\lambda)$, that is X_t follows an exponential distribution with parameter λ. □

It is possible to define a compound Poisson process in which the jump magnitudes themselves are random.

Definition 1.3.10 (Homogeneous compound Poisson process)**.** *A compound Poisson process with rate $\lambda \in \mathbb{R}^+$ and jump distribution ν is a stochastic process $(Y_t)_{t \geq 0}$ defined for $t \geq 0$ by:*

$$Y_t = \sum_{i=0}^{N(t)} J_i \tag{1.14}$$

where $(N(t))_{t \geq 0} \sim \mathcal{PP}(\lambda)$ and the $(J_i)_{i \in \mathbb{N}}$ are independent and identically distributed random variables with distribution ν. In that case we write $(Y_t)_{t \geq 0} \sim \mathcal{P}(\lambda, \nu)$.

Poisson processes in general describe nuclear reactions, which have the known property of being memoryless. Compound Poisson processes are used specifically to describe spontaneous fission neutron sources. The spontaneous fission occurs at a random time given by a memoryless exponential distribution and produces a random number of neutrons given by a known multiplicity distribution.

1.3.3 Branching processes

Galton-Watson processes, also named branching processes were first introduced as discrete-time stochastic processes to model the survival of family names along generations. Since then, they have been extended to continuous time and have been widely used to model nuclear reactions and genetic evolution.

Intuitively, to model nuclear reactions, we would like our process to induce some reaction at some random times. These reactions should follow exponential distributions

to satisfy the memoryless property of nuclear reactions. Besides, each reaction gives birth to multiple new particles, which can then be seen as independent sub-branches following the same dynamics.

Such stochastic processes are known as Markov branching processes. They are the focus of this section.

Definition 1.3.11 (Markov branching process)**.** *If $(N_t)_{t\geq 0}$ is a continuous-time Markov process and if for all $t \geq 0$ the transition probabilities $(p_{i,j}(t))_{i,j\in\mathbb{N}}$ verify for all $z \in \mathbb{R}$ such that $|z| \leq 1$:*

$$\sum_{j=0}^{+\infty} p_{i,j}(t) z^j = \left(\sum_{j=0}^{+\infty} p_{1,j}(t) z^j \right)^i \tag{1.15}$$

then $(N_t)_{t\geq 0}$ is known as a Markov branching process or a continuous-time branching process.

Equation (1.15) is known as the branching property. Intuitively, it states that the law of the process $\{N_t | N_0 = i\}$ is the same as that of the sum of i independent processes $N_t^{(k)} = \{N_t | N_0 = 1\}$ with $1 \leq k \leq i$. The process is Markovian and the branching events produce independent branching chains.

Proposition 1.3.2. *If $(N_t)_{t\geq 0}$ is a Markov branching process, then there exists $\lambda > 0$ and a discrete probability distribution f with probabilities $(f_k)_{k\in\mathbb{N}}$ such that the transition rate $q_{i,j}$ can be expressed for $i, j \in \mathbb{N}$ and $i \neq j$ as:*

$$q_{i,j} = \begin{cases} i\lambda f_{j+1-i} & \text{if } j \geq i-1 \\ 0 & \text{otherwise.} \end{cases} \tag{1.16}$$

The probability distribution f is known as the offspring distribution and λ is the rate of the branching reaction. The branching process is denoted by $(N_t)_{t\geq 0} \sim \mathcal{B}(f, \lambda)$.

Proof. Let $i, j \in \mathbb{N}$ such that $i \neq j$. We study the probability $p_{i,j}(dt) = \mathbb{P}(N_{dt} = j | N_0 = i)$. The generating function of the random variable $(N_t | N_0 = i)$ is the left term of (1.15). The right term corresponds to the branching property of $S_i = \sum_{k=1}^{i} N_t^{(k)}$ where the $(N_t^{(k)})_{1\leq k\leq i}$ are independent random variables with the same law as $(N_t | N_0 = 1)$. Thus, $(N_t | N_0 = i)$ has the same law as S_i. One can then deduce:

$$\begin{aligned} p_{i,j}(dt) &= \mathbb{P}(N_{dt} = j | N_0 = i) \\ &= \sum_{n_1+\ldots+n_i=j} \prod_{k=1}^{i} \mathbb{P}(N_{dt}^{(k)} = n_k) \\ &= \sum_{n_1+\ldots+n_i=j} \prod_{k=1}^{i} \left(p_{1,n_k}(0) + q_{1,n_k} dt + o(dt) \right). \end{aligned} \tag{1.17}$$

One may notice that the term in the product is order 0 in dt only if $n_k = 1$. Since $p_{i,j}(dt) = q_{i,j}dt + o(dt)$ by definition, one can keep only the terms of order 1 in dt.

We first consider the case $j < i - 1$. Since $n_1 + \ldots + n_i = j$ in the sum, then necessarily there exists at least k_1, k_2 such that $n_{k_1} \neq 1$ and $n_{k_2} \neq 1$. Thus the right term is $\mathcal{O}(dt^2)$. Since $p_{i,j}(dt) = q_{i,j}dt + o(dt)$ this implies necessarily that $q_{i,j} = 0$.

We now focus on the case $j \geq i - 1$. At first order, since $p_{1,n_k}(0) = \delta_{1,n_k}$, the sum simplifies to i identical terms. Indeed, at the first order w.r.t. there exists $1 \leq k_0 \leq i$

such that $n_{k_0} \neq 1$ and all the others $n_k = 1$. Since $n_1 + ... + n_i = j$ then $n_{k_0} = j + 1 - i$ necessarily. From this, one can thus write:

$$q_{i,j}dt + o(dt) = i(q_{1,j+1-i}\, dt) + o(dt). \tag{1.18}$$

Going to the limit $dt \to 0^+$ yields $q_{i,j} = iq_{1,j+1-i}$. To retrieve equation (1.16), one only has to normalize the $q_{1,k}$ by defining:

$$\begin{cases} f_k = \frac{q_{1,k}}{\sum\limits_{p\neq 1} q_{1,p}} & \text{for } k \geq 1 \\ f_k = 1 - \sum\limits_{p\neq 1} f_p & \text{for } k = 1 \end{cases} \tag{1.19}$$

$$\lambda = \sum_{k\neq 1} q_{1,k}. \tag{1.20}$$

□

From a qualitative point of view, λ is the rate at which the process induces branching events, which are memoryless. Thus the interval between two branching events follows an exponential distribution. On the other hand, the probability f_k represents the probability that the branching event produces k new branches, for $k \geq 0$.

In what follows we consider a continuous-time branching process $(N_t)_{t\geq 0} \sim \mathcal{B}(f, \lambda)$.

Definition 1.3.12. *The probability-generating function (PGF) of the process $(N_t)_{t\geq 0}$ is the function g_{n_0} defined for $t \geq 0$, $|z| \leq 1$ and $n_0 \in \mathbb{N}$ by:*

$$g_{n_0}(z,t) = \mathbb{E}_{n_0}\left[z^{N_t}\right] \tag{1.21}$$

where $\mathbb{E}_{n_0}$ is the expectation conditioned by $\{N_0 = n_0\}$.

Proposition 1.3.3. *The generating function g_1 is governed by the differential equation:*

$$\frac{\partial g_1}{\partial t} + \lambda g_1(z,t) = \lambda g_f\left(g_1(z,t)\right) \tag{1.22}$$

where g_f is the PGF of the offspring distribution defined for $|z| \leq 1$ by $g_f(z) = \sum\limits_{n=0}^{+\infty} f_n z^n$.

Proof. This result is obtained by the backward Kolmogorov equation applied to the transition probabilities $p_{1,j}(t)$ and summing over $j \in \mathbb{N}$:

$$\frac{\partial g_1}{\partial t} + \lambda g_1(z,t) = \lambda \sum_{j=0}^{+\infty}\sum_{k=0}^{+\infty} f_k p_{k,j}(t) z^j. \tag{1.23}$$

Inverting the sum and using the branching property (1.15) we have:

$$\frac{\partial g_1}{\partial t} + \lambda g_1(z,t) = \lambda \sum_{k=0}^{+\infty} f_k g_1(z,t)^k = \lambda g_f(g_1(z,t)) \tag{1.24}$$

which concludes the proof. □

Definition 1.3.13 (Multiplication factor)**.** *The multiplication factor is the quantity k defined by $k = \left(\frac{dg_f}{dz}\right)_{z=1}$. The process has three distinct regimes which are:*

- *supercritical if* $k > 1$.
- *critical if* $k = 1$.
- *subcritical if* $k < 1$.

The multiplication factor defines the overall behavior of the chain and whether or not the trend is exponentially increasing, stable, or exponentially decreasing. This multiplication factor finds its analog in neutron noise analysis with the prompt multiplication factor which is introduced later in this chapter. We will provide a few typical properties of the process based on the value of the multiplication factor, but this is not meant to be a comprehensive study and we refer to more complete works on the subject for in-depth results such as [AN04; Har+63].

Proposition 1.3.4. *The mean* $M(t) = \mathbb{E}\left[N_t | N_0 = 1\right]$ *of the process exists and is given by:*

$$M(t) = e^{\lambda(k-1)t}. \tag{1.25}$$

Besides, if $m_{2,f} = \left(\frac{d^2 g_f}{dz^2}\right)_{z=1} < +\infty$, *then the variance* $V(t) = \mathbb{V}\left[N_t | N_0 = 1\right]$ *exists and is given by:*

$$V(t) = \begin{cases} \lambda m_{2,f} \times t & \text{if } k = 1 \\ \left(\frac{m_{2,f}}{k-1} - 1\right)\left(e^{2\lambda(k-1)t} - e^{\lambda(k-1)t}\right) & \text{otherwise .} \end{cases} \tag{1.26}$$

Proof. Let us differentiate equation (1.21) w.r.t. z and set $z = 1$ which yields:

$$\frac{dM}{dt} = \lambda(k-1)M(t). \tag{1.27}$$

Solving this ordinary differential equation with the initial condition $M(0) = 1$ gives us (1.25). We proceed similarly for $V(t)$ by differentiating two times w.r.t. z and defining $M_2(t) = \mathbb{E}\left[N_t(N_t - 1)|N_0 = 1\right]$:

$$\frac{dM_2}{dt} = \lambda(k-1)M_2(t) + m_{2,f}M(t)^2. \tag{1.28}$$

Solving this equation gives us:

$$M_2(t) = \begin{cases} \lambda m_{2,f} \times t & \text{if } k = 1 \\ \frac{m_{2,f}}{k-1}\left(e^{2\lambda(k-1)t} - e^{\lambda(k-1)t}\right) & \text{otherwise} \end{cases} \tag{1.29}$$

and we can conclude with the relation $V(t) = M_2(t) + M(t) - M(t)^2$. □

This short introduction to Markov branching processes is meant to provide enough notions to describe branching processes in neutron noise studies in a simplified context, which is done in the next sections.

1.3.4 Neutron evolution as a branching process

Consider the specific case of neutron dynamics in a multiplying medium without a source. We consider monoenergetic neutrons, in an infinite medium that can either be captured with a corresponding reaction intensity λ_c or induce fissions with an

intensity λ_f. The random variable which describes the number of offspring neutrons generated by a single fission is denoted by ν. For $i \in \mathbb{N}$, the probabilities of the offspring distribution are introduced as $f_i = \mathbb{P}(\nu = i)$. The mean is denoted by $\overline{\nu} = \sum_{i=0}^{+\infty} i f_i$.

Since the nuclear reactions are independent, the fission and absorption random times are independent. These times are given by exponential distributions with parameters λ_c and λ_f. The minimum of these two random variables is thus given by an exponential distribution with parameter $\lambda_t = \lambda_c + \lambda_f$ which represents the total reaction intensity.

Let us denote by $(N_t)_{t\geq 0}$ the evolution of the neutron population. This process is a branching process due to the memorylessness of nuclear reactions. In this section, we will leverage the theory of Markov branching processes by identifying the physical parameters to the concepts introduced in the previous paragraph.

For that purpose, let us retrieve the Kolmogorov backward equation by calling on physics intuition. For $n \in \mathbb{N}$ and $t \geq 0$, $p_n(t)$ is the probability of having n neutrons in the system, with initial condition $N_0 = 1$. We want to write this probability as a function of what happened in the infinitesimal interval $[t, t + dt]$.

Consider that we have n events at $t + dt$ and only one reaction can occur in $[t, t + dt]$ (we are neglecting terms in $o(dt^2)$). Then, the possible events are the following:

- There were n neutrons at t and no reaction occurred.
- There were $n + 1$ neutrons at t and a capture reaction occurred.
- There were $n - k$ neutrons at t for some $k \leq n - 1$ and a fission occurred and produced k neutrons.

This partition of the event space leads to the following equation with the law of total probability:

$$p_n(t + dt) = (1 - \lambda_c dt - \lambda_f dt)p_n(t) + \lambda_c dt p_{n+1}(t) + \lambda_f dt \sum_{k=0}^{n} f_k p_{n+1-k}(t). \quad (1.30)$$

Going to the limit $dt \to 0^+$, we have:

$$\frac{dp_n}{dt} + \lambda_t p_n(t) = \lambda_f \sum_{k=0}^{n} f_k p_{n+1-k}(t) + \lambda_c p_{n+1}(t). \quad (1.31)$$

Identifying the Kolmogorov equation to the intensity rates in (1.16), one can then express the generating function g_f as:

$$g_f(z) = \frac{\lambda_c}{\lambda_t} + \frac{\lambda_f}{\lambda_t} \sum_{k=1}^{+\infty} f_k z^k \quad (1.32)$$

This generating function is the generating function of the offspring distribution for the case where we consider capture and fission reactions in the medium. It allows us to define the multiplication factor.

Definition 1.3.14 (Prompt multiplication)**.** *The prompt multiplication factor k_p is defined by:*

$$k_p = \frac{\overline{\nu}\lambda_f}{\lambda_c + \lambda_f}. \tag{1.33}$$

The process is said to be subcritical if $k_p < 1$, critical if $k_p = 1$, and supercritical if $k_p > 1$.

Remark. Throughout this work the effects of delayed neutrons are neglected, thus the denomination of prompt multiplication factor k_p instead of effective multiplication factor. The effects of delayed neutrons on neutron fluctuations and multiplicity counting have been described in [PP07] for example.

Proposition 1.3.5. *We introduce the prompt decay constant $\alpha = \lambda_t(k_p - 1)$. Assuming one initial neutron $N_0 = 1$, the mean and variance of the neutron population is given by:*

$$\mathbb{E}\left[N_t | N_0 = 1\right] = e^{\alpha t} \tag{1.34}$$

$$\mathbb{V}\left[N_t | N_0 = 1\right] = \begin{cases} \lambda_f \nu_2 t & \text{if } k_p = 1 \\ \left(\nu_2 \frac{k_p/\overline{\nu}}{k_p - 1} - 1\right)\left(e^{2\alpha t} - e^{\alpha t}\right) & \text{otherwise} \end{cases} \tag{1.35}$$

where $\nu_2 = \sum_{i=0}^{+\infty} \frac{i(i-1)}{2} f_i$ is the binomial moment of order 2 of the fission distribution.

Proof. This is easily derived from equations (1.25) and (1.26). □

This simple introduction to stochastic processes in neutron noise analysis is useful to grasp the main concepts and methods involved. In the next section, our focus is on a more complex setting, in which we will introduce the number of detected neutrons and study the influence of a neutron source. The general methodology though, is the same as what was presented in this section. We derive Kolmogorov equations and deduce a PDE for the generating function, which allows us to obtain the moments of the distribution. At the end of the section, the main results for gamma correlations are also introduced though they are not formally derived.

1.4 A study of neutron and gamma fluctuations

This section presents a study on the fluctuations of neutrons and gamma rays in a low-power multiplying system. The neutron and gamma particles are modeled as stochastic processes and our goal is to study the evolution of their population. We will also derive the moments of the distribution of neutrons (and gammas) which are quantities of interest in the rest of this thesis.

In the majority of this section, the focus is on neutrons. In the first paragraph, the framework of the mathematical model is presented. Then the Kolmogorov forward and backward equations are derived for the case of a single initial neutron, and the influence of the source is then discussed. The derivation of the moments and the multiplicity rates are then obtained for the neutrons, both with and without source terms. Finally, the gamma multiplicity rates are derived in the last part and the corresponding moments are obtained. Most of the work developed in this section is adapted from [PP07]. The methodologies were initially developed by [Pal58] and [Bel65].

1.4.1 The point model framework

The neutron transport equation describes the evolution of the neutron flux ϕ. In the presence of a single fissile nuclide, the general form of this equation is:

$$\left(\frac{1}{v}\frac{\partial}{\partial t} + \vec{\Omega}\cdot\vec{\nabla} + \Sigma_t(\vec{r},\vec{\Omega},E,t)\right)\phi(\vec{r},\vec{\Omega},E,t) = \int_0^E dE' \int_{4\pi} d\vec{\Omega}'\Sigma_s(\vec{r},\vec{\Omega}'\to\vec{\Omega},E'\to E,t)\phi(\vec{r},\vec{\Omega},E,t) + \frac{\chi(E)}{4\pi}\int_0^{+\infty} dE' \int_{4\pi} d\vec{\Omega}'\nu(E')\Sigma_f(\vec{r},E',t)\phi(\vec{r},\vec{\Omega},E,t) \tag{1.36}$$

where $\vec{r}$ is the position vector and $\vec{\Omega}$ is the solid angle. Σ_s, Σ_f, and Σ_t are respectively the scattering, fission, and total macroscopic cross sections. $\nu(E')$ is the average number of neutrons produced by an induced fission with incident neutron energy E' and $\chi(E)$ is the density of the energy distribution of the neutrons produced by induced fissions.

This equation is at the heart of reactor analysis and neutron studies in general. However, the integro-differential form of the equation, added to the large number of unknowns makes it impossible to solve without approximations or simplifying assumptions. Yet, an analytical description of neutron fluctuations in zero-power systems is possible with the following assumptions, which were first introduced in [Bel65] and [Pal58]. They make up the so-called **point model theory**.

Assumption 1. *The medium is homogeneous and infinite. There is no leakage of neutrons.*

Assumption 2. *Neutrons all have the same energy. The dependence in E of the cross-sections is removed.*

Assumption 3. *Only two reactions are considered: fission and capture. If a neutron is captured, it is detected by the detector with some fixed probability.*

These assumptions are widely used in neutron noise analysis. We will use them extensively in the rest of this chapter.

1.4.2 A single initial neutron

Our goal is now to derive the evolution of the neutron population and to obtain quantities of interest, among which are the binomial and Feynman moments of the distribution. We are working in the point model framework, with all the assumptions described in the previous paragraph.

We start our study with a single initial neutron. The neutron population is defined by a continuous-time discrete stochastic process $(N_t)_{t\geq 0}$ where $N_t \in \mathbb{N}$ for $t \geq 0$. We suppose for now that $N_0 = 1$. We also introduce another integer-valued stochastic process $(D_t)_{t\geq 0}$ which models the number of detected neutrons at time t.

We assume that the number of neutrons produced by induced fissions is given by an integer-valued random variable ν and we define $f_i = \mathbb{P}(\nu = i)$ for $i \in \mathbb{N}$. These quantities are known.

Definition 1.4.1. *Let $T \geq 0$. The detection system is modeled by a detector efficiency ε_C given for $t \geq 0$ by:*

$$\varepsilon_C(t) = \begin{cases} \varepsilon & \text{if } t \in [0, T] \\ 0 & \text{if } t > T. \end{cases} \tag{1.37}$$

Definition 1.4.2. *The macroscopic cross sections for the fission and capture reactions are denoted by Σ_f and Σ_c. Let us also define the total cross section as $\Sigma_t = \Sigma_c + \Sigma_f$ (only capture and fission reactions are considered). With our assumptions, these quantities are constant.*

Definition 1.4.3. *The reaction rates (for one neutron) λ_f and λ_c for the fission and capture reactions are defined by $\lambda_f = v\Sigma_f$ and $\lambda_c = v\Sigma_c$ where v is the average neutron velocity in the medium. Similarly, the total reaction rate is $\lambda_t = v\Sigma_t$.*

Remark. The quantity $\lambda_f dt$ is the probability for one neutron to induce fission in a time interval dt. Similarly, $\lambda_c dt$ is the probability that a neutron is captured in an infinitesimal time dt.

1.4.2.1 Master equation with one neutron

In this section, we are interested in the number of detected neutrons at a given time T depending on the number of neutrons in the medium at a previous instant t. Let us introduce some notations.

Definition 1.4.4. *We introduce the notation $p(n, t|n', t') = \mathbb{P}\left(D_t = n | N_{t'} = n'\right)$ for $t \geq t' \geq 0$ and $n, n' \in \mathbb{N}$. It is the probability of detecting n neutrons at t knowing that n' neutrons were present in the system at $t' \leq t$. Similarly, let us introduce the notation $p_n(t) = p(n, T|1, T - t)$.*

Proposition 1.4.1. *The probability $p(n, T|1, t - dt)$ for $n \in \mathbb{N}$ and $t \leq T$ with an infinitesimal time dt is given by:*

$$\begin{aligned} p(n, T|1, t - dt) = (1 - \lambda_t dt) p(n, T|1, t) + \lambda_c dt \left(\varepsilon_C(t)\delta_{n,1} + (1 - \varepsilon_C(t))\delta_{n,0}\right) \\ + \lambda_f dt \sum_{i=0}^{+\infty} f_i p(n, T|i, t). \end{aligned} \tag{1.38}$$

Proof. We will consider a backward approach. Assuming that we still have one neutron in the system at time $t - dt$ with an infinitesimal time dt. How can we estimate the probability $p(n, T|1, t - dt)$?

In the interval $[t - dt, t]$, a single neutron at time $t - dt$ can be captured in the interval $[t - dt, t]$ meaning that we have no neutrons left at time t. If this event is denoted as $\mathcal{C}_t$, the probability $p(n, T|1, t - dt, \mathcal{C}_t)$ is given by:

$$p(n, T|1, t - dt, \mathcal{C}_t) = \begin{cases} \varepsilon_C(t) & \text{if } n = 1 \\ 1 - \varepsilon_C(t) & \text{if } n = 0 \\ 0 & \text{otherwise.} \end{cases} \tag{1.39}$$

This probability can be rewritten using the Kronecker symbol $\delta_{i,j} = \mathbb{1}_{i=j}$:

$$p(n, T|1, t - dt, \mathcal{C}_t) = \varepsilon_C(t)\delta_{n,1} + (1 - \varepsilon_C(t))\,\delta_{n,0}. \tag{1.40}$$

The probability of the event $\mathcal{C}_t$ conditioned by $\{N_{t-dt} = 1\}$ denoted as $p(\mathcal{C}_t|1, t - dt)$ is given by $p(\mathcal{C}_t|1, t - dt) = \lambda_c dt$.

Then, we consider the case where no reaction occurs in the interval $[t-dt,t]$. This event is denoted as $\mathcal{N}_t$. Conditioned by that event, the probability $p(n,T|1,t-dt,\mathcal{N}_t)$ is:

$$p(n,T|1,t-dt,\mathcal{N}_t) = p(n,T|1,t). \tag{1.41}$$

The neutron population is unchanged and the conditional probability only depends on $p(n,T|1,t)$. The probability of the event $\mathcal{N}_t$ conditioned by $\{N_{t-dt}=1\}$, which is denoted as $p(\mathcal{N}_t|1,t-dt)$ is expressed as:

$$p(\mathcal{N}_t|1,t-dt) = (1-\lambda_t dt) \tag{1.42}$$

since it requires that no other reaction occurs in the time interval of length dt.

Finally, the initial neutron may induce a fission in the interval $[t-dt,t]$. This event is denoted as $\mathcal{F}_t$. In that case, we have i neutrons in the medium at time t with probability f_i. The events of having i neutrons present at time t are all mutually independent such that the conditional probability $p(n,T|1,t-dt,\mathcal{F}_t)$ can be written as:

$$p(n,T|1,t-dt,\mathcal{F}_t) = \sum_{i=0}^{+\infty} f_i p(n,T|i,t). \tag{1.43}$$

Besides, the probability of the event $\mathcal{F}_t$ conditioned by $\{N_{t-dt}=1\}$, which is denoted as $p(\mathcal{F}_t|1,t-dt)$ is given by:

$$p(\mathcal{F}_t|1,t-dt) = \lambda_f dt. \tag{1.44}$$

Finally, by remarking that the events $\mathcal{C}_t$, $\mathcal{N}_t$ and $\mathcal{F}_t$ are mutually exclusive and form a partition of the space of possible events, one can use the law of total probability to obtain $p(n,T|1,t-dt)$:

$$\begin{aligned} p(n,T|1,t-dt) &= p(n,T|1,t,\mathcal{C}_t)p(\mathcal{C}_t|1,t-dt) \\ &+ p(n,T|1,t,\mathcal{N}_t)p(\mathcal{N}_t|1,t-dt) \\ &+ p(n,T|1,t,\mathcal{F}_t)p(\mathcal{F}_t|1,t-dt) \end{aligned} \tag{1.45}$$

which gives equation (1.38). □

This backward equation can then be used to derive a differential equation for the evolution of the probability $p_n(t)$.

Proposition 1.4.2. *Let $\overline{\nu} = \sum_{i=0}^{+\infty} i f_i$ be the average number of neutrons produced by induced fissions. The prompt multiplication factor is given in (1.33) by $k_p = \frac{\overline{\nu}\lambda_f}{\lambda_t}$. The probability $p_n(t)$ is governed by the differential equation:*

$$\lambda_t^{-1}\frac{dp_n}{dt} + p_n(t) = \frac{k_p}{\overline{\nu}}\sum_{i=0}^{+\infty} f_i p_{n,i}(t) + \left(1-\frac{k_p}{\overline{\nu}}\right)\left(\varepsilon_C(t)\delta_{n,1} + (1-\varepsilon_C(t))\delta_{n,0}\right) \tag{1.46}$$

where $p_{n,i}(t) = \sum_{n_1+..+n_i=n} p_{n_1}(t)...p_{n_i}(t)$.

Proof. Let $t \geq 0$. Starting from equation (1.38) and applying to $t' = T - t$ yields:

$$p_n(t+dt) = (1-\lambda_t dt)p_n(t) + \lambda_f dt \sum_{i=0}^{+\infty} f_i p(n,T|i,T-t) \\ - \lambda_c dt \left(\varepsilon_C(T-t)\delta_{n,1} + (1-\varepsilon_C(T-t))\delta_{n,0}\right). \tag{1.47}$$

Then, taking the limit $dt \to 0^+$, one can obtain the following differential equation:

$$\frac{dp_n}{dt} + \lambda_t p_n(t) = \lambda_f \sum_{i=0}^{+\infty} f_i p(n,T|i,T-t) + \lambda_c \left(\varepsilon_C(T-t)\delta_{n,1} + (1-\varepsilon_C(T-t))\delta_{n,0}\right). \tag{1.48}$$

Let us introduce $e_C(t) = \varepsilon_C(T-t)$. Dividing by λ_t and replacing respectively $\frac{\lambda_f}{\lambda_t}$ and $\frac{\lambda_c}{\lambda_t}$ by $\frac{k_p}{\overline{\nu}}$ and $\left(1 - \frac{k_p}{\overline{\nu}}\right)$ one can obtain:

$$\lambda_t^{-1}\frac{dp_n}{dt} + p_n(t) = \frac{k_p}{\overline{\nu}} \sum_{i=0}^{+\infty} f_i p(n,T|i,T-t) + \left(1 - \frac{k_p}{\overline{\nu}}\right)\left(e_C(t)\delta_{n,1} + (1-e_C(t))\delta_{n,0}\right). \tag{1.49}$$

Finally, one can rewrite the expression of $p(n,T|i,T-t)$ for $i \geq 0$ by noticing that this probability of detected n neutrons at T given that we have i neutrons in the medium at time $T-t$ can be obtained by summing on the number of detections n_k induced at time T by the k-th neutron present at time $T-t$. In other words, each one of the i individual neutron at time $T-t$ yields a number n_k, for $1 \leq k \leq i$ of detections at time T. To obtain n detections at T we need to have $\sum_{k=0}^{i} n_k = n$.

Consequently, the probability $p(n,T|i,T-t)$ can be written as:

$$p(n,T|i,T-t) = \sum_{n_1+\ldots+n_i=n} p_{n_1}(t)\ldots p_{n_i}(t). \tag{1.50}$$

This equation is only valid for $i \geq 1$. For $i = 0$, one can easily see that $p_{n,0}(t) = \delta_{n,0}$. We extend the previous notation to this specific case. □

Remark. We focus on the backward Kolmogorov equation here, but one can also derive a forward equation with a similar approach.

The master equation cannot be solved as such, one needs to introduce a generating function for the probabilities $(p_n(t))_{n\in\mathbb{N}}$.

1.4.2.2 Equation for the generating function

The quantities of interest in our study are the moments of the distribution from which the Feynman moments are derived. They can be derived from a partial differential equation verified by the generating function.

Definition 1.4.5. *For $p \geq 1$, the detection distribution at T induced by a single neutron in the medium at $T-t$ is described by a stochastic process $(X_t)_{t\geq 0}$ such that $X_t = (D_T|N_{T-t} = 1)$ for $t \geq 0$. The moment of order p of the stochastic process $(X_t)_{t\geq 0}$ is the function $\mathcal{M}_p(t)$ defined for $t \geq 0$ by:*

$$\mathcal{M}_p(t) = \mathbb{E}\left[X_t^p\right] = \sum_{n=0}^{+\infty} p_n(t) n^p. \tag{1.51}$$

Similarly, the binomial moment of order of the detection distribution described by the stochastic process $(X_t)_{t\geq 0}$ is the function $m_p(t)$ defined for $t \geq 0$ by:

$$m_p(t) = \mathbb{E}\left[\binom{X_t}{p}\right] = \sum_{n=0}^{+\infty} p_n(t)\binom{n}{p}. \tag{1.52}$$

with the convention that $\binom{n}{p} = 0$ when $n < p$.

Definition 1.4.6. *The probability-generating function (PGF) of the detection distribution described by the stochastic process $(X_t)_{t\geq 0}$ is the function*

$$\begin{aligned} g\colon \mathbb{C}\times\mathbb{R}^+ &\longrightarrow \mathbb{R} \\ (z,t) &\longmapsto g(z,t) \end{aligned}$$

defined for $|z| \leq 1$ and $t \geq 0$ by $g(z,t) = \sum_{n=0}^{+\infty} p_n(t) z^n$.

Proposition 1.4.3. *The probability-generating function is linked to the binomial moments $(m_p)_{p\geq 1}$ by:*

$$m_p(t) = \frac{1}{p!}\left(\frac{\partial^p g}{\partial z^p}\right)_{z=1^-} \tag{1.53}$$

where the notations $z = 1^-$ refers to the left-limit at 1.

Proof. For any $t \geq 0$, the PGF $g(z,t)$ is a power series and since $\sum_{n=0}^{+\infty} p_n(t) = 1$ by definition, then we know that it has a radius of convergence $R \geq 1$. It is thus $\mathcal{C}^\infty$ on $(-1,1)$ and all the derivatives have a left-limit in $z = 1^-$. Consider $p \geq 1$. Then we have for $|z| \leq 1$ and $t \geq 0$:

$$\frac{\partial^p g}{\partial z^p}(z,t) = \sum_{n=0}^{+\infty} p_n(t)\left(\prod_{k=0}^{p}(n-k)\right) p_n(t) z^{n-p}. \tag{1.54}$$

Dividing by $p!$ and taking the left-limit to 1, we thus obtain:

$$\frac{1}{p!}\frac{\partial^p g}{\partial z^p}(1^-,t) = \sum_{n=0}^{+\infty} p_n(t)\left(\prod_{k=0}^{p}(n-k)\right) p_n(t) = \mathbb{E}\left[\binom{X_t}{p}\right] = m_p(t). \tag{1.55}$$

□

Now we apply the master equation from (1.38) to the PGF.

Proposition 1.4.4. *The probability-generating function g is governed by the following equation for $|z| \leq 1$:*

$$\lambda_t^{-1}\frac{\partial g}{\partial t} + g(z) = \frac{k_p}{\overline{\nu}}\sum_{i=0}^{+\infty} f_i g(z)^i + \left(1 - \frac{k_p}{\overline{\nu}}\right)(e_C(t)(z-1)+1). \tag{1.56}$$

Proof. Let $t \geq 0$ and $|z| \leq 1$. Starting from equation (1.38), we multiply by z^n and sum over n which leads to:

$$\begin{aligned}\lambda_t^{-1}\frac{\partial g}{\partial t} + g(z) &= \frac{k_p}{\overline{\nu}}\sum_{n=0}^{+\infty}\sum_{i=0}^{+\infty} f_i p(n,T|i,T-t) + \left(1-\frac{k_p}{\overline{\nu}}\right)(e_C(t)z + (1-e_C(t))) \\ &= \frac{k_p}{\overline{\nu}}\sum_{n=0}^{+\infty}\sum_{i=0}^{+\infty} f_i p(n,T|i,T-t) + \left(1-\frac{k_p}{\overline{\nu}}\right)(e_C(t)(z-1)+1). \quad (1.57)\end{aligned}$$

To further simplify this, one needs to remark that the PGF associated with the probability $p(n,T|i,T-t) = \sum\limits_{n_1+\ldots+n_i=n} p_{n_1}(t)\ldots p_{n_i}(t)$ is the i-th iterated Cauchy product of the PGF $p_n(t)$. The Cauchy product between two series $\sum\limits_{n=0}^{+\infty} a_n$ and $\sum\limits_{n=0}^{+\infty} b_n$, denoted $\left(\sum\limits_{n=0}^{+\infty} a_n\right) \otimes \left(\sum\limits_{n=0}^{+\infty} b_n\right)$ is the series $\sum\limits_{n=0}^{+\infty} c_n$ given by the coefficients $(c_n)_{n\geq 0}$ defined by:

$$c_n = \sum_{p=0}^{n} a_p b_{n-p}. \quad (1.58)$$

The Cauchy product is the discrete equivalent of the convolution product for integrals. One can thus write the series $\sum\limits_{n=0}^{+\infty} p(n,T|i,T-t)z^n$ has the i-th iterated Cauchy product of $\sum\limits_{n=0}^{+\infty} p_n(t)z^n$ since the z^{n_i} terms in the convolutions multiply up to z^n.

$$\sum_{n=0}^{+\infty} p(n,T|i,T-t)z^n = \left(\sum_{n=0}^{+\infty} p_n(t))z^n\right) \otimes \ldots \otimes \left(\sum_{n=0}^{+\infty} p_n(t))z^n\right). \quad (1.59)$$

Then using Abel's series theorem which states that if $\sum\limits_{n=0}^{+\infty} a_n$ and $\sum\limits_{n=0}^{+\infty} b_n$ are convergent series and if their Cauchy product $\left(\sum\limits_{n=0}^{+\infty} a_n\right) \otimes \left(\sum\limits_{n=0}^{+\infty} b_n\right) = \sum\limits_{n=0}^{+\infty} c_n$ is also convergent, then we have the equality:

$$\sum_{n=0}^{+\infty} c_n = \left(\sum_{n=0}^{+\infty} a_n\right) \times \left(\sum_{n=0}^{+\infty} b_n\right). \quad (1.60)$$

Using this result on equation (1.59), we can finally conclude that:

$$\sum_{n=0}^{+\infty} p(n,T|i,T-t)z^n = \left(\sum_{n=0}^{+\infty} p_n(t))z^n\right)^i = g(z,t)^i \quad (1.61)$$

which concludes the proof. □

Definition 1.4.7. *For $i,n \geq 1$, we define the binomial moments of order n of the neutron distribution induced by i initial neutrons as $m_{n,i}(t)$. Since the probabilities of this distribution are given by $(p(n,T|i,t))_{n\in\mathbb{N}}$, to which correspond the PGF function $(z,t) \mapsto g(z,t)^i$, then these moments are defined as:*

$$m_{n,i}(t) = \frac{1}{n!}\left(\frac{\partial^n g^i}{\partial z^n}\right)_{z=1}. \quad (1.62)$$

1.4.2.3 Binomial moments without source

From the general equation of the generating function, one can derive evolution equations for each of the binomial moments.

Proposition 1.4.5. *We introduce the binomial moments $(\nu_p)_{p\geq 1}$ of the fission multiplicity distribution defined as $\nu_p = \sum_{i=0}^{+\infty} \binom{i}{p} f_i$. In particular, the average number of neutrons produced by induced fissions is denoted by $\overline{\nu} = \nu_1$.*

Moreover, we introduce the Feynman efficiency $\varepsilon_F(t) = \left(\frac{\overline{\nu}}{k_p} - 1\right)\varepsilon_C(t)$, the reactivity $\rho = \frac{k_p - 1}{k_p}$ and $\alpha = \lambda_t^{-1}(1-k_p)$ the prompt reactivity constant. We recall that $\rho < 0$ since we consider subcritical systems in which $k_p < 1$. For $t \geq 0$, the first three binomial moments $m_p(t)$ verify the following differential equations:

$$\alpha^{-1}\frac{dm_1}{dt} + m_1(t) = -\frac{\varepsilon_F(T-t)}{\rho\overline{\nu}} \tag{1.63}$$

$$\alpha^{-1}\frac{dm_2}{dt} + m_2(t) = -\frac{\nu_2}{\rho\overline{\nu}} m_1(t)^2 \tag{1.64}$$

$$\alpha^{-1}\frac{dm_3}{dt} + m_3(t) = -\frac{1}{\rho\overline{\nu}}\left(\nu_3 m_1(t)^3 + 2\nu_2 m_1(t) m_2(t)\right). \tag{1.65}$$

Proof. Differentiating equation (1.56) with respect to z provides the following equation on $m_1(t)$:

$$\lambda_t^{-1}\frac{dm_1(t)}{dt} + m_1(t) = \frac{k_p}{\overline{\nu}}\sum_{i=0}^{+\infty} i f_i m_1(t) g(z)^{i-1} + \left(1 - \frac{k_p}{\overline{\nu}}\right) e_C(t). \tag{1.66}$$

Noticing that $g(1) = 1$, the sum over i simplifies as:

$$\lambda_t^{-1}\frac{dm_1(t)}{dt} + m_1(t) = k_p m_1(t) + \left(1 - \frac{k_p}{\overline{\nu}}\right) e_C(t). \tag{1.67}$$

Dividing by $(1-k_p)$ on both sides yields equation (1.63). The same approach is applied for the two other moments by diving respectively two and three times equation (1.56). □

Remark. The initial conditions for the binomial moments are given by $m_p(0) = 0$ for any ≥ 1 since $g(z,0) = 1$ for all $|z| \leq 1$ because $p_n(0) = \delta_{n,0}$.

With equations (1.63), (1.64) and (1.65), and the initial conditions $m_p(0) = 0$ for all $p \geq 1$, one can now solve the differential equations.

Proposition 1.4.6. *The binomial moments $m_1(t)$ and $m_2(t)$ are given by the following equations:*

$$m_1(t) = \begin{cases} -\frac{\varepsilon_F}{\rho\overline{\nu}}\left(1 - e^{-\alpha t}\right) & \text{if } t \in [0,T] \\ -\frac{\varepsilon_F}{\rho\overline{\nu}}\left(1 - e^{-\alpha t}\right) e^{\alpha(T-t)} & \text{if } t > T \end{cases} \tag{1.68}$$

$$m_2(t) = \begin{cases} -\frac{\nu_2}{\rho\overline{\nu}}\left(-\frac{\varepsilon_F}{\rho\overline{\nu}}\right)^2 \left(1 - e^{-2\alpha t} - 2\alpha t e^{-\alpha t}\right) & \text{if } t \in [0,T] \\ -\frac{\nu_2}{\rho\overline{\nu}}\left(-\frac{\varepsilon_F}{\rho\overline{\nu}}\right)^2 e^{\alpha(T-t)} & \\ \times\left(1 - e^{-2\alpha t} - 2\alpha t e^{-\alpha t} + (1-e^{-\alpha t})^2(e^{\alpha(T-t)} - 1)\right) & \text{if } t > T. \end{cases} \tag{1.69}$$

where we introduced the abuse of notation ε_F as the value of the function $\varepsilon_F(t)$ for $t \in [0, T]$, i.e. when the detection gate is open.

Proof. One can simplify verify that the proposed expression follows the evolution equations (1.63) and (1.64), the continuity at $t = T$ and the initial conditions $m_1(0) = m_2(0) = 0$. □

The results presented here are focused on the binomial moments $m_p(t)$ and were obtained with the help of the PGF. Similar results can be obtained using the moment generating function (MGF) which is defined as $G(z,t) = \sum_{n=0}^{+\infty} p_n(t)e^{zn}$. The MGF provides an easy derivation of the ordinary moments $\mathcal{M}_p(t) = \sum_{n=0}^{+\infty} p_n(t)n^p$ since for any $p \geq 0$:

$$\frac{\partial^p G}{\partial z^p}(0,t) = \mathcal{M}_p(t). \tag{1.70}$$

The evolution equation for G can be obtained with the same approach as for the PGF, and from here evolution equations for the ordinary moments can be obtained. This approach is strictly equivalent and is simply another way to derive the Feynman moments, based on the ordinary moments instead of the binomial moments.

It is also possible to provide a similar framework with the characteristic function $\mathfrak{g}$ defined for $t \geq 0$ and $z \in \mathbb{R}$ by:

$$\mathfrak{g}(z,t) = \sum_{n=0}^{+\infty} p_n(t)e^{izn}. \tag{1.71}$$

This characteristic function can be useful to work with the probabilities $p_n(t)$ since they can be obtained from the Fourier transform of the characteristic function:

$$p_n(t) = \frac{1}{2\pi}\int_{-\pi}^{\pi} e^{-inz}\mathfrak{g}(z,t)dz \tag{1.72}$$

which can be obtained numerically by Fast Fourier Transform (FFT) [CCW16].

1.4.3 Including the source

Let us now consider a spontaneous fission source term.

Definition 1.4.8. *The source is modeled as a homogeneous compound Poisson process $(S_t)_{t\geq 0} \sim \mathcal{P}(S, f_s)$ where S is the intensity of the source (the average number of source events per second) and f_s is the distribution of the neutrons created by the source event. For $X \sim f_s$ and $i \in \mathbb{N}$, the probabilities $\mathbb{P}(X = i)$ are denoted by $f_{s,i}$. For $p \geq 1$, the p-th binomial moment of the source distribution f_s is $\nu_{p,s}$ and it is defined by:*

$$\nu_{p,s} = \sum_{i=0}^{+\infty} \binom{i}{p} f_{s,i}. \tag{1.73}$$

The average number of neutrons produced per fission is denoted by $\overline{\nu_s} = \nu_{1,s}$.

We recall that for $i \in \mathbb{N}$ and $t \geq 0$, $m_{n,i}$ is the n-th binomial moment of the number of detected neutrons given i initial neutrons and no source.

Definition 1.4.9. *Let us define $(X_{s,t})_{t\geq 0}$, a stochastic process that represents the number of detected neutrons at T induced by a single source event at instant $T-t$ for $t \geq 0$. The probability distribution of $(X_{s,t})_{t\geq 0}$ is given by:*

$$p_{s,n}(t) = \sum_{i=0}^{+\infty} f_{s,i} p(n, T|i, T-t). \tag{1.74}$$

For $n \geq 1$, let us define the n-th binomial moment of $(X_{s,t})_{t\geq 0}$ denoted by $m_{s,n}(t)$ which is given for $t \geq 0$ by:

$$m_{s,n}(t) = \sum_{i=0}^{+\infty} f_{s,i} m_{n,i}(t) \tag{1.75}$$

where the definition of $m_{n,i}(t)$ is (1.62).

In this section, we are interested in the average number of n-detections, which is denoted by $\Gamma_n(T)$.

Definition 1.4.10. *The average number of n-detections induced by one source event is given by:*

$$\Gamma_n(T) = \int_0^{+\infty} S m_{s,n}(t) dt = \int_0^{+\infty} S \sum_{i=0}^{+\infty} f_{s,i} m_{n,i}(t) dt. \tag{1.76}$$

Intuitively, $m_{s,n}(t)$ is the average number of n-detections recorded at T induced by a single source event that occurred at $T-t$. Since Sdt source events occur in a time step dt one can then sum over all the past source events. Since source events produce independent branches, the average number of n-detections is just the integral of the averages induced at every time step, thus $\Gamma_n(T) = \int_0^{+\infty} S m_{s,n}(t) dt$.

Definition 1.4.11. *For $n \geq 2$, the Feynman moment of order n, denoted by $Y_n(T)$ is defined as:*

$$Y_n(T) = \frac{n! \Gamma_n(T)}{\Gamma_1(T)}. \tag{1.77}$$

Most of the time, only the second and third Feynman moments are investigated. Thus, it is common practice to introduce the second moment as $Y(T)$ and the third as $X(T)$. Their corresponding asymptotic values for $T \to +\infty$ are denoted Y_∞ and X_∞.

The exact expression of the second Feynman moment is derived in what follows. For the third moment, since the analytical expression is much more cumbersome, we will only express its asymptotic value X_∞. Let us first derive $\Gamma_1(T)$.

Proposition 1.4.7. *The average number of detections $\Gamma_1(T)$ is given by:*

$$\Gamma_1(T) = -\frac{\varepsilon_F \overline{\nu_s} S}{\rho \overline{\nu}} T. \tag{1.78}$$

Proof. We recall that $\Gamma_1(T) = \int_0^{+\infty} S m_{s,1}(t) dt$. Besides, $m_{s,1}(t)$ can be computed explicitly as:

$$\begin{aligned} m_{s,1}(t) &= \sum_{i=0}^{+\infty} f_{s,i} m_{1,i}(t) \\ &= \left(\frac{\partial g}{\partial z}\right)_{z=1} \times \sum_{i=0}^{+\infty} f_{s,i} \times i g(1,t)^{i-1} \\ &= \overline{\nu_s} m_1(t) \end{aligned} \tag{1.79}$$

where we used the fact that $g(1,t) = 1$ for all $t \geq 0$.
From here, we have $\Gamma_1(T) = \int_0^{+\infty} S\overline{\nu_s} m_1(t)dt$ and we just need to integrate $m_1(t)$ using equation (1.68) and the boundary condition $m_n(\infty) = 0$ for all $n \geq 1$.

$$\begin{aligned}
\int_0^{+\infty} m_1(t)dt &= \int_0^T m_1(t)dt + \int_T^{+\infty} m_1(t)dt \\
&= -\frac{\varepsilon_F}{\rho\overline{\nu}} \left(\int_0^T (1 - e^{-\alpha t})dt + \int_T^{+\infty} e^{\alpha(T-t)}(1 - e^{-\alpha t})dt \right) \\
&= -\frac{\varepsilon_F}{\rho\overline{\nu}} \left(T + \frac{e^{-\alpha T} - 1}{\alpha} + \frac{1 + e^{-\alpha T}}{\alpha} \right) \\
&= -\frac{\varepsilon_F}{\rho\overline{\nu}} T. \qquad (1.80)
\end{aligned}$$

Thus $\Gamma_1(T) = -\frac{\varepsilon_F \overline{\nu_s} S}{\rho\overline{\nu}} T$. □

Proposition 1.4.8. *The average neutron count rate denoted by R does not depend on T and is given by:*

$$R = -\frac{\varepsilon_F \overline{\nu_s} S}{\rho\overline{\nu}}. \qquad (1.81)$$

Proof. The proof is easily derived from equation (1.78). □

Let us now introduce the Diven factors, which appear in the expression of the Feynman moments.

Definition 1.4.12 (Diven factors)**.** *The Diven factor of order n, with $n \geq 2$, for the induced fission distribution, which is denoted by D_n is defined as:*

$$D_n = \frac{n!\nu_n}{\overline{\nu}^n}. \qquad (1.82)$$

Similarly, we introduce a Diven factor $D_{n,s}$ for the source distribution.

$$D_{n,s} = \frac{n!\nu_{s,n}}{\overline{\nu_s}^n}. \qquad (1.83)$$

Proposition 1.4.9. *The second Feynman moment $Y(T)$ is given by:*

$$Y(T) = \frac{\varepsilon_F D_2}{\rho^2} \left(1 - \rho \frac{\overline{\nu_s} D_{2,s}}{\overline{\nu} D_2} \right) \left(1 - \frac{1 - e^{-\alpha T}}{\alpha T} \right). \qquad (1.84)$$

Proof. By definition $Y(T) = \frac{2\Gamma_2(T)}{\Gamma_1(T)}$ where $\Gamma_1(T)$ is given by equation (1.78). Let us first compute $m_{s,2}(t)$:

$$\begin{aligned}
m_{s,2}(t) &= \sum_{i=0}^{+\infty} f_{s,i} m_{2,i}(t) \\
&= \sum_{i=0}^{+\infty} \frac{f_{s,i}}{2} \left(i(i-1)g(1,t)^{i-2} \left(\frac{\partial g}{\partial z} \right)_{z=1}^2 + ig(1,t)^{i-1} \left(\frac{\partial^2 g}{\partial z^2} \right)_{z=1} \right) \\
&= \nu_{2,s} m_1(t)^2 + \overline{\nu_s} m_2(t). \qquad (1.85)
\end{aligned}$$

From here, the integrals $I_1(T) = \int_0^{+\infty} m_1(t)^2 dt$ and $I_2(T) = \int_0^{+\infty} m_2(t)dt$ must be computed.

Starting with $I_1(T)$, we get:

$$\begin{aligned} I_1(T) &= \left(\frac{\varepsilon_F}{\rho\overline{\nu}}\right)^2 \left[\int_0^T e^{2\alpha(T-t)}\left(1-e^{-\alpha T}\right)^2 dt + \int_0^T \left(1-e^{-\alpha T}\right)^2 dt\right] \\ &= T\left(\frac{\varepsilon_F}{\rho\overline{\nu}}\right)^2 \left(1-\frac{1-e^{-\alpha T}}{\alpha T}\right) \end{aligned} \tag{1.86}$$

$$\begin{aligned} I_2(T) =& -\frac{\nu_2\varepsilon_F^2}{(\rho\overline{\nu})^3}\int_0^T \left(1-e^{-2\alpha t}-2\alpha t e^{-\alpha t}\right) dt \\ &- \frac{\nu_2\varepsilon_F^2}{(\rho\overline{\nu})^3}\int_T^{+\infty} e^{\alpha(T-t)}\left(1-e^{-2\alpha t}-2\alpha t e^{-\alpha t}+(1-e^{-\alpha t})^2(e^{\alpha(T-t)}-1)\right) dt \\ =& \; T\left(\frac{\nu_2\varepsilon_F^2}{-(\rho\overline{\nu})^3}\right)\left(1-\frac{1-e^{-\alpha T}}{\alpha T}\right). \end{aligned} \tag{1.87}$$

From these two integrals, we can then conclude by introducing the Diven factors:

$$Y(T) = \frac{\varepsilon_F D_2}{\rho^2}\left(1-\rho\frac{\overline{\nu_s}D_{2,s}}{\overline{\nu}D_2}\right)\left(1-\frac{1-e^{-\alpha T}}{\alpha T}\right). \tag{1.88}$$

□

Proposition 1.4.10. *The asymptotic second-order Feynman moment is given by:*

$$Y_\infty = \frac{\varepsilon_F D_2}{\rho^2}\left(1-\rho\frac{\overline{\nu_s}D_{2,s}}{\overline{\nu}D_2}\right). \tag{1.89}$$

Proof. Taking equation (1.84) and going to the limit $T \to +\infty$, the result is easily obtained. □

As can be seen from the derivation of $Y(T)$, the computations involved quickly become cumbersome for higher-order Feynman moments. Hence the emphasis is on the asymptotic third Feynman X_∞ instead of the full dependence on T.

Proposition 1.4.11. *The asymptotic third Feynman moment X_∞ is given by:*

$$X_\infty = 3\left(\frac{\varepsilon_F D_2}{\rho^2}\right)\left(1-\rho\frac{\overline{\nu_s}D_{2,s}}{\overline{\nu}D_2}\right) + -\frac{\varepsilon_F D_3}{\rho^3}\left(1-\rho\frac{\overline{\nu_s}^2 D_{3,s}}{\overline{\nu}^2 D_3}\right). \tag{1.90}$$

Proof. By definition $X(T) = \frac{6\Gamma_3(T)}{\Gamma_1(T)}$. Let us first write $m_{s,3}(t)$.

$$\begin{aligned} m_{s,3}(t) &= \sum_{i=0}^{+\infty} f_{s,i} m_{3,i}(t) \\ &= \overline{\nu_s} m_3(t) + 2\nu_{2,s} m_1(t) m_2(t) + \nu_{3,s} m_1(t)^3. \end{aligned} \tag{1.91}$$

The limits of the asymptotic moments are given by:

$$\lim_{t\to+\infty} m_1(t) = m_{1,\infty} = -\frac{\varepsilon_F}{\rho\overline{\nu}} \tag{1.92}$$

$$\lim_{t\to+\infty} m_2(t) = m_{2,\infty} = \frac{\nu_2\varepsilon_F^2}{(-\rho\overline{\nu})^3} \tag{1.93}$$

$$\lim_{t\to+\infty} m_3(t) = m_{3,\infty} = \frac{\nu_3\varepsilon_F^3}{(-\rho\overline{\nu})^4} + \frac{2\nu_2^2\varepsilon_F^3}{(-\rho\overline{\nu})^4}. \tag{1.94}$$

For $m_1(t)$ and $m_2(t)$, the limits can be easily derived from equations (1.68) and (1.69). For $m_3(t)$, one can obtain it by taking the limit in equation (1.65), which sets the derivative to 0. From here we have:

$$\begin{aligned} X_\infty &= \frac{6m_{3,\infty}}{m_{1,\infty}} + \frac{12\nu_{2,s}m_{2,\infty}}{\overline{\nu_s}} + \frac{6\nu_{3,s}m_{1,\infty}^2}{\overline{\nu_s}} \\ &= 3\left(\frac{\varepsilon_F D_2}{\rho^2}\right)\left(1-\rho\frac{\overline{\nu_s}D_{2,s}}{\overline{\nu}D_2}\right) - \frac{\varepsilon_F D_3}{\rho^3}\left(1-\rho\frac{\overline{\nu_s}^2 D_{3,s}}{\overline{\nu}^2 D_3}\right) \end{aligned} \tag{1.95}$$

which concludes the proof. □

We have now derived the expression for the second Feynman moment $Y(T)$ and the asymptotic third Feynman moment X_∞. In this study, the source is assumed to be a homogeneous compound Poisson process which models the spontaneous fission source. In most situations, however, the source term is not just a spontaneous fission term. Indeed, the (α, n) reactions can be predominant in nuclear materials, especially if there is a large proportion of light atoms. In that case, one can still model the source as a compound Poisson but we need to modify the jump probability distribution f_s.

We introduce the parameter x_s which defines the proportion of source events created by spontaneous fissions.

Definition 1.4.13. *Let S_α and S_f be the respective intensities of (α, n) and spontaneous fission reactions. The total source intensity is $S = S_\alpha + \overline{\nu_s}S_f$. We introduce the source description parameter $x_s \in [0,1]$ which is defined as the average ratio of neutrons produced by spontaneous fission events over the total number of source neutrons.*

$$x_s = \frac{\overline{\nu_s}S_f}{S_\alpha + \overline{\nu_s}S_f}. \tag{1.96}$$

Proposition 1.4.12. *To account for the (α, n) source term, we define a new compound Poisson process $(S_t)_{t\geq 0} \sim \mathcal{P}\left(S, f_s^{(C)}\right)$ where the probabilities of the distribution $f_s^{(C)}$ are denoted by $(f_{s,i}^{(C)})_{i\in\mathbb{N}}$. These probabilities are linked to the probability distribution of the spontaneous fission source by:*

$$\begin{cases} f_{s,i}^{(C)} = \frac{S_f}{S_\alpha+S_f} f_{s,i} & \text{if } i \neq 1 \\ f_{s,i}^{(C)} = \frac{S_f f_{s,i}+S_\alpha}{S_\alpha+S_f} & \text{if } i = 1. \end{cases} \tag{1.97}$$

Proof. We consider the two events A = "the source event is an (α, n) reaction" and B = "the source event is a spontaneous fission". They form a partition. Moreover, $\mathbb{P}(A) = \frac{S_\alpha}{S_f+S_\alpha}$ and $\mathbb{P}(B) = \frac{S_f}{S_f+S_\alpha}$.

Consider $X \sim f_s^{(C)}$ and let $i \in \mathbb{N}$. Since the (α, n) reactions only produce a single neutron we have:

$$\begin{aligned}\mathbb{P}(X=i) &= \mathbb{P}(X=i|A)\,\mathbb{P}(A) + \mathbb{P}(X=i|B)\,\mathbb{P}(B) \\ &= \frac{S_\alpha}{S_f+S_\alpha}\delta_{i,1} + \frac{S_f}{S_f+S_\alpha} f_{s,i}. \end{aligned} \tag{1.98}$$

This gives us the probabilities in equation (1.97) □

Remark. Nuclear fissions and (α, n) reactions are not the only reactions producing neutrons in the material. We could consider a more general source term accounting for the (α, xn) reactions (x being an arbitrary number of neutrons). However, these reactions are negligible compared to (α, n) reactions. Similarly, we neglect the $(n, 2n)$ or (n, xn) reactions in our study and assume that the neutrons multiply only through induced fissions. To model this, one would simply need to update the probability distribution of the source process accordingly.

With this new source distribution, one can update the expression of the Feynman moments to account for the source description parameter x_s.

Proposition 1.4.13. *With the new compound Poisson source process, the count rate and asymptotic Feynman moments are updated. The new expressions are:*

$$R = -\frac{\varepsilon_F S}{\rho\overline{\nu}} \frac{\overline{\nu_s}}{\overline{\nu_s} + x_s - x_s\overline{\nu_s}} \tag{1.99}$$

$$Y_\infty = \frac{\varepsilon_F D_2}{\rho^2}\left(1 - \rho x_s \frac{\overline{\nu_s} D_{2,s}}{\overline{\nu} D_2}\right) \tag{1.100}$$

$$X_\infty = 3\left(\frac{\varepsilon_F D_2}{\rho^2}\right)\left(1 - \rho x_s \frac{\overline{\nu_s} D_{2,s}}{\overline{\nu} D_2}\right) - \frac{\varepsilon_F D_3}{\rho^3}\left(1 - \rho x_s \frac{\overline{\nu_s}^2 D_{3,s}}{\overline{\nu}^2 D_3}\right). \tag{1.101}$$

Proof. The derivation is obtained by replacing the probabilities $f_{s,i}$ by $f_{s,i}^{(C)}$ in equations (1.81), (1.89) and (1.90). The derivation is then straightforward and is not further detailed. □

These equations are used numerous times throughout this work, they constitute the low-fidelity model of the neutron correlation model.

In this study, the focus is on the Feynman moments though it is possible to consider a different parametrization of the same problem. Many studies focus on the multiplicity rates instead of the Feynman moments. The corresponding equations were originally introduced in [Böh85].

Definition 1.4.14. *In the Böhnel framework, the four parameters considered are the following.*

- *The leakage multiplication*

$$M_L = \frac{1 - k_p/\overline{\nu}}{1 - k_p}. \tag{1.102}$$

- *The leakage efficiency ϵ_L defined as the number of counts per leaking neutrons (or non-fissioning neutrons)*

$$\varepsilon_L = \frac{k_p/\overline{\nu}}{1 - k_p/\overline{\nu}}\varepsilon_F. \tag{1.103}$$

- *The fission source S_f in fissions.s^{-1}.*
- *The alpha ratio α_R defined as the number of (α, n) over spontaneous fission source neutrons*

$$\alpha_R = \frac{S_\alpha}{\overline{\nu_s} S_f} = \frac{1}{x_s} - 1. \tag{1.104}$$

Proposition 1.4.14 (Böhnel equations)**.** *The singles, doubles, and triples count rates are denoted by R, D, and T and are given respectively by:*

$$R = S_f(\varepsilon_L M_L)(\overline{\nu_s}(1+\alpha_R)) \tag{1.105}$$

$$D = S_f \frac{(\varepsilon_L M_L)^2}{2}\left(\nu_{2,s} + \overline{\nu_s}(1+\alpha_R)\nu_2\left(\frac{M_L - 1}{\overline{\nu} - 1}\right)\right) \tag{1.106}$$

$$\begin{aligned} T = S_f \frac{(\varepsilon_L M_L)^3}{6}\Bigg(&\nu_{3,s} + (\overline{\nu_s}(1+\alpha_R)\nu_3 + 3\nu_{2,s}\nu_2)\left(\frac{M_L - 1}{\overline{\nu} - 1}\right) \\ &+ 3\overline{\nu_s}(1+\alpha_R)\nu_2^2\left(\frac{M_L - 1}{\overline{\nu} - 1}\right)^2\Bigg). \end{aligned} \tag{1.107}$$

1.4.4 Estimating the Feynman moments

1.4.4.1 Feynman and ordinary moments

To find estimators for the Feynman moments we now define the probability $P_{n,i}(t)$ which is the probability of having n counts at T assuming i neutrons were present at time $T - t$. We also assume a compound Poisson source with intensity S and jump distribution defined by (1.97) is present in the system between $T - t$ and T. We also define $P_n(t) = P_{n,1}(t)$. We recall that the detector is only active in $[0, T]$, though detections in $[0, T]$ can be induced by source events occurring for $t < 0$.

The proofs of the results are not all detailed in this section, since they are mostly similar to the previous section with the probability $p_n(t)$.

Proposition 1.4.15. *For $t \geq 0$ and $n \in \mathbb{N}$ the probability $P_n(t)$ is governed by the differential equation:*

$$\frac{dP_n(t)}{dt} + SP_n(t) = S\left(\sum_{i=0}^{+\infty} f_{s,i}\left(\sum_{j=0}^{n} p_{j,i}(t)P_{n-j}(t)\right)\right). \tag{1.108}$$

Similarly, we introduce the probability-generating function.

Definition 1.4.15. *Let us introduce the probability-generating function g_s associated with the probabilities $(P_n(t))_{n\in\mathbb{N}}$ which is the function*

$$\begin{aligned} g_s(z,t) \colon \mathbb{C} \times \mathbb{R}^+ &\longrightarrow \mathbb{R} \\ (z,t) &\longmapsto g_s(z,t) \end{aligned}$$

where $g_s(z,t) = \sum_{n=0}^{+\infty} P_n(t)z^n$ for $|z| \leq 1$ and $t \geq 0$.

Proposition 1.4.16. *For $t \geq 0$, the PGF of the probabilities $(P_n(t))_{n\in\mathbb{N}}$ is given by:*

$$g_s(z,t) = \exp\left(S\int_0^t \left(\sum_{i=0}^{+\infty} f_{s,i} g(z,u)^i - 1\right) du\right) \tag{1.109}$$

Proof. The proof is similar to what has been done previously. We derive a partial differential equation for g_s by summing (1.108) over n. The PDE can then be solved with the initial condition $g_s(z, t = 0) = 1$. □

Definition 1.4.16. *For $p \geq 1$, the asymptotic ordinary moment of order p, denoted by $M_p(T)$ is defined by:*

$$M_p(T) = \lim_{t\to+\infty} \sum_{n=0}^{+\infty} n^p P_n(t). \tag{1.110}$$

Definition 1.4.17. *For $p \geq 1$, the asymptotic central moment of order p, denoted by $\mu_p(T)$ is defined by:*

$$\mu_p(T) = \lim_{t\to+\infty} \sum_{n=0}^{+\infty} (n - M_1(T))^p P_n(t). \tag{1.111}$$

In particular, $\mu_2(T) = M_2(T) - M_1(T)^2$ is the variance.

Definition 1.4.18. *The binomial cumulant of order n denoted by $\kappa_n(T)$ associated with the probabilities $(P_n(t))_{n\in\mathbb{N}}$ is defined by:*

$$\kappa_n(T) = \frac{1}{n!} \lim_{t\to\infty} \left(\frac{\partial^n \log g_s}{\partial z^n}\right)_{z=1}. \tag{1.112}$$

Proposition 1.4.17. *The binomial cumulant of order n is precisely the average number of n-detections defined in (1.76):*

$$\kappa_n(T) = \Gamma_n(T) = \int_0^{+\infty} S m_{s,n}(t) dt \tag{1.113}$$

Proof. From the expression of g_s given in (1.109), one can differentiate and invert integral and derivative as well as derivative and sum (by the dominated convergence theorem):

$$\begin{aligned}\Gamma_n(T) &= S\int_0^{+\infty} \left(\sum_{i=0}^{+\infty} f_{s,i} \left(\frac{\partial^n \log g(z,t)^i}{\partial z^n}\right)_{z=1}\right) dt \\ &= S\int_0^{+\infty} \left(\sum_{i=0}^{+\infty} f_{s,i} m_{n,i}(t)\right) dt = \int_0^{+\infty} S m_{s,n}(t) dt\end{aligned} \tag{1.114}$$

□

Proposition 1.4.18. *The second and third Feynman moments are linked to the ordinary moments by:*

$$Y(T) = \frac{M_2(T)}{M_1(T)} - M_1(T) - 1 \tag{1.115}$$

$$X(T) = \frac{M_3(T)}{M_1(T)} + 2(M_1(T)^2 + 1) - 3\left(\frac{M_2(T)}{M_1(T)} + M_2(T) - M_1(T)\right). \tag{1.116}$$

Proof. The derivation of these relations is obtained directly from equation (1.76). □

One can also derive a similar relation with the central moments $\mu_p(T)$.

Proposition 1.4.19. *The second and third Feynman moments are linked to the central moments by:*

$$Y(T) = \left(\frac{\mu_2(T)}{M_1(T)} - 1\right) \tag{1.117}$$

$$X(T) = \left(\frac{\mu_3(T)}{M_1(T)} - 1\right) - 3\left(\frac{\mu_2(T)}{M_1(T)} - 1\right). \tag{1.118}$$

Proof. The proof is similar to the previous one. □

Equation (1.117) shows that the second Feynman moment can be interpreted as an excess of variance compared to a Poisson process. Indeed, for a Poisson distribution, the variance and mean are equal. However, with a multiplying medium, there is an excess of variance and the ratio $\mu_2(T)/M_1(T)$ is thus greater than 1.

1.4.4.2 Sequential binning estimation

The equations (1.115) and (1.116) allow us to derive estimators for the Feynman moments. Consider an experiment (real or numerical) in which neutron detection instants are recorded. The experiment, which has a total duration of t_{tot} is split into $N_w = \lfloor \frac{t_{\text{tot}}}{T} \rfloor$ time windows of size T. For $1 \leq w \leq N_w$, the number of detected windows if the w-th window is denoted by n_w.

Definition 1.4.19. *For $p \geq 1$, the estimator for the p-th ordinary moment is denoted by $\widehat{M_p}$ and defined by:*

$$\widehat{M_p} = \frac{1}{N_w} \sum_{w=1}^{N_w} n_w^p \tag{1.119}$$

Definition 1.4.20. *The sequential binning estimators for the second and third Feynman moments are defined by:*

$$\widehat{Y} = \frac{\widehat{M_2}}{\widehat{M_1}} - \widehat{M_1} - 1 \tag{1.120}$$

$$\widehat{X} = \frac{\widehat{M_3}}{\widehat{M_1}} + 2(\widehat{M_1}^2 + 1) - 3\left(\frac{\widehat{M_2}}{\widehat{M_1}} + \widehat{M_2} - \widehat{M_1}\right) \tag{1.121}$$

These estimators have a significant variance which is highly dependent on the number of time windows available. However since our goal is to work with asymptotic Feynman moments, we need to make sure that T is large enough for the Feynman moments to reach their asymptotic values. There is a compromise to be made between the variance of the estimators and the asymptotic time width selection. This compromise is highlighted in Figure 1.1, where the second and third Feynman moments are estimated with (1.120) and (1.121) for various values of T, with detection instants obtained with a neutronic Monte Carlo simulation.

1.4.4.3 Filtered triggered binning estimation

Consider now specifically the case of a numerical simulation. Neutrons are created initially from a source event, and then create fission chains through branching fission reactions. The fission chains are independent from one another.

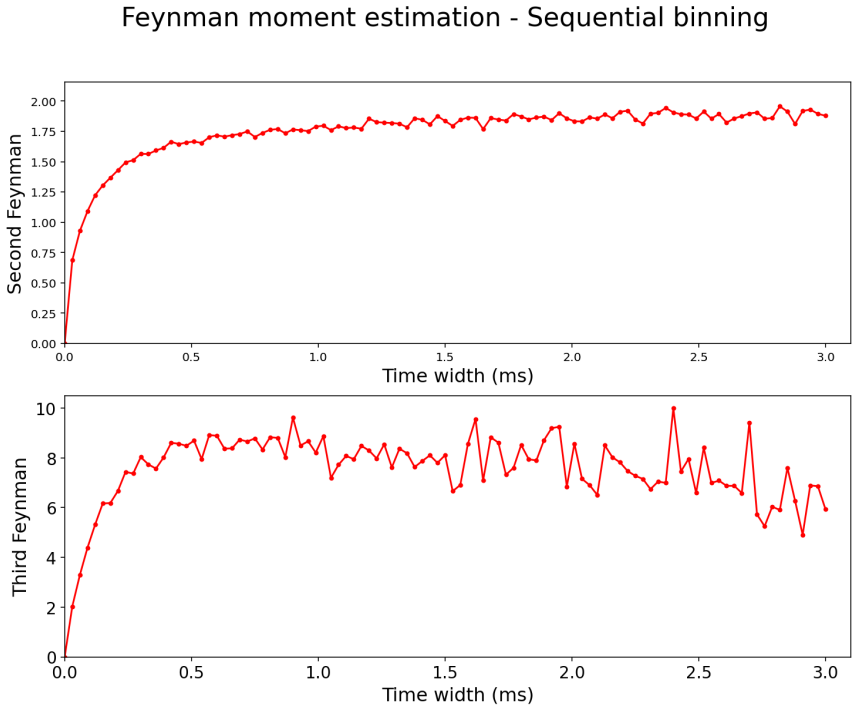

FIGURE 1.1: Evolution of the sequential binning estimators as a function of T.

A simultaneous detection of multiple neutrons is said to be truly correlated if the detected neutrons all belong to the same fission chain, that is they are induced by the same source event. By definition, the Feynman moments are linked to the number of truly correlated detections $\Gamma_n(T)$, or in other words they measure the average number of correlated double and triple detections (divided by the average number of counts $\Gamma_1(T)$).

With numerical simulations, it is possible to know with certainty whether or not two neutrons are truly correlated or not, based on the history number of the neutron in the Monte Carlo simulation.

Consider the following procedure. We have a total of N_d detected neutrons. When a neutron is detected, at time t_k for $1 \leq k \leq N_d$, we open a detection window of size T and we keep in memory the history number as the triggering neutron. In this window, we record only the neutrons which have the same history number. These neutrons thus belong to the same fission chain as the triggering neutron. We denote by $n_{k,\text{trig}}$ the number of such detections.

The procedure is described schematically in Figure 1.2. Consequently, $n_{k,\text{trig}}$ is the number of double correlated detections, where the triggering neutron is part of the doublet. Similarly, $n_{k,\text{trig}}(n_{k,\text{trig}} - 1)$ is the number of triple correlated detections for which the triggering neutron is part of the triplet. Thus the total numbers of double and triple-correlated detections are given by:

$$N_{2C} = \sum_{k=1}^{N_d} n_{k,\text{trig}} \tag{1.122}$$

$$N_{3C} = \frac{1}{2} \sum_{k=1}^{N_d} n_{k,\text{trig}}(n_{k,\text{trig}} - 1). \tag{1.123}$$

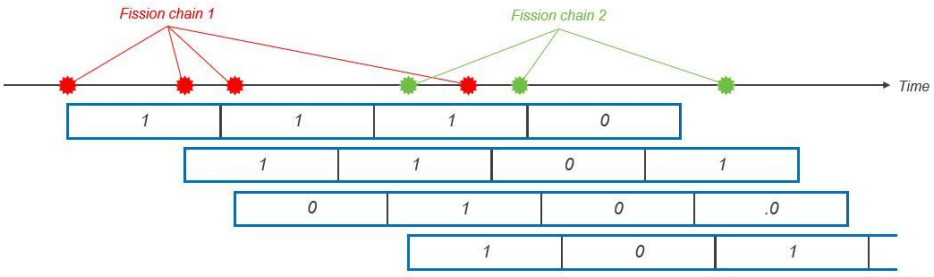

FIGURE 1.2: Illustration of the triggering of detection windows.

From this and recalling the definition of the Feynman moments (1.77), one can define new estimators for the Feynman moments which filter out the noise introduced by accidental correlations.

Definition 1.4.21. *The filtered triggered binning Feynman estimators* $\widehat{Y_{\text{trig}}}$ *and* $\widehat{X_{\text{trig}}}$ *for the second and third Feynman moments are defined by:*

$$\widehat{Y_{\text{trig}}} = \frac{2N_{2C}}{N_d} \tag{1.124}$$

$$\widehat{X_{\text{trig}}} = \frac{6N_{3C}}{N_d}. \tag{1.125}$$

where N_{2C} *and* N_{3C} *are defined in* (1.122) *and* (1.123).

These estimators allow us to filter out most of the noise caused by the random arrivals of independent neutrons in the same time window. The noise reduction is highlighted in Figure 1.3, where the Feynman moments from the same numerical experiment as in the previous section are estimated with the triggered binning estimators (1.124) and (1.125).

1.4.5 Rossi-α method

In this section we present briefly the Rossi-α method which was first introduced in [RG41]. This method is commonly used to estimate kinetic parameters of the multiplying medium such as the prompt decay constant α.

The Rossi-α method is based on the distribution of time intervals between each neutron detection. In a non-multiplying stationary system, the probability $p(t)$ of detecting a neutron in the infinitesimal time interval $[t, t+dt]$, knowing that a first neutron was detected at $t = 0$ should be given by the average count rate R. Assuming the neutron source follows a Poisson process, the count distribution in any given interval follows a discrete Poisson distribution with parameter $\lambda = Adt$ with A a constant to be determined. The probability of having a count in $[t, t+dt]$ is then $1 - p(X = 0)$ where X is a random variable following the Poisson distribution $X \sim \mathcal{P}(Adt)$. Then the probability of a count in $[t, t+dt]$ is :

$$p(t)dt = 1 - e^{-Adt} = Adt \tag{1.126}$$

In a stationary medium, the average of this law over a given time interval $[0, T]$ is AT which allows us to identify A to the average count rate $A = R$ such that $p(t)dt = Rdt$.

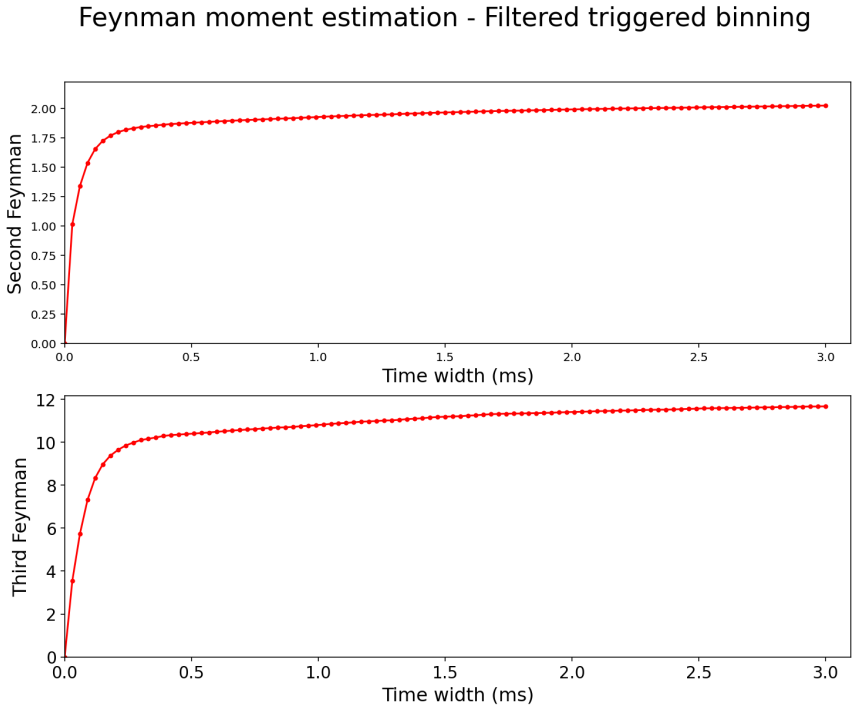

FIGURE 1.3: Evolution of the triggered binning estimators as a function of T.

In a multiplying medium, new correlations are introduced to the branching processes. Rossi showed that the probability of count could be written as :

$$p(t)dt = (R + Be^{\alpha t})dt \tag{1.127}$$

where $\alpha < 0$ is the prompt decay constant. The Rossi-α method, also known as the autocorrelation method, is based on the study of the autocorrelations of the detection statistics.

Let us consider an experiment or a numerical simulation in which all the successive neutron detection times $(t_k)_{1\leq k\leq n}$ are recorded. Let $\tau_k = t_{k+1} - t_k$ for $1 \leq k \leq n-1$. Then the distribution of the τ_k should be given by equation (1.127). Hence it is possible to identify the prompt decay constant by fitting the empirical distribution of the τ_k with equation (1.127). In practice, one often uses a least squares fit with a histogram distribution of the time intervals τ_k.

More recent works [Hua+20b; Bru+96] have suggested to use a modified equation with two exponential terms, based on two-region kinetics instead of standard point kinetics.

$$p(t)dt = (R + B_1e^{\alpha_1 t} + B_2e^{\alpha_2 t})dt \tag{1.128}$$

This approach is especially better suited for problems with reflector regions [Hua+20a].

1.4.6 Gamma noise theory

In all the previous discussions, we considered only neutron fluctuations. In this thesis, we discuss the possibility of using gamma correlations to further reduce the uncertainties in the fissile material characteristics.

For that purpose, let us introduce some results of gamma noise theory. The main results are derived in this section, but the proofs are not detailed. Most of the results presented here are provided in full detail in [PP07].

We focus on the multiplicity count rates which are the most used quantities in gamma multiplicity theory.

Definition 1.4.22 (Gamma multiplicity)**.** *We assume that the number of gammas produced by induced fissions is given by an integer-valued random variable μ and we define $f_i^{(\gamma)} = \mathbb{P}[\mu = i]$ for $i \in \mathbb{N}$. These quantities are known. We also define the factorial moments of the distribution as $\mu_p = \sum_{i=0}^{+\infty} f_i^{(\gamma)} \binom{i}{p}$ for $p \geq 1$ and we denote the mean by $\overline{\mu} = \mu_1$. The same quantities are introduced for spontaneous fissions multiplicities. The probabilities are denoted by $f_{s,i}^{(\gamma)}$ and the factorial moments by $\mu_{s,p}$.*

The singles, doubles and triples count rates for gamma detections are denoted respectively by R_γ, D_γ, and T_γ. We also denote the gamma multiplication M_γ and the gamma efficiency ε_γ.

Definition 1.4.23. *For a given gamma detector, the gamma efficiency ε_γ is defined as the average number of detected gammas over the total number of gammas absorbed by the detector. The gamma multiplication M_γ is defined as:*

$$M_\gamma = \frac{\overline{\mu} k_p / \overline{\nu}}{1 - k_p}. \tag{1.129}$$

Definition 1.4.24. *We introduce the random variables μ and μ_s which are the gamma analogs of ν and ν_s. They represent respectively the number of gammas produced per induced and spontaneous fissions. We also define μ_2 (resp. $\mu_{2,s}$) and μ_3 (resp. $\mu_{3,s}$) the second and third factorial moment of the distribution of gammas born from induced (resp. spontaneous) fissions, and $\overline{\mu}$ and $\overline{\mu_s}$ which are the average number of gammas produced by induced and spontaneous fissions.*

Proposition 1.4.20. *The multiplicity rates for the single, double, and triple detections, denoted respectively by R_γ, D_γ, and T_γ, are given by:*

$$R_\gamma = S_f \varepsilon_\gamma \left(\overline{\mu}_s + \alpha_R \overline{\nu}_s + M_\gamma \overline{\nu_s} (1 + \alpha_R) \right) \tag{1.130}$$

$$D_\gamma = \frac{\varepsilon_\gamma^2 S_f}{2} \left[\mu_{2,s} + \frac{2\overline{\nu}_s (\overline{\mu_s} + \alpha_R \overline{\nu}_s)(1 + \alpha_R)}{1 + \alpha_R \overline{\nu}_s} M_\gamma + \nu_{2,s} M_\gamma^2 + \overline{\nu}_s (1 + \alpha_R) g_2 \right] \tag{1.131}$$

$$\begin{aligned} T_\gamma = \frac{\varepsilon_\gamma^3 S_f}{6} \Bigg[& \mu_{3,s} + 3\nu_{2,s} g_2 + \overline{\nu}_s (1 + \alpha_R) g_3 + \frac{(\overline{\mu}_s + \alpha_R \overline{\nu}_s) \nu_{2,s} M_\gamma^2}{1 + \alpha_R \overline{\nu}_s} \\ & + \frac{3(\mu_{2,s} M_\gamma + \overline{\mu}_s g_2 + \alpha_R \overline{\nu}_s g_2) \overline{\nu}_s (1 + \alpha_R)}{1 + \alpha_R \overline{\nu}_s} + \nu_{3,s} M_\gamma^3 \Bigg]. \end{aligned} \tag{1.132}$$

where g_2 and g_3 represent the second and third factorial moments of the photon distribution from a single neutron.

$$g_2 = \frac{M_L - 1}{\overline{\nu} - 1} (\mu_2 + 2\overline{\mu}\, \overline{\nu} M_\gamma + \nu_2 M_\gamma^2) \tag{1.133}$$

$$g_3 = \frac{M_L - 1}{\overline{\nu} - 1} \left(\mu_3 + 3\mu_2 \overline{\nu} M_\gamma + 3\overline{\mu}(\nu_2 M_\gamma^2 + \overline{\nu} g_2) + \nu_3 M_\gamma^3 + 3\nu_2 M_\gamma g_2 \right). \tag{1.134}$$

1.4.7 Multi-group framework

All the previous work was derived in the point model framework which serves as a simplified model for an analytical derivation of the Feynman moments.

Yet, it is possible to work with multi-group approximations and still obtain an expression for the Feynman moments. In [PP21], a space-dependent model is derived from the general transport equation. It highlights a systematic underestimation of the factorial moments in the point model framework. This bias is quantified for a spherical object and different values of x_s.

A two-region formulation of the Feynman-α method is introduced in [And+12]. This formalism is extended to the multi-group and multi-region case in [MD15] and [SPE18]. In [EYY06], a space and energy-dependent formulation of the third moments is proposed, based on the α-expansion introduced in [OI65].

All these recent works highlight the possible derivation of more complete models for the Feynman moments. However, in this work, the point model itself is only used as a low-fidelity approximation to improve surrogate models. Therefore, the focus is on the simplest form of the Feynman moments expressions.

1.4.8 Simulations and computer codes

In neutronic analysis, computer codes can be divided into two main categories that are stochastic and deterministic codes. Stochastic codes are designed to estimate quantities of interest (neutron flux, power, population, ...) by creating many independent simulations of neutrons, where the neutrons behave like random variables. Each nuclear reactions occur with a given probability and averaging over a large number of runs gives Monte Carlo estimators of the quantities of interest. These methods are considered state-of-the-art, though they are computationally demanding. On the other hand, deterministic codes try to solve the neutron transport equation (1.36) using various approximations. As a result, these codes are often much faster than MC simulations, though they may not be as accurate.

1.4.8.1 Analog Monte Carlo simulations

Various Monte Carlo codes have been developed with the increasing available computing power. They differ in their functionalities though the general principle is the same. Most of this work was accomplished with MCNP6 [Goo+12]. MCNP6 is a general-purpose Monte Carlo neutronic code developed at Los Alamos National Laboratory designed to solve the 3D transport problem for a multitude of particle types.

Neutron noise analysis has also been investigated and developed with the code TRIPOLI-4®[Bru+14] which is a multi-purpose Monte Carlo code developed at CEA for both fission and fusion systems.

To improve the Monte Carlo estimators of the various quantities, all these codes use several variance reduction techniques. As an example, simulated neutrons are not simply deleted at each reaction but are given a weight coefficient which is taken into account in the MC estimators. When a neutron induces fission, it produces more neutrons. Instead of simulating those newly born neutrons, the weight of the inducing neutron is increased. Many more variance reduction techniques are

implemented in Monte Carlo simulations and for a more detailed documentation, the author refers to [HW03; KRR10; GF11].

However, when dealing with neutron noise analysis, our goal is to simulate the true evolution of the neutron population since we are interested in the temporal correlations of the detected neutrons. Thus, one cannot use such variance reduction techniques. The Monte Carlo codes used are thus used for "analog" simulations, in which all the variance reduction techniques are set aside.

To solve neutron correlation problems, one needs to mention MCNP-PoliMi [PPM03] which is an extension of MCNP developed specifically for neutron correlation studies by Politecnico di Milano and the University of Michigan.

1.4.8.2 Deterministic codes

Deterministic neutronic codes have also been used for neutron noise analysis. As an example, in [Hum19], the Kolmogorov backward equation and the binomial moments equation for a single initial neutron are solved iteratively based on a discrete ordinates code, which provides a coupled system of equations. Similarly in [Mat12], the Feynman moment is evaluated with a deterministic approach based on and compared with experimental results. The main drawback of these approaches is that they do not provide uncertainties in the predictive moments though some developments in that sense have arisen [OMA17]. In this thesis, numerical simulations are consistently performed with Monte Carlo simulations which remain state-of-the-art in neutronics despite their generally higher computation times.

1.5 Inverse problems and uncertainty quantification

In this section, a general introduction to inverse problems is presented. For this work, the formulation of inverse problems is as follows. Let $\mathcal{X} \subset \mathbb{R}^p$ and $\mathcal{Y} \subset \mathbb{R}^D$ be respectively the input and output space. Consider a function f known as the direct model defined by:

$$\begin{aligned} f\colon \mathcal{X} &\longrightarrow \mathcal{Y} \\ x &\longmapsto f(x). \end{aligned}$$

The direct model f can be an analytical function though in most cases of interest, it is given by a complex computer model or practical experiments.

Suppose we are provided with some observations $(y^{(k)})_{1\leq k\leq N}$ for $N \geq 1$. These observations may arise from practical experiments or a complex computer code. Most likely, they are direct realizations of f but rather noisy observations of $f(x_{\text{true}})$ where $x_{\text{true}} \in \mathcal{X}$ are the true unknown inputs. We seek a good estimate x_* of x_{true}.

1.5.1 Ill-posed inverse problems

Because of the noise introduced in the observations $\mathbf{y}$, we may not find a good estimate x_*. It is also possible to obtain multiple solutions to our inverse problem. Let us define formally the notion of ill-posed inverse problems, as first introduced by Hadamard.

Definition 1.5.1. *An inverse problem is said to be ill-posed if it verifies at least one of the three following conditions:*

- *The inverse problem has no solution.*
- *The solution exists but is not unique.*
- *At least a solution exists but it does not depend continuously on the observations.*

Ill-posed inverse problems arise frequently in complex models with noisy observations. An example of an ill-posed inverse problem is the inverse heat transfer problem [Web81; SJ60] which is based on the heat equation:

$$\frac{\partial T}{\partial t} - D_T \frac{\partial^2 T}{\partial x^2} = 0 \tag{1.135}$$

where $T(x,t)$ is the temperature at time $t \geq 0$ and position $0 \leq x \leq L$, and D_T is the thermal diffusion coefficient. We consider temperature measurements inside the material and we wish to get back to the boundary condition at $x = 0$ which are the heat flux density and temperature $T(x = 0, t)$. This problem is ill-posed and yet the forward equation is analytically tractable and provides continuous solutions (for sufficiently regular initial and boundary conditions).

To justify this, one can write the solution of the inverse heat problem based on Fourier analysis as:

$$T_0(x) = \int_{-\infty}^{+\infty} \widehat{T}(k,t) e^{ikx + D_T k^2 t} dk \tag{1.136}$$

where $\widehat{T}(k,t)$ is the Fourier transform of $T(x,t)$. For large t, this integral is only tractable if the function $T(x,t)$ is smooth, that is if $\widehat{T}(k,t) = \mathcal{O}(\exp(-k^{-\beta}))$ with $\beta > 2$. Thus the solution may not exist.

A standard approach to solve inverse problems is to solve a least-square minimization problem.

$$x_* \in \operatorname{argmin} \sum_{k=1}^{N} \|y^{(k)} - f(x)\|^2 \tag{1.137}$$

where $\|\cdot\|$ is the standard Euclidean norm on $\mathbb{R}^D$. In this thesis, one of our goals is to identify a fissile material based on observations of the neutron count rate R and the second and third-order asymptotic Feynman moments Y_∞ and X_∞ (for neutrons). In a first approach, we then consider $y = (R, Y_\infty, X_\infty)$. Based on the point model equations (1.99), (1.100) and (1.101), we introduce with the same notations $x = (k_p, \varepsilon_F, S, x_s)$.

Let us prove that our inverse problem is ill-posed. Consider a fictitious material characterized by $x_{\text{true}} = (0.90, 0.01, 10^5, 0.5)$. We generate $N = 5$ fictitious observations by introducing random zero-mean Gaussian noise on the prediction $f(x_{\text{true}})$ such that for $1 \leq k \leq N$:

$$y^{(k)} = f(x_{\text{true}}) + \varepsilon^{(k)} \tag{1.138}$$

where $\varepsilon^{(k)} \sim \mathcal{N}(0, C)$ where the covariance is chosen as:

$$C = \operatorname{diag}\left(f(x_{\text{true}}) \times (0.01^2, 0.05^2, 0.20^2)\right). \tag{1.139}$$

We thus introduced a relative noise of respectively 1%, 5% and 20% for R, Y_∞ and X_∞. Finally, we solve the inverse problem by least-square minimization to obtain x_*. The process is iterated 5 times for various sets of noisy observations $\mathbf{y}$. The least-square

TABLE 1.1: Successive estimates of x_*.

	k_p	ε_F	S	x_s
Run 1	0.771	0.070	1.56×10^5	0.55
Run 2	0.778	0.054	1.95×10^5	0.31
Run 3	0.898	0.022	1.93×10^5	0.81
Run 4	0.821	0.060	1.35×10^5	0.55
Run 5	0.880	0.027	1.82×10^5	0.25

estimates obtained are shown in Table 1.1. The estimates differ largely from one run to the other. Thus the estimation of x_* by least-squares minimization is not robust. The inverse problem is ill-posed. Besides, for this inverse problem, we chose an analytical direct model though in the work developed in this thesis the direct models are never analytical and lie in higher dimensions (either for the input or output space). In the next sections, we further discuss the basic theory of inverse problems. For complete references on the subject, one can for instance read [KS06; Tar05].

1.5.2 Linear inverse problems

Let us focus for now on linear inverse problems. Expanding on the notations from the previous paragraph, we define $A \in R^{D\times p}$ a matrix of rank p and we assume the direct model f is given by:

$$\begin{aligned} f\colon \mathcal{X} &\longrightarrow \mathcal{Y} \\ x &\longmapsto Ax. \end{aligned}$$

Given some observations $\mathbf{y} = (y^{(k)})_{1\leq k\leq N}$, the inverse problem can then be formulated as:

$$x_* \in \operatorname{argmin}_{x\in\mathcal{X}} \sum_{k=1}^{N} \|y^{(k)} - Ax\|^2. \tag{1.140}$$

This problem can be reformulated by considering the block matrix $\mathbf{A} = (A, ..., A) \in \mathbb{R}^{DN\times p}$ and $\tilde{\mathbf{y}} = (y_1^{(1)}, ..., y_D^{(1)}, y_1^{(2)}, ..., y_D^{(N)})^T \in \mathbb{R}^{DN}$ and thus:

$$x_* \in \operatorname{argmin}_{x\in\mathcal{X}} \|\tilde{\mathbf{y}} - \mathbf{A}x\|^2 \tag{1.141}$$

where the norm is now the Euclidean norm on $\mathbb{R}^{DN}$.

Proposition 1.5.1. *The inverse problem stated in equation (1.141) has the unique solution:*

$$x_* = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\tilde{\mathbf{y}}. \tag{1.142}$$

Proof. Consider the function $g\colon \mathcal{X} \mapsto \mathbb{R}$ defined for $x \in \mathcal{X}$ by $g(x) = \|\tilde{\mathbf{y}} - \mathbf{A}x\|^2$. This function is differentiable and its gradient is given by:

$$\frac{dg}{dx}(x) = 2\left(\mathbf{A}^T\mathbf{A}x - \mathbf{A}^T\tilde{\mathbf{y}}\right). \tag{1.143}$$

Since g is strictly convex its unique minimum is reached in x_* which verifies $\frac{dg}{dx}(x_*) = 0$. Thus, the least-squares estimate is given by:

$$\mathbf{A}^T\mathbf{A}x_* = \mathbf{A}^T\tilde{\mathbf{y}}. \tag{1.144}$$

Since $\text{rank}(A) = p$ we know that $\text{rank}(\mathbf{A}) = p$. This means that $\text{rank}(\mathbf{A}^T\mathbf{A}) = p$ and since $\mathbf{A}^T\mathbf{A}$ is a square matrix of size p it is invertible. Thus the previous equation can be inverted to retrieve $x_* = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\tilde{\mathbf{y}}$. □

Linear inverse problems can thus be solved analytically provided the matrix A has rank p. To study the sensitivity of the solution to the observations, let us look at the singular value decomposition (SVD) of the matrix $\mathbf{A}$, which gives $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ where $\mathbf{U}$ and $\mathbf{V}$ are real orthogonal matrices of sizes $DN \times DN$ and $p \times p$ and $\mathbf{\Sigma}$ is a $DN \times p$ matrix with non-negative diagonal coefficients, denoted as $(\sigma_i)_{1 \leq i \leq p}$.

Introducing the SVD in the expression of x_* gives us:

$$x_* = (\mathbf{V}\mathbf{\Sigma}^T\mathbf{\Sigma}\mathbf{V}^T)^{-1}\mathbf{V}\mathbf{\Sigma}^T\mathbf{U}^T\tilde{y} = \mathbf{V}(\mathbf{\Sigma}^T\mathbf{\Sigma})^{-1}\mathbf{\Sigma}^T\mathbf{U}^T\tilde{y}. \tag{1.145}$$

Now recall we assumed that the observations are obtained by an additive Gaussian noise such that $\tilde{\mathbf{y}} = \mathbf{A}x_{\text{true}} + \boldsymbol{\varepsilon}$ where $\boldsymbol{\varepsilon} \sim \mathcal{N}(0, \mathbf{C})$ and $\mathbf{C} = \mathcal{I}_N \otimes C$ with $\mathcal{I}_N$ the identity matrix of size N. Thus we then have:

$$x_* = x_{\text{true}} + \mathbf{V}(\mathbf{\Sigma}^T\mathbf{\Sigma})^{-1}\mathbf{\Sigma}^T\mathbf{U}^T\boldsymbol{\varepsilon}. \tag{1.146}$$

To simplify, we assume the covariance is given by $C = \sigma^2_{\text{obs}}\mathcal{I}_D$, which leads to $\mathbf{C} = \sigma_{\text{obs}}\mathcal{I}_{DN}$. Introducing $\mathbf{S} = (\mathbf{\Sigma}^T\mathbf{\Sigma})^{-1}\mathbf{\Sigma}^T$, the covariance of the error is then given by:

$$\mathbb{E}\left[(x_* - x_{\text{true}})(x_* - x_{\text{true}})^T\right] = \mathbf{V}\mathbf{S}\mathbf{S}^T\mathbf{V}^T. \tag{1.147}$$

Looking at the variance for the component $1 \leq i \leq p$:

$$\mathbb{V}\left[x_* - x_{\text{true}}\right]_i = \sum_{k=1}^{p} \frac{v^2_{i,k}}{\sigma^2_k}. \tag{1.148}$$

and summing over i, while recalling that $\mathbf{V}$ is orthogonal, we obtain:

$$\sum_{i=1}^{p} \mathbb{V}\left[x_* - x_{\text{true}}\right]_i = \mathbb{E}\left[\|x_* - x_{\text{true}}\|^2\right] = \sigma^2_{\text{obs}} \sum_{k=1}^{p} \frac{1}{\sigma^2_k} \tag{1.149}$$

Thus the mean squared error can explode if any of the singular value σ_k is close to 0. Even linear inverse problems, which have an exact analytical solution, can turn out to be difficult to solve if they are ill-posed.

1.5.3 Tikhonov regularization

Regularization is a widely used tool to limit the sensitivity of the solution in ill-posed inverse problems. The idea is to add a second penalizing term that removes the high volatility of the estimates in cases where one singular value of the matrix $\mathbf{A}$ is close to 0.

For linear inverse problems, regularization is introduced by the mean of a matrix $\mathbf{R} \in \mathbb{R}^{m \times p}$ with a given integer $m \geq 1$. The regularized inverse problem is now formulated for a $x_0 \in \mathcal{X}$ as:

$$x_* \in \text{argmin}_{x \in \mathcal{X}} \|\tilde{\mathbf{y}} - \mathbf{A}x\|^2 + \|R(x - x_0)\|^2_m \tag{1.150}$$

where $\|\cdot\|_m$ is a norm on $\mathbb{R}^m$. In practice, the most commonly used approach is known as Tikhonov regularization [Tik+43] and consists in choosing $R = \alpha\mathcal{I}_p$ for some real $\alpha \geq 0$, $x_0 = 0$ and $\|\cdot\|_m$ as the Euclidean norm which yields:

$$x_* \in \text{argmin}_{x\in\mathcal{X}} \|\tilde{\mathbf{y}} - \mathbf{A}x\|^2 + \|\alpha x\|^2. \tag{1.151}$$

The solution of this regularized inverse problem is given by:

$$x_* = (\mathbf{A}^T\mathbf{A} + \alpha^2\mathcal{I}_p)^{-1}\mathbf{A}^T\tilde{\mathbf{y}}. \tag{1.152}$$

The newly added regularization term removes the possibility of large inverse singular values in the inverted matrix. Many other approaches have been investigated to regularize inverse problems. More details on the subject are available in [GHO99; EHN96].

The choice of α is crucial since if α is too large, it does impact significantly the sought solution.

As an example, we apply the same procedure as in Section 1.5.1, to provide new estimates. As seen in Table 1.2, the estimates are now much more robust and consistent around the true value. However, there is still some residual variability in the point estimates because of the randomness of the noisy observations. This variability is expected to be larger in regimes with few observational data. This is why our goal is

TABLE 1.2: Successive estimate of x_* with Tikhonov regularization.

	k_p	ε_F	S	x_s
Run 1	0.840	0.057	1.23×10^5	0.46
Run 2	0.869	0.037	1.52×10^5	0.49
Run 3	0.855	0.045	1.38×10^5	0.51
Run 4	0.854	0.047	1.33×10^5	0.53
Run 5	0.863	0.041	1.44×10^5	0.50

to estimate the uncertainties in the resolution of the inverse problem, using a Bayesian approach whose goal is to provide a complete probability distribution for the inputs instead of point estimates.

1.5.4 Bayesian resolution

Our goal is not only to solve an inverse problem using neutron and gamma correlations for fissile matter identification. Ultimately, the objective is to provide a robust quantification of the underlying uncertainties. For that purpose, let us introduce Bayesian formalism in the inverse problems. This will serve as an introduction for non-expert readers. For a comprehensive review of Bayesian inverse problems, the readers may turn to [Stu10; DS17].

1.5.4.1 General methodology

For that motive, most of this work will focus on a Bayesian resolution of the inverse problems at stake. The Bayesian approach provides a full distribution of the possible x given the available observations $\mathbf{y}$. This can be used to provide a point estimate, which is generally chosen as the most likely input point x_m. However, having access to the full distribution also allows the construction of credible regions for the various inputs,

it gives access to the moments of the distribution, and the estimated uncertainties can also be passed on to other models afterward.

The fundamental idea behind the Bayesian approach is to consider the inputs x as random variables, upon which prior information is encoded with the so-called **prior distribution** with density $p(x)$. The prior density encodes the knowledge of the inputs before the observations. It can be influenced by previous experiments or results, simpler analytical models, or expert knowledge or it can be simply taken as mostly non-informative. More details on the prior's choice are discussed in a further section.

Now, given some observations $\mathbf{y}_N = (y^{(k)})_{1\leq k\leq N}$ for $N \geq 1$, the goal is to estimate the distribution of the inputs x conditioned by the knowledge of $\mathbf{y}$. This distribution whose density is denoted by $p(x|\mathbf{y}_N)$ is known as the **posterior distribution**.

Based on the renowned Bayes's theorem, the posterior is given up to a multiplicative constant by:

$$p(x|\mathbf{y}_N) \propto p(x)L(\mathbf{y}_N|x) \tag{1.153}$$

where we introduced the likelihood $L(\mathbf{y}_N|x)$ which is the probability of the observations given some known inputs x. The proportionality constant $p(\mathbf{y}_N)$ is known as the evidence:

$$p(\mathbf{y}_N) = \int_{\mathcal{X}} p(x)L(\mathbf{y}_N|x)dx. \tag{1.154}$$

In the particular case of iid Gaussian additive noise, where $y^{(k)} = f(x_{\text{true}}) + \varepsilon^{(k)}$ for $1 \leq k \leq N$ with some $\varepsilon^{(k)} \sim \mathcal{N}(0, C)$, then the likelihood is analytically tractable and is given by:

$$\begin{aligned} L(\mathbf{y}_N|x) &= |2\pi C|^{-1/2} \exp\left(-\frac{1}{2}\sum_{k=1}^{N}\left(y^{(k)} - f(x)\right)^T C^{-1}\left(y^{(k)} - f(x)\right)\right) \\ &= |2\pi C|^{-1/2} \exp\left(-\frac{1}{2}\sum_{k=1}^{N}\|y^{(k)} - f(x)\|_C^2\right) \end{aligned} \tag{1.155}$$

where we introduced the Mahanalobis norm which is defined for a vector $a \in \mathbb{R}^D$ and a positive definite matrix $C \in \mathbb{R}^{D\times D}$ as $\|a\|_C = \sqrt{a^T C^{-1} a}$.

Assuming the direct model f is available, the posterior distribution is known within a multiplicative constant. It is therefore possible to sample the distribution using Monte Carlo Markov chain methods (see Section 1.6).

1.5.4.2 Influence of the prior

In Bayesian statistics, the prior choice is a vast subject of study. Especially in low data regimes, the prior can drastically influence the posterior distribution obtained.

When no expert knowledge or previous information is available, the prior is often a non-informative distribution. However, even so, the prior choice is subjective to the user. Consider for example some input $x \in (0, a]$ for some $a \geq 0$ with an associated prior $p(x)$. For a non-informative prior, one could try to choose a uniform prior on $(0, a]$ such that $p(x) = a^{-1}$ for all $x \in (0, a]$. Now we reparametrize the problem such that we work with the parameters $\theta = x^{-1} \in [a^{-1}, +\infty)$. The prior density on θ is not uniform anymore and is given by $p(\theta) = \frac{1}{a\theta^2}$.

To circumvent the choice of a subjective prior, various studies have been conducted to determine objective priors for a given problem. An example of such a prior is known as Jeffreys' prior [Jef46]. In particular, Jeffreys' prior is invariant by reparametrization of the inverse problem. The density of Jeffreys' prior is given by:

$$p(x) \propto |I(x)|^{1/2} \tag{1.156}$$

where $I(x) \in \mathbb{R}^{p\times p}$ is the Fisher information matrix defined by:

$$I(x)_{i,j} = \mathbb{E}\left[\left(\frac{\partial}{\partial x_i}\log L(\mathbf{y}_N|x)\right)\left(\frac{\partial}{\partial x_j}\log L(\mathbf{y}_N|x)\right)\right] \tag{1.157}$$

for $1 \leq i, j \leq p$.

Other studies have been conducted to study objective or reference priors [BB92; BBS15]. For a more detailed overview of prior selection in Bayesian inference, the author recommends more complete works on the subject such as [Ber79; KW96; Gho11].

Overall in this work, the prior is considered to have little impact on the study as we are provided with enough observations. As such, it is generally chosen as uniform on a given compact region of the input space, unless explicitly mentioned otherwise.

1.5.4.3 Bernstein-von Mises regime

The Bayesian approach described in the previous sections is a powerful tool to quantify uncertainties in an inverse problem using the posterior distribution. However, what are the asymptotic guarantees that the posterior distribution is representative of the true parameter uncertainties?

This question is partially answered by the Bernstein-von Mises theorem. However, before introducing the theorem itself, we need to recall a few definitions and properties of estimators.

Consider an inverse problem given by a likelihood $L(\mathbf{y}_N|x)$ for some $x \in \mathcal{X}$ and some observations $\mathbf{y}_N = (y^{(k)})_{1\leq k\leq N}$. We also define the prior and posterior which have respective densities $p(x)$ and $p(\mathbf{y}_N|x)$. We denote by x_{true} the unknown true inputs. We assume the direct model f is at least twice differentiable.

Proposition 1.5.2 (Cramér-Rao bounds). *Consider $N \geq 1$ observations $\mathbf{y}_N = (y^{(k)})_{1\leq k\leq N}$ of a random variable and let $\hat{x}_N$ be an unbiased estimator for an unknown parameter $x_{\text{true}} \in \mathcal{X}$ given the observations. Then, under some regularity conditions detailed in [CRA99], the covariance of the estimator is bounded (in the sense of matrices):*

$$\text{Cov}\,(\hat{x}_N) \geq \frac{1}{N} I(x)^{-1} \tag{1.158}$$

where $I(x) \in \mathbb{R}^{p\times p}$ is the Fisher information matrix introduced in equation (1.157) and where the notation $A \geq B$ for $A, B \in \mathbb{R}^{p\times p}$ means that the matrix $A - B$ is positive semi-definite.

This bound is known as the Cramér-Rao bound [CRA99]. In particular, for $1 \leq k \leq p$ we have the inequality:

$$\mathbb{V}\left[(\hat{x}_N)_k\right] \geq \frac{1}{N}\mathbb{E}\left[\left(\frac{\partial}{\partial x_k}\log L(\mathbf{y}_N|x)\right)^2\right]^{-1}. \tag{1.159}$$

Definition 1.5.2. *The Maximum Likelihood estimator (MLE) $x_N^{(\mathrm{MLE})}$ is the estimator defined by:*

$$x_N^{(\mathrm{MLE})} = \mathrm{argmax}_{x\in\mathcal{X}} L(\mathbf{y}_N|x). \tag{1.160}$$

Proposition 1.5.3. *For Gaussian likelihoods with iid observations, if the model is identifiable, that is $L(\cdot|x_1) \neq L(\cdot|x_2)$ for $x_1 \neq x_2$, then under some regularity conditions found in [Leh99] the MLE estimator $x_N^{(\mathrm{MLE})}$ has the following properties:*

- *It is asymptotically unbiased $\mathbb{E}\left[x_N^{(\mathrm{MLE})}\right] \xrightarrow[N\to+\infty]{} x_{\mathrm{true}}$.*
- *The MLE is consistent, meaning that $x_N^{(\mathrm{MLE})} \xrightarrow[N\to+\infty]{p} x_{\mathrm{true}}$.*
- *The MLE estimator is said to be invariant. That is, for any function g defined on $\mathcal{X}$, the MLE of $g(x)$ is precisely $g(x_N^{(\mathrm{MLE})})$.*
- *The MLE has the asymptotic normality property. In other words:*

$$\sqrt{N}(x_N^{(\mathrm{MLE})} - x_{\mathrm{true}}) \sim \mathcal{N}\left(0, I(x_{\mathrm{true}})^{-1}\right). \tag{1.161}$$

  It achieves the Cramér-Rao bound asymptotically.

Proof. The proof is not detailed here as this is not the core of the work developed in this thesis. We refer to theorems 7.1.1 and 7.3.1 of [Leh99] for the complete proof. □

The MLE is one of the most-used estimators in the frequentist approach, thanks to its consistent and asymptotic properties. However, the Bayesian framework differs from the frequentist viewpoint. Here we are interested in the full distribution of the variables x. If we wish to provide a point estimate of the true point, this is generally done with the maximum-a-posteriori estimator.

Definition 1.5.3. *The maximum-a-posteriori (MAP), denoted by $x_N^{(\mathrm{MAP})}$ is an estimator defined as:*

$$x_N^{(\mathrm{MAP})} \in \mathrm{argmax}_{x\in\mathcal{X}} p(x|\mathbf{y}_N). \tag{1.162}$$

Now, the question that must be asked is whether or not the posterior distribution obtained with Bayesian inference has a statistical meaning. We hope the posterior distribution tends to concentrate around the true value x_{true} as the number of observational data increases. Under certain regularity conditions, this is indeed the case. This result is known as Doob's theorem.

Theorem 1.5.1 (Doob's convergence theorem)**.** *Let us denote by Π the prior measure with density $p(x)$ for $x \in \mathcal{X}$. Assuming that the model is identifiable, which means that $L(\cdot|x_1) \neq L(\cdot|x_2)$ for $x_1 \neq x_2$, then there exists $\mathcal{X}_* \subset \mathcal{X}$ with $\Pi(\mathcal{X}_*) = 1$ and such that for all $x_* \in \mathcal{X}_*$, if the observations $\mathbf{y}_N = (y^{(k)})_{1\leq k\leq N}$ are iid given by $L(\cdot|x_*)$ we have for all $\varepsilon > 0$:*

$$\lim_{N\to+\infty} \mathbb{P}\left(x \in V_\varepsilon(x_*)|(y^{(1)}, ..., y^{(N)})\right) = 1 \tag{1.163}$$

where $V_\varepsilon(x_*) = \{x \in \mathcal{X} s.t. \|x - x_*\| \leq \varepsilon\}$.

This convergence theorem states that as long as the model is identifiable and that as long as the true parameter is not taken from a null set under the prior measure, then

if the observations are iid the posterior collapses to the true value. The posterior is said to be consistent in that case. The detailed proof is not provided here, the author refers to [Vaa00; Mil18].

This result offers a strong guarantee for the consistency of Bayesian statistics even though it may be irrelevant in certain cases (see [DF86]).

The asymptotic behavior of Bayesian statistics is supported by another result known as the Bernstein-von Mises theorem, which states that under some conditions, the posterior distribution approaches a normal distribution as $N \to +\infty$.

Theorem 1.5.2 (Bernstein-von Mises)**.** *Suppose that the observations are iid and that the posterior density $p(x|\mathbf{y}_N)$ is twice-differentiable on $\mathcal{X}$ for all $N \in \mathbb{N}^*$. We introduce the MAP estimator $x_N^{(\mathrm{MAP})}$ obtained from observations $\mathbf{y}_N = (y^{(k)})_{1\leq k\leq N}$ drawn from the likelihood $L(\cdot|x_{\mathrm{true}})$. Under some regularity conditions detailed in [BS09], the posterior distribution converges to a Gaussian distribution:*

$$\sqrt{N}\left(x_{\mathrm{true}} - x_N^{(\mathrm{MAP})}\right)|\mathbf{y}_N \xrightarrow[N\to+\infty]{\mathcal{L}} \mathcal{N}\left(0, I(x_{\mathrm{true}})^{-1}\right). \tag{1.164}$$

The proof of this result can be found in many resources such as [BS09; GR11].

Remark. This theorem is also known as the Bayesian Central Limit Theorem (BCLT) in the literature.

This result can be seen as a justification of Bayesian inference methods. Indeed, with many observations, the credible regions obtained from the posterior distribution coincide with the confidence intervals obtained in a frequentist approach with the MLE. Bernstein-von Mises theorem reconciles the Bayesian and frequentist frameworks.

Yet, this result can fall off for more peculiar cases, for example, if the observations are not iid, or the model is misspecified. In that regard, recent developments on the asymptotic properties of Bayesian statistics have been brought to light [KV12; Mil21]. From now on, the number of observations is assumed to be set to a given N. The notation for the observations is simplified to $\mathbf{y}$.

1.6 Monte Carlo Markov chain sampling

In a Bayesian framework, one often needs to evaluate a posterior distribution. For the specific case of inverse problems, we would like to identify a set of parameters x given some observations $\mathbf{y} = (y^{(k)})_{1\leq k\leq N}$. The focus is on the posterior distribution $p(x|\mathbf{y})$ which quantifies the information brought up by the observations. From Bayes' theorem, this posterior distribution is given by:

$$p(x|\mathbf{y}) = \frac{p(x)L(\mathbf{y}|x)}{\int_{\mathcal{X}} p(x')L(\mathbf{y}|x')dx'}. \tag{1.165}$$

This expression is at the center of Bayesian statistics. The main obstacle in such a Bayesian framework is that the integral is often intractable and difficult to compute, especially for high-dimensional problems. Monte Carlo Markov Chain methods aim at sampling the posterior distribution using only the numerator on the above expression. The generic principle of MCMC methods is to build an ergodic Markov chain, whose invariant distribution is precisely the posterior distribution [Tie94]. In the rest of this section, an introduction to some of the most common MCMC

algorithms is described. This section only mentions the most well-known theoretical results on the asymptotic behavior of Markov chains. It does not claim to serve as a general reference in Markov chain theory. The author recommends [MT12; Dou+18; Nor98; RR04] for a more complete review of Markov chain theory.

1.6.1 Preliminary concepts

1.6.1.1 Markov chains

Definition 1.6.1. *Consider a discrete-time stochastic process $(X_n)_{n\in\mathbb{N}}$ with values in the countable state space $\mathcal{S}$ such that $X_n \in \mathcal{S}$ for all $n \in \mathbb{N}$. The discrete-time stochastic process $(X_n)_{n\in\mathbb{N}}$ is a Markov chain on the countable state space $\mathcal{S}$ if it satisfies the **Markov property**.*

$$\mathbb{P}(X_n = x_n | X_{n-1} = x_{n-1}, ..., X_0 = x_0) = \mathbb{P}(X_n = x_n | X_{n-1} = x_{n-1}) \tag{1.166}$$

for any $n \geq 1$ and for any $(x_0, ..., x_n) \in \mathcal{S}^n$.

Definition 1.6.2. *An homogeneous discrete-time Markov chain on the countable set $\mathcal{S}$ is a Markov chain which verifies for all $n \geq 1$ and $i, j \in \mathcal{S}$:*

$$\mathbb{P}(X_n = j | X_{n-1} = i) = P_{i,j}. \tag{1.167}$$

The transition probabilities from state i to state j are independent of n. The probabilities of transition $P_{i,j}$ can be represented as a transition matrix in finite state space. The knowledge of the initial state X_0 and the transition probabilities $P_{i,j}$ are enough to describe the whole chain.

In this thesis, the Markov chains considered in the MCMC methods lie in a continuous state space $\mathcal{X} \subset \mathbb{R}^p$. Let us focus then on the generalization of Markov chains to continuous non-countable state spaces. The generalization of the transition matrix is the transition kernel or transition density.

Definition 1.6.3. *Consider a continuous state space $\mathcal{X}$ and $\mathcal{B}(\mathcal{X})$ is the Borel set of $\mathcal{X}$. The function $P\colon \mathcal{B}(\mathcal{X}) \times \mathcal{X} \longrightarrow [0,1]$ is said to be a Markov kernel if it verifies the following conditions:*

- *For all $x \in \mathcal{X}$, the application $p_x\colon A \mapsto P(x, A)$ is a probability measure on $(\mathcal{X}, \mathcal{B}(\mathcal{X}))$.*
- *For all $A \in \mathcal{B}(\mathcal{X})$, the application $p_A\colon x \mapsto P(x, A)$ is measurable.*

The transition kernel is the continuous generalization of the transition probabilities $P_{i,j}$.

Definition 1.6.4. *Let P be a Markov kernel on $\mathcal{X} \times \mathcal{B}(\mathcal{X})$. The kernel is said to have a kernel density $p\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}^+$ if for $x \in \mathcal{X}$ and $A \in \mathcal{B}(\mathcal{X})$:*

$$P(x, A) = \int_A p(x, y)\mu(dy) \tag{1.168}$$

where μ is the Lebesgue measure on $\mathcal{X}$.

Definition 1.6.5. *Let $(X_n)_{n\in\mathbb{N}}$ be discrete-time stochastic process on the continuous state space $\mathcal{X}$, and P a Markov kernel on $\mathcal{X} \times \mathcal{B}(\mathcal{X})$. $(X_n)_{n\in\mathbb{N}}$ is a time-homogeneous Markov*

chain on $\mathcal{X}$ with initial measure μ if it verifies the Markov property for continuous state space:

$$\mathbb{P}\left(X_n \in A_n, ..., X_0 \in A_0\right) = \int_{A_0} ... \int_{A_{n-1}} P(A_n, y_{n-1}) P(dy_{n-1}, y_{n-2}) ... P(dy_1, y_0) \mu(dy_0). \quad (1.169)$$

In everything that follows, we define $(X_n)_{n\in\mathbb{N}}$ as a time-homogeneous Markov chain on the continuous state space $\mathcal{X}$. Moreover, we consider that the chain starts at a point $x \in \mathcal{X}$ such that the initial distribution is given by $\mu = \delta_x$ where δ_x is the Dirac distribution centered on x. The probability $\mathbb{P}\left(X_n \in A_n, ..., X_0 \in A_0\right)$ is denoted $P^n(A_n, x)$ in the rest of this work.

Definition 1.6.6. *For $x \in \mathcal{X}$, the probability measure conditioned by $X_0 = x$ is denoted as $\mathbb{P}_x$ and is such that for any $A \in \mathcal{B}(\mathcal{X})$, $\mathbb{P}_x(A) = \mathbb{P}(A|X_0 = x)$.*

Proposition 1.6.1. *Consider two Markov kernels P and Q defined on $\mathcal{B}(\mathcal{X}) \times \mathcal{X}$. The product kernel $P \circ Q \colon \mathcal{B}(\mathcal{X}) \times \mathcal{X}$ defined by:*

$$P \circ Q(A, x) = \int_{\mathcal{X}} P(A, y) Q(dy, x) \quad (1.170)$$

is also a Markov kernel. The product is not commutative though. Similarly, we define the iterated kernel product P^n for $n \geq 1$ by $P^n = P \circ ... \circ P$.

The associativity of Markov kernels allows us to define transitions over multiple time steps. For a Markov chain with transition kernel P and such that $X_{n_0} = x$, the distribution of the chain after $n \geq 1$ transitions is given by the transition kernel $P^n(\cdot, x)$.

1.6.1.2 Invariant measure and reversibility

Definition 1.6.7 (Invariant measure)**.** *A measure π on the measurable space $(\mathcal{X}, \mathcal{B}(\mathcal{X}))$ is said to be invariant for the Markov kernel P if for any $A \in \mathcal{B}(\mathcal{X})$:*

$$\pi(A) = \int_{\mathcal{X}} P(x, A) \pi(dx). \quad (1.171)$$

Proposition 1.6.2. *If π is a measure with density $\pi(x)$ for $x \in \mathcal{X}$, and the Markov kernel P has density p, then the measure is invariant for the kernel P if and only if for $x, y \in \mathcal{X}$:*

$$\pi(y) = \int_{\mathcal{X}} p(x, y) \pi(x) dx. \quad (1.172)$$

Definition 1.6.8. *Consider a Markov chain $(X_n)_{n\in\mathbb{N}}$ with Markov kernel P and kernel density p. The chain is said to follow the detailed balance condition if there exists a probability distribution π such that for any $x, x' \in \mathcal{X}$ we have:*

$$\pi(x) p(x, x') = \pi(x') p(x', x). \quad (1.173)$$

Such a Markov chain is said to be reversible.

Theorem 1.6.1. *Consider a reversible Markov chain $(X_n)_{n\in\mathbb{N}}$ and π a probability distribution that verifies the detailed balance condition. Then π is an invariant probability measure for the Markov chain.*

The detailed balance condition is extensively used in most MCMC algorithms since it allows us to easily prove that the invariant distribution is the target distribution. Many studies have looked at the necessary conditions for the existence of invariant measures and invariant probability measures. In this work, we do not further dwell on the subject as the Markov chain theory is used mainly for MCMC methods, in which the invariant probability measure is found with the detailed balance property.

1.6.1.3 Irreducibility and recurrence

Definition 1.6.9 (Return time)**.** *Let $A \in \mathcal{B}(\mathcal{X})$. The return time to A, denoted T_A is the random variable defined by:*

$$T_A = \begin{cases} +\infty & \text{if } X_n \notin A \text{ for all } n \in \mathbb{N}. \\ \min\{n \geq 1 \text{ s.t. } X_n \in A\} & \text{otherwise.} \end{cases} \tag{1.174}$$

Besides, for $x \in \mathcal{X}$, we say that x leads to A if $\mathbb{P}_x(T_A < +\infty) > 0$. In other words, there is a non-zero probability that the chain reaches A when starting from x.

Definition 1.6.10 (Number of visits)**.** *The number of visits to a set $A \in \mathcal{B}(\mathcal{X})$ is the random variable η_A defined by:*

$$\eta_A = \sum_{k=1}^{+\infty} \mathbb{1}_{\{X_k \in A\}} \tag{1.175}$$

which can be infinite.

Definition 1.6.11 (Irreducibility)**.** *Consider a σ-finite measure ϕ on the measurable space $(\mathcal{X}, \mathcal{B}(\mathcal{X}))$. The Markov chain $(X_n)_{n\in\mathbb{N}}$ is said to be ϕ-irreducible if for any $A \in \mathcal{B}(\mathcal{X})$ such that $\phi(A) > 0$, and for all $x \in \mathcal{X}$:*

$$\mathbb{P}_x(\tau_A < +\infty) > 0. \tag{1.176}$$

More intuitively, the ϕ-irreducibility states that for any starting point x and any set A with non-zero measure under ϕ, there is a non-zero probability that the chain will reach A in a finite number of transitions.

Definition 1.6.12 (Recurrence)**.** *Consider a σ-finite measure ϕ on the measurable space $(\mathcal{X}, \mathcal{B}(\mathcal{X}))$. The Markov chain $(X_n)_{n\in\mathbb{N}}$ is said to be ϕ-recurrent if for any $A \in \mathcal{B}(\mathcal{X})$ such that $\phi(A) > 0$, and for all $x \in \mathcal{X}$:*

$$\mathbb{E}_x[\eta_A] = +\infty. \tag{1.177}$$

The chain is expected to visit any set with non-zero measure under ϕ an infinite number of times, whatever the starting point.

Definition 1.6.13 (Periodicity)**.** *Let $(X_n)_{n\in\mathbb{N}}$ be a Markov chain with invariant distribution π. The chain is said to be periodic with period d if there exists $d \geq 2$ and disjoint subsets $A_1, ..., A_d \in \mathcal{B}(\mathcal{X})$ such that $\pi(A_i) > 0$ and for $1 \leq i \leq d-1$ and $x \in A_i$ we have $P(x, A_{i+1}) = 1$ and for $x \in A_d$, we have $P(x, A_1) = 1$. If there is no such $d \geq 2$, the chain is said to be aperiodic.*

Remark. We may refer to a chain as an irreducible chain without further emphasizing the corresponding measure ϕ. The same goes for the recurrence. In most MCMC methods, ϕ can be identified as Lebesgue measure μ on $\mathcal{X}$.

Definition 1.6.14. *The total variation distance between two probability measures p and q is defined as:*

$$\|p(\cdot) - q(\cdot)\|_{TV} = \sup_{A \in \mathcal{B}(\mathcal{X})} |p(A) - q(A)|. \tag{1.178}$$

This distance verifies the usual properties of a distance. It serves as a comparison metric between probability distributions.

Let us now introduce the ergodic theorem for Markov chains in general state space.

Theorem 1.6.2. *[Ergodic theorem for irreducible chains] Let $(X_n)_{n\in\mathbb{N}}$ be an irreducible recurrent aperiodic Markov chain with transition kernel P. If the chain admits an invariant probability measure π then:*

$$\lim_{n\to+\infty} \|P^n(x,\cdot) - \pi(\cdot)\|_{TV} = 0 \tag{1.179}$$

for π-almost everywhere $x \in \mathcal{X}$.

Furthermore, for any $f \in L^1(\pi)$, that is any function such that $\int_{\mathcal{X}} |f(x)|\pi(dx) < +\infty$ we have a strong law of large numbers for the Markov chain:

$$\frac{1}{n}\sum_{i=0}^{n} f(X_i) \xrightarrow[n\to+\infty]{a.s.} \int_{\mathcal{X}} f(x)\pi(dx). \tag{1.180}$$

This theorem is one of the most important results of Markov chains theory. It states that for any function f such that $\int_{\mathcal{X}} |f(x)|\pi(dx) < +\infty$, the expectation of the function can be approximated as an expectation over the Markov chain. This can be used to compute moments of the probability distribution or credibility regions.

The main assumption of this theorem is the existence of the invariant probability measure. In most MCMC algorithms, the existence is obtained with the detailed balance condition.

The theorem is only valid π-almost everywhere since it may be possible to exhibit a null set $\mathcal{X}_0 \subset \mathcal{X}$ (w.r.t. the measure π) such that the chain does not converge for starting points $x \in \mathcal{X}_0$. A stronger condition is thus required.

Definition 1.6.15 (Harris recurrence)**.** *Once again, let ϕ be a σ-finite measure on the measurable space $(\mathcal{X}, \mathcal{B}(\mathcal{X}))$. The Markov chain $(X_n)_{n\in\mathbb{N}}$ is said to be Harris ϕ-recurrent if for any $A \in \mathcal{B}(\mathcal{X})$ such that $\phi(A) > 0$, and for all $x \in \mathcal{X}$:*

$$\mathbb{P}_x\left(\eta_A = +\infty\right) = 1. \tag{1.181}$$

The ϕ-Harris recurrence guarantees almost surely that the chain will reach A in a finite number of transitions.

The Harris recurrence is stronger than the standard ϕ-irreducibility but it allows to generalize the previous result to all $x \in \mathcal{X}$.

Theorem 1.6.3 (Ergodic theorem for Harris-recurrent chains [Har56])**.** *Let $(X_n)_{n\in\mathbb{N}}$ be a Harris-recurrent and aperiodic Markov chain with transition kernel P. If the chain admits an invariant probability measure π then:*

$$\lim_{n\to+\infty} \|P^n(x,\cdot) - \pi(\cdot)\|_{TV} = 0 \tag{1.182}$$

for all $x \in \mathcal{X}$. Besides, the strong law of large numbers (1.180) holds.

1.6.1.4 Convergence rates for ergodic Markov chains

Let us now discuss briefly the rates of convergence of Markov chains. The main results discussed here are presented in [Twe75].

Definition 1.6.16 (Uniform ergodicity)**.** *A Markov chain $(X_n)_{n\in\mathbb{N}}$ with transition kernel P and invariant measure π is said to be uniformly ergodic if there exists $M < +\infty$ and $r < 1$ such that for all $x \in \mathcal{X}$:*

$$\|P^n(x,\cdot) - \pi(\cdot)\|_{TV} \leq Mr^n. \tag{1.183}$$

Definition 1.6.17. *The set $C \in \mathcal{B}(\mathcal{X})$ is said to be small if there exists a probability measure ν, some $m \in \mathbb{N}$ and $\varepsilon > 0$ such that for all $x \in \mathcal{X}$ and $A \in \mathcal{B}(\mathcal{X})$:*

$$P^m(x,A) \geq \varepsilon\nu(A). \tag{1.184}$$

Definition 1.6.18 (Doeblin's condition)**.** *Let $(X_n)_{n\in\mathbb{N}}$ be a Harris ϕ-recurrent Markov chain on $\mathcal{X}$. The chain is said to satisfy the minimization condition if $\mathcal{X}$ is a small set.*

Theorem 1.6.4. *If $(X_n)_{n\in\mathbb{N}}$ is an aperiodic Markov chain that verifies Doeblin's condition then it is uniformly ergodic.*

Uniform ergodicity is a strong convergence result but it is often difficult to verify that the whole space $\mathcal{X}$ is a small set. A looser convergence rate is often introduced to relax the convergence assumptions.

Definition 1.6.19 (Geometric ergodicity)**.** *A Markov chain $(X_n)_{n\in\mathbb{N}}$ with transition kernel P and invariant measure π is said to be geometrically ergodic if there exists $M\colon \mathcal{X} \to \mathbb{R} \cup \{+\infty\}$ and $r < 1$ such that for all $x \in \mathcal{X}$:*

$$\|P^n(x,\cdot) - \pi(\cdot)\|_{TV} \leq M(x)r^n \tag{1.185}$$

and $M(x) < +\infty$ π-almost everywhere.

Definition 1.6.20 (Drift condition)**.** *A Markov chain $(X_n)_{n\in\mathbb{N}}$ with transition kernel P is said to satisfy a drift condition [Twe01] if there exists a function $V\colon \mathcal{X} \to [1,+\infty]$, a small set $C \in \mathcal{B}(\mathcal{X})$, $b \in \mathbb{R}$ and $0 < \lambda < 1$ such that for all $x \in \mathcal{C}$:*

$$\int_{\mathcal{X}} P(x,dy)V(y) \leq \lambda V(x) + b\mathbb{1}_C(x). \tag{1.186}$$

The drift condition is a condition to reach geometric ergodicity.

Theorem 1.6.5. *Let $(X_n)_{n\in\mathbb{N}}$ be a ϕ-irreducible aperiodic Markov chain with invariant distribution π. Suppose that the chain satisfies a drift condition for a small set C, some $0 < \lambda < 1$, $b \in \mathbb{R}$ and some function $V\colon \mathcal{X} \to [1,+\infty]$ with at least one $x \in \mathcal{X}$ such that $V(x) < +\infty$, then the chain is geometrically ergodic.*

The drift condition and Doeblin's condition are the two main assumptions that are used to derive uniform and geometric ergodicity respectively. In this work, we will not detail the convergence properties of the Markov chains generated by MCMC algorithms, though we show that Metropolis-Hastings does display geometric ergodicity, which is a necessary condition for the Central Limit Theorem for Markov chains.

Theorem 1.6.6 (Central Limit Theorem for Markov chains). *Consider $(X_n)_{n\in\mathbb{N}}$ an aperiodic Harris ϕ-recurrent Markov chain with invariant probability measure π. Suppose the chain is geometrically ergodic and let $f \in L^1(\pi)$. If $\int_{\mathcal{X}} f^2(x)\pi(dx) < +\infty$ then:*

$$\sqrt{n}\left(\frac{1}{n}\sum_{i=1}^{+\infty} f(X_i) - \mathbb{E}_\pi[f]\right) \xrightarrow[n\to+\infty]{\mathcal{L}} \mathcal{N}\left(0, \sigma_f^2\right) \tag{1.187}$$

with $\mathbb{E}_\pi[f] = \int_{\mathcal{X}} f(x)\pi(dx)$ and $\sigma_f^2 = \mathbb{V}[f(X_0)] + 2\sum_{i=1}^{+\infty} \mathrm{Cov}[f(X_0), f(X_i)] < +\infty$.

This central limit theorem provides an asymptotic variance for the estimator of $\mathbb{E}_\pi[f]$. This variance differs from the asymptotic variance in the standard CLT with iid samples since there are correlations between successive samples.

In numerical works, it is possible to define a decorrelation time τ, which is the average number of iterations required to effectively create independent samples.

Definition 1.6.21. *For a Markov chain $(X_n)_{n\in\mathbb{N}}$, the decorrelation time τ is defined by:*

$$\tau = 1 + 2\sum_{i=1}^{+\infty} \mathrm{Corr}(X_0, X_i) \tag{1.188}$$

where $\mathrm{Corr}(X, Y) = \frac{\mathrm{Cov}(X,Y)}{\sqrt{\mathbb{V}[X]\mathbb{V}[Y]}}$ for two random variables X and Y.

From the decorrelation time, one can define the effective sample size L_{eff} of a finite Markov chain $(X_n)_{0\le n\le L}$ as $L_{\text{eff}} = \frac{L}{\tau}$. This effective sample size is the effective number of uncorrelated samples in the chain.

The decorrelation time and effective sample size are important when estimating the asymptotic properties of MCMC algorithms.

Having introduced the main theoretical results on discrete-time Markov chains, our attention now shifts to several MCMC algorithms utilized throughout this work.

1.6.2 Metropolis-Hastings

Consider a target probability distribution π, whose density is known within a multiplicative constant. The Metropolis-Hastings (MH) algorithm [Met+53; Has70] aims at sampling a Markov chain whose invariant distribution is precisely the target distribution π. The MH algorithm only requires to have access to the target distribution density, within a multiplicative constant. Since the posterior distribution in Bayesian statistics can be expressed as the product of a prior and a likelihood, multiplied by an intractable integral, MH seems particularly suited to such a problem.

1.6.2.1 Pseudo-code description

The algorithm works as follows. Starting from an initial point X_0, a new candidate point $\widehat{X}$ is sampled randomly from a proposal distribution with density $q(x, y)$ describing the probability density from point x to y. Then, the candidate point is either accepted with probability $\alpha(X_0, \widehat{X}) = \min\left\{1, \frac{\pi(\widehat{X})q(\widehat{X}|X_0)}{\pi(X_0)q(X_0|\widehat{X})}\right\}$, in which case we set $X_1 = \widehat{X}$ or it is rejected with probability $1 - \alpha(\widehat{X}, X_0)$ and then $X_1 = X_0$. The process

is then iterated until we reach the number of samples desired. The full algorithm is described in Algorithm 1.

Algorithm 1: Metropolis-Hastings pseudo-code

```
Result: Sampling of the target distribution π
Choose the chain starting point X_0 and desired length K;
while Chain length < K do
    Generate candidate X̂ from proposal distribution described by a kernel
     density q;
    Evaluate α(X̂, X_n);
    Generate u ~ U[0,1] from a uniform distribution on [0,1];
    if α(X̂, X_n) > u then
        Add X̂ to the chain X_{n+1} = X̂;
    else
        Add X_n to the chain X_{n+1} = X_n;
    end
end
```

1.6.2.2 Reversibility and ergodicity

In what follows the proposal distribution is taken as a Gaussian distribution centered on the current point and with covariance matrix Σ. The corresponding proposal density is given by:

$$q(x,y) = \frac{1}{\sqrt{2\pi|\Sigma|}} \exp\left[-\frac{1}{2}(y-x)^T\Sigma^{-1}(y-x)\right] \tag{1.189}$$

where a^T is the transpose of vector $a \in \mathcal{X}$ and $|\Sigma|$ is the determinant of Σ. For this specific choice of proposal distribution, the Metropolis-Hastings algorithm is referred to as the Random Walk Metropolis (RWM) algorithm. We further assume that the target probability distribution has a density denoted by $\pi(x)$ for $x \in \mathcal{X}$.

Proposition 1.6.3. *Let q be the density of the proposal distribution. The transition density for the Metropolis-Hastings algorithm is given by:*

$$p(x,y) = \alpha(x,y)q(x,y) + \left(1 - \int_{\mathcal{X}} \alpha(x,u)q(x,u)du\right) \mathbb{1}_{\{x\}}(y) \tag{1.190}$$

where $\alpha(x,y) = \min\left\{1, \frac{\pi(y)}{\pi(x)}\right\}$ and $\mathbb{1}_{\{x\}}(y)$ is 1 if $x = y$ and 0 otherwise.

For $A \in \mathcal{B}(\mathcal{X})$ and $x \in \mathcal{X}$, the Markov kernel associated with this density is:

$$P(x,A) = \int_A Q(x,dy)\alpha(x,y) + \int_A \delta_x(dy)\left(1 - \int_{\mathcal{X}} \alpha(x,u)Q(x,du)\right)dy \tag{1.191}$$

where μ is the Lebesgue measure on $\mathcal{X}$ and δ_x is the Dirac measure centered on x.

Proposition 1.6.4. *The Markov chain obtained by the Random Walk Metropolis algorithm satisfies the detailed balance condition for the target distribution π. Besides, the chain is π-irreducible and aperiodic which is sufficient to conclude that it is ergodic.*

Proof. Let $x, y \in \mathcal{X}$. We want to show that $\pi(x)q(x,y) = \pi(y)q(y,x)$. If $x = y$, the detailed balance is trivially verified. Let us focus on the case $x \neq y$. In this case, the kernel density reduces to the left term in (1.190) and thus:

$$\pi(x)p(x,y) = \min\{\pi(x), \pi(y)\}\, q(x,y) \tag{1.192}$$

and since the proposal kernel is symmetric $q(x,y) = q(y,x)$:

$$\pi(x)p(x,y) = \min\{\pi(x), \pi(y)\}\, q(y,x) = \pi(y)p(y,x). \tag{1.193}$$

We can conclude that the MH kernel verifies the detailed balance equation for the target distribution π.

Since the proposal distribution is Gaussian there is a non-zero probability to reach any state $A \in \mathcal{X}$ provided that $\mu(A) > 0$ and thus the chain is μ-irreducible.

Finally, we need to prove the aperiodicity. A possible approach is to consider the proposition from [RCC99] which states that the chain is aperiodic if the probability of remaining in place is non-zero. That is if for all $x \in \mathcal{X}$ and for all $n \in \mathbb{N}$:

$$\mathbb{P}\left(X_{n+1} = x | X_n = x\right) > 0. \tag{1.194}$$

This probability can be written as:

$$\mathbb{P}\left(X_{n+1} = x | X_n = x\right) = \int_{\mathcal{X}} q(x,y)(1 - \alpha(x,y))dy \tag{1.195}$$

and since the density π is continuous on $\mathcal{X}$ there exists a subset $\mathcal{X}_0 \subset \mathcal{X}$ with $\mu(\mathcal{X}_0) > 0$ such that $\alpha(x,y) < 1$ for all $y \in \mathcal{X}_0$. This is enough to conclude that $\mathbb{P}\left(X_{n+1} = x | X_n = x\right) > 0$.

□

Remark. This detailed balance is verified for any proposal kernel density even if it is not symmetric, as long as the acceptance probability is modified to $\alpha(x,y) = \min\left\{1, \frac{q(x,y)\pi(y)}{q(y,x)\pi(x)}\right\}$.

Moreover, one can also show that the Metropolis-Hastings kernel is uniformly ergodic [JH00].

Proposition 1.6.5. *The Metropolis-Hastings kernel defined in equation (1.191) verifies Doeblin's condition and is uniformly ergodic.*

Proof. Let $A \in \mathcal{B}(\mathcal{X})$ and $x \in \mathcal{X}$. We will show that there exists $\varepsilon > 0$ and a measure ν such that $P(x,A) \geq \varepsilon\nu(A)$. Introducing μ the Lebesgue measure on $\mathbb{R}^p$ we have:

$$\begin{aligned} P(x,A) &= \int_A Q(x,dy)\alpha(x,y) + \int_A \delta_x(dy)\left(1 - \int_{\mathcal{X}} \alpha(x,u)Q(x,du)\right)dy \\ &\geq \int_A q(x,y)\alpha(x,y)\mu(dy) \geq \int_A q(x,y)\mu(dy). \end{aligned} \tag{1.196}$$

Since $\mathcal{X}$ is a compact set and the Gaussian density q is continuous it is bounded. We introduce $\varepsilon = \min_{x,y\in\mathcal{X}} q(x,y)$. Since the Gaussian density is non-zero $\varepsilon > 0$ and:

$$P(x,A) \geq \int_A q(x,y)\mu(dy) \geq \varepsilon\mu(A). \tag{1.197}$$

The kernel verifies Doeblin's condition, it is thus uniformly ergodic. □

1.6.2.3 Optimal target rate

With Metropolis-Hastings sampling, the proposal distribution can affect the convergence speed of the Markov distribution toward the invariant distribution. If the proposal distribution is too narrow, the candidate points are close to the current point and the sampling of the entire target distribution is slow. On the other hand, if the candidate points are too far away from the current point, most of them will be rejected.

The key feature that quantifies this phenomenon is the acceptance rate. In [RR98], an optimal target acceptance rate has been derived from optimal scaling theory [GGR97] for high dimensional target distributions, assuming the measure can be written as a product measure of the marginal densities.

Proposition 1.6.6. *Consider the Markov chain obtained by RWM $(X_n)_{n\in\mathbb{N}}$ with proposal distribution $q \sim \mathcal{N}\left(0, \sigma_d^2 \mathcal{I}_d\right)$ where $\mathcal{I}_d$ is the identity matrix in dimension $d \geq 2$. We assume that its target invariant measure is given by a product density such that $\pi(dx) = \prod_{i=1}^{p} \pi(x_i)\mu(dx_i)$ where μ is the Lebesgue measure on $\mathbb{R}$. The chain converges weakly to the Langevin diffusion process $(U_t)_{t\geq 0}$ given by the Stochastic Differential Equation :*

$$dU_t = \sqrt{h(l)}dB_t + h(l)\frac{f'(U_t)}{2f(U_t)}dt \tag{1.198}$$

where $l = \sqrt{d-1}\sigma_d$ and $h(l) = 2l^2\Phi\left(-\frac{l\sqrt{E}}{2}\right)$ where Φ is the Cumulative Distribution Function (CDF) of the standard normal distribution and $E = \mathbb{E}_f\left[\left(\frac{f'(X)}{f(X)}\right)^2\right]$. The quantity $h(l)$ is called the speed of the diffusion process and quantifies the speed of convergence of the diffusion process. Let $a_n(l)$ be the mean acceptance rate of the RWM defined by :

$$a_n(l) = \int_{\mathcal{X}}\int_{\mathcal{X}} \pi(x)q(x,y)\alpha(x,y)\mu(dx)\mu(dy). \tag{1.199}$$

The speed of diffusion is maximized for $\hat{l} = \frac{2.38}{\sqrt{E}}$ in the limit of high dimensional target distributions we have:

$$\lim_{n\to+\infty} a_n(l) = a(l) = \frac{h(l)}{l^2} \tag{1.200}$$

and in particular $a(\hat{l}) \simeq 0.23$.

This particular result stands in the limit of very high dimensional target distributions. Nonetheless, it may serve as an empirical rule for tuning the scaling parameter in the proposal distribution.

The theory on optimal scaling has been extended to non-iid target distributions [Bed08]. In this work, the target acceptance rate of 0.23 is only used as an empirical guideline and will not be further discussed.

1.6.3 Adaptive Metropolis

In the context of neutron noise analysis, the posterior distributions often lie on a small support. In that case, a simple RWM tends to sample very slowly the posterior

distribution since most of the candidate points are outside the support.

To bypass this difficulty, let us introduce the Adaptive Metropolis procedure (AM) introduced in [HST01].

1.6.3.1 Global covariance adaptation

Consider once again a Gaussian proposal distribution $q_n \sim \mathcal{N}(X_n, \mathcal{C}_n)$ which can change at each iteration of the MCMC sampling. The purpose of the AM algorithm is to update the proposal covariance $\mathcal{C}_n$ to sample candidate points closer to the true target distribution. The proposal covariance is set to the empirical covariance of all the previous points of the chain (with a scale factor s_n) such that:

$$\mathcal{C}_n = \frac{s_n}{n-1} \sum_{i=0}^{n-1} (X_i - \overline{X}_n)^T \tag{1.201}$$

where $\overline{X}_n = \frac{1}{n} \sum_{i=0}^{n-1} X_i$. The proposal covariance can be obtained with a faster recursive formula:

$$\mathcal{C}_n = \frac{n-1}{n} \mathcal{C}_{n-1} + \frac{s}{n} \left(n \overline{X}_{n-1} \overline{X}_{n-1}^T - (n+1) \overline{X}_n \overline{X}_n + X_n X_n^T \right). \tag{1.202}$$

The main difficulty introduced by the AM algorithm is that the chain is not Markovian anymore since $\mathcal{C}_n$ depends on all the previous states. This difficulty can be overcome by considering $(X_n, \mathcal{C}_n)$ as a Markovian process. From this, the ergodicity of the chain can be derived (see for example [AM06]).

1.6.3.2 Local covariance adaptation

A variant of the AM algorithm, known as Adaptive Proposal (AP) has been introduced in [HST99]. The idea is to change the covariance by considering only a limited history of the Markov chain instead of the full chain. Instead of estimating the empirical covariance on all the points of the chain, it is evaluated on the last H points where $H \in \mathbb{N}$. Then the covariance is given by $\mathcal{C}_n = s_d \times \mathrm{Cov}(X_{n-H}, ..., X_{n-1})$. The recursive formula in that case requires keeping in memory the point X_{n-H}.

$$\mathcal{C}_{n+1} = \mathcal{C}_n + \frac{s_d}{H} \left(X_{n+1} X_{n+1}^T - X_{n-H} X_{n-H}^T - (H+1) \left(\overline{X_{n+1}}\ \overline{X_{n+1}}^T - \overline{X_n}\ \overline{X_n}^T \right) \right). \tag{1.203}$$

The average $\overline{X_n}$ is calculated on the last $H+1$ points.

$$\overline{X_n} = \frac{1}{H+1} \sum_{i=0}^{H} X_{n-i}. \tag{1.204}$$

It is updated recursively with the following formula and needs to be kept in memory. Similarly, X_{n-H} needs to be remembered.

$$\overline{X_{n+1}} = \overline{X_n} + \frac{X_{n+1} - X_{n-H}}{H+1}. \tag{1.205}$$

The Markov chain generated by the AP algorithm is ergodic however the invariant distribution slightly differs from the actual target distribution π. In [HST99], the author showed that the bias does not alter the target distribution significantly.

1.6.4 Hamiltonian Monte Carlo

Hamiltonian Monte Carlo [Bet17] is a sampling method based on Hamiltonian mechanics introduced in [Nea+11]. This method introduces auxiliary momentum variables r and extends the sampling space by introducing a joint density $p(x, r)$. The momentum variables are drawn from Gaussian distribution, usually with identity covariance $r \sim \mathcal{N}(0, \mathcal{I})$. The joint density is given by $p(x, r) = \exp(-H(x, r))$ where H is the Hamiltonian defined by :

$$H(x, r) = \frac{1}{2} r^T M^{-1} r - \log \pi(x|\mathcal{D}_{\text{train}}) \tag{1.206}$$

where M is known as the mass matrix. Intuitively, the first term represents a potential energy term and the second is a kinetic energy term. The exploration of the extended space is a trade-off between the two. Hence the goal of the algorithm is to initialize random momentum variables to explore various trajectories in the extended space, based on the Hamilton equations.

$$\frac{dx}{dt} = \frac{\partial H}{\partial r} = M^{-1} r \tag{1.207}$$

$$\frac{dr}{dt} = -\frac{\partial H}{\partial x} = -\frac{d \log \pi(x|\mathcal{D}_{\text{train}})}{dx}. \tag{1.208}$$

where t is the time.

1.6.4.1 Leap frog integration

In HMC, Hamilton equations are discretized in the so-called Euler-Maruyama scheme, or "leapfrog" integrator [Mao15]. Starting from some r_0 and x_0, and given a step size $h > 0$ and a number of integration steps L, the discretized numerical scheme is the following :

$$r_{n+\frac{1}{2}} = r_n - \frac{h}{2} \frac{\partial H}{\partial x}(x_n) \tag{1.209}$$

$$x_{n+1} = x + h \frac{\partial H}{\partial r}(r_{n+\frac{1}{2}}) \tag{1.210}$$

$$r_{n+1} = r_{n+\frac{1}{2}} - \frac{h}{2} \frac{\partial H}{\partial x}(x_{n+1}). \tag{1.211}$$

The leapfrog integrator is symplectic, meaning it preserves the volume in the dual space. However, though Hamilton equations include the conservation of the Hamiltonian H, the discretization loses this property. A corrective Metropolis-Hastings step is added after the L integration steps of the leapfrog integrator. Standard HMC is summarized in algorithm 2. Integrating along the trajectories allows to sample points that are well decorrelated while maintaining a very high acceptance rate in the Metropolis-Hastings step. HMC also remains very efficient in high-dimensional space. However, the main drawback is the manual tuning required to reach such

Algorithm 2: Hamiltonian Monte Carlo

Result: Sampling of the target posterior distribution $\pi(x|\mathbf{y})$
Initialize: Starting point $x^{(0)}$, chain length K, number of leapfrog integration steps L and stepsize h;
while $k < K$ **do**
  Sample momentum variables $r_0 \sim \mathcal{N}(0, \mathcal{I})$;
  Set $x_0 = x^{(k)}$;
  Evaluate the Hamiltonian $H_0 = H(x_0, r_0)$;
  for $n = 0$ *to* $L - 1$ **do**
    $r_{n+\frac{1}{2}} = r_n - \frac{h}{2}\frac{\partial H}{\partial x}(x_n)$;
    $x_{n+1} = x + h\frac{\partial H}{\partial r}(r_{n+\frac{1}{2}})$;
    $r_{n+1} = r_{n+\frac{1}{2}} - \frac{h}{2}\frac{\partial H}{\partial x}(x_{n+1})$;
  Set $\hat{x} = x_L$ and $\hat{r} = r_L$ and evaluate $\hat{H} = H(\hat{x}, \hat{r})$;
  Sample $u \sim \mathcal{U}(0, 1)$;
  if $u \leq \min\{1, \exp(H_0 - \hat{H})\}$ **then**
    Accept the candidate point $\hat{x}$ and set $x^{(k+1)} = \hat{x}$;
  else
    Reject the candidate point $\hat{x}$ and set $x^{(k+1)} = x^{(k)}$;

high performance. Both the stepsize h and the number of leapfrog integration steps L need to be tuned precisely to provide high performance.

1.6.4.2 No-U-Turn Sampler

To overcome this difficulty, the No-U-Turn-Sampler (NUTS) variant of HMC was introduced in [HG+14]. NUTS-HMC automatically tunes the number of integration steps and provides an adaptive scheme for the stepsize.

The description provided here is only succinct. The idea of the NUTS sampler is to start from the initial point (x, r) and build a binary tree. In the first stage, one leapfrog step is performed, randomly forward or backward (with the same probability). Then, two forward or backward steps are taken, then four, and so on. The process builds a binary tree that explores the Hamiltonian trajectory. The trajectory can be explored backward in time or forward to ensure time reversibility, which is necessary to guarantee the convergence of HMC toward the target distribution. The doubling process can also be interrupted if the Hamiltonian of the candidate point becomes too large compared to the current Hamiltonian.

1.6.4.3 Adaptive choice of path length

NUTS mainly addresses the problem of choosing the number of integration steps in the leapfrog scheme. A variant has also been introduced to adaptively tune the choice of step size.

The method is derived from Nesterov's primal-dual method for accelerated optimization of convex functions [Nes09]. The optimal step size is chosen to reach an optimal acceptance probability which was proven to be around 0.65 [Bes+13]. This method is known as dual averaging.

1.6.4.4 Adaptive choice of mass matrix

The choice of the mass matrix M can impact the performance of the HMC (and NUTS) sampling. Let us first explain the meaning behind the mass matrix. If we consider the Cholesky decomposition $M^{-1} = L^T L$, and introduce $\tilde{r} = Lr$ and $\tilde{x} = L^{-1}x$, Hamilton equations then translate to:

$$\frac{d\tilde{x}}{dt} = \tilde{r} \tag{1.212}$$

$$\frac{d\tilde{r}}{dt} = -\frac{d \log \pi(\tilde{x}|\mathcal{D}_{\text{train}})}{d\tilde{x}} \tag{1.213}$$

which corresponds to Hamilton equations with identity mass matrix, in the new phase space $(\tilde{x}, \tilde{r})$. The mass matrix then effectively reshapes the parameter space.

HMC has proven to perform excellent results on standard normal target distributions, even in very high dimensions. One idea is thus to reshape the target distribution to effectively standardize it. The natural choice of the mass matrix is thus to choose:

$$M^{-1} = \mathbb{E}\left[(X - \mathbb{E}[X])(X - \mathbb{E}[X])^T\right] \tag{1.214}$$

with $X \sim \pi$.

Of course, customarily the target distribution is not Gaussian and cannot be reshaped as a standard normal distribution. However, this choice of mass matrix still improves the overall performance of HMC-NUTS. Likewise, it is possible to take a mass matrix $M(x)$ dependent on the phase space position. This changes the Hamilton equations and the numerical integration scheme by adding a second term in $\frac{\partial H}{\partial x}$. This variant is known as Riemannian HMC [GC11].

1.6.4.5 Ergodicity

The ergodicity of HMC methods is not trivially established. The ergodic convergence of HMC-NUTS has been studied extensively in [Dur+23]. This work is the foundation for the following results.

Proposition 1.6.7. *We introduce the stepsize $h \in \mathbb{R}^+$ in the HMC-NUTS procedure, and the maximum tree size $K_m \in \mathbb{N}$. The mass matrix is denoted by M. If the potential energy, denoted by $U(x)$ for $x \in \mathcal{X}$ where $\mathcal{X}$ is a compact set, verifies the following assumptions:*

- *For any $x, x' \in \mathcal{X}$, there exists a constant $L \in \mathbb{R}^+$ such that:*

$$\|M^{-1}\nabla U(x) - M^{-1}\nabla U(x')| \leq L\|x - x'\|. \tag{1.215}$$

- *The stepsize and the maximum tree size verify the inequality:*

$$\left(1 + hL^{1/2} + \frac{1}{2}(hL^{1/2})^2 + \frac{1}{4}(hL^{1/2})^3\right)^{2^{K_m}} - 1 < \frac{1}{4}. \tag{1.216}$$

Then the Markov chain generated by HMC-NUTS is geometrically ergodic and verifies the Markov chain CLT.

Proof. We are relying on Theorem 16 from [Dur+23]. The hypothesis $H1$ corresponds to the first assumption in this proposition.

To verify $H5(h, K_m)$ from [Dur+23], we rely on proposition 10 which states that $H5(h, K_m)$ can be reached with $H1$ and $H2(h, K_m)$. The second assumption in our proposition corresponds to $H2(h, K_m)$. Thus with our two assumptions, we are in the conditions of Theorem 16. The HMC-NUTS kernel is thus $\mathcal{V}_a$-uniformly geometrically ergodic for some $a > 0$ where $\mathcal{V}_a = \exp(a|x|)$, meaning that there exists $0 < \rho < 1$ and $C > 0$ such that for any $x \in \mathcal{X}$:

$$\|P^n(x, \cdot) - \pi\|_{\mathcal{V}_a} \leq C\rho^n \mathcal{V}_a(x). \tag{1.217}$$

Since $\mathcal{X}$ is a compact set, $\mathcal{V}_a$ is bounded on $\mathcal{X}$ and thus there exists $C' > 0$ such that:

$$\|P^n(x, \cdot) - \pi\|_{\mathcal{V}_a} \leq C'\rho^n. \tag{1.218}$$

Finally, since $\mathcal{V}_a$ is positive and bounded there exists $0 \leq C_- \leq C_+$ such that for any measure ν:

$$C_-\|\nu\| \leq \|\nu\|_{\mathcal{V}_a} \leq C_+\|\nu\| \tag{1.219}$$

which is enough to conclude that HMC-NUTS is uniformly ergodic. □

HMC-NUTS has already been proven to be efficient in an extensive number of numerical cases. This result, however, provides a theoretical justification for these empirical works. In the rest of this thesis, we will not further develop the convergence property of HMC and MCMC in general as this is not the main focus of this work.

1.7 On optimization problems

The main optimization algorithms used in this thesis are described in this section. Consider a function

$$\begin{aligned} f\colon \mathcal{X} &\longrightarrow \mathbb{R} \\ x &\longmapsto f(x) \end{aligned}$$

where $\mathcal{X} \subset \mathbb{R}^p$. This function is assumed to be at least $\mathcal{C}^1$ on the domain $\mathcal{X}$. The objective is to find the global minimum x_* of the function:

$$x_* \in \operatorname{argmin}_{x\in\mathcal{X}} f(x). \tag{1.220}$$

1.7.1 Newton and pseudo-Newton methods

One of the earliest and simplest optimization methods is Newton's method. It is based on the assumption that $f \in \mathcal{C}^2(\mathcal{X})$. In the vicinity of a given point x, the second-order Taylor expansion is given by $f(x + t) = f(x) + \nabla f(x) \cdot t + \frac{1}{2} t^T \mathbf{H}(x) t + o(t^2)$ where $\nabla f(x)$ is the gradient of f at point x, $\mathbf{H}(x) \in \mathbb{R}^{p\times p}$ is the Hessian matrix of f at point x given by $\mathbf{H}(x)_{i,j} = \frac{\partial f}{\partial x_i \partial x_j}(x)$ for $1 \leq i, j \leq p$, and $a \cdot b$ is the standard Euclidean scalar product. The Newton's method works as follows. Consider an initial point $x_0 \in \mathcal{X}$ and then iteratively select the next point $x_{k+1} \in \mathcal{X}$ for $k \geq 0$ to minimize the second-order Taylor expansion. In practice, the new point x_{k+1} is given by:

$$x_{k+1} = x_k - \mathbf{H}(x)^{-1} \nabla f(x). \tag{1.221}$$

This method has numerous drawbacks. For example, it may converge only to saddle points and require the function to be strongly convex to guarantee the convergence toward a minimum. Besides, the Hessian of the function needs to be available and invertible. Even when the Hessian is invertible, it still has a complexity of $\mathcal{O}(p^3)$ because of the matrix inversion.

To cope with these caveats, a wide range of methods have been derived from Newton's method. They are referred to as quasi-Newton methods in the literature. Most quasi-Newton methods are based on an approximate computation of the Hessian matrix, either to reduce the computation time or because it is unavailable (or both). As an example, let us describe the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm [Bro70; Fle70; Gol70; Sha70] which is extensively utilized in this thesis.

The algorithm starts with an initial point x_0 and an initial Hessian approximation H_0. Then at each iteration, we first solve the system $H_k d_k = -\nabla f(x_k)$ to obtain the direction of descent d_k. Then, the magnitude l_k of the step in direction d_k is evaluated. This is done by solving the one dimensional optimization $l_k = \text{argmin}_{l \in \mathbb{R}^+} f(x_k + l d_k)$. The new point is then given by $x_{k+1} = x_k + l_k d_k$. Finally, a new estimation is computed with the approximation:

$$g_k = \nabla f(x_{k+1}) - \nabla f(x_k) \tag{1.222}$$

$$H_{k+1} = H_k + \frac{1}{l_k g_k^T d_k} g_k g_k^T - \frac{1}{d_k^T H_k d_k} H_k d_k d_k^T H_k^T. \tag{1.223}$$

The algorithm is iterated until the gradient norm is below an arbitrary threshold η. The BFGS algorithm also offers an easier computation of the Hessian inverses as they can be obtained iteratively from equation (1.223) with the Sherman-Morrison formula.

Many other methods have been developed and we will not mention them all here. One of the main algorithms utilized in this thesis is derived from the BFGS algorithm and is known as the Limited Memory BFSG (L-BFGS) [LN89; Byr+95] which relies on an approximate resolution of the system $H_k d_k = -\nabla f(x_k)$ which is less memory intensive and thus especially useful in higher dimensions. For this reason, this algorithm is vastly used in the optimization step when training large machine-learning models. Many more optimization techniques are found in the literature such as conjugate gradient methods or the Levenberg-Marquardt algorithm. For more details on the subject, the author refers to more complete work such as [Fle00; Wri06].

1.7.2 Simulated annealing and dual annealing

In this section, the focus is on the simulated annealing and dual annealing algorithms. These two methods are tackled in detail in [VL+87] and [Xia+97]. They are stochastic approaches designed for constrained and unconstrained global optimization of the function f.

Let us first introduce simulated annealing. The principle of simulated annealing is to explore randomly the domain, first by favoring exploration and then gradually shifting the focus on optimization. The trade-off between exploration and optimization is governed by a temperature parameter T, which is analogous to the temperature of a system in Boltzmann distribution, which has the form $p(E) \propto \exp\left(-\frac{E}{k_b T}\right)$. The system wants to minimize the energy E but the temperature induces stochastic motions allowing it to move around the global minimum. Simulated annealing is an iterative

procedure in which the temperature parameter T starts high and gradually decreases to reach the global minimum.

More formally, we consider a starting point $x_0 \in \mathcal{X}$. At each iteration n, a candidate point $\hat{x}$ is proposed. This candidate is accepted with probability $p_n(\hat{x})$ given by:

$$p_n(\hat{x}) = \min\left\{1, \exp\left(\frac{f(x_n) - f(\hat{x})}{T_n}\right)\right\} \tag{1.224}$$

where T_n is the current temperature.

The two main ingredients are the choice of the candidate point and the temperature scheme. The choice of temperature evolution also has a crucial impact on the performance of the algorithm and also on its guarantee of convergence. The choice of a log-decreasing temperature such as $T_n = \frac{C}{\log n}$ can provide a guarantee that the Markov chain reaches a neighborhood of the set of minima of the function f [Loc00; BJS86]. That is, for any $\varepsilon > 0$, we have:

$$\lim_{n \to +\infty} \mathbb{P}\left(x_n \in B_\varepsilon(f)\right) = 1 \tag{1.225}$$

$$B_\varepsilon(f) = \left\{x \in \mathcal{X} \; s.t. \; f(x) \leq \min_{x \in \mathcal{X}} f(x) + \varepsilon.\right\} \tag{1.226}$$

The candidate point $\hat{x}$ is chosen with a so-called proposal distribution $Q(X_n, \cdot)$ which can be for example Gaussian $Q(x, y) = |2\pi\Sigma|^{-1/2} \exp\left(-\frac{1}{2}(x_n - y)^T \Sigma^{-1}(x_n - y)\right)$ with some covariance matrix Σ. The choice of the proposal distribution can impact the performance of the method. The most common approach is to use a Gaussian distribution with a covariance proportional to the identity, in which case the algorithm is known as Classical Simulated Annealing (CSA). This work utilizes the simulated annealing from the *scipy* module, which uses a distorted Cauchy distribution. This variant was introduced as Fast Simulated Annealing (FSA) in [SH87].

A generalized approach to CSA and FSA, introduced as dual annealing, is used in *scipy*. This algorithm uses the FSA proposal distribution conjointly with an altered acceptance probability [TS96]. This method has proven more efficient than CSA and FSA in numerous cases.

1.8 Conclusion

This chapter introduces the general mathematical background and applicative context underpinning the research conducted within this thesis. It states the critical role of neutron multiplicity counting, in conjunction with gamma spectroscopy, as the gold standard experimental technique for assessing unknown nuclear materials. The nuclear security strategy thus requires a robust foundation of the underlying numerical and mathematical methods for real-time field responses and comprehensive experimental analysis. The main difficulty emerging is the resolution of an ill-posed inverse problem, which depends on very noisy observations of temporal correlations between fission-induced neutrons. The uncertainty quantification in this inverse problem resolution is necessary to guarantee the robustness and reliability of the nuclear safeguards strategy. This defines the overarching goal of this thesis.

How can one provide a fast and reliable uncertainty quantification methodology for the inverse problem resolution in neutron multiplicity counting?

An intuitive first response is to direct our effort toward a Bayesian resolution of the inverse problem. This is the primary goal in the following chapters, though it comes with secondary challenges.

Throughout this thesis, we strive to build surrogate models tailored for implementation within a Bayesian framework for resolving the inverse problem. The various theoretical developments and numerical experiments conducted in this thesis are presented in detail in the subsequent chapters. Each chapter delineates a distinct contribution, prefaced by a theoretical exposition of the current state-of-the-art and foundational concepts specific to the subject tackled.

Chapter 2

Gaussian process surrogate modeling

This chapter develops the work conducted to build efficient Gaussian process surrogate models to solve the inverse problem posed by the observations of noisy measurements of neutron correlations. Our goal is to use such models as emulators of the direct model in the Bayesian resolution.

We begin with a theoretical introduction to Gaussian processes which were willingly set aside in the previous chapter. Then, we focus on the motivation for GP surrogates before diving into the database creation and the training of the various models. Finally, we solve two inverse problems using the GP surrogates built in this chapter.

2.1 Theoretical overview

We start this chapter with a general introduction to Gaussian processes (GPs). Beginning with the basics for scalar GP regression, we then focus on multi-output GPs. We also discuss the sparse approximations that may be used for high-dimensional GP models. The following theoretical overview is mainly based on [RW+06; Gra20].

2.1.1 Scalar Gaussian processes

2.1.1.1 Introductory concepts

Definition 2.1.1. *A Gaussian process f is a stochastic process $f\colon \mathcal{X} \times \Omega \to \mathbb{R}$ such that for any finite subset $(x_1, ..., x_n) \in \mathcal{X}^n$ with $n \geq 1$, the random vector $f(\mathbf{x}) = (f(x_1), ..., f(x_n))^T$ follows a multivariate normal distribution.*
The distribution of a Gaussian process is completely determined by its mean function $m\colon \mathcal{X} \to \mathbb{R}$ and its covariance function $k\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$.
If f is a Gaussian process with mean function m and covariance function k, it is denoted by:

$$f \sim \mathcal{GP}\left(m(x), k(x, x')\right). \tag{2.1}$$

Let $\mathbf{x} = (x_1, ..., x_n) \in \mathcal{X}^n$. The distribution of $f(\mathbf{x})$ is given by:

$$f(\mathbf{x}) \sim \mathcal{N}\left(m(\mathbf{x}), K(\mathbf{x})\right) \tag{2.2}$$

where $m(\mathbf{x})$ is the column vector defined by $m(\mathbf{x}) = (m(x_1), ..., m(x_n))^T$ and $K(\mathbf{x})$ is the positive semi-definite matrix with elements $k(x_i, x_j)$ for $1 \leq i, j \leq n$.

The Gaussian process formalizes the concept of probability distributions over functions.

Unless stated otherwise, the input space throughout this chapter is $\mathcal{X} \subset \mathbb{R}^p$ for $p \geq 1$. In everything that follows, we set a probability space $(\Omega, \mathcal{F}, \mathbb{P})$.

Definition 2.1.2. *Let $f \sim \mathcal{GP}(m(x), k(x, x'))$, for some mean and covariance functions. For any element ω of the sample space Ω, the function f_ω defined by:*

$$\begin{aligned} f_\omega : \mathcal{X} &\longrightarrow \mathbb{R} \\ x &\longmapsto f(x, \omega) \end{aligned}$$

is called a realization of the Gaussian process f.

2.1.1.2 Covariance kernels

The covariance function k (also known as the covariance kernel) defines the regularity of the functions sampled from the Gaussian process. Different families of covariance functions exist and are used depending on the expected shape and properties of the function to be learned. A common choice is to use a stationary covariance function.

Definition 2.1.3. *A covariance function is said to be stationary if it only depends on $x - x'$. Furthermore, a stationary covariance function is said isotropic if it only depends on $\|x - x'\|$ where $\|\cdot\|$ is the Euclidean norm on $\mathbb{R}^p$. The notation $r = \|x - x'\|$ is introduced in that case.*

We can extend continuity and differentiability to stochastic processes to study the influence of covariance kernels on the regularity of the sampled functions. What follows is taken mostly from [Adl10].

Definition 2.1.4 (Mean-square continuity)**.** *The Gaussian process f is said to be mean-square continuous at $x \in \mathcal{X}$ if for any sequence $(x_k)_{k\in\mathbb{N}}$ such that $\|x_k - x\| \xrightarrow[k\to+\infty]{} 0$, we have $\mathbb{E}\left[|f(x_k) - f(x)|^2\right] \xrightarrow[k\to+\infty]{} 0$. If f is mean-square continuous for every $x \in \mathcal{X}$, it is said to be mean-square continuous on $\mathcal{X}$.*

Definition 2.1.5 (Mean-square differentiability)**.** *Let $f \sim \mathcal{GP}(m(x), k(x, x'))$. Let $(e_i)_{1\leq i\leq p}$ be a basis of $\mathbb{R}^p$ and $h > 0$. For any $x \in \mathcal{X}$, f is said to be mean-square differentiable at x if for all $1 \leq i \leq p$, the limit in mean-square of $\Delta_f(he_i)$ exists for $h \to 0^+$ where:*

$$\Delta_f(he_i) = \frac{f(x + he_i) - f(x)}{h}. \tag{2.3}$$

In that case, the mean-square derivative of f at x is defined as:

$$\frac{\partial f}{\partial x} = \left(\frac{\partial f}{\partial x_1}, ..., \frac{\partial f}{\partial x_p}\right) \tag{2.4}$$

where $\frac{\partial f}{\partial x_i} = \lim_{h\to 0^+}^{(m.s.)} \Delta_f(he_i)$ is the mean-square limit of $\Delta_f(he_i)$ when $h \to 0^+$.
If f is mean-square differentiable for every $x \in \mathcal{X}$, it is simply said to be mean-square differentiable on $\mathcal{X}$.
The definition is extended to higher-order derivatives in the same manner as traditional derivatives.

The concept of mean-square continuity and differentiability extends the basic notions of continuity and differentiability to stochastic processes. The continuity and differentiability can directly be inferred from the covariance function.

Proposition 2.1.1. *Let $f \sim \mathcal{GP}(m(x), k(x, x'))$ be a Gaussian process. If k is continuous on $\mathcal{X} \times \mathcal{X}$, then f is mean-square continuous on $\mathcal{X}$. Besides, if k is $2j$ times differentiable for $j \geq 1$, then f is j times mean-square differentiable.*

Let us now provide examples of standard covariance kernels for regression tasks.

Definition 2.1.6. *The isotropic squared exponential covariance function k_{SE}, with variance $\sigma^2 > 0$ and correlation length $l > 0$, is a stationary isotropic covariance function defined for $r \geq 0$ by :*

$$k_{\text{SE}}(r) = \sigma^2 \exp\left(-\frac{r^2}{2l^2}\right). \tag{2.5}$$

The correlation length is the same for all dimensions here. The squared-exponential covariance function is also known as the Radial Basis Function (RBF) kernel.

Definition 2.1.7. *The anisotropic squared exponential covariance function $k_{\text{SE,an}}$ is a stationary covariance function defined for $x, x' \in \mathcal{X}$ by :*

$$k_{\text{SE,an}}(x, x') = \sigma^2 \prod_{i=1}^{p} \exp\left(-\frac{(x_i - x_i')^2}{2l_i^2}\right) \tag{2.6}$$

where $l_i > 0$ is the correlation length associated to the i-th input dimension, for $1 \leq i \leq p$. The correlation lengths are different for each input dimension which may allow more flexibility at the cost of more hyperparameters.

Squared exponential covariance functions are infinitely differentiable. As a result, the corresponding Gaussian processes produce smooth realizations. The Gaussian process is infinitely mean-square differentiable. These covariance kernels are widely used in the field of Gaussian process modeling. However, the realizations of a GP with a squared exponential kernel may be too smooth, which in turn may hinder its predictive capabilities [Ste12]. In that regard, Matérn kernels may offer an alternative.

Definition 2.1.8. *The Matérn class of stationary isotropic covariance functions $k_{\text{Mat},\nu}$ is defined for $r \geq 0$ by :*

$$k_{\text{Mat},\nu}(r) = \sigma^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu} r}{l}\right)^{\nu} K_\nu \left(\frac{\sqrt{2\nu} r}{l}\right) \tag{2.7}$$

where K_ν is the modified Bessel function and Γ is the gamma function.
Similarly, for $x, x' \in \mathcal{X}$, one can define an anisotropic variant of the Matérn kernel:

$$k_{\text{Mat},\nu,\text{an}}(x, x') = \sigma^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \prod_{i=1}^{p} \left(\frac{\sqrt{2\nu}|x_i - x_i'|}{l_i}\right)^{\nu} K_\nu \left(\frac{\sqrt{2\nu}|x_i - x_i'|}{l_i}\right). \tag{2.8}$$

The Matérn functions are a broad class of covariance functions parametrized by ν which defines the regularity of the covariance kernel. When $\nu \longrightarrow +\infty$, the covariance function approaches the squared exponential.

$$\lim_{\nu \to +\infty} k_{\text{Mat},\nu}(r) = k_{\text{SE}}(r) \quad \forall r \in \mathbb{R}^+. \tag{2.9}$$

Proposition 2.1.2. *A Gaussian process with covariance function $k_{\text{Mat},\nu}$ or $k_{\text{Mat},\nu,\text{an}}$ is j times mean-square differentiable for $j < \nu$. The larger ν, the smoother the Gaussian process.*

Proposition 2.1.3. *The Matérn covariances can be expressed as a product of an exponential and a polynomial of order p for $\nu = p + 1/2$.*

$$k_{\text{Mat},p+1/2}(r) = \sigma^2 \exp\left(-\frac{\sqrt{2\nu}r}{l}\right) \frac{\Gamma(p+1)}{\Gamma(2p+1)} \sum_{i=0}^{p} \frac{(p+i)!}{i!(p-i)!} \left(\frac{\sqrt{8\nu}r}{l}\right)^{p-i}. \tag{2.10}$$

Because Matérn kernels can be simplified for half-integer, they are most used with half-integer values of ν.

This work mainly focused on Matérn and squared exponential covariances, though building well-suited covariance kernels is a vast field of research. The author refers to [RW+06] for more examples of covariance kernels.

To illustrate the regularity of the Matérn and Squared Exponential covariance kernels, Figure 2.1 displays sampled functions from Gaussian processes with zero-mean function and various covariance functions. This figure shows that a lower ν leads to

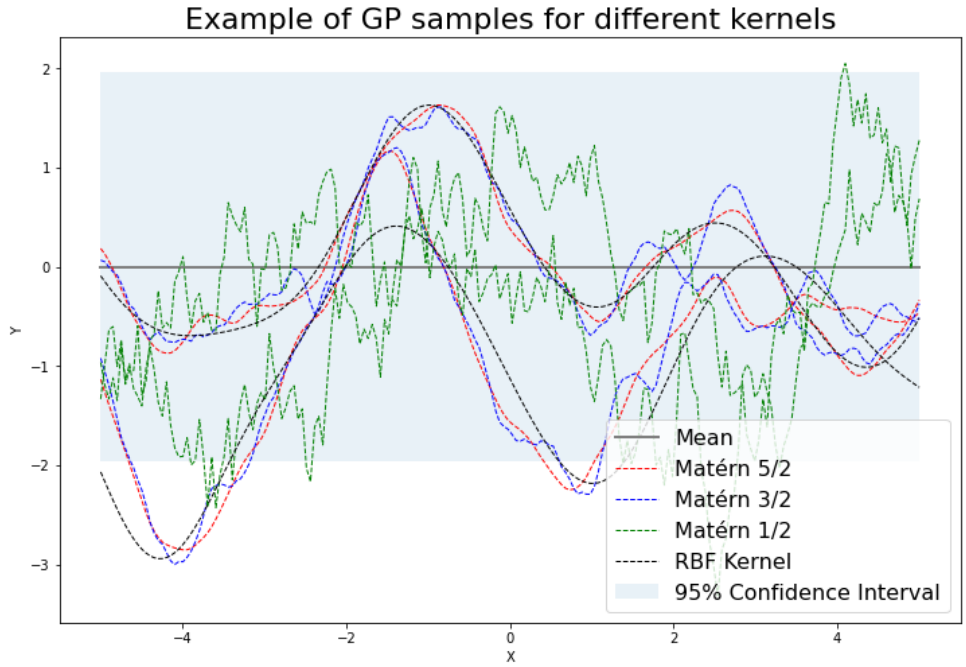

FIGURE 2.1: Random realizations of a Gaussian process prior with zero-mean and various covariance kernels.

sampled functions with more erratic behavior, while a larger ν produces smoother realizations.

2.1.1.3 Predictive distributions

Let us consider a zero-mean Gaussian process $f \sim \mathcal{GP}(0, k(x, x'))$ for some covariance function k and some training inputs $\mathbf{X} = (X_1, ..., X_n)^T \in \mathcal{X}^n$ and their associated outputs $\mathbf{f} = (f_1, ..., f_n)^T \in \mathbb{R}^n$, for $n \geq 1$. Similarly, let $\mathbf{X}_* \in \mathcal{X}^{n_*}$ be n_* test inputs for $n_* \geq 1$. Our task is to provide predictions for the yet-unknown outputs $\mathbf{f}_*$. We assume that they are modeled by the GP prior f such that $\mathbf{f}_* = f(\mathbf{X}_*)$. The results

presented hereafter extend to GP with a non-zero mean function, but we impose the zero-mean condition for simplicity's sake.

Assuming a Gaussian process model for the observations, the GP model provides a joint distribution for $f(\mathbf{X}, \mathbf{X}_*)$:

$$f(\mathbf{X}, \mathbf{X}_*) = \begin{pmatrix} f(\mathbf{X}) \\ f(\mathbf{X}_*) \end{pmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{pmatrix} K(\mathbf{X}, \mathbf{X}) & K(\mathbf{X}, \mathbf{X}_*) \\ K(\mathbf{X}_*, \mathbf{X}) & K(\mathbf{X}_*, \mathbf{X}_*) \end{pmatrix}\right). \tag{2.11}$$

To obtain a predictive distribution, we condition this joint distribution by the knowledge of $\mathbf{f}$ which are observations of the random variable $f(\mathbf{X})$. For that purpose, we make use of the Gaussian conditioning theorem.

Theorem 2.1.1. *Let X and Y be Gaussian vectors such that :*

$$\begin{pmatrix} X \\ Y \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \mu_X \\ \mu_Y \end{pmatrix} \begin{pmatrix} K_X & K_{XY} \\ K_{XY}^T & K_Y \end{pmatrix}\right). \tag{2.12}$$

Then the conditional law $Y|X$ has a Gaussian distribution given by :

$$Y|X \sim \mathcal{N}(\mu_C, K_C) \tag{2.13}$$

with $\mu_C = \mu_Y + K_{XY}^T K_X^{-1}(\mathbf{f} - \mu_X)$ and $K_C = K_Y - K_{XY}^T K_X^{-1} K_{XY}$.

Based on this theorem, the predictive distribution for the outputs $\mathbf{f}_*$ given $\mathbf{X}$, $\mathbf{X}_*$ and $\mathbf{f}$ is :

$$\begin{aligned} \mathbf{f}_*|\mathbf{f}, \mathbf{X}, \mathbf{X}_* &\sim \mathcal{N}(\mu_C, K_C) \\ \mu_C &= K(\mathbf{X}, \mathbf{X}_*)^T K(\mathbf{X}, \mathbf{X})^{-1} \mathbf{f} \\ K_C &= K(\mathbf{X}_*, \mathbf{X}_*) - K(\mathbf{X}, \mathbf{X}_*)^T K(\mathbf{X}, \mathbf{X})^{-1} K(\mathbf{X}, \mathbf{X}_*). \end{aligned} \tag{2.14}$$

This conditional distribution provides a way to predict the mean outputs from given input points $\mathbf{X}_*$ and the corresponding variances. The main interest of Gaussian process regression for our application is its ability to quantify the uncertainty of the predictions.

For most situations, the available outputs, denoted by $\mathbf{Y}$, often display noisy values such that $\mathbf{Y} = \mathbf{f} + \boldsymbol{\varepsilon}$ with $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathcal{I}_n)$. $\mathcal{I}_n$ refers to the identity matrix of size n. The noise is assumed Gaussian independent and identically distributed. Under the GP model, the covariance function for the outputs $\mathbf{Y}$ is now given by:

$$\mathrm{Cov}(Y_p, Y_q) = k(x_p, x_q) + \sigma^2 \delta_{p,q} \tag{2.15}$$

for $1 \leq p, q \leq n$, where δ is the Kronecker symbol. Thus equation (2.14) holds if $K(\mathbf{X}, \mathbf{X})$ is replaced by $K(\mathbf{X}, \mathbf{X}) + \sigma^2 \mathcal{I}_n$. The noise parameter σ^2 is known as the nugget noise and helps in regularizing the GP predictions and avoiding overfitting.

2.1.1.4 Hyperparameter selection

A Gaussian process has to be trained to provide reasonable predictions for regression or classification problems. The training phase involves optimizing the choice of the hyperparameters in the covariance kernel based on the training data.

The common practice for selecting the hyperparameters is to find the values that maximize the marginal likelihood $p(\mathbf{Y}|\mathbf{X})$. The marginal likelihood refers to the probability of the measures $\mathbf{Y}$, integrated over the latent functional space defined by

the GP model. It is defined by :

$$p(\mathbf{Y}|\mathbf{X}) = \int_{\mathbb{R}^n} p(\mathbf{Y}|\mathbf{f},\mathbf{X})p(\mathbf{f}|\mathbf{X})d\mathbf{f} \tag{2.16}$$

where $\mathbf{f}|\mathbf{X} \sim \mathcal{N}(\mathbf{0}, K(\mathbf{X},\mathbf{X}))$ are the latent function values whose distribution is given by the GP prior and $\mathbf{Y}|\mathbf{f} \sim \mathcal{N}(\mathbf{f}, \sigma^2\mathcal{I}_n)$, since we assume Gaussian likelihood. The log-marginal likelihood is thus given by :

$$\log p(\mathbf{Y}|\mathbf{X}) = -\frac{1}{2}\mathbf{Y}^T K_\sigma^{-1}\mathbf{Y} - \frac{1}{2}\log|K_\sigma| - \frac{n}{2}\log(2\pi) \tag{2.17}$$

with $K_\sigma = K(\mathbf{X},\mathbf{X}) + \sigma^2\mathcal{I}_n$, and $|K_\sigma|$ being the determinant of the matrix K_σ.

The log-marginal likelihood can be optimized with standard optimization algorithms. In this work, the limited memory Broyden–Fletcher–Goldfarb–Shanno algorithm for bound constraint (also known as L-BFGS-B) is used [Byr+95]. The optimization algorithm is restarted 10 times with different initial hyperparameter values. The optimal set of hyperparameters chosen is the one that provides the highest log-marginal likelihood of the 10 iterations. This approach reduces the risk of being stuck in a local optimum. Once the optimal set of hyperparameters is found, predictions can be made using equations (2.14).

Another commonly used approach for hyperparameter selection is the leave-one-out (LOO) cross-validation. The general idea is to train using the whole dataset except the training point i for $1 \leq i \leq n$. Similarly to the standard case, one can compute a log-marginal likelihood. We denote by $\mathbf{Y}_{-i} = (Y_1, ..., Y_{i-1}, Y_{i+1}, ..., Y_n)$ the set of all training outputs except Y_i and similarly we define $\mathbf{X}_{-i}$ as the set of all training inputs except X_i. Then the log-probability of the training output Y_i is known and summing over all i gives the LOO predictive error:

$$L_{\text{CV}}(\mathbf{X},\mathbf{Y}) = \sum_{i=1}^{N} \log p(Y_i|\mathbf{X}_{-i},\mathbf{Y}_{-i}). \tag{2.18}$$

This expression can be computed analytically and maximizing it with regard to the kernel hyperparameters yields a second criterion for hyperparameter selection. The computational burden for LOO-CV can be reduced, as most of the covariance matrices in the previous expression can be obtained from the covariance $K(\mathbf{X},\mathbf{X})$. Overall both the marginal likelihood and the LOO-CV have a similar computational budget. It has been argued that LOO-CV could be more robust in cases where the model is misspecified (see for example [Wah90]).

2.1.2 Multi-output Gaussian processes

In this section, we are extending the framework of GP surrogate models to higher output dimensions. This extension is not straightforward as we must keep a positive definite covariance structure for the global multi-output predictor. We present several methods to build such a multi-output GP (MOGP).

Once again, we consider some training inputs $\mathbf{X} = (X_1, ..., X_n)^T \in \mathbb{R}^{n\times p}$ where $X_i \in \mathcal{X} \subset \mathbb{R}^p$ for $1 \leq i \leq n$. The associated noisy training outputs are denoted by $\mathbf{Y} = (Y_1, ..., Y_n)^T \in \mathbb{R}^{n\times D}$, where $Y_i = (Y_{i,j})_{1\leq j\leq D} \in \mathbb{R}^D$ for $1 \leq i \leq n$ and $D \geq 2$. Our objective is to build a GP surrogate model f_s such that for any input $x \in \mathcal{X}$, the

surrogate provides a predictive distribution in the form of :

$$f_s(x) \sim \mathcal{N}\left(\overline{f_s(x)}, \mathbf{C}_s(x)\right) \tag{2.19}$$

where $\overline{f_s(x)} \in \mathbb{R}^D$ and $\mathbf{C}_s(x) \in S_D^+(\mathbb{R})$, with $S_D^+(\mathbb{R})$ is the set of real-valued positive-definite matrices of size D.

2.1.2.1 Naive approach

When building multi-output Gaussian process surrogate models, a simple procedure is to build one GP for each output channel. One can merge them to obtain a multi-output predictor yielding a diagonal predictive covariance, where the diagonal elements are the predictive variances of the individual scalar GPs.

Consider D independent scalar GP surrogate models $f_s^{(j)}$ for $1 \leq j \leq D$. The training data associated to the surrogate $f_s^{(j)}$ is $(\mathbf{X}, \mathbf{Y}_{\cdot,j})$ where $\mathbf{Y}_{\cdot,j} = (Y_{1,j}, ..., Y_{n,j})^T \in \mathbb{R}^n$. For any input $x \in \mathcal{X}$, the scalar GPs provide a predictive distribution:

$$f_s^{(j)}(x) \sim \mathcal{N}\left(\overline{f_s^{(j)}(x)}, \sigma_j^2(x)\right). \tag{2.20}$$

A multi-output GP surrogate model can then be built by aggregating the prediction of the individual scalar GPs such that:

$$\overline{f_s(x)} = \left(\overline{f_s^{(1)}(x)}, ..., \overline{f_s^{(D)}(x)}\right)^T \tag{2.21}$$

$$\mathbf{C}_s(x) = \text{diag}\left(\sigma_1^2(x), ..., \sigma_D^2(x)\right). \tag{2.22}$$

The main drawback of this method is clear: the correlations between the output channels are lost in the process.

2.1.2.2 Linear Model of Coregionalization

One possible method to account for correlations across outputs is to create independent scalar GPs and mix them afterward with a transition matrix. This is the premise of the Linear Model of Coregionalization (LMC) [BCW07]. This method allows correlations between the output channels while maintaining a positive definite covariance structure. A brief description of this approach is presented in the next paragraphs.

Let us consider $Q \geq 1$ independent scalar GPs $(u_q)_{1 \leq q \leq Q}$. For simplicity's sake, only zero-mean processes are considered:

$$u_q \sim \mathcal{GP}\left(0, k_q(x, x')\right). \tag{2.23}$$

These are called latent Gaussian processes. Now let us consider a real mixing matrix $W \in \mathbb{R}^{D \times Q}$, whose elements are denoted by w_{jq} for $1 \leq q \leq D$ and $1 \leq q \leq Q$. Let $f_s^{(i)}$ be a scalar GP corresponding to the channel i for $1 \leq j \leq D$, which we define for $x \in \mathcal{X}$ by the relation:

$$f_s^{(j)}(x) = \sum_{q=1}^{Q} w_{jq} u_q(x). \tag{2.24}$$

With this definition, the flattened output vector follows a multivariate normal distribution.

$$\begin{pmatrix} f_s^{(1)}(X_1) \\ \vdots \\ f_s^{(1)}(X_n) \\ \vdots \\ f_s^{(D)}(X_1) \\ \vdots \\ f_s^{(D)}(X_n) \end{pmatrix} \sim \mathcal{N}\left(\mathbf{0}, \sum_{q=1}^{Q} \begin{pmatrix} w_{1q}w_{1q}K_q(\mathbf{X}) & \dots & w_{1q}w_{Dq}K_q(\mathbf{X}) \\ \vdots & \ddots & \vdots \\ w_{Dq}w_{1q}K_q(\mathbf{X}) & \dots & w_{Dq}w_{Dq}K_q(\mathbf{X}) \end{pmatrix}\right)$$

where $K_q(\mathbf{X}) = \left(k_q(X_i, X_j)\right)_{1\leq i,j\leq n}$ is the covariance matrix of the inputs $\mathbf{X} = (X_i)_{1\leq i\leq n}$ for the covariance kernel k_q. We may also write the full covariance matrix using the Kronecker product $\otimes$.

Definition 2.1.9 (Kronecker product)**.** *Let $A =\in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{p\times q}$ two real matrices. The Kronecker product is the block matrix $C = A \otimes B \in \mathbb{R}^{pm\times qn}$ defined by :*

$$C = A \otimes B = \begin{pmatrix} a_{11}B & \dots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \dots & a_{mn}B \end{pmatrix} \tag{2.25}$$

where the a_{ij} are the elements of A for $1 \leq i \leq m$ and $1 \leq j \leq n$.

With this notation, the full covariance matrix can be written as:

$$K_{\text{LMC}}(\mathbf{X}, \mathbf{X}) = \sum_{q=1}^{Q} W_{\cdot,q}W_{\cdot,q}^T \otimes K_q(\mathbf{X}) \tag{2.26}$$

where $W_{\cdot,q}$ is the column vector formed by the q-th column of W.
From here, the standard methods from GP regression apply. In particular, the mixing matrix W is considered a hyperparameter and is optimized by maximizing the log-marginal likelihood. Its diagonal elements are set to one since they would be redundant with the variance hyperparameters in k_q.

With this method, it is possible to effectively build a multi-output Gaussian process that accounts for correlations between the outputs. In the next section, we investigate another approach based on convolved Gaussian processes.

2.1.2.3 Convolved GPs

Convolved Gaussian Processes [AL11] are built similarly to LMC models. For $x \in \mathcal{X}$, the output for the channel $1 \leq j \leq D$ is built by the following relation:

$$f_s^{(j)}(x) = \sum_{q=1}^{Q} \int G_{j,q}(x - z)u_q(z)dz. \tag{2.27}$$

The functions $G_{j,q}$ represent filters in the convolution kernel. Let $G_{j,q}$ is a Gaussian filter with covariance P_j^{-1}, a positive definite matrix. Then the density $G_{j,q}(x)$ is the

density of a Gaussian random variable $\mathcal{N}(0, P_j^{-1})$ calculated at x with a multiplicative constant $S_{j,q}$. Let k_q be the squared exponential covariance function described in Section 2.1.1.2, with variance V_q and inverse correlation lengths given by the diagonal matrix Λ_q such that for $x, x' \in \mathcal{X}$:

$$k_q(x, x') = V_q \exp\left(-\frac{1}{2}(x - x')^T \Lambda_q (x - x')\right). \tag{2.28}$$

The convolution product in (2.27) becomes tractable. Given some inputs $x, x' \in \mathcal{X}$ and $1 \leq j, k \leq D$, the covariance between the channel outputs j and k is given by:

$$\mathrm{Cov}(f_s^{(j)}(x), f_s^{(k)}(x')) = \sum_{q=1}^{Q} \frac{S_{j,q} S_{k,q} V_q}{(2\pi)^{p/2} |C_{q,j,k}|^{1/2}} \exp\left(-\frac{1}{2}(x - x')^T C_{q,j,k}^{-1} (x - x')\right) \tag{2.29}$$

with $C_{q,j,k} = P_j^{-1} + P_k^{-1} + \Lambda_q^{-1}$. One may write the full covariance matrix between two inputs $x, x' \in \mathcal{X}$ as:

$$K_{\mathrm{Conv}}(x, x') = \sum_{q=1}^{Q} V_q \begin{pmatrix} S_{1,q} S_{1,q} K_{1,1,q}(x, x') & \dots & S_{D,q} S_{1,q} K_{1,D,q}(x, x') \\ \vdots & \ddots & \vdots \\ S_{D,q} S_{1,q} K_{D,1,q}(x, x') & \dots & S_{D,q} S_{D,q} K_{D,D,q}(x, x') \end{pmatrix} \tag{2.30}$$

where $K_{j,k,q}$ is given by :

$$K_{j,k,q}(x, x') = (2\pi)^{-p/2} |C_{q,j,k}|^{-1/2} \exp\left(-\frac{1}{2}(x - x')^T C_{q,j,k}^{-1} (x - x')\right). \tag{2.31}$$

If the matrices P_j are chosen diagonal, $K_{j,k,q}$ can be understood as an equivalent RBF kernel with correlation lengths given by the matrix $C_{q,j,k}$.
This covariance can also be seen as a sum of Kronecker products. Let us define $\mathbf{S}_q = (S_{j,q})_{1 \leq j \leq D} \in \mathbb{R}^D$ and $\mathbf{A}_q = \mathbf{S}_q \mathbf{S}_q^T \in \mathbb{R}^{D \times D}$. For $1 \leq q \leq Q$ and $1 \leq j, k \leq D$, let $K_{j,k,q}(\mathbf{X})$ be the matrix whose elements are $K_{j,k,q}(X_r, X_s)$ as given in (2.31), for $1 \leq r, s \leq n$. Then the covariance matrix over the whole training set is given by:

$$K_{\mathrm{Conv}}(\mathbf{X}) = \sum_{q=1}^{Q} V_q \begin{pmatrix} S_{1,q} S_{1,q} K_{1,1,q}(\mathbf{X}) & \dots & S_{D,q} S_{1,q} K_{1,D,q}(\mathbf{X}) \\ \vdots & \ddots & \vdots \\ S_{D,q} S_{1,q} K_{D,1,q}(\mathbf{X}) & \dots & S_{D,q} S_{D,q} K_{D,D,q}(\mathbf{X}) \end{pmatrix} \tag{2.32}$$

On top of this, a nugget noise may be added to account for noise in the measurements.

For convolved Gaussian processes, the number of hyperparameters to optimize is much larger than for independent processes since the coefficients of the matrices P_j, Λ_q and the scalar $S_{j,q}$ have to be learned. The Gaussian filters are chosen with diagonal covariance matrices P_j to ease the training process. This drastically reduces the number of hyperparameters, and the Gaussian processes remain flexible enough to account for correlations and to provide mean and covariance predictions.

2.1.3 Sparse approximations

Gaussian process surrogate models provide a flexible and easily implemented framework for non-linear regression, all the while having a native uncertainty quantification method. However, they suffer from a large computational complexity for large datasets due to the $\mathcal{O}(n^3)$ complexity of the inverse covariance computation. In a multi-output framework with D dimensions, this complexity even increases to $\mathcal{O}(D^3n^3)$. To deal with this issue, numerous sparse approximations have been considered. Hereafter, we introduce the method developed in [Tit09]. In this section, we focus on scalar GPs to alleviate the notation, but the framework is easily extended to multi-output cases.

For the most part, sparse methods aim at approximating the covariance matrix by a lower rank approximation, which reduces the cost of the inversion. The approximations are based on inducing points which serve as intermediate points between training and test points. In sparse GP methods, the predictive distribution is obtained by conditioning on the inducing points instead of the true training points. One can think of inducing points as a way to summarize the information brought by the training data. The development of accelerated GP inference with sparse methods is not new and various approaches have been introduced throughout the years, such as [SB00; SG05; SWL03]. The general principle however remains the same and is described hereafter.

Using the notations from Section 2.1.1, we know the predictive distribution $p(\mathbf{f}_*|\mathbf{X}, \mathbf{f}, \mathbf{X}_*)$ is obtained by conditioning the joint distribution $p(\mathbf{f}_*, \mathbf{f})$ by $(\mathbf{X}, \mathbf{f})$. Similarly, one can introduce $m \geq 1$ latent variables $(\mathbf{X}_m, \mathbf{f}_m)$ and obtain a predictive distribution by marginalizing out the latent variables. To alleviate the notations, we remove $\mathbf{X}, \mathbf{X}_*$ in the notation of the conditional distributions.

$$p(\mathbf{f}_*|\mathbf{Y}) = \int p(\mathbf{f}_*|\mathbf{f}_m, \mathbf{f}) p(\mathbf{f}|\mathbf{f}_m, \mathbf{y}) p(\mathbf{f}_m|\mathbf{y}) d\mathbf{f} d\mathbf{f}_m. \tag{2.33}$$

Generally, sparse GP approximations are composed of two steps. First of all, we make the assumption that $\mathbf{f}_*$ and $\mathbf{f}$ are independent given $\mathbf{f}_m$ which implies that $p(\mathbf{f}_*|\mathbf{f}_m, \mathbf{f}) \simeq p(\mathbf{f}_*|\mathbf{f}_m)$. All the information between training and predictive distribution transitions by the inducing points.

Then, we approximate the distribution $p(\mathbf{f}_m|\mathbf{y})$ by a variational distribution $\phi(\mathbf{f}_m)$. The predictive distribution thus reduces to:

$$p(\mathbf{f}_*|\mathbf{Y}) \simeq q(\mathbf{f}_*) = \int p(\mathbf{f}_*|\mathbf{f}_m) \phi(\mathbf{f}_m) d\mathbf{f}_m \tag{2.34}$$

where $\phi(\mathbf{f}_m) \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$.
Introducing the notations $\mathbf{K}_{m,m} = K(\mathbf{X}_m)$, $\mathbf{K}_{*,*} = K(\mathbf{X}_*)$ and $\mathbf{K}_{*,m} = \mathbf{K}_{m,*}^T = K(\mathbf{X}_*, \mathbf{X}_m)$ one can write the variational posterior as:

$$q(\mathbf{f}_*) \sim \mathcal{N}\left(\mathbf{K}_{*,m}\mathbf{K}_{m,m}^{-1}\boldsymbol{\mu}, \mathbf{K}_{*,*} - \mathbf{K}_{*,m}\mathbf{K}_{m,m}^{-1}\mathbf{K}_{m,*} + \mathbf{K}_{*,m}\mathbf{K}_{m,m}^{-1}\boldsymbol{\Sigma}\mathbf{K}_{m,m}^{-1}\mathbf{K}_{m,*}\right). \tag{2.35}$$

We also introduce $\mathbf{K}_{n,n} = K(\mathbf{X})$, $\mathbf{K}_{m,n} = \mathbf{K}_{n,m}^T = K(\mathbf{X}_m, \mathbf{X})$.

One key factor for the performance of sparse GPs is the location of the inducing points. One may select them as a subset of the training data, though this method is sub-optimal. In [Tit09], the inducing points optimal positions are learned, jointly with the hyperparameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, by minimizing a lower bound on the Kullback-Leibler

(KL) divergence [KL51] between the augmented variational posterior $q(\mathbf{f}, \mathbf{f}_m)$ and the true augmented posterior $p(\mathbf{f}, \mathbf{f}_m|\mathbf{Y})$. The KL divergence is a standard dissimilarity measure between two probability distributions. It is defined for two probability distributions with density p and q by:

$$\mathrm{KL}(p||q) = \int_{\mathcal{X}} p(x) \log\left(\frac{p(x)}{q(x)}\right) dx \tag{2.36}$$

Using the relation $p(\mathbf{f}, \mathbf{f}_m|\mathbf{Y})p(\mathbf{Y}) = p(\mathbf{f}, \mathbf{f}_m, \mathbf{Y})$ we have:

$$\begin{aligned}\mathrm{KL}\left(q(\mathbf{f}, \mathbf{f}_m)\|p(\mathbf{f}, \mathbf{f}_m|\mathbf{Y})\right) &= \int q(\mathbf{f}, \mathbf{f}_m) \log\left(\frac{q(\mathbf{f}, \mathbf{f}_m)p(\mathbf{Y})}{p(\mathbf{f}_*, \mathbf{f}_m, \mathbf{Y})}\right) d\mathbf{f}d\mathbf{f}_m \\ &= \log p(\mathbf{Y}) - \int q(\mathbf{f}, \mathbf{f}_m) \log\left(\frac{p(\mathbf{f}_*, \mathbf{f}_m, \mathbf{Y})}{q(\mathbf{f}, \mathbf{f}_m)}\right) d\mathbf{f}d\mathbf{f}_m. \end{aligned} \tag{2.37}$$

The second term on the right is the Evidence Lower Bound (ELBO), denoted by $F_V(\mathbf{X}_m, \phi, \sigma^2)$, where σ^2 is the variance in the Gaussian likelihood $p(\mathbf{Y}|\mathbf{f})$. Minimizing the KL w.r.t. ϕ (that is w.r.t. μ and Σ) and $\mathbf{X}_m$ is equivalent to maximizing the ELBO. Using the relation $q(\mathbf{f}, \mathbf{f}_m) = \phi(\mathbf{f}_m)p(\mathbf{f}|\mathbf{f}_m)$, it can be written as:

$$\begin{aligned}F_V(\mathbf{X}_m, \phi, \sigma^2) &= \int p(\mathbf{f}|\mathbf{f}_m)\phi(\mathbf{f}_m) \log\left(\frac{p(\mathbf{Y}|\mathbf{f})p(\mathbf{f}_m)}{\phi(\mathbf{f}_m)}\right) d\mathbf{f}d\mathbf{f}_m \\ &= \int \phi(\mathbf{f}_m) \log\left(\frac{p(\mathbf{f}_m)\Phi(\mathbf{Y}, \mathbf{f}_m)}{\phi(\mathbf{f}_m)}\right) d\mathbf{f}_m \end{aligned} \tag{2.38}$$

where $\Phi(\mathbf{Y}, \mathbf{f}_m) = \exp\left(\int p(\mathbf{f}|\mathbf{f}_m) \log p(\mathbf{Y}|\mathbf{f}) d\mathbf{f}\right)$. We used the fact that $p(\mathbf{Y}|\mathbf{f}, \mathbf{f}_m) = p(\mathbf{Y}|\mathbf{f})$. Finally, one can notice that:

$$F_V(\mathbf{X}_m, \phi, \sigma^2) = \log \mathcal{Z} - \mathrm{KL}\left(\phi(\mathbf{f}_m)\|\phi'(\mathbf{f}_m)\right) \tag{2.39}$$

where ϕ' is the probability distribution whose density is given by $\phi'(\mathbf{f}_m) = \frac{p(\mathbf{f}_m)\Phi(\mathbf{y}, \mathbf{f}_m)}{\mathcal{Z}}$ and $\mathcal{Z} = \int p(\mathbf{f}_m)\Phi(\mathbf{y}, \mathbf{f}_m)d\mathbf{f}_m$ is the normalization constant.
With a Gaussian likelihood with variance noise σ^2, maximizing the ELBO yields $\phi(\mathbf{f}_m) = \phi'(\mathbf{f}_m)$ and:

$$F_V(\mathbf{X}_m, \phi', \sigma^2) = \log \mathcal{N}\left(\mathbf{Y}|\mathbf{0}, \sigma^2\mathcal{I}_n + \mathbf{K}_{n,m}\mathbf{K}_{m,m}^{-1}\mathbf{K}_{m,n}\right) - \frac{1}{2\sigma^2}\mathrm{Tr}\left(\mathbf{K}_{n,n} - \mathbf{K}_{n,m}\mathbf{K}_{m,m}^{-1}\mathbf{K}_{m,n}\right) \tag{2.40}$$

where the notation $\mathcal{N}(x|\mu, \Sigma)$ should be understood as the density at x of the distribution $\mathcal{N}(\mu, \Sigma)$.
As a final step, this expression can be maximized w.r.t. σ^2 and $\mathbf{X}_m$ to obtain the optimal nugget noise variance σ^2 and inducing points location $\mathbf{X}_m$. This sparse approximation, commonly known as the sparse variational Gaussian process (SVGP) method, has been extensively used since its first introduction. Yet, the optimal location of inducing points is still a vivid field of research (see for example [UCJ21; Wu+21]). In this thesis, sparse approximations are exclusively centered on SVGP. In cases where the computational budget is sufficient, exact GP regression is used. However, we will see that sparse GP may become necessary for some of the applications developed in this thesis.

2.2 Why use surrogate models?

In this section, we illustrate why surrogate models are needed for reliable uncertainty quantification in inverse problem resolution. Let us consider a real-world problem of a metallic plutonium sphere with an unknown composition. We would like to know what is inside the material. The plutonium sphere is modeled with the Monte Carlo code MCNP6 (see Section 1.4.8) to represent the BERP ball, which has been used in numerous studies in neutron multiplicity counting. The simplified description of the BERP ball is given in Section 2.5.2, and for a fully detailed description of the geometry and material composition, we refer to Appendix A.

Let us consider some observations $\mathbf{y} = (y^{(k)})_{1 \leq k \leq N}$ obtained from MCNP simulations, with sequential binning estimators. We assume these observations are noisy with a zero-mean Gaussian noise with covariance $\mathbf{C}_{\text{obs}}$. The noise covariance can be estimated using higher-order moments of the detection statistics or with a simple empirical covariance estimator. More details are given in Section 2.3.2.2.
We model these observations with a simplified version of the point model f_p, given by equations (1.99), (1.100) and (1.101) and setting $x_s = 1$ since we are considering a spontaneous fission source in this case. We denote the simplified point model by:

$$\begin{aligned} f_p \colon \mathcal{X} &\longrightarrow \mathbb{R}^3 \\ (k_p, \varepsilon_F, S) &\longmapsto (R, Y_\infty, X_\infty) \end{aligned}$$

with $\mathcal{X} \subset \mathbb{R}^3$. The direct model provides the link between inputs and observations such that for $1 \leq k \leq N$:

$$y^{(k)} = f_p(x_{\text{true}}) + \varepsilon^{(k)} \text{ with } \varepsilon^{(k)} \sim \mathcal{N}\left(0, \mathbf{C}_{\text{obs}}\right). \tag{2.41}$$

We have a total of $N = 20$ observations available. As an example, five observations are given in Table 2.1. We wish to solve the inverse problem by a Bayesian approach.

TABLE 2.1: Count rate and Feynman moment observations for the copper-reflected BERP ball.

R	Y_∞	X_∞
7319.7	0.389	0.424
7328.3	0.386	0.389
7308.4	0.395	0.410
7300.0	0.382	0.452
7381.8	0.418	0.514

The likelihood is Gaussian and analytical, and the prior is assumed to be uniform on $\mathcal{X} = [0.7, 0.9] \times [0.001, 0.05] \times [1 \times 10^5, 2 \times 10^5]$. The posterior distribution is sampled by Metropolis-Hastings (with a Gaussian proposal with adaptive scaling factor) with $L = 2 \times 10^6$ samples. The autocorrelation and posterior distribution marginals are displayed respectively in Figures 2.2 and 2.3.

These results highlight the main difficulty at stake in this inverse problem. One can see that the true value lies outside of the support of the posterior distribution. The posterior distribution that is obtained is not reliable. The reason is that we assumed the observations followed the direct model f_p. Since this model is only an approximation, there may be a systematic bias between f_p and the true direct model. This bias is not considered in this approach.

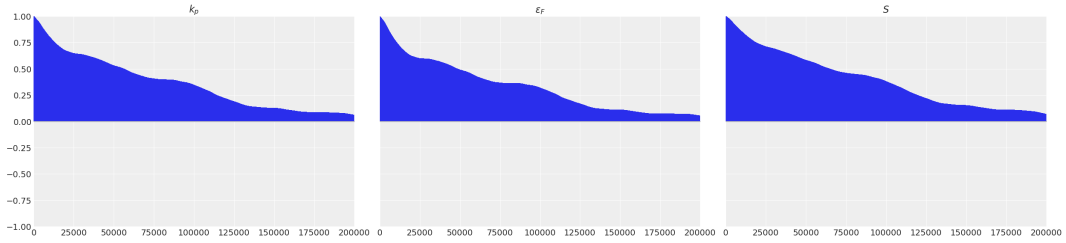

FIGURE 2.2: Autocorrelations of the MCMC samples.

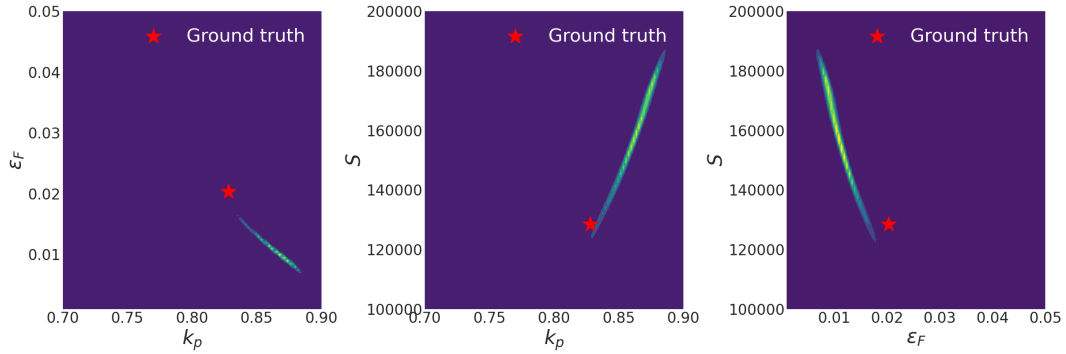

FIGURE 2.3: 2D marginals of the posterior distribution with the direct model f_p.

We should also notice the peculiar shape of the posterior distribution here. The support of the distribution is mainly in a one-dimensional manifold in the three-dimensional space. As a result, the MCMC sampling is made more difficult since most candidate points may be rejected if they are outside the support. This is highlighted by the autocorrelation plot in Figure 2.2. One can see that the decorrelation time $\hat{\tau}$ is significant with Metropolis-Hastings, with $\hat{\tau} \simeq 2 \times 10^5$, meaning that it takes around 2×10^5 MCMC iterations to decorrelate two samples. The effective sample size is thus $L_{\text{eff}} \simeq 10$ in that case which is not sufficient. This is why Metropolis-Hastings is not well-suited for the exploration and sampling of such posterior distributions, and other methods will be explored (see sections 2.5.3.1 and 2.5.3.2 for example). Keep in mind that our goal is to provide posterior distributions that quantify uncertainties from both the model and the observations. Therefore, we may expect the distributions to be broader, and thus the sampling could be somewhat easier in our next applications. However, this could very well be counterbalanced by the higher dimension of our input space $\mathcal{X}$ in the next applications (see Section 2.3.1).

We now move on to the data collection process, which is the first mandatory step for building the needed surrogate models.

2.3 Collecting the data

In this section, the method used to build datasets for neutron correlations in zero-power systems is described in detail. In this work, two neutron datasets were used. The first is based on simulations of metallic plutonium spherical objects and the second is built upon simulation results from cylindrical objects with liquid uranyl nitrate.

In the first paragraphs, let us describe the inputs used to extend the point model framework and how they can be obtained in a neutronic Monte Carlo simulation with MCNP6. Then we focus on the estimation of the Feynman moments and how one can leverage the supplementary information of numerical simulations on neutron histories to largely reduce the estimation noise. Finally, we provide a brief description of the two datasets generated.

2.3.1 Extended feature space

First, we wish to generate a training dataset to build an efficient and reliable surrogate model. The point model (see equations (1.99), (1.100) and (1.101)) provides us with an approximation for the true direct model. It seems natural to consider the same inputs and outputs. However, since the point model is not very accurate and can fail to provide a reliable uncertainty quantification (see Figure 2.3) we choose to extend the feature space $\mathcal{X}$ to a higher-dimensional space by increasing the number of features considered. The strongest assumptions of the point model are that the neutrons are monoenergetic, the medium is infinite, and there are only fission and capture reactions. Intuitively, we would like to add new features to reduce the influence of these assumptions.

Firstly let us introduce the feature Φ defined as the ratio of thermal over fast neutron flux. We consider a two-energy group setting with a limit energy $E_{\text{lim}} = 1$ keV. Then, to account for other nuclear reactions we introduce a parasitic absorption efficiency ε_A which is defined similarly to the Feynman efficiency ε_F as the average number of parasitic absorptions over the average number of induced fissions. This quantity may be related to a leakage efficiency ε_L in a setting where parasitic absorptions are considered. Finally, to account for the finiteness of the medium we introduce a neutronic albedo J which is defined as the ratio of outward and inward neutron currents at the outermost layer of the geometry. This parameter quantifies the reflective power of the outer geometry bounds.

Overall, the dataset contains a total of seven features:

- k_p is the prompt multiplication factor.
- ε_F is the Feynman efficiency defined as the number of neutron detections over the number of induced fissions.
- S is the source intensity in events per second.
- x_s is the source type and is defined as the ratio of source neutrons produced per spontaneous fissions over the total number of source neutrons.
- Φ is the ratio of thermal over fast neutron flux.
- ε_A is the parasitic efficiency, which is the ratio of parasitic absorptions over induced fissions.
- J is the neutronic albedo which is the ratio of inward over outward neutron current in the outer layer of the geometry.

In the next paragraphs, we describe briefly how these quantities are obtained in the Monte Carlo simulations in MCNP6.

2.3.1.1 Eigenvalue calculation

The prompt multiplication factor k_p is obtained via an eigenvalue calculation in MCNP6. The eigenvalue problem is obtained by seeking a stationary solution to the neutron transport equation. This is done by introducing a factor k_p to reach an equilibrium between the neutron production and destruction terms. The general eigenvalue problem is:

$$\begin{aligned}\vec{\Omega}\cdot\vec{\nabla}\psi(\vec{r},E,\vec{\Omega})+\sum_k N_k(\vec{r})\sigma_k(E)\psi(\vec{r},E,\vec{\Omega})\\ =\frac{1}{k_p}\frac{1}{4\pi}\sum_k N_k(\vec{r})\int_0^{+\infty}dE'\nu_{t,k}(E')\sigma_{f,k}(E')\chi_{p,k}(E'\to E)\phi(\vec{r},E')\\ +\sum_k N_k(\vec{r})\int_0^{+\infty}dE'\int_{4\pi}d\vec{\Omega}'\sigma_{s,k}(E'\to E,\vec{\Omega}'\to\vec{\Omega})\psi(\vec{r},E,\vec{\Omega})\end{aligned} \tag{2.42}$$

where $\vec{\Omega}$ is the solid angle, $\vec{r}\in\mathbb{R}^3$ is the position vector, E is the energy, $\psi(\vec{r},E,\vec{\Omega})$ is the angular neutron flux, $N_k(\vec{r})$ is the concentration of the k-th nuclide at position $\vec{r}$, $\nu_{t,k}(E)$ is the average number of emitted neutrons after a fission on a nuclide k with an incident neutron energy E, $\sigma_{f,k}(E)$ is the microscopic fission cross section for nuclide k and energy E, $\chi_{p,k}(E'\to E)$ is the energy spectrum of neutrons emitted per fission on a nuclide k with incident neutron energy E', $\Phi(\vec{r},E)$ is the scalar neutron flux defined as the angular flux integrated over the solid angle and $\sigma_{s,k}(E'\to E,\vec{\Omega}'\to\vec{\Omega})$ is the microscopic cross section for a scattering reaction changing the neutron energy and solid angle from E and $\vec{\Omega}$ to E' and $\vec{\Omega}'$.
This eigenvalue problem is solved by a power iteration method [DH76]. The general idea is to sample a fixed number n_b of Monte Carlo batches. Starting from an arbitrary initial fission neutron distribution $Q^{(n)}$ at batch n, the Monte Carlo simulation provides a new neutron distribution $Q^{(n+1)}$ at the end of the batch. This is iterated until convergence and the multiplication factor is then found. The iterative updates of the eigenvalue are given by:

$$k^{(n+1)}=k^{(n)}\times\frac{\int Q^{(n+1)}(\vec{r})d\vec{r}}{\int Q^{(n)}(\vec{r})d\vec{r}}. \tag{2.43}$$

The common practice is to start with a few inactive cycles in which the iterations are only used to compute a better initial fission neutron distribution $Q^{(0)}$ without recording the evolution of the eigenvalue.
The eigenvalues calculations performed in this work are done with 20 inactive cycles and 200 active cycles of 10^5 neutrons each. MCNP6 provides the 95% confidence interval for the estimated k_p. The width of the 95% confidence interval may vary depending on the calculations, but on average for the calculations considered, the confidence interval has a width of roughly 100 pcm (where 1 pcm $= 10^{-5}$, the denomination pcm refers to "per cent mille").

2.3.1.2 Tally estimations for dataset features

Most of the other features are estimated through tally estimations in MCNP6. The general principle is simple. When an event occurs in the region of interest (either in the space, angle, or energy domain), the event is recorded and given a value from a tally estimator. The choice of the tally estimator may vary depending on the medium, or the reaction considered as some are more suited than others depending

on the case. Overall, this produces a score for each event which is then averaged over the whole Monte Carlo simulation to yield a mean quantity with its associated confidence interval. Tally estimations are one of the key aspects of neutronic Monte Carlo codes and are used to estimate neutron fluxes and currents, reaction rates, and other quantities. In this work, they are used to provide estimates for ε_F, ε_A, Φ, and J.

2.3.1.3 Creating the MCNP input files

To produce a dataset, we start with a reference test case, either an experimental configuration of the SILENE reactor facility or a subcritical experiment conducted on a metallic plutonium sphere. These two reference cases are described in Section 2.5.2. Then, the compositions, geometries, and densities of the materials are changed randomly to produce a new input file. Of course, we make sure to remain in subcritical cases. This means all dataset instances are obtained from randomly generated problems while remaining similar to the test case of interest.

The parasitic absorptions are considered to be occurring only in the fissile region for each case. The thermal over fast neutron flux Φ is measured outside the geometry far from the object. This quantity provides information on the moderating power of the unknown object. In practical experiments, it may be accessed by non-intrusive neutron flux measurements.
We also added an outer layer of concrete modeling the room reflection, for which the albedo J provides information.
The source type x_s is a direct input in the MCNP files, used when defining the neutron source. The spontaneous fission source is a volumic source uniformly spread in the fissile region, while the (α, n) source is modeled as a point source.
To summarize, we provide an example of the MCNP input for the dataset instances in Appendix C. These files are generated from a Python script that randomly samples the various compositions, densities, and geometries of the MCNP simulation, as well as the source term.
Thorough descriptions of the reference cases are found in the Appendix A.

2.3.1.4 Post-treatment of the PTRAC files

The careful reader may have noticed that we did not mention the source intensity S in the previous section. Indeed, in MCNP calculations the source intensity is not provided as for a single calculation only the number of neutrons to be sampled is given by the user. The source intensity is introduced in a post-treatment step.

The MCNP simulations return a PTRAC file, also known as a *time list file*, in which all the neutron detection instants and neutron history numbers are recorded. These PTRAC files can then be used to obtain the Feynman moments (and the count rates) using the sequential binning estimators (1.120) and (1.120), or the triggered binning estimators (1.124) and (1.125). However, MCNP does not have a notion of a trans-history timescale. Each neutron history has its own timescale starting at the neutron birth instant with the source event. To mimic a real-life experiment, it is required to introduce the global timescale, by sampling the source events.
As mentioned in 1.4.3, radioactive sources are modeled as Poisson processes. For a source intensity S, let us then consider a compound Poisson process $(S_t)_{t\geq 0} \sim \mathcal{P}(S, f^{(C)})$. As described in Section 1.3.2, the intervals between the jumps of a compound Poisson process with intensity S are given by exponential distributions. Thus

the k-th source event occurs at time $t_k = \sum_{i=0}^{k-1}(t_{i+1} - t_i)$ with $t_0 = 0$ and where $t_{i+1} - t_i \sim \mathcal{E}(S)$ where $\mathcal{E}(S)$ refers to the exponential distribution with intensity S. The source instants are thus sampled randomly from the distribution $\mathcal{E}(S)$.

2.3.1.5 Sensitivity analysis

Let us conclude this study with a qualitative analysis of the influence of each feature on the predicted outputs (R, Y_∞, X_∞) using covariance-based sensitivity analysis indices. Since this analysis is marginal in the grand scheme of this thesis, we are not discussing sensitivity analysis theory in this section. For a comprehensive introduction, the reader may refer to [Sud08; DV+21].

Consider a scalar function f of a random variable $x = (x_1, ..., x_p) \in \mathcal{X}$ for $p \geq 1$. The model response can be decomposed into a sum of contributions such as:

$$f(x) = f_0 + \sum_{i=1}^{p} f_i(x_i) + \sum_{1 \leq i,j \leq p} f_{i,j}(x_i, x_j) + ... = \sum_{A \in \mathcal{P}_p} f_A(x_A) \tag{2.44}$$

where $\mathcal{P}_p$ is the set of subsets of $\{1, ..., p\}$ and x_A is the subvector of x where only the components in A are kept.
This result is known as the Hoeffding decomposition. In the case of independent inputs, the terms of the decomposition are orthogonal. The sensitivity analysis of the model response $f(x)$ can then be performed using an ANOVA (analysis of variance) decomposition of $f(x)$ such that:

$$\text{Var}[f(x)] = \sum_{i=1}^{p} V_i + \sum_{1 \leq i,j \leq p} V_{i,j} + ... = \sum_{A \in \mathcal{P}_p} V_A \tag{2.45}$$

where $V_A = \text{Var}\left[f_A(x_A)\right] = \sum_{B \subset A} (-1)^{|A|-|B|} \text{Var}\,\mathbb{E}\left[f(x|x_B)\right]$. From this decomposition, one can define the well-known Sobol indices [Sob93]. For $A \in \mathcal{P}_p$, the Sobol index S_A associated to the subset A is defined by:

$$S_A = \frac{V_A}{\text{Var}\left[f(x)\right]}. \tag{2.46}$$

Common quantities of interest, in that case, are the first-order indices $S_{\{j\}}$ which quantify the contribution of the input j by itself, and the total-order indices $S_{\{j\}}^{(\text{tot})}$ defined by:

$$S_{\{j\}}^{(\text{tot})} = 1 - \sum_{A \subset \overline{\{j\}}} S_A \tag{2.47}$$

where $\overline{\{j\}}$ is the complementary of $\{j\}$. The total indices quantify the contribution of input j in association with all the other inputs. However, in this particular scenario, we are dealing with dependent inputs as seen in Figure 2.5. The ANOVA decomposition is only valid for independent inputs, in which case the elements of the Hoeffding decomposition are orthogonal. For dependent inputs, however, it is possible to introduce covariance-based indices linked to an ANCOVA (analysis of covariance) decomposition:

$$\text{Var}[f(x)] = \sum_{A \in \mathcal{P}_p} \text{Cov}\left[f_A(x_A), f(x)\right] \tag{2.48}$$

where $\text{Cov}\left[f_A(x_A), f(x)\right] = V_A + \sum_{B \neq A} \text{Cov}\left[f_A(x_A), f_B(x_B)\right]$.

In [HG18], the authors introduce two indices $S_A^{(C)}$ and $S_A^{(U)}$ as:

$$S_A^{(C)} = \frac{\sum_{B \neq A} \text{Cov}\left[f_A(x_A), f_B(x_B)\right]}{\text{Var}(f(x))} \tag{2.49}$$

$$S_A^{(U)} = \frac{\text{Var}\left[f_A(x_A)\right]}{\text{Var}\left[f(x)\right]}. \tag{2.50}$$

$S_A^{(C)}$ quantifies the dependence w.r.t. the subset A induced by the correlation between the inputs of A and other inputs. On the other hand, $S_A^{(U)}$ quantifies the contribution of the input subset A by itself, which may be referred to as the structural contribution.

We will use these inputs for our analysis, although other approaches have been developed for dependent inputs. Assume we have n independent input samples $\mathbf{X} = (X^{(1)}, ..., X^{(n)})$. As is often done in the literature, we rely on a polynomial chaos expansion (PCE) which serves as a surrogate model [Sud08; LGMS16; CLMM09]. We are not describing the theory underlying PCE in this section and refer to the Appendix F for more details. The only required knowledge for this section is that one can write the model response as an infinite multivariate polynomial expansion and build a surrogate model by truncating the expansion to a finite subset of the multivariate indices $\mathcal{T} \subset \mathbb{N}^p$:

$$f(x) = \sum_{\boldsymbol{\alpha} \in \mathbb{N}^p} c_{\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}(x) \simeq \sum_{\boldsymbol{\alpha} \in \mathcal{T}} c_{\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}(x) \tag{2.51}$$

where $\psi_{\boldsymbol{\alpha}}(x) = \prod_{i=1}^{p} \psi_{\alpha_i}(x_i)$ is an orthogonal basis of tensorized polynomials. We denote the truncated expansion as $g(x) = \sum_{\boldsymbol{\alpha} \in \mathcal{T}} c_{\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}(x)$. The PCE expansion gives an easy way to express the functions f_A as for any subset $A \in \mathcal{P}_p$:

$$f_A(x_A) \simeq g_A(x_A) = \sum_{\boldsymbol{\alpha} \in \mathcal{T}_A} c_{\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}(x) \tag{2.52}$$

where $\mathcal{T}_A = \{\boldsymbol{\alpha} \in \mathcal{T} \text{ s.t. } \alpha_k \neq 0 \text{ for all } k \in A\}$ [SC13]. Then the dependent Sobol indices can be estimated by:

$$S_A^{(U)} \simeq \frac{\sum_{i=1}^{n} \left(g_A(X_A^{(i)}) - \overline{y_A}\right)^2}{\sum_{i=1}^{n} \left(g(X^{(i)}) - \overline{y}\right)^2} \tag{2.53}$$

$$S_A^{(C)} \simeq \frac{\sum_{i=1}^{n} \left(g(X^{(i)}) - \overline{y}\right)\left(g(X_A^{(i)}) - \overline{y_A}\right)}{\sum_{i=1}^{n} \left(g(X^{(i)}) - \overline{y}\right)^2} - S_A^{(U)}. \tag{2.54}$$

To investigate the influence of each variable, we are evaluating the first-order Sobol indices. They are calculated on the BERP dataset. Half of the points are used to tune the PCE models, and the other half is used to evaluate the Sobol indices. The training and test set contains 1094 instances each. The first order-Sobol indices $S_j^{(U)}$ and $S_j^{(C)}$, as defined in (2.53) and (2.54), are shown in Table 2.2 for each variables. We also

TABLE 2.2: First-order Sobol indices.

$S_j^{(U)}$	R	Y_∞	X_∞
k_p	5.13×10^{-1}	8.33×10^{-1}	9.22×10^{-1}
ε_F	5.89×10^{-1}	3.57×10^{-1}	4.19×10^{-1}
S	8.53×10^{-2}	7.59×10^{-6}	5.21×10^{-5}
x_s	1.04×10^{-2}	7.87×10^{-3}	4.60×10^{-3}
ε_A	7.27×10^{-4}	2.26×10^{-4}	3.92×10^{-4}
Φ	3.83×10^{-4}	4.40×10^{-4}	2.90×10^{-3}
J	4.57×10^{-5}	8.23×10^{-4}	2.35×10^{-3}

$S_j^{(C)}$	R	Y_∞	X_∞
k_p	-1.16×10^{-1}	-1.02×10^{-1}	-1.86×10^{-1}
ε_F	-9.42×10^{-2}	-1.15×10^{-1}	-1.76×10^{-1}
S	2.90×10^{-3}	-4.11×10^{-5}	-4.80×10^{-4}
x_s	-1.40×10^{-3}	1.09×10^{-3}	3.29×10^{-3}
ε_A	1.10×10^{-3}	-1.03×10^{-4}	6.14×10^{-4}
Φ	6.94×10^{-3}	-4.48×10^{-3}	-5.59×10^{-3}
J	6.07×10^{-4}	-3.66×10^{-3}	-6.48×10^{-3}

highlight the confidence in the estimation of the indices in Figure 2.4, where we show a box plot of the estimated structural indices $S_j^{(U)}$. The uncertainties on the indices estimations were obtained by bootstrapping. We did not plot the indices for k_p and ε_F for readability.

This analysis highlights the impact of the newly introduced inputs (ε_A, Φ, J) for

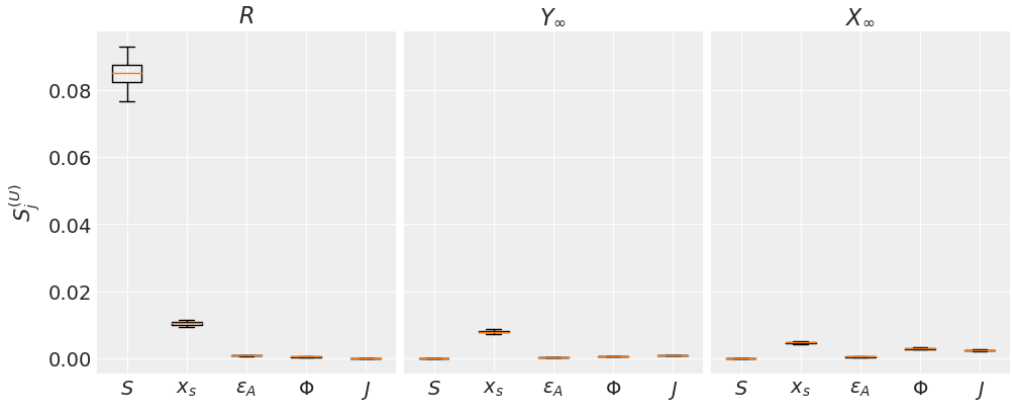

FIGURE 2.4: Box plot of the estimated structural indices $S_j^{(U)}$ for each output channel.

each output considered. As expected, their impact is far less significant than that of k_p or ε_F, but they still contribute noteworthily. One may also notice the negligible contribution of S for the Feynman moments Y_∞ and X_∞ which is in accord with the point model equations.

2.3.2 Feynman moments estimation in MCNP

2.3.2.1 Estimation from the PTRAC files

Each simulation conducted with MCNP provides a PTRAC file and the corresponding input $x = (k_p, \varepsilon_F, S, x_s, \varepsilon_A, \Phi, J) \in \mathcal{X} \subset \mathbb{R}^7$. To obtain the corresponding outputs, the PTRAC file is post-processed to sample the instants of the source events and then, the neutron detection instants and the associated history numbers are extracted.

From here, the count rate is evaluated as the number of detections over the total time. The Feynman moments are estimated by filtered triggered binning with the estimators introduced in (1.124) and (1.125). The triggered binning largely reduces the variances of the Feynman moments estimators by leveraging the knowledge of the history number of neutrons. When solving an inverse problem, we would like to mimic real-world observations for the count rates and Feynman moments. Thus the observations of the direct model would be obtained with sequential binning in

that context since the history number is only available in numerical experiments. However, when building the dataset used to train our surrogate models, it is possible to leverage the additional knowledge brought by the numerical simulations to reduce the noise in the estimations of the Feynman moments and thus improve the surrogate models. This explains why we use triggered binning estimators in the dataset, and sequential binning for the direct model observations.

2.3.2.2 Observation covariance estimation

Let us also mention briefly how one can estimate the covariance structure of the outputs $y = (R, Y_\infty, X_\infty)$.
A first approach if we have multiple iid observations $\mathbf{y} = (y^{(k)})_{1\leq k\leq N}$ from the same calculation, is to estimate the noise covariance by the standard empirical covariance estimator:

$$\widehat{\mathbf{C}_{\text{obs}}} = \frac{1}{N-1}\sum_{k=1}^{N} y^{(k)}(y^{(k)})^T - \overline{y}\,\overline{y}^T \tag{2.55}$$

where $\overline{y} = \frac{1}{N}\sum_{k=1}^{N} y^{(k)}$ is the empirical mean.

Another approach is to find the true covariance with triggered binning. Since the Feynman moments and the count rates are linked to the ordinary moments of the detection statistics (see equations (1.115) and (1.116)), the theoretical covariance is known provided we have access to higher order ordinary moments (up to the order 6). However, using moments with order $p > 3$ is often impossible in practice because of the extremely noisy estimates in the higher-order moment estimators. This method is only suitable for numerical simulations for which we can filter out accidental correlations. Let us consider $M = (M_1, M_2, M_3)^T$ the vector containing the first three moments. This vector is random and its covariance is given by:

$$\text{Cov}(M) = \begin{pmatrix} M_2 - M_1^2 & M_3 - M_1M_2 & M_4 - M_1M_3 \\ M_3 - M_1M_2 & M_4 - M_2^2 & M_5 - M_2M_3 \\ M_4 - M_1M_3 & M_5 - M_2M_3 & M_6 - M_3^2 \end{pmatrix}. \tag{2.56}$$

The covariance of the estimators $\widehat{y} = (\widehat{R}, \widehat{Y_\infty}, \widehat{X_\infty})^T$ is thus given by the delta method:

$$\text{Cov}(\widehat{y}) \simeq \nabla\psi(\widehat{M_1}, \widehat{M_2}, \widehat{M_3})\widehat{\text{Cov}(M)}\nabla\psi(\widehat{M_1}, \widehat{M_2}, \widehat{M_3})^T \tag{2.57}$$

where ψ is the function that maps the ordinary moments to (R, Y_∞, X_∞):

$$\begin{aligned} \psi\colon \mathbb{R}^3 &\longrightarrow \mathbb{R}^3 \\ (M_1, M_2, M_3)^T &\longmapsto (R, Y_\infty, X_\infty)^T \end{aligned}$$

and $\widehat{\text{Cov}(M)}$ is given by (2.56) replacing the moments M_p by their estimators $\widehat{M_p}$.
Let us illustrate this in the BERP example. From the set of observations obtained by sequential binning, the empirical covariance $\widehat{\mathbf{C}_{\text{obs}}}$is given by:

$$\widehat{\mathbf{C}_{\text{obs}}} = \begin{pmatrix} 7.2\times10^{2} & 2.3\times10^{-1} & 1.4\times10^{-2} \\ 2.3\times10^{-1} & 1.0\times10^{-3} & 4.4\times10^{-3} \\ 1.4\times10^{-2} & 4.4\times10^{-3} & 8.5\times10^{-2} \end{pmatrix}. \tag{2.58}$$

Now, estimating the covariance with the higher-order moments with filtered triggered binning yields:

$$\mathrm{Cov}(\hat{y}) = \begin{pmatrix} 3.0 \times 10^{1} & 2.1 \times 10^{-3} & 4.8 \times 10^{-2} \\ 2.1 \times 10^{-3} & 4.7 \times 10^{-5} & 1.3 \times 10^{-4} \\ 1.4 \times 10^{-2} & 1.3 \times 10^{-4} & 1.0 \times 10^{-3} \end{pmatrix} \tag{2.59}$$

The true covariance matrix displays much lower variances overall, with a determinant $|\mathrm{Cov}(\hat{y})| \simeq 3.76 \times 10^{-6} << |\widehat{\mathbf{C}_{\mathrm{obs}}}| = 4.18 \times 10^{-2}$. The triggered binning removes the noise introduced by accidental correlations. The only source of uncertainty comes from the limited number of neutron histories available. In $\widehat{\mathbf{C}_{\mathrm{obs}}}$ however, uncertainties are also caused by accidental correlations. To reduce those uncertainties, we would like to increase the ratio of true correlated detections over accidental correlations, which amounts to having a larger k_p.
In practical experiments, $\mathrm{Cov}(\hat{y})$ is not available and we are working exclusively with $\widehat{\mathbf{C}_{\mathrm{obs}}}$. However, the comparison $|\mathrm{Cov}(\hat{y})| << |\widehat{\mathbf{C}_{\mathrm{obs}}}|$ can be used as a condition to verify that enough neutron histories are used in MCNP runs.

2.3.3 Presentation of the datasets

Now that the methodology used to produce the datasets has been introduced, we describe in detail the datasets themselves.

2.3.3.1 BERP dataset

The first dataset was generated on simulations with a metallic plutonium spherical object with a copper reflector shell. This dataset is known hereafter as the **BERP dataset** since it is based on a reference geometry extracted from the ICSBEP Handbook [Bes+20], which describes a subcritical experiment on the BERP ball. The BERP ball is a metallic plutonium sphere cast in Los Alamos National Laboratory in 1980 which was extensively used for multiplicity and criticality experiments. The reference geometry will serve as our first test case and is described in Section 2.5.2 hereafter. More details are given in Appendix A.

Each data instance corresponds to a randomly generated MCNP geometry, obtained from the reference case by changing the compositions, densities, and geometries of the materials. The eigenvalue calculations were performed with 20 inactive cycles and 200 active cycles with 2×10^5 neutrons in each cycle. The other quantities were obtained with tally measurements. The source intensity is chosen with a uniform distribution on $[1 \times 10^5, 2 \times 10^5]$ to mimic real-world measurements. As an illustration, the source is approximately 1.3×10^5 events per second in the ICSBEP benchmark experiments. The outputs were obtained with filtered triggered binning with an asymptotic time window $T_\infty = 20$ ms. This asymptotic time width is taken such that $T_\infty = \frac{10}{|\alpha|}$ where α is the minimal fitted prompt decay constant obtained with Rossi-α method (see Section 1.4.5) on the data instances.
The dataset contains 2132 instances, split into 1706 training points and 426 test points. The marginal densities and correlation between features and outputs are highlighted respectively in Figures 2.5 and 2.6. One can see that the outputs are highly correlated, especially the second and third Feynman moments.

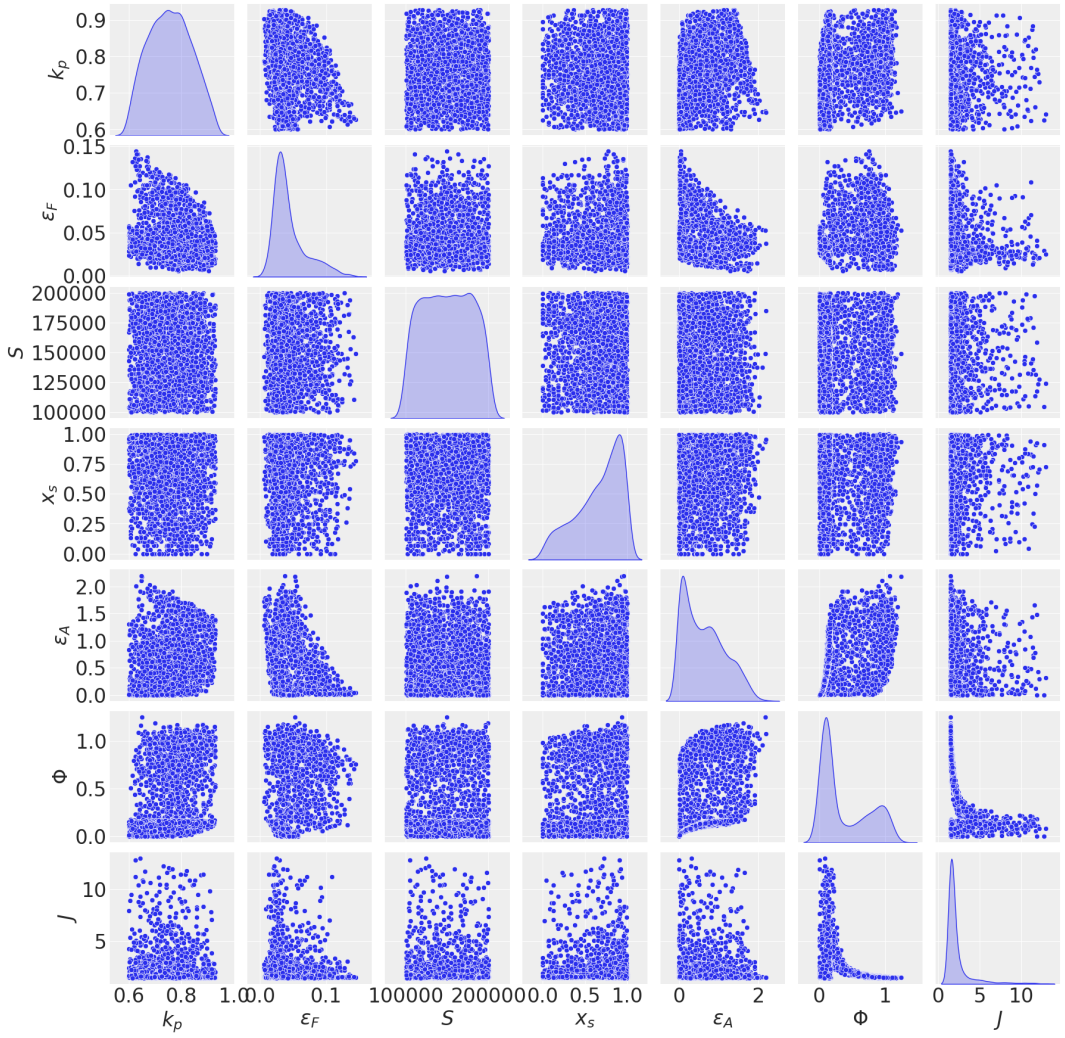

FIGURE 2.5: Marginal densities and correlation plots for the inputs - BERP dataset.

2.3.3.2 SILENE dataset

A second dataset was created to tackle a different case. The reference cases for this new dataset are extracted from a measurement campaign on the SILENE reactor facility, which was a liquid-fueled reactor designed for pulsed experiments as well as subcritical multiplicity measurements [Bar93]. The detailed description of the core is provided in Appendix A, and the applicative test cases are described succinctly in Section 2.5.2.

The dataset is generated with the same procedure as the previous one, though fewer data are available here, with a total of 318 data instances split into a training set of 255 data points and a test set with 63 instances. The dataset will be referred to as the **SILENE dataset** throughout this thesis. The marginal densities and correlation between features and outputs can be found in Figures E.1 and E.1 of Appendix E.

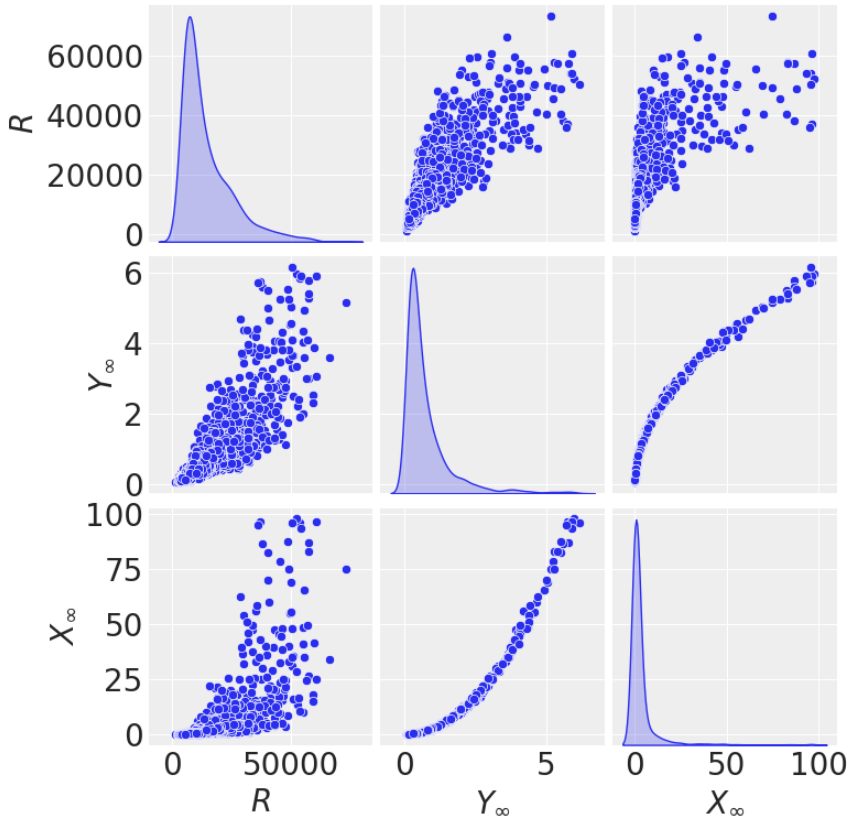

FIGURE 2.6: Marginal densities and correlation plots for the outputs - BERP dataset.

2.3.3.3 Preprocessing

Since the datasets presented in the previous section are meant to serve as training data for the Gaussian process surrogate models, one could raise the question of the suitability of GP models to describe such highly non-Gaussian data. This issue is resolved by preprocessing the dataset.
A common preprocessing step to improve virtually any supervised learning model is to standardize the dataset by setting the empirical mean to zero and the variance to one with a linear transform. While this is good practice, from the look of the correlations in our dataset we will not reach a Gaussian target with just linear transforms. We thus focus on non-linear power transforms.

Let us discuss the two classes of power transforms considered in this work, which are Box-Cox transforms [Sak92] and Yeo-Johnson transforms [YJ00]. Box-Cox transforms are a class of non-linear transformations g_λ parametrized by $\lambda \in \mathbb{R}$ and defined for $x > 0$ by:

$$g_\lambda(x) = \begin{cases} \frac{x^\lambda - 1}{\lambda} & \text{if } \lambda \neq 0 \\ \log x & \text{otherwise.} \end{cases} \tag{2.60}$$

Consider some scalar data $\mathbf{X} = (X_1, ..., X_n) \in \mathbb{R}^n$, and let $\mathbf{Y}^{(\lambda)} = (Y_1^{(\lambda)}, ..., Y_n^{(\lambda)})$ be the transformed data. To make the data more Gaussian, the parameter λ must be chosen appropriately. Thus, we suppose that the transformed data follows a Gaussian distribution $\mathbf{Y}^{(\lambda)} \sim \mathcal{N}(\beta^T \mathbf{X}, \sigma^2)$ where (β, σ^2) are parameters identified by MLE. This leaves us only with λ which is also estimated by MLE.

The same procedure applies to Yeo-Johnson transforms. Yeo-Johnson transforms are a class of non-linear transformations h_λ defined for $x \in \mathbb{R}$ by:

$$h_\lambda(x) = \begin{cases} \frac{(x+1)^\lambda - 1}{\lambda} & \text{if } x \geq 0 \text{ and } \lambda \neq 0 \\ \log(x+1) & \text{if } x \geq 0 \text{ and } \lambda = 0 \\ \frac{(1-x)^{2-\lambda} - 1}{\lambda - 2} & \text{if } x < 0 \text{ and } \lambda \neq 2 \\ -\log(1-x) & \text{if } x < 0 \text{ and } \lambda = 2. \end{cases} \tag{2.61}$$

They have the main advantage of applying to negative data while Box-Cox transforms are restricted to positive data. In this work, Box-Cox transforms are generally used except for models in which we are learning a bias between the true model and the direct model (see Section 2.4.4.1), in which case the data may be negative.

2.4 Performance review of the surrogate models

In this section, we describe the various surrogate models used, how they are built and trained, and their overall performance.

All the scalar GPs and latent scalar GPs are built with Matérn kernel with $\nu = 5/2$, as these provided the best results among standard covariance kernels.
The following naming conventions are adopted for the different GP models.

- **GP-Ind**: each output is a homoscedastic scalar GP.
- **GP-LMC2**: homoscedastic Linear Model of Coregionalization with 2 independent latent Gaussian processes.
- **GP-LMC3**: homoscedastic Linear Model of Coregionalization with 3 independent latent Gaussian processes.
- **GP-Conv**: homoscedastic convolved GPs with 2 independent latent Gaussian processes.

The GP models are trained by maximization of the log-marginal likelihood using the L-BFGS-B algorithm implemented in the *scipy* package. We wish to evaluate the predictive performance of these models as well as the reliability of their uncertainty quantification.

2.4.1 Performance metrics

2.4.1.1 Metrics for the predictive means

Our first focus is on the predictive capabilities of the GPs. Namely, we want the predictive means to be accurate predictors. In a supervised learning framework for regression problems, the most commonly used metrics to assess the model's performance are the Root Mean Squared Error (RMSE) and the Mean Absolute Error (MAE). We focus here on the normalized version of these metrics, known as the NRMSE and NMAE. If f_s is a GP surrogate with predictive distribution $f_s(x) \sim \mathcal{N}\left(\overline{f_s(x)}, \mathbf{C}_s(x)\right)$, the NMAE and NRMSE are defined for the test set $(\mathbf{X}_*, \mathbf{Y}_*)$ by:

$$\text{NMAE} = \frac{1}{n_*} \sum_{i=1}^{n_*} \frac{|\overline{f_s(X_{*,i})} - Y_{*,i}|}{\overline{Y}_*} \tag{2.62}$$

$$\text{NRMSE} = \sqrt{\frac{1}{n_*} \sum_{i=1}^{n_*} \left(\frac{\overline{f_s(X_{*,i})} - Y_{*,i}}{\overline{Y}_*} \right)^2}. \tag{2.63}$$

Let us also introduce the Q^2 coefficient which is defined as:

$$Q^2 = 1 - \frac{\sum_{i=1}^{n_*} \left(f_s(X_{*,i}) - Y_{*,i}\right)^2}{\sum_{i=1}^{n_*} \left(Y_{*,i} - \overline{Y}_*\right)^2} \tag{2.64}$$

where $\overline{Y}_* = \frac{1}{n_*} \sum_{i=1}^{n_*} Y_{*,i}$. The Q^2 coefficient summarized the predictive performance of the GP over the dataset by comparing the predictive mean squared error to the variance of the dataset. The closer to 1 the better. This metric is useful for performance diagnostics in GP regression as it provides a single scalar value that summarizes how well the surrogate model fits our data.
Of course, the metrics presented here only give information on the predictive mean of the trained GP. The GP surrogate model grants us more than just the predictive mean, and the reliability of the predictive variance (or covariance) must also be investigated.

2.4.1.2 Coverage probabilities

To study the reliability of the uncertainty prediction by a GP surrogate, our main focus is on the coverage probabilities. Let us first focus on the scalar case.

Definition 2.4.1. *Consider a scalar GP f_s with predictive distribution at $x \in \mathcal{X}$ given by $f_s(x) \sim \mathcal{N}\left(\overline{f_s(x)}, v_s(x)\right)$. Let $\alpha \in (0,1)$ be some level of credibility. The predictive credibility interval of level α at $x \in \mathcal{X}$ is defined by:*

$$I_\alpha(x) = \left[\overline{f_s(x)} - q_{\frac{1+\alpha}{2}} \sqrt{v_s(x)}, \overline{f_s(x)} + q_{\frac{1+\alpha}{2}} \sqrt{v_s(x)}\right] \tag{2.65}$$

The coverage probability of level α denoted by $C_p(\alpha)$ is defined by:

$$C_p(\alpha) = \frac{1}{n_*} \sum_{i=1}^{n_*} \mathbb{1}_{I_\alpha(X_{*,i})}\left(Y_{*,i}\right) \tag{2.66}$$

It represents the fraction of test points that effectively fall in the credible interval of level α.

For multi-output GP, the coverage probabilities are defined similarly but the $I_\alpha(x)$ are not intervals but credibility regions in the multidimensional space. They are defined with the help of the Mahalanobis distance.

Definition 2.4.2 (Mahalanobis norm)**.** *If C is a positive definite matrix $C \in S_D^+(\mathbb{R})$ and $a \in \mathbb{R}^D$ is a vector, the Mahalanobis norm of a w.r.t. the matrix C is defined by:*

$$\|a\|_C = \sqrt{a^T C^{-1} a}. \tag{2.67}$$

Definition 2.4.3. *Given a multi-output GP f_s, with predictive distribution at $x \in \mathcal{X}$ given by $f_s(x) \sim \mathcal{N}\left(\overline{f_s(x)}, \mathbf{C}_s(x)\right)$, the multi-dimensional credibility region of confidence level $\alpha \in (0,1)$ is defined by:*

$$I_\alpha(x) = \left\{ y \in \mathbb{R}^D \text{ s. t. } \|\overline{f_s(x)} - y\|^2_{\mathbf{C}_s(x)} \leq \chi_\alpha \right\} \tag{2.68}$$

where χ_α is the quantile of level α of the χ^2 distribution with D degrees of freedom.

Throughout this thesis, the coverage probabilities will be the metric of choice to measure the quality of the covariance prediction.

2.4.2 Performance review for the BERP dataset

In this section, we present the performance results of the GP surrogates applied to the BERP dataset. On top of the NMAE, NRMSE, and Q^2 which are shown in Table 2.3, the coverage probabilities for different levels of credibility α are also displayed in Figure 2.7 for all the surrogate models. From these results, several conclusions

TABLE 2.3: Performance of the homoscedastic Gaussian processes surrogate models on the test set - BERP dataset.

GP-Ind	NMAE	NRMSE	Q^2
R	0.008	**0.011**	**0.9994**
Y_∞	**0.022**	**0.040**	**0.998**
X_∞	**0.069**	**0.243**	**0.983**
GP-LMC2	NMAE	NRMSE	Q^2
R	0.008	0.012	0.9993
Y_∞	0.027	0.055	0.995
X_∞	0.087	0.253	0.980
GP-LMC3	NMAE	NRMSE	Q^2
R	**0.008**	0.013	0.9993
Y_∞	0.028	0.060	0.995
X_∞	0.094	0.292	0.973
GP-Conv	NMAE	NRMSE	Q^2
R	0.39	0.51	-0.01
Y_∞	0.066	0.17	0.96
X_∞	0.15	0.53	0.91

are highlighted, for both datasets. First of all, the convolved GPs are not suited to our problem. The random realizations of convolved GPs are likely too smooth to provide good predictors, due to the combination of Gaussian filters, RBF kernels, and convolution products. One can also notice that the naive approach provides the best overall performance when looking only at the predictive means. However, when looking at coverage probabilities, it is clear that the predictive covariances are much less reliable since this model assumes independence between the outputs. Finally, both LMC models offer similar results: they provide reliable uncertainty quantification and sound predictive errors for the mean.

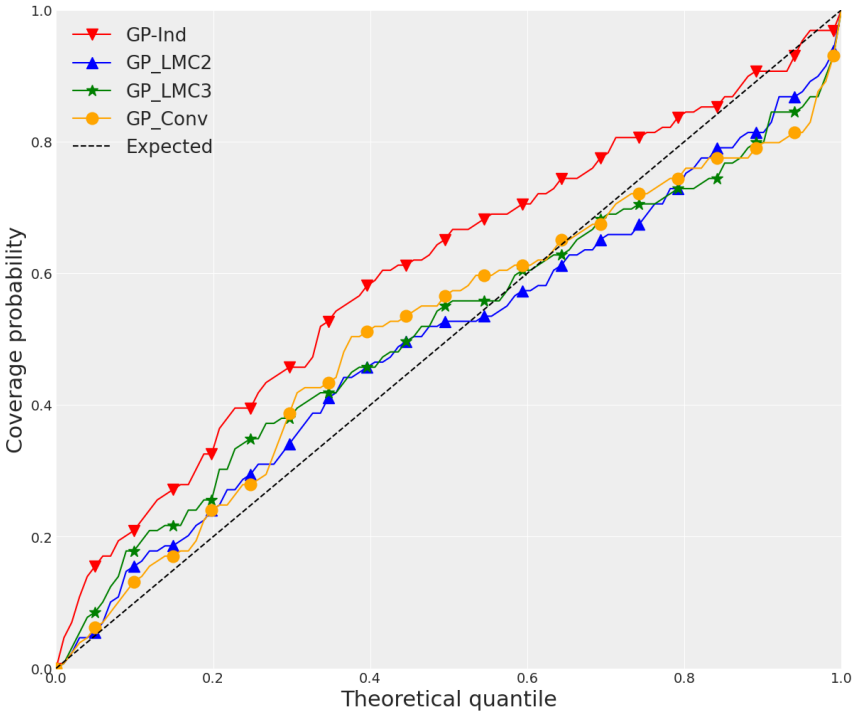

FIGURE 2.7: Coverage probabilities for the surrogate models - BERP dataset.

2.4.3 Performance review for the SILENE dataset

The same metrics are evaluated for GP models built on the SILENE dataset. Fewer data are available but the models tend to perform better than those trained on the BERP dataset, as seen in Table 2.4. The explanation lies in the fact that the BERP dataset spans up to higher k_p values. The resulting Feynman moment distributions are more heavy-tailed and the predictive error on outlier data (say for example $X_\infty > 50$) impacts the performance metrics. The presence of outlier data can be seen in Figure 2.6. This phenomenon is especially pronounced for the third Feynman moment which is more sensitive to k_p variations. This is why the difference in performance metrics is more noticeable for X_∞.

One can also highlight that the number of latent GPs in LMC models is more impactful than in the previous case. The **GP-LMC3** model has better mean predictions, but displays over-estimated coverage probabilities compared to **GP-LMC2**, as can be seen in Figure 2.8. The predictive covariances are generally less reliable, because of the smaller number of data instances for this case. The coverage probabilities are further from their theoretical expected value, and the models tend to overestimate the uncertainties. This overestimation is especially significant for the naive multi-output model **GP-Ind** which does not include correlations between outputs.

2.4.4 Improving the GP surrogates

In the previous section, we saw that LMC GP surrogate models display the best overall performance regarding predictive power and uncertainty quantification. However, there may still be some room for improvement. Namely, we highlight two possible paths to improve our surrogate models in the next paragraphs. The first path relies

TABLE 2.4: Performance of the homoscedastic Gaussian processes surrogate models on the test set - SILENE dataset.

GP-Ind	NMAE	NRMSE	Q^2
R	0.008	0.010	0.9998
Y_∞	0.030	0.051	0.996
X_∞	0.078	0.171	0.992
GP-LMC2	NMAE	NRMSE	Q^2
R	0.008	0.010	0.9998
Y_∞	0.035	0.044	0.996
X_∞	0.081	0.131	0.993
GP-LMC3	NMAE	NRMSE	Q^2
R	**0.007**	**0.009**	**0.9998**
Y_∞	**0.020**	**0.029**	**0.998**
X_∞	**0.069**	**0.115**	**0.995**
GP-Conv	NMAE	NRMSE	Q^2
R	0.046	0.078	0.989
Y_∞	0.030	0.052	0.995
X_∞	0.086	1.65	-0.08

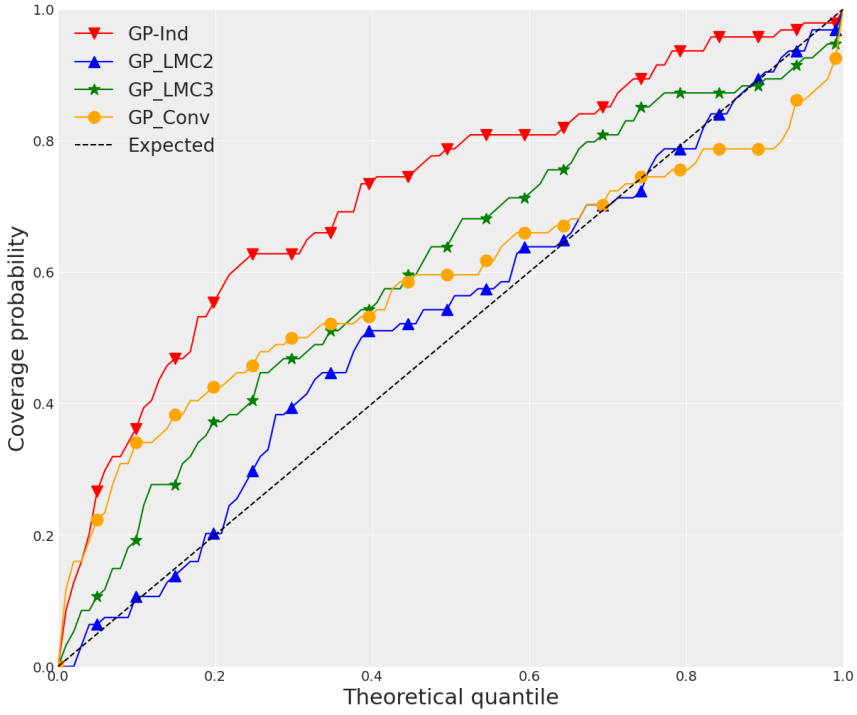

FIGURE 2.8: Coverage probabilities for the surrogate models - SILENE dataset.

on multi-fidelity modeling and the second uses heteroscedastic noise modeling. Since the GP model performance study offers a similar conclusion across both datasets, we only present the results obtained for the BERP dataset, although the results for the SILENE dataset are shown in Appendix E.

2.4.4.1 Multi-fidelity models

When designing surrogate models of costly computer codes, it is often possible to derive a cheaper code or analytical model known as the low-fidelity model. On the other hand, the costly code is known as the high-fidelity model. An example of a low-fidelity model is the diffusion approximation in neutron transport, which simplifies the neutron transport equation to a simpler problem. It can serve as a low-fidelity model, while the full neutron transport is the high-fidelity model. In this work, the low-fidelity model is the analytical point model.
The knowledge of a low-fidelity model can be leveraged to improve surrogate modeling. Indeed, if one only has access to a few data points for the high-fidelity model, it is possible to incorporate the low-fidelity model to improve predictions. The low-fidelity model is generally easy to solve, or typically analytical.

Let f be the high-fidelity model, f_p the low-fidelity model, and f_s the GP surrogate. A simple multi-fidelity model can be obtained by choosing the GP prior such that the mean function is the low-fidelity model $f_s \sim \mathcal{GP}\left(f_p(x), k(x,x')\right)$. This is similar to learning the bias between f_p and the true model f with a zero-mean GP $z \sim \mathcal{GP}\left(0, k(x,x')\right)$:

$$y = f(x) + \varepsilon = f_p(x) + z(x) + \varepsilon \tag{2.69}$$

with ε a Gaussian noise. In [KO01], the high-fidelity response is modeled similarly with an additional multiplicative parameter in front of the low-fidelity response.

$$y = \rho f_p(x) + z(x) + \varepsilon \tag{2.70}$$

However, contrary to this work we are not using a Bayesian hierarchical model, meaning that we are not considering a Bayesian approach for the hyperparameters and the noise parameter. We are only interested in point estimates for the covariance kernel hyperparameters and ρ. They are obtained by maximization of the marginal likelihood.

We are focusing only on GP models built with the Linear Coregionalization Model, with either two or three latent GPs, since they were producing the best results. We study the performance of GP models using the low-fidelity point model. As a first approach, we impose $\rho = 1$, which gives us the two models **BL-LMC2-GP** and **BL-LMC3-GP**, where BL stands for "bias learning", and whose performance are shown in Table 2.5. Similar models are investigated when ρ is not set. These are named

TABLE 2.5: Performance of the multi-fidelity GP with $\rho = 1$ - BERP dataset.

BL-LMC2-GP	NMAE	NRMSE	Q^2
R	**0.008**	**0.010**	**0.9994**
Y_∞	**0.022**	**0.033**	**0.998**
X_∞	**0.069**	0.200	0.988
BL-LMC3-GP	NMAE	NRMSE	Q^2
R	**0.008**	0.011	**0.9994**
Y_∞	**0.022**	0.034	**0.998**
X_∞	**0.069**	**0.186**	**0.990**

respectively **MF-LMC2-GP** and **MF-LMC3-GP**, where MF stands for "multi-fidelity". Their error metrics are displayed in Table 2.6. One can see that for both surrogate models considered, bias learning yields overall better performance when fixing $\rho = 1$

TABLE 2.6: Performance of the multi-fidelity GP with $\rho \neq 1$ - BERP dataset.

MF-LMC2-GP	NMAE	NRMSE	Q^2
R	**0.014**	**0.018**	**0.9986**
Y_∞	**0.064**	**0.120**	**0.978**
X_∞	**0.106**	**0.259**	**0.981**
MF-LMC3-GP	NMAE	NRMSE	Q^2
R	0.017	0.021	0.9979
Y_∞	0.080	0.157	0.962
X_∞	0.151	0.399	0.955

However when ρ is not fixed, the optimization step is made more difficult due to the three additional parameters (for multi-output GPs), which can in turn impact the performance of the surrogate model.
Including the low-fidelity model in the GP, regression improves its mean predictions. Comparing to the results from Table 2.3, one can see that **BL-LMC2-GP** slightly outperforms the model **GP-Ind** which was so far the best model for mean predictions only. There is little difference between the models with 2 or 3 latent GPs. We thus favor the simplest model of the two, which is **BL-LMC2-GP**.

All the results displayed in this section were obtained for the BERP dataset, but similar results were obtained for the SILENE dataset. They can be found in Tables E.1 and E.2 in Appendix E.

Overall, bias learning models can provide improved mean prediction performance. The coverage probabilities for these models are plotted in Figure 2.10 and are very similar to the ones before for the LMC models. Now we introduce heteroscedasticity in our models to improve the reliability of the uncertainty quantification.

2.4.4.2 Heteroscedastic GP

The surrogate models presented up to this point all share an assumption that has not been verified. Indeed, it was assumed that the noise in the training data is not input-dependent: in that case, the model is said to be homoscedastic. However, the noise in the training data could very well be heteroscedastic, that is, input-dependent.

To test the validity of this assumption, let us look at the residuals of our surrogate model predictors. More specifically, since the outputs (R, Y_∞, X_∞) are strongly dependent on the input k_p, let us look at the evolution of the residuals as a function of k_p for the test dataset, which is plotted in log scale in Figure 2.9. One can see that the residuals tend to increase largely with k_p which points toward a heteroscedastic behavior. The Breusch-Pagan-Koenker test [Koe81; PH83] provides a more quantitative diagnostics. It is a χ^2 test in which the squared residuals are modeled with a linear regression w.r.t. the inputs.
The test statistics and p-values are exhibited in Table 2.7 for the BERP dataset. Similar results are obtained for the SILENE dataset. Since the p-values are well below 0.05 for all output channels, one can conclude that the null hypothesis of homoscedastic data is rejected. Then the question is how can we construct heteroscedastic GP surrogate models? Several approaches have been developed, for example in [Mar+12; BC09; RBS10]. The procedure chosen in this work is to model the heteroscedastic noise with

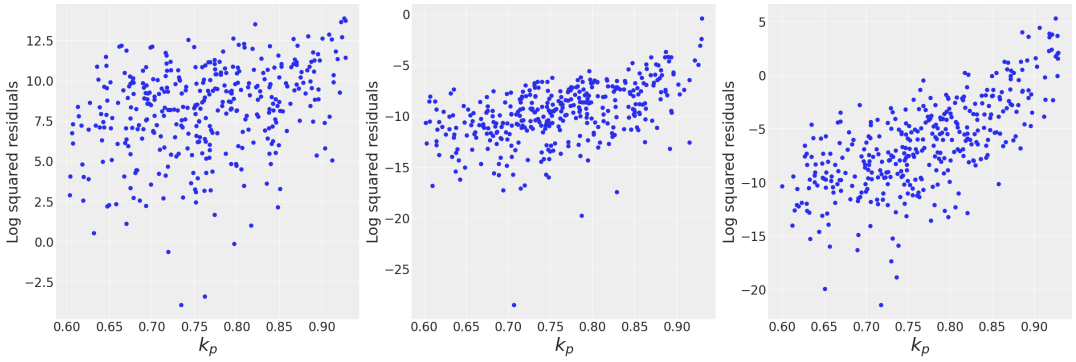

FIGURE 2.9: Log squared residuals for R (left), Y_∞ (center) and X_∞ (right) as a function of k_p, plotted for the test dataset and the surrogate model **GP-LMC2**.

TABLE 2.7: Breusch-Pagan test statistics and p-value for the BERP dataset.

	R	Y_∞	X_∞
p-value	1.1×10^{-5}	3.1×10^{-6}	5.0×10^{-7}
Test statistic	34.9	38.0	42.1

an auxiliary GP as is done in [Ker+07]. We adapt this method to the multi-output case. The methodology is summarized hereafter.

Let us first start by building a homoscedastic GP surrogate model f_1 with the training data $(\mathbf{X}, \mathbf{Y})$ Its predictive distribution is given by $f_1(x) \sim \mathcal{N}\left(\overline{f_1}(x), \mathbf{C}_1(x)\right)$ for any $x \in \mathcal{X}$. Then we draw n_s sampled outputs $(f_{i,j})_{1 \leq j \leq n_s}$ of this GP for every training input X_i such that $f_{i,j} \sim \mathcal{N}\left(\overline{f_1}(X_i), \mathbf{C}_1(X_i),\right)$ and $n_s \in \mathbb{N}$. The heteroscedastic covariances at each input point are estimated by:

$$\Sigma(X_i) = \frac{1}{2n_s} \sum_{j=1}^{n_s} (Y_i - f_{i,j})^T (Y_i - f_{i,j}). \tag{2.71}$$

From here, we have a dataset for the heteroscedastic covariance $(\mathbf{X}, \Sigma(\mathbf{X}))$ where $\Sigma(\mathbf{X}) = (\Sigma(X_1), ..., \Sigma(X_n))$.

With this noise dataset, one can train an auxiliary GP model to predict the heteroscedastic covariance. However, let us pre-preprocess this dataset first. Instead of learning the covariances $\Sigma(X_i)$ we will try to learn the elements of the Cholesky decomposition $L(X_i)$ defined by $L(X_i)^T L(X_i) = \Sigma(X_i)$. The diagonal elements of the Choleksy triangular matrix are strictly positive if $\Sigma(X_i)$ is positive-definite. Finally, let us define the vectors $r(X_i) \in \mathbb{R}^K$ with $K = \frac{D(D+1)}{2}$, by the relations:

$$r(X_i)_k = \begin{cases} \log L(X_i)_{p,q} & \text{for } k = \frac{p(p-1)}{2} + q \text{ if } p = q \\ L(X_i)_{p,q} & \text{for } k = \frac{p(p-1)}{2} + q \text{ if } p > q. \end{cases} \tag{2.72}$$

for $1 \leq k \leq K$ and $1 \leq q \leq p \leq D$.

An auxiliary GP f_2 is trained on the dataset $(\mathbf{X}, r(\mathbf{X}))$. Since we are not interested in the correlation structure of $r(\mathbf{X})$, the auxiliary GP is a naive multi-output GP, with independent scalar GP for each output channel.

This model can be understood as a GP with latent variables being the input-dependent variables. Instead of having a fully Bayesian treatment and integrating out over the distribution of these latent values, we use a most likely estimation and keep only the most likely predictive covariances, which are the predictive means of the auxiliary GP f_2. Inspired by the scalar approach developed in [Ker+07] which describes the optimization step as a procedure similar to the EM algorithm [Moo96], we iterate this procedure until convergence.
A heteroscedastic GP f_{het} is then obtained such that for any input $x \in \mathcal{X}$, the predictive distribution is given by $f_{\text{het}}(x) \sim \mathcal{N}\left(\overline{f_1}(x), \Sigma(x)\right)$ where $\Sigma(x)$ is obtained from the mean prediction $\overline{r(x)}$ of f_2, and then applying the inverse procedure: the Cholesky matrix $L(x)$ is rebuilt, and the heteroscedastic covariance is obtained as $\Sigma(x) = L(x)^T L(x)$.

The metrics used to evaluate the performance of the heteroscedastic GP f_{het} are the same as for homoscedastic GPs. This methodology is used to create a heteroscedastic GP model which uses an LMC homoscedastic GP with 2 latent GP for f_1. This new model is denoted by **HGP-LMC2**. This model includes bias learning since it has been proven beneficial in all cases. The performance of the mean predictions of the heteroscedastic model is the same as its homoscedastic counterpart. The coverage probabilities are not however, and they are plotted, along with those of the newly introduced multi-fidelity models in Figure 2.10, for the BERP dataset. Overall, the coverage probabilities are improved compared to the ones obtained in Figure 2.7. For

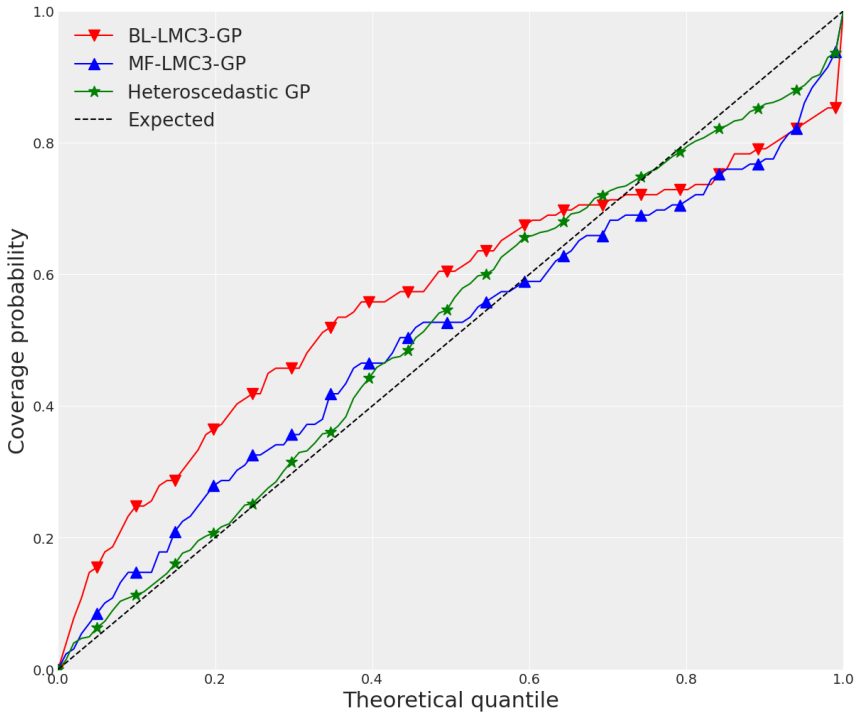

FIGURE 2.10: Coverage probabilities for the MF and BL models - BERP dataset.

the SILENE dataset, the coverage probabilities for the heteroscedastic model, along with those for the multi-fidelity models, are displayed in Figure E.3 in Appendix E.

2.5 GP surrogate models and inverse problems

2.5.1 Global likelihood for epistemic and aleatoric uncertainties

Our final task is to solve the inverse problem with a Bayesian approach. However, our goal is to provide a conservative posterior distribution that accounts for all sources of uncertainties. This means that we need to include the uncertainty induced by the noise in the observations (**aleatoric uncertainty**) as well as the uncertainty of the model itself (**epistemic uncertainty**) since it is still is an approximation of the true underlying physical processes.

Consider some observations denoted by $\mathbf{y} = (y^{(k)})_{1\leq k\leq N}$. Let f_s be a GP surrogate whose predictive distribution at any given point $x \in \mathcal{X}$ is given by $f_s(x) \sim \mathcal{N}\left(\overline{f_s(x)}, \mathbf{C}_s(x)\right)$, where $\overline{f_s(x)} \in \mathbb{R}^D$ is the predictive mean and $\mathbf{C}_s(x) \in S_D^+(\mathbb{R})$ is the positive semi-definite predictive covariance (see equation (2.14)).
The statistical model takes the form:

$$y^{(k)} = \overline{f_s(x)} + \varepsilon_a^{(k)} + \varepsilon_e(x) \tag{2.73}$$

where the $\varepsilon_a^{(k)}$ are iid zero-mean Gaussian random variables $\varepsilon_a^{(k)} \sim \mathcal{N}(\mathbf{0}, \mathbf{C}_{\text{obs}})$ representing the aleatoric uncertainty and $\varepsilon_e(x) \sim \mathcal{N}(\mathbf{0}, \mathbf{C}_s(x))$ represents the epistemic uncertainty. The random variables $\varepsilon_e(x)$ and $\varepsilon_a^{(k)}$ are assumed independent for all $1 \leq k \leq N$.
Let us introduce the flattened vector of observations $\mathbf{y}_{\text{flat}}$ and predictive means $\overline{\mathbf{f_s}(x)}_{\text{flat}}$ defined respectively by:

$$\mathbf{y}_{\text{flat}} = (y_1^{(1)}, ...y_D^{(1)}, ..., y_1^{(N)}, ..., y_D^{(N)})^T \in \mathbb{R}^{DN} \tag{2.74}$$

$$\overline{\mathbf{f_s}(x)}_{\text{flat}} = \left(\overline{f_s(x)}_1, ..., \overline{f_s(x)}_D, ..., \overline{f_s(x)}_1, ...\overline{f_s(x)}_D\right)^T \in \mathbb{R}^{DN}. \tag{2.75}$$

Intuitively, this statistical model assumes that our observations are connected through the model-induced noise which is the same for each observation, but are independent w.r.t. the observational noise which depends mostly on accidental correlations in the detection system and is independent between two MCNP simulations. This statistical model is at the heart of the inverse problem resolution proposed in this thesis.

Proposition 2.5.1. *The likelihood $L(\mathbf{y}|x)$ for $x \in \mathcal{X}$ associated to the statistical model (2.73) is given by:*

$$L(\mathbf{y}|x) = \frac{1}{\sqrt{(2\pi)^{DN}|\mathbf{\Sigma}(x)|}} \exp\left(-\frac{1}{2}\left(\mathbf{y}_{\text{flat}} - \overline{\mathbf{f_s}(x)}_{\text{flat}}\right)^T \mathbf{\Sigma}(x)^{-1}\left(\mathbf{y}_{\text{flat}} - \overline{\mathbf{f_s}(x)}_{\text{flat}}\right)\right) \tag{2.76}$$

where $\mathbf{\Sigma}(x) = \mathcal{U}_N \otimes \mathbf{C}_s(x) + \mathcal{I}_N \otimes \mathbf{C}_{\text{obs}} \in S_{DN}^+(\mathbb{R})$ and $\mathcal{U}_N$ is the square matrix of ones of size N.

Proof. Let $x \in \mathcal{X}$. Starting from the model (2.73), one can write in a more compact form:

$$\mathbf{y}_{\text{flat}}|x = \overline{\mathbf{f_s}(x)}_{\text{flat}} + \mathcal{E}_a + \mathcal{E}_e(x) \tag{2.77}$$

where $\mathcal{E}_a = (\varepsilon_a^{(1)}, ..., \varepsilon_a^{(N)})^T \sim \mathcal{N}(\mathbf{0}, \mathcal{I}_N \otimes \mathbf{C}_{\text{obs}})$ since the $\varepsilon_a^{(k)}$ are mutually independent. Furthermore, we introduced $\mathcal{E}_e(x) = (\varepsilon_e(x), ..., \varepsilon_e(x))^T \sim \mathcal{N}(\mathbf{0}, \mathcal{U}_N \otimes C_s(x))$.

Since $\mathcal{E}_e(x)$ and $\mathcal{E}_a$ are independent, it follows that $\mathbf{y}_{\text{flat}}|x \sim \mathcal{N}\left(\overline{\mathbf{f_s}(x)}_{\text{flat}}, \mathbf{\Sigma}(x)\right)$ with $\mathbf{\Sigma}(x) = \mathcal{U}_N \otimes \mathbf{C}_s(x) + \mathcal{I}_N \otimes \mathbf{C}_{\text{obs}} \in S_{DN}^+(\mathbb{R})$. □

The matrix $\mathbf{\Sigma}(x)$ expresses that observations are independent w.r.t. the observational noise, but correlated by the model error. This likelihood can be further simplified.

Proposition 2.5.2. *The likelihood $L(\mathbf{y}|x)$ is proportional to the following simplified likelihood:*

$$\overline{L(\mathbf{y}|x)} = \frac{1}{\sqrt{2\pi|N\mathbf{C}_s(x) + \mathbf{C}_{\text{obs}}|}} \times \exp\left(-\frac{1}{2}\left(\overline{y} - \overline{f_s(x)}\right)^T \left(\mathbf{C}_s(x) + \frac{1}{N}\mathbf{C}_{\text{obs}}\right)^{-1} \left(\overline{y} - \overline{f_s(x)}\right)\right) \tag{2.78}$$

where $\overline{y} = \frac{1}{N}\sum_{k=1}^{N} y^{(k)}$

Proof. The proof is left in Appendix B.1. □

This likelihood is at the heart of all the MCMC inference methods used to solve the inverse problem in the rest of this work.

2.5.2 Presentation of the test cases

2.5.2.1 BERP test case

For our first application, we are considering the multiplicity benchmark ICSBEP [Bes+19] which contains numerous cases of sub-critical multiplicity measurements. In particular, our focus of interest is the experiment **ICSBEP_FUND-NCERC-PU-HE3-MULT-003_EXP1**, and specifically on the case of the copper-reflected BERP ball. The BERP ball itself is composed of mostly metallic plutonium. The isotopic composition of the plutonium is dominated by the isotope ^{239}Pu (93 wt %) with some ^{240}Pu (7 wt%). The notation wt % refers to the weight percentage. The copper reflectors have a thickness of 1.18 cm. The source is mostly spontaneous fissions, with some residual (α, n) and a total source intensity of $S = 132500$ events.s^{-1}. For a more comprehensive description of the BERP ball characteristics and the experimental set-up, the author refers to Appendix A.

The observations are obtained from time list files via MCNP simulations. The geometry of the MCNP model is slightly simplified compared to the true BERP ball. However, the effective multiplication factor is close to the predicted value in the ICSBEP Benchmark, with a difference of $\Delta k_p = 270$ pcm.
The total number of simulated neutrons is 2×10^6 for each MCNP run. We have a total of $N = 16$ MCNP simulations, each with its own time list file.

The Feynman moments are evaluated by sequential binning of the time list files. Since the values of the Feynman moments depend on the choice of the width T of the detection window, one needs to select the value of T_∞ carefully to guarantee that we are estimating the asymptotic Feynman moments Y_∞ and X_∞. For that matter, we are first conducting a Rossi-α analysis (see Section 1.4.5), which allows us to estimate the prompt decay constant α. The Rossi curve is displayed in Figure 2.11. The estimated value for the prompt decay constant is $\alpha \simeq -9.6$ ms^{-1}. The asymptotic time T_∞ is

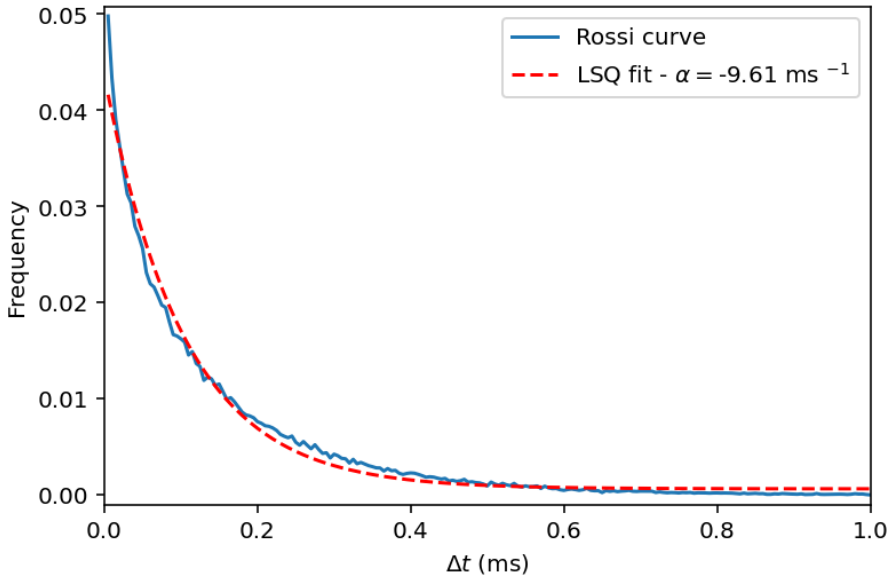

FIGURE 2.11: Rossi curve - BERP test case.

thus chosen to be much greater than α^{-1}, that is $T_\infty = \frac{20}{|\alpha|} \simeq 2$ ms.
The set of observations is summarized in Table E.4 in Appendix E.

2.5.2.2 SILENE test case

Let us now focus on our second example, which is the study of subcritical configurations of the SILENE pulsed reactor. In this section, we are using observations derived from real-world measurements on the SILENE facility. The measurements were obtained for various configurations of fissile height $h \in \{5, 10, 15, 20, 25, 30\}$cm. We are focusing on the cases $h \in \{15, 20, 25\}$cm.
The SILENE reactor is a cylindrical core filled with uranyl nitrate with around 95 wt% enrichment in ^{235}U. The core has a center channel which is designed to shelter instrumentation. In this case, the center channel contains a BF_3 neutron gas counter. The source is an (α, n) point source placed below the core. A precise description of SILENE is provided in Appendix A.

Our observations are real-world measurements obtained during experimental campaigns on SILENE. They come in the form of time list files which must be post-processed to extract the Feynman moments, which is done with sequential binning. The prompt decay constant is estimated with the Rossi-α method, as was done for the BERP test case. The Rossi-α curves are displayed in Figure 2.12 for each configuration studied. The corresponding estimations for α obtained by least-squares estimation are shown in Table 2.8. We also provide the k_p for each configuration, estimated by an MCNP eigenvalue calculation.

TABLE 2.8: Prompt decay constants and multiplication estimation for the different configurations.

h (cm)	15	20	25
α (ms^{-1})	-0.43	-0.72	-1.46
k_p	0.718	0.833	0.906

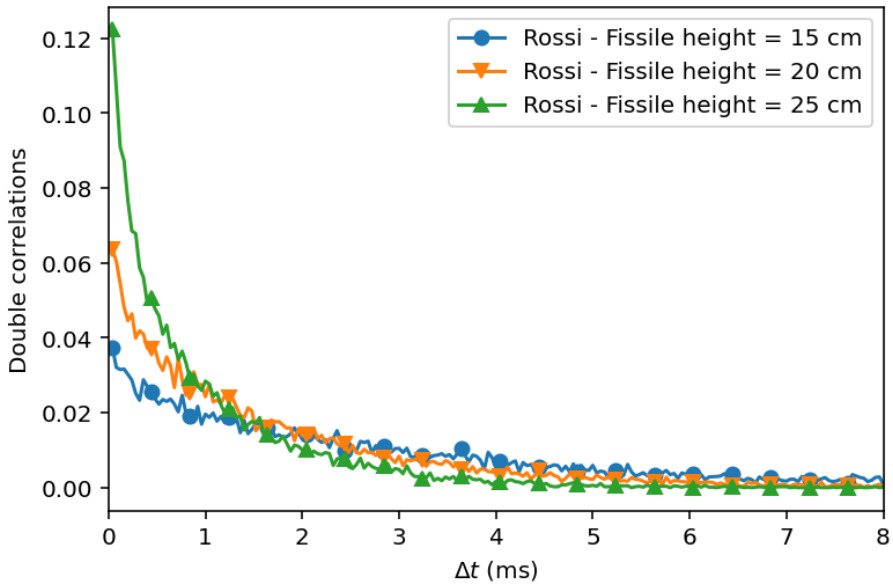

FIGURE 2.12: Rossi-α curve for the SILENE experiment with $h \in \{15, 20, 25\}$ cm of fissile solution height.

2.5.3 Posterior distribution sampling

Let us now solve the Bayesian inverse problems, using some direct model observations and the likelihood (2.78). In this section, we are considering two applications. As a first application, we solve an inverse problem where the observations are obtained from numerical simulations of an experiment from the ICSBEP benchmark. For the second application, we focus on real-world measurements obtained from subcritical multiplicity measurements in the SILENE reactor facility.

For each application, various GP surrogate models and MCMC methods are investigated. The observational data for the test cases are summarized in Table E.4 in Appendix E which presents the observational means $\overline{y}$ and effective covariances $\mathbf{C}_{\text{eff}} = \frac{1}{N}\mathbf{C}_{\text{obs}}$.

2.5.3.1 Results for the BERP test case

To compare the various MCMC methods, let us first sample the posterior distribution with the same surrogate model (**GP-LMC2** in this case). The MCMC sampling is performed on a local computer without parallelization.The autocorrelations for the three first inputs (k_p, ε_F, S) are shown in Figures 2.13, 2.14 and 2.15. The decorrelation times and effective sample sizes over running times are shown for each MCMC algorithm in Table 2.9. One can see that AM greatly reduces the decorrelation time

TABLE 2.9: Summary of the MCMC algorithms performance for the model **GP-LMC2**.

	MH	AM	NUTS
$\widehat{\tau}$	6900	690	280
$\frac{L_{\text{eff}}}{t}$ (min^{-1})	1.0	8.3	1.8

compared to MH, at a minimum cost. NUTS is even more efficient in reducing the

correlation between samples, though it is more costly due to the leapfrog integration. To conclude on MCMC methods, our guideline for the rest of this work is the following: when studying only neutron correlations, AM is preferred as it provides the best compromise between run time and decorrelation. However, in Chapter 5 when gamma correlations are introduced, increasing the inverse problem dimension, NUTS is favored since it performs best in higher dimensions. Let us now visualize the pos-

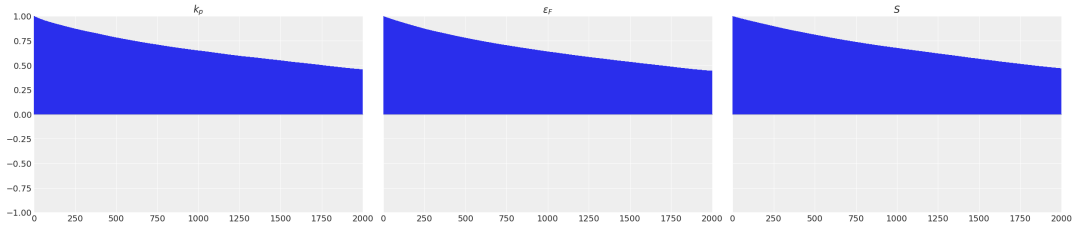

FIGURE 2.13: Autocorrelation for the MH sampling with $L = 1 \times 10^6$ MCMC samples and the **GP-LMC2** model - BERP test case.

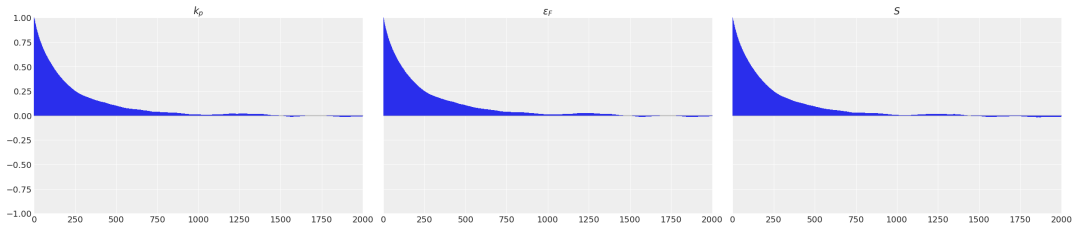

FIGURE 2.14: Autocorrelation for the AM sampling with $L = 1 \times 10^6$ MCMC samples and the **GP-LMC2** model - BERP test case.

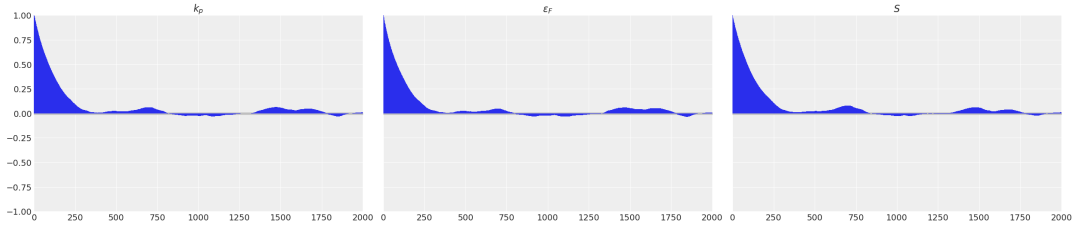

FIGURE 2.15: Autocorrelation for the NUTS sampling with $L = 1 \times 10^6$ MCMC samples and the **GP-LMC2** model - BERP test case.

terior distributions obtained with the various surrogate models tested. We focus on **GP-Ind**, **GP-LMC2**, and **GP-LMC3** since convolved GPs have been shown to display poor predictive capabilities. We are not providing all the two-dimensional marginals here for readability purposes. However, the reader may refer to Figure E.5 Appendix E where the one-dimensional and two-dimensional marginals are all plotted. Then, let us compare the posterior distributions obtained for different surrogate models (sampled with AM). For each surrogate model, the corresponding marginals are plotted respectively in Figures 2.16, 2.17 and 2.18. The posterior distributions appear quite similar, though one may notice a few distinct features. Firstly, the **GP-Ind** seems to provide a reasonable posterior distribution, though we know that its uncertainty quantification is not as reasonable as the LMC models. Besides, the **GM-LMC3** model yields a more concentrated posterior, with a MAP that lies further away from the ground truth. This may be explained by its slightly higher prediction errors when

compared to **GP-LMC2** (see Table 2.3). From these initial figures, **GP-LMC2** appears as the best model.

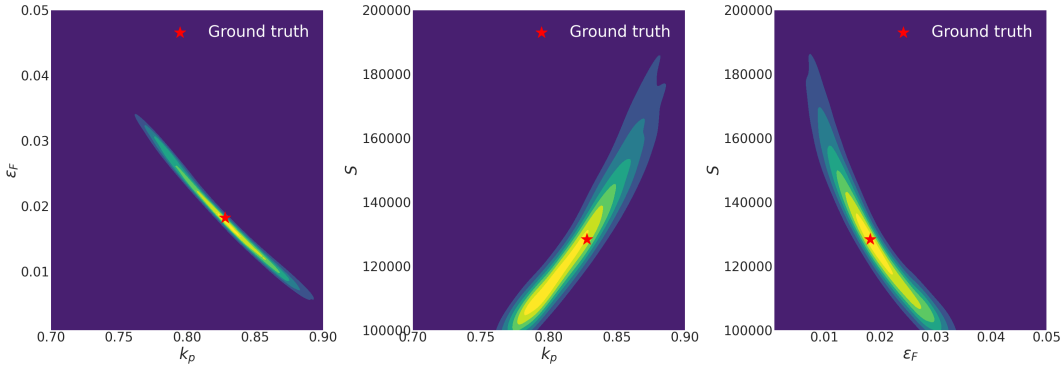

FIGURE 2.16: Posterior distribution marginals obtained by AM for the BERP test case with **GP-Ind**.

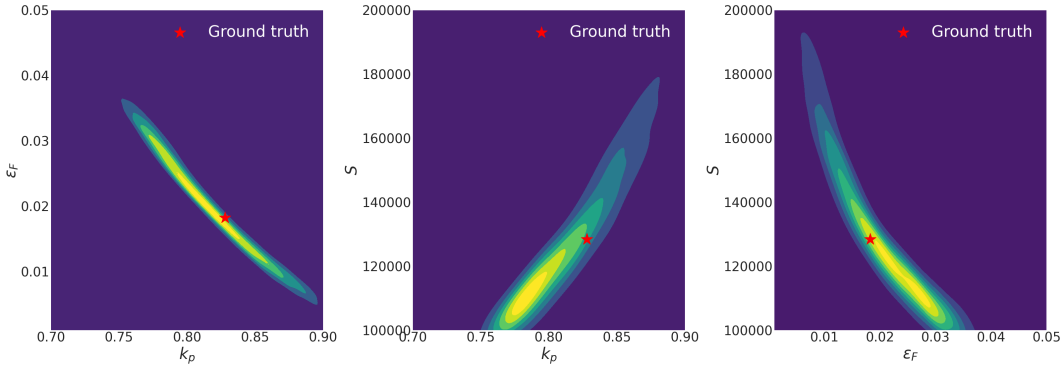

FIGURE 2.17: Posterior distribution marginals obtained by AM for the BERP test case with **GP-LMC2**.

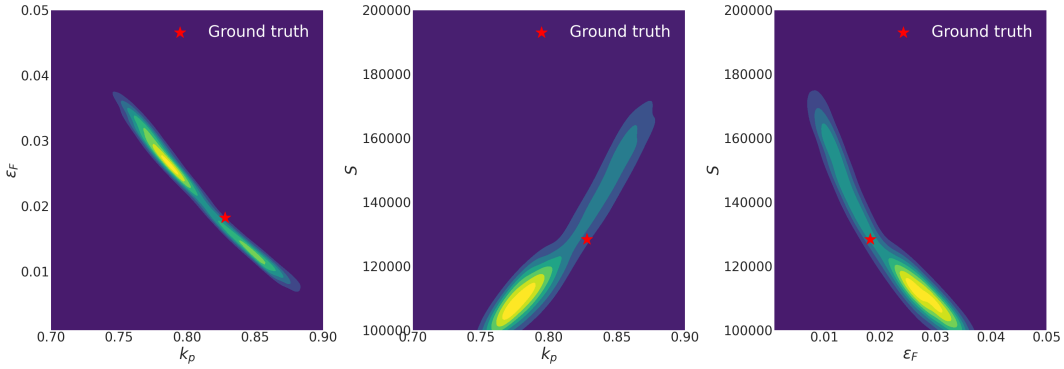

FIGURE 2.18: Posterior distribution marginals obtained by AM for the BERP test case with **GP-LMC3**.

Let us also add to that a three-dimensional scatter plot visualization of the MCMC chain (for the (k_p, ε_F, S) components), where the points are colorized based on their (non-normalized) likelihood. This 3D visualization is displayed in Figure 2.19, for the **GP-LMC2** model. Finally, to highlight the improvement brought by heteroscedastic GPs and bias learning, let us sample the posterior distribution for the BERP test case, with the model **BL-LMC2-GP** with bias learning, and the heteroscedastic model **HGP-LMC2**. The posteriors are displayed in Figures 2.20 and 2.21 respectively. The

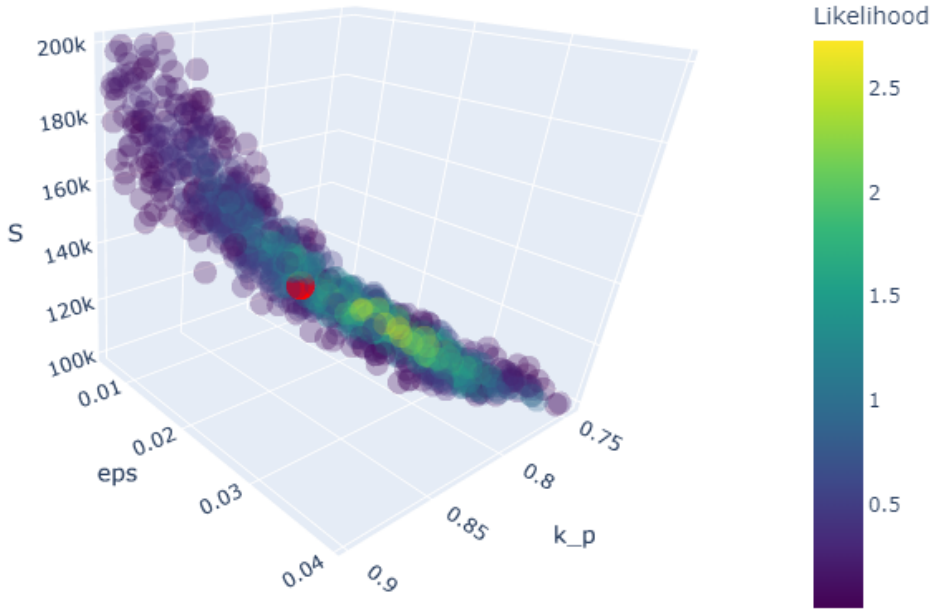

FIGURE 2.19: 3D scatter plot of the MCMC chain obtained for **GP-LMC2**. The ground truth is plotted in red.

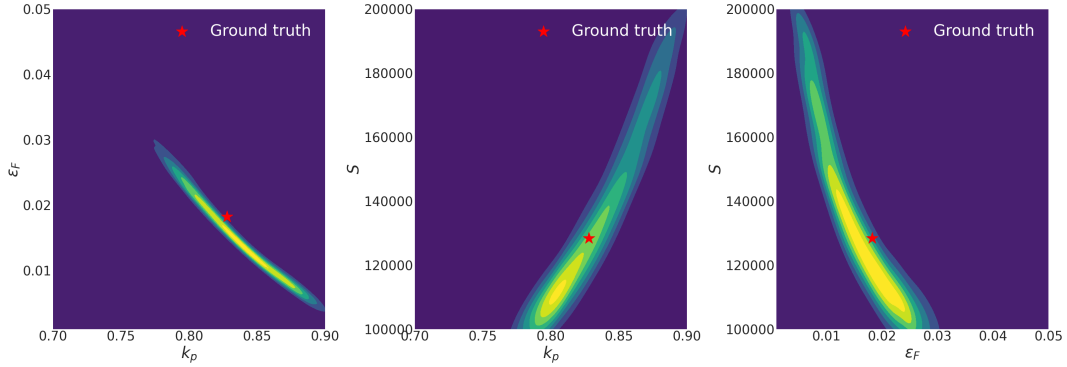

FIGURE 2.20: Posterior distribution marginals obtained by AM for the BERP test case with **BL-LMC2-GP**.

posterior obtained with **BL-LMC2-GP** seems similar to the previous cases, though one can notice a smaller spread in the direction of k_p and a wider spread on S. This may be a direct consequence of the better predictions for X_∞ of the **BL-LMC2-GP** model. Since k_p is especially influential on X_∞, the marginal over k_p is more localized. Overall, we expect this posterior distribution to be more accurate because of the better predictive performances of the model. The heteroscedastic model however yields a wider posterior distribution, with more uncertainties in the estimation of ε_F and S especially. The reason behind this is not straightforward. Our guess is that the covariance varies weakly w.r.t. the inputs, and this dependence is most visible at extreme values of k_p. We may gain some information with heteroscedastic modeling, but the predictive covariances now depend on an auxiliary GP. Without enough data, this auxiliary GP may provide less reliable covariance overall, despite the improved coverage probabilities.

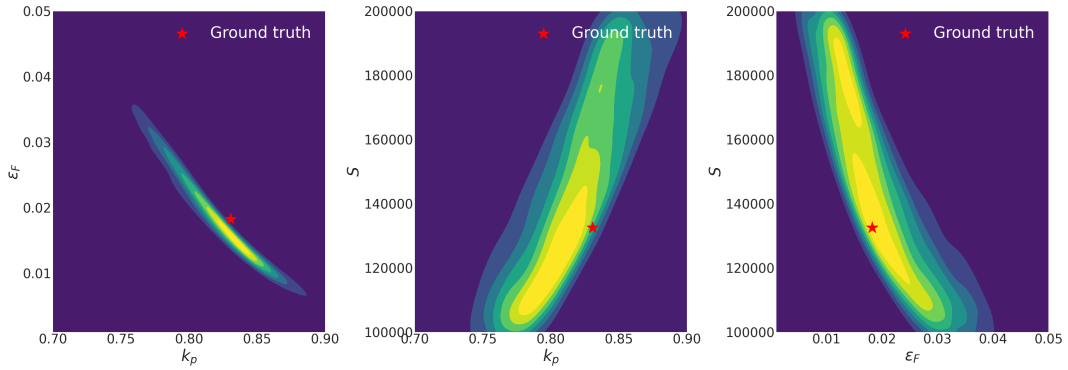

FIGURE 2.21: Posterior distribution marginals obtained by AM for the BERP test case with **HGP-LMC2**.

2.5.3.2 Results for the SILENE test case

Now let us focus on the SILENE test case, where the configurations of interest are $h \in \{15, 20, 25\}$ cm of fissile solution. For each configuration, the posterior distribution is sampled by AM, with 2×10^6 samples, and with the **BL-LMC2-GP** model. The marginals are plotted in Figures 2.22, 2.23 and 2.24. The posterior

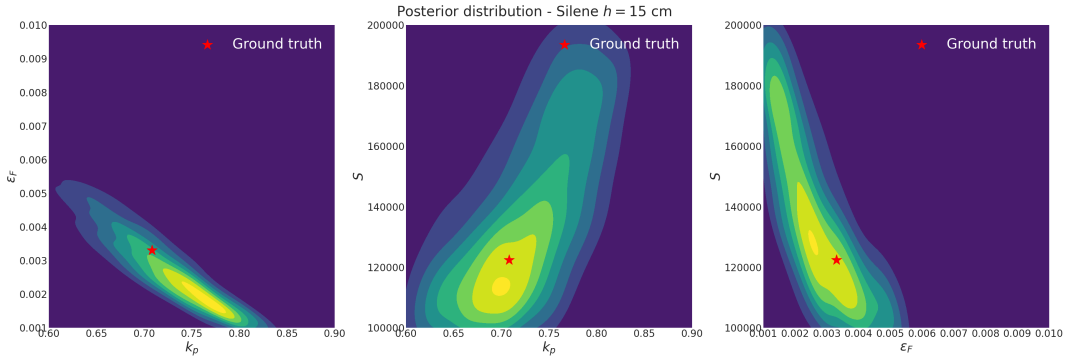

FIGURE 2.22: Marginals of the posterior distribution obtained for $h = 15$ cm of fissile solution height.

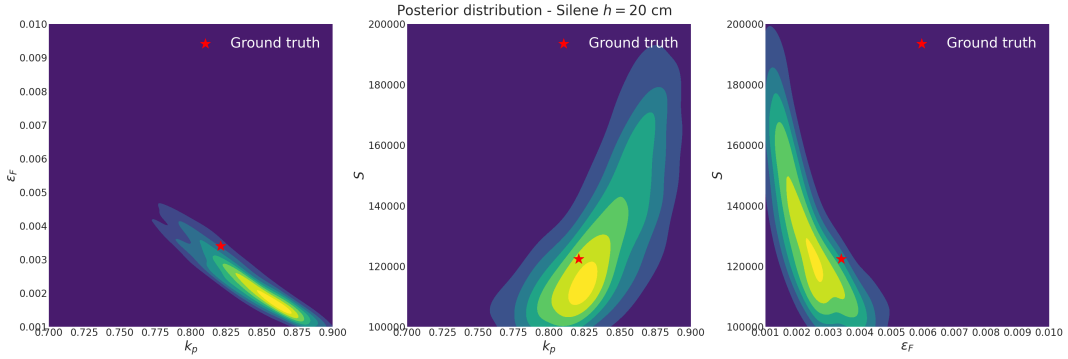

FIGURE 2.23: Marginals of the posterior distribution obtained for $h = 20$ cm of fissile solution height.

distributions obtained can provide good predictions, even on a test case obtained from real-world measurements. The configuration $h = 25$ cm seems to provide less reliable predictions and uncertainties since the ground truth lies slightly outside

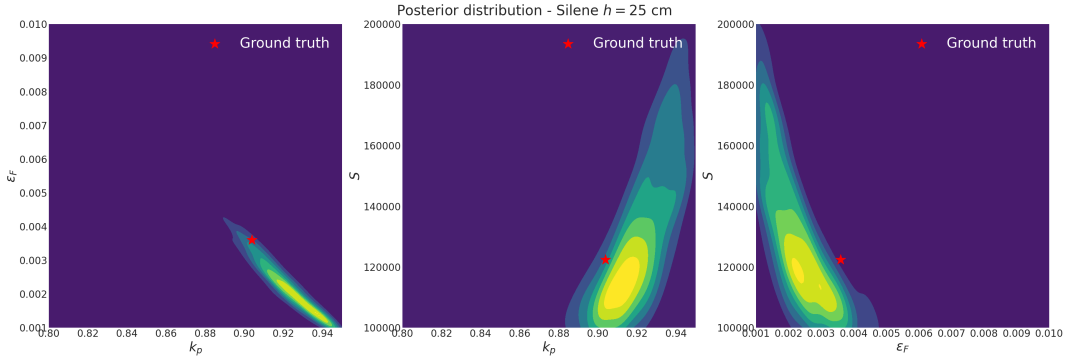

FIGURE 2.24: Marginals of the posterior distribution obtained for $h = 25$ cm of fissile solution height.

the posterior support. The two main explanations are the lack of data, which limits the performance of the GP surrogate, and the fact that this test case is close to the boundary of our dataset, with $k_p = 0.906$. More data at high multiplication could improve the model for that specific case. This also explains why we limited ourselves to the configurations studied here. Indeed, for $h > 25$ cm or $h < 15$ cm, the lack of data at high or low k_p would make the inverse problem resolution difficult.

2.5.4 Conclusion

In this chapter, we have studied the use of Gaussian process surrogate models to solve the inverse problem of interest, while quantifying epistemic and aleatoric uncertainties. The best-performing GPs were built with the Linear Model of Coregionalization. It enables accounting for correlations between outputs, which is especially important in this work. We have shown that the surrogates presented excellent predictive capabilities and reliable uncertainty quantification. Besides, we have been able to further improve these models by using the point model equations as a low-fidelity model, and by using heteroscedastic GP regression.

The surrogates were used as emulators of the direct problem in the Bayesian resolution of the inverse problem. This methodology was applied to two test cases. The first one, adapted from an experiment of the ICSBEP campaign, was a metallic plutonium sphere surrounded by copper reflectors. The second test case was the study of the subcritical behavior of the SILENE facility and was based on real-world measurements instead of numerical simulations.
The sampling of the posterior distribution is conducted via MCMC methods, the most efficient of which was Adaptive Metropolis for this case. The sampling can be performed in less than an hour with sufficient statistics, which makes our methodology applicable for field measurement analysis. The posterior distributions obtained provided reliable uncertainty quantification, and the improvement brought by the low-fidelity inclusion in the surrogates was visible.

In the next chapter, we turn toward Bayesian Neural Networks, a class of surrogate models that may be more flexible than Gaussian processes. Such models can provide output distributions more general than Gaussian distributions and hence could relax the Gaussian approximation made for GP surrogates.

Chapter 3

Bayesian neural network surrogates

This chapter discusses the possibility of using Bayesian Neural Networks (BNN) as surrogate models in the inverse problem at stake. We begin this chapter with a general introduction to BNNs before delving into the construction of our models. We subsequently discuss the resolution of inverse problems with BNN surrogates before applying our developments to the SILENE and BERP test cases.

3.1 An overview of BNN

3.1.1 Artificial neural networks

Artificial neural networks are a class of mathematical models that have attracted increasing attention over the last decades. They have been applied for example to image recognition [LB+95], natural language processing [SP97] or to generative tasks [Bal12; HS97]. In their simplest form, they consist of successive layers where each layer is connected to the next. The output of one layer is obtained from its inputs by linear matrix operations, often followed by a non-linear activation function. One of the characteristics of ANN is the large number of tunable parameters which turn them into highly polyvalent mathematical models. The range of applications of ANN also depends on their architecture and the type of layers in the network. Developments in the underlying architecture of neural networks led for instance to the Large Language Models or the image recognition machines. In this work, we focus mainly on simple dense fully connected layers, for supervised learning applications, which are described hereafter.

Consider some inputs $x \in \mathcal{X} \subset \mathbb{R}^p$. Let us transform these with a simple matrix operation such that we obtain $z = wx + b \in \mathbb{R}^j$ where $w \in \mathbb{R}^{j \times p}$ is a real matrix, known as the weight matrix, and $b \in \mathbb{R}^j$ is known as the bias vector, for some $j \geq 1$. Finally, we apply a non-linear transform in the form of a so-called univariate activation function h. We obtain the final output $y = h(z) = (h(z_i))_{1 \leq i \leq j} \in \mathbb{R}^j$. This mathematical model denoted by $y = h(wx + b)$ is a fully-connected dense layer.
To build a neural network, we apply $L \geq 1$ successive layers as follows:

$$z^{(l+1)} = h^{(l+1)}(w^{(l+1)} z^{(l)} + b^{(l+1)}) \text{ for } 0 \leq l \leq L \tag{3.1}$$

where $z^{(0)} = x \in \mathbb{R}^p$ are the inputs, $z^{(L+1)} = y \in \mathbb{R}^D$ are the model outputs, where for $1 \leq l \leq L$, the weight matrix, the bias vector and the activation function of the l-th hidden layer are respectively $w^{(l)} \in \mathbb{R}^{j_l \times j_{l-1}}$, $b^{(l)} \in \mathbb{R}^{j_l}$ and $h^{(l)}$ with $j_0 = p$. Similarly, $w^{(L+1)} \in \mathbb{R}^{D \times j_L}$, $b^{(L+1)} \in \mathbb{R}^D$ and $h^{(L+1)}$ are the weights, biases, and activation function of the output layer.

The output of the NN model is summarized as $y = N_{\boldsymbol{\phi}}(x)$ where $\boldsymbol{\phi} = (\mathbf{w}, \mathbf{b})$ is the set of parameters of the network, where $\mathbf{w} = \left(w^{(1)}, ..., w^{(L+1)}\right)$ and $\mathbf{b} = \left(b^{(1)}, ..., b^{(L+1)}\right)$.
A NN model can be used for either regression or classification tasks. It requires a training set to tune the model parameters. Consider a dataset $\mathcal{D}_{\text{train}} = (\mathbf{X}, \mathbf{Y})$ with $\mathbf{X} = (X_1, ..., X_n)^T \in \mathcal{X}^n$ and $\mathbf{Y} = (Y_1, ..., Y_n)^T \in \mathbb{R}^{n \times D}$ for $n \geq 1$. The goal of the training is to choose the network parameters $\boldsymbol{\phi} = (\mathbf{w}, \mathbf{b})$ that minimize a given loss function $\mathcal{L}$ which has the form:

$$\mathcal{L}(\mathbf{X}, \mathbf{Y}) = \sum_{i=1}^{n} \ell\left(N_{\boldsymbol{\phi}}(X_i), Y_i\right). \tag{3.2}$$

For regression, ℓ is usually the L^2-norm for example, while for binary classification it can be a binary cross-entropy loss function.
The training phase thus reduces to an optimization problem, which is usually solved with stochastic gradient methods [KB14; DHS11], which can deal with large amounts of data efficiently.

3.1.2 Bayesian extension

Artificial neural networks have two main drawbacks. First of all, their interpretation is difficult. The neural network tends to act like a black box mainly because of the very large number of parameters in the model. Besides artificial neural networks do not provide uncertainty quantification in the predictions. The accuracy of the predictions can only be estimated using a validation dataset but it is not possible to obtain a confidence interval for each prediction.

For our task, we would like to quantify the epistemic uncertainty, that is, the uncertainty of the model. This uncertainty mainly stems from a lack of data. For example, in low-data regimes, the weights and biases obtained after optimization may depend on the network initializations and various developments have been made to tackle this issue [SMG13; MM15].
In this work, we would like to transform ANN into probabilistic models able to provide a prediction as well as confidence in the prediction. For that purpose, we studied Bayesian Neural Networks (BNNs) [Nea12; Mac92].

The idea behind BNNs is to apply Bayesian inference to all weights and biases. Instead of optimizing the point values of the weights based on the training data, their posterior distribution is obtained. New predictions can then be made by integrating over the posterior distribution of the weights.
Let us dive into the BNN formalism more thoroughly. The outputs are assumed scalar in the notations for clarity, but the method can be extended to vector outputs. We assume some knowledge is available about $\boldsymbol{\phi}$, which is quantified by a prior distribution density $p(\boldsymbol{\phi})$. The objective is to evaluate the posterior distribution of the weights after training on the data set $\mathcal{D}_{\text{train}} = (\mathbf{X}, \mathbf{Y})$. Once the posterior distribution is evaluated, the predictions $\mathbf{Y}_* = (Y_{*,1}, ..., Y_{*,n_*})^T \in \mathbb{R}^{n_* \times D}$ for the inputs $\mathbf{X}_* = (X_{*,1}, ..., X_{*,n_*})^T \in \mathcal{X}^{n_*}$, for $x_* \geq 1$, can be obtained by marginalizing over the posterior distribution of the weights.

$$p(\mathbf{Y}_* | \mathcal{D}_{\text{train}}, \mathbf{X}_*) = \int p(\mathbf{Y}_* | \mathbf{X}_*, \boldsymbol{\phi}) p(\boldsymbol{\phi} | \mathcal{D}_{\text{train}}) d\boldsymbol{\phi} \tag{3.3}$$

where $p(\mathbf{Y}_* | \mathbf{X}_*, \boldsymbol{\phi})$ is the likelihood, for fixed network parameters, which is assumed

Gaussian. For fixed $\boldsymbol{\phi}$, the BNN model output Y_* for a given input $X_* \in \mathcal{X}$ is thus given by:

$$Y_* = N_{\boldsymbol{\phi}}(X_*) + \varepsilon \tag{3.4}$$

where $N_{\boldsymbol{\phi}}$ is the deterministic function of the network for fixed weights and biases $\boldsymbol{\phi}$. The quantity $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ is added to model random noise since the network function $N_{\boldsymbol{\phi}}$ is only an approximation of the true model. The regularization parameter σ^2 is treated similarly to the other network parameters. Our goal is thus to obtain the posterior distribution $p(\boldsymbol{\phi}, \sigma | \mathcal{D}_{\text{train}})$. Keep in mind that the regularization term in our application will be a vectorial noise term containing a full covariance, instead of a simpler scalar term as it is here. However, for the sake of simplicity, we keep the notation σ for now. The implementation of the covariance regularization is discussed in Section 3.2.3

The main difficulty is the posterior distribution $p(\boldsymbol{\phi}, \sigma | \mathcal{D}_{\text{train}})$ which is often very complex and makes the integral in (3.3) intractable. If samples $(\boldsymbol{\phi}_m, \sigma_m)_{1 \leq m \leq M}$ of the network parameters and regularization parameter can be obtained, the predictive distribution can be obtained by Monte Carlo:

$$p(\mathbf{Y}_* | \mathcal{D}_{\text{train}}, \mathbf{X}_*) \simeq \frac{1}{M} \sum_{m=1}^{M} \frac{1}{\sqrt{2\pi\sigma_m^2}} \exp\left(-\frac{1}{2\sigma_m^2} \sum_{i=1}^{n_*} (Y_{*,i} - N_{\boldsymbol{\phi}_m}(X_{*;i}))^2 \right). \tag{3.5}$$

The mean and variance for a single prediction Y_* can also be obtained.

$$\mathbb{E}[Y_*] \simeq \frac{1}{M} \sum_{m=1}^{M} N_{\boldsymbol{\phi}_m}(X_*) \tag{3.6}$$

$$\mathbb{V}[Y_*] \simeq \frac{1}{M} \sum_{m=1}^{M} N_{\boldsymbol{\phi}_m}(X_*)^2 + \frac{1}{M} \sum_{m=1}^{M} \sigma_m^2 - \left(\frac{1}{M} \sum_{m=1}^{M} N_{\boldsymbol{\phi}_m}(X_*) \right)^2. \tag{3.7}$$

Hence, the objective is to be able to sample from the posterior distribution $p(\boldsymbol{\phi}, \sigma | \mathcal{D}_{\text{train}})$. Various approaches have been considered to obtain the posterior, and we will discuss them in the next section.

3.1.3 Inference with BNN models

In this section, we introduce the various methods developed to evaluate the predictive distribution of the BNN, which is given by the intractable integral in equation (3.3). In what follows, we include the regularization parameter σ in the definition of $\boldsymbol{\phi}$ to alleviate the notations.

We do not claim to provide a comprehensive review of all possible inference methods for BNNs. Amongst the main methods that are not covered in this thesis, we may think of Laplace approximations [RBB18], ensemble methods [LPB17], or Stochastic Gradient MCMC [WT11; CFG14; Zha+19].

3.1.3.1 MCMC sampling of the weights posterior

One of the most common approaches for sampling the posterior $p(\boldsymbol{\phi} | \mathcal{D}_{\text{train}})$, which is usually considered to be the reference, is to use MCMC sampling. Since the number of network parameters is often very large, this method can be very computationally intensive and is not always tractable. However, the models considered in this work

only contain a few thousand parameters at most and the posterior distribution $p(\boldsymbol{\phi}|\mathcal{D}_{\text{train}})$ can be sampled by MCMC methods.

The state-of-the-art is to use HMC-NUTS which is highly efficient in high dimensions, especially compared to simpler methods like Metropolis-Hastings sampling. We recall that thinning the resulting MCMC chain is necessary to obtain uncorrelated samples.
With MCMC, one can thus obtain a set of $M \geq 1$ uncorrelated samples $(\boldsymbol{\phi}_m)_{1\leq m\leq M}$ taken from $p(\boldsymbol{\phi}|\mathcal{D}_{\text{train}})$. The predictive distribution of the BNN can then be approximated by a Monte Carlo estimation (3.5).

3.1.3.2 Variational inference and Bayes-by-backprop

Another approach, known as variational inference [HVC93; Gra11; BKM17; JJ00], is to approximate the posterior $p(\boldsymbol{\phi}|\mathcal{D}_{\text{train}})$ using a class of parametric distributions $q_\beta(\boldsymbol{\phi})$ with the appropriate choice of parameter β. This approximate distribution is known as the variational distribution. The integral can then be approximated by a Monte Carlo estimation using samples $(\boldsymbol{\phi}_m)_{1\leq m\leq M}$ from the variational distribution $q_\beta(\boldsymbol{\phi})$. The quality of the approximation is dependent on the choice of the variational distribution. How can we choose the best parameter β so that $q_\beta(\boldsymbol{\phi})$ is close to the true posterior $p(\boldsymbol{\phi}|\mathcal{D}_{\text{train}})$?

Various strategies have been considered but the most common approach is to choose the β minimizing a dissimilarity measure between the two distributions. Evaluating such a distance is not always trivial, especially considering that $p(\boldsymbol{\phi}|\mathcal{D}_{\text{train}})$ is inaccessible. One way to overcome this obstacle is to consider the Kullback-Leibler divergence [KL51]. We recall its definition for two probability distributions with density p and q:

$$\text{KL}(p||q) = \int_{\mathcal{X}} p(x) \log\left(\frac{p(x)}{q(x)}\right) dx \tag{3.8}$$

This metric is not formally a distance since it is not symmetric. However, it defines a dissimilarity measure between probability distributions.
In variational inference, the KL divergence is often chosen as a metric to tune the variational distribution. The best parameter $\hat{\beta}$ is chosen as:

$$\hat{\beta} \in \text{argmin}_\beta \, \text{KL}\left(q_\beta(\cdot)||p(\cdot|\mathcal{D}_{\text{train}})\right). \tag{3.9}$$

Though we now have a metric to compare probability distributions, our problem remains. This optimization problem can not be solved since the KL divergence cannot be computed without knowing the posterior.
Let us rephrase the KL divergence using the fact that $p(\boldsymbol{\phi}|\mathcal{D}_{\text{train}}) = \frac{L(\mathcal{D}_{\text{train}}|\boldsymbol{\phi})p(\boldsymbol{\phi})}{p(\mathcal{D}_{\text{train}})}$. Similarly to what is done for sparse GP approximations, minimizing the KL divergence is equivalent to maximizing the evidence lower bound (ELBO) defined by:

$$\text{ELBO} = \mathbb{E}_{q_\beta}\left[\log L(\mathcal{D}_{\text{train}}|\boldsymbol{\phi})\right] - \text{KL}\left(q_\beta(\cdot)||p(\cdot)\right) \tag{3.10}$$

where $L(\mathcal{D}_{\text{train}}|\boldsymbol{\phi})$ is the likelihood of the training data, given fixed network parameters $\boldsymbol{\phi}$. This likelihood is given by:

$$L(\mathcal{D}_{\text{train}}|\boldsymbol{\phi}) = p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\phi}) = \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(Y_i - N_{\boldsymbol{\phi}}(X_i))^2\right). \tag{3.11}$$

The ELBO consists of a data term and a regularization term $\mathrm{KL}\left(q_\beta(\cdot)||p(\cdot)\right)$. The best variational distribution is a compromise between data fit and similarity to the prior. With the right choice of prior $p(\boldsymbol{\phi})$ and variational distribution $q_\beta(\boldsymbol{\phi})$, the KL divergence between $p(\boldsymbol{\phi})$ and $q_\beta(\boldsymbol{\phi})$ is analytically tractable. The first term can also be evaluated by a Monte Carlo approximation since it is possible to obtain samples $(\boldsymbol{\phi}_m)_{1\leq m\leq M}$ from the variational distribution $q_\beta(\boldsymbol{\phi})$:

$$\mathbb{E}_{q_\beta}\left[\log L(\mathcal{D}_{\text{train}}|\boldsymbol{\phi})\right] \simeq \frac{1}{M}\sum_{m=1}^{M}\log\left(L(\mathcal{D}_{\text{train}}|\boldsymbol{\phi}_m)\right). \tag{3.12}$$

The inference strategy for BNN can thus be reduced to an optimization problem. However, one difficulty remains. This optimization problem could be solved with a stochastic gradient descent since the objective function is expressed as a sum over all the data points. But can we obtain the gradient of this objective function with respect to the variational parameter β? This is not straightforward since the objective function itself is a Monte Carlo approximation of an expectation with respect to $q_\beta(\boldsymbol{\phi})$.

Let us consider the data term of the ELBO. We would like to evaluate the gradient of this term with respect to β to fit our variational distribution. Though the ELBO can be computed by Monte Carlo approximation, the evaluation of its gradient is not so trivial.
In [Blu+15; KSW15], the following "parametrization trick" derived from the variational Gaussian trick from [OA09] was introduced to overcome this obstacle.

Proposition 3.1.1 (Reparametrization trick)**.** *Let $\boldsymbol{\phi}$ be a random variable with distribution given by the density $q_\beta(\boldsymbol{\phi})$ and let $g(\boldsymbol{\phi},\beta)$ be a function differentiable with respect to β. Assume that $\boldsymbol{\phi} = t(\beta,\theta)$ where t is a deterministic function and θ is a random variable with density $q(\theta)$ and such that $q(\theta)d\theta = q_\beta(\boldsymbol{\phi})d\boldsymbol{\phi}$.*

$$\frac{\partial}{\partial\beta}\mathbb{E}_{q_\beta(\boldsymbol{\phi})}\left[g(\boldsymbol{\phi},\beta)\right] = \mathbb{E}_{q(\theta)}\left[\frac{\partial g(\boldsymbol{\phi},\beta)}{\partial\boldsymbol{\phi}}\frac{\partial\boldsymbol{\phi}}{\partial\beta} + \frac{\partial g(\boldsymbol{\phi},\beta)}{\partial\beta}\right] \tag{3.13}$$

How is this reparametrization trick used in practice? For most cases, the variational distribution is chosen to be Gaussian such that $q_\beta \sim \mathcal{N}(\mu,\sigma^2)$ and $\beta = (\mu,\sigma^2)$. Thus $\boldsymbol{\phi}$ can be written as $\boldsymbol{\phi} = \mu + \sigma t$ where $t \sim \mathcal{N}(0,1)$. The gradient of the ELBO can thus be computed by an unbiased Monte Carlo estimator derived from equation (3.13). This strategy has been widely applied to variational models, for example with stochastic variational inference [Hof+13] where it helps in reducing the variance of the gradient estimators. However, it is constrained by the existence of a transformation $\boldsymbol{\phi} = t(\beta,\theta)$. This usually restricts the user to certain variational families such as Gaussian distributions, though some recent developments have tried to generalize the reparametrization to more general distributions [FMM18].

3.1.3.3 Monte Carlo Dropout

Dropout was introduced in [Hin+12]. It is a regularization technique used in standard neural networks to prevent overfitting. In standard dropout, a fraction of the neurons in each layer are deactivated at each epoch during the training phase. The probability p of a neuron being turned off is set by the user. Monte Carlo Dropout (MCD), which was introduced in [GG16], is founded on the same principle. This paper highlights the connection between dropout layers and variational deep Gaussian processes. We briefly describe MCD hereafter.

Let us first consider a fully connected neural network with L layers, with K_l neurons in layer l. In each layer l, the neurons can be dropped out with probability $p_l \in (0,1)$. Suppose we have some training data $\mathcal{D}_{\text{train}} = (\mathbf{X}, \mathbf{Y})$, with $\mathbf{X} = (X_1, ..., X_n)^T \in \mathcal{X}^n$ and $\mathbf{Y} = (Y_1, ..., Y_n)^T \in \mathbb{R}^{n \times D}$ The network parameters are denoted $\boldsymbol{\phi}$ as in the previous sections. Assuming a square loss, the objective function of the network is :

$$\mathcal{L}(\mathbf{X}, \mathbf{Y}) = \sum_{i=1}^{n} \|Y_i - N_{\boldsymbol{\phi}}(X_i)\|_2^2 + R(\boldsymbol{\phi}) \tag{3.14}$$

where $N_{\boldsymbol{\phi}}(X_i)$ is the feed-forward prediction of the network for the input point X_i and the network parameters $\boldsymbol{\phi}$. The term $R(\boldsymbol{\phi})$ refers to a regularization term. Usually it is a L^2-regularization on the weights matrices $w^{(l)} \in \mathbb{R}^{j_l \times j_{l-1}}$ and the bias vector $b^{(l)} \in \mathbb{R}^{j_l}$.
It is possible to show that this formulation is equivalent to a variational deep Gaussian process with latent variables $\mathbf{w} = \left(w^{(1)}, ..., w^{(L+1)}\right)$ where $w^{(l)} = M^{(l)} \text{diag}(z_1^{(l)}, ..., z_{j_l}^{(l)})$ where $M^{(l)}$ is a variational matrix parameter, and the $z_j^{(l)}$ are Bernouilli variables with probability p_l. We are not detailing the proof of this result here. Deep GPs are not the focus of this thesis, but the interested reader can refer to [DL13] for the general description of such models.

Therefore, Monte Carlo dropout can be understood as a variational approximation of deep Gaussian processes, which makes it possible to quantify uncertainties. The uncertainties on a given prediction can be obtained by repeated feed-forward calls to the network, where for each call, a fraction of the neurons are deactivated. For a given input X, assuming a total of M calls to the network, with each having parameters $\boldsymbol{\phi}_m$ and returning an output $N_{\boldsymbol{\phi}_m}(X)$ for $1 \leq m \leq M$, the predictive mean and variances are simply obtained by (3.6) and (3.7). This is easily generalized to a multi-output setting.
This method allows for uncertainty quantification in neural networks. However one can wonder about the reliability of the uncertainties provided. Indeed, the underlying variational distribution may not be well suited to approximate the true posterior distribution in a Bayesian setting. The limits of Monte Carlo Dropout have been pointed out in [Liu+21; PF18], in which over-confident predictions are obtained for out-of-data instances.

3.2 Probabilistic modeling with BNNs

In the previous section, we described the general theory behind Bayesian Neural Networks, and especially the various methods that can be employed to provide an output predictive distribution. In this section, we focus on the construction of surrogate models with BNNs.

3.2.1 Network architecture

We consider a network composed of fully-connected dense layers. Each layer contains $K_n = 50$ neurons, and we are considering a model with $L = 2$ hidden layers. There is no need for a deep architecture in this work since the direct model that we seek to model is rather simple and low-dimensional. Some details on the ablation study leading to this final architecture are given in Appendix D.

The weight matrices are denoted by $w^{(1)} \in \mathbb{R}^{K_n \times p}$, $w^{(2)} \in \mathbb{R}^{K_n \times K_n}$ and $w^{(3)} \in \mathbb{R}^{D \times K_n}$. Similarly, the bias vectors are $b^{(1)} \in \mathbb{R}^{K_n}$, $b^{(2)} \in \mathbb{R}^{K_n}$ and $b^{(3)} \in \mathbb{R}^{D}$. The activation functions are sigmoid for every layer except the final layer in which a linear function is used.
For an input $x \in \mathcal{X}$, the values of the neurons in the l-th layer, for $1 \leq l \leq L+1$, is denoted by $z^{(l)} = h^{(l)}\left(w^{(l)} z^{(l-1)} + b^{(l)}\right)$ where we write $z^{(0)} = x$ and $z^{(L+1)} = N_{\boldsymbol{\phi}}(x)$ is the model output, and where $h^{(l)}$ is the activation function associated with the l-th layer. The network has a total of 3112 parameters.

3.2.2 Loss function

Consider an input $x \in \mathcal{X}$. We want the BNN to provide a multivariate normal distribution $Y_{\boldsymbol{\phi},\mathbf{C}}(x) \sim \mathcal{N}\left(N_{\boldsymbol{\phi}}(x), \mathbf{C}\right)$ where $(\boldsymbol{\phi}, \mathbf{C})$ are themselves randomly distributed according to a posterior $p(\boldsymbol{\phi}, \mathbf{C} | \mathcal{D}_{\text{train}})$. Let us first begin with a homoscedastic approach in which the covariance $\mathbf{C}$ does not depend on x.
Let $(\mathbf{X}, \mathbf{Y})$ be our training dataset. For a given set of network parameters $\boldsymbol{\phi}$ and a covariance $\mathbf{C}$, we define the loss function $\mathcal{L}_{\boldsymbol{\phi},\mathbf{C}}(\mathbf{X}, \mathbf{Y})$ as:

$$\mathcal{L}_{\boldsymbol{\phi},\mathbf{C}}(\mathbf{X}, \mathbf{Y}) = \left((2\pi)^D |\mathbf{C}|\right)^{-n/2} \exp\left(-\frac{1}{2}\sum_{i=1}^{n}\left(Y_i - N_{\boldsymbol{\phi}}(X_i)\right)^T \mathbf{C}^{-1}\left(Y_i - N_{\boldsymbol{\phi}}(X_i)\right)\right). \tag{3.15}$$

This loss imposes a Gaussian distribution output. We wish to estimate the posterior distribution of $\boldsymbol{\phi}$ and $\mathbf{C}$ given this likelihood.
Of course, imposing a Gaussian distribution is an approximation. However, compared to a Gaussian process, this BNN model integrates over the whole posterior $p(\boldsymbol{\phi}, \mathbf{C} | \mathcal{D}_{\text{train}})$. If we consider a Monte Carlo approximation of the integral in (3.3), the predictive distribution of the BNN is thus a mixture of multivariate Gaussian distributions with means $\boldsymbol{\phi}_m$ and covariances $\mathbf{C}_m$, where $\boldsymbol{\phi}_m$ and $\mathbf{C}_m$ are sampled from $p(\boldsymbol{\phi}, \mathbf{C} | \mathcal{D}_{\text{train}})$, for $1 \leq m \leq M$.

3.2.3 Choice of the priors

The choice of the priors can largely impact the performance of a BNN. In this section, we describe the choices made for the prior distributions of the weights, biases, and covariances.

3.2.3.1 Weights and biases priors

The priors on the network weights were first chosen as independent identically distributed zero-mean Gaussian priors $p(w) \sim \mathcal{N}\left(0, \sigma_w^2\right)$. The same is done for the biases with a prior $p(b) \sim \mathcal{N}\left(0, \sigma_b^2\right)$. This choice is mainly motivated by simplicity in this work. An additional idea is to initialize the prior means to the values obtained for a point neural network. This alternate approach appears beneficial for larger networks only. For the architecture chosen, zero-mean priors provide better predictive performances. The results are highlighted in Appendix D. The choice and influence of the priors in BNN is still a vivid field of research [For+21; For22; GY20]. The optimal selection of the variance is not obvious and was obtained by a grid-search approach, looking for values $\sigma_w^2, \sigma_b^2 \in \{0.1, 0.5, 1.0, 2.0, 5.0, 10.0\}$. The best estimates for the prior were found to be $\sigma_w^2 = 5.0$ and $\sigma_b^2 = 5.0$.

3.2.3.2 Covariance priors

For a given set of parameters, we want our BNN models to output a multivariate normal distribution with some covariance $\mathbf{C} \in S_D(\mathbb{R})^+$. We need to define some prior distribution $p(\mathbf{C})$ for the covariance $\mathbf{C}$.
A naive guess would be to choose a diagonal covariance and select priors on the diagonal elements $c_{i,i}$ for $1 \leq i \leq D$. However, as can be seen from the datasets in Figure 2.6, the outputs are strongly correlated. Their covariance structure cannot be fully described by a diagonal output covariance.

Instead, we focus on two classes of priors for positive-definite matrices, namely the Inverse-Wishart distribution [ANS14] and the Lewandowski-Kurowicka-Joe (LKJ) prior [HW13; LKJ09].

Definition 3.2.1. *Let $\mathbf{M} \in S_D(\mathbb{R})^+$ be a positive definite matrix known as the scale matrix and let $\mu > D-1$ be a real number, defined as the degrees of freedom. For $\mathbf{C} \in S_D(\mathbb{R})^+$, the Inverse-Wishart distribution is a probability distribution defined by the density:*

$$f(\mathbf{C}) = \frac{|\mathbf{M}|^{\mu/2}}{2^{\mu D/2}\Gamma_D(\mu/2)}|\mathbf{C}|^{-(\mu+D+1)/2}\exp\left(-\frac{1}{2}\mathrm{Tr}(\mathbf{M}\mathbf{C}^{-1})\right). \tag{3.16}$$

When a random variable $\mathbf{C}$ has the Inverse-Wishart distribution with scale matrix $\mathbf{M}$ and degrees of freedom $\mu > D-1$, we write $\mathbf{C} \sim \mathcal{IW}(\mathbf{M}, \mu)$.

If $\mathbf{C} \sim \mathcal{IW}(\mathbf{M}, \mu)$, then $\mathbf{C}^{-1}$ follows a Wishart distribution. The Wishart distribution is the multivariate equivalent of the χ^2 distribution. Indeed, considering independent Gaussian vectors $(X_i)_{1\leq i\leq \mu}$ with distribution $X_i \sim \mathcal{N}(0, \mathbf{M})$ then the matrix $\mathbf{C} = \sum_{i=1}^{\mu} X_i X_i^T$ follows a Wishart distribution $\mathbf{C} \sim \mathcal{W}(\mathbf{M}, \mu)$. The inverse-Wishart distribution is extensively used in Bayesian statistics as a covariance prior since it is the conjugate prior to the multivariate Gaussian for the estimation of covariance matrices (see for example [DP02]).
Now let us define the LKJ distribution, which is the other covariance prior investigated in this work.

Definition 3.2.2. *Let $\eta > 0$ be a real, known as the concentration. Let $S_{D,c}(\mathbb{R})^+$ be the subset of $S_D(\mathbb{R})^+$ comprising matrices with unit diagonals, otherwise known as correlation matrices. For $\mathbf{\Sigma} \in S_{D,c}(\mathbb{R})^+$, the LKJ prior is defined by the density function:*

$$f(\mathbf{\Sigma}) = K \times |\mathbf{\Sigma}|^{\eta} \tag{3.17}$$

where K is a normalization constant. When a random variable $\mathbf{\Sigma}$ has the LKJ distribution with concentration η, we write $\mathbf{\Sigma} \sim \mathcal{LKJ}(\eta)$.

The LKJ prior is defined only over correlation matrices, thus one also needs to introduce a diagonal matrix $\mathbf{S} = \mathrm{diag}(\sigma_1, ..., \sigma_D)$, to obtain a covariance matrix $\mathbf{C} = \mathbf{S\Sigma S}$. In this work, we chose to sample the diagonal vector σ from a Half-Cauchy distribution truncated at zero $\sigma \sim \mathcal{HC}(0)$.
To investigate the influence of the covariance prior, we studied the LKJ prior with some grid-search for $\eta \in \{0.5, 0.8, 1.0, 1.5, 3.0\}$ and $\sigma \sim \mathcal{HC}(0)$. The optimal choice of the concentration η obtained with the grid-search approach is $\eta = 0.8$. Since lower values of η tend to favor correlations in the matrix, this value is in agreement with our intuition, since we expect the covariance to display correlations as the outputs

are strongly correlated. However, the value of η remains close to 1 to keep a rather non-informative prior for the predictive covariance.

Then, we study the Inverse-Wishart prior $\mathbf{C} \sim \mathcal{IW}(\mathbf{M}, \mu)$ for $\mu \in \{3, 5, 8, 10\}$. The scale matrix $\mathbf{M}$ is taken to be close to the expected covariance matrix in the output. It is taken as the empirical covariance of the outputs evaluated over the whole training dataset, which is for the BERP dataset:

$$\mathbf{M} = \begin{pmatrix} 1.12 \times 10^8 & 6.43 \times 10^3 & 5.09 \times 10^4 \\ 6.43 \times 10^3 & 4.94 \times 10^{-1} & 4.39 \times 10^0 \\ 5.09 \times 10^4 & 4.39 \times 10^0 & 4.77 \times 10^1 \end{pmatrix}. \tag{3.18}$$

The optimal choice found for the degrees of freedom was $\mu = 10$ for our problem. The interpretation μ is less straightforward than for the concentration parameter in LKJ priors.

3.2.4 Performance review

Let us now describe the overall performance of the BNN models studied. The main metrics of interest are the same as for GP surrogates (see Section 2.4.1). All models have the same architecture but differ in their inference strategy. The following models are considered:

- **LKJ-BNN**: HMC-NUTS inference with LKJ prior for the covariance.
- **IW-BNN**: HMC-NUTS inference with Inverse-Wishart prior for the covariance.
- **VI-BNN**: BNN model with variational inference.
- **MCD-BNN**: BNN model with Monte Carlo Dropout inference.

The performance metrics are displayed in Table 3.1. Monte Carlo dropout does not appear as a reliable inference method for our BNN surrogate as it struggles to provide satisfying predictions for all outputs. On the other hand, the predictive mean capabilities of all the other methods are mostly similar, or slightly superior to the GP models presented in the previous chapter. The emphasis is on the **LKJ-BNN** model which offers the best mean predictions.

Now let us look at the coverage probabilities for each of the models. Since the approximated output distribution is a Gaussian mixture when using a Monte Carlo estimation for the integral in (3.3), one cannot directly use the coverage probabilities introduced in definition 2.4.3 to measure the reliability of the predictive covariances. However, it is possible to take the mean of the samples (either obtained by variational inference or MCMC) and compute coverage probabilities with the mean of the predictive means and covariances. One justification for looking at these averaged quantities is that BNN surrogates will be used in the inverse problem by averaging their prediction so that we fall back to a Gaussian output distribution (see Section 3.3.1). These parameter-averaged coverage probabilities are presented in Figure 3.1. All the BNN models tend to consistently overestimate the uncertainties. In particular, **IW-BNN** provides the worst coverage probabilities, despite having good predictive mean metrics. On the other hand, **LKJ-BNN** displays the best coverage probabilities while still having satisfying mean predictive performance. Now, one can also study the number of samples required to effectively represent the posterior distribution $p(\boldsymbol{\phi}|\mathcal{D}_{\text{train}})$. To inquire about this aspect, let us vary the number M of samples $(\boldsymbol{\phi}_m)_{1 \leq m \leq M}$ used

TABLE 3.1: Performance of the homoscedastic BNN - BERP dataset.

MCD-BNN	NMAE	NRMSE	Q^2
R	0.3836	0.5457	0.4581
Y_∞	0.4742	0.8391	0.3688
X_∞	0.7629	2.5679	0.072
VI-BNN	NMAE	NRMSE	Q^2
R	0.0079	0.0128	0.9997
Y_∞	**0.0196**	0.0365	0.9987
X_∞	0.0606	0.1965	0.9946
LKJ-BNN	NMAE	NRMSE	Q^2
R	0.0084	0.0133	**0.9997**
Y_∞	0.0201	**0.0362**	**0.9988**
X_∞	**0.0600**	**0.176**	**0.9956**
IW-BNN	NMAE	NRMSE	Q^2
R	**0.0076**	**0.0126**	0.9997
Y_∞	0.0205	0.0417	0.9984
X_∞	0.0723	0.2811	0.9889

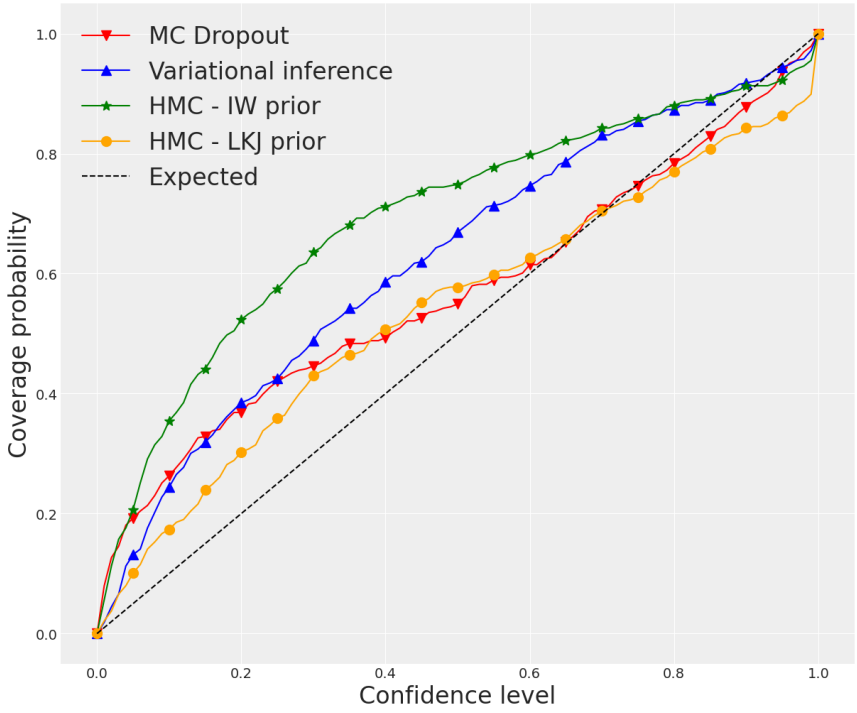

FIGURE 3.1: Coverage probabilities for the various parameter-averaged models - BERP dataset.

to compute the coverage probabilities. For a given model (**IW-BNN** in this case), we compute the coverage probabilities 100 times where each iteration is done for a different subset of samples, with size $W \in \{5, 10, 30, 100, 300, 1000\}$. For each value of W, we obtain a mean and confidence interval for the coverage probabilities. These are plotted in Figure 3.2. One can see that the coverage probabilities do not spread exceedingly even at a low number of samples. This is explained by the relatively low epistemic error since the BERP dataset contains sufficient data. The predictive

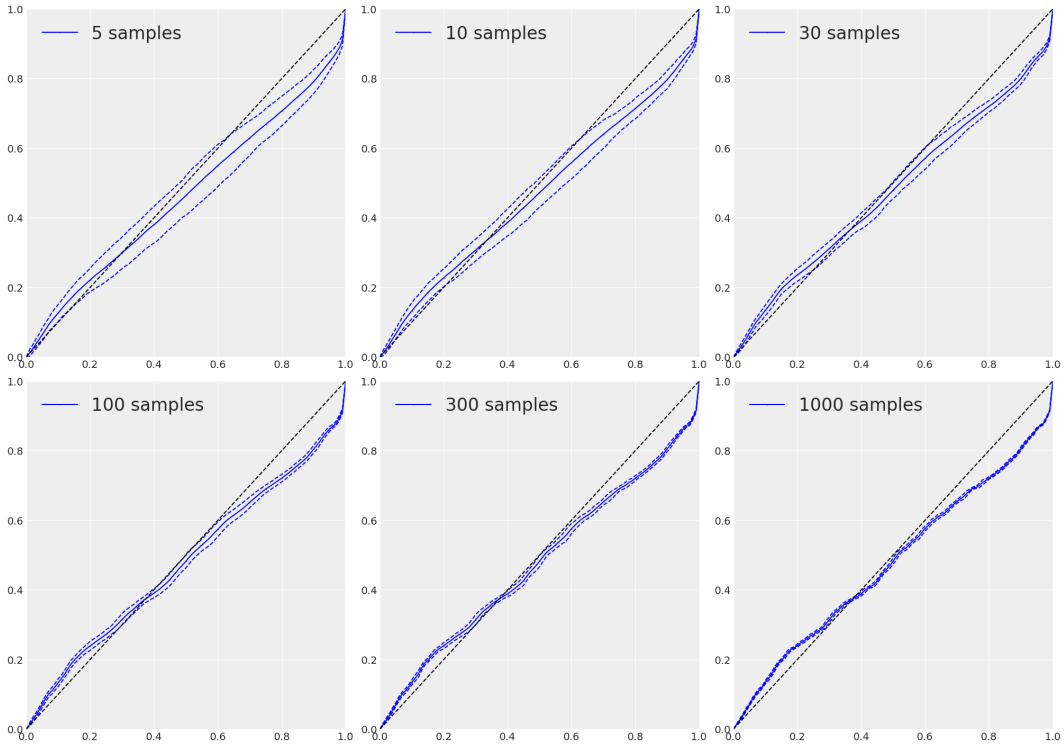

FIGURE 3.2: Mean and 95% confidence interval for the coverage probabilities evaluated by averaging predictive means and variances over a varying number of BNN samples - BERP dataset.

distribution (3.3) of the BNN is computed with a Monte Carlo estimation, thus there is a trade-off between the number of samples and the accuracy of the predictive distribution. From this study, one can imagine that keeping only $W \simeq 100$ samples should be enough to obtain reliable uncertainty quantification.

Let us now turn our attention to models trained with the SILENE dataset, which has fewer data. We will not provide all the performance metrics here, as they can be found in Appendix E. However, we would like to highlight the difference in epistemic uncertainty in the models. Let us look once more at the mean and deviation of the coverage probabilities when averaging over the posterior samples. For the SILENE dataset, we obtain the results shown in Figure 3.3. These were obtained with the **LKJ-BNN** model as well. The spread of the coverage probabilities is much more important for the SILENE dataset. Even with $W = 1000$ posterior samples, the deviation is non-negligible and is close to that of the BERP models with $W = 100$. To use BNN surrogate models for the SILENE test case, more posterior samples are required.

Now that we have already obtained some initial BNN surrogates, we will discuss the possibility of improving them with either multi-fidelity modeling or heteroscedasticity modeling (or both).

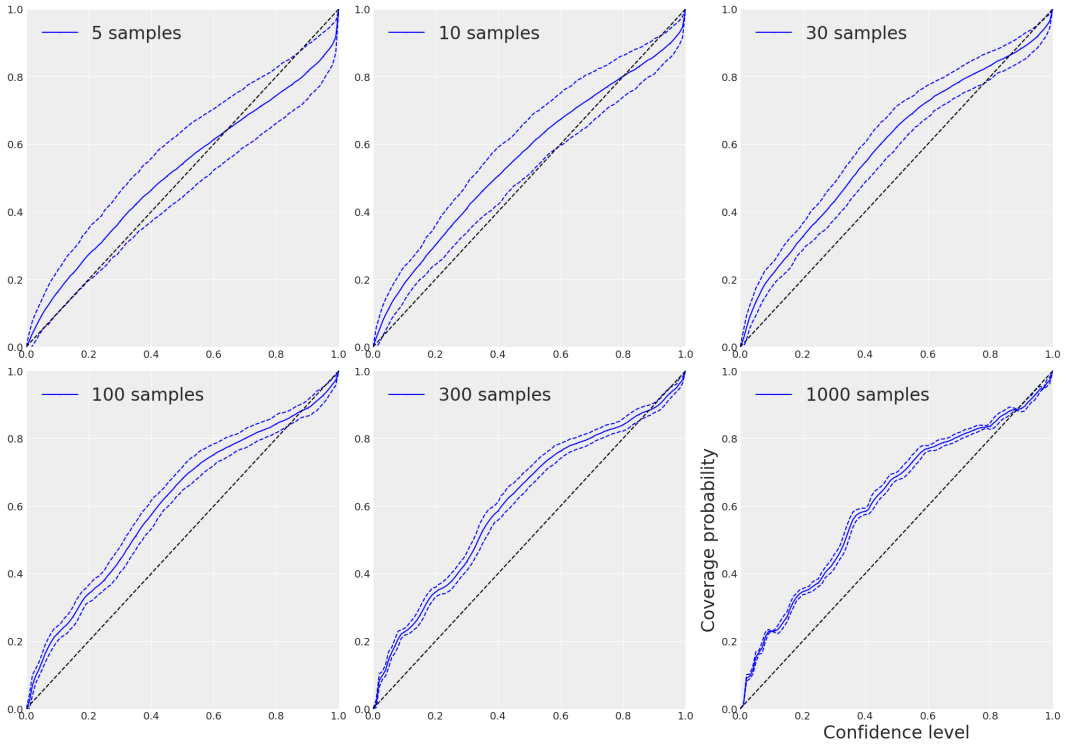

FIGURE 3.3: Mean and 95% confidence interval for the coverage probabilities evaluated by averaging predictive means and variances over a varying number of BNN samples - SILENE dataset.

3.2.5 Improving the BNN

3.2.5.1 Heteroscedastic covariance estimation

As a first improvement, we are looking at heteroscedastic modeling of the predictive covariance, such that for a given set of parameters $\boldsymbol{\phi}$ and an input $x \in \mathcal{X}$, the network outputs a predictive distribution $\mathcal{N}\left(N_{\boldsymbol{\phi}}(x), \mathbf{C}_{\boldsymbol{\phi}}(x)\right)$. For this, we decided to modify the architecture of the network. Starting from the inputs, we keep two dense fully-connected layers of 50 neurons. Then, the network is split into two paths. One path is dedicated to the mean prediction, while the other is focused on the covariance prediction. The mean path is composed of an additional hidden layer of $K_\mu = 50$ neurons, while the covariance path is made of one layer of $K_\sigma = 50$ neurons. This new model is named **HBNN**. We keep a common network part since we expect to have common information between the predictive means and covariances. With this architecture, the total number of model parameters is extended to 5963. The architecture is illustrated in Figure 3.4. The neuron values in the mean layer are given by:

$$z^{(\mu,1)} = h^{(\mu,1)}\left(w^{(\mu,1)} z^{(L)} + b^{(\mu,1)}\right) \tag{3.19}$$

with $L = 2$, and the mean output is:

$$N_{\boldsymbol{\phi}}(x) = h^{(\mu,2)}\left(w^{(\mu,2)} z^{(\mu,1)} + b^{(\mu,2)}\right). \tag{3.20}$$

Similarly to what was done for heteroscedastic GP regression in the previous chapter, we do not learn directly the covariance $\mathbf{C}_{\boldsymbol{\phi}}(x)$ for a given input $x \in \mathcal{X}$ but instead we

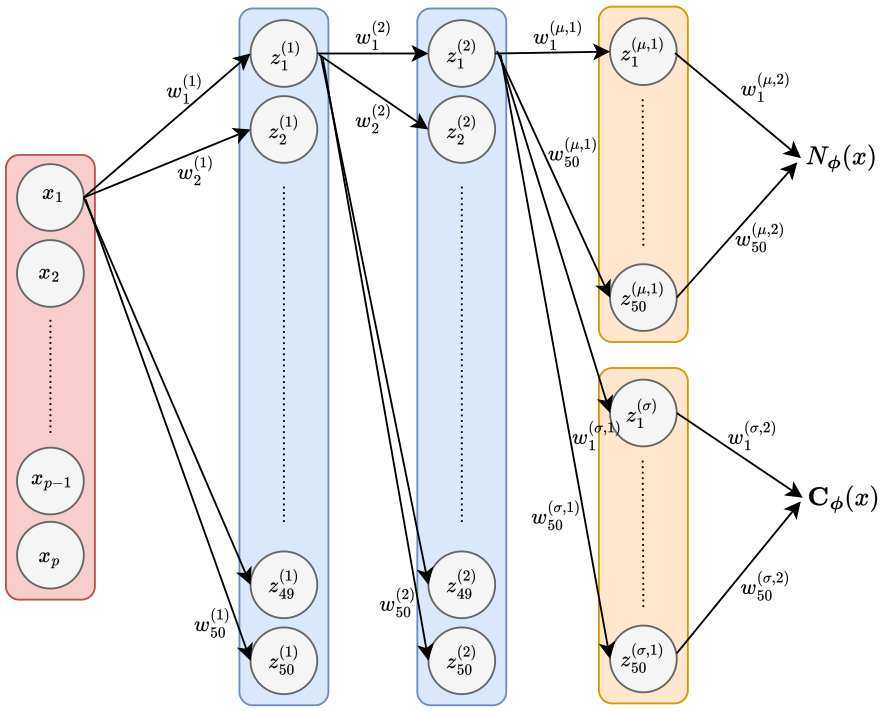

FIGURE 3.4: Simplified architecture of the heteroscedastic BNN.

focus on the Cholesky decomposition $\mathbf{C}_{\boldsymbol{\phi}}(x) = \mathbf{L}_{\boldsymbol{\phi}}(x)^T \mathbf{L}_{\boldsymbol{\phi}}(x)$ and we apply the log to the diagonal elements of $\mathbf{L}_{\boldsymbol{\phi}}(x)$.
More formally, let us introduce the vector $\sigma_{\boldsymbol{\phi}}(x) \in \mathbb{R}^{D(D+1)/2}$ linked to the predictive covariance by:

$$\begin{cases} \sigma_{\boldsymbol{\phi}}(x)_{j+\frac{i(i-1)}{2}} = \log \mathbf{L}_{\boldsymbol{\phi}}(x)_{i,j} & \text{if } i = j \\ \sigma_{\boldsymbol{\phi}}(x)_{j+\frac{i(i-1)}{2}} = \mathbf{L}_{\boldsymbol{\phi}}(x)_{i,j} & \text{if i > j} \end{cases} \tag{3.21}$$

for $1 \leq j \leq i \leq D$. The notations for the covariance layer are similar to the mean layer, and the predictive covariance is obtained by:

$$z^{(\sigma,1)} = h^{(\sigma,1)}\left(w^{(\sigma,1)} z^{(L)} + b^{(\sigma,1)}\right) \tag{3.22}$$

$$\sigma_{\boldsymbol{\phi}}(x) = h^{(\sigma,2)}\left(w^{(\sigma,2)} z^{(\sigma,1)} + b^{(\sigma,2)}\right). \tag{3.23}$$

The training and inference procedures for this new model are very similar. The posterior distribution of the network parameters is obtained by MCMC sampling with HMC-NUTS.

3.2.5.2 Multi-fidelity BNN

We then focus on including a low-fidelity model, namely the point model defined by (1.99), (1.100) and (1.101), to our BNN models. Using the inputs of the model, the predicted point model outputs are evaluated, assuming the nuclear data for the Diven factors and mean neutron multiplicity are known.

Denoting by $f_p : \mathbb{R}^4 \to \mathbb{R}^3$ the point model, we modify the architecture of the network as follows. We introduce a new weight matrix $w^{(\mathrm{pm})} \in \mathbb{R}^{D \times D}$ and define the model

mean output as:

$$N_{\boldsymbol{\phi}}(x) = h^{(\mu,2)}\left(w^{(\mu,2)}z^{(\mu,1)} + b^{(\mu,2)}\right) + w^{(\mathrm{pm})}f_p(x). \tag{3.24}$$

The notation $f_p(x)$ is abusive since f_p is defined over $\mathbb{R}^4$ and not $\mathbb{R}^7$. It must be understood as $f_p(x) = f_p(x_1, x_2, x_3, x_4)$. This new model, which also comes with a heteroscedastic architecture, is named **BL-HBNN**.

3.2.5.3 Mixed GP-BNN models

Since GP surrogate models are readily available for the problem investigated, one could try to leverage the information brought by these models to improve the BNN surrogate models.

The predictive means and covariances of the GP surrogates can be introduced in the network as was done in the previous section for the point model. Consider a GP surrogate model with predictive distribution $f_s(x) \sim \mathcal{N}\left(\overline{f_s(x)}, \mathbf{C}_s(x)\right)$ for $x \in \mathcal{X}$. Let us also define the vector $\sigma_s(x)$ associated with the predictive covariance $\mathbf{C}_s(x)$ of the GP which is defined in the same way as $\sigma_{\boldsymbol{\phi}}(x)$ in (3.21). Keeping the same notations as in Section 3.2.5.1, we define a new model **GP-HBNN** by:

$$N_{\boldsymbol{\phi}}(x) = h^{(\mu,2)}\left(w^{(\mu,2)}z^{(\mu,1)}(x) + b^{(\mu,2)}\right) + w^{(\mu,\mathrm{GP})}\overline{f_s(x)} \tag{3.25}$$

$$\sigma_{\boldsymbol{\phi}}(x) = h^{(\sigma,2)}\left(w^{(\sigma,2)}z^{(\sigma,1)}(x) + b^{(\sigma,2)}\right) + w^{(\sigma,\mathrm{GP})}\sigma_s(x) \tag{3.26}$$

where $w^{(\mu,\mathrm{GP})}$ and $w^{(\sigma,\mathrm{GP})}$ are two newly introduced weight matrices of respective size $D \times D$ and $\frac{D(D+1)}{2} \times \frac{D(D+1)}{2}$.

To include the GP information, it is also possible to use a Gauss-Hermite quadrature as was done in [KCG24] to sample the posterior distribution from the GP instead of feeding only its mean and covariance to the BNN. However, this method leads to an augmented dataset and also increases the number of network parameters. It was not viable for our problem. Indeed, the BNN surrogates are then used in the subsequent MCMC sampling, and we must keep fast prediction times, which was not the case with this approach.

3.2.5.4 Performance of the improved models

What are the improvements brought by the heteroscedasticity modeling and the multi-fidelity approaches? The error metrics for the predictive means are shown in Table 3.2. Looking at the predictive means one can highlight the **BL-HBNN** model which has very low predictive errors. The **GP-HBNN** model also provides excellent predictive performance for R and Y_∞ but seems to fall short for X_∞.

Now let us shift our focus to the coverage probabilities. One can obtain coverage probabilities by averaging the predictive means $N_{\boldsymbol{\phi}_m}(x)$ and covariances $\mathbf{C}_{\boldsymbol{\phi}_m}(x)$ obtained over the set of samples $(\boldsymbol{\phi}_m)_{1\leq m\leq M}$. The averaged coverage probabilities are plotted in Figure 3.5. From this Figure, one can see that the heteroscedastic model **HBNN** does not yield excellent coverage probabilities on its own. However, adding the point model largely improves the coverage probabilities. Moreover, we see that

TABLE 3.2: Performance of the heteroscedastic BNN models - BERP dataset.

HBNN	NMAE	NRMSE	Q^2
R	0.0085	0.0143	0.9996
Y_∞	0.0206	0.0392	0.9986
X_∞	0.0645	0.2169	0.9934
BL-HBNN	NMAE	NRMSE	Q^2
R	0.0074	0.0122	0.9997
Y_∞	0.0202	**0.0372**	**0.9988**
X_∞	**0.0644**	**0.2071**	**0.9940**
GP-HBNN	NMAE	NRMSE	Q^2
R	**0.0057**	**0.0097**	**0.9998**
Y_∞	**0.0182**	0.0439	0.9983
X_∞	0.090	0.4628	0.9699

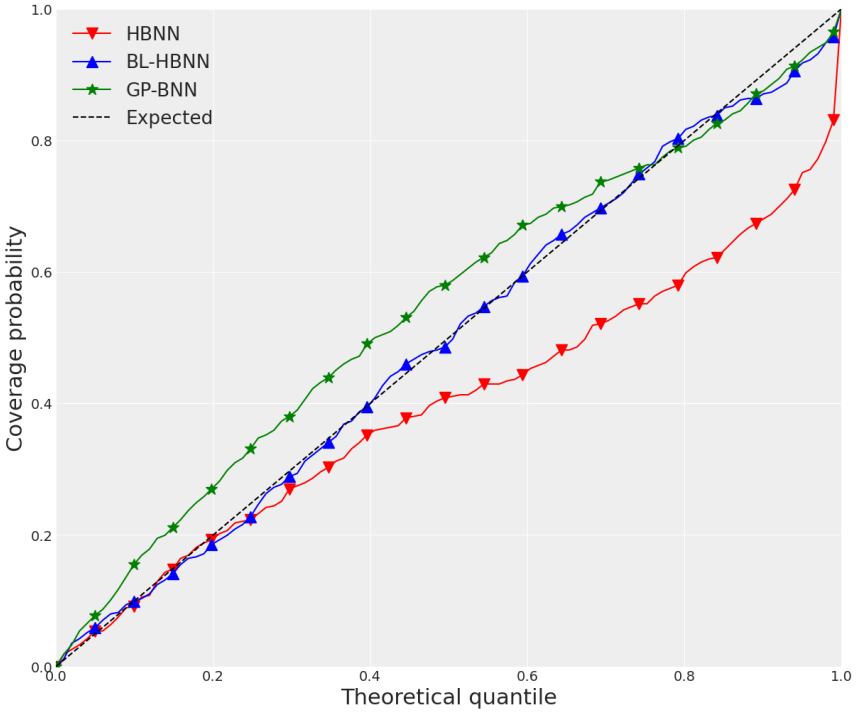

FIGURE 3.5: Averaged coverage probabilities for the multi-fidelity and heteroscedastic BNN models.

the **GP-HBNN** also benefits from the added information from the GP predictions, as it provides more reliable uncertainties than **HBNN**.

From Figure 3.5 and Table 3.2, we conclude that our best BNN surrogate is **BL-HBNN**. It comes with a cost as it comes with approximately twice the number of parameters of simpler models such as **LKJ-BNN**.

3.3 Inverse problems with BNN surrogates

In this section, we will see how one can include the BNN surrogate models in the resolution of a Bayesian inverse problem. Two approaches are studied: we begin by providing the predictive averaged quantities to the MCMC sampler, which represents a simplification of the true predictive distribution at the cost of a loss of information. Then, in a second approach, we consider a more accurate likelihood which uses the whole BNN predictive distribution seen as a Gaussian mixture.
The notations are the same as in Chapter 2. The observations are $\mathbf{y} = (y^{(k)})_{1\leq k\leq N}$ and have a Gaussian observational noise with covariance $\mathbf{C}_{\text{obs}}$.

3.3.1 Predictive-averaged BNN likelihood

Consider a BNN surrogate model, for which we have access to M samples of the network parameters $(\boldsymbol{\phi}_m)_{1\leq m\leq M}$. We assume here that we have a heteroscedastic BNN model such that the predictive distribution obtained for one set of parameters $\boldsymbol{\phi}_m$ and an input $x \in \mathcal{X}$ is $f_{\text{NN}}(x) \sim \mathcal{N}\left(N_{\boldsymbol{\phi}_m}(x), \mathbf{C}_{\boldsymbol{\phi}_m}(x)\right)$. The predictive distribution of the BNN can be approximated by Monte Carlo which yields (3.5). Instead of working with this distribution, one can average the predictive means and covariances over the samples, which outputs once more a Gaussian distribution $\overline{f_{\text{NN}}(x)}$:

$$\overline{f_{\text{NN}}(x)} \sim \mathcal{N}\left(\overline{N_{\boldsymbol{\phi}}(x)}, \overline{\mathbf{C}_{\boldsymbol{\phi}}(x)}\right) \tag{3.27}$$

where $\overline{N_{\boldsymbol{\phi}}(x)} = \frac{1}{M}\sum_{m=1}^{M} N_{\boldsymbol{\phi}_m}(x)$ and $\overline{\mathbf{C}_{\boldsymbol{\phi}}(x)} = \frac{1}{M}\sum_{m=1}^{M} \mathbf{C}_{\boldsymbol{\phi}_m}(x)$. We can then use the following statistical model:

$$y^{(k)}|x = \overline{f_{\text{NN}}(x)} + \varepsilon^{(k)} \tag{3.28}$$

where $\varepsilon^{(k)} \sim \mathcal{N}(\mathbf{0}, \mathbf{C}_{\text{obs}})$ are the iid random variables corresponding to the observational noise.
This model is the same as for the Gaussian process surrogate models. The corresponding likelihood, denoted by $\overline{L(\mathbf{y}|x)}$ is thus given by:

$$\begin{aligned}\overline{L(\mathbf{y}|x)} = &\frac{1}{\sqrt{(2\pi)^D |N\overline{\mathbf{C}_{\boldsymbol{\phi}}(x)} + \mathbf{C}_{\text{obs}}|}} \\ &\times \exp\left(-\frac{1}{2}\left(\overline{y} - \overline{N_{\boldsymbol{\phi}}(x)}\right)^T \left(\overline{\mathbf{C}_{\boldsymbol{\phi}}(x)} + \frac{1}{N}C_{\text{obs}}\right)^{-1} \left(\overline{y} - \overline{N_{\boldsymbol{\phi}}(x)}\right)\right).\end{aligned} \tag{3.29}$$

3.3.2 Gaussian mixture likelihood

Instead of working with the averaged predictive quantities, one could try to use the full BNN output distribution to reflect more accurately the uncertainties in the inverse problem.
This time, we assume that the observations are given by the Monte Carlo approximation of the predictive distribution (3.3).

$$y^{(k)}|x = Y|x + \varepsilon^{(k)} \tag{3.30}$$

where $\varepsilon^{(k)}$ are the same as in the previous section, and where for a given input $x \in \mathcal{X}$:

$$p(Y|\mathcal{D}_{\text{train}}, x) \simeq \frac{1}{M}\sum_{m=1}^{M} \frac{\exp\left(-\frac{1}{2}(Y - N_{\boldsymbol{\phi}_m}(x))^T \mathbf{C}_{\boldsymbol{\phi}_m}(x)^{-1}(Y - N_{\boldsymbol{\phi}_m}(x))\right)}{\sqrt{(2\pi)^D |\mathbf{C}_{\boldsymbol{\phi}_m}(x)|}}. \tag{3.31}$$

From this statistical model, one can derive a likelihood for the inverse problem with observations $\mathbf{y}$.

Proposition 3.3.1. *The likelihood of the observations* $\mathbf{y}$ *is given by:*

$$L(\mathbf{y}|x) = \frac{1}{M}\sum_{m=1}^{M} \mathcal{N}\left(\mathbf{y}_{\text{flat}}|\mathbf{N}_{\boldsymbol{\phi}_m}(x), \mathbf{C}_{\text{tot}}^{(m)}(x)\right) \tag{3.32}$$

where we introduced:

$$\mathbf{y}_{\text{flat}} = (y_1^{(1)}, ...y_D^{(1)}, ..., y_1^{(N)}, ..., y_D^{(N)})^T \in \mathbb{R}^{DN} \tag{3.33}$$

$$\mathbf{N}_{\boldsymbol{\phi}_m}(x) = \left(N_{\boldsymbol{\phi}_m}(x), ..., N_{\boldsymbol{\phi}_m}(x)\right)^T \in \mathbb{R}^{DN}. \tag{3.34}$$

and where $\mathbf{C}_{\text{tot}}^{(m)}(x) = \mathcal{U}_N \otimes \mathbf{C}_{\boldsymbol{\phi}_m}(x) + \mathcal{I}_N \otimes \mathbf{C}_{\text{obs}} \in S_{DN}^+(\mathbb{R})$. *The notation* $\mathcal{N}(x|\mu, \boldsymbol{\Sigma})$ *refers to the density of the multivariate Gaussian distribution with mean* μ *and covariance* $\boldsymbol{\Sigma}$.

Proof. We introduce the flattened vector of the noise terms, defined by $\mathcal{E} = \left(\varepsilon^{(1)}, ..., \varepsilon^{(N)}\right)^T \in \mathbb{R}^{DN}$. Since the $\varepsilon^{(k)}$ are iid, we have $\mathcal{E} \sim \mathcal{N}(\mathbf{0}, \mathcal{I}_N \otimes \mathbf{C}_{\text{obs}})$.
Let us first write the distribution of $\mathbf{Y} = (Y, ..., Y) \in \mathbb{R}^{DN}$. The distribution of $\mathbf{Y}$ is given by the likelihood:

$$p(\mathbf{Y}|\mathcal{D}_{\text{train}}, x) = \frac{1}{M}\sum_{m=1}^{M} \mathcal{N}\left(\mathbf{Y}|\mathbf{N}_{\boldsymbol{\phi}_m}(x), \mathcal{U}_N \otimes \mathbf{C}_{\boldsymbol{\phi}_m}(x)\right). \tag{3.35}$$

We know that $\mathbf{y}_{\text{flat}} = \mathbf{Y} + \mathcal{E}$. Since $\mathbf{Y}$ and $\mathcal{E}$ are independent random variables with known densities, their sum has a density given by the convolution product:

$$\begin{aligned} p(\mathbf{y}|\mathcal{D}_{\text{train}}, x) &= \int p(\mathbf{Y}|\mathcal{D}_{\text{train}}, x)\mathcal{N}(\mathbf{y}_{\text{flat}} - \mathbf{Y}|\mathbf{0}, \mathcal{I}_N \otimes \mathbf{C}_{\text{obs}})\, d\mathbf{Y} \\ &= \frac{1}{M}\sum_{m=1}^{M} \int \mathcal{N}\left(\mathbf{Y}|\mathbf{N}_{\boldsymbol{\phi}_m}(x), \mathcal{U}_N \otimes \mathbf{C}_{\boldsymbol{\phi}_m}(x)\right) \mathcal{N}(\mathbf{y}_{\text{flat}} - \mathbf{Y}|\mathbf{0}, \mathcal{I}_N \otimes \mathbf{C}_{\text{obs}})\, d\mathbf{Y} \\ &= \frac{1}{M}\sum_{m=1}^{M} \mathcal{N}\left(\mathbf{y}_{\text{flat}}|\mathbf{N}_{\boldsymbol{\phi}_m}(x), \mathbf{C}_{\text{tot}}^{(m)}(x)\right) \end{aligned} \tag{3.36}$$

where we used the fact that the convolution of Gaussian densities is a Gaussian density with mean (resp. covariance) being the sum of the elements means (resp. covariances). □

With this likelihood, one can solve the inverse problem whilst using the full extent of the predictive capabilities of BNN surrogate models. Of course, this comes with a cost since the likelihood is much more computationally expensive than that of GP surrogate models or BNN models with the averaged likelihood (3.29). Such a method is not necessarily suited for applications where the computational budget is limited but may be of service when time is not as much of a constraint.

3.4 Applications

Let us consider our application cases and investigate the benefits of BNN surrogate models compared to GP models.

3.4.1 ICSBEP Benchmark

Consider first the BERP test case. The context is the same as in Chapter 2.
To have an idea of the required number M of network samples $(\boldsymbol{\phi}_m)_{1\leq m\leq M}$ to properly represent the BNN posterior, we may consider the analysis shown in Figure 3.2. From these results, it seems reasonable to keep only $W = 100$ samples for the predictions. For the test cases, we generated Markov chains with MCMC with a length of $L = 3 \times 10^5$ because of the more computationally demanding likelihood involved.

To study the two likelihoods proposed in this chapter, we consider two posterior distributions obtained with the same model (**LKJ-BNN** in this case). The first one, shown in Figure 3.6 is obtained from the averaged likelihood (3.29). The second one is obtained from the Gaussian mixture likelihood (3.32) and is displayed in Figure 3.7.

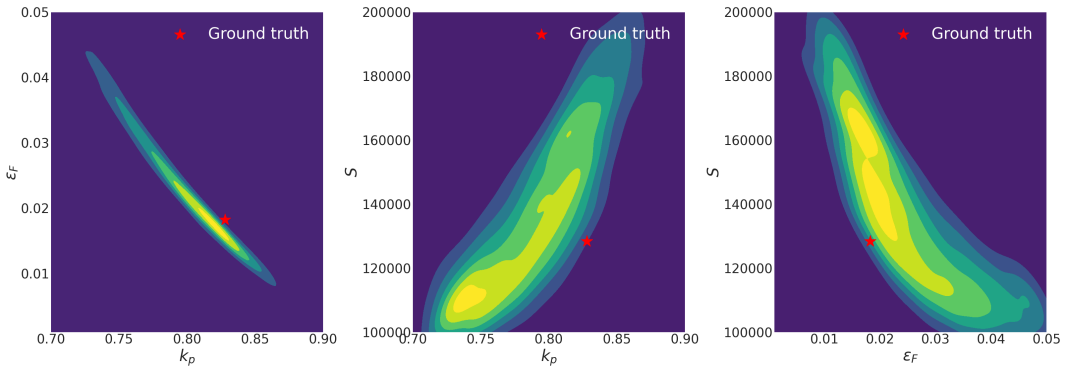

FIGURE 3.6: Posterior distribution marginals for the averaged likelihood (3.29) and the model **LKJ-BNN** with $W = 100$ - BERP test case.

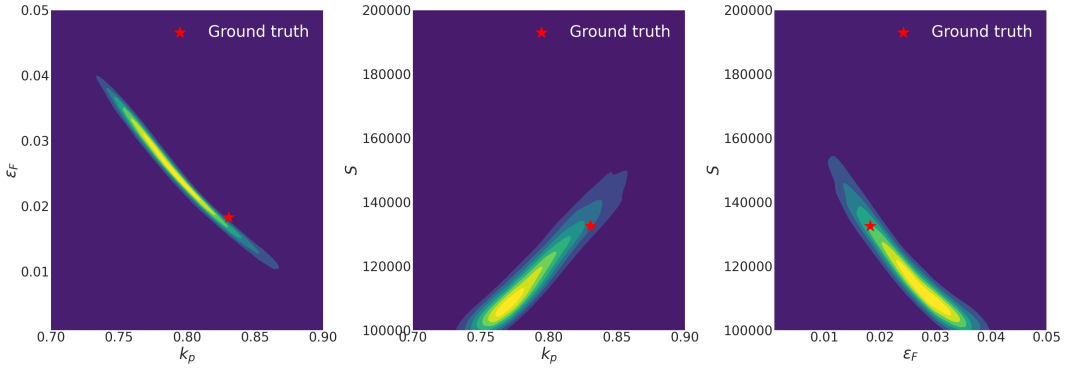

FIGURE 3.7: Posterior distribution marginals obtained with the mixture likelihood (3.32) and the model **LKJ-BNN** with $W = 100$ - BERP test case.

The first thing that stands out is the large overestimation of the uncertainties displayed by the averaged likelihood sampling. Compared to the distributions obtained in

Chapter 2 with GP surrogates, Figure 3.6 displays a much broader distribution. On the other hand, the mixture likelihood leads to a distribution closer to the ones obtained in the previous chapter, though the distribution is narrower.
We then show the two same posterior distributions, but this time the surrogate model used is the heteroscedastic model **BL-HBNN**. The posteriors are plotted in Figures 3.8 and 3.9.
Once again the averaged likelihood seems quite inaccurate when compared to the

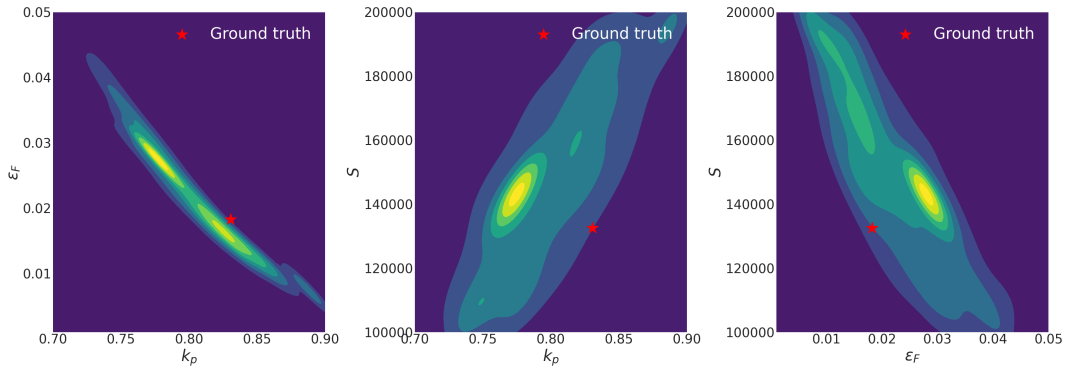

FIGURE 3.8: Posterior distribution marginals obtained with the averaged likelihood (3.29) and the model **BL-HBNN** with $W = 100$ - BERP test case.

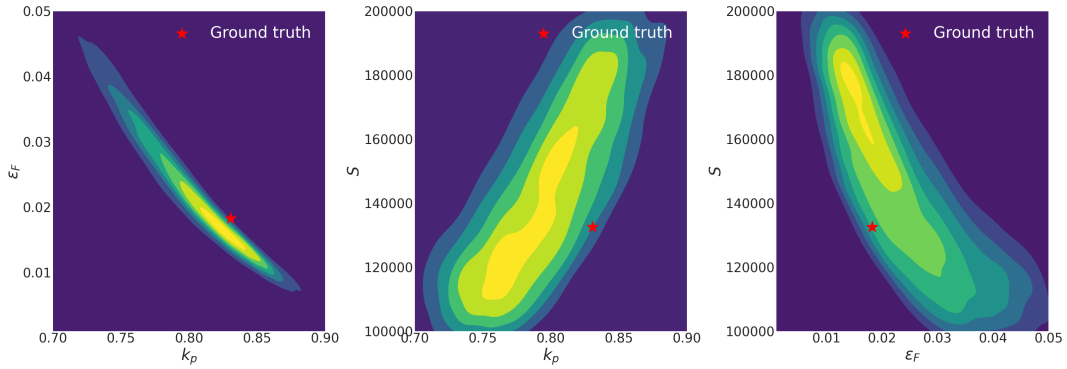

FIGURE 3.9: Posterior distribution marginals obtained with the mixture likelihood (3.32) and the model **BL-HBNN** with $W = 100$ - BERP test case.

posterior distributions obtained with the GP surrogates. However, for the mixture likelihood, the posterior distribution is broader than for the homoscedastic models. There are two possible explanations for this phenomenon. Either, the homoscedastic models fail to capture some of the uncertainties, or the heteroscedastic model is not well-trained and displays too much epistemic uncertainty compared to the other models. To investigate this problem, we looked at the histogram plot of the log variances of the marginal posterior distributions of the network parameters. This can be seen in Figure 3.10. One can see that the log variances tend to be higher for the heteroscedastic model which points towards a larger epistemic uncertainty. The good news is that this uncertainty may be reduced by feeding additional data to the training set $\mathcal{D}_{\text{train}}$ of the model.

Finally, to conclude this study we explore the influence of the number of samples W. To that end, we sample the posteriors with the simpler **BNN-LKJ** model with

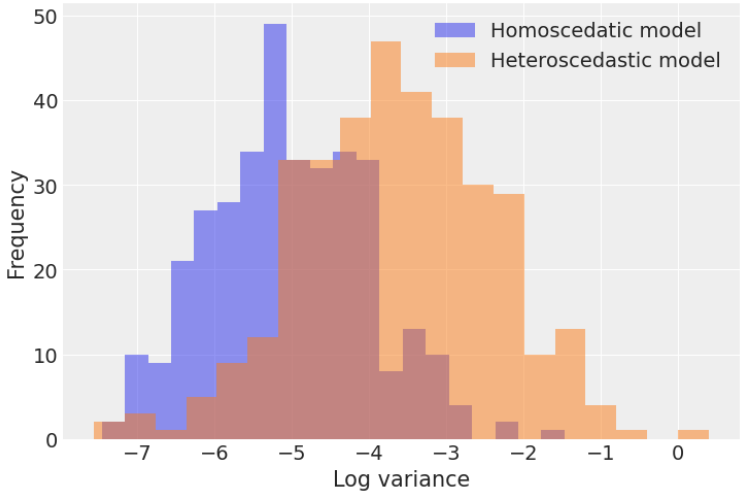

FIGURE 3.10: Histogram plot of the log-variance of the marginal posterior distribution of each network parameter. Comparison between the models **LKJ-BNN** (blue) and **BL-HBNN** (orange).

$W = 10$ and $W = 1000$ with the mixture likelihood. The posteriors obtained are shown in Figures 3.11 and 3.12. One can see that the distributions look very similar

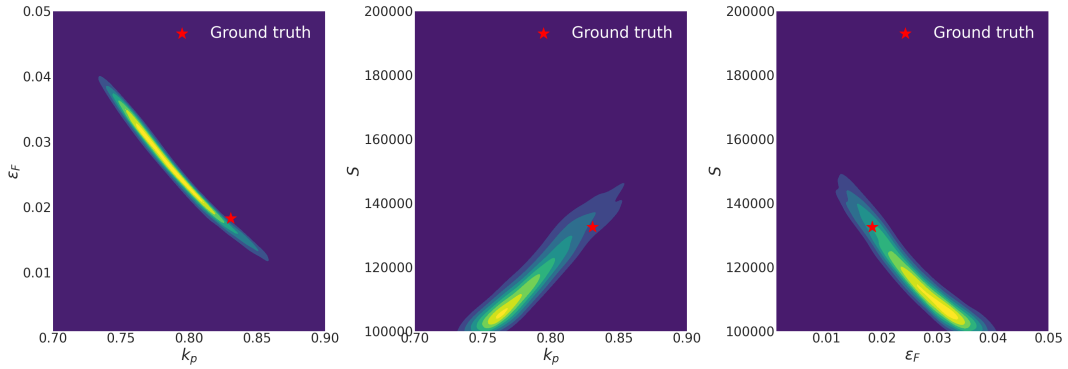

FIGURE 3.11: Posterior distribution marginals obtained with the mixture likelihood (3.32) and the model **BNN-LKJ** with $W = 10$ - BERP test case.

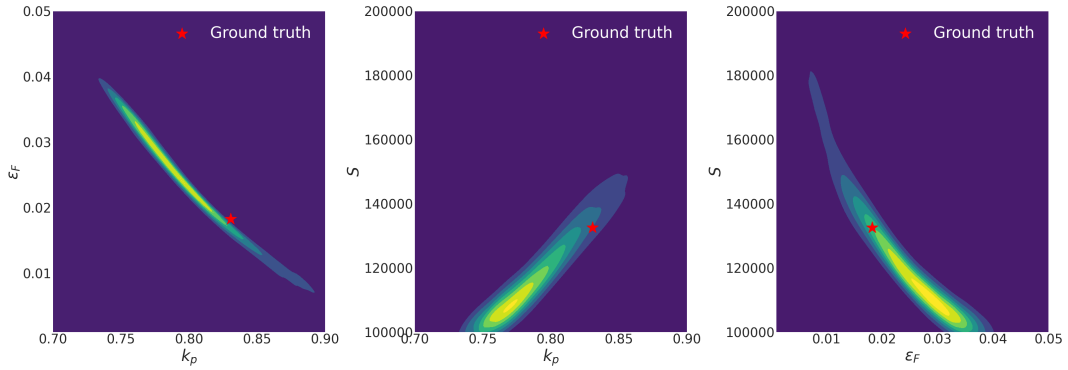

FIGURE 3.12: Posterior distribution marginals obtained with the mixture likelihood (3.32) and the model **BNN-LKJ** with $W = 1000$ - BERP test case.

which may be explained by the large number of data instances for this particular test case. The epistemic uncertainty in the model is small, which means that the posterior distribution over the network parameters is likely to be very narrow. A few samples are enough to represent the distribution accurately. This is corroborated by the previous analysis of the coverage probabilities variability in Figure 3.2.

3.4.2 SILENE test case

Let us conduct a similar study for the SILENE test case. We focus only on the configuration with $h = 20$ cm of fissile solution height. We expect this study to uncover more dependence especially with respect to W since the models were trained with a smaller dataset. Theoretically, they should thus have a larger epistemic uncertainty. This can be highlighted by looking at the log variances of the marginal posteriors on the network parameters. Specifically, we plot in Figure 3.13 the log variances of the **LKJ-BNN** models trained with the SILENE and BERP datasets respectively. As expected, the log variances are in general higher with the SILENE model since fewer data are available. Our first point of focus, compared to the previous test case, is that

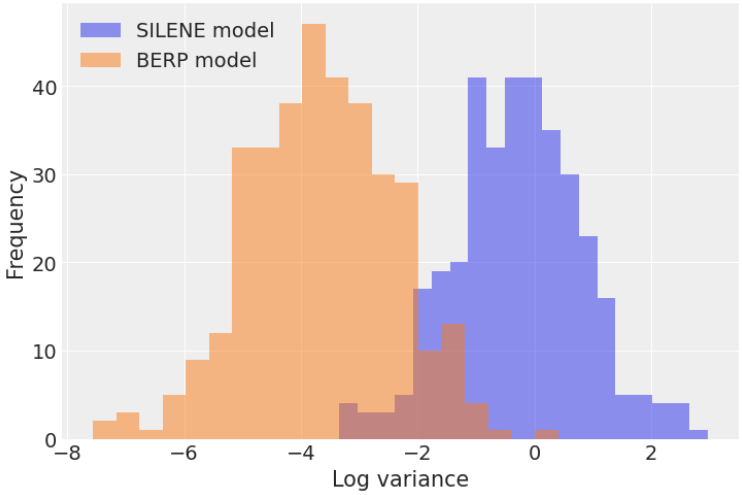

FIGURE 3.13: Histogram plot of the log-variance of the marginal posterior distribution of each weight. Comparison between the models **LKJ-BNN** trained on the SILENE (blue) and BERP (orange) datasets.

the averaged likelihood requires a lot of samples W. In practice, for $W < 1000$, the likelihood is too volatile and its high variability prevents sampling with MCMC. Even with $W = 1000$ samples, one can see in Figure 3.14, that the posterior distribution is very inaccurate. The MCMC sampling struggles to explore the distribution. We conclude that this averaged likelihood is not suited for problems with large epistemic uncertainty.

We then investigate the influence of W by considering the posteriors obtained with **LKJ-BNN** with varying choices of W, with the mixture likelihood. The posterior distributions obtained respectively for $W = 100$ and $W = 1000$ are shown in Figures 3.15 and 3.16. For the same reason as before, we need to take at least $W > 100$ to limit the volatility of the likelihood. Compared to the BERP case, one can see the stronger influence of W. For $W = 100$, the distribution is more spread out (especially for the source intensity S).

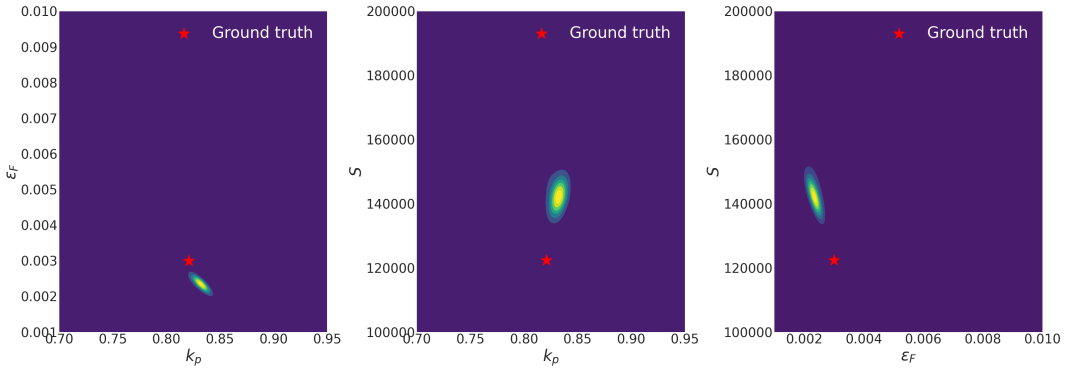

FIGURE 3.14: Posterior distribution marginals obtained with the averaged likelihood (3.29) and the model **BNN-LKJ** with $W = 1000$ - SILENE test case.

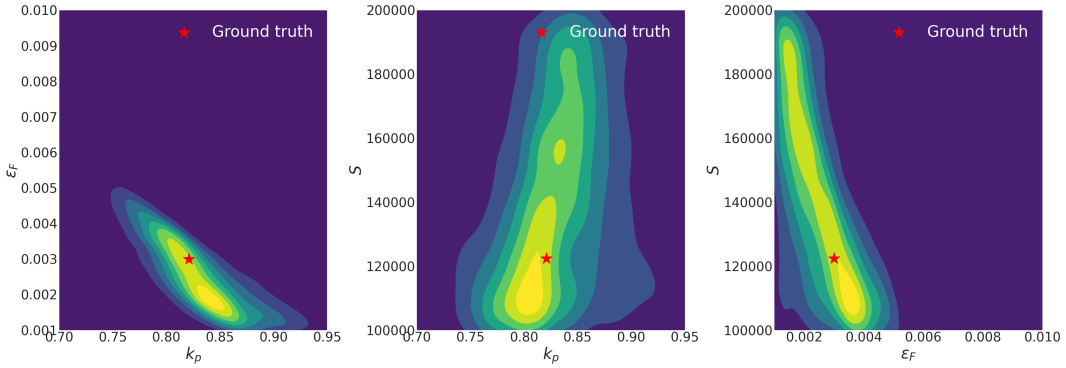

FIGURE 3.15: Posterior distribution marginals obtained with the mixture likelihood (3.32) and the model **BNN-LKJ** with $W = 100$ - SILENE test case.

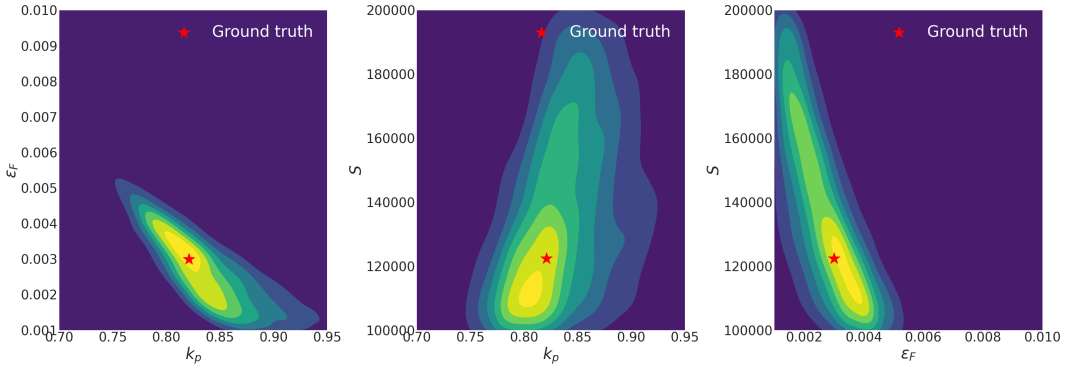

FIGURE 3.16: Posterior distribution marginals obtained with the mixture likelihood (3.32) and the model **BNN-LKJ** with $W = 1000$ - SILENE test case.

Finally, we compare the homoscedastic and heteroscedastic models. The posterior marginals obtained with **BL-HBNN** are shown in Figure 3.17 for the mixture likelihood with $W = 100$ samples. This posterior distribution is noticeably narrower than that of Figure 3.15. This fact, combined with the more accurate coverage probabilities of the **BL-HBNN** model seen in Figure 3.5, demonstrates the improvement brought by the heteroscedasticity modeling. We expect the posterior distribution in Figure 3.17 to be more reliable than the one obtained with the **LKJ-BNN** model, which

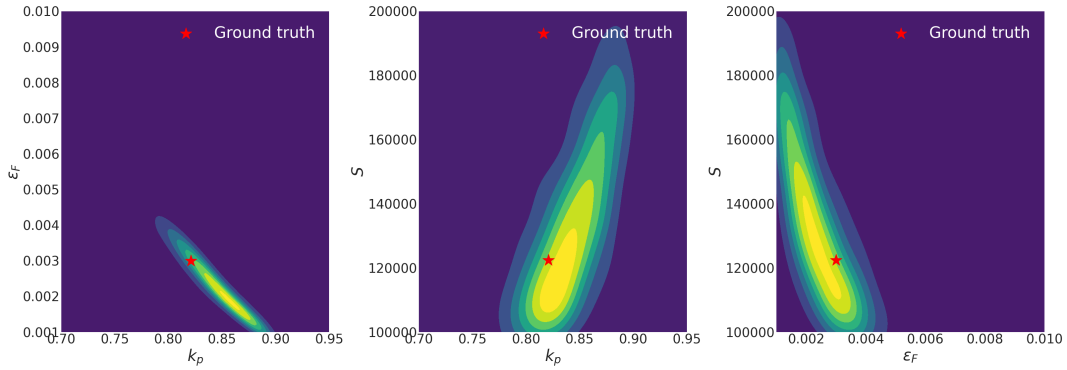

FIGURE 3.17: Posterior distribution marginals obtained with the mixture likelihood (3.32) and the model **BL-HBNN** with $W = 100$ - SILENE test case.

overestimates the uncertainties.

3.5 Conclusion

The BNN surrogates presented in this chapter offer performance comparable to the GP surrogates of Chapter 2. Some small differences may be noted, such as the slight improvements in the coverage probabilities. However, BNNs are more computationally intensive. This fact combined with the cost of the Gaussian mixture likelihood makes the sampling of the posterior distribution significantly slower. As an example, the running times for Adaptive Metropolis sampling for a total of $L = 10^6$ samples with respectively **GP-LMC3** and **BNN-LKJ** were 8 and 15 minutes. This computational cost is further amplified for more elaborate models such as **BL-HBNN**, where the same sampling takes around 55 minutes.

As a general rule, we would advocate for GP surrogates, especially for cases where the posterior distributions must be obtained rapidly. BNN surrogates however may be of interest to provide an additional method to verify the suitability of GPs to a particular problem. They may also be helpful to provide more reliable estimations, at the cost of longer running times. In particular, for our problem, we believe that heteroscedastic BNNs, such as the model **BL-HBNN**, are better at modeling the input-dependent covariance than their GP counterpart. The posterior distributions 3.9 and 3.17 are thought to be more reliable than those obtained with GP surrogates or homoscedastic BNNs.

To conclude on this subject, we would recommend the following practice. GP surrogate models should be used whenever the problem at stake is subject to time constraints. If not, however, BNN surrogate models can serve as reliable alternatives, being superior to GPs for capturing heteroscedasticity for example. One should keep in mind that their construction may be more strenuous to make the most out of the data and low-fidelity models available.

Chapter 4

Sequential designs for Bayesian inverse problems

In this chapter, we will discuss optimal sequential design strategies in the specific case of Bayesian inverse problems. This is motivated by the need for fine-tuning surrogate models for a given inverse problem. An introduction to optimal design strategies is presented in the foreword of this chapter. We then propose two sequential design strategies and apply them to various test cases.

4.1 Overview of optimal design strategies

For high-dimensional problems or in few-data regimes, optimal design strategies seek to provide a set of rules to select design points to maximize the model performance. Throughout this chapter, our guideline is to improve a preexisting surrogate, acting as an emulator for a complex computer code. This problem was first tackled in [Cur+88]. Optimal design strategies have found many applications in Bayesian optimization [Sha+15], importance sampling [DSD13], reliability analysis [LJ08] or in Bayesian model calibration [SPW23]. Sequential design is a subset of optimal design methods in which new design points are added iteratively based on the current state of the surrogate model [Rob52; Wan+14].

4.1.1 Space-filling designs

Space-filling designs are a subclass of optimal designs whose goal is to provide a set of design points forming a good coverage of the whole space. Such designs may be helpful as an initial design set to ensure that the surrogate model performs well on the whole design space. They serve a purpose similar to that of quasi-Monte Carlo methods [Caf98].

4.1.1.1 Latin hypercube

Consider some measurable function f on $[0,1]^p$. Let U be a random variable with uniform distribution on $[0,1]^p$. Suppose that $\mathbb{E}\left[f(U)^2\right] < +\infty$. If we want to estimate $\mathbb{E}\left[f(U)\right]$, the naive approach is to draw n iid samples $(u_i)_{1\leq i\leq n}$ from U and compute the unbiased estimator $\hat{f}_n = \frac{1}{n}\sum_{i=1}^{n} f(u_i)$. The variance of the estimator is given by $\mathbb{V}\left[\hat{f}_n\right] = \frac{1}{n}\mathbb{V}\left[f(U)\right]$. The Latin hypercube sampling [Hus+11] is a method designed

to sample n points s_i which yield an estimator $\hat{f}_{n,\text{lhs}}$ with a lower variance than that of the naive approach with iid samples.

Definition 4.1.1. *Consider the p-dimensional hypercube $[0,1]^p$. Let n be the number of samples to be obtained. A Latin hypercube is given by the matrix $\mathbf{L} \in \mathbb{R}^{n\times p}$ such that each column $L_{\cdot,j}$ is a permutation of the vector $(\frac{n+1}{2} - k)_{1\leq k\leq n}$.*
A Latin hypercube sample of size n of the hypercube $[0,1]^p$, denoted by a matrix $\mathbf{S} \in \mathbb{R}^{n\times p}$ can be obtained from a Latin hypercube $\mathbf{L}$ by the relation $d_{i,j} = \frac{1}{n}\left(l_{i,j} + u_{i,j} + \frac{n-1}{2}\right)$ for $1 \leq i \leq n$ and $1 \leq j \leq p$ where the $u_{i,j}$ are independent random samples of a uniform distribution on $[0,1]$.

In [Ste87], it was shown that the empirical variance estimated from a Latin hypercube sample is lower than that of the standard empirical variance estimator $\hat{f}_n$, assuming f is monotonic w.r.t. each variable (see Theorem 3.1 in [HD03]).

Proposition 4.1.1. *Consider a measurable function f on the hypercube $[0,1]^p$, which is supposed monotonic w.r.t. each input variable. Let U be a random variable with uniform distribution on $[0,1]^p$. We assume that $\mathbb{E}\left[f(U)^2\right] < +\infty$. Let $u = (u_1, ..., u_n)$ be a sample of size n from the distribution U and let $s = (s_1, .., s_n)$ be a Latin hypercube sample on $[0,1]^p$. The corresponding estimators are denoted $\hat{f}_n = \frac{1}{n}\sum_{i=1}^{n} f(u_i)$ and $\hat{f}_{n,\text{lhs}} = \frac{1}{n}\sum_{i=1}^{n} f(s_i)$. We have the following results on the empirical variance.*

$$\mathbb{V}\left[\hat{f}_{n,\text{lhs}}\right] = \mathbb{V}\left[\hat{f}_n\right] - \frac{1}{n}\sum_{j=1}^{p}\mathbb{V}\left[\mathbb{E}\left[f(x)|x_j\right]\right] + o\left(\frac{1}{n}\right). \tag{4.1}$$

The second term in (4.1) quantifies the sensitivity of the function f to the j-th dimension. This property of LHS designs, along with their simplicity, makes them attractive designs when constructing an initial design set for a surrogate model. It is one of the possible approaches which can be easily set up.

4.1.1.2 Minimax and maximin distance

Consider a distance d on the space of interest $\mathcal{X}$. We are interested in building a design set $\mathcal{S} = \{X_1, ..., X_n\} \subset \mathcal{X}^n$ with a finite number n of design points. In [JMY90], two possible strategies were introduced as minimax and maximin design sets.

Definition 4.1.2. *Consider a fixed number of design points n and let us denote the distance between a design set $\mathcal{S} = \{X_1, ..., X_n\} \subset \mathcal{X}^n$ and a given point $X \in \mathcal{X}$ as $d(X, \mathcal{S}) = \min_{X'\in\mathcal{S}} d(X, X')$. A minimax design set $\mathcal{S}^*$ is a design set which verifies:*

$$\max_{X\in\mathcal{X}} d(X, \mathcal{S}^*) = \min_{S}\max_{X\in\mathcal{X}} d(X, \mathcal{S}) \tag{4.2}$$

or in other words, $\mathcal{S}^*$ is a design set that minimizes the maximal distance between a point $X \in \mathcal{X}$ and itself.

A minimax design set can be an alternative to a Latin Hypercube sampling to provide good coverage of the design space $\mathcal{X}$.

Johnson [JMY90] also introduced maximin designs which aim at maximizing the minimal distance between points of the set itself.

Definition 4.1.3. *A maximin design set $\mathcal{S}^*$ is a design set which verifies:*

$$\min_{X,X' \in \mathcal{S}^*} d(X, X') = \max_{S} \min_{X,X' \in \mathcal{S}} d(X, X'). \tag{4.3}$$

A maximin design guarantees that the design points are widely spread out and can thus also provide a good starting design for a surrogate model.

The space-filling designs introduced in this section are amongst the most commonly used strategies for building an initial training dataset, though many other approaches have been introduced. The focus of this thesis is not on space-filling designs however, and the author refers to more comprehensive works on the subject such as [Nie92; San+03; FLS05].

4.1.2 Criterion-based sequential designs

Space-filling designs are widely used to produce an initial design set for a computer model. However, their goal is mainly to provide good coverage of the design space. Goal-oriented designs have been introduced to generate designs suited for specific tasks, whether it is Bayesian optimization, calibration, reliability analysis, or any other application. Contrary to space-filling designs, they are built with a given objective in mind and thus generally outperform them for that specific objective.

In what follows, we give a brief overview of some sequential design strategies. Throughout the rest of this chapter, the direct model f is assumed to be a complex computer code, and we assume that we only have access to an emulator in the form of a GP surrogate.

Let us assume that our surrogate is given by a multi-output zero-mean Gaussian process surrogate f_s. We suppose that we are given a training dataset $(\mathbf{X}, \mathbf{Y})$ of n instances such that the predictive distribution obtained after conditioning is given by (2.14) and written as $f_s^{(n)}(x) \sim \mathcal{N}\left(m_n(x), k_n(x)\right)$ for any $x \in \mathcal{X}$. Our objective is to find a new design point X_{n+1} which is optimal in some sense.

4.1.2.1 D-optimal designs

Among the many interesting properties of a GP model, one can show that the predictive mean $m_n(x)$, also known as the kriging predictor, is the best linear unbiased predictor (BLUP) which minimizes the mean squared error. In [SSW89], design criteria are introduced as functions of the mean squared error (MSE), which boils down to the predictive variance for GP surrogates:

$$\mathrm{MSE}_n(x) = \mathbb{E}\left[\left(f_s^{(n)}(x) - m_n(x)\right)^2\right] = k_n(x). \tag{4.4}$$

Suppose that we can obtain a new data point (X_{n+1}, Y_{n+1}), which is used to update the GP. How should we choose X_{n+1}? This is the fundamental question behind sequential design.

A common criterion in optimal design theory is to minimize the determinant of the Fisher information matrix. This is known as a D-optimal design strategy. This strategy has been extensively used over the years in experimental design [CN80; He09; Rus+09] and has more recently been adapted to computer experiments [AZ10]. With a multi-output GP model, the Fisher matrix is the inverse of the predictive

covariance such that the new design point X_{n+1} in a D-optimal design strategy is chosen as:

$$X_{n+1} \in \text{argmin}_{X \in \mathcal{X}} |\text{MSE}_n(X)|^{-1} = \text{argmax}_{X \in \mathcal{X}} |k_n(X)|. \tag{4.5}$$

For GP surrogate models, D-optimal designs coincide with maximal entropy designs, whose goal is to select design points that maximize the expected information gain [SW87]. Other strategies are designed to minimize the trace of the inverse Fisher matrix (A-optimal), or the maximum MSE (G-optimal). Due to the strong correlations in the predictive covariances, D-optimal appears to be better suited for our purpose.

4.1.2.2 I-optimal designs

Other approaches that have attracted increasing interest are based on the Integrated Mean Squared Prediction Error (IMSPE) instead of the MSE [Pic+10; MW92; Bat+96]. The IMSPE was first considered in [SSW89] and is defined as:

$$\text{IMSPE}_n = \int_{\mathcal{X}} \text{MSE}_n(x) dx. \tag{4.6}$$

The advantage of this strategy is that we shift our focus to the whole space and thus it provides a more global view. More generally one can introduce a weight function w to the IMSPE, s.t. $\int_{\mathcal{X}} w(x)dx = 1$. For Bayesian inverse problems, the weight function is often chosen as the prior density. In that case, IMSPE-based strategies can avoid the pitfall of selecting design points with low prior density. Designs that minimize the IMSPE are known as I-optimal designs.

In the specific context of multi-output GP surrogates, we generalize the IMSPE to:

$$\text{IMSPE}_n = \int_{\mathcal{X}} |k_n(x)| dx. \tag{4.7}$$

The I-optimal sequential design strategy is thus obtained by:

$$X_{n+1} \in \text{argmin}_{X \in \mathcal{X}} \int_{\mathcal{X}} |k_{n+1}(x|X)| dx \tag{4.8}$$

where $k_{n+1}(x|X)$ is the updated predictive covariance obtained by conditioning the GP on the new data instance (X, Y) where Y is the yet unknown output of the black-box function. With GP models, this new predictive covariance $k_{n+1}(\,\cdot\,|X)$ is analytical and only depends on the design point X.

4.1.2.3 Bayes risk minimization

Most of the literature on sequential design for computer experiments is not centered around Bayesian inverse problems. However, [SN17] introduced a sequential design strategy based on the minimization of the Bayes risk of the likelihood estimation. This approach is described in the next paragraphs.

To alleviate the notations, we consider for this section a scalar GP surrogate denoted by f_s modeling a black-box function f. The method generalizes to multi-output GP. Let $\mathbf{X} = (X_1, ..., X_n)$ and $\mathbf{Y} = (Y_1, ..., Y_n)$ be respectively the training inputs and outputs on which the GP is conditioned for $n \geq 1$. The GP obtained after conditioning on $(\mathbf{X}, \mathbf{Y})$ is denoted by $f_s^{(n)}$ and has predictive distribution $f_s^{(n)}(x) \sim \mathcal{N}(m_n(x), k_n(x))$

for $x \in \mathcal{X}$.
Now consider a Bayesian inverse problem for which we have noisy observations $(y^{(k)})_{1\leq k\leq N}$ for $N \geq 1$ which are governed by a Gaussian likelihood defined for $x \in \mathcal{X}$ by:

$$L(\mathbf{y}|x, f) = \prod_{k=1}^{N} \exp\left(-\frac{1}{2\sigma^2}\left(y^{(k)} - f(x)\right)^2\right) \tag{4.9}$$

with σ^2 being the observational variance of the iid noise, and f is the unknown direct model. This likelihood supposes that we know f.
With the GP surrogate, we can build an estimator for the likelihood, which we denote by $\widetilde{L}_n$ and which depends only on the training data $(\mathbf{X}, \mathbf{Y})$. To obtain an estimator for the likelihood, we would like to minimize a loss function between $\widetilde{L}_n$ and the true likelihood $L(\mathbf{y}|\cdot, f)$. Since f is not available, one can replace it with the surrogate $f_s^{(n)}$ which yields a random likelihood $L(\mathbf{y}|\cdot, f_s^{(n)})$ since $f_s^{(n)}$ is a Gaussian process. Taking as a loss function the prior-weighted L^2-norm, an estimate $\widetilde{L}_n$ of the likelihood can be obtained by minimizing the average loss which after inverting the integrals yields the Bayes risk:

$$r_n = \int_{\mathcal{X}} p(x)\mathbb{E}_{f_s^{(n)}}\left(L(\mathbf{y}|x, f_s^{(n)}) - \widetilde{L}_n(\mathbf{y}|x)\right)^2 dx \tag{4.10}$$

where $p(x)$ is the prior density. The best estimator L_n^* is chosen as the estimator which minimizes the Bayes risk (4.10). Since the observations $\mathbf{y}$ are fixed, this amounts to minimizing the integrand. Coincidentally, the best estimator L_n^* is exactly the likelihood (2.76) that is used throughout this thesis (see [SN17] for more details). With L_n^*, it is possible to show that the Bayes risk reduces to:

$$r_n^* = \int_{\mathcal{X}} p(x)\mathrm{Var}_{f_s^{(n)}}\left[L(\mathbf{y}|x, f_s^{(n)})\right] dx. \tag{4.11}$$

Now let us consider a new design point $X_{n+1} \in \mathcal{X}$ and its output $Y_{n+1} \in \mathbb{R}$. The model response Y_{n+1} is assumed to be given by the predictive distribution of the GP surrogate at stage n such that $Y_{n+1} \sim f_s^{(n)}(X_{n+1})$. The Bayes risk at iteration $n+1$, denoted by $r_{n+1}(X_{n+1}, Y_{n+1})$, is thus a random variable. The sequential design strategy introduced in [SN17] proposes to sample the new design point X_{n+1} as the point that minimizes the expectation of the Bayes risk over the whole range of possible model response Y_{n+1}:

$$X_{n+1} \in \mathrm{argmin}_{\mathcal{X}}\mathbb{E}_{Y_{n+1}}\left[r_{n+1}(X_{n+1}, Y_{n+1})\right] \tag{4.12}$$

$$\mathbb{E}_{Y_{n+1}}\left[r_{n+1}(X_{n+1}, Y_{n+1})\right] = \int_{\mathcal{X}} \mathbb{E}_{Y_{n+1}}\left[\mathrm{Var}_{f_s^{(n+1)}}\left[L\left(\mathbf{y}|x, f_s^{(n+1)}\right)\right]\right] p(x)dx \tag{4.13}$$

where $f_s^{(n+1)}$ is the updated GP obtained by conditioning on (X_{n+1}, Y_{n+1}). In this chapter, our new developments will be compared to this approach.

4.1.3 Stepwise uncertainty reduction methods

We now present the Stepwise Uncertainty Reduction (SUR) paradigm, a theoretical framework common to a large number of sequential design strategies, which was first introduced in [VVW09] and extensively discussed in [Bec+12; BBG19; Pic14; Pic15;

Che+14; VVW09]. What follows is a theoretical overview of SUR methods based on the aforementioned works.

4.1.3.1 Theoretical overview

Let $\mathbb{F}$ be a functional space, and $\mathbb{M}$ the set of Gaussian measures on $\mathbb{F}$. From now on, let us focus on the space of continuous real-valued functions. Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space, and $\mathcal{X} \subset \mathbb{R}^p$ compact. In everything that follows, we will consider Gaussian processes with continuous sample paths, defined on $\mathcal{X} \times \Omega$. Such Gaussian processes can be understood as random elements of $\mathbb{F}$ [VTC87; Bog98].

Proposition 4.1.2. *For any Gaussian measure $\nu \in \mathbb{M}$, there exists a Gaussian process with continuous sample paths whose probability distribution is ν [VZ+08]. The corresponding mean and covariance functions are denoted $m_\nu \colon \mathcal{X} \to \mathbb{R}$ and $k_\nu \colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. On the other hand, the probability distribution P^f of any given Gaussian process $f_s \colon \mathcal{X} \times \Omega$ is a Gaussian measure on $\mathbb{F}$, and thus $P^f \in \mathbb{M}$.*

Let us consider a functional $\mathcal{H} \colon \mathbb{M} \to \mathbb{R}_+$ which will serve as the metric of uncertainty that we would like to minimize.

Definition 4.1.4. *Let f_s be a Gaussian process and consider that to any design point $X \in \mathcal{X}$ corresponds a random output given by $Y = f_s(X)$. A sequential design is a collection $(X_n)_{n\geq 1}$ such that for all $n \geq 2$, X_n is $\mathcal{F}_{n-1}$-measurable, with $\mathcal{F}_{n-1}$ the σ-algebra generated by the collection $(X_1, Y_1, ..., X_{n-1}, Y_{n-1})$.*

Let f_s be a Gaussian process and $(X_n)_{n\geq 1}$ a sequential design with corresponding values $(Y_n)_{n\geq 1}$. It can be shown that for any $n \geq 1$, there exists a Gaussian measure denoted as P_n^f, which is the probability distribution of f_s conditioned by $\mathcal{F}_n$. More simply put, conditioning a GP with respect to a finite number of observations still yields a GP. The corresponding mean and covariance function are denoted by m_n and k_n. The Gaussian measure P_n^f can thus be interpreted as a random element of $\mathbb{M}$, and $\mathcal{H}(P_n^f)$ is a positive real-valued random variable.

The goal of the SUR method is to find a sequential design $(X_n)_{n\geq 1}$ which guarantees the almost-sure convergence of the metric of uncertainty towards 0.

$$\mathcal{H}(P_n^f) \xrightarrow[n\to+\infty]{a.s.} 0. \tag{4.14}$$

Prior works on the subject have introduced the supermartingale property for the functional $\mathcal{H}$ which is defined as follows.

Definition 4.1.5. *Consider a metric $\mathcal{H} \colon \mathbb{M} \to \mathbb{R}_+$. The metric $\mathcal{H}$ is said to have the supermartingale property if and only if, for any Gaussian process f_s, the sequence $\mathcal{H}(P_n^f)$ is a $\mathcal{F}_n$-supermartingale or in other words for any $X \in \mathcal{X}$:*

$$\mathbb{E}_{n,X}\left[\mathcal{H}(P_{n+1}^f)\right] \leq \mathcal{H}(P_n^f) \tag{4.15}$$

where $\mathbb{E}_{n,X}$ denotes the conditional expectation with respect to $\mathcal{F}_n$ with $X_{n+1} = X$

To study sequential designs and their convergence, a few more results and definitions must be introduced.

Proposition 4.1.3. *[BBG19] proved that there exists a measurable mapping*

$$\begin{aligned}(\mathcal{X}\times\mathbb{R})^n\times\mathbb{M}&\longrightarrow\mathbb{M}\\(X_1,Y_1,...,X_n,Y_n,\nu)&\longmapsto\nu|(X_1,Y_1,...,X_n,Y_n)\end{aligned}$$

such that for any Gaussian process f_s with probability measure $\nu\in\mathbb{M}$ and any sequential design $(X_1,Y_1,...,X_n,Y_n)$, the measure $\nu|(X_1,Y_1,...,X_n,Y_n)$ is the probability distribution of f_s given $\mathcal{F}_n=\sigma(X_1,Y_1,...,X_n,Y_n)$.

Definition 4.1.6. *If $\mathcal{H}$ has the supermartingale property, for $X\in\mathcal{X}$ let us introduce the functional $\mathcal{J}_X\colon\mathbb{M}\to\mathbb{R}_+$ which is the expectation of the metric of uncertainty when conditioning the GP with respect to a new data point (X,Y) and given by:*

$$\mathcal{J}_X(\nu)=\mathbb{E}_Y\left[\mathcal{H}\left(\nu|(X,Y)\right)\right] \tag{4.16}$$

for $\nu\in\mathbb{M}$ and where $Y\sim\mathcal{N}\left(m_\nu(X),k_\nu(X)\right)$ and $\mathbb{E}_Y$ refers to the expectation with respect to the random output Y.

Definition 4.1.7. *For $\mathcal{H}$ with the supermartingale property, we introduce a functional $\mathcal{G}\colon\mathbb{M}\to\mathbb{R}_+$ defined for $\nu\in\mathbb{M}$ by*

$$\mathcal{G}(\nu)=\sup_{X\in\mathcal{X}}\left(\mathcal{H}(\nu)-\mathcal{J}_X(\nu)\right). \tag{4.17}$$

The sets of zeros of $\mathcal{H}$ and $\mathcal{G}$ are denoted respectively by $\mathbb{Z}_\mathcal{H}$ and $\mathbb{Z}_\mathcal{G}$. The inclusion $\mathbb{Z}_\mathcal{H}\subset\mathbb{Z}_\mathcal{G}$ is always true. Indeed, if $\mathcal{H}(\nu)=0$ then $\mathcal{J}_X(\nu)=0$ for any $X\in\mathcal{X}$ which means that $\mathcal{G}(\nu)=0$.

This functional $\mathcal{G}$ plays an important role in the convergence results obtained for the sequential designs in the SUR paradigm later in this chapter.

Let us conclude this introduction with the SUR strategy itself, which is given iteratively, for any functional $\mathcal{H}$ by the minimization of the expected metric of uncertainty at the next iteration.

Definition 4.1.8. *A SUR sequential design for the functional $\mathcal{H}$ is defined as a sequential design such that for all $n\geq n_0$ with n_0 a given integer:*

$$X_{n+1}\in\operatorname{argmin}_{X\in\mathcal{X}}\left\{\mathbb{E}_{n,X}\left[\mathcal{H}(P^f_{n+1})\right]\right\}. \tag{4.18}$$

From now on, the following notations are introduced:

$$H_n=\mathcal{H}(P^f_n) \tag{4.19}$$

$$J_n(X)=\mathbb{E}_Y\left[\mathcal{H}(P^f_n|(X,Y)\right]=\mathbb{E}_{n,X}\left[\mathcal{H}(P^f_{n+1})\right]. \tag{4.20}$$

4.1.3.2 Standard metrics of uncertainty

The SUR framework has found numerous applications in different fields due to its very general formulation. The two main obstacles when developing a SUR sequential design strategy are the choice of the metric of uncertainty and the tractability of the minimization step (4.18). We provide a brief overview of the potential applications of this paradigm in this section.

In excursion problems, one would like to estimate an excursion set, which is a subset of the state space such that a given black-box function f exceeds a given threshold. Introducing some notations, let $T \in \mathbb{R}$ be a given threshold, let f be the real-valued black-box function, and let $S_T(f)$ be the corresponding excursion set defined as:

$$S_T(f) = \{x \in \mathcal{X} \text{ s.t. } f(x) \geq T\}. \tag{4.21}$$

SUR strategies have been developed for such excursion problems. A method introduced in [Che+14] considers the functional $\mathcal{H}$ defined for any Gaussian measure $\nu \in \mathbb{M}$ by:

$$\mathcal{H}(\nu) = \int_{\mathcal{X}} p_\nu(x)(1 - p_\nu(x))dx \tag{4.22}$$

where $p_\nu(x) = \int_{\mathcal{X}} \mathbb{1}_{f(x)\geq T}\ \nu(df)$. This quantity $p_\nu(x)$ can be understood as the probability that the GP model associated with the Gaussian measure ν exceeds the threshold at x. This metric is known as the integrated Bernoulli variance. Other approaches have been considered such as the variance of excursion volume functional [Bec+12] defined by:

$$\mathcal{H}(\nu) = \int (\alpha(f) - \overline{\alpha_\nu})^2\ \nu(df) \tag{4.23}$$

where $\alpha(f) = \mu(S_T(f))$, μ is the Lebesgue measure, and $\overline{\alpha_\nu} = \int \alpha(f)\nu(df)$.

SUR strategies have also been investigated for Bayesian optimization problems in which the goal is to minimize a black-box function with the help of a GP surrogate. Calls to the black-box function are obtained as a compromise between exploration of the state space to reduce the GP uncertainty and exploitation of the promising points inferred from the GP predictions. For Bayesian optimization problems, the efficient global optimization algorithm (EGO) [JSW98; SLA12] can be interpreted as a SUR sequential design. The goal of Bayesian optimization is to minimize a costly black-box function f by building a GP surrogate model. For such problems, the EGO algorithm was introduced in [JSW98]. It is based on the maximization of the expected improvement. For a given $x \in \mathcal{X}$, the expected improvement is $\mathrm{EI}(x) = \mathbb{E}\left[\min(Y_1, ..., Y_n) - f_s^{(n)}(x)\right]$ which can be computed explicitly since the predictive distribution is Gaussian. The next point can thus be chosen as follows:

$$X_{n+1} \in \operatorname{argmin}_{X\in\mathcal{X}} \mathbb{E}\left[\min(Y_1, ..., Y_n) - f_s^{(n)}(X)\right]. \tag{4.24}$$

Though the derivation is not so straightforward, it was shown in [BBG19] that the EGO algorithm falls under the SUR paradigm with an appropriate functional.

The same can be said about the knowledge gradient policy which is another method used in Bayesian optimization, when noise is present in the black-box function evaluations [FPD08; SFP11; FPD09]. The knowledge-gradient criterion selects the design point that maximizes the quantity $\mathrm{KG}(x)$ defined for $x \in \mathcal{X}$ by:

$$\mathrm{KG}(X) = \mathbb{E}_{n,X}\left[\max_{x\in\mathcal{X}} m_{n+1}(x)\right] - \max_{x\in\mathcal{X}} m_n(x). \tag{4.25}$$

This criterion is equivalent to the SUR criterion associated with the functional:

$$\mathcal{H}(\nu) = \int \max_{x\in\mathcal{X}} f(x)\nu(df) - \max_{x\in\mathcal{X}} m_\nu(x) \tag{4.26}$$

where m_ν is the predictive mean of the GP associated with the measure ν.

In this section, we only provided a few examples of SUR designs, though many others have been developed in recent years (see for example [AL+22; Str+22; BBV11]). More than a simple reformulation, the SUR paradigm allows to derive convergence results which can then apply to a wide variety of problems.

4.1.3.3 Convergence of sequential designs

Let us finish this introduction with the main result used for the proof of convergence of SUR strategies.

In [Bog98; VTC87] the convergence of Gaussian measures is defined as follows:

Definition 4.1.9. *For any sequence of Gaussian measures $(\nu_n)_{n\geq 0}$, we say that the sequence $(\nu_n)_{n\geq 0}$ converges towards the limit measure $\nu_\infty \in \mathbb{M}$ when $(m_{\nu_n})_{n\geq 0}$ converges uniformly in $\mathcal{X}$ towards m_{ν_∞}, and $(k_{\nu_n})_{n\geq 1}$ converges uniformly in $\mathcal{X}\times\mathcal{X}$ towards k_{ν_∞}.*

It was proven in [BBG19] that for any sequential design $(X_n)_{n\geq 1}$ and for any Gaussian process f_s with continuous sample paths, the Gaussian measure P_n^f converges towards a limit measure $P_\infty^f \in \mathbb{M}$. The following convergence theorem was derived in the same work.

Theorem 4.1.1 (Convergence of SUR designs)**.** *Let $\mathcal{H}$ be a non-negative uncertainty functional on $\mathbb{M}$ with the supermartingale property, and let $\mathcal{G}$ defined by (4.17). Consider $(X_n)_{n\geq 1}$ a SUR sequential design for $\mathcal{H}$. If $\mathbb{Z}_\mathcal{H} = \mathbb{Z}_\mathcal{G}$ and $\mathcal{H}(P_n^f) \xrightarrow[n\to+\infty]{a.s.} \mathcal{H}(P_\infty^f)$ and $\mathcal{G}(P_n^f) \xrightarrow[n\to+\infty]{a.s.} \mathcal{G}(P_\infty^f)$, then $H_n = \mathcal{H}(P_n^f) \xrightarrow[n\to+\infty]{a.s.} 0$.*

This theorem is one of the strengths of the SUR paradigm. This framework provides a simple unified methodology to build sequential design strategies and exhibits a convergence theorem that relies only on a few assumptions. In addition, this theorem extends to quasi-SUR sequential designs, as stated by [BBG19]. $(X_n)_{n\in\mathbb{N}}$ is a quasi-SUR sequential design if there exists a sequence $(\varepsilon_n)_{n\in\mathbb{N}}$ of non-negative real numbers such that $\varepsilon_n \xrightarrow[n\to+\infty]{} 0$, and if there exists $n_0 \in \mathbb{N}$ such that $(X_n)_{n\in\mathbb{N}}$ verifies $J_n(X_{n+1}) \leq \inf_{x\in\mathcal{X}} J_n(x) + \varepsilon_n$ for all $n \geq n_0$.
This remark is crucial for numerical applications since there is often no guarantee that the true global minimum is reached in the optimization step for a SUR strategy. The convergence theorem for the quasi-SUR designs is more flexible in that regard and ensures convergence even for numerical applications.

4.2 Sequential designs for Bayesian inverse problems

Before diving into the heart of this chapter, let us emphasize the following notation conventions. In everything that follows, we consider a GP surrogate model f_s. Unless stated explicitly otherwise, f_s is a multi-output GP. We denote by $f_s^{(n)}$ the GP obtained by conditioning on the data $(\mathbf{X}, \mathbf{Y})$, where $\mathbf{X} = (X_1, ..., X_n) \in \mathcal{X}^n$ and $\mathbf{Y} = (Y_1, ..., Y_n) \in \mathbb{R}^{n\times D}$ for some $n \geq 1$. The predictive distribution of the GP for any $x \in \mathcal{X}$ is given by $f_s^{(n)}(x) \sim \mathcal{N}(m_n(x), k_n(x))$ where $m_n(x) \in \mathbb{R}^D$ and $k_n(x)$ is a $D \times D$ positive semi-definite matrix.
We also introduce the predictive covariance between two input points $x, x' \in \mathcal{X}$ as $k_n(x, x')$ which is also obtained from the standard predictive equations (2.14).

Specifically, $k_n(x, x')$ is a $D \times D$ matrix, and for $1 \leq i, j \leq D$, $k_n(x, x')_{i,j}$ is the covariance between the i-th output at x and the j-th output at x'.

Keeping the notations of chapter 2, our goal is to solve an inverse problem and obtain the posterior distribution $p(x|\mathbf{y})$ where $\mathbf{y} = (y^{(k)})_{1 \leq k \leq N}$ for $N \geq 1$. We insist on the fact that N is fixed throughout this chapter. The observations, as per usual, are noisy with iid observational noise with zero-mean and covariance $\mathbf{C}_{\text{obs}}$. The likelihood of the observations is given by (2.78) and is denoted by $L_n(\mathbf{y}|x)$ for $x \in \mathcal{X}$. The dependence in n occurs via the surrogate model $f_s^{(n)}$. Similarly, the posterior distribution associated with this likelihood is denoted by $p_n(x|\mathbf{y})$.

During this chapter, our goal is to develop sequential design strategies to select a new design point $X_{n+1} \in \mathcal{X}$ to improve the surrogate model, in the context of a Bayesian inverse problem. Two different strategies are proposed in this chapter. Our first approach to this problem is adapted from D-optimal designs, while the second one is derived from SUR strategies.

4.2.1 Constraint set query (CSQ)

Intuitively, when adding a new point to the dataset, we would like this point to yield the greatest improvement to the surrogate model. In this case, one could try to choose the points whose predictive variances are the highest, or equivalently in a multi-output context, where the determinant of the predictive covariance is the highest. This is equivalent to a D-optimal design.
However, such points may lie well outside the posterior distribution $p(x|\mathbf{y})$ of the inverse problem and thus may not bring any improvement when solving the inverse problem. Our method aims to look at new data points that are both expected to yield a good improvement to the surrogate, but which also lie in regions of high posterior density.

Consider a multi-output GP surrogate trained with $n \geq 1$ data points. Its predictive covariance at input point x is denoted $k_n(x)$. Intuitively, one could adapt the D-optimal strategy by choosing our next training point $X_{n+1} \in \mathcal{X}$ as the maximizer of the determinant of the covariance on a well-chosen subset $\mathcal{B} \subset \mathcal{X}$.

$$X_{n+1} \in \operatorname{argmax}_{x \in \mathcal{B}} |k_n(x)|. \tag{4.27}$$

How can we choose $\mathcal{B}$? As a first strategy, one could impose the sampled points to be near the maximum-a-posteriori (MAP) defined by:

$$x_m^{(n)} \in \operatorname{argmax}_{x \in \mathcal{X}} p_n(x|\mathbf{y}). \tag{4.28}$$

Then for any $h \in \mathbb{R}_+$ we define the subset $\mathcal{B}_h$ such as:

$$\mathcal{B}_h = \left\{ x \in \mathcal{X} | \log p_n(x_m^{(n)}|\mathbf{y}) - \log p_n(x|\mathbf{y}) \leq h \right\}. \tag{4.29}$$

Note that the MAP needs not be unique for $\mathcal{B}_h$ to be defined. To get an intuition of the influence of h, let us look at the acceptance probability $\alpha(x_1, x_2)$ for a jump from point x_1 to x_2 in the context of Metropolis-Hastings sampling:

$$\alpha(x_1, x_2) = \min \left\{ 1, \frac{p_n(x_2|\mathbf{y})}{p_n(x_1|\mathbf{y})} \right\}. \tag{4.30}$$

Then $\mathcal{B}_h = \left\{x \in \mathcal{X} | \log \alpha(x_m^{(n)}, x) \geq -h\right\} = \left\{x \in \mathcal{X} | \alpha(x_m^{(n)}, x) \geq e^{-h}\right\}$. Thus, e^{-h} for $h \in \mathbb{R}_+$ represents the lowest possible acceptance probability of a Metropolis-Hastings jump from the MAP to $x \in \mathcal{B}_h$. Depending on the choice of h, the new query point X_{n+1} will be chosen quite close to the MAP (if h is close to 0) or is allowed to spread far from the MAP (if h is large).

This constraint set query (CSQ) method constitutes the first simple approach for sequential design in the context of Bayesian inverse problems. This method can be understood as a D-optimal sequential design strategy restricted to a subset of the domain.

4.2.2 IP-SUR methods for scalar GP surrogates

In this section, we develop our inverse problem SUR (IP-SUR) strategy, for scalar GP surrogates. For this section only, we have training outputs $\mathbf{Y} = (Y_1, ..., Y_n) \in \mathbb{R}^n$. This strategy is extended to multi-output Gaussian processes in Section 4.2.3.

4.2.2.1 IP-SUR criterion

Consider a GP surrogate model f_s, conditioned on n pairs input-output. The Gaussian measure obtained after conditioning w.r.t. the training data $(\mathbf{X}, \mathbf{Y})$ is P_n^f and the associated mean and covariance functions are $m_n \colon \mathcal{X} \to \mathbb{R}$ and $k_n \colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. Let X be a new design point and Y the corresponding random output. At stage $n+1$, we condition the GP with the new datapoint (X, Y) which updates the predictive distribution. For $x, x_1, x_2 \in \mathcal{X}$, the updated mean and covariance functions of the newly conditioned GP are:

$$m_{n+1}(x|X, Y) = m_n(x) + \frac{k_n(X, x)}{k_n(X)}(Y - m_n(X)) \tag{4.31}$$

$$k_{n+1}(x_1, x_2|X) = k_n(x_1, x_2) - \frac{k_n(X, x_1)k_n(X, x_2)}{k_n(X)}. \tag{4.32}$$

We consider $N \geq 1$ observations $\mathbf{y} = (y^{(k)})_{1 \leq k \leq N}$ of the unknown direct model which are noisy with a zero-mean Gaussian noise with variance σ_m^2.

To develop a SUR strategy, the first step is to provide an uncertainty functional $\mathcal{H}$ defined over Gaussian measures. Before doing so, we introduce a few notations and definitions.

Definition 4.2.1. *For any Gaussian measure $\nu \in \mathbb{M}$, we introduce the posterior probability distribution $p_\nu(\,\cdot\,|\mathbf{y})$. Its density $p_\nu(x|\mathbf{y})$ at $x \in \mathcal{X}$ is proportional to the product of the prior density and the global likelihood defined in (2.76), where the surrogate model f_s is a GP with probability measure ν. The existence of a GP with continuous sample paths stems from proposition 4.1.2.*

When considering the Gaussian measure P_n^f, the posterior distribution is simply denoted by $p_n(\,\cdot\,|\mathbf{y})$. We also write explicitly the dependence of the matrix $\mathbf{\Sigma}(x)$ used in (2.76) w.r.t. n. The new notation used in this chapter is $\mathbf{\Sigma}_n(x)$ which is defined for a scalar inverse problem by:

$$\mathbf{\Sigma}_n(x) = \sigma_m^2 \mathcal{I}_N + k_n(x)\mathcal{U}_N. \tag{4.33}$$

Our first task is to define a metric of uncertainty.

Definition 4.2.2. *For scalar GP surrogates, the IP-SUR functional, denoted by $\mathcal{H}$, is defined for any Gaussian measure $\nu \in \mathbb{M}$ by:*

$$\mathcal{H}(\nu) = \int_{\mathcal{X}} k_\nu(x) p_\nu(x|\mathbf{y}) dx. \tag{4.34}$$

This functional is adapted from IMSPE designs where the posterior density is used as a weight function. It is a relevant uncertainty functional for our task.

Considering the SUR framework, we would like to express the quantity $H_n = \mathcal{H}(P_n^f)$ defined by:

$$H_n = \int_{\mathcal{X}} k_n(x) p_n(x|\mathbf{y}) dx. \tag{4.35}$$

More importantly, for $X \in \mathcal{X}$, we want to express the quantity $J_n(X)$:

$$J_n(X) = \mathbb{E}_{n,X}\left[\mathcal{H}(P_{n+1}^f)\right] = \mathbb{E}_Y\left[H_{n+1}(X,Y)\right] \tag{4.36}$$

$$H_{n+1}(X,Y) = \mathcal{H}\left(P_n^f|(X,Y)\right) = \int_{\mathcal{X}} k_{n+1}(x|X) p_{n+1}(x|\mathbf{y},X,Y) dx \tag{4.37}$$

where $p_{n+1}(x|\mathbf{y},X,Y) \propto L_{n+1}(\mathbf{y}|x,X,Y)p(x)$ is the updated posterior density and $L_{n+1}(\mathbf{y}|x,X,Y)$ is the updated likelihood given by:

$$L_{n+1}(\mathbf{y}|x,X,Y) = ((2\pi)^N |\mathbf{\Sigma}_{n+1}(x|X)|)^{-1/2} \exp\left[-\frac{1}{2}\left\|\mathbf{y} - \mathbf{m_{n+1}}(x|X,Y)\right\|^2_{\mathbf{\Sigma}_{n+1}}\right] \tag{4.38}$$

$$\mathbf{\Sigma}_{n+1}(x|X) = \mathbf{\Sigma}_n(x) - \lambda_n(x,X)\mathbf{u}\mathbf{u}^T \tag{4.39}$$

with $\mathbf{u} = (1,...,1)^T \in \mathbb{R}^N$, $\mathbf{m_{n+1}}(x|X,Y) = m_{n+1}(x|X,Y)\mathbf{u}$ and:

$$\lambda_n(x,X) = \frac{k_n(X,x)^2}{k_n(X)}. \tag{4.40}$$

We also point out that we remove the dependency of $\mathbf{\Sigma}_{n+1}(x|X)$ w.r.t. x and X in the Mahalanobis norm $\|\cdot\|_{\mathbf{\Sigma}_{n+1}}$. The same is done for any other matrix, to simplify the notations.

In practice we would like to write a SUR criterion for this functional $\mathcal{H}$. However, using $\mathcal{H}$ leads to two caveats. First, the SUR criterion is not tractable. The reason lies in the normalization constant of the posterior distribution C_n defined by:

$$C_n = \int_{\mathcal{X}} p(x) L_n(\mathbf{y}|x) dx. \tag{4.41}$$

At iteration $n+1$, this normalization constant depends on Y and complicates the computation of $J_n(X)$.

Additionnally, for similar reasons, we cannot prove that $\mathcal{H}$ has the supermartingale property, which means that we cannot use Theorem 4.1.1 to show the convergence of H_n.

To overcome these difficulties, we introduce the functional $\mathcal{D}$ defined for any Gaussian measure ν as:

$$\mathcal{D}(\nu) = \int_{\mathcal{X}} k_\nu(x) p(x) L_\nu(\mathbf{y}|x) dx. \tag{4.42}$$

One can notice that $D_n = \mathcal{D}(P_n^f) = \frac{H_n}{C_n}$. We modify our objective and now we seek to write a SUR criterion for the functional $\mathcal{D}$ and show that under this design, we have the convergence of H_n to zero almost surely.

We introduce the notation:

$$F_n(X) = \mathbb{E}_{n,X}\left[\mathcal{D}(P_{n+1}^f)\right] = \mathbb{E}_Y\left[\mathcal{D}_{n+1}(X,Y)\right] \tag{4.43}$$

where:

$$D_{n+1}(X,Y) = \int_{\mathcal{X}} k_{n+1}(x|X)L_{n+1}(\mathbf{y}|x,X,Y)p(x)dx. \tag{4.44}$$

The SUR criterion corresponding to this new functional $\mathcal{D}$ is given by:

$$X_{n+1} \in \operatorname{argmin}_{X\in\mathcal{X}} F_n(X). \tag{4.45}$$

From now on, this design is referred to as the IP-SUR design.

We can claim that the IP-SUR strategy is interesting if we can solve the following two problems. First, we need to make sure the quantity $F_n(X)$ can be evaluated (up to a multiplicative constant) to solve (4.45). Otherwise, a suitable approximation must be used as is done in [ZLR19] for an expected improvement criterion for example. Second, we need to show the convergence of the metric of uncertainty.

Let us begin with the first task. For $X \in \mathcal{X}$, we would like to compute $F_n(X)$. Since in our work, we have access to an ergodic Markov chain for the posterior distribution $p_n(\,\cdot\,|\mathbf{y})$, we want to write $F_n(X)$ as an expectation w.r.t. this posterior and use the ergodicity of the Markov chain to evaluate it.

Proposition 4.2.1. *For $X \in \mathcal{X}$, the quantity $F_n(X)$ is given by:*

$$F_n(X) = \int_{\mathcal{X}} k_{n+1}(x|X)L_n(\mathbf{y}|x)p(x)dx = \int_{\mathcal{X}} \left(k_n(x) - \lambda_n(x,X)\right) L_n(\mathbf{y}|x)p(x)dx. \tag{4.46}$$

Proof. Let $X \in \mathcal{X}$. By inverting the two integrals, we write $F_n(X)$ as:

$$\begin{aligned} F_n(X) &= \mathbb{E}_Y\left[\int_{\mathcal{X}} k_{n+1}(x|X)L_{n+1}(\mathbf{y}|x,X,Y)p(x)dx\right] \\ &= \int_{\mathcal{X}}\int_{\mathbb{R}} k_{n+1}(x|X)p(x)L_{n+1}(\mathbf{y}|x,X,Y)p_n(Y|X)dYdx \end{aligned} \tag{4.47}$$

where $p_n(Y|X)$ is the Gaussian density of the distribution $Y \sim \mathcal{N}\left(m_n(X), k_n(X)\right)$. One can notice that $p_n(Y|X) = p_n(Y|X,x)$ since this distribution does not depend on x. Thus we have:

$$\begin{aligned} F_n(X) &= \int_{\mathcal{X}}\int_{\mathbb{R}} k_{n+1}(x|X)p(x)L_{n+1}(\mathbf{y}|x,X,Y)p_n(Y|X,x)dYdx \\ &= \int_{\mathcal{X}} p(x)k_{n+1}(x|X)\int_{\mathbb{R}} p_n(Y,\mathbf{y}|X,x)dYdx \end{aligned} \tag{4.48}$$

where $p_n(Y,\mathbf{y}|X,x)$ is the joint density of Y and $\mathbf{y}$ given X and x. Finally, the integral on Y leaves only the conditional density of $\mathbf{y}$ given X and x, which is the likelihood:

$$F_n(X) = \int_{\mathcal{X}} p(x)k_{n+1}(x|X)L_n(\mathbf{y}|X,x)dx. \tag{4.49}$$

□

One can notice that $F_n(X) = C_n \times \int_{\mathcal{X}} k_{n+1}(x|X)p_n(x|\mathbf{y})dx$ where C_n is defined by (4.41). Since C_n does not depend on X, the SUR criterion (4.45) is equivalent to:

$$X_{n+1} \in \operatorname{argmin}_{X\in\mathcal{X}} \widehat{F_n(X)} \tag{4.50}$$

$$\widehat{F_n(X)} = \int_{\mathcal{X}} k_{n+1}(x|X)p_n(x|\mathbf{y})dx. \tag{4.51}$$

Assuming we have access to an ergodic Markov chain $(\widehat{X_l})_{1\leq l\leq L}$ for $L \geq 1$ and with invariant distribution given by the posterior $p_n(\,\cdot\,|\mathbf{y})$, then $\widehat{F_n(X)}$ can be evaluated for all $X \in \mathcal{X}$ by:

$$\widehat{F_n(X)} \simeq \frac{1}{L}\sum_{l=1}^{L} k_{n+1}(X_l|X). \tag{4.52}$$

The IP-SUR criterion is thus tractable since we can evaluate the objective function $\widehat{F_n(X)}$ to minimize. In the next paragraphs, we discuss the convergence property of the IP-SUR design.

4.2.2.2 Supermartingale property and almost-sure convergence

In this section, some results on the convergence of the functional $\mathcal{H}$ are highlighted. The theoretical foundation of this work can be found in [BBG19].

As a first step, the supermartingale property is shown for the auxiliary functional $\mathcal{D}$. This lemma is then used to derive the main convergence theorem.

Lemma 4.2.1. *The functional $\mathcal{D}\colon \mathbb{M} \to \mathbb{R}_+$ defined by (4.42) has the supermartingale property. In other words, for any sequential design $(X_n)_{n\in\mathbb{N}}$, there exists $n_0 \in \mathbb{N}$ such that for all $n \geq n_0$, and for all $X \in \mathcal{X}$*

$$\mathbb{E}_{n,X}\left[\mathcal{D}(P^f_{n+1})\right] \leq \mathcal{D}(P^f_n). \tag{4.53}$$

Proof. Let us prove the supermartingale property. Let f_s be a GP and $(X_n)_{n\in\mathbb{N}}$ be a sequential design. Let $X \in \mathcal{X}$. We can write:

$$D_n - F_n(X) = \int_{\mathcal{X}} \left(k_n(x) - k_{n+1}(x|X)\right) L_n(\mathbf{y}|x)p(x)dx. \tag{4.54}$$

The supermartingale property stems directly from this equation since $k_n(x) - k_{n+1}(x|X) = \lambda_n(x, X) \geq 0$ for all $x \in \mathcal{X}$. Thus $D_n - F_n(X) \geq 0$. □

Lemma 4.2.2. *If $C_n = \int_{\mathcal{X}} L_n(\mathbf{y}|x)p(x)dx$, then the sequence $(C_n)_{n\in\mathbb{N}}$ converges almost surely and its limit is positive and given by:*

$$C_\infty = \int_{\mathcal{X}} L_\infty(\mathbf{y}|x)p(x)dx \tag{4.55}$$

where $L_\infty(\mathbf{y}|x)$ is defined for $x \in \mathcal{X}$ by:

$$L_\infty(\mathbf{y}|x) = \left((2\pi)^N |\mathbf{\Sigma}_\infty(x)|\right)^{-1/2} \exp\left[-\frac{1}{2}\left\|\mathbf{y} - \mathbf{m}_\infty(x)\right\|^2_{\mathbf{\Sigma}_\infty}\right] \tag{4.56}$$

with m_∞ and k_∞ being the respective limits of the GP mean function $(m_n)_{n\in\mathbb{N}}$ and covariance function $(k_n)_{n\in\mathbb{N}}$ and $\mathbf{\Sigma}_\infty(x) = k_\infty(x)\mathbf{u}\mathbf{u}^T + \sigma_m^2\mathcal{I}_N$ for $x \in \mathcal{X}$.

Proof. Let us first prove the convergence of the mean functions $(m_n)_{n\in\mathbb{N}}$. From proposition 2.9 in [BBG19], for any sequential design and any Gaussian process, the probability distribution of the GP given $\mathcal{F}_n = \sigma(X_1, Y_1, ..., X_n, Y_n)$, which is denoted by P_n^f, converges almost surely to a limit Gaussian measure $P_\infty^f \in \mathbb{M}$. Since the convergence of Gaussian measures is defined as the uniform convergence of the mean functions $(m_n)_{n\in\mathbb{N}}$ and covariance functions $(k_n)_{n\in\mathbb{N}}$ we can then define $m_\infty = \lim_{n\to+\infty} m_n$ which is the mean function of the GP whose probability distribution is P_∞^f. Furthermore, since m_n and k_n are continuous and $(m_n)_{n\in\mathbb{N}}$ (resp. $(k_n)_{n\in\mathbb{N}}$) converges uniformly to m_∞ (resp. k_∞), then m_∞ and k_∞ are continuous.
From here we show that $\mathbf{\Sigma}_n(x) \xrightarrow[n\to+\infty]{a.s.} \mathbf{\Sigma}_\infty(x) = k_\infty(x)\mathbf{u}\mathbf{u}^T + \sigma_m^2\mathcal{I}_N$ with $k_\infty(x) \geq 0$.
By continuity of all the other matrix operations, we have $L_n(\mathbf{y}|x) \xrightarrow[n\to+\infty]{a.s.} L_\infty(\mathbf{y}|x)$ with:

$$L_\infty(\mathbf{y}|x) = ((2\pi)^N|\mathbf{\Sigma}_\infty(x)|)^{-1/2} \exp\left[-\frac{1}{2}\|\mathbf{y} - \mathbf{m}_\infty(x)\|^2_{\mathbf{\Sigma}_\infty}\right]. \tag{4.57}$$

To conclude, we need to notice that L_∞ is continuous (with respect to x) since m_∞ and k_∞ are continuous. It is thus bounded on the compact set $\mathcal{X}$ and we can make use of the dominated convergence theorem to conclude:

$$\lim_{n\to+\infty}\int_{\mathcal{X}} L_n(\mathbf{y}|x)p(x)dx = \int_{\mathcal{X}} \lim_{n\to+\infty} L_n(\mathbf{y}|x)p(x)dx = \int_{\mathcal{X}} L_\infty(\mathbf{y}|x)p(x)dx = C_\infty. \tag{4.58}$$

Besides, since $L_\infty(\mathbf{y}|x)p(x) > 0$ for all $x \in \mathcal{X}$ and $\mathcal{X}$ is a compact, $0 < C_\infty < +\infty$ almost-surely. □

Then, let us investigate the convergence of the functional $\mathcal{H}$. The theorem 4.1.1 is our main tool to prove the convergence of $\mathcal{H}$. We also use the fact that $\mathcal{D}(\nu) = C_\nu \times \mathcal{H}(\nu)$ with $C_\nu = \int_{\mathcal{X}} L_\nu(\mathbf{y}|x)p(x)dx$.

Theorem 4.2.1. *Consider the functional $\mathcal{D}$ defined in (4.42) and a SUR sequential design $(X_n)_{n\in\mathbb{N}}$ associated with this functional and given by (4.50). Then, the sequence $(D_n)_{n\geq 1}$ where $D_n = \mathcal{D}(P_n^f)$ converges almost surely to 0:*

$$D_n \xrightarrow[n\to+\infty]{a.s.} 0.$$

Besides, if $\mathcal{H}$ is the functional defined by (4.34), the same convergence holds for the sequence $(H_n)_{n\geq 1}$ where $H_n = \mathcal{H}(P_n^f)$:

$$H_n \xrightarrow[n\to+\infty]{a.s.} 0.$$

Proof. This proof is divided into two parts. First, we show that $D_n = \mathcal{D}(P_n^f) \xrightarrow[n\to+\infty]{a.s.} 0$ using the supermartingale property and the convergence theorem from [BBG19]. Then, we show the convergence for H_n using lemma 4.2.2.
Let us first show that $D_n \xrightarrow[n\to+\infty]{a.s.} 0$. Consider a Gaussian process f_s and a SUR sequential design $(X_n)_{n\in\mathbb{N}}$ given by the strategy (4.45) defined for the functional $\mathcal{D}$. Let us verify the conditions of theorem 4.1.1 for the functional $\mathcal{D}$. We introduce the

functional $\mathcal{G}$ which is defined for $\nu \in \mathbb{M}$ by:

$$\mathcal{G}(\nu) = \sup_{X \in \mathcal{X}} \left(\mathcal{D}(\nu) - \mathbb{E}_Y \left[\mathcal{D}(\nu | (X, Y)) \right] \right). \tag{4.59}$$

If P_n^f is the probability of f_s given $\mathcal{F}_n = \sigma(X_1, Y_1, .., X_n, Y_n)$, then there exists an $\mathcal{F}_\infty$-measurable random element $P_\infty^f \in \mathbb{M}$ such that $P_n^f \xrightarrow[n \to +\infty]{a.s.} P_\infty^f$, with $\mathcal{F}_\infty = \sigma\left(\bigcup_{n \geq 1} \mathcal{F}_n\right)$, and such that P_∞^f is the conditional probability of f_s given $\mathcal{F}_\infty$. The associated mean and covariance functions are introduced as m_∞ and k_∞. Since the functions m_n are continuous and the sequence $(m_n)_{n \in \mathbb{N}}$ converge uniformly toward the limit mean function m_∞, then m_∞ is also continuous. The same reasoning holds for k_∞.
From here, the convergence of $\mathcal{D}(P_n^f) \xrightarrow[n \to +\infty]{a.s.} \mathcal{D}(P_\infty^f)$ and $\mathcal{G}(P_n^f) \xrightarrow[n \to +\infty]{a.s.} \mathcal{G}(P_\infty^f)$ is obtained by dominated convergence on the compact $\mathcal{X}$ and by continuity of the mean and variance of the Gaussian posterior measure.
Now, let us show that $\mathbb{Z}_\mathcal{D} = \mathbb{Z}_\mathcal{G}$, where $\mathbb{Z}_\mathcal{D}$ and $\mathbb{Z}_\mathcal{G}$ are the set of zeros of the functionals $\mathcal{D}$ and $\mathcal{G}$. The first inclusion $\mathbb{Z}_\mathcal{D} \subset \mathbb{Z}_\mathcal{G}$ is trivial since $0 \leq \mathcal{G}(\nu) \leq \mathcal{D}(\nu)$ for all $\nu \in \mathbb{M}$. Now, let $P^f \in \mathbb{Z}_\mathcal{G}$. It is a Gaussian measure associated to a GP f_s. We want to show that $\mathcal{D}(P^f) = 0$. We will prove this result for the posterior measure P_n^f for all $n \geq 0$ to be able to re-use the previous notations, though we are only interested in the case $n = 0$. In particular $P^f = P_0^f$.
Let us introduce $F_n(X) = \mathbb{E}_{n,X}\left[\mathcal{D}(P_{n+1}^f)\right] = \mathbb{E}_Y\left[\mathcal{D}\left(P_n^f | (X, Y)\right)\right]$. From this, $\mathcal{G}(P_n^f) = D_n - \inf_{X \in \mathcal{X}} F_n(X) = 0$. Then, with the supermartingale property of $\mathcal{D}$, it follows that for all $X \in \mathcal{X}$:

$$0 \leq D_n - F_n(X) \leq D_n - \inf_{X' \in \mathcal{X}} F_n(X') = 0 \tag{4.60}$$

thus $D_n - F_n(X) = 0$ for all $X \in \mathcal{X}$.
Using equation (4.54), if $D_n - F_n(X) = 0$, then for almost all $x \in \mathcal{X}$, we have $k_n(x, X) = 0$, since $L_n(\mathbf{y}|x) > 0$. In other words, the set $\mathcal{C}_X = \{x \in \mathcal{X} | k_n(x, X) \neq 0\}$ has Lebesgue measure zero, for all $X \in \mathcal{X}$.
Let us proceed by contradiction. We suppose that there exists $\mathcal{X}_1 \subset \mathcal{X}$ such that $\mu(\mathcal{X}_1) > 0$ and for $x \in \mathcal{X}_1$, we have $k_n(x) > 0$. Let $x \in \mathcal{X}_1$. Taking $X = x$, we have $k_n(x, X) = k_n(x) > 0$. Thus by continuity of k_n, there exists an open set $\mathcal{X}_2 \subset \mathcal{X}$ such that $\mu(\mathcal{X}_2) > 0$ and for all $x \in \mathcal{X}_2$, $k_n(x, X) > 0$. We have our contradiction since $\mathcal{X}_2 \subset \mathcal{C}_X$ and $\mu(\mathcal{C}_X) = 0$. We can conclude that $\mathbb{Z}_\mathcal{D} = \mathbb{Z}_\mathcal{G}$.
All the assumptions of the theorem 4.1.1 are verified, and the almost sure convergence of D_n is proven.
Now consider $(C_n)_{n \in \mathbb{N}}$ the sequence of normalizing constants, defined by $C_n = \int_\mathcal{X} L_n(\mathbf{y}|x) p(x) dx$. From lemma 4.2.2 and using the same notations, $\lim_{n \to +\infty} C_n = C_\infty = \int_\mathcal{X} L_\infty(\mathbf{y}|x) p(x) dx$ which is positive almost-surely. Since $D_n \xrightarrow[n \to +\infty]{a.s.} 0$ and $H_n = \frac{D_n}{C_n}$, we can conclude that:

$$H_n \xrightarrow[n \to +\infty]{a.s.} 0. \tag{4.61}$$

□

4.2.3 IP-SUR methods for multi-output GP surrogates

The results presented in the previous section naturally extend to multi-output Gaussian processes. The proofs however are postponed to Appendix B. They are very similar to the scalar case. We extend all the notations from the scalar case. The observational covariance is denoted by $\mathbf{C}_{\text{obs}}$. Let us introduce the total covariance $\mathbf{\Sigma}_n(x)$ for $x \in \mathcal{X}$.

$$\mathbf{\Sigma}_n(x) = \mathcal{U}_N \otimes k_n(x) + \mathcal{I}_N \otimes \mathbf{C}_{\text{obs}} \in \mathbb{R}^{DN \times DN} \tag{4.62}$$

where $\otimes$ is the Kronecker product for matrices.
When conditioning by an input-output pair (X, Y), the mean and covariance functions are updated as follows.

$$m_{n+1}(x|X,Y) = m_n(x) + k_n(X,x)k_n(x)^{-1}(Y - m_n(X)) \tag{4.63}$$

$$k_{n+1}(x_1,x_2|X) = k_n(x_1,x_2) - k_n(x_1,X)k_n(X)^{-1}k(X,x_2) \tag{4.64}$$

$$|k_{n+1}(x|X)| = |k_n(x)| \times |\mathcal{I}_D - k_n(x)^{-1}k_n(x,X)k_n(X)^{-1}k_n(x,X)^T| \tag{4.65}$$

$$|\mathbf{\Sigma}_n(x)| = |\mathbf{C}_{\text{obs}}|^{N-1}|Nk_n(x) + \mathbf{C}_{\text{obs}}|. \tag{4.66}$$

4.2.3.1 Derivation of the SUR criteria

In this multi-output framework, the metric of uncertainty $\mathcal{H}\colon \mathbb{M} \to \mathbb{R}_+$ is slightly different.

Definition 4.2.3. *The IP-SUR functional $\mathcal{H}$ for the multi-output case is defined for any Gaussian measure $\nu \in \mathbb{M}$ by:*

$$\mathcal{H}(\nu) = \int_{\mathcal{X}} |k_\nu(x)| p_\nu(x|\mathbf{y}) dx. \tag{4.67}$$

Once again, we introduce the functional $\mathcal{D}$ defined for any $\nu \in \mathbb{M}$ by:

$$\mathcal{D}(\nu) = \int_{\mathcal{X}} |k_\nu(x)| L_\nu(\mathbf{y}|x) p(x) dx \tag{4.68}$$

and we work with $\mathcal{D}$ to derive the SUR criterion.
We wish to evaluate the quantity $F_n(X) = \mathbb{E}_{n,X}\left[\mathcal{D}(P^f_{n+1})\right]$. We recall the global likelihood from (2.76):

$$L_n(\mathbf{y}|x) = ((2\pi)^{DN}|\mathbf{\Sigma}_n(x)|)^{-1/2} \exp\left[-\frac{1}{2}\|\mathbf{y} - \mathbf{m_n}(x)\|^2_{\mathbf{\Sigma}_n}\right] \tag{4.69}$$

where the bold notation denotes the flattened vectors:

$$\mathbf{y} - \mathbf{m_n}(x) = \begin{pmatrix} y^{(1)} - m_n(x) \\ \vdots \\ y^{(N)} - m_n(x) \end{pmatrix} \in \mathbb{R}^{DN}. \tag{4.70}$$

For practical applications, we recall that the likelihood can be simplified to yield (2.78).

Proposition 4.2.2. *The SUR criterion for the metric $\mathcal{D}$ defined in (4.68) is given by:*

$$X_{n+1} \in \operatorname{argmin}_{X \in \mathcal{X}} F_n(X) \tag{4.71}$$

where $F_n(X) = \mathbb{E}_{n,X}\left[\mathcal{D}(P^f_{n+1})\right]$ for all $X \in \mathcal{X}$. This SUR criterion is equivalent to the criterion:

$$X_{n+1} \in \operatorname{argmin}_{X \in \mathcal{X}} \widehat{F_n(X)} \tag{4.72}$$

where the quantity $\widehat{F_n(X)}$ is given for $X \in \mathcal{X}$ by:

$$\widehat{F_n(X)} = \int_{\mathcal{X}} |k_{n+1}(x|X)| p_n(x|\mathbf{y}) dx. \tag{4.73}$$

Proof. The proof is given in Appendix B.2.1. □

In this criterion, the quantity $\widehat{F_n(X)}$ can be evaluated using the ergodic theorem for a Markov chain $(\widehat{X_l})_{1 \le l \le L}$ with target distribution $p_n(\,\cdot\,|\mathbf{y})$:

$$\widehat{F_n(X)} \simeq \frac{1}{L} \sum_{l=1}^{L} |k_{n+1}(\widehat{X_l}|X)|. \tag{4.74}$$

Our next objective is to show that the convergence result obtained for the scalar case extends to this multi-output framework.

4.2.3.2 Almost sure convergence

The derivation of the almost-sure convergence of H_n is similar to the scalar case. The following lemma is used as an intermediate result:

Lemma 4.2.3. *The functional $\mathcal{D}\colon \mathbb{M} \to \mathbb{R}_+$ defined by (4.68) has the supermartingale property. In other words, for any sequential design $(X_n)_{n \in \mathbb{N}}$, there exists $n_0 \in \mathbb{N}$ such that for all $n \ge n_0$, and for all $X \in \mathcal{X}$*

$$\mathbb{E}_{n,X}\left[\mathcal{D}(P^f_{n+1})\right] \le \mathcal{D}(P^f_n). \tag{4.75}$$

Proof. The proof is given in Appendix B.2.2. □

With this lemma, we extend the convergence theorem to multi-output GP.

Theorem 4.2.2. *Consider the functional $\mathcal{H}$ defined in (4.67) and a SUR sequential design $(X_n)_{n \in \mathbb{N}}$ for the functional $\mathcal{D}$ defined in (4.68). Then, the sequence $(D_n)_{n \ge 1}$ where $D_n = \mathcal{D}(P^f_n)$ converges almost surely to 0:*

$$D_n \xrightarrow[n \to +\infty]{a.s.} 0.$$

Besides, we have the same convergence for the sequence $(H_n)_{n \ge 1}$ where $H_n = \mathcal{H}(P^f_n)$:

$$H_n \xrightarrow[n \to +\infty]{a.s.} 0.$$

Proof. The proof is the same as for the scalar case. □

We can then finally conclude that the IP-SUR strategy proposed in this chapter does extend to multi-output GP surrogate models, which are at the core of this thesis.

4.2.4 Tempered SUR strategy

In this section, we introduce a variant of the IP-SUR strategy relying on a tempered likelihood [Nea01; HZ02]. This strategy, named TIP-SUR (Tempered IP-SUR) is based on a temperature coefficient $\beta \in [0,1]$ which smoothes the likelihood.

Definition 4.2.4. *For $\beta \in [0,1]$, the TIP-SUR functional $\mathcal{H}_\beta$ is defined for any $\nu \in \mathbb{M}$ by:*

$$\mathcal{H}_\beta(\nu) = \frac{1}{C_{\nu,\beta}} \int_{\mathcal{X}} |k_\nu(x)| \left(L_\nu(\mathbf{y}|x)\right)^\beta p(x)dx \tag{4.76}$$

$$C_{\nu,\beta} = \int_{\mathcal{X}} \left(L_\nu(\mathbf{y}|x)\right)^\beta p(x)dx. \tag{4.77}$$

The TIP-SUR functional serves as an intermediate between the IP-SUR functional defined in equation (4.67) (for $\beta = 1$) and the IMSPE criteria defined in (4.7) (for $\beta = 0$). We expect this method to be efficient when dealing with highly peaked likelihood, since the tempering parameter β tends to smooth out the distribution. Once again, the functional $\mathcal{H}_\beta$ cannot be used directly to derive a tractable SUR design, and we introduce the functional $\mathcal{D}_\beta$ defined for any Gaussian measure $\nu \in \mathbb{M}$ by:

$$\mathcal{D}_\beta(\nu) = \int_{\mathcal{X}} |k_\nu(x)| \left(L_\nu(\mathbf{y}|x)\right)^\beta p(x)dx. \tag{4.78}$$

Similarly, we introduce the conditional expectation:

$$F_{n,\beta}(X) = \mathbb{E}_{n,X}\left[\mathcal{D}_\beta(P^f_{n+1})\right]. \tag{4.79}$$

Proposition 4.2.3. *The TIP-SUR criterion associated to the metric $\mathcal{D}_\beta$ can be simplified to:*

$$X_{n+1} \in \operatorname{argmin}_{X\in\mathcal{X}} F_{n,\beta}(X) \tag{4.80}$$

$$F_{n,\beta}(X) = \int_{\mathcal{X}} |k_{n+1}(x|X)| f_\beta(x,X) I_\beta(x,X) \left(L_n(\mathbf{y}|x)\right)^\beta p(x)dx. \tag{4.81}$$

where we define:

$$f_\beta(x,X) = \frac{|\mathbf{\Sigma}_n(x)|^{\beta/2}}{|\mathbf{\Sigma}_{n+1}(x|X)|^{\beta/2}} \exp\left(-\frac{\beta}{2}\|\mathbf{y} - \mathbf{m_n}(x)\|^2_{\left(\mathbf{\Sigma}_{n+1}^{-1} - \mathbf{\Sigma}_n^{-1}\right)^{-1}}\right) \tag{4.82}$$

$$C_{n,\beta} = \int_{\mathcal{X}} \left(L_n(\mathbf{y}|x)\right)^\beta p(x)dx \tag{4.83}$$

$$I_\beta(x,X) = \left(|k_n(X)||A_\beta|\right)^{-1/2} \exp\left(\frac{1}{2} B_\beta^2 A_\beta^{-1} B_\beta\right) \tag{4.84}$$

$$A_\beta = k_n(X)^{-1}\left(\mathcal{I}_D + \beta N k_n(X,x)\mathbf{M}_{n+1}^{-1} k_n(x,X) k_n(X)^{-1}\right) \tag{4.85}$$

$$B_\beta = \beta N k_n(X)^{-1} k_n(X,x)\mathbf{M}_{n+1}^{-1}(\overline{y} - m_n(x)). \tag{4.86}$$

Proof. The proof is similar to the IP-SUR case. We can simplify the expressions of $|A_\beta|$ and $B_\beta A_\beta^{-1} B_\beta$ by writing:

$$\begin{aligned} k_n(x,X)A_\beta k_n(X,x) &= \lambda_n(x,X) + \beta N \lambda_n(x,X) \mathbf{M_{n+1}}^{-1} \lambda_n(x,X) \\ &= \lambda_n(x,X)\mathbf{M}_{n+1}^{-1}\left(\mathbf{M}_n - N(1-\beta)\lambda_n(x,X)\right) \end{aligned} \tag{4.87}$$

which leads to:

$$|A_\beta| = |k_n(X)|^{-1} \frac{|\mathbf{M}_n - N(1-\beta)\lambda_n(x,X)|}{|\mathbf{M}_{n+1}|}. \tag{4.88}$$

The inverse of A_β can be written as:

$$\begin{aligned} A_\beta^{-1} &= k_n(X) - N\beta k_n(X,x)\left(\mathbf{M}_{n+1} + N\beta k_n(x,X)k_n(X)^{-1}k_n(X,x)\right)^{-1} k_n(x,X) \\ &= k_n(X) - N\beta k_n(X,x)\left(\mathbf{M}_n - N(1-\beta)\lambda_n(x,X)\right)^{-1} k_n(x,X). \end{aligned} \tag{4.89}$$

We cannot further simplify this expression contrary to the IP-SUR case. However, this criterion is still tractable. □

The TIP-SUR criterion necessitates an ergodic Markov chain whose invariant distribution is a tempered posterior whose density is $p_{n,\beta}(\,\cdot\,|\mathbf{y}) = \frac{(L_n(\mathbf{y}|x))^\beta p(x)}{C_{n,\beta}}$.

Proposition 4.2.4. *Let $\beta \in (0,1)$ and $(X_n)_{n\geq 1}$ be a SUR sequential design for the criterion (4.80). Then, the sequence $(D_{n,\beta})_{n\geq 0}$ where $D_{n,\beta} = \mathcal{D}_\beta(P_n^f)$ converges almost surely to zero:*

$$D_{n,\beta} \xrightarrow[n\to+\infty]{a.s.} 0.$$

The same convergence holds for the sequence $(H_{n,\beta})_{n\geq 0}$ with $H_{n,\beta} = \mathcal{H}_\beta(P_n^f)$:

$$H_{n,\beta} \xrightarrow[n\to+\infty]{a.s.} 0.$$

Proof. The proof is given in Appendix B.3. □

4.3 Applications

In the following applications, we apply the CSQ and IP-SUR strategies to various test cases. The methodology is similar for each test case. The sequential design strategy is iterated a fixed number of times up to $n = 10$. At each step, an MCMC chain is sampled to obtain the posterior distribution $p_n(x|\mathbf{y})$, with the current GP model $f_s^{(n)}$. Then, the KL divergence of each distribution $p_n(x|\mathbf{y})$ with the distribution $p_\infty(x|\mathbf{y})$ obtained by MCMC with a GP model trained on a large number of data points $n_\infty = 1000$ is quantified.

$$\kappa_n = \mathrm{KL}\left(p_n \| p_\infty\right) = \int_{\mathcal{X}} p_n(\mathbf{y}|x) \log\left(\frac{p_n(x|\mathbf{y})}{p_\infty(x|\mathbf{y})}\right) dx. \tag{4.90}$$

To estimate this KL, the ergodicity of the Markov chain is used once again, in combination with the kernel density estimates for the posterior densities:

$$\kappa_n \simeq \frac{1}{L}\sum_{l=1}^{L} r_n(\widehat{X}_l) \text{ with } r_n(x) = \log\left(\frac{p_n(x|\mathbf{y})}{p_\infty(x|\mathbf{y})}\right) \text{ for } x \in \mathcal{X}. \tag{4.91}$$

This procedure is run 100 times to produce uncertainties on the KL.

Two other metrics are introduced. Namely, we are looking at the differential entropy S_n and the predictive variance integrated over the posterior, i.e. $H(P_n^f)$. The entropy is also easily obtained by the ergodic theorem:

$$S_n = -\int_{\mathcal{X}} p_n(x|\mathbf{y}) \log p_n(x|\mathbf{y}) dx \tag{4.92}$$

$$S_n \simeq -\frac{1}{L}\sum_{l=1}^{L} \log p_n(\widehat{X}_l|\mathbf{y}). \tag{4.93}$$

4.3.1 Description of the test cases

4.3.1.1 Banana posterior distribution

In this first test case, the target posterior distribution has a banana shape as displayed in Figure 4.1. This posterior distribution is similar to the one introduced in [SPW23]

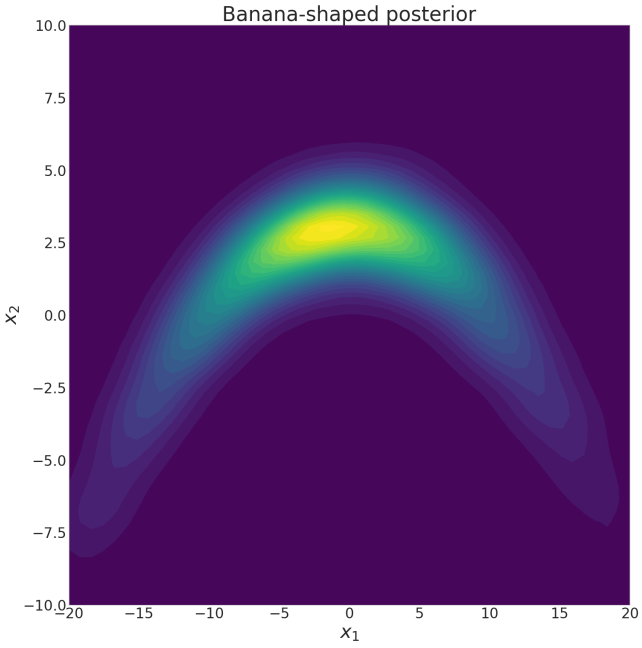

FIGURE 4.1: Banana-shaped target posterior distribution.

and is described by the following analytical direct model:

$$\begin{aligned} f_{\mathrm{b}} \colon \mathcal{X}_b &\longrightarrow \mathbb{R}^2 \\ (x_1, x_2) &\longmapsto (x_1, x_2 + 0.03x_1^2) \end{aligned}$$

where $\mathcal{X}_b = [-20, 20] \times [-10, 10] \subset \mathbb{R}^2$. For a single observation $y = (y_1, y_2)$, the posterior has the density:

$$p_{\mathrm{b}}(x|y) \propto \exp\left(-\frac{1}{2}\frac{(x_1 - y_1)^2}{100} - \frac{1}{2}\left(x_2 + 0.03x_1^2 - y_2\right)^2\right). \tag{4.94}$$

The observations $\mathbf{y} = (y^{(k)})_{1\leq k\leq N}$ are generated with $N = 5$ and such that for $1 \leq k \leq N$ we have $y^{(k)} \sim \mathcal{N}(\mu, \mathbf{C}_{\mathrm{obs}})$ with $\mu = (0, 3)$ and:

$$\mathbf{C}_{\mathrm{obs}} = \begin{pmatrix} 100 & 0 \\ 0 & 1 \end{pmatrix}. \tag{4.95}$$

The evolution of the metrics for the banana-shaped posterior is plotted in Figure 4.3.

4.3.1.2 Bimodal posterior distribution

In this second test case, the target posterior is bimodal as plotted in Figure 4.2. The corresponding direct model is f_m defined as:

$$\begin{aligned} f_{\mathrm{m}} \colon \mathcal{X}_m &\longrightarrow \mathbb{R}^2 \\ (x_1, x_2) &\longmapsto (x_2 - x_1^2, x_2 - x_1) \end{aligned}$$

where $\mathcal{X}_m = [-6, 6] \times [-4, 8] \subset \mathbb{R}^2$. For a single observation $y = (y_1, y_2)$, the posterior has the density:

$$p_{\mathrm{b}}(x|y) \propto \exp\left(-\frac{\sqrt{0.2}}{10}(x_2 - x_1^2 - y_1)^2 - \frac{\sqrt{0.75}}{10}\left(x_2 + 0.03x_1^2 - y_2\right)^2\right). \tag{4.96}$$

The observations $\mathbf{y} = (y^{(k)})_{1\leq k\leq N}$ are generated with $N = 10$ and such that for $1 \leq k \leq N$ we have $y^{(k)} \sim \mathcal{N}(\mu, \mathbf{C}_{\mathrm{obs}})$ with $\mu = (0, 2)$ and:

$$\mathbf{C}_{\mathrm{obs}} = \begin{pmatrix} \frac{5}{\sqrt{0.2}} & 0 \\ 0 & \frac{5}{\sqrt{0.75}} \end{pmatrix}. \tag{4.97}$$

The metrics for the bimodal posterior are plotted in Figure 4.4.

4.3.1.3 Application to the point model

Our final test case is derived from our applicative context. Namely, we consider the point model:

$$\begin{aligned} f_{\mathrm{pm}} \colon \mathbb{R}^4 &\longrightarrow \mathbb{R}^3 \\ (k_p, \varepsilon_F, S, x_s) &\longmapsto (R, Y_\infty, X_\infty) \end{aligned}$$

where (R, Y_∞, X_∞) are given by (1.99), (1.100) and (1.101). We consider this model instead of the extended one introduced in chapter 2, to limit ourselves to a low-dimensional problem to make the strategies more tractable. Let $\mathbf{y} = (y^{(k)})_{1\leq k\leq N}$ be a set of observations generated from the point model with $x_{\mathrm{true}} = (0.8, 0.04, 130000, 0.4)$

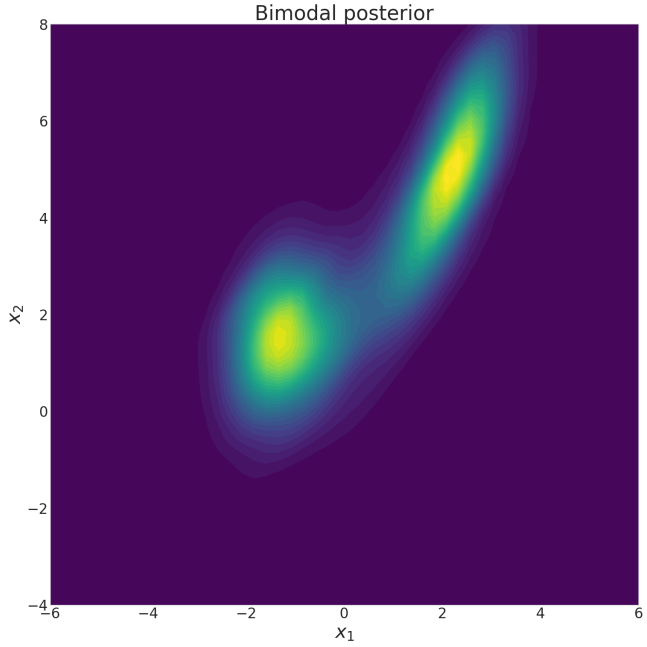

FIGURE 4.2: Bimodal target posterior distribution.

and:

$$\mathbf{C}_{\text{obs}} = \begin{pmatrix} 1.50 \times 10^{2} & 1.05 \times 10^{-1} & 7.50 \times 10^{-1} \\ 1.05 \times 10^{-1} & 1.20 \times 10^{-4} & 1.20 \times 10^{-3} \\ 7.50 \times 10^{-1} & 1.20 \times 10^{-3} & 1.20 \times 10^{-2} \end{pmatrix} \tag{4.98}$$

We consider the design space $\mathcal{X} \subset \mathbb{R}^4$ given by the prior support:

$$\mathcal{X} = [0.7, 0.9] \times [0.01, 0.10] \times [1 \times 10^5, 2 \times 10^5] \times [0.0, 1.0]. \tag{4.99}$$

The number of MCMC samples is $L = 2 \times 10^5$ iterations per run for that specific test case. The metrics are shown in Figure 4.5.

4.3.2 Results

Both the IP-SUR and the CSQ methods require to solve an optimization problem with a non-convex function. The optimization problem is solved with a dual annealing approach [Xia+97]. The IP-SUR strategy and the CSQ method are applied iteratively 10 times to produce new design points. For the CSQ strategy, the hyperparameter h introduced in (4.29) is set to $h = 3$. The influence of h is discussed afterward. The evolutions of the metrics are plotted in Figures 4.3, 4.4 and 4.5 for both sequential design strategies and for the naive strategy where the design points are chosen with a uniform distribution on the prior domain. The empirical 95 % confidence interval for each metric and strategy is also displayed.

Both the CSQ and SUR strategies perform largely better than the naive strategy. The naive strategy samples design points randomly in the parameter space and thus targets away from the posterior distribution. The performance metrics are still decreasing but at a much slower rate than for the CSQ and SUR strategies, especially in higher dimensions. These two design plans tend to perform similarly on the two-dimensional test cases but the IP-SUR strategy is superior for the neutronic test case, for which all the metrics are considerably improved.

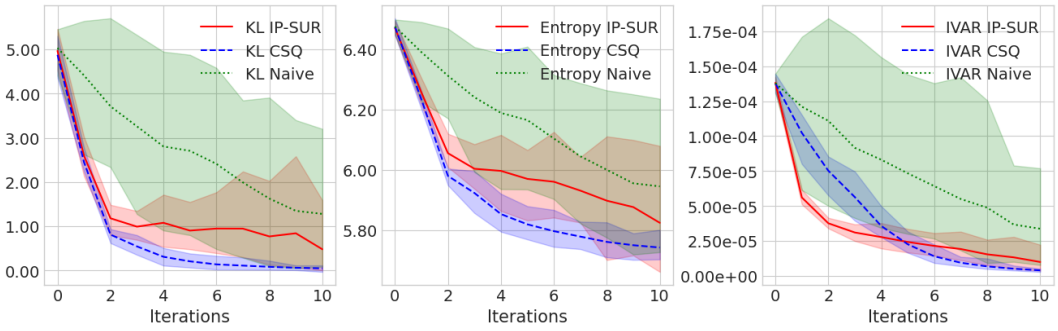

FIGURE 4.3: Performance metrics - Banana test case.

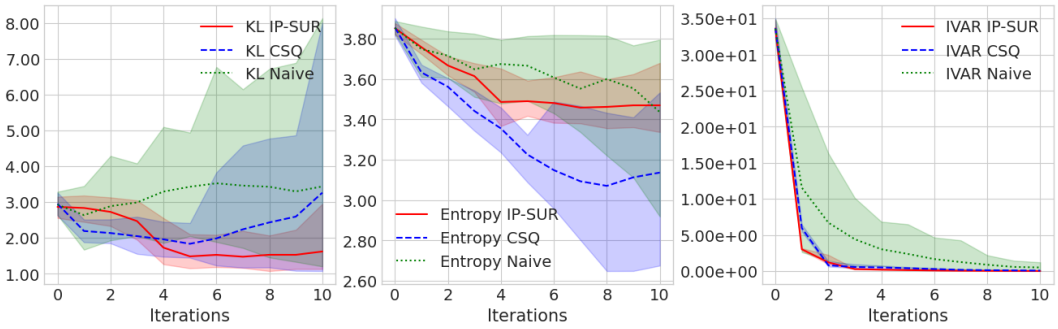

FIGURE 4.4: Performance metrics - Bimodal test case.

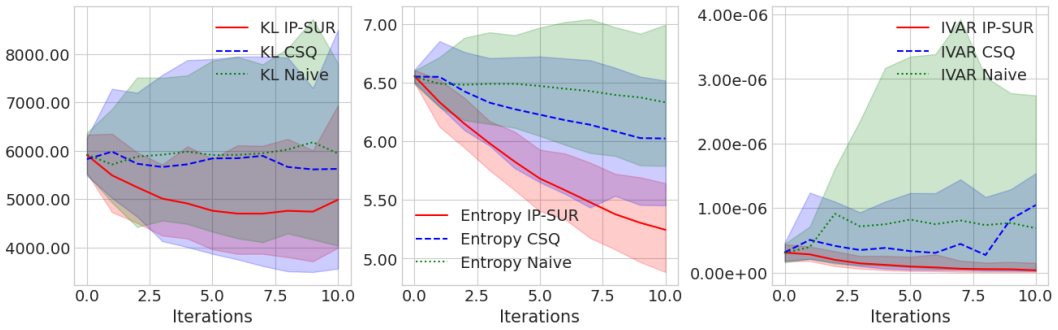

FIGURE 4.5: Performance metrics - Neutronic test case.

Based on these results, one could argue that the CSQ strategy can be situationally better as it is easier to set up while providing similar performance in the end. However, two counter-arguments can be pointed out. First of all, the IP-SUR strategy does exhibit a guarantee for the convergence of the integrated variance, which offers a strong theoretical foundation. Besides, though the acquisition function in the IP-SUR strategy is more computationally intensive, the method does not rely on the prior setting of an arbitrary hyperparameter. The influence of the hyperparameter h is discussed in the next section.

4.3.3 Influence of the hyperparameter in CSQ

The CSQ strategy depends on a hyperparameter $h \geq 0$ introduced in the definition of the subset $\mathcal{B}_h^{(n)}$ in (4.29). There is no rule to select the optimal value of this hyperparameter. From an intuitive point of view, we want it to be large enough to

allow design points to be away from the MAP but not too large to keep only those relevant to the inverse problem. The optimal value may vary largely between two different posterior distributions.

To assess the influence of the hyperparameter h, the CSQ strategy is applied for various values of h on the two test cases. Arbitrarily, the values of h are selected in $h \in \{1,2,3\}$. The metrics obtained for the banana-shaped and bimodal posterior distributions are shown in Figures 4.6 and 4.7 respectively. One can see that for

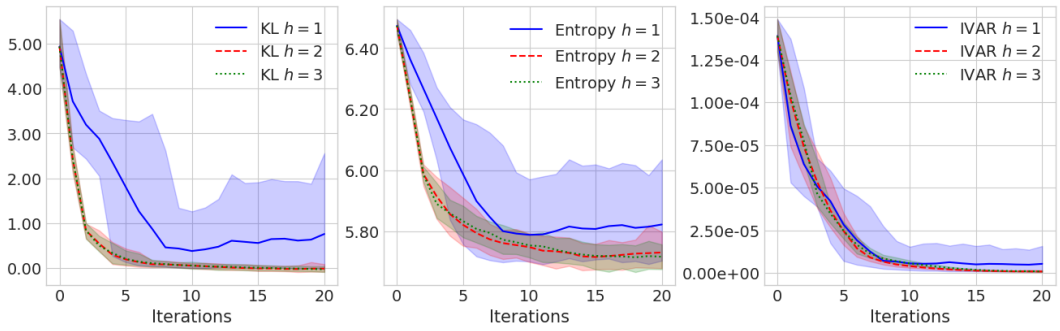

FIGURE 4.6: Influence of h - Banana test case.

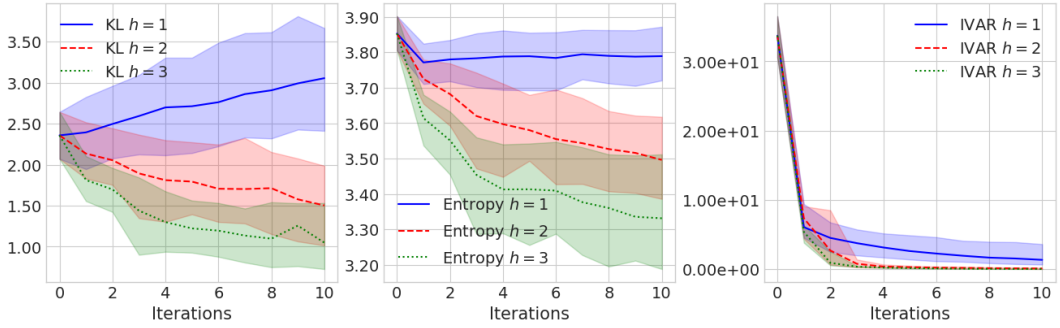

FIGURE 4.7: Influence of h - Bimodal test case.

$h = 1$, the design is significantly worsened. We recall that all the previous test cases were conducted with $h = 3$ which provides the best results among the selected values. However, the optimal choice of h is likely dependent on the application and cannot be found easily. For this reason, the IP-SUR strategy presented in this work seems superior as it does not carry the burden of the selection of a hyperparameter.

4.3.4 Comparison to standard strategies

As an example, we consider the standard I-optimal and D-optimal strategies introduced at the beginning of this chapter. These two strategies are compared to the IP-SUR strategy, on the bimodal test case for 5 iterations. Though the surrogate models may be largely improved with these designs, most improvements may be irrelevant to the inverse problem. We can further emphasize that point by looking at the posterior distributions obtained for the GP models with the new design points of each strategy. One of the marginals of such posteriors is displayed in Figure 4.8 for the previously mentioned strategies. From this analysis, we highlight the inadequacy

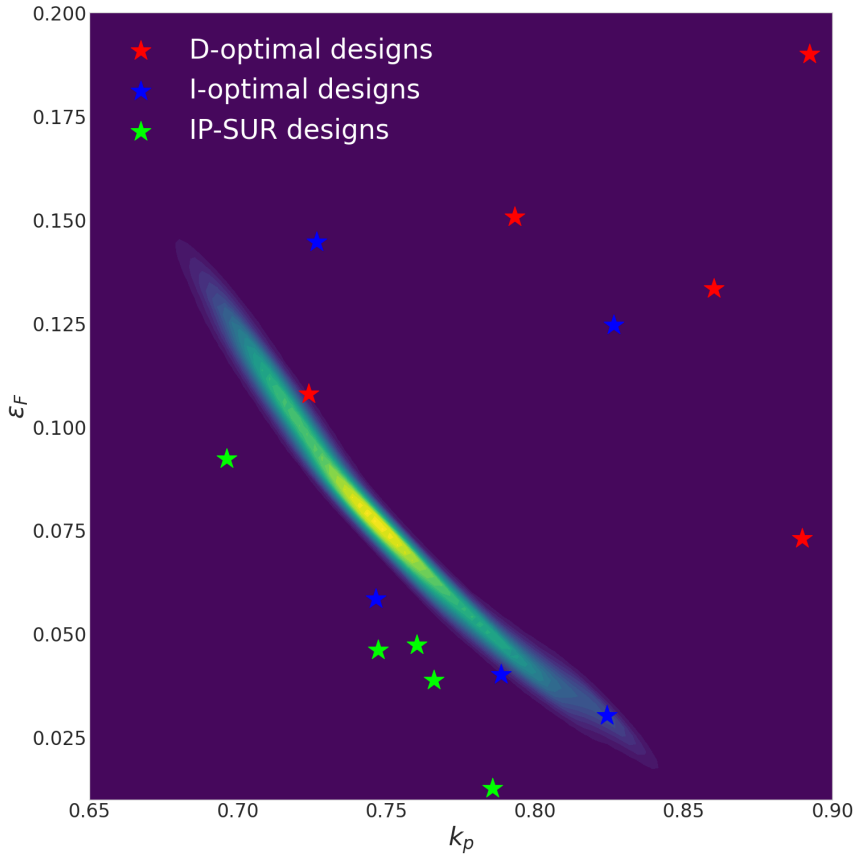

FIGURE 4.8: Design points obtained with various strategies. The posterior density $p_\infty(\cdot|\mathbf{y})$ is also displayed.

of the standard D-optimal and I-optimal designs and we can conclude that our methods are significantly better than standard optimal design strategies in the context of Bayesian inverse problems.

4.3.5 Comparison to a Bayes risk minimization strategy

Until now, our sequential design strategies were only compared to a naive strategy in which the design points are chosen uniformly on the state space. The main question is now, how do they compare to state-of-the-art strategies in Bayesian inverse problems? To answer that, we compare our methods to the one introduced briefly in Section 4.1.2.3. The loss function used for the criteria can be estimated with MCMC and we refer to [SN17] for more details on the methods involved. Applying the same procedure as before, CSQ and IP-SUR strategies are compared to this strategy. The results are displayed in Figures 4.9, 4.10 and 4.11 for each test case. The two methods offer overall similar performance with regard to the metrics investigated, though the IP-SUR method seems more reliable for the bimodal case. Besides, it is backed by a convergence guarantee which is not the case for the method of Sinsbeck *et al.*. For the latter, the supermartingale property seems unreachable.

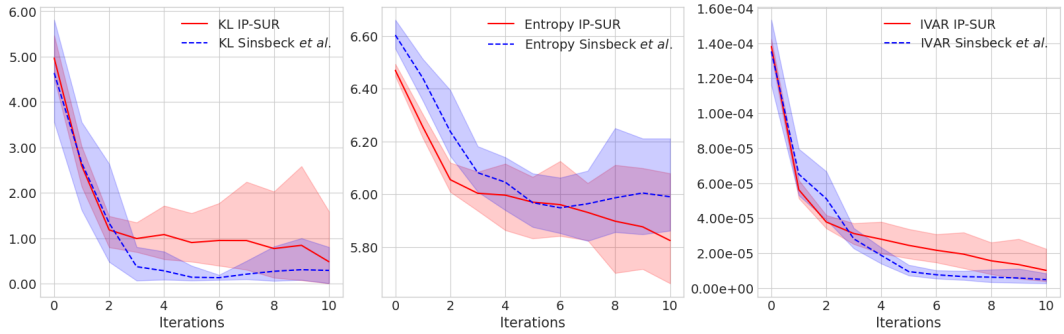

FIGURE 4.9: Comparison with Sinsbeck *et al.* - Banana test case.

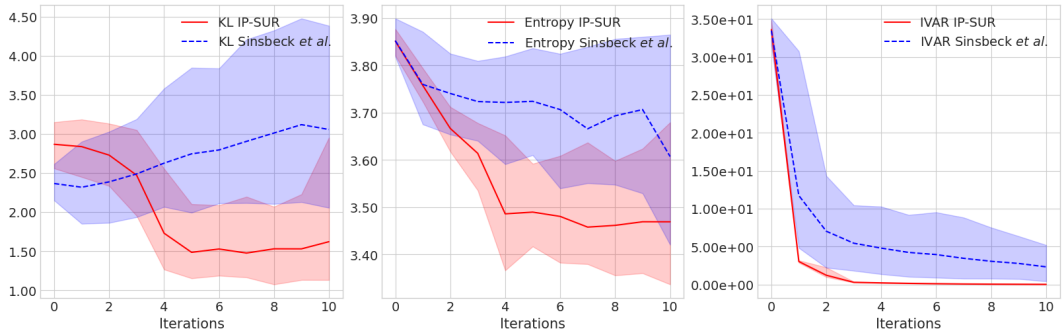

FIGURE 4.10: Comparison with Sinsbeck *et al.* - Bimodal test case.

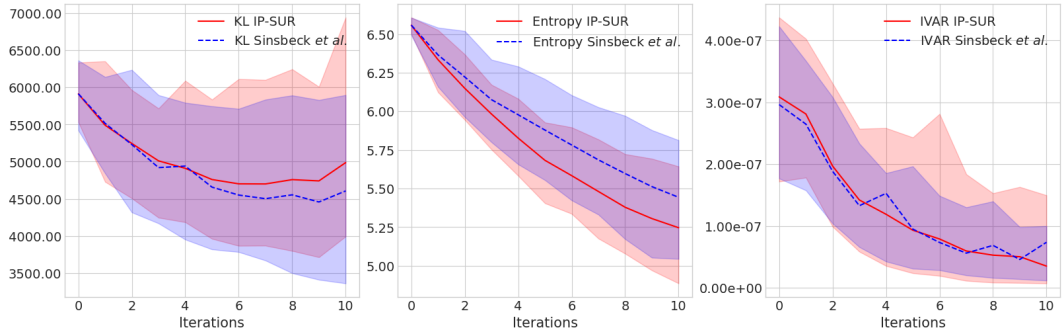

FIGURE 4.11: Comparison with Sinsbeck *et al.* - Neutronic test case.

4.3.6 An application for tempered IP-SUR

We also investigate the feasibility and performance of the TIP-SUR strategy. The focus is only on the point model test case here. The same methodology is applied to obtain various metrics of uncertainties, with different values of $\beta \in \{0.2, 0.5, 0.8\}$. These values are chosen arbitrarily. We also test a more subtle strategy in which we introduce a dependence of β w.r.t. n and we gradually increase the values of β_n from low values, starting at $\beta_1 = 0.1$ to 1 such that $\lim_{n \to +\infty} \beta_n = 1$. The following scheme is applied:

$$\beta_n = 1.0 - 0.9^n. \tag{4.100}$$

In Figure 4.12, we plot $\mathrm{KL}\left(p_n\left(\cdot\,|\mathbf{y}\right) \| p_\infty\left(\cdot\,|\mathbf{y}\right)\right)$ for each strategy. This figure does not allow us to conclude on the superiority of either IP-SUR or TIP-SUR strategies. The evolution of the KL is very similar for each case and is within the confidence interval. Further studies may be required to exhibit cases where the TIP-SUR strategy

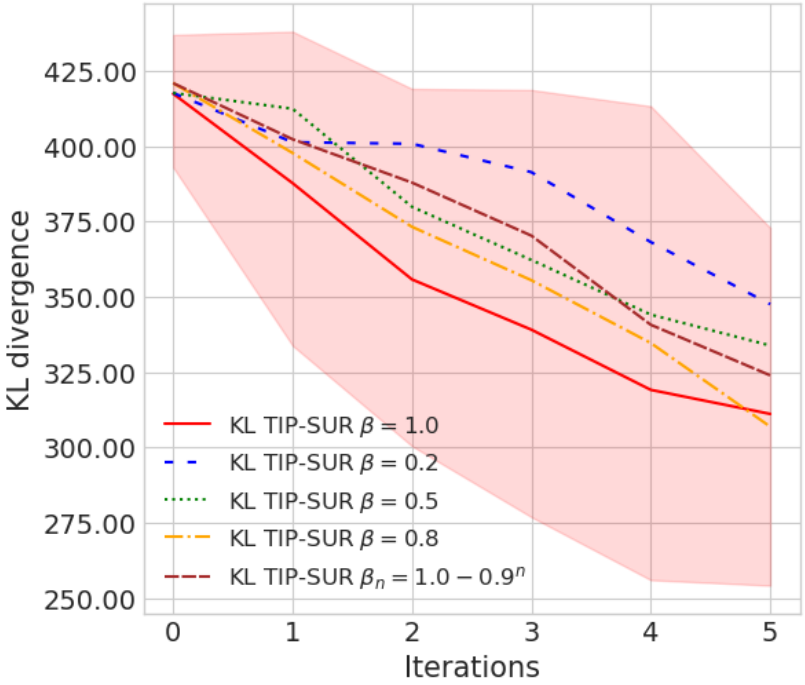

FIGURE 4.12: KL divergence obtained for the TIP-SUR strategy for various choices of β on the point model test case. Only the 95% confidence interval for $\beta = 1$ is shown.

could be superior to the IP-SUR design.

4.4 Conclusion

In this chapter, we derived two sequential design strategies for Gaussian process surrogate models in Bayesian inverse problems. The first one, known as the Constraint Set Query (CSQ) method, can be understood as a D-optimal design strategy constrained to a ball for the Mahalanobis distance and centered around the MAP.

The second strategy named IP-SUR, is an adaptation of the Stepwise Uncertainty Reduction (SUR) paradigm to the case of Bayesian inverse problems. The functional serving as a metric of uncertainty is a posterior-weighted IMSPE. We proved that this strategy is tractable for both scalar and multi-output GP models, provided we have access to an ergodic Markov chain whose stationary distribution is the posterior distribution. Moreover, it comes with a guarantee of convergence for the metric of interest, which is otherwise not trivially acquired for sequential design strategies. Both strategies have shown increased performance in the various applications presented compared to standard D-optimal, I-optimal, and naive designs. They are also comparable to the Bayes risk minimization strategy from [SN17].

Finally, we also showed that both the prior-weighted IMSPE and the IP-SUR strategies can be interpreted as specific cases of a broader strategy, named tempered IP-SUR (TIP-SUR). The metric of uncertainty for the TIP-SUR is a likelihood-weighted IMSPE with a temperature exponent $\beta \in [0, 1]$ on the likelihood. It comes with the same convergence guarantee as IP-SUR and could be advantageous for posteriors with thin support, though further analysis is required to confirm this intuition.

Chapter 5

Joint analysis of neutron and gamma correlations

In this chapter, we discuss the possibility of improving our uncertainty estimates by including the gamma correlations in our study. For that purpose, we need to adapt the inverse problem resolution introduced in previous chapters. The inclusion of gamma correlations follows a similar methodology, though it comes with its own challenges which are discussed in detail throughout this chapter.

In the first section of this chapter, we describe how to study gamma noise in zero-power systems with MCNP. We then describe how to include gamma correlations in our inverse problem and present the newly developed surrogate models. Finally, we make use of the sequential design strategies developed in the previous chapter to improve these models. We conclude by applying the methodology developed to the SILENE test case.

5.1 Gamma correlations in MCNP

In this section, we focus on the data collection process for gamma noise measurements. This is conducted within MCNP (see Section 1.4.8), with a similar approach to what was done for the study of neutron correlations.

5.1.1 PTRAC files for gamma correlations

Our first task is to generate Monte Carlo simulations including both neutron and gamma reactions. Then we would like to obtain a time list file containing the instants of detections of gamma rays, as was done for neutron correlations.

MCNP has the capability of performing particle transport for both neutrons and gammas at the same time. There is no exterior gamma source in the medium, all the gammas come from fissions and other nuclear reactions (mainly (n, γ) reactions). The gamma detector is not modeled. Instead, we record all gamma captures occurring in the fissile region. This is an oversimplification that will yield very high efficiency and is unattainable in practical experiments. However, our goal in this chapter is not to provide the highest quality model, but rather it is to prove the feasibility and potential gain in bringing gamma correlations into fissile matter identification techniques. For a robust benchmarking of the methodology on real-world applications, it would be necessary to consider a more realistic model in which a gamma detector is modeled via MCNP.

To build a dataset for our gamma surrogate models, we apply the same methodology as described in Chapter 2: starting from a reference configuration, the geometry of the problem (and the source) is randomly modified to provide a new data instance. However, we would like to obtain a dataset recording both neutron and gamma observations here. Since MCNP does not provide PTRAC files for both neutron and gamma reactions from one simulation, each instance requires two identical simulations, in which the random seed is kept identical, and where we record either the neutron detections or the gamma detections.

One of the first difficulties we ran into was the sheer number of gamma events occurring in the simulations. The average number of gammas produced by induced fissions in MCNP is $\overline{\mu} = 5.1$. Each fission produces much more gammas than neutrons. On top of that, a large number of (n, γ) reactions occur in the medium, leading to a large source term. The number of gamma events is thus generally far greater than that of neutrons. To give an idea of the amount of data collected, for a single simulation the neutron PTRAC file has a size of 10 MB while the gamma file has a size of 1.2 GB. Of course, the size of the PTRAC files varies depending on the simulation, it is larger for larger multiplication factor k_p. However, the amount of gamma events limits the number of particles sampled in the MCNP simulations to keep reasonably small files. Thus the total number of neutrons sampled in each simulation is 5×10^5. For the dataset presented in Chapter 2, we used 1×10^6 neutrons per simulation. One can thus expect slightly more noisy neutron observations.

5.1.2 Gamma Feynman moments

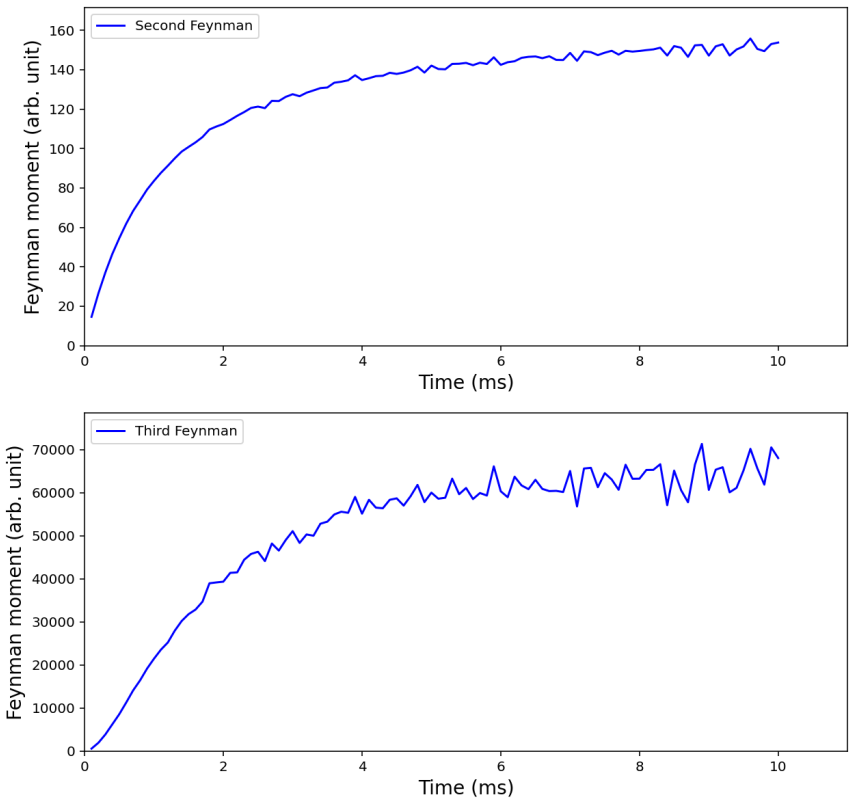

FIGURE 5.1: Second and third Feynman moments obtained by sequential binning.

Once the PTRAC files are available, one can post-process them to obtain the Feynman moments for the gammas. Similarly to what was done for the neutrons, we can estimate the second and third Feynman moments, denoted respectively by $Y_\infty^{(\gamma)}$ and $X_\infty^{(\gamma)}$, using either sequential or filtered triggered binning. Filtered binning is used for the training cases, while sequential binning is used to obtain the direct model observations which are used for the inverse problem resolution (see sections 1.4.4.2 and 1.4.4.3).
An example of the Feynman curves obtained by sequential binning is displayed in Figure 5.1. One may notice the longer asymptotic time $T_\infty \simeq$ 30ms compared to the neutron moments, which may be caused by the different reactions involved and the greater multiplicity.

5.1.3 Comparison to the point model equations

Before delving into the inverse problem resolution, we would like to compare our data to the point model equations for the gammas which are given in (1.130), (1.131) and (1.132). We recall that these equations refer to the singles, doubles, and triples detection rates (denoted respectively by $R^{(\gamma)}$, D_γ, and T_γ) and not the Feynman moments. These can be obtained from the PTRAC files using analogs of the estimators (1.122) and (1.123). The prediction errors of the point model equations are quantified in terms of NMAE and NRMSE over the whole dataset. They are summarized in Table 5.1.

TABLE 5.1: Prediction errors for the gamma point model equations.

	$R^{(\gamma)}$	D_γ	T_γ
NMAE	0.12	0.52	0.87
NRMSE	0.14	0.61	1.35

The gamma point model has poor predictive capabilities. Many possible factors could explain these large errors. Firstly, the nuclear data multiplicity for the gammas is not well-known. The mean values $\overline{\mu}$ and $\overline{\mu}_s$ are provided by MCNP in our work. However, we do not have access to the factorial moments. We assumed that the gamma Diven factors are roughly constant (this is the case for neutrons), and then used the gamma Diven factors calculated in [PP05]. The uncertainties on those parameters are thus considerable. Moreover, we are very far away from the idealized conditions of the point model framework, which should also hinder its predictive capabilities. Our current version of MCNP may not be well-suited to study gamma correlations because of the unknown behavior of the gamma multiplicity data. An alternative could be the MCNP-PoliMi code [PPM03], which was developed for neutron noise analysis and thus better describes the neutron and gamma multiplicity data.

Consequently, the point model for the gamma correlations seems particularly incompatible with the data obtained. However, it may still be used as a mean function of a GP surrogate model. Even though it displays poor predictive capabilities, it is still more informative than a constant mean function.

5.2 Neutron and gamma inverse problem

In this section, we discuss the possibility of including gamma noise measurements in our inverse problem resolution, with two different approaches.

5.2.1 Sequential approach

Our first method uses the Bayesian framework to leverage the additional knowledge brought by gamma correlations. Indeed, the prior distribution provides an easy way to incorporate new data in a Bayesian model. Throughout this thesis, we only considered non-informative priors. We did not focus on objective or reference priors since we generally have enough observations to keep the prior choice relatively unimportant.
However, now that we have additional information in the form of gamma correlations, the prior can be used to incorporate this new knowledge. This has led us to propose a sequential approach for the neutron/gamma inverse problem, which is described hereafter.

Consider some neutron observations $\mathbf{y} = (y^{(k)})_{1\leq k\leq N}$ and some gamma observations $\mathbf{y}^{(\gamma)} = (y^{(k,\gamma)})_{1\leq k\leq N_\gamma}$ for $N, N_\gamma \geq 1$. Throughout this chapter, the neutron observations are as usual (R, Y_∞, X_∞) and the gamma observations are $(R^{(\gamma)}, Y_\infty^{(\gamma)}, X_\infty^{(\gamma)})$. We assume that the observations are related to some input x and $x^{(\gamma)}$. However, x and $x^{(\gamma)}$ are not necessarily in the same design space. As an example $x^{(\gamma)}$ includes the gamma efficiency ε_γ while x do not. Thus x and $x^{(\gamma)}$ live in two different input spaces $\mathcal{X}$ and $\mathcal{X}^{(\gamma)}$. Here the inputs of interest for x are the same as in previous chapters, such that $x = (k_p, \varepsilon, S, x_s, \varepsilon_A, \Phi, J)$ and $\mathcal{X} \subset \mathbb{R}^7$. For $x^{(\gamma)}$, based on the point model equations for the gamma correlations (1.130), (1.131) and (1.132), we choose $x^{(\gamma)} = (k_p, S, x_s, M_\gamma, \varepsilon_\gamma)$ and $\mathcal{X}^{(\gamma)} \subset \mathbb{R}^5$.

In this chapter, we focus on Gaussian process surrogate models. We assume that GP surrogates are available for both the neutron and the gamma model. They are denoted respectively by $f_s^{(n)}$ and $f_s^{(\gamma)}$. They replace the unknown direct models such that the observations are:

$$y^{(k)} = f_s^{(n)}(x) + \varepsilon_k^{(n)} \text{ for } 1 \leq k \leq N \tag{5.1}$$

$$y^{(k,\gamma)} = f_s^{(\gamma)}(x^{(\gamma)}) + \varepsilon_k^{(\gamma)} \text{ for } 1 \leq k \leq N_\gamma \tag{5.2}$$

where $\varepsilon_k^{(n)} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{C}_{\text{obs}}^{(n)}\right)$ and $\varepsilon_k^{(\gamma)} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{C}_{\text{obs}}^{(\gamma)}\right)$ are the iid random variables for the observational noise. For $x \in \mathcal{X}$ and $x^{(\gamma)} \in \mathcal{X}^{(\gamma)}$, we also introduce a neutron likelihood $L^{(n)}(\mathbf{y}|x)$ and a gamma likelihood $L^{(\gamma)}\left(\mathbf{y}^{(\gamma)}|x^{(\gamma)}\right)$ which are defined with the general likelihood (2.76) introduced in Chapter 2.

To solve our inverse problem, we proceed as follows. Starting from a prior distribution with density $p(x)$ for $x \in \mathcal{X}$, we solve the Bayesian inverse problem for the neutron correlations only, using the likelihood $L^{(n)}(\mathbf{y}|x)$. We thus obtain a posterior distribution with density $p^{(n)}(x|\mathbf{y})$ for $x \in \mathcal{X}$. Since the posterior is obtained by MCMC sampling, its density is derived from the Markov chain by a Gaussian kernel density estimation.
Then, this posterior distribution is used as a prior in a second inverse problem where we introduce the gamma observations. However, because the neutron and gamma

models have different input spaces the priors must be adjusted accordingly. One can notice that x and $x^{(\gamma)}$ share three parameters: k_p, S and x_s. Thus we define a prior on the parameters (k_p, S, x_s) with the marginals of the posterior distribution $p^{(n)}(\,\cdot\,|\mathbf{y})$. Then we affect a uniform prior for the two missing inputs $(M_\gamma, \varepsilon_\gamma)$. For $x^{(\gamma)} \in \mathcal{X}^{(\gamma)}$, the prior density for the gamma inverse problem, which is denoted by $p^{(\gamma)}(x^{(\gamma)})$ is then given by:

$$p^{(\gamma)}(x^{(\gamma)}) = p(M_\gamma, \varepsilon_\gamma) \int_{\varepsilon_F, \varepsilon_A, \Phi, J} p(k_p, \varepsilon_F, S, x_s, \varepsilon_A, \Phi, J|\mathbf{y}) d\varepsilon_F d\varepsilon_A d\Phi dJ \tag{5.3}$$

where $p(M_\gamma, \varepsilon_\gamma)$ denotes the uniform prior density on a given subset of $\mathbb{R}^2$ for the inputs M_γ and ε_γ. One could also derive a posterior distribution for the intersection of x and $x^{(\gamma)}$ in a similar fashion.
With this new prior distribution for the gamma inverse problem, we can compute a second posterior distribution $p^{(\text{seq})}$ whose density is given, for $x^{(\gamma)} \in \mathcal{X}^{(\gamma)}$ by:

$$p^{(\text{seq})}(x^{(\gamma)}) \propto p^{(\gamma)}(x^{(\gamma)}) L^{(\gamma)}\left(\mathbf{y}^{(\gamma)}|x^{(\gamma)}\right). \tag{5.4}$$

We highlight that this methodology is applicable in the reversed order, in which the gamma inverse problem is first solved before adding the neutron observations.
This final posterior distribution thus regroups the knowledge of neutron and gamma correlations. It requires surrogate models for both the neutron and gamma direct models. However, it has one main drawback: it treats the gamma and neutron problem sequentially and thus does not account for any correlations between the phenomena. This is a simplification since we expect the gamma and neutron models to be strongly correlated. The observations $\mathbf{y}$ and $\mathbf{y}^{(\gamma)}$ should not be treated independently from one another. This is why we describe a second approach in the next section.

5.2.2 Joint resolution

To account for correlations between neutron and gamma observations, one could try to solve the joint inverse problem encompassing both types of observations.
We introduce the notations $x^{(n,\gamma)} = (k_p, \varepsilon_F, S, x_s, \varepsilon_A, \Phi, J, M_\gamma, \varepsilon_\gamma) \in \mathcal{X}^{(n,\gamma)} \subset \mathbb{R}^9$ and $p(x^{(n,\gamma)})$ the uniform prior density on $\mathcal{X}^{(n,\gamma)}$. Let $N_{n,\gamma}$ be the number of observations for the joint inverse problem. If we have $N_{n,\gamma} = N = N_\gamma$, the observations are denoted by $\mathbf{y}^{(n,\gamma)} = (\mathbf{y}, \mathbf{y}^{(\gamma)})^T \in \mathbb{R}^{N_{n,\gamma} \times 6}$. If not, we consider $N_{n,\gamma} = \gcd(N, N_\gamma)$ and we can come back to the previous case by forming batches of observations with size $N/N_{n,\gamma}$ for the neutrons and $N_\gamma/N_{n,\gamma}$ for the gammas. We then average over each batch and divide $\mathbf{C}_{\text{obs}}$ by $N/N_{n,\gamma}$ and $\mathbf{C}_{\text{obs}}^{(\gamma)}$ by $N_\gamma/N_{n,\gamma}$.

Consider a GP surrogate model $f_s^{(n,\gamma)}$ serving as an emulator for the joint direct model. The observations are given by the following statistical model:

$$y_k^{(n,\gamma)} = f_s^{(n,\gamma)}\left(x^{(n,\gamma)}\right) + \varepsilon_k^{(n,\gamma)} \tag{5.5}$$

for $1 \leq k \leq N_{n,\gamma}$ and where $\varepsilon_k^{(n,\gamma)} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{C}_{\text{obs}}^{(n,\gamma)}\right)$ where $\mathbf{C}_{\text{obs}}^{(n,\gamma)}$ is obtained by the empirical covariance estimator applied to $\mathbf{y}^{(n,\gamma)}$.
With this surrogate model, one can then define a likelihood $L^{(n,\gamma)}\left(\mathbf{y}^{(n,\gamma)}|x^{(n,\gamma)}\right)$ for $x^{(n,\gamma)} \in \mathcal{X}^{(n,\gamma)}$ with (2.76). Finally, we obtain a posterior distribution $p^{(n,\gamma)}(\,\cdot\,|\mathbf{y}^{(n,\gamma)})$

which accounts for the correlations between neutron and gamma observations and whose density is given for $x^{(n,\gamma)} \in \mathcal{X}^{(n,\gamma)}$ by:

$$p^{(n,\gamma)}(x^{(n,\gamma)}|\mathbf{y}^{(n,\gamma)}) \propto L^{(n,\gamma)}\left(\mathbf{y}^{(n,\gamma)}|x^{(n,\gamma)}\right) p(x^{(n,\gamma)}). \tag{5.6}$$

Though this posterior distribution is theoretically more accurate, it requires a surrogate model of higher dimension, able to emulate both neutron and gamma correlations. The higher dimension for both the input and output spaces may hinder the predictive capabilities of this model, which would, in turn, make this approach less reliable.

5.2.3 Gamma surrogate models

In this chapter, we will investigate both of the approaches developed in the previous paragraphs. Consequently, we require a gamma surrogate model $f_s^{(\gamma)}$ (**GSM**) and a joint surrogate model $f_s^{(n,\gamma)}$ (**JSM**) able to predict both neutron and gamma correlations. We will describe briefly these two new models in this section.

Starting with the gamma model, we are using our new dataset of 232 instances to build a multi-output GP model. The model is based upon the Linear Model of Coregionalization described in Chapter 2, with 2 latent scalar GPs. The scalar GPs have anisotropic Matérn kernels with $\nu = 5/2$. This GP is built using the bias learning approach used previously for the neutron models.
The joint GP model is also an LMC model with 4 latent GPs, each having anisotropic Matérn kernels. However, due to the higher dimension, both for the input and output spaces, exact GP inference is too computationally expensive in this case. Indeed, we recall that the exact inference requires the inversion of a matrix of size Dn where D is the output dimension, and $n = 190$ is the number of training data. This amounts to a complexity $\mathcal{O}((Dn)^3)$ which makes the inference intractable.
Hence there is a need for sparse approximations to accelerate both training and inference. Sparse GP approximations were introduced in Section 2.1.3. The sparse variational Gaussian process (SVGP) approach, introduced in [Tit09], is used in this work. Our numerical developments are based on the *GPyTorch* package developed for Python [Gar+18].

The error metrics for the predictive means are shown in Table 5.2. One can see that the **NSM** and **GSM** models perform well. The errors on the gamma tend to be smaller due to the higher number of events in gamma correlations. However, the joint model **JSM** exhibits inferior predictive capabilities than these models. This can be explained by the higher dimensionality of the task for the **JSM** model, as well as by the sparse approximation. We also remind the reader that the dataset used in this chapter is not the same as the **SILENE dataset** from Chapter 2. Because we were limited by the size of the PTRAC files, this dataset is built with fewer simulated neutrons and the training data are thus more noisy than in Chapter 2. The performance of the **NSM** model is not to be compared with the performance of the models in Chapter 2.

We also looked at the coverage probabilities for all the models. They are plotted in Figure 5.2. Since the test set has only 42 instances, the coverage probabilities cannot be very accurate, but one can still notice that all models provide reasonable coverage probabilities. While the **JSM** model does show some tendency to overestimate uncertainty, it maintains satisfactory coverage probabilities, which is particularly noteworthy given its higher output dimension.

TABLE 5.2: Error metrics for the neutron model (**NSM**), gamma model (**GSM**) and joint model (**JSM**).

NSM	NMAE	NRMSE	Q^2
R	0.008	0.011	0.9997
Y_∞	0.027	0.038	0.9983
X_∞	0.051	0.153	0.9919
GSM	NMAE	NRMSE	Q^2
$R^{(\gamma)}$	0.004	0.006	0.9999
$Y_\infty^{(\gamma)}$	0.022	0.031	0.9953
$X_\infty^{(\gamma)}$	0.080	0.169	0.9773

JSM	NMAE	NRMSE	Q^2
R	0.012	0.016	0.9990
Y_∞	0.046	0.092	0.9854
X_∞	0.089	0.158	0.9937
JSM	NMAE	NRMSE	Q^2
$R^{(\gamma)}$	0.012	0.018	0.9992
$Y_\infty^{(\gamma)}$	0.016	0.023	0.9992
$X_\infty^{(\gamma)}$	0.025	0.047	0.9992

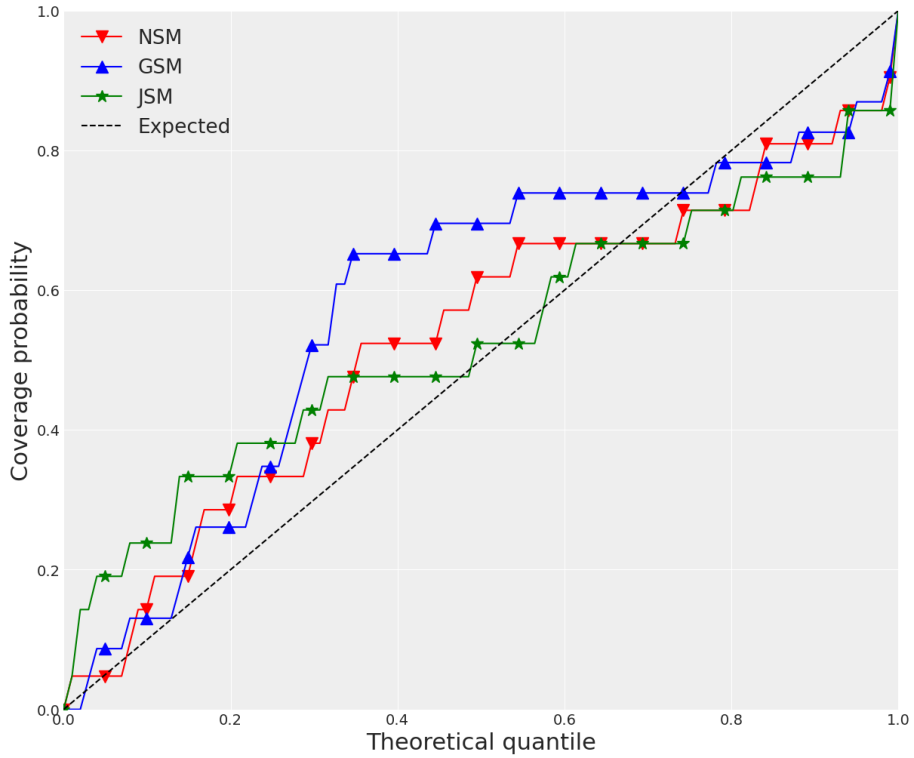

FIGURE 5.2: Coverage probabilities evaluated on the test set for **NSM**, **GSM** and **JSM**.

5.2.4 A first application

Our objective is now to apply the two proposed methods to a test case. In this chapter, we focus on the SILENE test case. More specifically, we will consider only the case with a fissile height of $h = 20$ cm. Since we do not have access to experimental gamma measurements, the gamma observations $\mathbf{y}^{(\gamma)}$ are obtained from numerical simulations with MCNP. We have a total of $N_\gamma = 16$ independent gamma observations. On the other hand, the neutron observations $\mathbf{y}$ are derived from the experimental time list files as was done previously.

As usual, the posterior distributions are obtained by MCMC sampling. The AM algorithm is used for the sequential approach, in which the two inverse problems are solved one after the other. However, for the joint posterior distribution $p^{(n,\gamma)}(\,\cdot\,|\mathbf{y}^{(n,\gamma)})$, we favor HMC-NUTS due to the higher dimension of the problem, for which it is

more suited.
To highlight the gain in information brought by the gamma correlations, we plot in Figure 5.3 the two-dimensional marginals for the inputs (k_p, S) for the distributions $p^{(\text{seq})}(\,\cdot\,|\mathbf{y}^{(n,\gamma)})$, $p^{(n,\gamma)}\left(\,\cdot\,|\mathbf{y}^{(n,\gamma)}\right)$ and $p^{(n)}\,(\,\cdot\,|\mathbf{y})$. As one can see from this figure, the

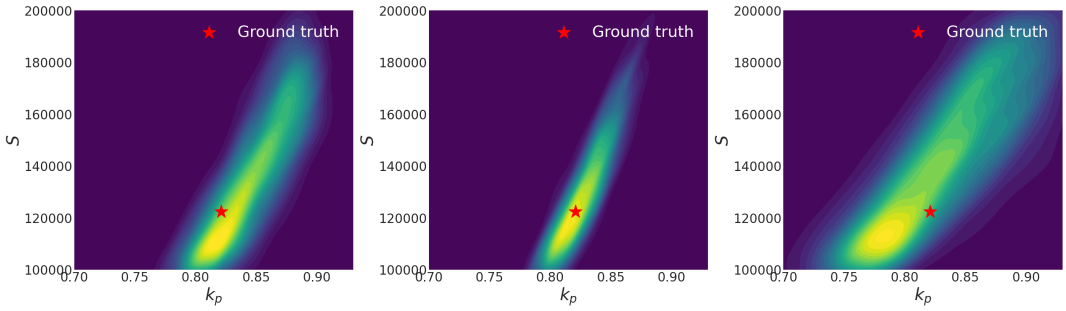

FIGURE 5.3: Two-dimensional marginals w.r.t. (k_p, S) of the posterior distributions obtained for the neutron inverse problem (left), the sequential approach (center) and the joint approach (right).

distribution is significantly narrower when we add the gamma correlations with the sequential approach. However, the joint approach yields a wider distribution, that lies further away from the ground truth. Looking back at the model performance in Table 5.2 and Figure 5.2, the joint model does have poorer prediction capabilities, which may impact the accuracy of the inverse problem resolution. However, we hope to resolve this limitation by efficiently adding more data points, with the help of the sequential design strategies derived in Chapter 4.

5.3 Sequential design strategy for the JGP model

5.3.1 SUR or CSQ design?

In Chapter 4, we presented two sequential design strategies specifically suited for Bayesian inverse problems. The IP-SUR strategy has a guarantee of convergence but is overall more costly than the CSQ strategy because it requires MCMC sampling at each iteration. We recommend using preferably CSQ designs when the cost of a call to the black-box code is of the same order of magnitude as the running times of the MCMC samplings. However, if the computer code has much longer running times than the generation of the Markov chain, we would advocate for adopting the IP-SUR strategy.
In this particular example, one iteration of the CSQ strategy, including the design point selection detailed later in this chapter, the MCMC simulations, and the GP update, took roughly 1 hour. On the other hand, one iteration of the IP-SUR strategy amounts to almost 3 hours of run time. For that reason, we will focus on the CSQ strategy in this application.
However, we expect both strategies to be difficult to apply to very high-dimensional problems. In our work, the input space has a rather small dimension and we can get away with both methods. For other applications in higher-dimensional spaces, we recommend screening the variables in the problem as much as possible to reduce the dimensionality.

5.3.2 Dealing with uncontrolled inputs

One of the obstacles encountered while adapting the sequential design strategies to our case is the lack of control over the inputs x. Indeed, in a neutronic Monte Carlo simulation, k_p is not a quantity that can be controlled. For simple geometries, one can expect how the k_p might evolve when changing the geometry of the problem, but we have no complete control over it. Similarly, ε_F is an output of the MC code. How can we apply our strategies to actively improve the surrogate models in that context?

We propose the following approach. Consider $x_* \in \mathcal{X}^{(n,\gamma)}$ the target design point recommended by the sequential design strategy (either CSQ or IP-SUR). Let us introduce a cost function $L(x, x_*)$ defined for $x \in \mathcal{X}^{(n,\gamma)}$ by:

$$L(x, x_*) = \sum_{j=1}^{p} \omega_j \left(x_j - x_{*,j}\right)^2 \tag{5.7}$$

where the $\omega_j \geq 0$ are weight factors such that $\sum_{j=1}^{p} \omega_j = 1$ and where $p = 9$ is the dimension of the input space. Ideally, we would like to reach $x = x_*$ which translates to $L(x, x_*) = 0$. Since we do not have complete control over k_p and ε_F, reaching $x = x_*$ appears difficult, though we can try to minimize L.
The weights $(\omega_j)_{1 \leq j \leq p}$ should be defined to account for the influence of the variables. Our first attempt was to define arbitrary weights based on our knowledge of the problem. While this may work, we refined our method by using sensitivity analysis to better select the weights. Recalling the sensitivity analysis from Chapter 2, one may define them using the calculated Sobol indices. Intuitively, one could increase the contributions of the outputs whose observational variances are low, since the inverse problem is more sensitive to these outputs. As an example, we choose the non-normalized weights ω_j defined by:

$$\omega_j = \sum_{i=1}^{D} s_{j,i} \frac{\overline{y^2}_i}{\sigma_i^2} \tag{5.8}$$

where $s_{j,i}$ is the uncorrelated first-order Sobol index associated to the j-th input and to the i-th output, σ_i^2 is the observational variance of the i-th output, for $1 \leq i \leq D$, and $\overline{y^2} = \frac{1}{N^{(n,\gamma)}} \sum_{k=1}^{N_{n,\gamma}} \left(y_k^{(n,\gamma)}\right)^2$. We then normalize the weights ω_j to have $\sum_{j=1}^{p} \omega_j = 1$.
With this empirical approach, we hope to put more weight on the outputs R and $R^{(\gamma)}$ which exhibit low relative variance.

Due to the addition of M_γ and ε_γ, a new sensitivity analysis study is needed. It is performed with the same method as in Chapter 2. However, we are dealing with a higher-dimensional model with only limited data. In Chapter 2, the dataset was split into a training set used to build the PCE models, and a test set used to estimate the Sobol indices. However, here we do not have enough data for those two tasks. We thus decided to create new synthetic data by fitting a Gaussian copula to the data. Knowing the marginal distributions of the dataset inputs $\mathbf{X}$ and the fitted copula, we can then sample new synthetic data which can be used to evaluate the Sobol indices. With this approach, the test set is not required and all the data can be used to build the PCE models. We still keep 10% of the data as a validation set to measure the performance of the PCE models (namely the NMAE and NRMSE). An introduction to copulas is presented in the following chapter. Since we want the weights ω_j of

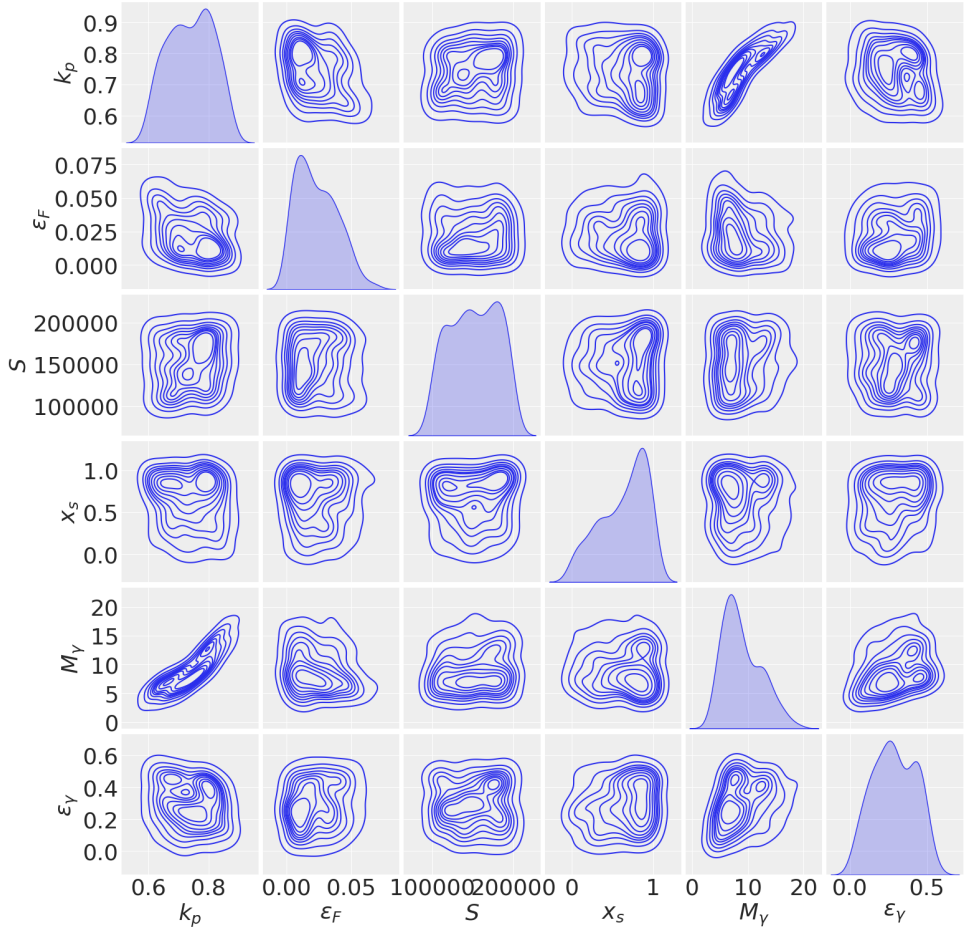

FIGURE 5.4: Marginal distributions and correlations in the original dataset.

the most influential parameters in priority, we decided to focus only on the most influential parameters. The inputs of interest in this sensitivity analysis study are thus restricted to $(k_p, \varepsilon_F, S, x_s, M_\gamma, \varepsilon_\gamma)$. In Figure 5.4, we represented the marginal distributions and correlations between the inputs in the original dataset. The same plot is given for a synthetic dataset of $n = 1000$ samples in Figure 5.5. One can see that the correlation structure is rather close to Figure 5.4 and the copula fit appears to be efficient.

The Sobol indices obtained are shown in Table 5.3. These results highlight the importance of the newly introduced M_γ and ε_γ, especially in relation to the gamma correlations.

TABLE 5.3: First-order Sobol indices.

$S_j^{(U)}$	R	Y_∞	X_∞	$R^{(\gamma)}$	$Y_\infty^{(\gamma)}$	$X_\infty^{(\gamma)}$
k_p	4.3×10^{-1}	5.7×10^{-1}	6.6×10^{-1}	4.1×10^{-4}	2.6×10^{-1}	2.9×10^{-1}
ε_F	6.8×10^{-1}	3.9×10^{-1}	7.4×10^{-1}	1.9×10^{-6}	2.1×10^{-5}	2.7×10^{-6}
S	5.7×10^{-2}	1.1×10^{-4}	7.2×10^{-4}	4.9×10^{-2}	4.4×10^{-5}	2.2×10^{-5}
x_s	6.5×10^{-2}	3.2×10^{-3}	1.5×10^{-3}	5.0×10^{-2}	2.5×10^{-3}	8.7×10^{-4}
M_γ	5.3×10^{-2}	2.6×10^{-2}	1.9×10^{-1}	1.8×10^{-1}	1.2×10^{-1}	3.5×10^{-1}
ε_γ	7.2×10^{-2}	7.9×10^{-3}	2.2×10^{-2}	4.8×10^{-1}	5.3×10^{-1}	4.7×10^{-1}

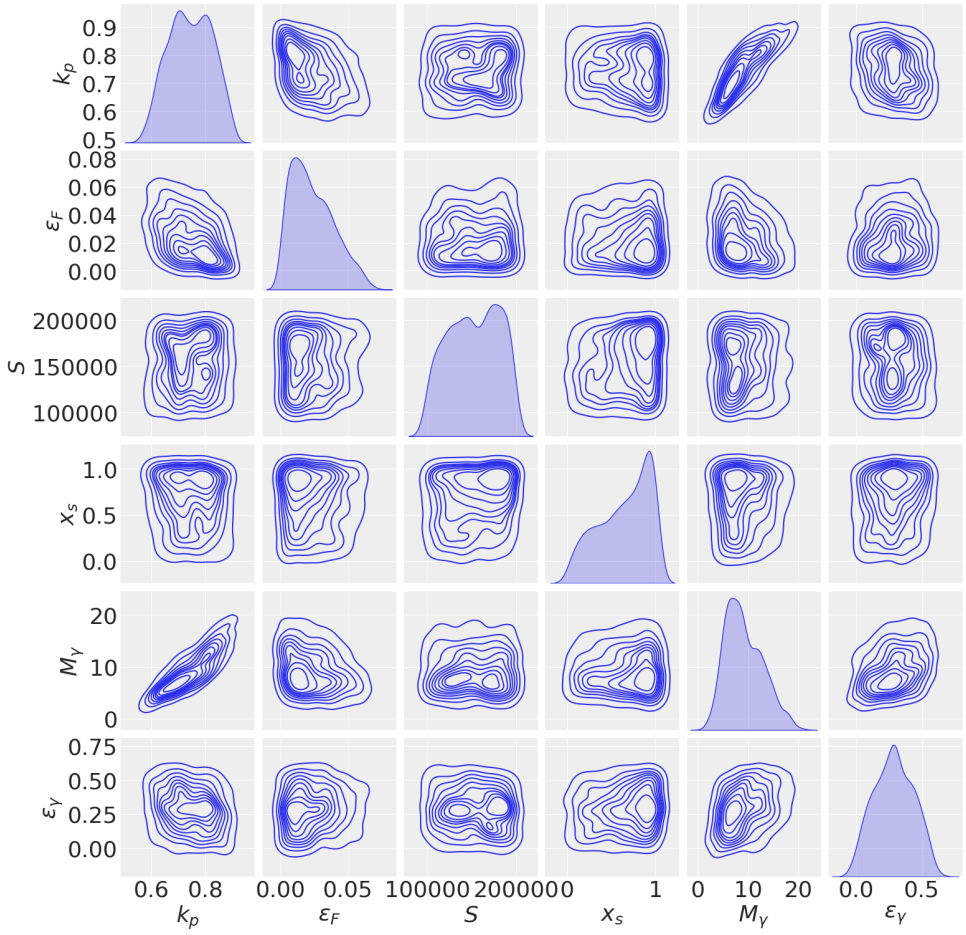

FIGURE 5.5: Marginal distributions and correlations in the synthetic dataset.

5.3.3 Updated surrogate models

Using the CSQ method, we update the joint model by adding 20 new design points. Each new simulation is conducted with a total of 5×10^5 simulated neutrons. The optimization step in CSQ is performed with dual annealing. We present in Table 5.4 the mean relative error between the target design points and the design points obtained.

TABLE 5.4: Mean relative error between the target optimal design points given by CSQ and the actual points found with MCNP.

	k_p	ε_F	M_γ	ε_γ
Rel. error (%)	3.6	71	63	40

The lack of control in the selection of the inputs is significant, with some parameters such as ε_F and M_γ displaying a relative error superior to 50%. However, we manage to get a satisfying precision on k_p which appears as one of the main contributors to the joint inverse problem.

The new metrics for the improved joint model are shown in Table 5.5. The coverage probabilities can be seen in Figure 5.6.

TABLE 5.5: Error metrics for the previous **JSM** and the updated **JSM**.

JSM	NMAE	NRMSE	Q^2
R	0.012	0.016	0.9990
Y_∞	0.046	0.092	0.9854
X_∞	0.089	0.158	0.9937
JSM	NMAE	NRMSE	Q^2
$R^{(\gamma)}$	0.012	0.018	0.9992
$Y_\infty^{(\gamma)}$	0.016	0.023	0.9992
$X_\infty^{(\gamma)}$	0.025	0.047	0.9992

Updated **JSM**	NMAE	NRMSE	Q^2
R	0.007	0.009	0.9997
Y_∞	0.035	0.045	0.9965
X_∞	0.096	0.317	0.9337
Updated **JSM**	NMAE	NRMSE	Q^2
$R^{(\gamma)}$	0.009	0.013	0.9996
$Y_\infty^{(\gamma)}$	0.016	0.023	0.9992
$X_\infty^{(\gamma)}$	0.030	0.058	0.9988

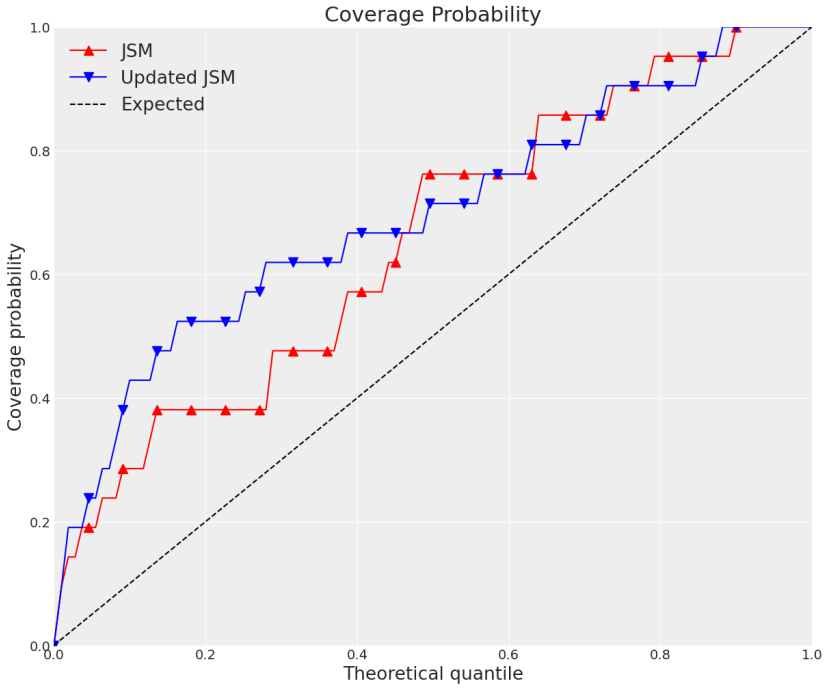

FIGURE 5.6: Coverages probabilities for the **JSM** and updated **JSM** models.

To highlight the reduction in the epistemic uncertainty in the model, we also compared the mean determinant of the predictive covariances over the test set (MCD). For a surrogate model with predictive distribution $f_s(x) \sim \mathcal{N}\left(\overline{f_s}(x), \mathbf{C}_s(x)\right)$, this quantity is defined by:

$$\text{MCD} = \frac{1}{N_*} \sum_{i=1}^{N_*} |\mathbf{C}_s(X_{*,i})|. \tag{5.9}$$

The ratio of these quantities, obtained for the previous joint model and the updated model is $\frac{\text{MCD}_{\text{old}}}{\text{MCD}_{\text{new}}} \simeq 2.3$. The determinant of the predictive covariances is reduced by a factor of more than 2. Besides, thanks to the CSQ strategy, this uncertainty reduction is likely more prominent in regions of high posterior density.

5.3.4 Improved posterior distribution

Using the newly enriched joint GP model, we can sample a new joint posterior distribution with HMC-NUTS and $L = 1 \times 10^5$ samples. The marginal distribution

for (k_p, S) is displayed in Figure 5.7 along with the ones obtained for the sequential approach and with neutron correlations only. The drastic improvement of the joint

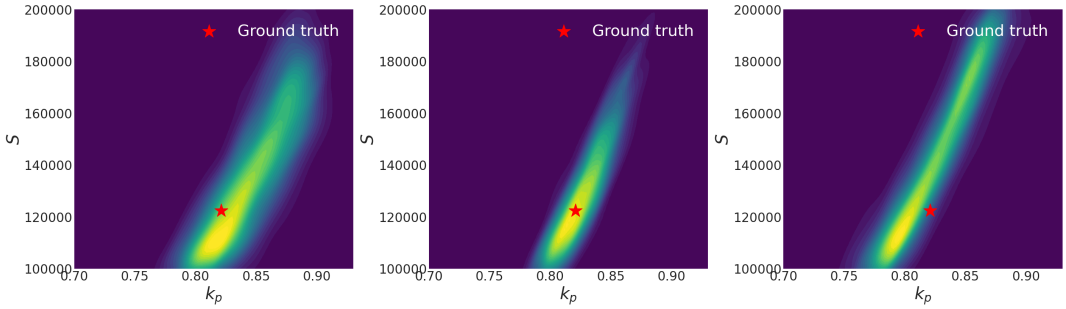

FIGURE 5.7: Two-dimensional marginals w.r.t. (k_p, S) for the neutron inverse problem (left), the sequential approach (center) and the joint approach with the updated **JSM** (right).

surrogate model is visible compared to Figure 5.3. However, the posterior distribution remains more spread out than the one obtained with the sequential approach. More design points are needed to further improve the joint surrogate model. As of now, the sequential approach appears to be more efficient in reducing the uncertainties. However, sequential design strategies coupled with a larger training set could make the joint model viable. The sequential designs are also hindered by the lack of control over the inputs in the design space and possibly by the sparse approximation in the **JSM** model. Developments in this direction would greatly benefit this approach.

5.4 Conclusion

In this chapter, we explored gamma correlations and their potential to diminish uncertainties in the inverse problem. We have seen that they can provide additional information, thus mitigating the uncertainties in the estimation of unknown nuclear parameters. However, using gamma correlations increases the dimensionality of the problem which makes the design of the surrogate models more complex. To circumvent this issue, one can instead look at two successive inverse problems though this approach ignores the correlations between gamma and neutron observations. If we wish to treat the joint inverse problem, we need to carefully choose the design points. Specifically, we highlighted the improvements brought by the sequential design strategies discussed in the previous chapter, by applying them to this joint inverse problem. Nonetheless, the joint approach still yields wider posterior distributions that are likely linked to a lack of training data for the joint surrogate model. Further analyses are needed in that regard.

The use of gamma correlations in the identification of fissile matter is promising but requires further development to ensure the approach's robustness. Notably, this chapter did not address uncertainties in nuclear data. The evaluation of neutron multiplicity data has seen significant improvements in recent years, for example with the developments of new fission models such as FREYA [VRV15]. The gamma multiplicity on the other hand has received less attention, necessitating further development to achieve a comprehensive and reliable uncertainty quantification framework for scenarios involving gamma correlations. Moreover, gamma correlations are particularly sensitive to data uncertainties due to the higher number of detection events. Consequently, gamma observations tend to exhibit seemingly lower noise levels, though

uncertainties may be concealed within the nuclear data. Furthermore, this study relied exclusively on numerical simulations. Therefore, validating the robustness of our approach using real-world gamma measurements is essential for ensuring dependable uncertainty quantification.

Chapter 6

Inverse problem learning

In this chapter, the objective is to develop a framework to circumvent the large computational cost of MCMC when solving an inverse problem. In that regard, one could think of learning the posterior distribution of nuclear parameters directly for a given set of observations and the observation noise covariance matrix. To build a predictive model that provides such a posterior distribution, two preliminary requirements must be met.
First of all, the posterior distributions should be approximated by parametric distributions such that one only has to predict the distribution parameters. The predictive model output is thus reduced to a finite dimensional prediction making it much more reasonable. Then, the predictive model must be trained on a dataset representative of the applicative cases. These two steps are discussed throughout this chapter. We emphasize that this work is motivated by the need for immediate estimation of the nuclear parameters, even if it is detrimental to the quality and reliability of the estimation and uncertainty quantification.

6.1 Quantile-parametrized distributions

6.1.1 Generalized lambda distributions

Let us first discuss the scalar case, where the posterior distribution is one-dimensional. The most simple approach would be to consider that the posterior follows a given class of distributions, for example, a Gaussian such that $p(x|\mathbf{y}) \sim \mathcal{N}\left(\mu, \sigma^2\right)$ and then fit the parameters of the Gaussian accordingly with a method of moments for instance. To reflect various shapes of posterior distributions, the class of distributions considered must be broader than just Gaussian distributions and that is why our focus is on quantile distributions, which are distributions defined by their quantile functions.

Definition 6.1.1. *Consider a real-valued random variable X with cumulative distribution function F_X. The quantile function of X denoted by $Q\colon (0,1) \to \mathbb{R}$ is defined for $p \in (0,1)$ by:*

$$Q(p) = \inf\{x \in \mathbb{R} \; s.t. \; F_X(x) \geq p\} \tag{6.1}$$

Definition 6.1.2. *Let X be a random variable with quantile function Q. If the quantile function Q is differentiable, then the quantile density function q exists and is defined as the derivative of the quantile function Q. For $p \in (0,1)$:*

$$q(p) = Q'(p) \tag{6.2}$$

Proposition 6.1.1. *Let X be a random variable with quantile function Q and with density f. Then the quantile density function q exists and its given for $p \in (0,1)$ by:*

$$q(p) = \frac{1}{f(Q(p))} \tag{6.3}$$

The Lambda Tukey distribution was first introduced in [HJ+47] and is defined as follows.

Definition 6.1.3. *Let $\lambda \in \mathbb{R}$ be a shape parameter and $t \in \mathbb{R}$ and $s \in \mathbb{R}^+$ be respectively location and scale parameters. The Tukey Lambda distribution denoted $\mathcal{T}(\lambda, t, s)$ is the distribution defined by its quantile function $Q_{\lambda,t,s}$ such that for $p \in (0,1)$:*

$$Q_{\lambda,t,s}(p) = \begin{cases} t + \frac{s}{\lambda}\left(p^\lambda - (1-p)^\lambda\right), & \text{if } \lambda \neq 0 \\ t + s\log\left(\frac{p}{1-p}\right), & \text{if } \lambda = 0 \end{cases} \tag{6.4}$$

Proposition 6.1.2. *The quantile density function of the Tukey lambda distribution $\mathcal{T}(\lambda, t, s)$ is given for $p \in (0,1)$ by:*

$$q(p) = s\left(p^{\lambda-1} + (1-p)^{\lambda-1}\right) \tag{6.5}$$

The Tukey Lambda distribution can reflect many common distributions such as Cauchy, uniform, logistic, or Gaussian distributions. The probability density function of the Tukey lambda distribution cannot be obtained analytically but can easily be obtained numerically. A plot of the pdf for different values of the shape parameter λ is displayed in Figure 6.1 for a standard Tukey distribution ($s = 1$ and $t = 0$).

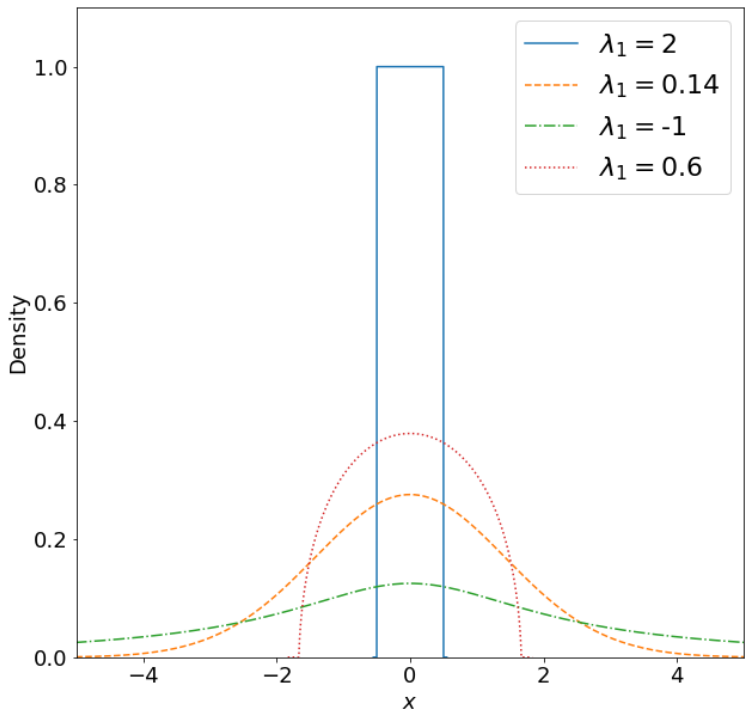

FIGURE 6.1: Density plots for different shape parameters of Tukey Lambda distributions, with $t = 0$ and $s = 1$.

However, all the Tukey lambda distributions are symmetric. To broaden the possible distributions, variants of Tukey lambda distributions have been introduced for example in [RS74]. In this work, the focus is on the generalized lambda distribution (GLD) class introduced in [Fre+88].

Definition 6.1.4. *Generalized lambda distributions are a class of distributions defined by two shape parameters $\lambda_1 \neq 0, \lambda_2 \neq 0$ as well as a location parameter $t \in \mathbb{R}$ and a scale parameter $s \in \mathbb{R}^+$. The generalized lambda distribution with parameters $(\lambda_1, \lambda_2, t, s)$ denoted $\mathcal{G}(\lambda_1, \lambda_2, t, s)$ is defined implicitly for by its quantile function $Q_{\lambda_1,\lambda_2,t,s}$ such that for $p \in (0, 1)$:*

$$Q_{\lambda_1,\lambda_2,t,s}(p) = t + s\left(\frac{p^{\lambda_1}}{\lambda_1} - \frac{(1-p)^{\lambda_2}}{\lambda_2}\right) \tag{6.6}$$

The generalized lambda distributions (GLD) can describe a wider variety of distributions since they are not restricted to symmetric distributions. They are often used to model unknown distributions with only a handful of parameters [CSW12; ARRE08]. For instance, in [ZS20], GLDs are used as response distributions of stochastic simulators, and their parameters are modeled by polynomial chaos expansion in [ZS21].

An example of the pdf of some GLDs is displayed in Figure 6.2. The pdf plots are non-symmetrical and are thus more adapted to model unknown distributions.

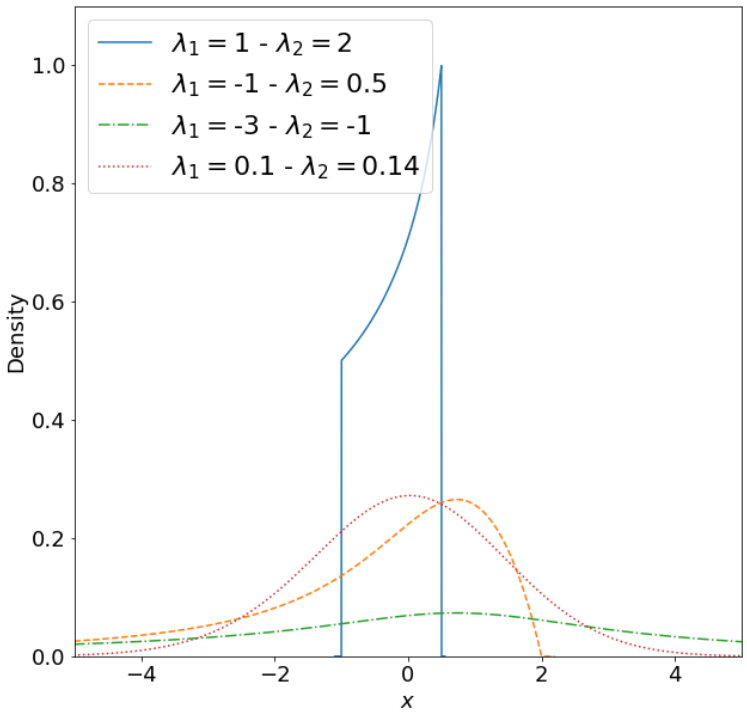

FIGURE 6.2: Density plots for different shape parameters of GLD distributions, with $t = 0$ and $s = 1$.

6.1.2 Multivariate quantile distributions and copula

Now let us consider a D-dimensional posterior distribution. How can we generalize the GLDs to such higher dimensional space, while enclosing correlations between the

dependent variables? To answer this question, we need to provide a brief introduction to copulas.

Definition 6.1.5. *Consider a D-dimensional real-valued random variable X, with $D > 1$. The cumulative distribution function F_X (CDF) is extended in this multivariate setting. It is defined, for $x = (x_1, ..., x_D) \in \mathbb{R}^D$ by:*

$$F_X(x) = \mathbb{P}\left(X_1 \leq x_1, ..., X_D \leq x_D\right). \tag{6.7}$$

Definition 6.1.6 (Copula)**.** *A copula in a D-dimensional space is a cumulative distribution function C defined on the hypercube $[0,1]^D$ with uniform marginals. In other words, $C\colon [0,1]^D \to [0,1]$ is a copula if and only if it verifies the following properties:*

- *For any $p \in [0,1]^D$ and for any $1 \leq i \leq D$ the function $C_i\colon x \mapsto C(p_1, ..., x, ..., p_D)$ where the i-th component of p is replaced by $x \in [0,1]$ is non-decreasing.*
- *The marginals of C are uniform, meaning that for $u \in [0,1]^D$ and $1 \leq i \leq D$, $C(1, ..., 1, u_i, 1, ..., 1) = u_i$.*
- *The probability of U being in the rectangle $[a_1, b_1] \times ... \times [a_D, b_D]$ where $a_i \leq b_i$ for all $1 \leq i \leq D$ is non negative and thus $\sum_{i_1=1}^{2} \sum_{i_2=1}^{2} ... \sum_{i_D=1}^{2} (-1)^{i_1+...+i_D} C\left(z_{1,i_1}, ..., z_{D,i_D}\right) \geq 0$ where $z_{j,i_j} = a_j$ if $i_j = 1$ and $z_{j,i_j} = b_j$ if $i_j = 2$.*

One of the main theorems on copula theory was derived in [Skl59].

Theorem 6.1.1 (Sklar's theorem)**.** *Let F be a D-dimensional CDF with marginal CDFs F_i for $1 \leq i \leq D$. Then there exists a copula C such that for any $1 \leq i \leq D$ and for any $x_i \in [-\infty, +\infty]$ we have:*

$$F(x_1, ..., x_D) = C\left(F_1(x_1), ..., F_d(x_D)\right). \tag{6.8}$$

Moreover, if all the marginal CDFs F_i are continuous, the copula C is unique.
Similarly, for any copula C, the function F defined by equation (6.8) is a cumulative distribution function with marginal CDFs F_i.

This theorem is of utmost importance for this work, since our objective is to extend GLD to multivariate distributions. Using Sklar's theorem, one can see that it is possible to build a multivariate CDF with marginals given by GLDs and entirely defined by the choice of a copula.

Definition 6.1.7. *Consider a copula C which is differentiable with respect to x_i for all $1 \leq i \leq D$, and consider differentiable marginal CDFs F_i. Then, the copula density c is defined for $(x_1, ..., x_D) \in \mathbb{R}^D$ as:*

$$c(x_1, ..., x_D) = \frac{\partial^D C}{\partial x_1 ... \partial x_D}(x_1, ..., x_D). \tag{6.9}$$

Sklar's theorem provides a link between the density f of the random variable with CDF being F, and the associated copula density c:

$$f(x_1, ..., x_D) = c(x_1, ..., x_D) \prod_{i=1}^{D} p_i(x_i) \tag{6.10}$$

where $p_i(x_i) = \frac{dF_i}{dx_i}(x_i)$.

Definition 6.1.8 (Gaussian copula)**.** *Let Σ be a correlation matrix of size D. The Gaussian copula with correlation matrix Σ is defined by:*

$$C_\Sigma(x_1, ..., x_D) = \Phi_\Sigma\left(Q(x_1), ..., Q(x_D)\right) \tag{6.11}$$

where Q is the quantile function of the standard normal distribution $\mathcal{N}(0,1)$ and Φ_Σ is the CDF of the multivariate normal distribution $\mathcal{N}(\mathbf{0}, \Sigma)$ with mean 0 and covariance matrix Σ. The density of the Gaussian copula is given by:

$$c(x_1, ..., x_D) = |\Sigma|^{-1/2} \exp\left(-\frac{1}{2} Q(\mathbf{x})^T (\Sigma^{-1} - \mathcal{I}_D) Q(\mathbf{x})\right) \tag{6.12}$$

where $Q(\mathbf{x}) = (Q(x_1), ..., Q(x_D))$ and $\mathcal{I}_D$ is the identity matrix of size D.

The Gaussian copula is introduced in this work to create a correlation structure between univariate quantile-parametrized distributions, as is done in [PLS23] for instance. With the help of scalar quantile-parametrized distributions and Gaussian copula, one can create a large class of multivariate distributions. In this chapter, we seek to use these distributions to approach the true posterior distribution of an inverse problem. The inverse problem resolution can then be reduced to a prediction task for a finite number of parameters, which greatly simplifies the resolution and avoids the need for computationally intensive MCMC sampling

To illustrate the use of Gaussian copulas, we plot in Figures 6.3 through 6.5 examples of bivariate GLDs built with GLD marginals and a Gaussian copula.

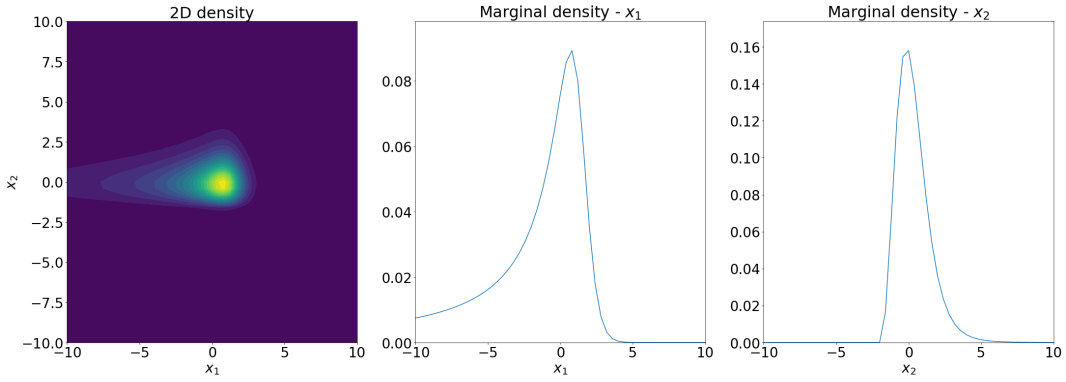

FIGURE 6.3: 2D density and marginal densities of a bivariate GLD with correlation $c = 0$ and marginals $\mathcal{G}(-4, 0.1, 0, 1)$ and $\mathcal{G}(0.5, -0.2, 0, 1)$.

6.2 Learning a posterior

6.2.1 Dataset creation

Consider an inverse problem for which we have a noise covariance given by $\mathbf{C}_{\text{obs}}$ and some observations $\mathbf{y} = (y^{(1)}, ..., y^{(N)}) \in \mathbb{R}^{N \times D}$ for $N \geq 1$. This inverse problem boils down to a single-observation case, with an effective covariance $\mathbf{C}_{\text{eff}} = \frac{1}{N}\mathbf{C}_{\text{obs}}$, and the mean observation $\overline{y} = \frac{1}{N}\sum_{k=1}^{N} y^{(k)}$ (see proposition 2.5.2).

To solve this inverse problem, one may sample the posterior distribution $p(x|\mathbf{y})$ with MCMC sampling to yield a Markov chain whose invariant distribution is the

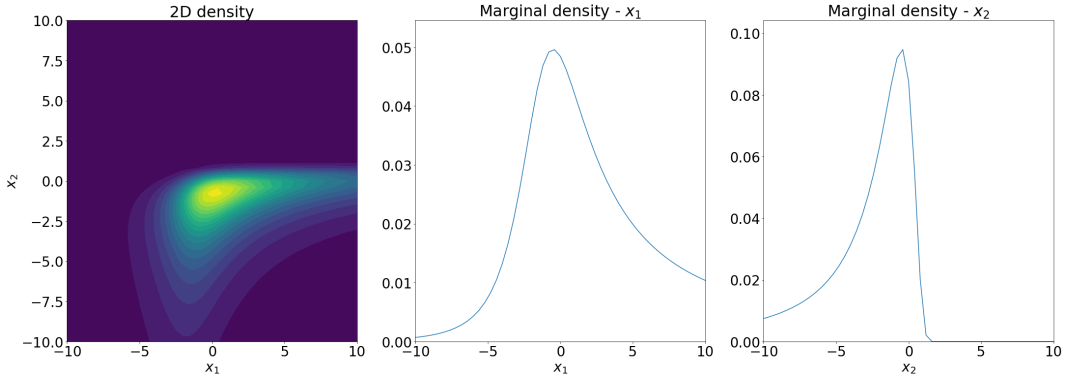

FIGURE 6.4: 2D density and marginal densities of a bivariate GLD with correlation $c = 0.9$ and marginals $\mathcal{G}(-1,-5,0,1)$ and $\mathcal{G}(-2,0.5,0,1)$.

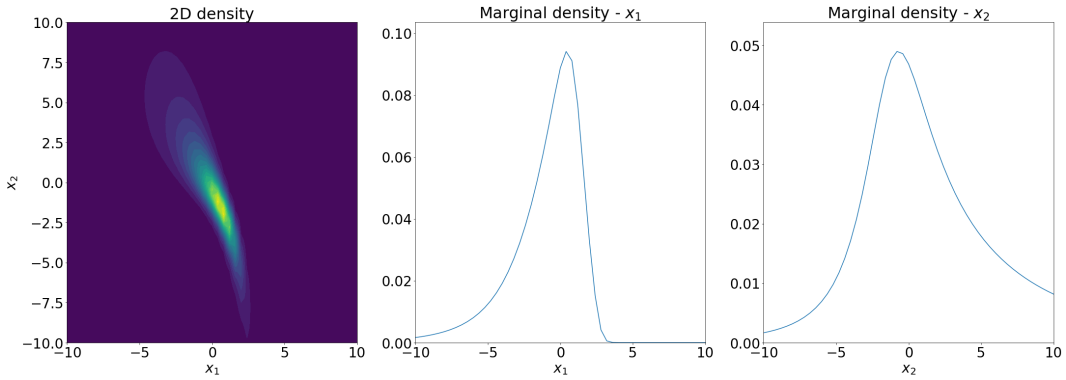

FIGURE 6.5: 2D density and marginal densities of a bivariate GLD with correlation $c = -0.99$ and marginals $\mathcal{G}(-4,-1,0,1)$ and $\mathcal{G}(-1,-5,0,1)$.

posterior. This approach is the gold standard and was used multiple times in the rest of this work.

Now consider a class of probability distribution $(p_\lambda)_{\lambda\in\mathbb{R}^m}$ for $m \geq 1$. One could try to approximate the posterior distribution with the right choice of λ, as is done in variational inference strategies. The natural metric when comparing two probability distributions is the KL divergence, and thus the optimal choice of parameters is given by:

$$\lambda^*(\overline{y}, \mathbf{C}_{\text{eff}}) \in \text{argmin}_{\lambda\in\mathbb{R}^m} \text{KL}\left(p(\,\cdot\,|\mathbf{y})\|p_\lambda\right). \tag{6.13}$$

This optimal $\lambda^*(\overline{y}, \mathbf{C}_{\text{eff}})$ is of course dependent on the inverse problem considered. To solve the inverse problem for any set of observations and noise covariance matrix, our approach aims at learning the function $\lambda(\overline{y}, \mathbf{C}_{\text{eff}})$ for any $\overline{y} \in \mathbb{R}^D$ and any $\mathbf{C}_{\text{eff}} \in S_D(\mathbb{R})^+$.

To learn this function, a dataset is needed for supervised learning methods. This dataset is built by creating some observation and covariance $(\overline{y}, \mathbf{C}_{\text{eff}})$ and then solving the optimization problem (6.13) for this particular value. Repeating this process yields a dataset that can then be used for predictive models.

6.2.2 KL minimization

How do we solve the optimization problem (6.13) in practice? Consider a given mean observation $\overline{y}$ and an effective noise covariance $\mathbf{C}_{\text{eff}}$. First of all, we need to obtain the true posterior distribution $p(x|\overline{y}, \mathbf{C}_{\text{eff}})$ with the help of MCMC sampling, which provides an ergodic Markov chain $(X_l)_{1\leq l\leq L}$ whose invariant distribution is $p(x|\overline{y}, \mathbf{C}_{\text{eff}})$.
From here, for any $\lambda \in \mathbb{R}^m$, $\mathrm{KL}\left(p(\cdot|\overline{y}, \mathbf{C}_{\text{eff}})\|p_\lambda\right)$ can be evaluated. Based on the ergodic theorem 1.6.2, the KL divergence can be approximated with the estimator:

$$\mathrm{KL}\left(p(\cdot|\overline{y}, \mathbf{C}_{\text{eff}})\|p_\lambda\right) \simeq \frac{1}{L}\sum_{l=1}^{L} \log\left(\frac{p(X_l|\overline{y}, \mathbf{C}_{\text{eff}})}{p_\lambda(X_l)}\right). \tag{6.14}$$

When solving the optimization problem, numerous performance factors must be taken into account. The approximation (6.14) requires a large number of MCMC samples, or more specifically a large number of decorrelated samples since this estimator follows the assumption of theorem 1.6.6. As a criteria, if τ is the estimated decorrelation time, our objective is to have at least $L \geq 100\tau$ to guarantee a good KL approximation. However, since the computation of (6.14) scales with a larger number of samples, the MCMC chain is thinned out by a factor τ to only keep decorrelated samples.
Finally, the optimization problem can be quite difficult since the KL can be particularly sensitive to a small change in λ, and the target function is not convex. Thus, a global optimization algorithm is required. We focus on dual annealing, which is available in the *scipy* package.

6.3 Applications

6.3.1 A scalar inverse problem

We start with a simple scalar inverse problem. We consider the analytical direct model $f\colon \mathbb{R} \longrightarrow \mathbb{R}$ defined by $f(x) = 0.04x^3$ for $x \in \mathbb{R}$. We consider an inverse problem in which we have $N = 4$ noisy observations $\mathbf{y} = (y^{(k)})_{1\leq k\leq N}$ given by:

$$y^{(k)} = f(x_{\text{true}}) + \varepsilon^{(k)}. \tag{6.15}$$

Here we chose $x_{\text{true}} = -2.5$ and the $\varepsilon^{(k)}$ are iid standard Gaussian random variables with distribution $\mathcal{N}(0, 1)$.

To mimic our general methodology, we are not using the direct model in its analytical form, but rather use an emulator built with a Gaussian process model $f_s \sim \mathcal{GP}(0, k(x, x'))$ with k a Matérn 5/2 covariance function. The GP surrogate is conditioned on $n = 50$ training instances $(\mathbf{X}, \mathbf{Y})$ where the inputs $\mathbf{X}$ are sampled from a uniform distribution on $[-5, 5]$ and where $Y_i = f(X_i) + \eta_i$ with $\eta_i \sim \mathcal{N}\left(0, 0.1^2\right)$ for $1 \leq i \leq n$.

The next step is to generate a dataset of size $M = 120$ of inverse problem resolutions, for varying observations and observational noise. Observations of the inverse problem are generated randomly by sampling uniformly in the image space $\mathcal{Y} = f(\mathcal{X}) = [-5, 5]$. Naturally, in cases where the direct model is inaccessible, the exact image space is unknown. We discuss this obstacle in the next section. The new noisy observations are denoted by $\mathbf{z} = (z^{(j)})_{1\leq j\leq M}$ and are sampled uniformly in

$\mathcal{Y}$. Each observation $z^{(j)}$ is associated with an observational standard deviation $\sigma^{(j)}$ which is sampled uniformly in $[0.1, 2]$.
Then, for each pair $(z^{(j)}, \sigma^{(j)})$ we solve an inverse problem with a Bayesian resolution with MCMC to obtain the corresponding posterior distribution $p(\,\cdot\,|z^{(j)}, \sigma^{(j)})$. Finally, the optimization problem (6.13) is solved (with dual annealing) to provide the parameters $\lambda^{(j)}$ of the GLD approximating the true posterior distribution, for the observation $z^{(j)}$ and standard deviation $\sigma^{(j)}$. This gives us a dataset $((\mathbf{z}, \boldsymbol{\sigma}), \boldsymbol{\lambda})$ on which a predictor can be trained. We chose to use a simple fully-connected neural network with L2 loss function in this case.

The GLDs have four parameters in this scalar case which makes both the optimization problems and the MCMC relatively easy. The mean KL over the test set is $\overline{\mathrm{KL}} \simeq 1.9$. In Figure 6.6, we compare the predicted GLD distributions with the posterior distributions obtained via MCMC for some test data taken from the dataset $(\mathbf{z}, \boldsymbol{\sigma})$. Overall,

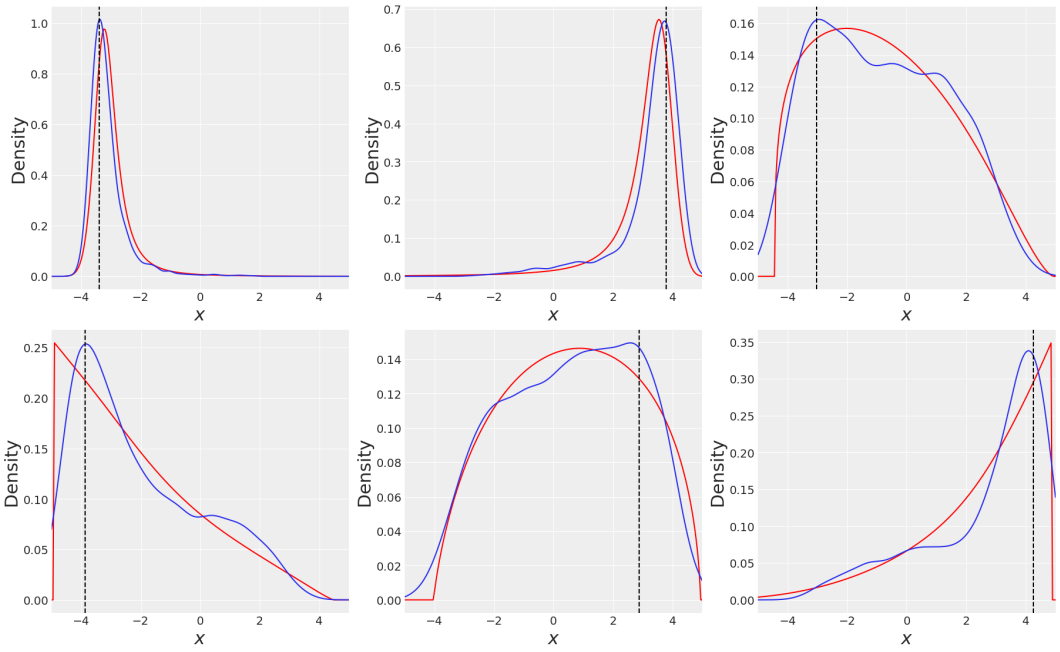

FIGURE 6.6: Comparison between the GLD densities (red) and posterior densities obtained via MCMC with KDE (blue) for 6 test instances. The MAP of the MCMC posterior is represented as a vertical dotted line.

the GLD densities are close to the posterior densities obtained with MCNP. The GLD densities tend to be smoother and are not able to model non-smooth target densities. They are also unable to model multi-modal distributions. However, we know from the previous chapters that the posterior distributions arising in the context of neutron noise analysis are smooth and unimodal. The reader may also notice that the estimation of the MAP with GLDs may lead to some misestimations. This is especially the case for non-informative densities such as the bottom center plot in Figure 6.6.

When applied to this toy inverse problem, the predictor provides a GLD predictive distribution which is displayed in Figure 6.7 along with the posterior obtained by MCMC. The predictive GLD appears close to the posterior density obtained by MCMC, though the GLD distribution is wider and the MAP is displaced slightly to the left. This first application to a simple scalar inverse problem serves as an introductory example of the methodology. In the next section, we seek to apply this

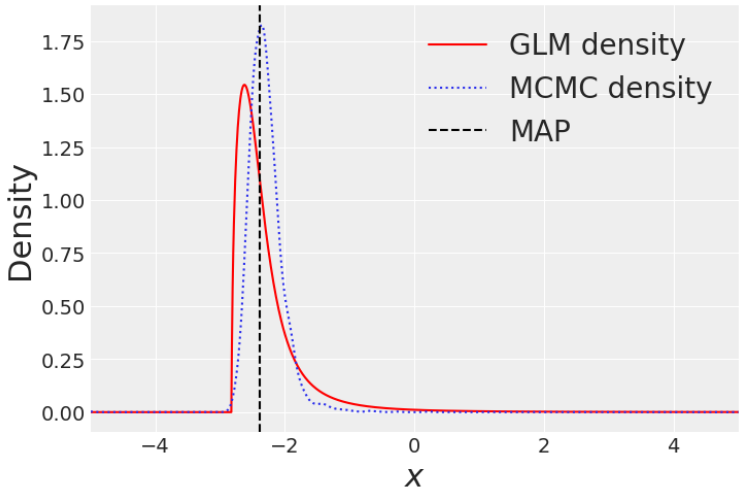

FIGURE 6.7: Comparison between the predictive GLD density (red) and the true posterior density (red) obtained for the observations given in (6.15).

procedure to neutron correlation measurements, for the SILENE and BERP test cases.

6.3.2 A multivariate inverse problem

We now shift our attention toward the resolution of a multivariate inverse problem. The general principle is similar and we use the same notations: we need to generate a set of M observations $(\mathbf{z}, \boldsymbol{\sigma})$ from which posterior distributions $p(\,\cdot\,|z^{(j)}, \sigma^{(j)})$ are obtained with MCMC for $1 \leq j \leq M$. These distributions are then used to find the optimal parameters $\boldsymbol{\lambda}^{(j)}$ of the GLD representations leading to a new dataset $((\mathbf{z}, \boldsymbol{\sigma}), \boldsymbol{\lambda})$ which in turn is utilized to train a predictor.

Let us proceed step by step. To create the dataset $(\mathbf{z}, \boldsymbol{\sigma})$ we need to provide credible observations and covariances. We will make use of the data available in our two datasets (see Chapter 2). To provide reasonable observations we fit a Gaussian copula to the output data as was done in Chapter 5 for the input dataset. Once the copula is fitted, it is possible to generate synthetic output data. An example of the correlations and marginals of a synthetic dataset with $M = 1000$ data instances is shown in Figure 6.8. This synthetic dataset resembles the original dataset which was displayed in Figure 2.6. For precaution, we need to filter out values that are sampled outside the bounds of the original dataset to avoid overly confident synthetic data. The next phase is to create realistic observational covariances. This step is more complex and will be based on empirical evidence. From the observations gathered throughout this thesis, one can derive some knowledge of the correlations and standard deviations for the outputs. Generally, the count rate has low variance. For the BERP test case, the relative standard deviations for R, Y_∞, and X_∞ are around 0.5%, 3%, and 9%. In the inverse problem likelihood, the variances are divided by the number of observations. Thus, the range of covariances in the dataset needs to be wide enough to encompass scenarios with both few and many observations. The correlations between outputs also vary depending on the test cases. Similarly, the correlation coefficients are sampled with sufficiently large bounds to include a variety of cases, while keeping the matrix positive-definite.

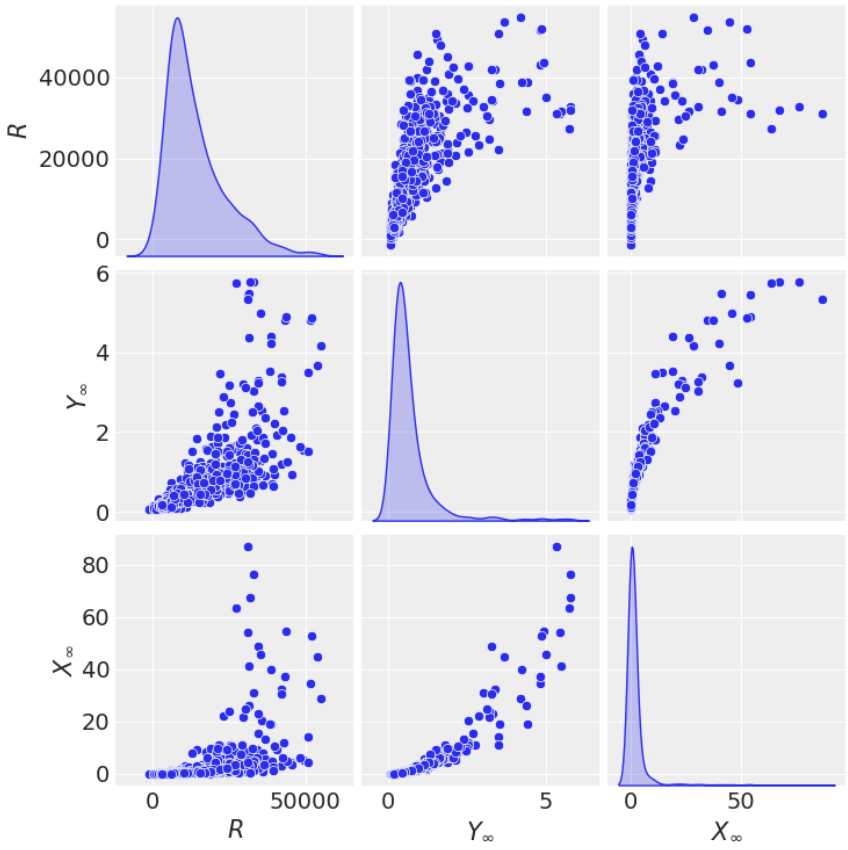

FIGURE 6.8: Marginal densities and correlation plots for the synthetic data.

We can now build the dataset $((\mathbf{z}, \boldsymbol{\sigma}), \boldsymbol{\lambda})$. However, the optimization problem and the subsequent predictive tasks are much more difficult than in the scalar case presented in the previous section. We cannot hope to use the surrogate models developed in Chapters 2 and 3. The dimension of the input space would demand too many parameters in the multivariate GLDs (49 parameters to be exact). Thus, we limit our study to a three-dimensional input space, which includes only the parameters (k_p, ε_F, S). Even in such a low dimension, the parametrization of the posterior distributions with GLDs and Gaussian copulas necessitates $m = 15$ parameters.

The predictive task consists in providing the 15 parameters λ given $z \in \mathbb{R}^3$ and $\sigma \in S_3(\mathbb{R})^+$. This task requires sufficient training data. Thus, the generation of the Markov chains for the posteriors $p(\,\cdot\,|(z^{(j)}, \sigma^{(j)}))$ and the subsequent optimization problems to obtain the $\lambda^{(j)}$ are parallelized on 40 cores.
In the optimization process, we reduce the number of parameters to optimize by choosing the position parameters as the MAP estimates of the posterior distributions. This choice is supported by the position parameter being precisely the mode of a GLD. Moreover, the optimization bounds are defined to pre-select copulas with large correlations as is the case for the expected shape of the posterior distributions. Additionally, the shape parameters are chosen to be negative. Indeed, the support of the posterior distribution is infinite though the density rapidly drops toward zero

Two examples of the predicted multivariate GLDs estimated on test instances are shown in Figures 6.9 and 6.10 along with the true posterior sampled by MCMC.

The methodology described in this section is applied separately to the BERP and

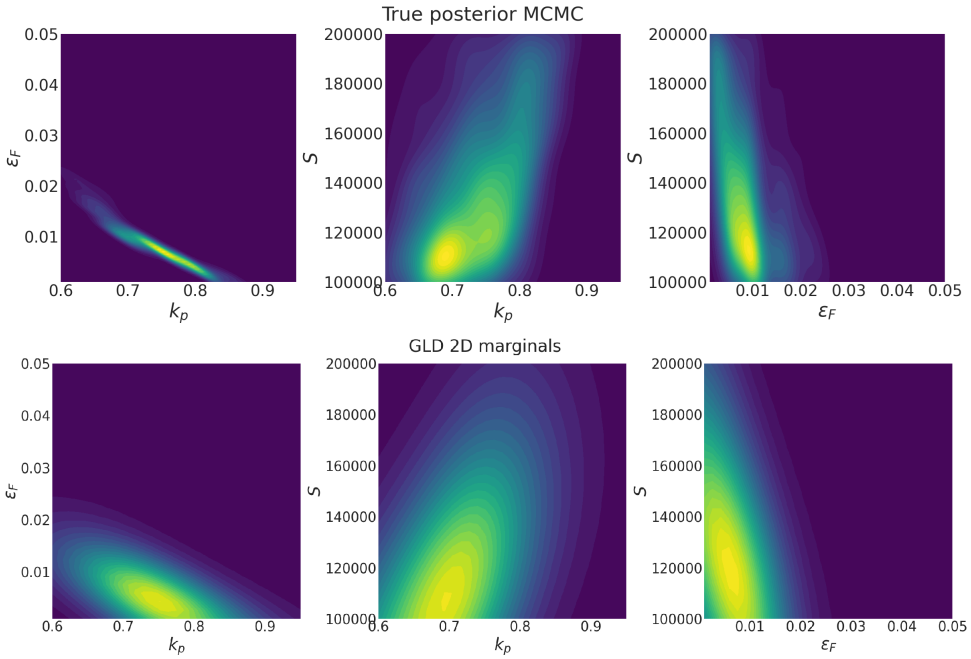

FIGURE 6.9: Comparison of the 2D marginals for the MCMC posterior distribution (top) and the predictive multivariate GLD (bottom) on a test instance.

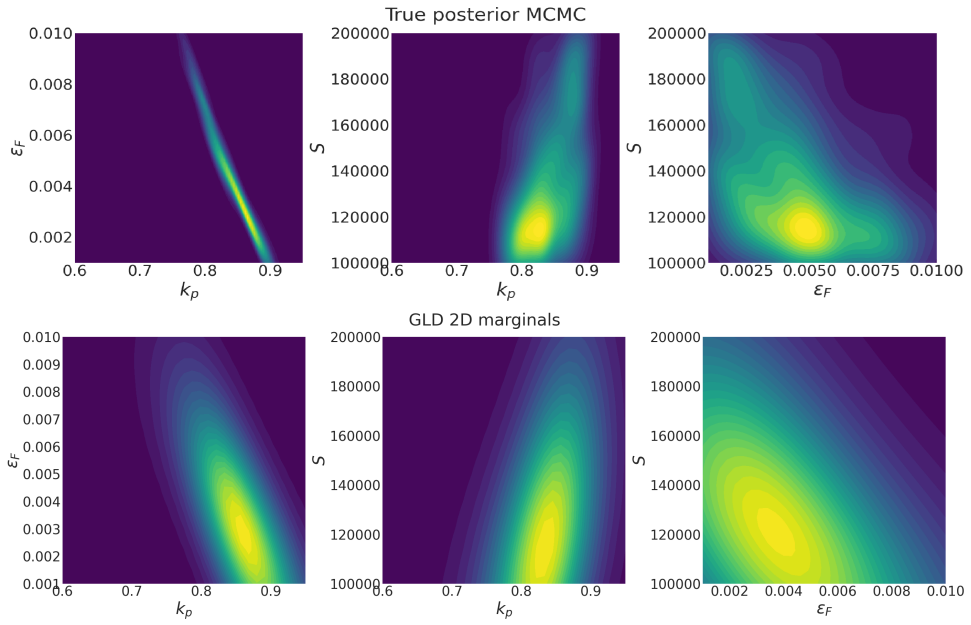

FIGURE 6.10: Another example of true posterior (top) compared to the predicted GLD (bottom).

SILENE test cases. Two distinct datasets $((\mathbf{z}, \boldsymbol{\sigma}), \boldsymbol{\lambda})$ are built. We detail these examples in the next two sections.

6.3.2.1 An application to SILENE

We apply our methodology to the various configurations of the SILENE reactor. The GLD predictor is trained over 280 instances. The mean KL between predictive GLDs

and MCMC posterior distributions over the test set is $\overline{\mathrm{KL}} = 7.2$.
We then use the observational data from the SILENE experiments to predict a multivariate GLD approaching the true posterior distribution. The observational data are given by the experimental measurements summarized in Table E.4 in Appendix E. The marginals of the predicted multivariate GLDs for respectively $h = 15$ cm, $h = 20$ cm, and $h = 25$ cm of fissile solution are exhibited in Figures 6.11, 6.12 and 6.13. The MCMC posterior distributions have been presented in Figures 2.22 through 2.24.

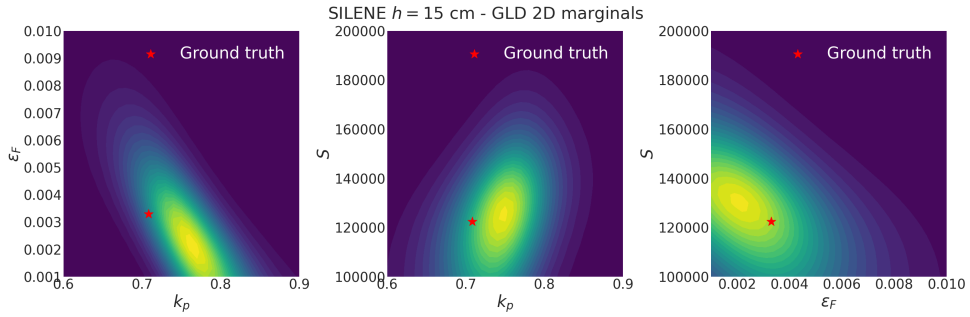

FIGURE 6.11: Predictive GLD for the SILENE test case with $h = 15$ cm of fissile solution.

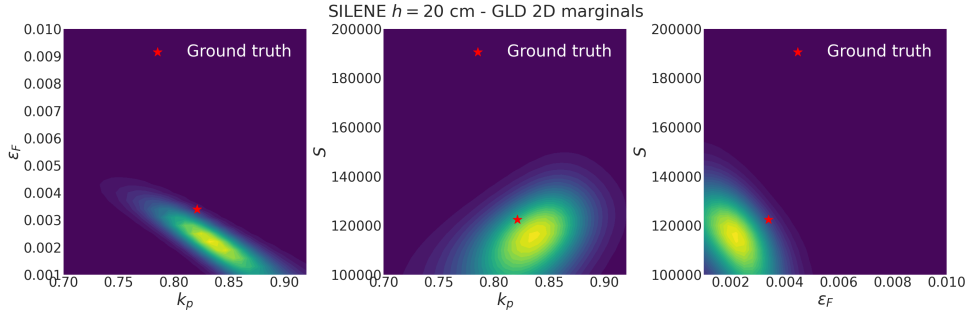

FIGURE 6.12: Predictive GLD for the SILENE test case with $h = 20$ cm of fissile solution.

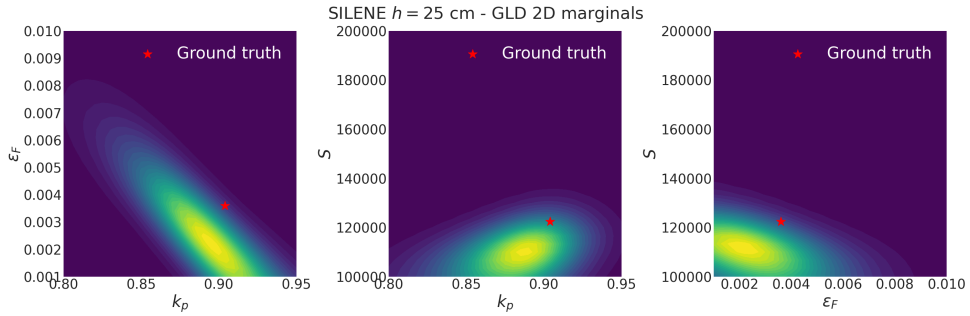

FIGURE 6.13: Predictive GLD for the SILENE test case with $h = 25$ cm of fissile solution.

Though the multivariate GLD distributions differ from the true posterior distributions obtained by MCMC, they still provide acceptable uncertainty quantification.

One must also note that the multivariate GLDs extend outside the physical space. For example, we have non-negligible densities in regions of $\varepsilon_F < 0$ which is physically inaccessible. The predictive GLDs can thus be truncated to respect physical constraints.

To further investigate the capabilities of the multivariate GLDs, we provide similar distributions by multiplying only the effective covariance $\mathbf{C}_{\text{eff}}$ by a factor $s_{\text{eff}} \in \{1/4, 1, 4\}$ which amounts to multiplying or dividing by two the number of observations available. The predictive distributions obtained are displayed in Figure 6.14. We observe some changes in the spread of the GLDs, though the variations are not

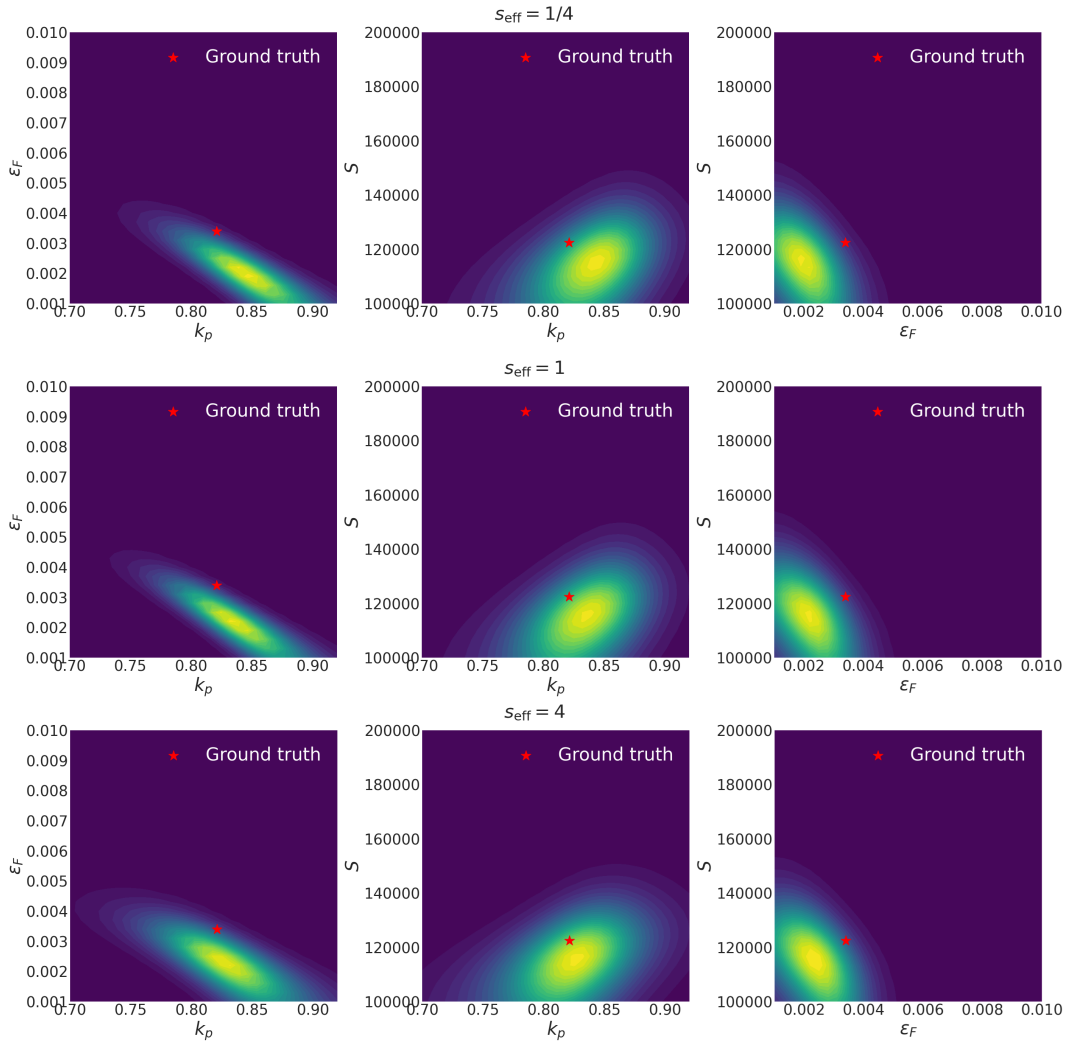

FIGURE 6.14: 2D marginal densities of the predictive GLDs obtained with $s_{\text{eff}} \in \{1/4, 1, 4\}$ for the SILENE $h = 20$ cm configuration.

as noticeable as with MCMC sampling. The main explanation is the larger model uncertainty in the underlying GP used to build the training set since only the three inputs (k_p, ε_F, S) were considered. Thus, reducing the observational variance does not necessarily lead to a drastic reduction of the global uncertainty.
Overall, the multivariate GLDs offer an immediate uncertainty quantification, though not as reliable as the Bayesian resolution of the inverse problem. Let us look at the BERP case now.

6.3.2.2 An application to the BERP case

Newt, we want to compare the posterior distributions obtained in Chapter 2 with GP surrogates, to the predictive multivariate GLD provided by our predictor for the BERP test case. The methodology is the same as for the SILENE test case. The mean KL obtained on the test set is $\overline{\mathrm{KL}} = 8.4$. The training set contains 287 instances and the test set 40 instances.The 2D marginals of the predictive multivariate GLD are displayed in Figure 6.15.

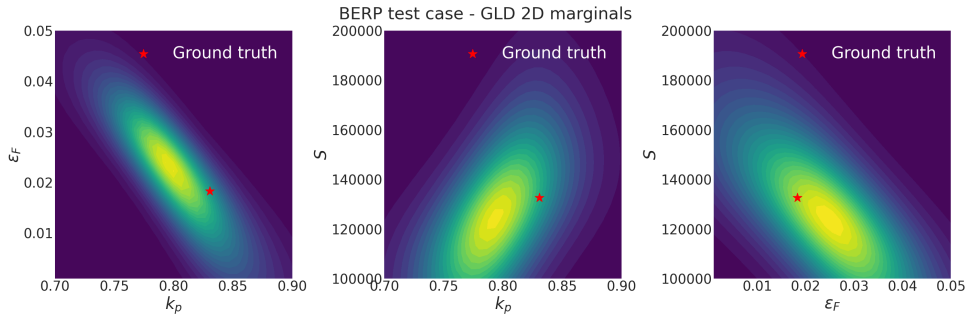

FIGURE 6.15: 2D marginals of the predictive multivariate GLD obtained with the BERP observational data (see E.4).

The multivariate GLD appears more conservative than the true posterior obtained with MCMC sampling (see for example Figure 2.20). Although the predicted uncertainties are not as reliable as the standard Bayesian resolution of the inverse, this method offers an interesting addition to our methodological toolkit, especially for very constrained predictions.

6.3.3 GLDs as prior distributions

The previous GLDs can be used to provide initial estimates of posterior distributions. However, as we have seen in the previous section, the distributions display some inaccuracy. These predictive GLDs are usable mostly for initial estimates under stringent time constraints. However, they may be suited to serve as prior distributions in the Bayesian inverse problem. In this section, we explore this possibility.

The main difficulty is that predictive multivariate GLDs require some knowledge of the observations $\mathbf{y}$. These observations are already used in the likelihood of the inverse problem during the Bayesian resolution. If we wish to use multivariate GLDs as prior, we could split a fraction of our observations and feed it to the GLD predictor, while the rest is kept for the Bayesian resolution of the inverse problem. The main question is whether or not this is beneficial in reducing the uncertainties of the nuclear parameters.

Let us highlight this methodology on the SILENE test case, with $h = 20$ cm. Starting from the initial set of observations $\mathbf{y}$, half of the observations are used to predict a multivariate GLD which serves as a prior distribution. The other half is used as inverse problem observations. We sample the posterior distribution with AM with $L = 5 \times 10^5$ samples. The prior distribution for (k_p, ε_F, S) is given by the multivariate GLD and is uniform on the other parameters. The marginal densities of the distribution are plotted in Figure 6.16 along with the distribution obtained in Chapter 2. The two distributions exhibit strong differences. Using the GLD prior largely reduces the spread of the posterior distribution. The question that arises with

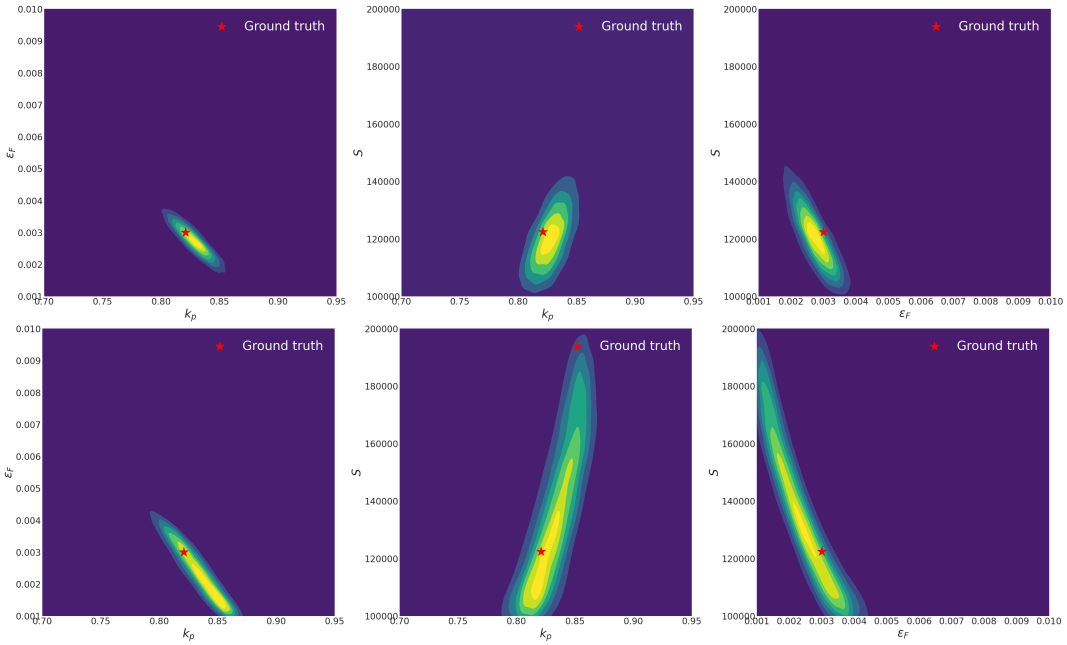

FIGURE 6.16: Comparison of marginal densities of the posterior distributions obtained with the multivariate GLD prior (top) and with a uniform prior (bottom).

this method is whether or not the uncertainties are reliable with this approach. The large disparities between the distributions advise caution. The GLD prior may be too restrictive to keep reliable uncertainty quantification. For this reason, one may temper the GLD prior with an exponent $\beta \in [0, 1]$ (as was done in Chapter 4 for the TIP-SUR strategy). An example of the posterior distribution obtained with a tempering parameter $\beta = 0.3$ is displayed in Figure 6.17. The tempered prior limits

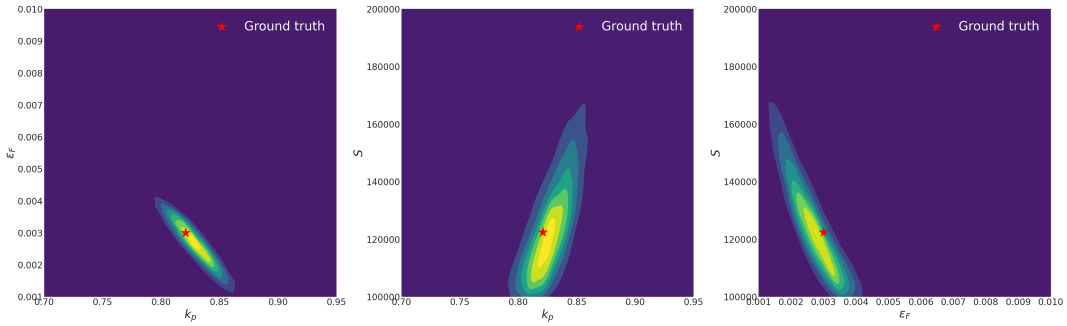

FIGURE 6.17: Marginal densities of the posterior distribution obtained with a GLD prior and a tempering parameter $\beta = 0.3$.

the contribution of the multivariate GLD and thus interpolates between the GLD prior and the uniform prior. The difficulty resides in ensuring the reliability of the posterior distribution.

Therefore, using multivariate GLD priors in the neutron correlations inverse problem may be beneficial, but further analyses are required to ensure the prior is not too restrictive and hinders the reliability of the uncertainty quantification procedure. Specific work on the matter is needed, especially in regimes with few observational data where the choice of the prior bears more importance.

6.4 Conclusion

In this chapter, we studied multivariate GLDs built with marginal GLDs with a parametric Gaussian copula. They can be used to provide initial estimates for the posterior distributions in a low-dimensional inverse problem. For our application, this may be useful to deliver instantly a point estimate of the nuclear parameters with some notion of uncertainty quantification. Evidently, this approach is not as reliable as the standard resolution of the inverse problem with MCMC but its strengths are dual. Firstly, the multivariate GLD allows for an immediate resolution of the inverse problem, where MCMC sampling requires at least a few minutes. This can be especially useful for field measurements to provide an immediate determination of the nuclear parameters along with their uncertainties. Furthermore, the predicted multivariate GLDs can then serve as prior distributions in the Bayesian inverse problems, effectively reducing the uncertainties, especially in a setting with few observations. Therefore, this methodology is aimed mainly at field measurements in time-constrained operations, when the estimation of nuclear parameters is required immediately.

Similar methodologies could have been used to learn the posterior distribution and circumvent the MCMC sampling involved in the Bayesian resolution of the inverse problem. Most notably, methods involving conditional generative models may be of interest for this particular application. we mention for example Conditional Variational Autoencoders (CVAE) [KW13; TD20; Goh+19], conditional diffusion models [Son+20; Son+23] or conditional Generative Adversarial Networks (CGAN) [Ray+22; PRO22]. However, we expect conditional generative models to be slower than the multivariate GLD approach presented in this chapter, which takes only a few seconds. They may serve as an intermediate step between MCMC resolutions and multivariate GLD predictions on the speed-accuracy trade-off.

Chapter 7

Conclusion

7.1 Main contributions

The objective of this thesis was to develop a methodology for the determination and uncertainty quantification of nuclear parameters with passive neutron noise measurements. This task was restricted by the necessity for a robust and conservative UQ methodology with a constrained computational budget. In our framework, the direct model, which is the forward mapping of material characteristics to neutron correlation measurements, is modeled by a complex computer code such as a Monte Carlo neutronic code. This project focuses on the inverse problem. Provided some noisy observations of neutron correlations, we wish to determine the corresponding material characteristics and their uncertainties. A Bayesian resolution of the inverse problem appears as the most natural path toward this objective. In this paradigm, one seeks to estimate the posterior distribution of the material characteristics x given the observations $\mathbf{y}$. However, the Bayesian resolution of an inverse problem often involves MCMC sampling which requires a large number of calls to the direct model. Because the direct model is a complex computer code, this approach is not computationally tractable as such.

In that context, we investigated uncertainty quantification methods based on surrogate models. The purpose was to build emulators of the direct models able to provide fast and reliable predictions, as well as predictive uncertainties. Our initial attempt was centered around Gaussian process models. Several GP models were proposed and all the models were trained with two simulated datasets of neutron correlation measurements obtained with the neutronic Monte Carlo code MCNP. In particular, we highlighted that GP models built with a Coregionalization structure reliably predict uncertainties for the correlated quantities at stake in our problem. Furthermore, these models can benefit from multi-fidelity and heteroscedastic noise modeling to further improve the reliability of the uncertainty quantification. The low computational cost for inference with the newly developed GP emulators makes them usable in an MCMC sampling strategy. To solve the inverse problem, we account for the epistemic and aleatoric uncertainties to provide a conservative quantification of the uncertainties. The MCMC sampling can be performed in less than an hour on a laptop and provides a posterior distribution that comprises all possible quantities of interest, such as the variance, the maximum-a-posteriori, or higher-order moments.

To diversify our models, we explored a comparable methodology utilizing Bayesian Neural Network surrogates instead of GP-based surrogates. Such models are more complex than GPs but their predictive distribution is less restrictive. The output distribution of a BNN model is intractable. It must be approximated which results in a Gaussian mixture output distribution. We have seen that this distribution can

be implemented directly in the likelihood during the inverse problem resolution. We also explored various architectures including heteroscedastic and multi-fidelity models. Overall, BNN models are an interesting addition to our toolkit but are prohibitive for cases with a limited computational budget. They are better suited for post-experiment analysis and from our experience, they may provide better modeling of the heteroscedastic structure of the data.

We then developed two sequential design strategies specifically tailored for Bayesian inverse problems. The goal of these methods is to select new design points to improve the surrogate models, given the observations of an inverse problem. They can be understood as a fine-tuning of the models for a specific inverse problem. While most sequential design strategies target design points where the uncertainty of the model is large, our methods are built upon a trade-off between high model uncertainty and high posterior density. The first proposed approach, known as Constraint Set Query (CSQ) selects design points with the highest predictive covariance determinant in a subset of the design space close to the maximum-a-posteriori. We then offer a second method, denoted by IP-SUR (Inverse Problem Stepwise Uncertainty Reduction), which is based on the SUR paradigm and uses a metric of uncertainty derived from I-optimal designs. We highlighted in numerous applications that they perform better than more standard strategies. Their applicability depends on the cost of a call to the direct model. While CSQ is simpler and does not require an intermediate MCMC sampling, it is not supported by theoretical results and relies on a hyperparameter whose tuning is purely empirical. On the other hand, the IP-SUR strategy (and its variant TIP-SUR), comes with a theoretical guarantee of convergence for the metric of interest and does not rely on hyperparameters, but requires intermediate MCMC sampling.

To further reduce the uncertainties in the nuclear parameters, we questioned the possibility of including gamma correlations in the inverse problem. This extension of our work was dictated by recent developments in the joint neutron and gamma detection systems using scintillation detectors. The inclusion of gamma correlations can be done in two ways. Either we solve sequentially two inverse problems using the first posterior distribution as a prior in the second problem. Or, we group neutron and gamma correlations into a single joint model and solve only one inverse problem. While the latter is supposedly better, it is hindered by the higher dimensionality of the problem. We tried to leverage the sequential design strategies developed to improve the joint model and while it did bring significant improvements to the model, the joint approach is still not as reliable as the sequential approach.

Finally, to accelerate the resolution of the inverse problem, we investigated a more direct approach, modeling the posterior distributions using parametric distributions. Generalized lambda distributions with Gaussian copulas were used to parametrize the posterior distributions. The goal was to create a predictor providing the corresponding parameters, given some observations of the direct model (and the observational covariance). Though this resolution is not as reliable as the MCMC-based resolution, it still provides in a few seconds an approximate distribution, which can be used to provide a first estimate of the uncertainties. Furthermore, this distribution could also serve as a prior in the Bayesian resolution to improve the resulting posterior in low-data regimes, though further studies are required in that area.

7.2 Perspectives and future work

This thesis is far from covering all the scientific inquiries raised by the subject and there are still many opportunities for further research. We highlight some of the possible developments brought into perspective by our work.

One aspect that has not been thoroughly studied in this thesis is the choice of additional input parameters. We chose to add three parameters (ε_A, Φ, J) to quantify the parasitic absorptions, neutron energy spectrum, and the reflection. However, the optimal parametrization of the problem remains unknown, and more relevant parameters may be found.

We also extensively discussed the use of IP-SUR designs to improve a surrogate model in the context of a Bayesian inverse problem. Furthermore, the TIP-SUR design strategy was introduced as a natural extension of the IP-SUR design. However, there is still work to be done on the applicability of these methods and especially to fully understand in which case TIP-SUR might outperform IP-SUR. The temperature parameter β also appears as a crucial input to the TIP-SUR strategy. A robust selection strategy for β may be interesting to overcome the troublesome user-dependent selection.

Incorporating gamma correlations into the inverse problem has proven beneficial in this thesis. However, this aspect of our work was focused solely on gamma measurements obtained through Monte Carlo simulations. These simulations were quite simplistic and were not fully representative of real experimental conditions. An application to real-world gamma measurements is required to guarantee the robustness of this method. Additionally, further investigation into the impact of gamma multiplicity data is necessary to confirm the reliability of this methodology and address all sources of uncertainty. Another potential improvement lies in the study of cross-correlations between neutrons and gammas, that is, the simultaneous detections of neutrons and gammas. This aspect was purposefully left aside in this thesis but could be the subject of future developments. Implementing this enhancement would require a detection system capable of recording and discriminating between neutrons and gammas.

Finally, further work may focus on conditional generative models to directly solve the inverse problem. These models have the potential to bypass the computational expense of MCMC sampling, offering near-instantaneous responses for on-field measurements. Several conditional generative models have already been discussed in the literature such as conditional GANs, diffusion models, or VAE. Investigating the reliability of these methods in accurately predicting nuclear parameters and their uncertainties presents an intriguing direction for future studies.

7.3 Contributions

Paper publications

- Paul Lartaud, Philippe Humbert, and Josselin Garnier. “Multi-output Gaussian processes for inverse uncertainty quantification in neutron noise analysis”. In: *Nuclear Science and Engineering* 197.8 (2023), pp. 1928–1951
- Paul Lartaud, Philippe Humbert, and Josselin Garnier. “Sequential design for surrogate modeling in Bayesian inverse problems”. In: *arXiv preprint*

arXiv:2402.16520 (2024)

Proceedings

- Paul Lartaud, Philippe Humbert, and Josselin Garnier. "Uncertainty quantification in neutron noise analysis using Monte Carlo Markov chain methods: an application to nuclear waste drum assay". In: *Proceedings of the International Conference on Physics of Reactors*. 2022, pp. 2674–2683
- Paul Lartaud, Philippe Humbert, and Josselin Garnier. "Bayesian Inverse Problem and Uncertainty Quantification in the Joint Analysis of Neutron and Gamma Corrrelations". In: *Proceedings of the International Conference on Physics of Reactors*. 2024, pp. 541–550

Conference presentations

- PHYSOR 2022, Pittsburgh (United States), 2022: Uncertainty Quantification in Neutron Noise Analysis using Monte Carlo Markov Chain Methods: an Application to Nuclear Waste Drum Assay (presentation).
- Mascot-Num 2022, Clermont-Ferrand (France), 2022: Supervised learning and Monte Carlo Markov Chain methods for inverse problem resolution in random neutronics (poster).
- Mascot-Num 2023, Le Croisic (France), 2023: Surrogate modeling and uncertainty quantification for inverse problems in stochastic neutronics (poster).
- ISNET-9 meeting, Saint-Louis (United States), 2023: Multi-output Gaussian processes for inverse uncertainty quantification in random neutronics (presentation).
- UNCECOMP 2023, Athens (Greece), 2023: Multi-output Gaussian processes for inverse uncertainty quantification in random neutronics (presentation).
- SIAM-UQ 2024, Trieste (Italy), 2024: Sequential design for Bayesian inverse problems (presentation).
- Mascot-Num 2024, Hyères (France), 2024: I-optimal sequential design for Bayesian inverse problems with Gaussian process surrogate models (presentation).
- PHYSOR 2024, San Francisco (United States), 2024: Inverse UQ in the joint analysis of neutron and gamma correlations for fissile matter identification (presentation).

Bibliography

[Adl10] Robert J Adler. *The geometry of random fields*. SIAM, 2010.

[AH+83] Søren Asmussen, Heinrich Hering, et al. *Branching processes*. Vol. 3. Springer, 1983.

[AL11] Mauricio A Alvarez and Neil D Lawrence. "Computationally efficient convolved multiple output Gaussian processes". In: *The Journal of Machine Learning Research* 12 (2011), pp. 1459–1500.

[AL+22] Romain Ait Abdelmalek-Lomenech et al. "Bayesian sequential design of computer experiments for quantile set inversion". In: *arXiv preprint arXiv:2211.01008* (2022).

[AM06] Christophe Andrieu and Éric Moulines. "On the ergodicity properties of some adaptive MCMC algorithms". In: *The Annals of Applied Probability* 16.3 (2006), pp. 1462–1505.

[AMI17] Samira Abbaspour, Babak Mahmoudian, and Jalil Islamian. "Cadmium telluride semiconductor detector for improved spatial and energy resolution radioisotopic imaging". In: *World journal of nuclear medicine* 16.02 (2017), pp. 101–107.

[AN04] Krishna B Athreya and Peter E Ney. *Branching processes*. Courier Corporation, 2004.

[And+12] Johan Anderson et al. "Two-point theory for the differential self-interrogation Feynman-alpha method". In: *The European Physical Journal Plus* 127 (2012), pp. 1–9.

[And12] William J Anderson. *Continuous-time Markov chains: An applications-oriented approach*. Springer Science & Business Media, 2012.

[And+75] Yasumasa Ando et al. "Void detection in BWR by noise analysis". In: *Journal of Nuclear Science and Technology* 12.9 (1975), pp. 597–599.

[ANS14] Ignacio Alvarez, Jarad Niemi, and Matt Simpson. "Bayesian inference for a covariance matrix". In: *arXiv preprint arXiv:1408.4050* (2014).

[ARH14] Nicolas Authier, Benoît Richard, and Philippe Humbert. "Initiation of persistent fission chains in the fast burst reactor Caliban". In: *Nuclear Science and Engineering* 177.2 (2014), pp. 169–183.

[ARRE08] Erdem Acar, Masoud Rais-Rohani, and Christopher Eamon. "Reliability estimation using dimension reduction and extended generalized lambda distribution". In: *49th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, 16th AIAA/ASME/AHS Adaptive Structures Conference, 10th AIAA Non-Deterministic Approaches Conference, 9th AIAA Gossamer Spacecraft Forum, 4th AIAA Multidisciplinary Design Optimization Specialists Conference*. 2008, p. 1893.

[AZ10] Alessandro Baldi Antognini and Maroussa Zagoraiou. "Exact optimal designs for computer experiments via Kriging metamodelling". In: *Journal of Statistical Planning and Inference* 140.9 (2010), pp. 2607–2617.

[BA60] HW Broek and CE Anderson. "The Stilbene Scintillation Crystal as a Spectrometer for Continuous Fast-Neutron Spectra". In: *Review of Scientific Instruments* 31.10 (1960), pp. 1063–1069.

[Bab67] Dšsan Babala. "Interval distributions in neutron counting statistics". In: *Nuclear Science and Engineering* 28.2 (1967), pp. 243–246.

[Bal12] Pierre Baldi. "Autoencoders, unsupervised learning, and deep architectures". In: *Proceedings of ICML workshop on unsupervised and transfer learning*. JMLR Workshop and Conference Proceedings. 2012, pp. 37–49.

[Bar93] Francis Y Barbry. *A review of the SILENE criticality excursions experiments*. Tech. rep. 1993.

[BAS55] R Batchelor, R Aves, and THR Skyrme. "Helium-3 Filled Proportional Counter for Neutron Spectroscopy". In: *Review of Scientific Instruments* 26.11 (1955), pp. 1037–1047.

[Bat15] Cameron Russell Bates. *Development of metallic magnetic calorimeters for nuclear safeguards applications*. University of California, Berkeley, 2015.

[Bat+96] RA Bates et al. "Experimental design and observation for large systems". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 58.1 (1996), pp. 77–94.

[BB92] James O Berger and José M Bernardo. "On the development of the reference prior method". In: *Bayesian statistics* 4.4 (1992), pp. 35–60.

[BBC99] P Baeten, M Bruggeman, and R Carchon. "A generalized multiplication correlation model for the description of fast and thermal neutron multiplication in neutron coincidence counting and time interval analysis". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 428.2-3 (1999), pp. 466–477.

[BBG19] Julien Bect, François Bachoc, and David Ginsbourger. "A supermartingale approach to Gaussian process based sequential design of experiments". In: *Bernoulli* 25.4A (2019), pp. 2883 –2919.

[BBS15] James O Berger, Jose M Bernardo, and Dongchu Sun. "Overall objective priors". In: (2015).

[BBV11] Romain Benassi, Julien Bect, and Emmanuel Vazquez. "Robust Gaussian process-based global optimization using a fully Bayesian expected improvement criterion". In: *Learning and Intelligent Optimization: 5th International Conference, LION 5, Rome, Italy, January 17-21, 2011. Selected Papers 5*. Springer. 2011, pp. 176–190.

[BC09] Alexis Boukouvalas and Dan Cornford. "Learning heteroscedastic Gaussian processes for complex datasets". In: *Group* 44.0 (2009).

[BC98] D.H Beddingfield and F.E Cecil. "Identification of fissile materials from fission product gamma-ray spectra". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 417.2 (1998), pp. 405–412.

[BCW07] Edwin V Bonilla, Kian Chai, and Christopher Williams. "Multi-task Gaussian process prediction". In: *Advances in neural information processing systems* 20 (2007).

[Bec+12] Julien Bect et al. "Sequential design of computer experiments for the estimation of a probability of failure". In: *Statistics and Computing* 22 (2012), pp. 773–793.

[Bed08] Mylene Bedard. "Optimal acceptance rates for Metropolis algorithms: Moving beyond 0.234". In: *Stochastic Processes and their Applications* 118.12 (2008), pp. 2198–2222.

[Bel65] George I Bell. "On the stochastic theory of neutron transport". In: *Nuclear Science and Engineering* 21.3 (1965), pp. 390–401.

[Ber+15] Guillaume HV Bertrand et al. "Pulse shape discrimination between (fast or thermal) neutrons and gamma rays with plastic scintillators: State of the art". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 776 (2015), pp. 114–128.

[Ber79] Jose M Bernardo. "Reference posterior distributions for Bayesian inference". In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 41.2 (1979), pp. 113–128.

[Bes+13] Alexandros Beskos et al. "Optimal tuning of the hybrid Monte Carlo algorithm". In: (2013).

[Bes+19] John D Bess et al. *The 2019 edition of the ICSBEP handbook*. Tech. rep. Idaho National Lab.(INL), Idaho Falls, ID (United States), 2019.

[Bes+20] John D Bess et al. "The 2020 Edition of the ICSBEP Handbook". In: *Nuclear Energy Agency, Organization for Economic Co-Operations and Development* (2020).

[Bet17] Michael Betancourt. "A conceptual introduction to Hamiltonian Monte Carlo". In: *arXiv preprint arXiv:1701.02434* (2017).

[BJS86] Ihor O Bohachevsky, Mark E Johnson, and Myron L Stein. "Generalized simulated annealing for function optimization". In: *Technometrics* 28.3 (1986), pp. 209–217.

[BKM17] David M Blei, Alp Kucukelbir, and Jon D McAuliffe. "Variational inference: A review for statisticians". In: *Journal of the American statistical Association* 112.518 (2017), pp. 859–877.

[Blu+15] Charles Blundell et al. "Weight uncertainty in neural network". In: *International conference on machine learning*. PMLR. 2015, pp. 1613–1622.

[Bog98] Vladimir Igorevich Bogachev. *Gaussian measures*. American Mathematical Soc., 1998.

[Böh85] K Böhnel. "The effect of multiplication on the quantitative determination of spontaneously fissioning isotopes by neutron correlation analysis". In: *Nuclear Science and Engineering* 90.1 (1985), pp. 75–82.

[Bra+02] DS Bracken et al. "Application guide to safeguards calorimetry". In: *Los Alamos Manual, LA-13867-M* (2002).

[Bro70] Charles George Broyden. "The convergence of a class of double-rank minimization algorithms 1. general considerations". In: *IMA Journal of Applied Mathematics* 6.1 (1970), pp. 76–90.

[Bru+14] Emeric Brun et al. "Tripoli-4®, CEA, EDF and AREVA reference Monte Carlo code". In: *SNA+ MC 2013-Joint International Conference on Supercomputing in Nuclear Applications+ Monte Carlo*. EDP Sciences. 2014, p. 06023.

[Bru+96] M Bruggeman et al. "Neutron coincidence counting based on time interval analysis with one-and two-dimensional Rossi-alpha distributions: an application for passive neutron waste assay". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 382.3 (1996), pp. 511–518.

[BS09] José M Bernardo and Adrian FM Smith. *Bayesian theory*. Vol. 405. John Wiley & Sons, 2009.

[BS10] Géraud Blatman and Bruno Sudret. "An adaptive algorithm to build up sparse polynomial chaos expansions for stochastic finite element analysis". In: *Probabilistic Engineering Mechanics* 25.2 (2010), pp. 183–197.

[Byr+95] Richard H Byrd et al. "A limited memory algorithm for bound constrained optimization". In: *SIAM Journal on scientific computing* 16.5 (1995), pp. 1190–1208.

[CAD13] Carl Eugene Crouthamel, Freddy Adams, and Richard Dams. *Applied gamma-ray spectrometry*. Vol. 41. Elsevier, 2013.

[Caf98] Russel E Caflisch. "Monte Carlo and quasi-monte Carlo methods". In: *Acta numerica* 7 (1998), pp. 1–49.

[CB91] TW Crane and MP Baker. "Neutron detectors". In: *Passive Nondestructive Assay of Nuclear Materials* 13 (1991), pp. 1–28.

[CCW16] David H Chambers, Hema Chandrasekaran, and Sean E Walston. "Fourier method for calculating fission chain neutron multiplicity distributions". In: *Nuclear Science and Engineering* 184.2 (2016), pp. 244–253.

[CFG14] Tianqi Chen, Emily Fox, and Carlos Guestrin. "Stochastic gradient Hamiltonian Monte Carlo". In: *International conference on machine learning*. PMLR. 2014, pp. 1683–1691.

[Che+14] Clément Chevalier et al. "Fast parallel kriging-based stepwise uncertainty reduction with application to the identification of an excursion set". In: *Technometrics* 56.4 (2014), pp. 455–465.

[Che+15] Dina Chernikova et al. "The neutron–gamma Feynman variance to mean approach: Gamma detection and total neutron–gamma detection (theory and practice)". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 782 (2015), pp. 47–55.

[CK18] Hee-Dong Choi and Junhyuck Kim. "Basic characterization of uranium by high-resolution gamma spectroscopy". In: *Nuclear Engineering and Technology* 50.6 (2018), pp. 929–936.

[CLMM09] Thierry Crestaux, Olivier Le Maıtre, and Jean-Marc Martinez. "Polynomial chaos expansion for sensitivity analysis". In: *Reliability Engineering & System Safety* 94.7 (2009), pp. 1161–1172.

[CN80] R Dennis Cook and Christopher J Nachtrheim. "A comparison of algorithms for constructing exact D-optimal designs". In: *Technometrics* 22.3 (1980), pp. 315–324.

[Coh60] Charles Erwin Cohn. "A simplified theory of pile noise". In: *Nuclear Science and Engineering* 7.5 (1960), pp. 472–475.

[Con+58] RE Connally et al. *Uranium analysis by gamma absorptiometry*. Tech. rep. General Electric Co. Hanford Atomic Products Operation, Richland, Wash., 1958.

[Coo+16] CM Cooling et al. "Uncertainty and global sensitivity analysis of neutron survival and extinction probabilities using polynomial chaos". In: *Annals of Nuclear Energy* 88 (2016), pp. 158–173.

[CRA99] HARALD CRAMÉR. "Theory of Estimation." In: *Mathematical Methods of Statistics (PMS-9)*. Princeton University Press, 1999, pp. 477–483. ISBN: 9780691005478. (Visited on 07/15/2024).

[Cro+12] S Croft et al. "Feynman variance-to-mean in the context of passive neutron coincidence counting". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 686 (2012), pp. 136–144.

[CSW12] Yohan Chalabi, David J Scott, and Diethelm Wuertz. "Flexible distribution modeling with the generalized lambda distribution". In: (2012).

[Cur+88] Carla Currin et al. *A Bayesian approach to the design and analysis of computer experiments*. Tech. rep. Oak Ridge National Lab., TN (USA), 1988.

[Dew+16] Shaheen A Dewji et al. "Validation of gamma-ray detection techniques for safeguards monitoring at natural uranium conversion facilities". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 823 (2016), pp. 135–148.

[DF+17] A Di Fulvio et al. "Passive assay of plutonium metal plates using a fast-neutron multiplicity counter". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 855 (2017), pp. 92–101.

[DF86] Persi Diaconis and David Freedman. "On the consistency of Bayes estimates". In: *The Annals of Statistics* (1986), pp. 1–26.

[DH76] James J Duderstadt and Louis J Hamilton. *Nuclear reactor analysis*. Wiley, 1976.

[DH88] Klaus Debertin and Richard G Helmer. "Gamma-and X-ray spectrometry with semiconductor detectors". In: (1988).

[DHS11] John Duchi, Elad Hazan, and Yoram Singer. "Adaptive subgradient methods for online learning and stochastic optimization." In: *Journal of machine learning research* 12.7 (2011).

[DL13] Andreas Damianou and Neil D Lawrence. "Deep Gaussian processes". In: *Artificial intelligence and statistics*. PMLR. 2013, pp. 207–215.

[Dou+18] Randal Douc et al. *Markov chains*. Vol. 1. Springer, 2018.

[DP02] Michael J Daniels and Mohsen Pourahmadi. "Bayesian analysis of covariance matrices and dynamic models for longitudinal data". In: *Biometrika* 89.3 (2002), pp. 553–566.

[DP09] Christophe Demaziere and Imre Pazsit. "Numerical tools applied to power reactor noise analysis". In: *Progress in nuclear energy* 51.1 (2009), pp. 67–81.

[DP16] Krishnan Dhamodharan and Anitha Pius. "Determination of plutonium present in highly radioactive irradiated fuel solution by spectrophotometric method". In: *Nuclear Engineering and Technology* 48.3 (2016), pp. 727–732.

[DS17] Masoumeh Dashti and Andrew M. Stuart. "The Bayesian Approach to Inverse Problems". In: *Handbook of Uncertainty Quantification*. Cham: Springer International Publishing, 2017, pp. 311–428.

[DSD13] Vincent Dubourg, Bruno Sudret, and Franois Deheeger. "Metamodel-based importance sampling for structural reliability analysis". In: *Probabilistic Engineering Mechanics* 33 (2013), pp. 47–57.

[Dur+21] Matthew Durbin et al. "K-nearest neighbors regression for the discrimination of gamma rays and neutrons in organic scintillators". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 987 (2021), p. 164826.

[Dur+23] Alain Durmus et al. "On the convergence of dynamic implementations of Hamiltonian Monte Carlo and No U-Turn Samplers". In: *arXiv preprint arXiv:2307.03460* (2023).

[DV+21] Sébastien Da Veiga et al. *Basics and trends in sensitivity analysis: Theory and practice in R*. SIAM, 2021.

[DYL18] A Drescher, M Yoho, and S Landsberger. "Gamma–gamma coincidence in neutron activation analysis". In: *Journal of Radioanalytical and Nuclear Chemistry* 318 (2018), pp. 527–532.

[EHN96] Heinz Werner Engl, Martin Hanke, and Andreas Neubauer. *Regularization of inverse problems*. Vol. 375. Springer Science & Business Media, 1996.

[Ern+12] Oliver G Ernst et al. "On the convergence of generalized polynomial chaos expansions". In: *ESAIM: Mathematical Modelling and Numerical Analysis* 46.2 (2012), pp. 317–339.

[EYY06] Tomohiro Endo, Yoshihiro Yamane, and Akio Yamamoto. "Space and energy dependent theoretical formula for the third order neutron correlation technique". In: *Annals of Nuclear Energy* 33.6 (2006), pp. 521–537.

[Fle00] Roger Fletcher. *Practical methods of optimization*. John Wiley & Sons, 2000.

[Fle70] Roger Fletcher. "A new approach to variable metric algorithms". In: *The computer journal* 13.3 (1970), pp. 317–322.

[FLS05] Kai-Tai Fang, Runze Li, and Agus Sudjianto. *Design and modeling for computer experiments*. Chapman and Hall/CRC, 2005.

[FMM18] Mikhail Figurnov, Shakir Mohamed, and Andriy Mnih. "Implicit reparameterization gradients". In: *Advances in neural information processing systems* 31 (2018).

[For+21] Vincent Fortuin et al. "Bayesian neural network priors revisited". In: *arXiv preprint arXiv:2102.06571* (2021).

[For22] Vincent Fortuin. "Priors in Bayesian deep learning: A review". In: *International Statistical Review* 90.3 (2022), pp. 563–591.

[FPD08] Peter I Frazier, Warren B Powell, and Savas Dayanik. "A knowledge-gradient policy for sequential information collection". In: *SIAM Journal on Control and Optimization* 47.5 (2008), pp. 2410–2439.

[FPD09] Peter Frazier, Warren Powell, and Savas Dayanik. "The knowledge-gradient policy for correlated normal beliefs". In: *INFORMS journal on Computing* 21.4 (2009), pp. 599–613.

[Fre+88] Marshall Freimer et al. "A study of the generalized Tukey lambda family". In: *Communications in Statistics-Theory and Methods* 17.10 (1988), pp. 3547–3567.

[Fry71] DN Fry. "Experience in reactor malfunction diagnosis using on-line noise analysis". In: *Nuclear Technology* 10.3 (1971), pp. 273–282.

[Gar+18] Jacob Gardner et al. "Gpytorch: Blackbox matrix-matrix gaussian process inference with gpu acceleration". In: *Advances in neural information processing systems* 31 (2018).

[Gar23] Josselin Garnier. *MAP 568 - Gestion des incertitudes et analyse de risque*. Cours de l'École polytechnique. 2023.

[GC11] Mark Girolami and Ben Calderhead. "Riemann manifold Langevin and Hamiltonian Monte Carlo methods". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 73.2 (2011), pp. 123–214.

[GF11] G Gualdrini and P Ferrari. "Monte Carlo variance reduction techniques: an overview with some practical examples". In: *Radiation protection dosimetry* 146.4 (2011), pp. 425–433.

[GG16] Yarin Gal and Zoubin Ghahramani. "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning". In: *international conference on machine learning*. PMLR. 2016, pp. 1050–1059.

[GGR97] Andrew Gelman, Walter R Gilks, and Gareth O Roberts. "Weak convergence and optimal scaling of random walk Metropolis algorithms". In: *The annals of applied probability* 7.1 (1997), pp. 110–120.

[Gho11] Malay Ghosh. "Objective priors: An introduction for frequentists". In: (2011).

[GHO99] Gene H Golub, Per Christian Hansen, and Dianne P O'Leary. "Tikhonov regularization and total least squares". In: *SIAM journal on matrix analysis and applications* 21.1 (1999), pp. 185–194.

[Goh+19] Hwan Goh et al. "Solving Bayesian inverse problems via variational autoencoders". In: *arXiv preprint arXiv:1912.04212* (2019).

[Gol70] Donald Goldfarb. "A family of variable-metric methods derived by variational means". In: *Mathematics of computation* 24.109 (1970), pp. 23–26.

[Goo+12] T Goorley et al. "Initial MCNP6 release overview". In: *Nuclear technology* 180.3 (2012), pp. 298–315.

[GR11] J. Ghosh and R. Ramamoorthi. "Bayesian Nonparametrics". In: *Springer Series in Statistics* 16 (Jan. 2011).

[Gra11] Alex Graves. "Practical variational inference for neural networks". In: *Advances in neural information processing systems* 24 (2011).

[Gra20] Robert B Gramacy. *Surrogates: Gaussian process modeling, design, and optimization for the applied sciences*. Chapman and Hall/CRC, 2020.

[Gri+20] J Griffiths et al. "Pulse shape discrimination and exploration of scintillation signals using convolutional neural networks". In: *Machine Learning: Science and Technology* 1.4 (2020), p. 045022.

[GY20] Andrew Gelman and Yuling Yao. "Holes in Bayesian statistics". In: *Journal of Physics G: Nuclear and Particle Physics* 48.1 (2020), p. 014002.

[Han60] GE Hansen. "Assembly of fissionable material in the presence of a weak neutron source". In: *Nuclear Science and Engineering* 8.6 (1960), pp. 709–719.

[Har56] Theodore E Harris. "The existence of stationary measures for certain Markov processes". In: *Proceedings of the Third Berkeley Symposium on Mathematical Statistics and Probability*. Vol. 2. 1956, pp. 113–124.

[Har+63] Theodore Edward Harris et al. *The theory of branching processes*. Vol. 6. Springer Berlin, 1963.

[Har65] DR Harris. "Neutron fluctuations in a reactor of finite size". In: *Nuclear Science and Engineering* 21.3 (1965), pp. 369–381.

[Has70] W Keith Hastings. "Monte Carlo sampling methods using Markov chains and their applications". In: (1970).

[HD03] Jon C Helton and Freddie Joe Davis. "Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems". In: *Reliability Engineering & System Safety* 81.1 (2003), pp. 23–69.

[He09] Xiaofei He. "Laplacian regularized d-optimal design for active learning and its application to image retrieval". In: *IEEE Transactions on Image Processing* 19.1 (2009), pp. 254–263.

[Hen+18] Brian S Henderson et al. "Experimental demonstration of multiple monoenergetic gamma radiography for effective atomic number identification in cargo inspection". In: *Journal of Applied Physics* 123.16 (2018).

[HG+14] Matthew D Hoffman, Andrew Gelman, et al. "The No-U-Turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo." In: *J. Mach. Learn. Res.* 15.1 (2014), pp. 1593–1623.

[HG18] Joseph Hart and Pierre A Gremaud. "An approximation theoretic perspective of Sobol'indices with dependent variables". In: *International Journal for Uncertainty Quantification* 8.6 (2018).

[Hin+12] Geoffrey E Hinton et al. "Improving neural networks by preventing co-adaptation of feature detectors". In: *arXiv preprint arXiv:1207.0580* (2012).

[HJ+47] Cecil Hastings Jr et al. "Low moments for small samples: a comparative study of order statistics". In: *The Annals of Mathematical Statistics* 18.3 (1947), pp. 413–426.

[HM04] Philippe Humbert and Boukhmes Méchitoua. "Simulation of Caliban reactor burst wait time and initiation probability using a point reactor model and PANDA code". In: *Proc. Physics of Fuel Cycles and Advanced Nuclear Systems: Global Developments (PHYSOR 2004), Chicago, Illinois* (2004).

[Hof+13] Matthew D Hoffman et al. "Stochastic variational inference". In: *Journal of Machine Learning Research* (2013).

[Hou18] Xiaolin Hou. "Liquid scintillation counting for determination of radionuclides in environmental and nuclear application". In: *Journal of Radioanalytical and Nuclear Chemistry* 318.3 (2018), pp. 1597–1628.

[HPL08] Rebecca E Hamon, David R Parker, and Enzo Lombi. "Advances in isotopic dilution techniques in trace element research: a review of methodologies, benefits, and limitations". In: *Advances in agronomy* 99 (2008), pp. 289–343.

[HS97] Sepp Hochreiter and Jürgen Schmidhuber. "Long short-term memory". In: *Neural computation* 9.8 (1997), pp. 1735–1780.

[HST01] Heikki Haario, Eero Saksman, and Johanna Tamminen. "An adaptive Metropolis algorithm". In: *Bernoulli* (2001), pp. 223–242.

[HST99] Heikki Haario, Eero Saksman, and Johanna Tamminen. "Adaptive Proposal distribution for random walk Metropolis algorithm". In: *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete* 47.2 (1999), pp. 119–137.

[Hua+20a] Michael Y Hua et al. "Derivation of the two-exponential probability density function for Rossi-alpha measurements of reflected assemblies and validation for the special case of shielded measurements". In: *Nuclear Science and Engineering* 194.1 (2020), pp. 56–68.

[Hua+20b] Michael Y Hua et al. "Validation of the two-region Rossi-alpha model for reflected assemblies". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 981 (2020), p. 164535.

[Hum19] Philippe Humbert. "Deterministic transport solution of multiplicity counting equations". In: *International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering*. 2019.

[Hus+11] Bart GM Husslage et al. "Space-filling Latin hypercube designs for computer experiments". In: *Optimization and Engineering* 12 (2011), pp. 611–630.

[HVC93] Geoffrey E Hinton and Drew Van Camp. "Keeping the neural networks simple by minimizing the description length of the weights". In: *Proceedings of the sixth annual conference on Computational Learning Theory*. 1993, pp. 5–13.

[HW03] Alireza Haghighat and John C Wagner. "Monte Carlo variance reduction with deterministic importance functions". In: *Progress in Nuclear Energy* 42.1 (2003), pp. 25–53.

[HW13] Alan Huang and Matthew P Wand. "Simple marginally noninformative prior distributions for covariance matrices". In: (2013).

[HWP78] Michael K Holland, Jon R Weiss, and Charles E Pietri. "Controlled-potential coulometric determination of plutonium". In: *Analytical Chemistry* 50.2 (1978), pp. 236–240.

[HZ02] Feifang Hu and James V. Zidek. "The weighted likelihood". In: *Canadian Journal of Statistics* 30.3 (2002), pp. 347–371.

[Ing54] MARK G Inghram. "Stable isotope dilution as an analytical tool". In: *Annual Review of Nuclear Science* 4.1 (1954), pp. 81–92.

[Jef46] Harold Jeffreys. "An invariant form for the prior probability in estimation problems". In: *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences* 186.1007 (1946), pp. 453–461.

[JH00] Søren Fiig Jarner and Ernst Hansen. "Geometric ergodicity of Metropolis algorithms". In: *Stochastic processes and their applications* 85.2 (2000), pp. 341–361.

[JJ00] Tommi S Jaakkola and Michael I Jordan. "Bayesian parameter estimation via variational methods". In: *Statistics and Computing* 10 (2000), pp. 25–37.

[JMY90] Mark E Johnson, Leslie M Moore, and Donald Ylvisaker. "Minimax and maximin distance designs". In: *Journal of statistical planning and inference* 26.2 (1990), pp. 131–148.

[JSW98] Donald R Jones, Matthias Schonlau, and William J Welch. "Efficient global optimization of expensive black-box functions". In: *Journal of Global optimization* 13 (1998), pp. 455–492.

[Kar+86] CV Karekar et al. "Determination of plutonium and uranium in the same aliquot by potentiometric titration". In: *Journal of radioanalytical and nuclear chemistry* 107 (1986), pp. 297–305.

[Kat+07] JI Katz et al. "X-radiography of cargo containers". In: *Science & Global Security* 15.1 (2007), pp. 49–56.

[KB14] Diederik P Kingma and Jimmy Ba. "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980* (2014).

[KCG24] Baptiste Kerleguer, Claire Cannamela, and Josselin Garnier. "A Bayesian neural network approach to multi-fidelity surrogate modeling". In: *International Journal for Uncertainty Quantification* 14.1 (2024).

[KE05] Y Kaschuck and B Esposito. "Neutron/γ-ray digital pulse shape discrimination with organic scintillators". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 551.2-3 (2005), pp. 420–428.

[Ker+07] Kristian Kersting et al. "Most likely heteroscedastic Gaussian process regression". In: *Proceedings of the 24th international conference on Machine learning*. 2007, pp. 393–400.

[KL51] Solomon Kullback and Richard A Leibler. "On information and sufficiency". In: *The annals of mathematical statistics* 22.1 (1951), pp. 79–86.

[KO01] Marc C Kennedy and Anthony O'Hagan. "Bayesian calibration of computer models". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 63.3 (2001), pp. 425–464.

[Koe81] Roger Koenker. "A note on studentizing a test for heteroscedasticity". In: *Journal of econometrics* 17.1 (1981), pp. 107–112.

[KRA+09] Farheen Khan, Nafisur Rahman, Syed Najmul Hejaz Azmi, et al. "Spectrophotometric determination of uranium (VI) via complexation with piroxicam". In: (2009).

[KRR10] Jack PC Kleijnen, Ad Ridder, and Reuven Rubinstein. "Variance reduction techniques in Monte Carlo methods". In: (2010).

[KS06] Jari Kaipio and Erkki Somersalo. *Statistical and computational inverse problems*. Vol. 160. Springer Science & Business Media, 2006.

[KSW15] Durk P Kingma, Tim Salimans, and Max Welling. "Variational dropout and the local reparameterization trick". In: *Advances in neural information processing systems* 28 (2015).

[KV12] Bas JK Kleijn and Aad W Van der Vaart. "The Bernstein-von-Mises theorem under misspecification". In: (2012).

[KW13] Diederik P Kingma and Max Welling. "Auto-encoding variational bayes". In: *arXiv preprint arXiv:1312.6114* (2013).

[KW96] Robert E Kass and Larry Wasserman. "The selection of prior distributions by formal rules". In: *Journal of the American statistical Association* 91.435 (1996), pp. 1343–1370.

[LB+95] Yann LeCun, Yoshua Bengio, et al. "Convolutional networks for images, speech, and time series". In: *The handbook of brain theory and neural networks* 3361.10 (1995), p. 1995.

[Lee+14] Seung Kyu Lee et al. "Development of large-area composite stilbene scintillator for fast neutron detection". In: *Journal of Nuclear Science and Technology* 51.1 (2014), pp. 37–47.

[Leh99] Erich Leo Lehmann. *Elements of large-sample theory*. Springer, 1999.

[LGMS16] Loïc Le Gratiet, Stefano Marelli, and Bruno Sudret. "Metamodel-based sensitivity analysis: polynomial chaos expansions and Gaussian processes". In: *arXiv preprint arXiv:1606.04273* (2016).

[LHG22] Paul Lartaud, Philippe Humbert, and Josselin Garnier. "Uncertainty quantification in neutron noise analysis using Monte Carlo Markov chain methods: an application to nuclear waste drum assay". In: *Proceedings of the International Conference on Physics of Reactors*. 2022, pp. 2674–2683.

[LHG23] Paul Lartaud, Philippe Humbert, and Josselin Garnier. "Multi-output Gaussian processes for inverse uncertainty quantification in neutron noise analysis". In: *Nuclear Science and Engineering* 197.8 (2023), pp. 1928–1951.

[LHG24a] Paul Lartaud, Philippe Humbert, and Josselin Garnier. "Bayesian Inverse Problem and Uncertainty Quantification in the Joint Analysis of Neutron and Gamma Corrrelations". In: *Proceedings of the International Conference on Physics of Reactors*. 2024, pp. 541–550.

[LHG24b] Paul Lartaud, Philippe Humbert, and Josselin Garnier. "Sequential design for surrogate modeling in Bayesian inverse problems". In: *arXiv preprint arXiv:2402.16520* (2024).

[Liu+21] Yehao Liu et al. "The peril of popular deep learning uncertainty estimation methods". In: *arXiv preprint arXiv:2112.05000* (2021).

[LJ08] Tae Hee Lee and Jae Jun Jung. "A sampling technique enhancing accuracy and efficiency of metamodel-based RBDO: Constraint boundary sampling". In: *Computers & Structures* 86.13-14 (2008), pp. 1463–1476.

[LKJ09] Daniel Lewandowski, Dorota Kurowicka, and Harry Joe. "Generating random correlation matrices based on vines and extended onion method". In: *Journal of multivariate analysis* 100.9 (2009), pp. 1989–2001.

[LN89] Dong C Liu and Jorge Nocedal. "On the limited memory BFGS method for large scale optimization". In: *Mathematical programming* 45.1-3 (1989), pp. 503–528.

[Loc00] Marco Locatelli. "Simulated annealing algorithms for continuous global optimization: convergence conditions". In: *Journal of Optimization Theory and applications* 104 (2000), pp. 121–133.

[LPB17] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. "Simple and scalable predictive uncertainty estimation using deep ensembles". In: *Advances in neural information processing systems* 30 (2017).

[Lut+07] Gerhard Lutz et al. *Semiconductor radiation detectors*. Springer, 2007.

[Mac92] David J. C. MacKay. "A Practical Bayesian Framework for Backpropagation Networks". In: *Neural Computation* 4.3 (May 1992), pp. 448–472.

[Mao15] Xuerong Mao. "The truncated Euler–Maruyama method for stochastic differential equations". In: *Journal of Computational and Applied Mathematics* 290 (2015), pp. 370–384.

[Mar+12] Amandine Marrel et al. "Global sensitivity analysis of stochastic computer models with joint metamodels". In: *Statistics and Computing* 22 (2012), pp. 833–847.

[Mat12] John Mattingly. "Computation of neutron multiplicity statistics using deterministic transport". In: *IEEE Transactions on Nuclear Science* 59.2 (2012), pp. 314–322.

[MD15] T Malinovitch and C Dubi. "A multi-region multi-energy formalism for the Feynman-alpha formulas". In: *Annals of Nuclear Energy* 76 (2015), pp. 297–304.

[MDM06] NL Misra, S Dhara, and KD Singh Mudher. "Uranium determination in seawater by total reflection X-ray fluorescence spectrometry". In: *Spectrochimica Acta Part B: Atomic Spectroscopy* 61.10-11 (2006), pp. 1166–1169.

[Met+53] Nicholas Metropolis et al. "Equation of state calculations by fast computing machines". In: *The journal of chemical physics* 21.6 (1953), pp. 1087–1092.

[Mil18] Jeffrey W Miller. "A detailed treatment of Doob's theorem". In: *arXiv preprint arXiv:1801.03122* (2018).

[Mil21] Jeffrey W Miller. "Asymptotic normality, concentration, and coverage of generalized posteriors". In: *Journal of Machine Learning Research* 22.168 (2021), pp. 1–53.

[MM15] Dmytro Mishkin and Jiri Matas. "All you need is a good init". In: *arXiv preprint arXiv:1511.06422* (2015).

[Moo96] Todd K Moon. "The expectation-maximization algorithm". In: *IEEE Signal processing magazine* 13.6 (1996), pp. 47–60.

[MRH15] Kathryn G McIntosh, Sean D Reilly, and George J Havrilla. "Determination of plutonium in spent nuclear fuel using high resolution X-ray". In: *Spectrochimica Acta Part B: Atomic Spectroscopy* 110 (2015), pp. 91–95.

[MT12] Sean P Meyn and Richard L Tweedie. *Markov chains and stochastic stability*. Springer Science & Business Media, 2012.

[MT93] Sean P Meyn and Richard L Tweedie. "Stability of Markovian processes II: Continuous-time processes and sampled chains". In: *Advances in Applied Probability* 25.3 (1993), pp. 487–517.

[MW92] J Steve Marron and Matt P Wand. "Exact mean integrated squared error". In: *The Annals of Statistics* 20.2 (1992), pp. 712–736.

[Nea01] Radford M Neal. "Annealed importance sampling". In: *Statistics and computing* 11 (2001), pp. 125–139.

[Nea+11] Radford M Neal et al. "MCMC using Hamiltonian dynamics". In: *Handbook of Markov chain monte carlo* 2.11 (2011), p. 2.

[Nea12] Radford M Neal. *Bayesian learning for neural networks*. Vol. 118. Springer Science & Business Media, 2012.

[Nes09] Yurii Nesterov. "Primal-dual subgradient methods for convex problems". In: *Mathematical programming* 120.1 (2009), pp. 221–259.

[Nie92] Harald Niederreiter. *Random number generation and quasi-Monte Carlo methods*. SIAM, 1992.

[Nor98] James R Norris. *Markov chains*. 2. Cambridge university press, 1998.

[OA09] Manfred Opper and Cédric Archambeau. "The variational Gaussian approximation revisited". In: *Neural computation* 21.3 (2009), pp. 786–792.

[OI+16] Buckley E O'Day III et al. "Initial results from a multiple monoenergetic gamma radiography system for nuclear security". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 832 (2016), pp. 68–76.

[OI65] M Otsuka and T Iijima. "SPACE-DEPENDENT FORMULA FOR ROSSI-*alpha* MEASUREMENTS". In: *Nukleonik (West Germany) Discontinued with vol. 12* 7 (1965).

[OMA17] Sean O'Brien, John Mattingly, and Dmitriy Anistratov. "Sensitivity analysis of neutron multiplicity counting statistics using first-order perturbation theory and application to a subcritical plutonium metal benchmark". In: *Nuclear Science and Engineering* 185.3 (2017), pp. 406–425.

[OV+06] Javier Ortiz-Villafuerte et al. "BWR online monitoring system based on noise analysis". In: *Nuclear engineering and design* 236.22 (2006), pp. 2394–2404.

[PA80] I Pázsit and G Th Analytis. "Theoretical investigation of the neutron noise diagnostics of two-dimensional control rod vibrations in a PWR". In: *Annals of Nuclear Energy* 7.3 (1980), pp. 171–183.

[Pak+18] O Pakari et al. "Current mode neutron noise measurements in the zero power reactor CROCUS". In: *EPJ Web of Conferences*. Vol. 170. EDP Sciences. 2018, p. 04017.

[Pal58] Lo Pal. "On the theory of stochastic processes in nuclear reactors". In: *Nuovo Cimento (Italy) Divided into Nuovo Cimento A and Nuovo Cimento B* 10 (1958).

[PBJ19] HMO Parker, JS Beaumont, and Malcolm John Joyce. "Passive, non-intrusive assay of depleted uranium". In: *Journal of hazardous materials* 364 (2019), pp. 293–299.

[Peu00] Anthony J Peurrung. "Recent developments in neutron detection". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 443.2-3 (2000), pp. 400–415.

[PF18] Remus Pop and Patric Fulop. "Deep ensemble Bayesian active learning: Addressing the mode collapse issue in Monte Carlo dropout via ensembles". In: *arXiv preprint arXiv:1811.03897* (2018).

[PH83] Adrian Rodney Pagan and Anthony David Hall. "Diagnostic tests as residual analysis". In: *Econometric Reviews* 2.2 (1983), pp. 159–218.

[Pic+10] Victor Picheny et al. "Adaptive designs of experiments for accurate approximation of a target region". In: (2010).

[Pic14] Victor Picheny. "A stepwise uncertainty reduction approach to constrained global optimization". In: *Artificial intelligence and statistics*. PMLR. 2014, pp. 787–795.

[Pic15] Victor Picheny. "Multiobjective optimization using Gaussian process emulators via stepwise uncertainty reduction". In: *Statistics and Computing* 25.6 (2015), pp. 1265–1280.

[PLS23] Dmytro Perepolkin, Erik Lindsröm, and Ullrika Sahlin. "Quantile-parameterized distributions for expert knowledge elicitation". In: (2023).

[PP05] Imre Pázsit and Sara A Pozzi. "Calculation of gamma multiplicities in a multiplying sample for the assay of nuclear materials". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 555.1-2 (2005), pp. 340–346.

[PP07] Imre Pázsit and Lénard Pál. *Neutron fluctuations: A treatise on the physics of branching processes*. Elsevier, 2007.

[PP21] Imre Pázsit and Lénárd Pál. "Multiplicity theory beyond the point model". In: *Annals of Nuclear Energy* 154 (2021), p. 108119.

[PPM03] Sara A Pozzi, Enrico Padovani, and Marzio Marseguerra. "MCNP-PoliMi: a Monte-Carlo code for correlation measurements". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 513.3 (2003), pp. 550–558.

[PRO22] Dhruv V Patel, Deep Ray, and Assad A Oberai. "Solution of physics-based Bayesian inverse problems with deep generative priors". In: *Computer Methods in Applied Mechanics and Engineering* 400 (2022), p. 115428.

[Ray+22] Deep Ray et al. "The efficacy and generalizability of conditional GANs for posterior inference in physics-based inverse problems". In: *arXiv preprint arXiv:2202.07773* (2022).

[RBB18] Hippolyt Ritter, Aleksandar Botev, and David Barber. "A scalable Laplace approximation for neural networks". In: *6th international conference on learning representations, ICLR 2018-conference track proceedings*. Vol. 6. International Conference on Representation Learning. 2018.

[RBS10] Timothy J Robinson, Jeffrey B Birch, and B Alden Starnes. "A semi-parametric approach to dual modeling when no replication exists". In: *Journal of Statistical Planning and Inference* 140.10 (2010), pp. 2860–2869.

[RCC99] Christian P Robert, George Casella, and George Casella. *Monte Carlo statistical methods*. Vol. 2. Springer, 1999.

[RG41] Bruno Rossi and Kenneth Greisen. "Cosmic-ray theory". In: *Reviews of Modern Physics* 13.4 (1941), p. 240.

[Rob52] Herbert Robbins. "Some aspects of the sequential design of experiments". In: (1952).

[RR04] Gareth O Roberts and Jeffrey S Rosenthal. "General state space Markov chains and MCMC algorithms". In: (2004).

[RR98] Gareth O Roberts and Jeffrey S Rosenthal. "Optimal scaling of discrete approximations to Langevin diffusions". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 60.1 (1998), pp. 255–268.

[RS74] John S Ramberg and Bruce W Schmeiser. "An approximate method for generating asymmetric random variables". In: *Communications of the ACM* 17.2 (1974), pp. 78–82.

[Rus+09] Kenneth G Russell et al. "D-optimal designs for Poisson regression models". In: *Statistica Sinica* (2009), pp. 721–730.

[RW+06] Carl Edward Rasmussen, Christopher KI Williams, et al. *Gaussian processes for machine learning*. Vol. 1. Springer, 2006.

[Sak92] Remi M Sakia. "The Box-Cox transformation technique: a review". In: *Journal of the Royal Statistical Society: Series D (The Statistician)* 41.2 (1992), pp. 169–178.

[San+03] Thomas J Santner et al. *The design and analysis of computer experiments*. Vol. 1. Springer, 2003.

[SB00] Alex Smola and Peter Bartlett. "Sparse greedy Gaussian process regression". In: *Advances in neural information processing systems* 13 (2000).

[SC13] Bruno Sudret and Yves Caniou. *Analysis of covariance (ANCOVA) using polynomial chaos expansions*. CRC Press/Balkema, 2013.

[SFP11] Warren Scott, Peter Frazier, and Warren Powell. "The correlated knowledge gradient for simulation optimization of continuous parameters using Gaussian process regression". In: *SIAM Journal on Optimization* 21.3 (2011), pp. 996–1026.

[SG05] Edward Snelson and Zoubin Ghahramani. "Sparse Gaussian processes using pseudo-inputs". In: *Advances in neural information processing systems* 18 (2005).

[SH87] Harold Szu and Ralph Hartley. "Fast simulated annealing". In: *Physics letters A* 122.3-4 (1987), pp. 157–162.

[Sha+15] Bobak Shahriari et al. "Taking the human out of the loop: A review of Bayesian optimization". In: *Proceedings of the IEEE* 104.1 (2015), pp. 148–175.

[Sha70] David F Shanno. "Conditioning of quasi-Newton methods for function minimization". In: *Mathematics of computation* 24.111 (1970), pp. 647–656.

[Sif+75] P Siffert et al. "Cadmium telluride nuclear radiation detectors". In: *IEEE Transactions on Nuclear Science* 22.1 (1975), pp. 211–225.

[SJ60] G Stolz Jr. "Numerical solutions to an inverse problem of heat conduction for simple shapes". In: (1960).

[Skl59] M Sklar. "Fonctions de répartition à n dimensions et leurs marges". In: *Annales de l'ISUP*. Vol. 8. 3. 1959, pp. 229–231.

[SLA12] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. "Practical Bayesian optimization of machine learning algorithms". In: *Advances in neural information processing systems* 25 (2012).

[SMG13] Andrew M Saxe, James L McClelland, and Surya Ganguli. "Exact solutions to the nonlinear dynamics of learning in deep linear neural networks". In: *arXiv preprint arXiv:1312.6120* (2013).

[SN17] Michael Sinsbeck and Wolfgang Nowak. "Sequential design of computer experiments for the solution of Bayesian inverse problems". In: *SIAM/ASA Journal on Uncertainty Quantification* 5.1 (2017), pp. 640–664.

[Sob93] IM SoboĺI. "Sensitivity estimates for nonlinear mathematical models". In: *Math. Model. Comput. Exp.* 1 (1993), p. 407.

[Son+20] Yang Song et al. "Score-based generative modeling through stochastic differential equations". In: *arXiv preprint arXiv:2011.13456* (2020).

[Son+23] Jiaming Song et al. "Pseudoinverse-guided diffusion models for inverse problems". In: *International Conference on Learning Representations*. 2023.

[SP97] Mike Schuster and Kuldip K Paliwal. "Bidirectional recurrent neural networks". In: *IEEE Transactions on Signal Processing* 45.11 (1997), pp. 2673–2681.

[SPE18] JEM Saxby, Anil K Prinja, and MD Eaton. "Energy dependent Transport Model of the neutron number probability distribution in a subcritical multiplying assembly". In: *Nuclear Science and Engineering* 189.1 (2018), pp. 1–25.

[SPW23] Özge Sürer, Matthew Plumlee, and Stefan M Wild. "Sequential Bayesian experimental design for calibration of expensive simulation models". In: *Technometrics* 0.ja (2023), pp. 1–26. DOI: 10.1080/00401706.2023.2246157.

[SSW89] Jerome Sacks, Susannah B Schiller, and William J Welch. "Designs for computer experiments". In: *Technometrics* 31.1 (1989), pp. 41–47.

[Ste12] Michael L Stein. *Interpolation of spatial data: some theory for kriging*. Springer Science & Business Media, 2012.

[Ste87] Michael Stein. "Large sample properties of simulations using Latin hypercube sampling". In: *Technometrics* 29.2 (1987), pp. 143–151.

[Str+22] Rémi Stroh et al. "Sequential design of multi-fidelity computer experiments: maximizing the rate of stepwise uncertainty reduction". In: *Technometrics* 64.2 (2022), pp. 199–209.

[Stu10] Andrew M Stuart. "Inverse problems: a Bayesian perspective". In: *Acta numerica* 19 (2010), pp. 451–559.

[Sud08] Bruno Sudret. "Global sensitivity analysis using polynomial chaos expansions". In: *Reliability engineering & system safety* 93.7 (2008), pp. 964–979.

[SW87] Michael C Shewry and Henry P Wynn. "Maximum entropy sampling". In: *Journal of applied statistics* 14.2 (1987), pp. 165–170.

[SWL03] Matthias W Seeger, Christopher KI Williams, and Neil D Lawrence. "Fast forward selection to speed up sparse Gaussian process regression". In: *International Workshop on Artificial Intelligence and Statistics*. PMLR. 2003, pp. 254–261.

[Tan24] Francesco Tantillo. "New class of solutions of the ultimate extinction probability from Galton-Watson processes". In: *Nuclear Engineering and Design* 417 (2024), p. 112827.

[Tar05] Albert Tarantola. *Inverse problem theory and methods for model parameter estimation*. SIAM, 2005.

[TD20] Daniel J Tait and Theodoros Damoulas. "Variational autoencoding of PDE inverse problems". In: *arXiv preprint arXiv:2006.15641* (2020).

[Tie94] Luke Tierney. "Markov chains for exploring posterior distributions". In: *the Annals of Statistics* (1994), pp. 1701–1728.

[Tik+43] Andrey Nikolayevich Tikhonov et al. "On the stability of inverse problems". In: *Dokl. akad. nauk sssr*. Vol. 39. 5. 1943, pp. 195–198.

[Tit09] Michalis Titsias. "Variational learning of inducing variables in sparse Gaussian processes". In: *Artificial intelligence and statistics*. PMLR. 2009, pp. 567–574.

[Toh+16] Mona Tohamy et al. "Passive Non-Destructive Assay based on gamma-ray spectrometry to verify UO2 samples in the form of powder and pellet". In: *Annals of Nuclear Energy* 87 (2016), pp. 186–191.

[TS96] Constantino Tsallis and Daniel A Stariolo. "Generalized simulated annealing". In: *Physica A: Statistical Mechanics and its Applications* 233.1-2 (1996), pp. 395–406.

[Twe01] RL Tweedie. "Drift conditions and invariant measures for Markov chains". In: *Stochastic processes and their applications* 92.2 (2001), pp. 345–354.

[Twe75] Richard L Tweedie. "Sufficient conditions for ergodicity and recurrence of Markov chains on a general state space". In: *Stochastic Processes and their Applications* 3.4 (1975), pp. 385–403.

[UCJ21] Anders Kirk Uhrenholt, Valentin Charvet, and Bjørn Sand Jensen. "Probabilistic selection of inducing points in sparse Gaussian processes". In: *Uncertainty in Artificial Intelligence*. PMLR. 2021, pp. 1035–1044.

[Vaa00] Aad W Van der Vaart. *Asymptotic statistics*. Vol. 3. Cambridge university press, 2000.

[VL+87] Peter JM Van Laarhoven et al. *Simulated annealing*. Springer, 1987.

[VRV15] Jerome M Verbeke, J Randrup, and Ramona Vogt. "Fission Reaction Event Yield Algorithm, FREYA—For event-by-event simulation of fission". In: *Computer Physics Communications* 191 (2015), pp. 178–202.

[VTC87] Nicholas Vakhania, Vazha Tarieladze, and S Chobanyan. *Probability distributions on Banach spaces*. Vol. 14. Springer Science & Business Media, 1987.

[VVW09] Julien Villemonteix, Emmanuel Vazquez, and Eric Walter. "An informational approach to the global optimization of expensive-to-evaluate functions". In: *Journal of Global Optimization* 44 (2009), pp. 509–534.

[VZ+08] Aad W van der Vaart, J Harry van Zanten, et al. "Reproducing kernel Hilbert spaces of Gaussian priors". In: *IMS Collections* 3 (2008), pp. 200–222.

[Wah90] Grace Wahba. *Spline models for observational data*. SIAM, 1990.

[Wan+14] Donghui Wang et al. "Structural design employing a sequential approximation optimization approach". In: *Computers & structures* 134 (2014), pp. 75–87.

[Web81] Charles F Weber. "Analysis and solution of the ill-posed inverse heat conduction problem". In: *International Journal of Heat and Mass Transfer* 24.11 (1981), pp. 1783–1792.

[Wie38] Norbert Wiener. "The homogeneous chaos". In: *American Journal of Mathematics* 60.4 (1938), pp. 897–936.

[Wil06] Christofer Willman. "Applications of gamma ray spectroscopy of spent nuclear fuel for safeguards and encapsulation". PhD thesis. Acta Universitatis Upsaliensis, 2006.

[Wil16] MMR Williams. "Burst wait-times in the Caliban reactor using the Gamma probability distribution function". In: *Nuclear Science and Engineering* 183.1 (2016), pp. 116–125.

[Wil79] MMR Williams. "An exact solution of the extinction problem in supercritical multiplying systems". In: *Annals of Nuclear Energy* 6.9-10 (1979), pp. 463–472.

[WP15] MMR Williams and I Pázsit. "The time dependence of the extinction probability with delayed neutrons". In: *Annals of Nuclear Energy* 75 (2015), pp. 107–115.

[Wri06] Stephen J Wright. *Numerical optimization*. 2006.

[WT11] Max Welling and Yee W Teh. "Bayesian learning via stochastic gradient Langevin dynamics". In: *Proceedings of the 28th international conference on machine learning (ICML-11)*. Citeseer. 2011, pp. 681–688.

[Wu+21] Luhuan Wu et al. "Hierarchical inducing point Gaussian process for inter-domian observations". In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2021, pp. 2926–2934.

[Xia+97] Yang Xiang et al. "Generalized simulated annealing algorithm and its application to the Thomson model". In: *Physics Letters A* 233.3 (1997), pp. 216–220.

[YHK22] Masahiko Yamamoto, Kazushi Horigome, and Takehiko Kuno. "Accurate and precise measurement of uranium content in uranium trioxide by gravimetry: Comparison with isotope dilution mass spectrometry and its uncertainty estimation". In: *Applied Radiation and Isotopes* 190 (2022), p. 110460. ISSN: 0969-8043. DOI: https://doi.org/10.1016/j.apradiso.2022.110460.

[YJ00] In-Kwon Yeo and Richard A Johnson. "A new family of power transformations to improve normality or symmetry". In: *Biometrika* 87.4 (2000), pp. 954–959.

[YWF15] Takayuki Yanagida, Kenichi Watanabe, and Yutaka Fujimoto. "Comparative study of neutron and gamma-ray pulse shape discrimination of anthracene, stilbene, and p-terphenyl". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 784 (2015), pp. 111–114.

[Zai+12] Natalia Zaitseva et al. "Plastic scintillators with efficient neutron/gamma pulse shape discrimination". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 668 (2012), pp. 88–93.

[Zha+19] Ruqi Zhang et al. "Cyclical stochastic gradient MCMC for Bayesian deep learning". In: *arXiv preprint arXiv:1902.03932* (2019).

[ZLR19] Ru Zhang, C Devon Lin, and Pritam Ranjan. "A sequential design approach for calibrating dynamic computer simulators". In: *SIAM/ASA Journal on Uncertainty Quantification* 7.4 (2019), pp. 1245–1274.

[ZS20] Xujia Zhu and Bruno Sudret. "Replication-based emulation of the response distribution of stochastic simulators using generalized lambda distributions". In: *International Journal for Uncertainty Quantification* 10.3 (2020).

[ZS21] Xujia Zhu and Bruno Sudret. "Emulation of stochastic simulators using generalized lambda models". In: *SIAM/ASA Journal on Uncertainty Quantification* 9.4 (2021), pp. 1345–1380.

Appendix A

Descriptive review of the benchmark and test cases

A.1 ICSBEP Benchmark

The first test case is an experimental configuration studied in the context of the International Criticality Safety Benchmark Evaluation Project (ICSBEP). This project, started by the US Department of Energy, became an international cooperation project of the Nuclear Energy Agency (NEA) in 1995. The goal of the project was to compile criticality safety experiments into one single benchmark. The complete benchmark contains almost 5000 different configurations, with various multiplications, fuels, shielding materials, and detectors.

The test case of interest is the experiment **FUND-NCERC-PU-HE3-MULT-003**. This experiment regroups 17 subcritical configurations of an α-phase metallic plutonium sphere surrounded by a varying thickness of copper or polyethylene, acting as the reflector material. The particular case of interest is "Experiment 1" of the benchmark, that is the configuration with only one layer of copper reflector. In the next paragraphs, we offer a comprehensive description of each element of the experimental configuration.

The object of interest is the metallic plutonium sphere. This sphere, known as the BERP ball, was cast in 1980 at Los Alamos National Laboratory. It is an α-phase plutonium sphere with an external diameter of 7.5876 cm, and an estimated density of 19.6039 g.cm^{-3}. The sphere is surrounded by SS-304 stainless steel cladding with a thickness of 0.03048 cm. The BERP ball contains mainly ^{239}Pu and ^{240}Pu with some traces of other isotopes. The isotopic composition (in 1980 at the time of the experiment) is given in Table A.1.

TABLE A.1: Isotopic composition of the BERP ball in 1980.

Element	^{238}Pu	^{239}Pu	^{240}Pu	^{241}Pu	^{242}Pu	^{241}Am	Impurities
wt. %	0.020	93.73	5.96	0.268	0.028	0.00056	0.004

A close-up photograph of the BERP ball is shown in Figure A.1. The copper reflector used in the experimental configuration also consists of two shells of Copper C101 which can be locked together. This alloy has a density 8.94 g.cm^{-3}. The chemical composition of the copper reflector is given in Table A.2. Only the main elements composing the alloy are shown here.

The detectors are two NOMAD detectors. They consist of polyethylene matrices in which 15 ^{3}He proportional counters are embedded. The tubes have a diameter of

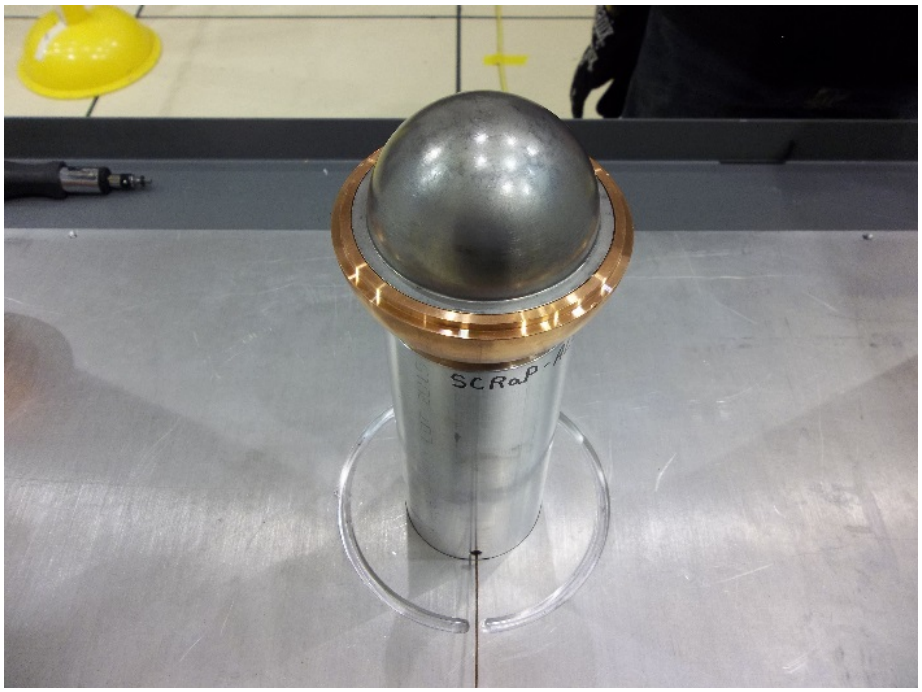

FIGURE A.1: Close-up view of the BERP ball with its surrounding copper reflector [Bes+20].

TABLE A.2: Simplified chemical composition of the copper reflector.

Element	Cu	Ag	S	Ni	Fe	O	Sb
wt. %	99.99	0.00090	0.00023	0.00020	0.00020	0.00020	0.00020

2.46 cm and a height of 38.1 cm. They are filled with a mixture of ^{3}He with CO_2 acting as a quench gas, with a pressure of 10.13 bars. The detectors have three rows of proportional counters, with respectively 7, 6, and 2 tubes at the front, middle and back rows. The polyethylene has a density of 1.27 g.cm^{-3}. The detectors are not modeled as such in our simplified MCNP model, but rather they are modeled by a single cylindrical ^{3}He counter surrounded by a polyethylene block, such that the active volume of the detector and moderator are preserved.

The source term in this experiment originates from spontaneous fissions in the plutonium sphere, as well as (α, n) reactions in the alloys. The spontaneous fission and (α, n) reaction rates were computed with the code SOURCES 4C. The total source intensity obtained was $S = 132582$ events per second, with a ratio of spontaneous fission with $x_s \simeq 0.989$.

The experimental configuration is the following: the BERP ball is located at the center of the experiment on a steel cart with a total height of 80 cm, on top of an aluminum support plate. The two NOMAD detectors are placed on either side of the sphere. The distance between the front of the detectors and the center of the sphere is 47.0 cm. A schematic view of the experiment is presented in Figure A.3, and a photograph can be seen in Figure A.2.

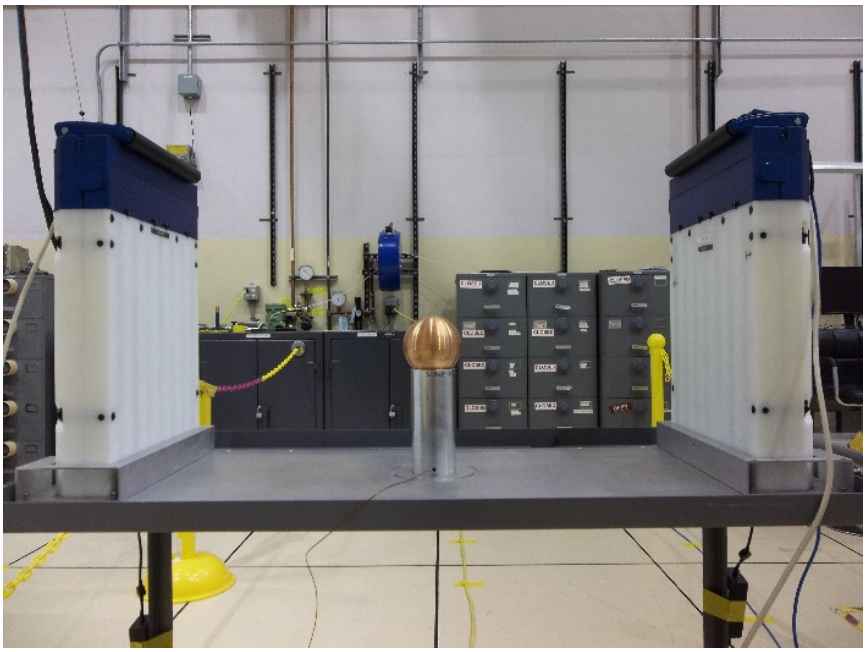

FIGURE A.2: Side view of the ICSBEP experiment [Bes+20].

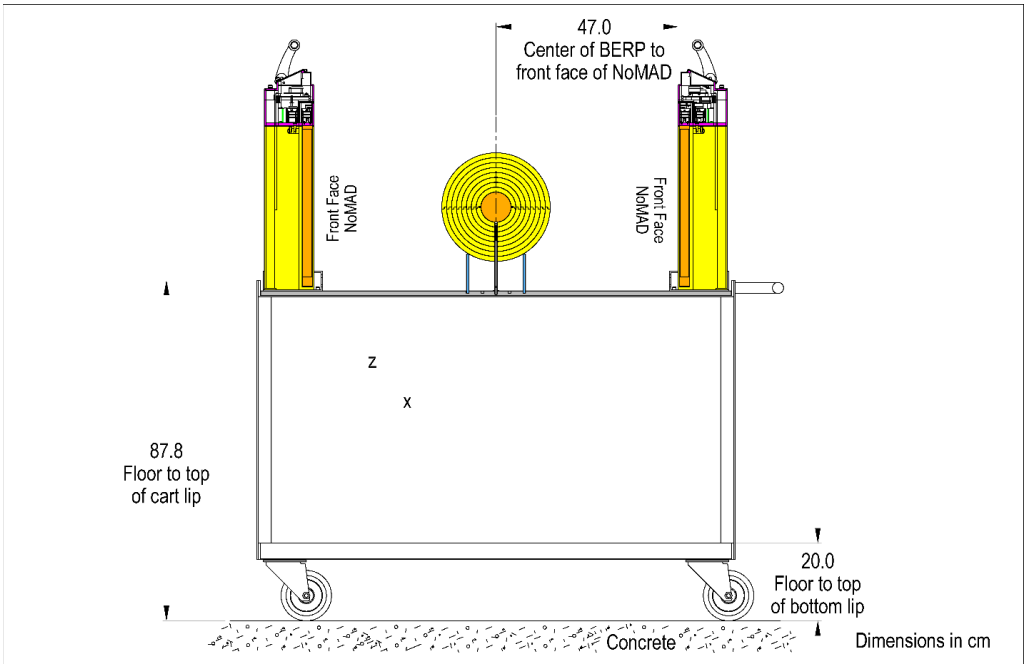

FIGURE A.3: Schematic view of the ICSBEP experiment [Bes+20].

A.2 SILENE reactor facility

The SILENE facility was an experimental reactor designed for pulsed experiments and subcritical multiplicity measurements. The reactor was designed to study criticality accidents occurring with fissile solutions. It was operated between 1974 and 2014.

The core is a cylindrical tank of 36 cm outer diameter filled with highly enriched (93 wt. % in ^{235}U) uranyl nitrate. A control rod is located at the center of the core to avoid the initial power excursion when the fissile solution is pumped into the core. The core is placed in a large room with thick concrete walls. A photograph of the SILENE

core with steel reflectors is shown in Figure A.4.

FIGURE A.4: Photograph of the SILENE core with steel reflectors.

The reactor had three main operation modes. It was mainly used for pulsed experiments in which the center control rod is rapidly removed from the core to create a power excursion up to 1000 MW. It was also possible to slowly remove the rod with an additional source to mimic the free evolution of a criticality accident. Finally, the reactor could also be operated in steady-state mode, with slow adjustments of the control rod.

The internal control rod is either a boron rod with a reactivity worth of 5.8$ or a cadmium rod with a reactivity worth of 4.1$. The rod is inserted in a canal of 7 cm of internal diameter at the center of the core.

An external neutron source is often added below the core. The source is a 100 mCi Am-Be source. The objective of the external neutron source is to limit the variance of the burst time due to stochastic fluctuations at low neutron populations.

Further details on the material compositions are given in Table A.3 for the fissile solution and in Table A.4 for the detector.

TABLE A.3: Isotopic composition of the uranyl nitrate fissile solution.

Isotope	^{1}H	^{14}N	^{16}O	^{235}U	^{238}U
wt. %	62.51	1.59	35.72	0.17	0.11

TABLE A.4: Isotopic composition of the neutron detection gas.

Isotope	^{10}B	^{11}B	^{19}F
wt. %	23.75	1.25	75.00

A schematic view of the SILENE core is presented in Figure A.5. The dotted region below the core represents the location of the Am-Be source.

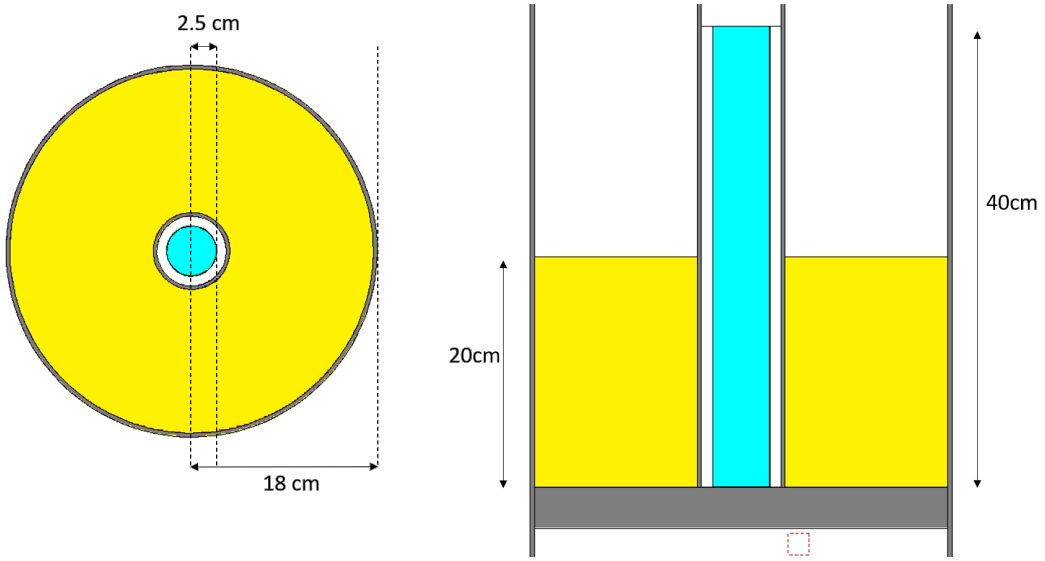

FIGURE A.5: Upper view (left) and side view (right) of the SILENE core as modeled in MCNP. The fissile solution is displayed in yellow, the steel in grey, and the detector in cyan.

Appendix B

Additional proofs

B.1 Proof of proposition 2.5.2

Proposition 2.5.2. *The likelihood $L(\mathbf{y}|x)$ is proportional to the following simplified likelihood:*

$$\overline{L(\mathbf{y}|x)} = \frac{1}{\sqrt{2\pi|N\mathbf{C}_s(x)+\mathbf{C}_{\text{obs}}|}} \times \exp\left(-\frac{1}{2}(\overline{y}-\overline{f_s(x)})^T\left(\mathbf{C}_s(x)+\frac{1}{N}\mathbf{C}_{\text{obs}}\right)^{-1}\left(\overline{y}-\overline{f_s(x)}\right)\right) \tag{2.78}$$

where $\overline{y} = \frac{1}{N}\sum_{k=1}^{N} y^{(k)}$

Proof. Let $x \in \mathcal{X}$. Consider first the total covariance matrix $\mathbf{\Sigma}(x) = \mathcal{U}_N \otimes \mathbf{C}_s(x) + \mathcal{I}_N \otimes \mathbf{C}_{\text{obs}}$. Its inverse is given by:

$$\mathbf{B}_{\text{tot}}(x) = \mathcal{I}_N \otimes \mathbf{C}_{\text{obs}}^{-1} - \mathcal{U}_N \otimes \left[(N\mathbf{C}_s(x)+\mathbf{C}_{\text{obs}})^{-1}\mathbf{C}_s(x)\mathbf{C}_{\text{obs}}^{-1}\right]. \tag{B.1}$$

It is easy to verify that $\mathbf{C}_{\text{tot}}(x)\mathbf{B}_{\text{tot}}(x) = \mathbf{B}_{\text{tot}}(x)\mathbf{C}_{\text{tot}}(x) = \mathcal{I}_{DN}$.
For concision sake, the inverse is written as:

$$\mathbf{\Sigma}(x)^{-1} = \mathcal{U}_N \otimes \mathbf{B}_1 + \mathcal{I}_N \otimes \mathbf{B}_2 \tag{B.2}$$

with $\mathbf{B}_1$ and $\mathbf{B}_2$ defined with equation (B.1) Similarly, its determinant is given by:

$$|\mathbf{C}_{\text{tot}}(x)| = |\mathbf{C}_{\text{obs}}|^{N-1}|N\mathbf{C}_s(x)+\mathbf{C}_{\text{obs}}|. \tag{B.3}$$

Consider now the eigenvalues $(\lambda_j)_{1\leq j\leq N}$ of $\mathcal{U}_N$ such that $\lambda_1 = N$ and $\lambda_j = 0$ for $j \geq 2$. We are interested in an orthonormal basis of eigenvectors $(\mathbf{e}_j)_{1\leq j\leq N}$ where $\mathbf{e}_j$ is associated to the eigenvalue λ_j for $1 \leq j \leq N$. In particular, if $\mathbf{u} = (1,....,1)^T \in \mathbb{R}^N$, then $\mathcal{U}_N\mathbf{u} = N\mathbf{u}$ and we have $\mathbf{e}_1 = \frac{1}{\sqrt{N}}\mathbf{u}$.
Let us introduce $\mathbf{y}_d = (y_d^{(1)},...,y_d^{(N)})^T \in \mathbb{R}^N$. Writing it on the basis $(\mathbf{e}_j)_{1\leq j\leq N}$ leads to $\mathbf{y}_d = \sum_{j=1}^{N} c_{d,j}\mathbf{e}_j$. For $1 \leq j \leq D$, we denote by $c_j = (c_{1,j},...,c_{D,j})^T \in \mathbb{R}^D$. We can write in a more concise form:

$$\mathbf{y}_{\text{flat}} = \sum_{j=1}^{N} \mathbf{e}_j \otimes c_j \tag{B.4}$$

$$\overline{\mathbf{f_s}}(x)_{\text{flat}} = \sqrt{N}\mathbf{e}_1 \otimes \overline{f_s(x)}. \tag{B.5}$$

One can notice that $c_1 = (c_{1,1}, ..., c_{D,1})^T = \sqrt{N}\overline{y}$.
Finally, to conclude the proof we need to keep only the terms depending on x in the exponential, the rest is absorbed in the multiplicative constant. In the vector $\left(\mathbf{y}_{\text{flat}} - \overline{\mathbf{f_s}(x)}_{\text{flat}}\right)$ this means that we keep only the first component in the basis of the $(\mathbf{e}_j)_{1\leq j\leq N}$. Thus we have:

$$\begin{aligned}
L(\mathbf{y}|x) &\propto \exp\left(-\frac{N}{2}\left(\mathbf{e}_1 \otimes (\overline{y} - \overline{f_s(x)})\right)^T \boldsymbol{\Sigma}(x)^{-1}\left(\mathbf{e}_1 \otimes (\overline{y} - \overline{f_s(x)})\right)\right) \\
&= \exp\left(-\frac{N}{2}\left(\mathbf{e}_1^T \otimes (\overline{y} - \overline{f_s(x)})^T\right)\left(\mathbf{e}_1 \otimes (\overline{y} - \overline{f_s(x)}) + N\mathbf{e}_1 \otimes \mathbf{B}_1(\overline{y} - \overline{f_s(x)})\right)\right) \\
&= \exp\left(-\frac{N}{2}(\overline{y} - \overline{f_s(x)})^T\left(\mathbf{C}_{\text{obs}}^{-1} - (\mathbf{C}_{\text{obs}} + N\mathbf{C}_s(x))^{-1}N\mathbf{C}_s(x)\mathbf{C}_{\text{obs}}^{-1}\right)(\overline{y} - \overline{f_s(x)})\right) \\
&= \exp\left(-\frac{N}{2}(\overline{y} - \overline{f_s(x)})^T(\mathbf{C}_{\text{obs}} + N\mathbf{C}_s(x))^{-1}(\overline{y} - \overline{f_s(x)})\right) \\
&= \exp\left(-\frac{1}{2}(\overline{y} - \overline{f_s(x)})^T\left(\frac{1}{N}\mathbf{C}_{\text{obs}} + \mathbf{C}_s(x)\right)^{-1}(\overline{y} - \overline{f_s(x)})\right). \qquad \text{(B.6)}
\end{aligned}$$

which concludes the proof. □

B.2 IP-SUR strategy for MOGP

B.2.1 Derivation of the SUR criteria

Proposition 4.2.4. *The SUR criterion for the metric $\mathcal{D}$ defined in (4.68) is given by:*

$$X_{n+1} \in \text{argmin}_{X\in\mathcal{X}} F_n(X) \qquad (4.71)$$

where $F_n(X) = \mathbb{E}_{n,X}\left[\mathcal{D}(P^f_{n+1})\right]$ for all $X \in \mathcal{X}$. This SUR criterion is equivalent to the criterion:

$$X_{n+1} \in \text{argmin}_{X\in\mathcal{X}} \widehat{F_n(X)} \qquad (4.72)$$

where the quantity $\widehat{F_n(X)}$ is given for $X \in \mathcal{X}$ by:

$$\widehat{F_n(X)} = \int_{\mathcal{X}} |k_{n+1}(x|X)|p_n(x|\mathbf{y})dx. \qquad (4.73)$$

Proof. The proof is very similar to the scalar case. The quantity of interest for a SUR design applied to the functional $\mathcal{D}$ is $F_n(X) = \mathbb{E}_Y\left[D_{n+1}(X,Y)\right]$ where $D_{n+1}(X,Y) = \mathcal{D}(P^f_{n+1})$. Using the result $\mathbb{E}_Y\left[L_{n+1}(\mathbf{y}|x,X,Y)\right] = L_n(\mathbf{y}|x)$ from the proof of proposition 4.2.1 we obtain:

$$\begin{aligned}
F_n(X) &= \int_{\mathbb{R}^D}\int_{\mathcal{X}} L_{n+1}(\mathbf{y}|x,X,Y)p(x)|k_{n+1}(x|X)|\mathcal{N}\left(Y|m_n(X),k_n(X)\right)dYdx \\
&= \int_{\mathcal{X}} L_n(\mathbf{y}|x)p(x)|k_{n+1}(x|X)|dx. \qquad \text{(B.7)}
\end{aligned}$$

Adding multiplicative constants does not change the SUR criterion, and thus one can rewrite the strategy using $\widehat{F_n(X)} = \frac{F_n(X)}{C_n} = \int_{\mathcal{X}} |k_{n+1}(x|X)|p_n(x|\mathbf{y})dx$ which concludes the proof. □

B.2.2 Proof of the supermartingale property 4.2.3 for $\mathcal{D}$

Lemma 4.2.3. *The functional $\mathcal{D}\colon \mathbb{M} \to \mathbb{R}_+$ defined by (4.68) has the supermartingale property. In other words, for any sequential design $(X_n)_{n\in\mathbb{N}}$, there exists $n_0 \in \mathbb{N}$ such that for all $n \geq n_0$, and for all $X \in \mathcal{X}$*

$$\mathbb{E}_{n,X}\left[\mathcal{D}(P^f_{n+1})\right] \leq \mathcal{D}(P^f_n). \tag{4.75}$$

Proof. Let $X \in \mathcal{X}$. We would like to show that $F_n(X) \leq D_n$ for all $n \geq 1$, where $F_n(X) = \mathbb{E}_{n,X}\left[\mathcal{D}(P^f_n)\right] = \mathbb{E}_Y\left[D_{n+1}(X,Y)\right]$. Let $X \in \mathcal{X}$ and $n \geq 1$. Using (4.73), we show that:

$$D_n - F_n(X) = \int_{\mathcal{X}} \left(|k_n(x)| - |k_{n+1}(x|X)|\right) L_n(\mathbf{y}|x)p(x)dx \tag{B.8}$$

The supermartingale property derives naturally from this expression since we have $|k_{n+1}(x|X)| \leq |k_n(x)|$ for all $x \in \mathcal{X}$. □

B.3 TIP-SUR strategy for MOGP

Proposition 4.2.6. *Let $\beta \in (0,1)$ and $(X_n)_{n\geq 1}$ be a SUR sequential design for the criterion (4.80). Then, the sequence $(D_{n,\beta})_{n\geq 0}$ where $D_{n,\beta} = \mathcal{D}_\beta(P^f_n)$ converges almost surely to zero:*

$$D_{n,\beta} \xrightarrow[n\to+\infty]{a.s.} 0.$$

The same convergence holds for the sequence $(H_{n,\beta})_{n\geq 0}$ with $H_{n,\beta} = \mathcal{H}_\beta(P^f_n)$:

$$H_{n,\beta} \xrightarrow[n\to+\infty]{a.s.} 0.$$

Proof. To prove the almost-sure convergence of $\mathcal{H}_\beta$, we follow an analogous procedure to the one used in the IP-SUR case. Let $\beta \in (0,1)$. We introduce the functional $\mathcal{D}_\beta$ defined for $\nu \in \mathbb{M}$ by:

$$\mathcal{D}_\beta(\nu) = \int_{\mathcal{X}} |k_\nu(x)| \left(L_\nu(\mathbf{y}|x)\right)^\beta p(x)dx = C_{\nu,\beta}\mathcal{H}_\beta(\nu). \tag{B.9}$$

We then show the convergence for the functional $\mathcal{D}_\beta$ and use an equivalent of the lemma 4.2.2 to prove the convergence for $\mathcal{H}_\beta$.

Introducing $C_{n,\beta} = \int_{\mathcal{X}} \left(L_n(\mathbf{y}|x)\right)^\beta p(x)dx$, the convergence of the sequence $(C_{n,\beta})_{n\in\mathbb{N}}$ is guaranteed with the same argument as for the IP-SUR case, and thus $C_{n,\beta} \to C_{\infty,\beta} > 0$.

To verify the assumptions of theorem 4.1.1, the only difficult task is to obtain the supermartingale property for $\mathcal{D}_\beta$. The other assumptions are verified with the same reasoning as with the IP-SUR case. In what follows, we thus focus on the proof of the supermartingale property for the functional $\mathcal{D}_\beta$, which is the last requirement to complete this proof.

Let $X \in \mathcal{X}$. Our goal is to show that $F_{n,\beta}(X) - D_{n,\beta} \leq 0$ where $F_{n,\beta}(X)$ was introduced in (4.81) and $D_{n,\beta} = \mathcal{D}_\beta(P^f_n) = \int_{\mathcal{X}} |k_n(x)| \left(L_n(\mathbf{y}|x)\right)^\beta p(x)dx$.

Using (4.81) we obtain:

$$F_{n,\beta}(X) = \int_{\mathcal{X}} |k_{n+1}(x|X)| f_\beta(x,X) I_\beta(x,X) \left(L_n(\mathbf{y}|x)\right)^\beta p(x) dx. \tag{B.10}$$

We begin by showing that $|A_\beta|^{-1/2}|k_n(X)|\frac{|\Sigma_n(x)^{\beta/2}|}{|\Sigma_{n+1}(x|X)|^{\beta/2}} \leq 1$. Using (4.88), one can write:

$$|A_\beta|^{-1/2}|k_n(X)|\frac{|\Sigma_n(x)^{\beta/2}|}{|\Sigma_{n+1}(x|X)|^{\beta/2}} = \frac{|\mathbf{M}_{n+1}|^{\frac{1-\beta}{2}}|\mathbf{M}_n|^{\beta/2}}{|\mathbf{M}_{n,\beta}|^{1/2}} \tag{B.11}$$

where we define $\mathbf{M}_{n,\beta} = \mathbf{M}_n - N(1-\beta)\lambda_n(x,X)$. One can notice that $\mathbf{M}_{n,\beta} = \beta\mathbf{M}_n + (1-\beta)\mathbf{M}_{n+1}$.
Using the log-concavity, we have:

$$|\mathbf{M}_{n,\beta}| = |\beta\mathbf{M}_n + (1-\beta)\mathbf{M}_{n+1}| \geq |\mathbf{M}_n|^\beta |\mathbf{M}_{n+1}|^{1-\beta}. \tag{B.12}$$

Taking the square root we obtain:

$$|A|^{-1/2}|k_n(X)|\frac{|\Sigma_n(x)|^{\beta/2}}{|\Sigma_{n+1}(x|X)|^{\beta/2}} \leq 1. \tag{B.13}$$

Then we want to show that $B_\beta^T A_\beta^{-1} B_\beta \leq \|\mathbf{y} - \mathbf{m_n}(x)\|^2_{\left(\Sigma_{n+1}^{-1} - \Sigma_n^{-1}\right)^{-1}}$. Using the inverse of A_β in (4.89), we write:

$$\begin{aligned}
B_\beta^T A_\beta^{-1} B_\beta &= (\beta N)^2 \left(\overline{y} - m_n(x)\right)^T \mathbf{M}_{n+1}^{-1} \\
&\times \left(\lambda_n(x,X) - N\beta\lambda_n(x,X)\mathbf{M}_{n,\beta}^{-1}\lambda_n(x,X)\right)\mathbf{M}_{n+1}^{-1}\left(\overline{y} - m_n(x)\right) \\
&= (\beta N)^2 \left(\overline{y} - m_n(x)\right)^T \mathbf{M}_{n+1}^{-1}\lambda_n(x,X)\mathbf{M}_{n,\beta}^{-1} \\
&\times \left(\mathbf{M}_{n,\beta} - \beta N\lambda_n(x,X)\right)\mathbf{M}_{n+1}^{-1}\left(\overline{y} - m_n(x)\right) \\
&= (\beta N)^2 \left(\overline{y} - m_n(x)\right)^T \mathbf{M}_{n+1}^{-1}\lambda_n(x,X)\mathbf{M}_{n,\beta}^{-1}\left(\overline{y} - m_n(x)\right).
\end{aligned} \tag{B.14}$$

We can then see that:

$$\begin{aligned}
\beta\mathbf{M}_{n,\beta}^{-1} &= \beta(\beta\mathbf{M}_n + (1-\beta)\mathbf{M}_{n+1})^{-1} \\
&= \mathbf{M}_n^{-1}\left(\mathcal{I}_D + \frac{(1-\beta)}{\beta}\mathbf{M}_n^{-1}\mathbf{M}_{n+1}\right)^{-1}
\end{aligned} \tag{B.15}$$

which leads to:

$$\begin{aligned}
\|\mathbf{y} - \mathbf{m_n}(x)\|^2_{\left(\Sigma_{n+1}^{-1} - \Sigma_n^{-1}\right)^{-1}} - B_\beta^T A_\beta^{-1} B_\beta &= N^2 \left(\overline{y} - m_n(x)\right)^T \mathbf{M}_{n+1}^{-1} \\
&\times \lambda_n(x,X)\left(\mathbf{M}_n^{-1} - \beta\mathbf{M}_{n,\beta}^{-1}\right)\left(\overline{y} - m_n(x)\right).
\end{aligned} \tag{B.16}$$

Let us now define $\Sigma_{n,\beta}(x,X) = \mathcal{I}_N \otimes \mathbf{C}_{\text{obs}}^{-1} + \mathcal{U}_N \otimes (k_n(x) - \beta\lambda_n(x,X))$. One can easily show that:

$$\Sigma_{n,\beta}(x,X)^{-1} = \mathcal{I}_N \otimes \mathbf{C}_{\text{obs}}^{-1} - \mathcal{U}_N \otimes \left(\mathbf{M}_{n,\beta}^{-1}(k_n(x) - \beta\lambda_n)\mathbf{C}_{\text{obs}}^{-1}\right) \tag{B.17}$$

$$\Sigma_{n+1}(x|X)^{-1} - \Sigma_{n,\beta}(x,X) = (1-\beta)\mathcal{U}_N \otimes \left(\mathbf{M}_{n+1}^{-1}\lambda_n\mathbf{M}_{n,\beta}^{-1}\right). \tag{B.18}$$

One can thus write (B.16) as:

$$\begin{aligned}\|\mathbf{y}-\mathbf{m_n}(x)\|^2_{\left(\Sigma_{n+1}^{-1}-\Sigma_n^{-1}\right)^{-1}} = B_\beta^T A_\beta^{-1} B_\beta + \beta^2 \left(\mathbf{y}-\mathbf{m_n}(x)\right)^T \\ \left(\left(\Sigma_{n+1}^{-1}-\Sigma_n^{-1}\right) - \frac{\beta}{1-\beta}\left(\Sigma_{n+1}^{-1}-\Sigma_{n,\beta}^{-1}\right)\right) \\ \left(\mathbf{y}-\mathbf{m_n}(x)\right). \end{aligned} \tag{B.19}$$

We conclude this proof by showing that the matrix $\mathbf{N}_\beta = \left(\Sigma_{n+1}^{-1}-\Sigma_n^{-1}\right) - \frac{\beta}{1-\beta}\left(\Sigma_{n+1}^{-1}-\Sigma_{n,\beta}^{-1}\right)$ is symmetric positive definite. We know it is symmetric since it is a sum of symmetric matrices. We define the partial ordering on the symmetric matrices as such. If P and Q are square matrices of the same size, we write $P \succcurlyeq Q$ is $P-Q$ is positive semi-definite. If we further have P and Q invertible we have $Q^{-1} \succcurlyeq P^{-1}$. In our case, we know that $\Sigma_n \succcurlyeq \Sigma_{n,\beta} \succcurlyeq \Sigma_{n+1}$ and thus $\Sigma_{n+1}^{-1} \succcurlyeq \Sigma_{n,\beta}^{-1} \succcurlyeq \Sigma_n^{-1}$. We need to distinguish between two cases. If $\beta \leq \frac{1}{2}$ one can write:

$$\begin{aligned}\mathbf{N}_\beta &= \left(1-\frac{\beta}{1-\beta}\right)\Sigma_{n+1}^{-1} + \frac{\beta}{1-\beta}\Sigma_{n,\beta}^{-1} - \Sigma_n^{-1} \\ &\succcurlyeq \left(1-\frac{\beta}{1-\beta}+\frac{\beta}{1-\beta}\right)\Sigma_{n,\beta}^{-1} - \Sigma_n^{-1} \\ &= \Sigma_{n,\beta}^{-1} - \Sigma_n^{-1} \succcurlyeq 0. \end{aligned} \tag{B.20}$$

On the other hand, if $\beta > \frac{1}{2}$ then $\frac{\beta}{1-\beta} > 1$, and the previous inequalities do not hold. However, one can write:

$$\begin{aligned}\mathbf{N}_\beta &= \left(1-\frac{\beta}{1-\beta}\right)\Sigma_{n+1}^{-1} + \frac{\beta}{1-\beta}\Sigma_{n,\beta}^{-1} - \Sigma_n^{-1} \\ &\succcurlyeq \left(1-\frac{\beta}{1-\beta}\right)\Sigma_{n+1}^{-1} - \left(1-\frac{\beta}{1-\beta}\right)\Sigma_n^{-1} \\ &= \left(1-\frac{\beta}{1-\beta}\right)\left(\Sigma_{n+1}^{-1}-\Sigma_n^{-1}\right) \succcurlyeq 0. \end{aligned} \tag{B.21}$$

We can finally conclude that $\mathbf{N}_\beta$ is positive semi-definite, which leads to:

$$\exp\left(-\frac{1}{2}\left(\|\mathbf{y}-\mathbf{m_n}(x)\|^2_{\left(\Sigma_{n+1}^{-1}-\Sigma_n^{-1}\right)^{-1}} - B_\beta^T A_\beta^{-1} B_\beta\right)\right) \leq 1. \tag{B.22}$$

Injecting the inequalities (B.13) and (B.22) into the expression of $F_{n,\beta}(X)$ yields the supermartingale property for the functional $\mathcal{D}_\beta$ which is enough to conclude the proof. □

Appendix C

MCNP input files

C.1 BERP ball training instance

```
c ********************** BLOCK 1 : CELL CARD *********************************
1    105 -0.0000001   -21                                          imp:n=1
2    101 -15.87        21 -22                                      imp:n=1
3    107 -1.00         22 -23                                      imp:n=1
4    103 -0.00175     -31  35 -36                                  imp:n=1
5    104 -0.93        -32  33 -34   #(-31 35 -36)                  imp:n=1
8    105 -0.001205  (41 -42 43 -44 45 -46) #(-32 33 -34) #(-24) #(-51) imp:n=1
9    106 -2.30      #(41 -42  43 -44  45 -46) (11 -12  13 -14  15 -16) imp:n=1
1001 105 -0.001205 23 -24                                          imp:n=1
1002 105 -0.001205 -51                                             imp:n=1
10 0               #(11 -12  13 -14  15 -16)                       imp:n=0
c ********************** BLOCK 2 : SURFACE CARD ******************************
11 PX -50
12 PX 550
13 PY -50
14 PY 550
15 PZ -50
16 PZ 250
21 S 200 250 100 4.307
22 S 200 250 100 6.322
23 S 200 250 100 8.173
24 S 200 250 100 8.673
31 C/Z 300 250 18.12
32 C/Z 300 250 22.65
33 PZ 30.0
34 PZ 170.0
35 PZ 35.0
36 PZ 165.0
41 PX -37.69
42 PX 537.69
43 PY -37.69
44 PY 537.69
45 PZ -37.69
46 PZ 237.69
51 S 225 350 200 10.0
c ********************** BLOCK 3 : DATA CARD *********************************
```

```
SDEF par=D1 POS=FPAR D2 RAD=FPAR D3 ERG = 20
SI1  L   SF          N
SP1  W   0.67         0.32
DS2  L   200 250 100 200 250 100
DS3  S   D5          D6
SI5  4.307   6.322
SI6  4.307   6.322
MODE N
TOTNU NO
PHYS:N 100 99
MPHYS OFF
PTRAC FILE=ASC EVENT=TER FILTER=4,ICL TYPE=N WRITE=ALL MAX=10000000
NPS 1000000
RAND SEED=120090
f1:n 41 -42 43 -44 45 -46 T
c Detector flux tally
f4:n  4
fm4  -1. 103 -2
c Pu flux tally
f14:n  2
fm14  -1. 101 -6
c Abs flux tally
f24:n  3
fm24  -1. 107 -2
c Prox flux tally
f34:n 1001
c Air flux tally
f44:n 1002
*C1 90 0
E4  0.001 20
E14 0.001 20
E24 0.001 20
E34 0.001 20
E44 0.001 20
c ********************** BLOCK 4 : MATERIAL CARD *********************************
c --- WG Plutonium (4-7% Pu-240) - Porosity 20%
m101  94238.80c  0.000100
      94239.80c  0.936559
      94240.80c  0.059677
      94241.80c  0.001981
      94242.80c  0.000296
      95241.80c  0.001387
c --- He-3 detector (14 atm)
m103  2003.80c   1.000000
c --- Non-borated polyethylene
m104  1001.80c   0.666662
      6000.80c   0.333338
mt104 poly.20t
c --- Air
m105  6000.80c   0.000150
      7014.80c   0.784431
```

```
      8016.80c   0.210748
      18036.80c  0.000016
      18038.80c  0.000003
      18040.80c  0.004652
c ---Concrete reflector (2.30 g/cm3)
      m106  1001.80c   0.305330
      6000.80c   0.002880
      8016.80c   0.500407
      11023.80c  0.009212
      12024.80c  0.000573
      12025.80c  0.000072
      12026.80c  0.000079
      13027.80c  0.010298
      14028.80c  0.139261
      14029.80c  0.007099
      14030.80c  0.004682
      19039.80c  0.003337
      19041.80c  0.000241
      20040.80c  0.014467
      20042.80c  0.000096
      20043.80c  0.000020
      20044.80c  0.000311
      26054.80c  0.000094
      26056.80c  0.001473
      26057.80c  0.000034
      26058.80c  0.000004
mt106 fe56.22t
      al27.22t
c --- Borated polyethylene
m107  1001.80c   0.627759
      5010.80c   0.009244
      5011.80c   0.037445
      6000.80c   0.325552
mt107 poly.20t
PRINT 115 117 118
```

C.2 SILENE with gamma correlations with $h = 20$ cm

```
    c------------- CELLS ---------------
c Vide exterieur
c
100 0   #(401 -402  403 -404  405 -406) imp:n=0
c
c Cuve en acier inox
c
110 11 -7.9  (100 -110 -220) imp:n=1
120 11 -7.9  (130 -140 -230) imp:n=1
130 11 -7.9  (110 -130 -210 200) imp:n=1
140 11 -7.9  (70  -130 -230 220)  imp:n=1
142 11 -7.9  (70  -80  -220) imp:n=1
```

```
c
c Solution fissile
c
150 20 -1.161  (110 -120 -220 210) imp:n=1
c
c Air canal
c
160 31 -1.293E-3 (124 -130 -200) imp:n=1
162 31 -1.293E-3 (110 -124 -200 198) imp:n=1
c
c Air au dessus de la solution
c
170 31 -1.293E-3 (120 -130 -220 210)  imp:n=1
c
c Gainage cuivre du detecteur
c
180 41 -8.96 (110 -112 -198)  imp:n=1
190 41 -8.96 (122 -124 -198)  imp:n=1
200 41 -8.96 (112 -122 -198 196)  imp:n=1
c
c Gaz BF3
c
210 51 -6.2505E-4 (112 -122 -196) imp:n=1
c
c Source AmBe
c
220 31 -1.293E-3 (92 -94 -240)  imp:n=1
c
c Air sous la cuve (pour positionner AmBe)
c
230 31 -1.293E-3 (80 -100 -220)  imp:n=1
c
c Beton des murs
c
240 61 -2.30 #(301 -302 303 -304 305 -306) (401 -402 403 -404 405 -406) imp:n=1
c
c Air autour de Silene
c
250 31 -1.293E-3 (301 -302 303 -304 305 -306) #(-230 70 -140) imp:n=1
c

c ---------------SURACES -----------------
c
70  PZ -65.40
80  PZ -65.00
c 90  PZ -2.50
92  PZ -2.31
94  PZ -0.40
100 PZ  0.00
110 PZ  3.60
112 PZ  3.65
```

```
120 PZ  23.60 $ H = 200mm
122 PZ  43.65
124 PZ  43.70
130 PZ  100.60
140 PZ  103.60
c
196 CZ  2.45
198 CZ  2.50
200 CZ  3.45
210 CZ  3.80
220 CZ  18.00
230 CZ  18.4
240 C/Z 0. 5. 0.9
c
301 PX -48
302 PX  48
303 PY -48
304 PY  48
305 PZ  -100
306 PZ  200
c
401 PX -60
402 PX  60
403 PY -60
404 PY  60
405 PZ -110
406 PZ  210

c ----------MATERIALS ----------------------
c  Acier
m11  26054.70c   0.00352220
     26056.70c   0.05463520
     26057.70c   0.00134020
     26058.70c   0.00019700
      6000.70c   0.00011745
     24050.70c   0.00067720
     24052.70c   0.01281600
     24053.70c   0.00146500
     24054.70c   0.00040080
     28058.70c   0.00597000
     28060.70c   0.00230700
     28061.70c   0.00010570
     28062.70c   0.00032580
     28064.70c   0.00009690
     28064.70c   0.00167428
c
c  Solution fissile
c
m20   1001.70c  6.258E-2 $ H1
   7014.70c  1.569E-3 $ N14
   8016.70c  3.576E-2 $ 016
```

```
  92234.70c  1.060E-6 $ U234
  92235.70c  1.686E-4 $ U235
  92236.70c  4.350E-7 $ U236
  92238.70c  1.10E-5 $ U238
c
mt20 lwtr.01t
c
c  Air
c
m31  8016.70c  1.0805E-5
  8017.70c  4.3240E-9
  7014.70c  4.3092E-5
  7015.70c  1.5800E-7
c
c  Cuivre
c
m41  29063.70c  0.6917  $ Cu63
  29065.70c  0.3083  $ Cu65
c
c  BF3
c
m51  5010.70c  0.2375  $ B10
  5011.70c  0.0125  $ B11
  9019.70c  0.7500  $ F19
c
c ---Concrete reflector (2.30 g/cm3)
   m61  1001.80c   0.305330
  6000.80c   0.002880
         8016.80c   0.500407
         11023.80c  0.009212
         12024.80c  0.000573
         12025.80c  0.000072
         12026.80c  0.000079
         13027.80c  0.010298
  14028.80c  0.139261
  14029.80c  0.007099
  14030.80c  0.004682
  19039.80c  0.003337
  19041.80c  0.000241
  20040.80c  0.014467
  20042.80c  0.000096
  20043.80c  0.000020
  20044.80c  0.000311
  26054.80c  0.000094
  26056.80c  0.001473
  26057.80c  0.000034
  26058.80c  0.000004
  mt61  fe56.22t
         al27.22t
c
c ---------------------  CALCUL --------------------
```

```
c
SDEF CEL=150 PAR=D1 AXS=0 0 1 RAD=D2 EXT=D3 ERG=D4
SI1  L   SF          N
SP1  W   0.4        0.6
SI2 3.85 17.5
SP2 -21 2
SI3 3.65 33.55
SP4 -3 0.988 2.249
F4:N 170
E4 1E-7 20
RAND SEED=8739
MODE N P
TOTNU NO
PHYS:N 100 99
MPHYS OFF
PTRAC FILE=ASC EVENT=TER FILTER=240,ICL TYPE=P WRITE=ALL MAX=10000000
NPS 500000
```

Appendix D

On the BNN architecture

In this appendix, we detail some of the numerical results that led to the choice of network architecture. Throughout this chapter, we tackle only the homoscedastic model **LKJ-BNN**. The models were trained on the BERP dataset.

D.1 Depth and width of the network

We first look at the number of hidden layers and neurons per layer of our model. In every case, the network is fully connected with sigmoid or tanh activation in the hidden layers and a linear activation in the output layer. We test the various metrics for different architectures. The architectures are summarized by the vector $(7, K_n^{(1)}, ..., K_n^{(L)}, 3)$ describing the number of neurons in each layer. All the models are homoscedastic with a LKJ prior on the covariance. The posterior distribution of the weights is sampled with HMC-NUTS to produce 5×10^3 samples. The metrics obtained are shown in Tables D.1, D.2 and D.3 for the various output channels.

TABLE D.1: Performance metrics for the output R for various architectures with sigmoid (left) or tanh (right) activation - BERP dataset.

sigmoid	NMAE	NRMSE	Q^2	tanh	NMAE	NRMSE	Q^2
$(7, 10, 10, 3)$	0.0074	0.0121	0.9997	$(7, 10, 10, 3)$	0.0094	0.0148	0.9996
$(7, 30, 30, 3)$	0.0072	0.0119	0.9997	$(7, 30, 30, 3)$	0.0088	0.0141	0.9996
$(7, 50, 50, 3)$	**0.0071**	**0.0117**	**0.9998**	$(7, 50, 50, 3)$	0.0323	0.0559	0.9943
$(7, 100, 100, 3)$	0.0132	0.0244	0.9989	$(7, 100, 100, 3)$	0.0589	0.0979	0.9825
$(7, 10, 10, 10, 3)$	0.0126	0.0129	0.9995	$(7, 10, 10, 10, 3)$	**0.0083**	**0.0126**	**0.9997**
$(7, 50, 50, 50, 3)$	0.1534	0.2214	0.9281	$(7, 50, 50, 50, 3)$	0.0880	0.1497	0.9592

TABLE D.2: Performance metrics for the output Y_∞ for various architectures with sigmoid (left) or tanh (right) activation - BERP dataset.

sigmoid	NMAE	NRMSE	Q^2	tanh	NMAE	NRMSE	Q^2
$(7, 10, 10, 3)$	0.0211	0.0446	0.9982	$(7, 10, 10, 3)$	**0.0207**	**0.0424**	**0.9983**
$(7, 30, 30, 3)$	**0.0204**	**0.0398**	**0.9986**	$(7, 30, 30, 3)$	0.0236	0.0504	0.9977
$(7, 50, 50, 3)$	0.0219	0.0440	0.9983	$(7, 50, 50, 3)$	0.0556	0.1176	0.9875
$(7, 100, 100, 3)$	0.0279	0.0609	0.9967	$(7, 100, 100, 3)$	0.0862	0.1711	0.9737
$(7, 10, 10, 10, 3)$	0.0337	0.0446	0.9971	$(7, 10, 10, 10, 3)$	0.0246	0.0593	0.9968
$(7, 50, 50, 50, 3)$	0.1646	0.2362	0.9181	$(7, 50, 50, 50, 3)$	0.1392	0.3038	0.9172

These results justify using sigmoid activation functions which tend to provide better predictions. Moreover, the best compromise between network complexity and predictive power appears to be the two hidden layers configuration with $K_n = 50$. This

TABLE D.3: Performance metrics for the output X_∞ for various architectures with sigmoid (left) or tanh (right) activation - BERP dataset.

sigmoid	NMAE	NRMSE	Q^2
$(7, 10, 10, 3)$	0.0828	0.3928	0.9783
$(7, 30, 30, 3)$	0.0752	0.3513	0.9826
$(7, 50, 50, 3)$	**0.0679**	**0.3089**	**0.9866**
$(7, 100, 100, 3)$	0.0916	0.3692	0.9808
$(7, 10, 10, 10, 3)$	0.0651	0.1074	0.9951
$(7, 50, 50, 50, 3)$	0.3319	0.6335	0.8288

tanh	NMAE	NRMSE	Q^2
$(7, 10, 10, 3)$	0.0903	0.4254	0.9745
$(7, 30, 30, 3)$	**0.0763**	**0.3472**	**0.9830**
$(7, 50, 50, 3)$	0.1405	0.5423	0.9586
$(7, 100, 100, 3)$	0.2180	0.7373	0.9234
$(7, 10, 10, 10, 3)$	0.1080	0.5420	0.9586
$(7, 50, 50, 50, 3)$	0.3859	1.3210	0.7543

study was only conducted for the case of the BNN with LKJ prior for the covariance, yet we assume that the conclusion extends to the other models.

D.2 Prior variances selection

In this section we consider zero-mean Gaussian iid priors $p(w) \sim \mathcal{N}\left(0, \sigma_w^2\right)$ for the weights and $p(b) \sim \mathcal{N}\left(0, \sigma_b^2\right)$ for the biases. We wish to select the best possible values for σ_b and σ_w. This is done by a grid-search approach with $\sigma_b, \sigma_w \in \{0.3, 1.0, 3.0, 5.0, 10.0\}$. The architecture is fixed to two hidden layers with sigmoid activation functions and $K_n = 50$ per layer. We evaluate the NMAE on the test set by averaging the predictive means obtained by various network parameters samples $\boldsymbol{\phi}$. The results are highlighted in Tables D.4, D.5 and D.6.

TABLE D.4: NMAE for the output R with various values of σ_w and σ_b with a fixed architecture and zero-mean iid Gaussian priors - BERP dataset.

σ_b \ σ_w	0.3	1.0	3.0	5.0	10.0
0.3	0.0075	0.0075	0.0081	0.0078	0.0079
1.0	0.0074	0.0075	0.0074	0.0075	0.0076
3.0	0.0075	0.0077	0.0073	0.0075	0.0078
5.0	0.0072	0.0074	**0.0071**	0.0072	0.0077
10.0	0.0075	0.0075	0.0077	0.0076	0.0076

TABLE D.5: NMAE for the output Y_∞ with various values of σ_w and σ_b with a fixed architecture and zero-mean iid Gaussian priors - BERP dataset.

σ_b \ σ_w	0.3	1.0	3.0	5.0	10.0
0.3	0.020	0.022	0.022	0.022	0.023
1.0	0.021	0.021	0.021	0.020	0.021
3.0	0.020	0.021	0.021	0.021	0.021
5.0	0.020	0.022	0.021	0.020	**0.019**
10.0	0.020	0.021	0.021	0.020	0.022

The prior variances do not have a significant impact on the predictive errors. The best metrics obtained for each output are highlighted in bold in the tables. From this analysis, the choice of $\sigma_w = \sigma_b = 5.0$ appears appropriate.

TABLE D.6: NMAE for the output X_∞ with various values of σ_w and σ_b with a fixed architecture and zero-mean iid Gaussian priors - BERP dataset.

σ_b \ σ_w	0.3	1.0	3.0	5.0	10.0
0.3	0.078	0.081	0.089	0.085	0.082
1.0	0.085	0.079	0.081	0.073	0.065
3.0	0.077	0.073	0.079	0.081	0.075
5.0	0.080	0.086	0.081	**0.069**	0.075
10.0	0.078	0.074	0.073	0.074	0.085

D.3 Prior means selection

In this section, we focus on the selection of the prior means for the weights and biases. A naive approach is to select zero-mean priors, as was done in the previous section for the architecture selection. Another possibility studied in this section is to train a standard ANN and use the point values obtained for the weights and biases as prior means. This methodology is conducted, the ANNs are trained with the Adam algorithm, and the results are presented in Tables D.7, D.8 and D.9 for each output channel.

TABLE D.7: NMAE for the output R for various architectures of the LKJ-BNN network with NN initialization or zero-mean initialization, and $\sigma_w = \sigma_b = 5.0$, $\eta = 0.8$ - BERP dataset.

R	$\mu = 0$	NN initialization
$(7, 10, 10, 3)$	0.0075	0.0076
$(7, 30, 30, 3)$	0.0072	0.0071
$(7, 50, 50, 3)$	0.0074	0.0077
$(7, 100, 100, 3)$	0.0132	0.0142
$(7, 10, 10, 10, 3)$	0.0126	0.0169
$(7, 50, 50, 50, 3)$	0.1535	0.0075

TABLE D.8: NMAE for the output Y_∞ for various architectures of the LKJ-BNN network with NN initialization or zero-mean initialization, and $\sigma_w = \sigma_b = 5.0$, $\eta = 0.8$ - BERP dataset.

R	$\mu = 0$	NN initialization
$(7, 10, 10, 3)$	0.0211	0.0216
$(7, 30, 30, 3)$	0.0205	0.0209
$(7, 50, 50, 3)$	0.0220	0.0222
$(7, 100, 100, 3)$	0.0272	0.0297
$(7, 10, 10, 10, 3)$	0.0339	0.0409
$(7, 50, 50, 50, 3)$	0.1646	0.0198

From this study, one can conclude that initializing the prior means with a point network is beneficial only for larger architectures, which were not ultimately chosen in this work. For this reason, we kept zero-mean priors throughout this thesis.

TABLE D.9: NMAE for the output X_∞ for various architectures of the LKJ-BNN network with NN initialization or zero-mean initialization, and $\sigma_w = \sigma_b = 5.0$, $\eta = 0.8$ - BERP dataset.

X_∞	$\mu = 0$	NN initialization
$(7, 10, 10, 3)$	0.0859	0.0902
$(7, 30, 30, 3)$	0.0771	0.0749
$(7, 50, 50, 3)$	0.0737	0.0801
$(7, 100, 100, 3)$	0.0916	0.0954
$(7, 10, 10, 10, 3)$	0.0651	0.1287
$(7, 50, 50, 50, 3)$	0.3319	0.0712

Appendix E

Additional numerical results

In this appendix, we present some additional numerical results which are not directly included in the main text.

E.1 SILENE dataset

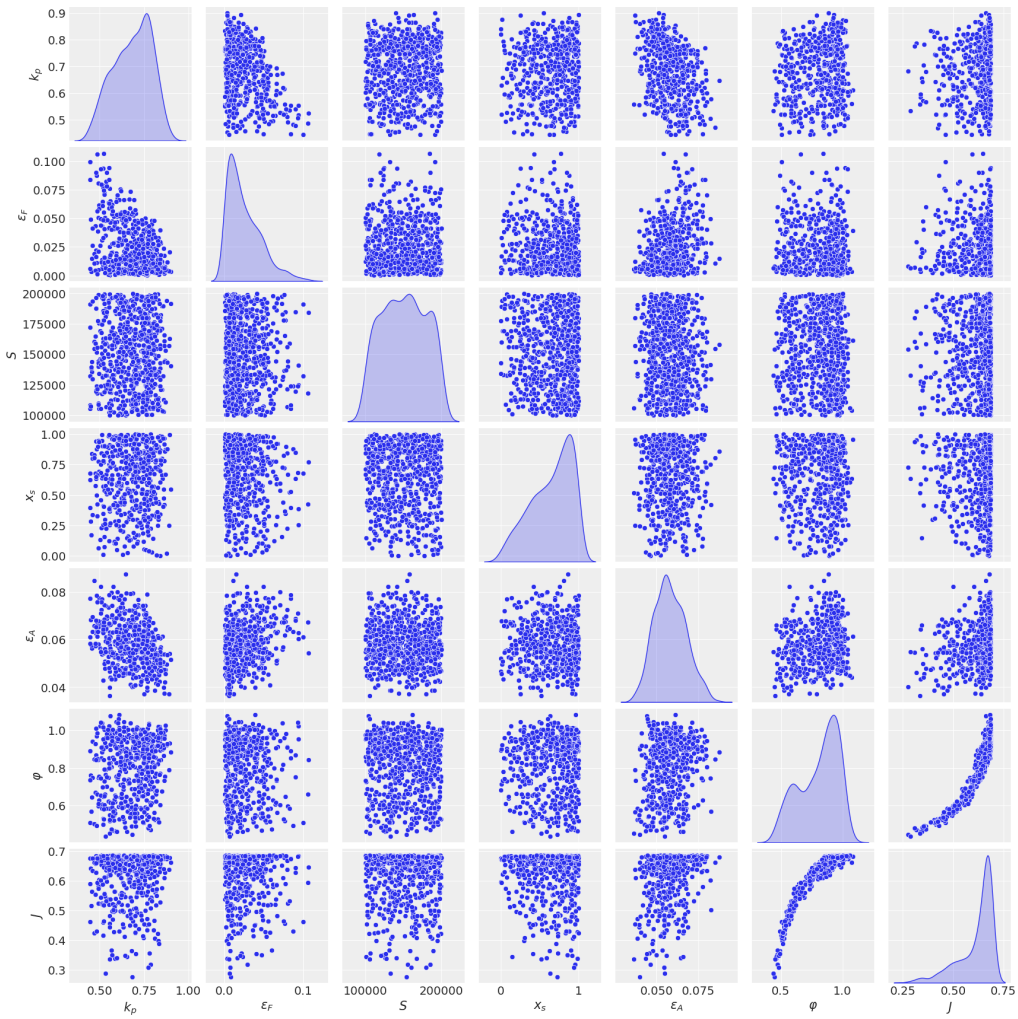

FIGURE E.1: Marginal densities and correlation plots for the inputs - SILENE dataset.

Figures E.1 and E.2 exhibit the correlations and marginal distributions of the inputs and outputs of the SILENE dataset.

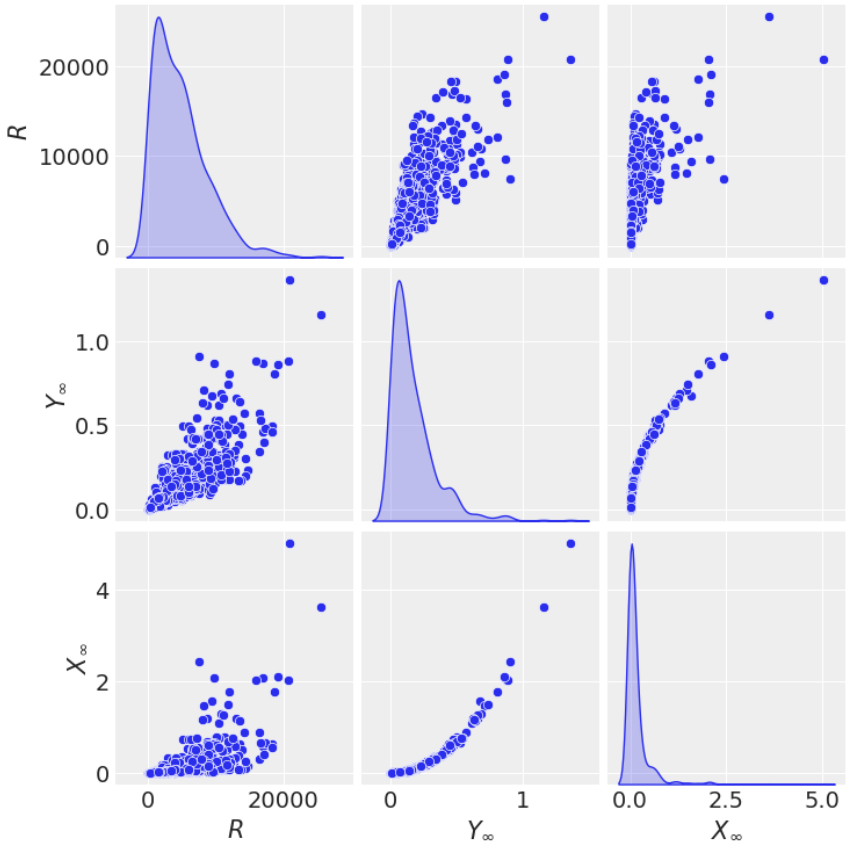

FIGURE E.2: Marginal densities and correlation plots for the outputs - SILENE dataset.

E.2 Model performance

In this section, the performance metrics for the surrogate models trained on the SILENE dataset are provided. The conclusions are similar to the BERP dataset. The multi-fidelity approach improves the surrogate models. The preferred models are the ones with $\rho = 1$.

TABLE E.1: Performance of the multi-fidelity GP with $\rho = 1$ - SILENE dataset.

BL-GP-LMC2	NMAE	NRMSE	Q^2
R	**0.008**	**0.011**	**0.9997**
Y_∞	**0.022**	0.043	0.997
X_∞	0.072	0.201	0.986
BL-GP-LMC3	NMAE	NRMSE	Q^2
R	**0.008**	**0.011**	**0.9997**
Y_∞	0.027	**0.038**	**0.998**
X_∞	**0.051**	**0.153**	**0.992**

TABLE E.2: Performance of the multi-fidelity GP with $\rho \neq 1$ - SILENE dataset.

MF-GP-LMC2	NMAE	NRMSE	Q^2
R	**0.023**	**0.034**	**0.9970**
Y_∞	0.031	0.055	**0.995**
X_∞	0.105	0.278	0.974
MF-GP-LMC3	NMAE	NRMSE	Q^2
R	**0.023**	**0.034**	**0.9970**
Y_∞	**0.021**	**0.040**	**0.997**
X_∞	**0.087**	**0.226**	**0.983**

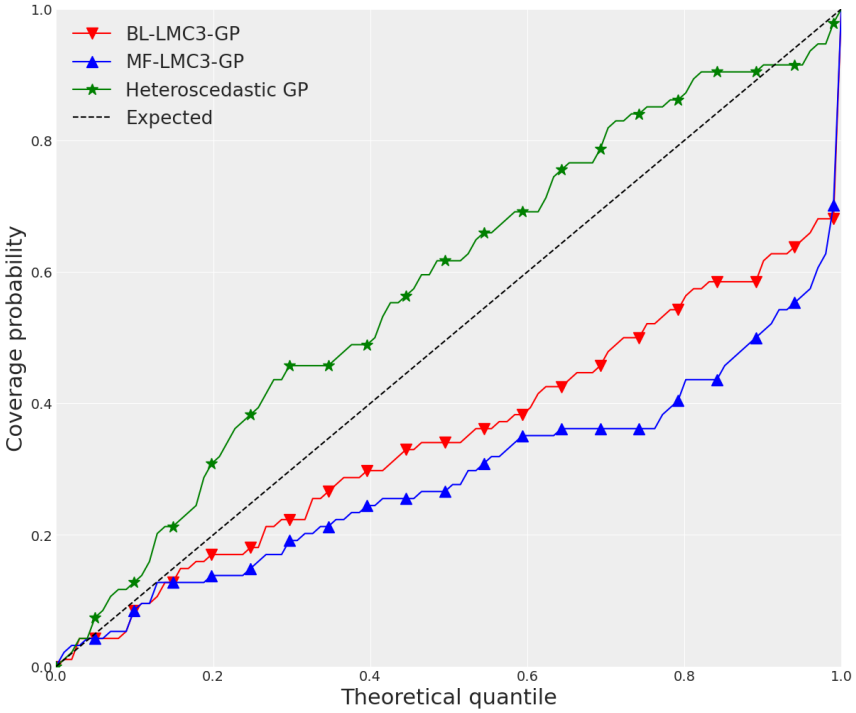

FIGURE E.3: Coverage probabilities for the multi-fidelity and heteroscedastic GP surrogate models - Silene dataset.

E.3 BNN surrogates performance for the SILENE dataset

We highlight the performance of the BNN surrogate models trained on the SILENE dataset. The performance metrics for the mean predictions are displayed in Table E.3 and the coverage probabilities are shown in Figure E.4. The heteroscedastic model appears to be the best-performing, although it tends to underestimate slightly the uncertainties. The coverage probabilities evaluated on the SILENE dataset are more difficult to exploit because the test set is quite small. The discrepancies with the theoretical quantiles may be linked either to the surrogate models or simply to a lack of data. The evaluation of the models' performance may not be as reliable as for the BERP dataset.

TABLE E.3: Performance of the BNN surrogates - SILENE dataset.

HMC - LKJ prior	NMAE	NRMSE	Q^2
R	0.0104	0.0161	0.9996
Y_∞	0.0594	0.08000	0.9906
X_∞	0.1343	0.2521	0.9729
HBNN	NMAE	NRMSE	Q^2
R	**0.0085**	**0.0143**	**0.9996**
Y_∞	**0.0206**	**0.0392**	**0.9986**
X_∞	0.0645	0.2169	0.9934
BL-HBNN	NMAE	NRMSE	Q^2
R	0.0344	0.0459	0.9969
Y_∞	0.0348	0.0443	0.9971
X_∞	**0.0593**	**0.0882**	**0.9967**
GP-HBNN	NMAE	NRMSE	Q^2
R	0.0103	0.0155	0.9997
Y_∞	0.0407	0.0551	0.9955
X_∞	0.0947	0.1825	0.9858

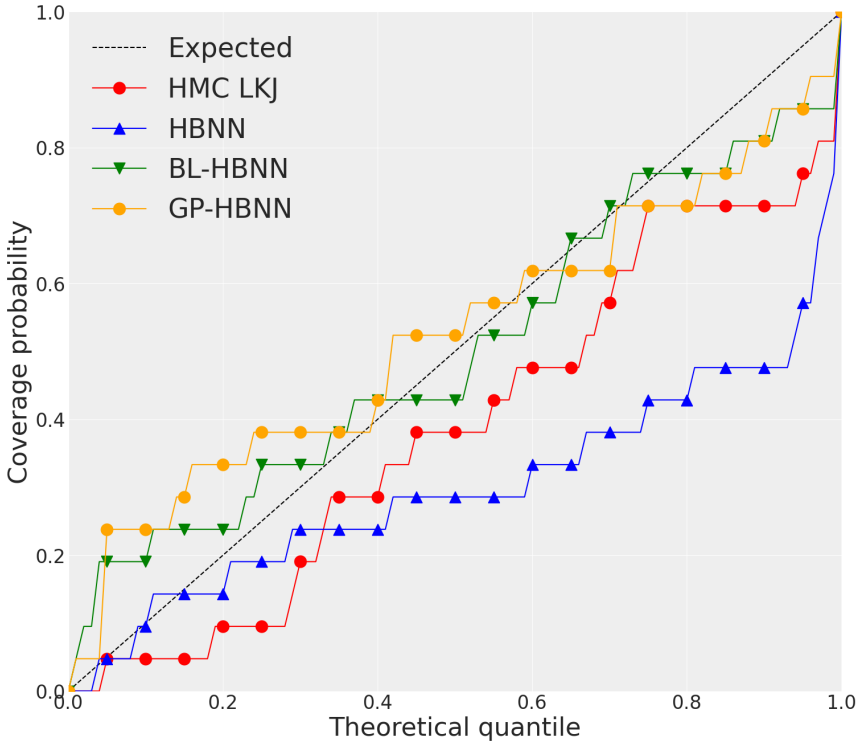

FIGURE E.4: Averaged coverage probabilities for the multi-fidelity and heteroscedastic BNN models - SILENE dataset.

E.4 Inverse problem observations

Table E.4 summarizes the observational data used for the Bayesian resolution of the inverse problems in the various test cases. We provide only the observational mean $\overline{y} = \frac{1}{N}\sum_{k=1}^{N} y^{(k)}$ and the effective covariance $\mathbf{C}_{\text{eff}} = \frac{1}{N}\mathbf{C}_{\text{obs}}$ which appears in the

simplified likelihood 2.78. The observational noise covariance is obtained by the standard empirical covariance. We recall that the observational data are obtained from an experimental campaign for the SILENE case, and numerical simulations for the BERP case. They are extracted from time list files with the sequential binning estimators (for the Feynman moments).

TABLE E.4: Summary of the observational data for the BERP and SILENE test cases.

	$\overline{y}$	$\mathbf{C}_{\text{eff}}$
SILENE $h = 15$ cm	$\begin{pmatrix} 3.6 \times 10^{2} \\ 2.2 \times 10^{-2} \\ 1.2 \times 10^{-2} \end{pmatrix}$	$\begin{pmatrix} 6.5 \times 10^{-2} & -2.6 \times 10^{-5} & 7.6 \times 10^{-4} \\ -2.6 \times 10^{-5} & 8.3 \times 10^{-6} & 2.7 \times 10^{-6} \\ 7.6 \times 10^{-4} & 2.7 \times 10^{-6} & 1.5 \times 10^{-4} \end{pmatrix}$
SILENE $h = 20$ cm	$\begin{pmatrix} 6.1 \times 10^{2} \\ 6.8 \times 10^{-2} \\ 2.5 \times 10^{-2} \end{pmatrix}$	$\begin{pmatrix} 3.2 \times 10^{-1} & 1.3 \times 10^{-4} & -2.3 \times 10^{-4} \\ 1.3 \times 10^{-4} & 1.1 \times 10^{-5} & 6.3 \times 10^{-6} \\ -2.3 \times 10^{-4} & 6.3 \times 10^{-6} & 2.8 \times 10^{-4} \end{pmatrix}$
SILENE $h = 25$ cm	$\begin{pmatrix} 1.1 \times 10^{3} \\ 2.5 \times 10^{-1} \\ 2.3 \times 10^{-1} \end{pmatrix}$	$\begin{pmatrix} 1.2 \times 10^{0} & 5.0 \times 10^{-4} & 4.4 \times 10^{-3} \\ 5.0 \times 10^{-4} & 2.1 \times 10^{-5} & 2.3 \times 10^{-5} \\ 4.4 \times 10^{-3} & 2.3 \times 10^{-5} & 7.9 \times 10^{-4} \end{pmatrix}$
BERP	$\begin{pmatrix} 7.3 \times 10^{3} \\ 3.9 \times 10^{-1} \\ 4.5 \times 10^{-1} \end{pmatrix}$	$\begin{pmatrix} 2.7 \times 10^{1} & -2.1 \times 10^{-3} & 7.2 \times 10^{-3} \\ -2.1 \times 10^{-3} & 4.4 \times 10^{-6} & 1.0 \times 10^{-5} \\ 7.2 \times 10^{-3} & 1.0 \times 10^{-5} & 5.7 \times 10^{-5} \end{pmatrix}$

E.5 Posterior distribution sampling

In Figure E.5 we provide the full set of 2D and 1D marginals of the posterior distribution for the BERP test case. The surrogate model used was **BL-GP-LMC2**. The marginal densities are estimated with Gaussian KDE.

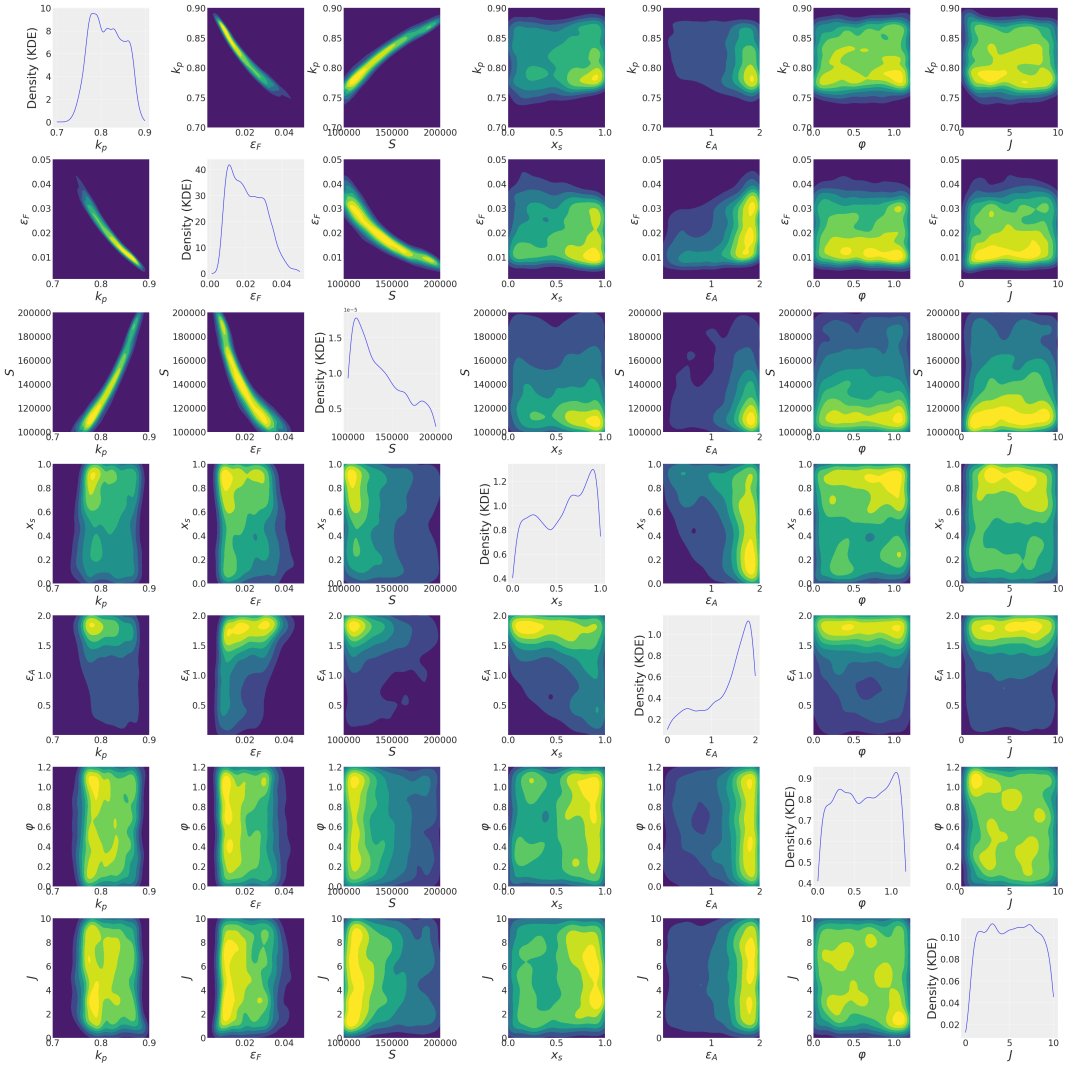

FIGURE E.5: 1D and 2D marginals obtained on the BERP test case with **BL-GP-LMC2**.

Appendix F

An introduction to Polynomial Chaos Expansions

This appendix presents a brief overview of Polynomial Chaos Expansion (PCE) and its use for surrogate modeling and sensitivity analysis. It is inspired by [Gar23] to which we refer for further details.

F.1 PCE for scalar inputs

Polynomial chaos expansion is a regression technique commonly used to build surrogate models. It consists of expressing a random variable as a polynomial function of the uncertain inputs [Wie38]. In this appendix, we provide an introduction to polynomial chaos expansion (PCE), which is used in Section 2.3.1.5 to perform sensitivity analysis on the extended input space of the neutron correlation inverse problem.

This appendix is organized as follows. To begin with we introduce the general results on orthogonal polynomials. After that, we focus on scalar PCE and then extend the framework to multi-output regression. Finally, we conclude this appendix by looking specifically at PCE for sensitivity analysis studies.

F.1.1 Orthogonal polynomials

Polynomial chaos expansion is based on the concept of orthogonal polynomial families. In this section, we provide a brief description of orthogonal polynomial families and some of their properties.

Definition F.1.1. *Let μ be a positive measure on $\mathbb{R}$. A family of polynomials $(P_n)_{n\in\mathbb{N}}$ is said to be orthogonal for the measure μ is $\deg(P_n) = n$ for all $n \in \mathbb{N}$ and there exists $(c_n)_{n\in\mathbb{N}}$ such that :*

$$\int_{\mathbb{R}} P_n(x)P_m(x)d\mu(x) = c_n\delta_{n,m} \tag{F.1}$$

where δ is the Kronecker symbol. If $c_n = 1$ for all $n \in \mathbb{N}$, the family is said to be orthonormal.

There exists a wide variety of orthogonal polynomial families. We are not covering them all in this appendix. However, we introduce Legendre and Hermite polynomials, which are the two most frequently encountered.

Definition F.1.2. *The family of Legendre polynomials* $(\mathrm{Le}_n)_{n\in\mathbb{N}}$ *is a family of polynomials that are implicitly defined by the following differential equation:*

$$(1-x^2)\frac{d^2\mathrm{Le}_n}{dx^2} - 2x\frac{d\mathrm{Le}_n}{dx} + n(n+1)\mathrm{Le}_n(x) = 0 \tag{F.2}$$

for $x \in [-1,1]$ *and* $n \in \mathbb{N}$.
The family of Legendre polynomials is orthogonal for the uniform measure on $[-1,1]$.

Definition F.1.3. *The family of Hermite polynomials* $(\mathrm{He}_n)_{n\in\mathbb{N}}$ *is defined for* $x \in \mathbb{R}$ *by:*

$$\mathrm{He}_n(x) = (-1)^n e^{\frac{x^2}{2}} \frac{d^n}{dx^n} e^{-\frac{x^2}{2}}. \tag{F.3}$$

The family of Hermite polynomials is orthogonal for the measure of density $\frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$ *which is the density of the standard normal distribution.*

Using these orthogonal families, it is possible to show that any function $f \in L^2(\mu)$ can be well-approximated by a polynomial development on an orthogonal family w.r.t. the measure μ.

Theorem F.1.1. *Let* $(P_n)_{n\in\mathbb{R}}$ *be a family of orthogonal polynomials for a given measure* μ. *Let* f *be a function in the Hilbert space* $L^2(\mu)$. *The orthogonal projection of* f *on the set of polynomials with degree smaller or equal* n, *denoted by* Π_n *is given by:*

$$\Pi_n f = \sum_{k=0}^{n} \frac{\langle f, P_k\rangle_{L^2(\mu)}}{\langle P_k, P_k\rangle_{L^2(\mu)}} P_k. \tag{F.4}$$

Besides the orthogonal projection converges towards f *in* $L^2(\mu)$ *when* $n \to +\infty$.

$$\|\Pi_n f - f\|^2_{L^2(\mu)} = \sum_{k=n+1}^{+\infty} \frac{\langle f, P_k\rangle^2_{L^2(\mu)}}{\langle P_k, P_k\rangle_{L^2(\mu)}} \xrightarrow[n\to+\infty]{} 0. \tag{F.5}$$

From this theorem, one can see that it is possible to build a surrogate model of the function f using a finite polynomial development on the orthogonal basis. This is the building block of polynomial chaos expansion.

F.1.2 Wiener polynomial expansion

Let us consider a random variable X representing the uncertain inputs. For now, the random variable is assumed to follow a standard normal distribution $X \sim \mathcal{N}(0,1)$. Let μ be the measure associated with the standard normal distribution density. Now let us consider that the output is of the form $Y = f(X)$ for $f\colon \mathbb{R} \to \mathbb{R}$ with $f \in L^2(\mu)$. Based on theorem F.1.1, the function f can be represented as a finite development on the family of Hermite polynomials, which are orthogonal for the measure μ. For a fixed order $M \geq 1$, the output is thus approximated by:

$$Y_M = \sum_{k=0}^{M} y_k \mathrm{He}_k(X). \tag{F.6}$$

This approximation is efficient and one can even show that Y_M is the best approximation of Y by a polynomial function of degree lower or equal to M. Based on the

orthogonal projection theorem, the coefficients of the development are given by:

$$y_k = \frac{\langle f, \mathrm{He}_k \rangle_{L^2(\mu)}}{\langle \mathrm{He}_k, \mathrm{He}_k \rangle_{L^2(\mu)}} = \frac{\mathbb{E}[Y \mathrm{He}_k(X)]}{\mathbb{E}[\mathrm{He}_k(X)^2]}. \tag{F.7}$$

F.2 Extension to multi-dimensional inputs

Definition F.2.1. *The input space $\mathcal{X}$ is now considered multi-dimensional $\mathcal{X} \subset \mathbb{R}^p$. A multivariate polynomial is a function $P : \mathbb{R}^p \to \mathbb{R}$ defined for any element $x = (x_1, ..., x_p) \in \mathcal{X}$ by:*

$$P(x) = \sum_{\boldsymbol{\alpha} \in \mathbb{N}^p} c_{\boldsymbol{\alpha}} x^{\boldsymbol{\alpha}} \tag{F.8}$$

where $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_p)$ is a multi-index, $x^{\boldsymbol{\alpha}} = \prod_{i=1}^{p} x_i^{\alpha_i}$ and where there are only a finite number of non-zero coefficients $c_{\boldsymbol{\alpha}}$. The degree of this polynomial is defined by:

$$\deg(P) = \max\{|\boldsymbol{\alpha}|,\ c_{\boldsymbol{\alpha}} \neq 0\}. \tag{F.9}$$

Definition F.2.2. *A family of multivariate polynomials $\{P_{\boldsymbol{\alpha}},\ \boldsymbol{\alpha} \in \mathbb{N}^p\}$ is said to be weakly orthogonal for the measure μ if for any polynomial Q such that $\deg(Q) < |\boldsymbol{\alpha}|$:*

$$\langle P_{\boldsymbol{\alpha}}, Q \rangle_{L^2(\mu)} = 0. \tag{F.10}$$

Definition F.2.3. *A family of multivariate polynomials $\{P_{\boldsymbol{\alpha}},\ \boldsymbol{\alpha} \in \mathbb{N}^p\}$ is said to be strongly orthogonal for the measure μ if for any $\boldsymbol{\alpha} \neq \boldsymbol{\beta}$:*

$$\langle P_{\boldsymbol{\alpha}}, P_{\boldsymbol{\beta}} \rangle_{L^2(\mu)} = 0. \tag{F.11}$$

In the case where the inputs are independent, meaning that the measure on $\mathcal{X} \subset \mathbb{R}^p$ is the product of the marginal measures $(\mu_i)_{1 \leq i \leq p}$:

$$d\mu(x) = d\mu_1(x_1)...d\mu_p(x_p) \tag{F.12}$$

then it is possible to build a strongly orthogonal polynomial family based on families of polynomials orthogonal to each marginal measure μ_i [Ern+12].

Theorem F.2.1. *If the measure μ verifies $d\mu(x) = d\mu_1(x_1)...d\mu_p(x_p)$ where the μ_i are the marginal measures, and if for $1 \leq i \leq p$, $\{P_{\alpha_i}^{(i)}, \alpha_i \in \mathbb{N}\}$ is an orthogonal family of polynomials for the measure μ_i, then the family $\{P_{\boldsymbol{\alpha}},\ \boldsymbol{\alpha} \in \mathbb{N}^p\}$ is strongly orthogonal for the measure μ where $P_{\boldsymbol{\alpha}}$ is defined by :*

$$P_{\boldsymbol{\alpha}}(x) = \prod_{i=1}^{p} P_{\alpha_i}^{(i)}(x_i). \tag{F.13}$$

The Wiener polynomial expansion shown in equation (F.6) is suited for one-dimensional input variables X. Now let us consider multi-dimensional inputs. Let $X = (X_i)_{1 \leq i \leq p}$ be a random variable, where the X_i are independent and have marginal densities μ_i. Let $(P_{\alpha_i}^{(i)})_{\alpha_i \in \mathbb{N}}$ be a family of polynomials orthogonal for the measure μ_i. Let us consider the output $Y = f(X)$ where $f : \mathbb{R}^p \to \mathbb{R}$ and $\mathbb{E}\left[f(X)^2\right] < +\infty$.

Then the tensorized polynomials defined by $P_{\boldsymbol{\alpha}}(x) = \prod_{i=1}^{p} P_{\alpha_i}^{(i)}(x_i)$ form a strongly

orthogonal family of multivariate polynomials. It is thus possible to write the output Y with a polynomial function :

$$Y = \sum_{\boldsymbol{\alpha} \in \mathbb{N}^p} y_{\boldsymbol{\alpha}} P_{\boldsymbol{\alpha}}(X). \tag{F.14}$$

The strong orthogonal property simplifies the expression of the coefficients, which can be obtained by orthogonal projection :

$$y_{\boldsymbol{\alpha}} = \frac{\mathbb{E}\left[f(X) P_{\boldsymbol{\alpha}}(X)\right]}{\mathbb{E}\left[P_{\boldsymbol{\alpha}}(X)^2\right]}. \tag{F.15}$$

In the same way that the polynomial expansion is truncated to a given order, one has to select only the multi-indices in a given finite subset $\mathcal{D} \subset \mathbb{N}^p$.

The first intuition that comes to mind is to limit the development up to a certain degree of the multivariate polynomial α_m such that $\mathcal{D}_{\alpha_m} = \{\boldsymbol{\alpha} \in \mathbb{N}^p, |\boldsymbol{\alpha}| \leq \alpha_m\}$. However, when the dimension increases, the number of multi-indices kept dramatically explodes, since $\text{Card}(\mathcal{D}_{\alpha_m}) = \frac{(p+\alpha_m)!}{p!\alpha_m!}$. As an illustration, for $p = 7$ and $\alpha_m = 10$, $\text{Card}(\mathcal{D}_{\alpha_m}) = 19448$.

Hence, the number of multi-indices must be reduced. One of the most common techniques (see [BS10]) is to consider that the most impactful polynomials are the univariate polynomials of high degrees and the multivariate of low degrees. This assumption usually corresponds to a function depending mainly on one of the inputs. This assumption is reflected by considering the subset $\mathcal{A}_{\alpha_m,r}$ defined for $r \in (0, 1]$ by :

$$\mathcal{A}_{\alpha_m,r} = \left\{ \boldsymbol{\alpha} \in \mathbb{N}^p, \left(\sum_{i=1}^{p} \alpha_i^r \right)^{1/r} \leq \alpha_m \right\}. \tag{F.16}$$

When $r = 1$, $\mathcal{A}_{\alpha_m,1} = \mathcal{D}_{\alpha_m}$ and for $r < 1$ the subset $\mathcal{A}_{\alpha_m,r}$ filters out the multivariate polynomials of high degree, which reduces the number of coefficients to estimate. Based on this subset, it is possible to build a surrogate model by truncating the development in equation (F.14) to the subset $\mathcal{A}_{\alpha_m,r}$:

$$Y \simeq Y_{\alpha_m,r} = \sum_{\boldsymbol{\alpha} \in \mathcal{A}_{\alpha_m,r}} y_{\boldsymbol{\alpha}} P_{\boldsymbol{\alpha}}(X). \tag{F.17}$$

F.3 Estimation of the coefficients

To obtain our surrogate model, one only has to estimate the coefficients $y_{\boldsymbol{\alpha}}$ in equation (F.17) for the multi-dimensional case. Two methods are usually considered.

F.3.1 Monte Carlo estimation

The coefficients can be expressed using the orthogonal projection of the truncated polynomial development of Y as shown in equation (F.15).
The expected values appearing in this equation are unknown and the first method to find the coefficients is to obtain them with a Monte Carlo estimation.
Assuming we have n independent samples $\mathbf{X} = (X^{(k)})_{1 \leq k \leq n}$ following the law μ, the

coefficients can be estimated by :

$$\widehat{y_{\boldsymbol{\alpha}}} = \frac{1}{n\mathbb{E}\left[P_{\boldsymbol{\alpha}}(X)^2\right]} \sum_{k=1}^{n} P_{\boldsymbol{\alpha}}(X^{(k)}) f(X^{(k)}). \tag{F.18}$$

F.3.2 Regression

The coefficients can also be estimated by regression. More specifically, let us consider the truncation $\mathcal{A}_{\alpha_m,r}$ and let $J = \mathrm{Card}(\mathcal{A}_{\alpha_m,r})$. Then the approximated surrogate model is given by equation (F.14). The objective is to find the coefficients that minimize the error between $Y_{\alpha_m,r}$ and the true output $Y = f(X)$. This is a minimization problem. Let $\mathbf{P} = (P_{\boldsymbol{\alpha}})_{\boldsymbol{\alpha}\in\mathcal{A}_{\alpha_m,r}}$ be the column vector of size J containing all the polynomials. The coefficients $\widehat{\mathbf{y}} = (\widehat{y_{\boldsymbol{\alpha}}})_{\boldsymbol{\alpha}\in\mathcal{A}_{\alpha_m,r}}$ verifying this are given by :

$$\widehat{\mathbf{y}} = \mathrm{argmin}_{\mathbf{c}\in\mathbb{R}^J}\, \mathbb{E}\left[\left(\mathbf{c}^T\mathbf{P}(X) - f(X)\right)^2\right]. \tag{F.19}$$

Once again, the true mean is unknown but can be approximated assuming we have n independent samples $\mathbf{X} = (X^{(k)})_{1\le k\le n}$:

$$\widehat{\mathbf{y}} \simeq \mathrm{argmin}_{\mathbf{c}\in\mathbb{R}^J}\, \frac{1}{n}\sum_{k=1}^{n}\left[\left(\mathbf{c}^T\mathbf{P}(X^{(k)}) - f(X^{(k)})\right)^2\right]. \tag{F.20}$$

This minimization problem can be solved directly in matrix form. Let us introduce the matrix $\mathbf{A} \in \mathcal{M}_{n,J}(\mathbb{R})$ defined by $A_{k,j} = P_j(X^{(k)})$ for $1 \le j \le J$ and $1 \le k \le n$, and the vector $\mathbf{F} \in \mathbb{R}^n$ defined by $F_k = f(X^{(k)})$ for $1 \le k \le n$. The vector of coefficients $\widehat{\mathbf{y}}$ is given by the product of the pseudo-inverse of $\mathbf{A}$ with the vector $\mathbf{F}$:

$$\widehat{\mathbf{y}} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{F}. \tag{F.21}$$

The estimation of the coefficients with the regression method is theoretically better since, by definition, they minimize the squared error. One can show that the error is a compromise between bias and variance of the estimation. The variance of the estimation can be reduced by increasing the size n of the training set. However, this does not change the bias, which is caused by the truncation of the polynomial development. To reduce the bias, one has to increase the polynomial order. At high order, however, the matrix $\mathbf{A}$ can become singular and can be required to use a regularization technique such as Tikhonov regularization.

F.3.3 Choice of the truncation

To choose an appropriate subset for the indices of the multivariate polynomials, one should have a metric to quantify the error. Considering the mean squared error, the metric to investigate is :

$$\mathcal{E} = \mathbb{E}\left[\left(\widehat{\mathbf{y}}^T\mathbf{P}(X) - f(X)\right)^2 \,\middle|\, \widehat{\mathbf{y}}\right]. \tag{F.22}$$

This error is estimated on a validation set $\mathbf{X}_* = (X_*^{(k)})_{1\le k\le n_*}$, independent from the training set $\mathbf{X} = (X^{(k)})_{1\le k\le n}$.

This error can also be estimated with a Leave-One-Out (LOO) cross-validation procedure. The idea is to estimate the coefficients with the whole training set, except the k-th value which is used to estimate an error. This method is iterated over the whole set, and the estimated error is the average error on the left-out case.

$$\mathcal{E}_{\text{LOO}} = \frac{1}{n}\sum_{k=1}^{N}\left(\widehat{\mathbf{y}}_{-k}^T\mathbf{P}(X^{(k)}) - f(X^{(k)})\right)^2 \tag{F.23}$$

where $\widehat{\mathbf{y}}_{-k}$ is the vector of coefficients estimated with the whole training set except the k-th value.
This procedure is time-consuming since the training phase is done N times. However, the LOO error can be simplified :

$$\mathcal{E}_{\text{LOO}} = \frac{1}{n}\sum_{k=1}^{n}\left(\frac{\widehat{\mathbf{y}}^T\mathbf{P}(X^{(k)}) - f(X^{(k)})}{1-q^{(k)}}\right)^2 \tag{F.24}$$

where $q^{(k)} = (\mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T)_{k,k} \in [0,1]$. This error can be seen in the mean-squared error estimated on the training set, weighted by the term $(1-q^{(k)})^{-1}$.

F.4 PCE in sensitivity analysis

In this section, we will see why PCE is widely used for sensitivity analysis. Recalling the introduction to sensitivity analysis in Section 2.3.1.5, Sobol indices are amongst the most-used sensitivity indices due to their interpretability and overall simplicity. Of course, Sobol's framework also comes with some limitations, which will not be further discussed here. Let us also assume that the inputs are independent here so that we can focus on standard Sobol indices. Keep in mind that what follows can still be generalized to Sobol indices for dependent inputs.
Suppose we have some surrogate model g of a black-box function f built with PCE, such that for $x \in \mathbb{R}^p$:

$$f(x) \simeq g(x) = \sum_{\boldsymbol{\alpha}\in\mathcal{A}} c_{\boldsymbol{\alpha}} P_{\boldsymbol{\alpha}}(x) \tag{F.25}$$

where $(P_{\boldsymbol{\alpha}})_{\boldsymbol{\alpha}\in\mathbb{N}^p}$ is the family of tensorized polynomials associated with the PCE coefficients $c_{\boldsymbol{\alpha}}$, and where $\mathcal{A} \subset \mathbb{N}^p$ is the truncation set.
Suppose we want to compute a Sobol index S_A for some subset $A \subset \{1,...,p\}$ which is defined by:

$$S_A = \frac{V_A}{\text{Var}\left[f(x)\right]} \tag{F.26}$$

where $V_A = \text{Var}\left[\mathbb{E}\left(f(x)|x_A\right)\right]$ as defined in Hoeffding decomposition (2.44) and x_A is the subvector of x where only the components in A are kept.

One can see clearly that the PCE model has the same form as the Hoeffding decomposition of the function f. Of course, the PCE decomposition is only an approximation due to the truncation. Provided the surrogate model is well-trained, the Sobol index S_A can be computed analytically by:

$$S_A = \frac{\sum_{\boldsymbol{\alpha}\in\mathbb{Z}_A^*}|c_{\boldsymbol{\alpha}}|^2}{\sum_{\boldsymbol{\alpha}\in\mathbb{N}^p,\boldsymbol{\alpha}\neq 0}|c_{\boldsymbol{\alpha}}|^2} \simeq \frac{\sum_{\boldsymbol{\alpha}\in\mathcal{A}\cap\mathbb{Z}_A^*}|c_{\boldsymbol{\alpha}}|^2}{\sum_{\boldsymbol{\alpha}\in\mathcal{A},\boldsymbol{\alpha}\neq 0}|c_{\boldsymbol{\alpha}}|^2} \tag{F.27}$$

where $\mathbb{Z}_A^* = \left\{\mathbf{k} \in \mathbb{N}^p | k_j = 0 \text{ if } j \notin A \text{ and } k_j \neq 0 \text{ if } j \in A\right\}$.
This is the strength of PCE for sensitivity analysis, it removes the need for Monte Carlo estimation of the Sobol indices.

ECOLE
DOCTORALE
DE MATHEMATIQUES
HADAMARD

Titre : Étude des mesures de corrélations neutroniques, modélisation et apprentissage automatique, estimation bayésienne de paramètres nucléaires

Mots clés : Problèmes inverses, méta-modélisation, incertitudes, neutronique

Résumé : La sécurité nucléaire consiste à protéger la population et l'environnement contre les actes de malveillance impliquant des substances radioactives. L'Agence Internationale de l'Énergie Atomique (AIEA) exhorte les états à présenter une stratégie de sécurité nucléaire rapide, robuste et fiable, en réponse aux tensions géopolitiques croissantes. En particulier, l'identification des matières nucléaires fissiles est un élément fondamental de toute stratégie de sécurité nucléaire. Pour garantir la fiabilité de la réponse à toute menace, la quantification et le contrôle des incertitudes présentes dans les méthodes mathématiques sous-jacentes sont obligatoires. Cette thèse se place à l'intersection de l'identification des matières fissiles et de la quantification des incertitudes. L'objectif général est de développer des méthodes mathématiques et numériques adaptées à l'analyse du bruit neutronique dans les systèmes sous-critiques de puissance nulle. Cette technique de mesure passive est, avec la spectroscopie gamma, un aspect capital pour l'identification de matière fissile.
La méthodologie présentée dans ce manuscrit se base sur une résolution bayésienne d'un problème inverse, dont les observations proviennent de l'étude des corrélations temporelles entre les neutrons induits par fission. La résolution standard de ce problème est irréalisable telle quelle, en raison du coût du code Monte-Carlo utilisé pour le transport des neutrons. Cette thèse présente un cadre dans lequel le code Monte-Carlo est remplacé par divers méta-modèles, dont les incertitudes intrinsèques sont introduites dans le problème inverse. La procédure de quantification des incertitudes englobe les incertitudes épistémiques et aléatoriques dans un cadre commun. Cette stratégie peut être améliorée à l'aide de plans d'expériences séquentiels conçus spécifiquement pour le problème inverse, ou avec l'introduction de corrélations gamma qui aident à réduire les incertitudes résiduelles. Enfin, nous abordons une approche connexe qui contourne la résolution du problème inverse par une paramétrisation de la classe de distributions a posteriori à l'aide de distributions lambda généralisées.

Title : Study of neutron correlations, modeling, supervised learning and Bayesian estimation of nuclear parameters

Keywords : Inverse problems, surrogate modeling, uncertainty, neutronics

Abstract : Nuclear security is the task of protecting the population and the environment against malicious acts involving radioactive substances. The International Atomic Energy Agency (IAEA) exhorts states to present a fast, robust, and reliable nuclear security strategy, in response to the increasing geopolitical tensions. In particular, the identification of nuclear fissile matter is a foundational element of any nuclear security strategy. To ensure the reliability of the response to any threat, the quantification and control of the uncertainties embedded in the underlying mathematical methods are mandatory. This thesis is located at the crossroads of fissile matter identification and uncertainty quantification. The general objective is to develop mathematical and numerical methods adapted to the neutron noise analysis in zero-power subcritical systems. This passive measurement technique is, along with gamma spectroscopy, a focal point of fissile matter identification.
The methodology presented in this manuscript is based on a Bayesian resolution of an inverse problem, whose observations come from the study of temporal correlations between fission-induced neutrons. The standard resolution of this problem is intractable due to the cost of the Monte Carlo code for neutron transport. This thesis presents a framework in which the computer model is replaced by various surrogates, whose intrinsic uncertainties are fed into the inverse problem. The uncertainty quantification procedure encompasses both epistemic and aleatoric uncertainties into a common framework. This strategy can be improved with the help of sequential design strategies built specifically for the inverse problem, or with the introduction of gamma correlations which helps in reducing the residual uncertainties. Finally, we discuss a connex approach that circumvents the resolution of the inverse problem with a parametrization of the class of posterior distributions with the help of generalized lambda distributions.

Institut Polytechnique de Paris
91120 Palaiseau, France