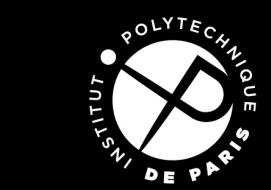
# Low-cost methods for constrained multi-objective optimization under uncertainty

Mickael Rivier



INSTITUT POLYTECHNIQUE DE PARIS

NNT : 2020IPPAX029

Thèse de doctorat

# Low-cost methods for constrained multi-objective optimization under uncertainty

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 30 Septembre 2020, par

**MICKAËL RIVIER**

Composition du Jury :

| | |
|---|---|
| Grégoire Allaire<br>Professeur, École polytechnique | Président |
| Régis Duvigneau<br>Chargé de Recherche, Inria | Rapporteur |
| Guillaume Perrin<br>Ingénieur-Chercheur, CEA-DAM | Rapporteur |
| Olivier Le Maître<br>Directeur de Recherche, CMAP-CNRS | Examinateur |
| Clémentine Prieur<br>Professeur, Université Grenoble | Examinateur |
| Pietro Marco Congedo<br>Chargé de Recherche, Inria | Directeur de thèse |
| Stéven Robidou<br>Ingénieur, ArianeGroup | Encadrant industriel |

# Abstract

Optimization Under Uncertainty is a fundamental axis of research in many companies nowadays, due to both the evergrowing computational power available and the need for efficiency, reliability and cost optimality. Among others, some challenges are the formulation of a suitable metric for the optimization problem of interest and the search for an ideal trade-off between computational cost and accuracy in the case of problems involving complex and expensive numerical solvers. The targeted class of problem here is constrained multi-objective optimization where fitness functions are uncertainty-driven metrics, such as statistical moments or quantiles.

This thesis relies on two main ideas. First, the accuracy for approximating the objectives and constraints at a given design should be driven by the probability for this design of being non-dominated. This choice permits to reduce the effort for evaluating designs which are unlikely to be optimal. To this extent, we introduce the concept of probabilistic dominance for constrained multi-objective optimization under uncertainty through the computation of the so-called Pareto-Optimal Probability (POP). Secondly, these approximated evaluations and their associated errors can be used to construct a predictive representation of the objectives and constraints over the whole design space to accelerate the optimization process. Overall, the approximation of different uncertainty-based metrics with tunable accuracy and the use of a Surrogate-Assisting strategy are the main ingredients of the proposed algorithm, called SAMATA. This approach is flexible in terms of metrics formulations and reveals very parsimonious. Note that this algorithm is applicable with generic optimization methods.

This thesis then explores the influence of the error distribution on the algorithm performance. We first make a simplifying and conservative assumption by considering a Uniform distribution of the error. In this case, the proposed formulation yields a Bounding-Box approach, where the estimation error can be regarded with the abstraction of an interval (in one-dimensional problems) or a product of intervals (in multi-dimensional problems) around the estimated value, naturally allowing for the computation of an approximated Pareto front. This approach is then supplemented by a Surrogate-Assisting strategy that directly estimates the objective and constraint values. Under some hypotheses, we study the convergence properties of the method in terms of the distance between the approximated Pareto front and the true continuous one.

Secondly, we propose to compute non-parametric approximations of the error distributions with a sampling-based technique. We propose a first algorithm relying on an approximation scheme with controlled accuracy for drawing large-scale Gaussian random field realizations in the coupled space between design and uncertain parameters.

It notably permits a sharper computation of the POP and detects possible correlations between the different objectives and constraints. Joint realizations can be drawn on multiple designs in order to generate Surrogate-Assisting models of the objectives and constraints. Since the construction of a Gaussian random field can be hard in the context of high dimensionality or non-parametric inputs, we also propose a KDE-based Surrogate-Assisting model as an extension of the classical heteroscedastic Gaussian process, with the capability to take as input disjoint objective and constraint realizations.

We assess the proposed variants on several analytical uncertainty-based optimization test-cases with respect to an a priori metamodel approach by computing a probabilistic modified Hausdorff distance to the exact Pareto optimal set. The method is then employed on several engineering applications: the two-bar truss, a thermal protection system for atmospheric reentry and the blades of an Organic Rankine Cycle turbine.

**Keywords** Uncertainty • Optimization • Robustness and reliability • Statistics • Surrogate model

---

L'optimisation sous incertitude est un axe de recherche fondamental chez de nombreuses entreprises, de par l'accroissement de la puissance de calcul et la recherche continuelle d'efficience, de fiabilité et d'optimalité des coûts. Parmi les difficultés associées, on peut citer, entre autres, la bonne formulation d'une métrique d'optimisation pour le problème d'intérêt et la recherche d'un compromis idéal entre parcimonie et précision lorsque des simulations complexes et coûteuses sont impliquées. Cet travail vise à traiter les problèmes d'optimisation multi-objectif sous contrainte, où les objectifs et contraintes tiennent compte des paramètres incertains sous la forme, par exemple, de moments statistiques ou de quantiles.

Cette thèse repose sur deux idées principales. Premièrement, la précision d'approximation des objectifs et contraintes à chaque point devrait être guidée par la probabilité de ce point d'être non-dominé. Cela permet de réduire l'effort alloué aux points dont l'optimalité est peu probable. Pour cela, nous introduisons le concept de dominance probabiliste pour l'optimisation multi-objectif sous contrainte, basé sur le concept de probabilité d'être Pareto-optimal (POP). Deuxièmement, dans le but d'accélérer le processus d'optimisation, ces approximations et leur erreur associée peuvent être exploitée afin de construire un modèle prédictif des objectifs et contraintes. Ces deux techniques, approximations des objectifs et contraintes avec une précision adaptative et assistance par métamodèle, sont au coeur de l'algorithme proposé, appelé SAMATA. Ce dernier est flexible en termes de métriques utilisées et se révèle très parcimonieux. Cet algorithme est d'ailleurs applicable avec des méthodes génériques d'optimisation.

Nous explorons ensuite l'influence de la distribution de l'erreur d'approximation. Une première supposition simplificatrice et conservative consiste à considérer cette distribution uniforme. La formulation proposée devient une approche par boîtes, ou l'erreur est réduite à un intervalle (en mono-dimensionel) ou un produit d'intervalles (en multi-dimensionel) autour de la valeur estimée, ce qui permet naturellement le calcul d'un front de Pareto imprécis. Cette approche est couplée à une assistance

par métamodèle, construit directement sur les objectifs et contraintes. Sous quelques hypothèses, nous étudions la convergence du front de Pareto approximé vers le front réel.

Par la suite, nous proposons une approximation non-paramétrique par échantillonnage de la distribution de l'erreur. Un premier algorithme repose sur une approximation avec contrôle de précision permettant de tirer des réalisations de champ Gaussien sur un très grand nombre de points dans l'espace couplé entre variables de contrôle et incertaines. Cela permet notamment des comparaisons probabilistes plus précises et la capture de corrélations entre les différents objectifs et contraintes. Des réalisations jointes sont tirées à différents points afin de générer un métamodèle pour assister l'optimisation. La construction de ce champ Gaussien dans l'espace couplé peut se révéler infaisable lorsque la dimensionalité est trop élevée ou que les incertitudes sont non-paramétriques. Nous proposons donc aussi la construction d'un métamodèle, construit sur des réalisations disjointes des objectifs et contraintes, comme extension des processus Gaussiens hétéroscedastiques classiques.

Les variantes proposées sont testées sur plusieurs cas analytiques d'optimisation sous incertitudes, et sont comparées à une approche par métamodèle a priori à l'aide d'un indicateur probabiliste de distance de Hausdorff au front de Pareto exact. La méthode est finalement mise en pratique sur plusieurs applications pour l'ingénierie : la charpente à deux barres, un système de protection thermique pour la rentrée atmosphérique et les pâles d'une turbine à cycle de Rankine organique.

**Mots Clefs** Incertitudes • Optimisation • Robustesse et fiabilité • Statistiques • Modèle de substitution

# Acknowledgments

Bien que mon nom soit mis en avant, cette thèse est avant tout le résultat de collaborations, de discussions et de moments partagés dans le cadre professionnel comme personnel.

Au premier plan de ces discussions se trouve bien sûr les membres du jury. Je remercie particulièrement Régis D et Guillaume P pour avoir accepté d'endosser le rôle de rapporteur, et Grégoire A pour avoir présidé la séance. Plus généralement, je suis reconnaissant envers l'ensemble des membres du jury pour leur bienveillance et les nombreuses pistes qu'ils auront su proposer après l'évaluation de ce travail. Je soulignerai finalement leur disponibilité qui aura rendu possible la soutenance de thèse malgré le contexte sanitaire spécial de cette année 2020.

J'ai eu la chance de réaliser cette thèse sous un contrat CIFRE chez ArianeGroup, ce qui m'a permis de côtoyer les mondes académique et industriel durant trois ans.

Du côté académique, mes plus sincères remerciements vont bien sûr à mon directeur de thèse Pietro Marco C, qui m'a proposé cette aventure doctorale à l'issue de deux stages que j'ai eu le plaisir de mener en sa compagnie. Cela fait donc maintenant plusieurs années que nous bourlinguons ensemble dans les contrées de la quantification d'incertitudes et de l'optimisation, et j'apprécie toujours autant son sens de l'organisation et de la rigueur, sa profonde gentillesse et la confiance qu'il a su m'accorder. Je suis particulièrement reconnaissant de l'impressionnante force de travail dont il fait preuve et de l'énergie qu'il a su me transmettre lors de la rédaction de ce manuscrit.

Durant les premières années de ce doctorat, j'ai fait partie de l'équipe de recherche CARDAMOM à Inria Bordeaux Sud-Ouest. Mes plus proches collègues furent les autres doctorants de Pietro, avec qui nous avons fièrement fondé la *UQTeam*. Je remercie ainsi François S, Nassim R et Andrea C pour les (très) nombreuses discussions et conseils techniques, mais aussi pour tous les bons moments passés ensemble, sous le soleil grec comme dans les pubs écossais et bordelais. Plus généralement, mes remerciements vont à tous les membres, passés comme présents, de l'équipe CARDAMOM, et particulièrement au chef d'équipe Mario R et à l'assistante d'équipe Anne-Laure G pour leur accompagnement administratif.

La fin de thèse s'est quant à elle déroulée au sein de l'équipe projet DeFI à Inria Saclay. J'y ai eu le plaisir de rencontrer Olivier LM, qui m'a fait profiter de sa très grande expérience en gestion des incertitudes, ainsi que les différents doctorants et post-doctorants de l'équipe. Je remercie particulièrement l'équipe administrative, notamment Marie E et Nasséra N, qui m'a accompagné jusqu'à la soutenance de thèse.

Chez ArianeGroup, de manière similaire, mes premiers remerciements iront à Stéven R, qui fut mon encadrant industriel dans cette aventure. Je lui suis très reconnaissant pour toute l'aide qu'il m'a apportée sur les diverses questions organisationnelles et administratives, mais également pour l'autonomie qu'il m'a offerte durant ces trois ans. Je le remercie aussi sincèrement pour l'énergie qu'il a su investir pour me faire connaître au sein de l'entreprise et afin de me proposer régulièrement de nouveaux challenges techniques, qui m'ont ouvert de nombreux axes de réflexion pour traiter ce sujet de thèse.

J'ai réalisé ce doctorat dans le service *Design methods*, qui a en charge le développement et le déploiement de méthodologies et d'outils transverses pour les autres services. J'ai notamment eu le plaisir de partager le bureau d'Ephi M et Florine W qui, en plus de leur bonne humeur, ont su me transmettre de nombreuses connaissances techniques sur l'ingénierie aérospatiale, les clusters de calcul ou encore l'optimisation topologique. Ces *Optim Girls*, source intarissable d'astuces informatiques et de potins en tout genre, jouèrent indiscutablement un rôle central dans ma pleine intégration au sein d'ArianeGroup, et je leur adresse ici ma profonde reconnaissance. Plus généralement, je remercie l'ensemble de l'équipe pour son chaleureux accueil et pour les nombreux moments conviviaux partagés. Thierry G-L, Laetitia P, Cyril A et Yves M se reconnaitront sûrement en évoquant les repas d'équipe, les VnB fléchettes à la main ou les parties enflammées de badminton. Je tiens enfin à remercier les différents alternants et stagiaires, Fabien T, Marceau B, Mathilde G, Médérick C et Morgane M pour leur bonne humeur et leur curiosité scientifique débordante.

En prenant encore un peu de recul, ce service fait partie du département *Analysis*, qui regroupe l'expertise technique en simulation numérique et travaille en collaboration avec les différents départements de conception. Je remercie donc chaleureusement André L et plus récemment Virginie V pour leur accueil et leur accompagnement durant ces trois ans. Je ne me risquerai pas à énumérer l'ensemble des membres du département, mais assure à chacun toute ma reconnaissance pour les échanges que nous avons pu avoir, avec un clin d'oeil particulier à mes coéquipiers d'escape game et autres compagnons biérophiles.

Je terminerai enfin le volet professionnel en remerciant toutes les personnes que j'ai pu croiser en conférence, lors de séminaires, ou à n'importe quelle occasion, et qui ont pu me transmettre un peu de leurs réflexions scientifiques. Je citerai notamment Sébastien DV pour ses nombreux conseils avisés, échangés devant un tableau blanc comme au comptoir d'un pub.

Afin de mener à bien des travaux de recherche, un bon environnement professionnel est certes indispensable. J'avancerai cependant qu'être bien entouré dans sa vie personnelle est plus crucial encore pour conserver une énergie positive et une vive curiosité d'esprit. Et de ce côté-là, je ne pourrais espérer meilleurs camarades.

L'école d'ingénieur dans laquelle je fis mes études, l'ENSEIRB-MATMECA fut prolifique en bonnes rencontres. Bien que nous nous soyons dispersés à travers la France, il reste un noyau dur à Bordeaux, que j'ai la joie de croiser régulièrement. Je remercie profondément Félix M, Guillaume M et Marie-Marthe G pour les après-midi jeux de société, les soirées film et les diverses dégustations de boisson houblonnée (encore elle). Je me réjouis d'avance de retourner passer des week-ends avec Laurent C et Claire M, qui sont maintenant bien installés dans la mégapole Belin-Béliet et espère trouver le temps de croiser plus souvent Corentin P, avec qui c'est toujours un plaisir de passer du temps. Enfin, je souhaite le meilleur à Joan R et Xavier L, qui nous ont quittés pour la vie canadienne et parisienne, respectivement.

Parmi les égarés parisiens, je me dois de citer et de remercier Valentin S pour les nombreux moments passés ensembles, à discourir littérature, politique, sciences, et littérature à nouveau, en espérant avoir l'occasion de se croiser aussi souvent que possible à l'avenir. Je termine mon tour de la capitale en remerciant Martin C, mon partenaire de choc de l'école d'ingénieur à l'humour et au charisme difficilement égalables, pour les nombreuses heures passées ensembles.

Retournons finalement en bord de Garonne dans la belle ville de Toulouse. J'ai la chance de pouvoir régulièrement retrouver la fine équipe, Lucile D, François H, Jean-François P et Sophie A, que ce soit pour le traditionnel nouvel an, les week-ends à Lac(h)anau ou les virées au ski. Ces moments sont de réelles bouffées d'énergie et je leur adresse ici toute ma gratitude pour ces précieux intermèdes. Je citerai finalement Théo M et Pacôme M, d'autres compagnons toulousains que je croise toujours avec le plus grand bonheur.

Certaines amitiés remontent plus loin encore et m'ont indéniablement permis de me construire au fil des ans. Je remercie ainsi Bernard D, avec qui nous formions un duo inséparable durant les années de classes préparatoires, et Anthony G, un ami très cher de longue date qui fut mon partenaire dans de nombreuses aventures.

Comme à l'accoutumée, je terminerai ces remerciements par le cercle familial. Je tiens ici à souligner la profonde reconnaissance que je ressens envers mes parents et mon frère, qui m'ont soutenu sans faille durant toutes les épreuves que j'ai pu traverser, et sans qui je ne serais certainement pas en train d'écrire ces mots. Merci du fond du cœur. J'en profite aussi pour remercier ma belle-famille pour sa profonde gentillesse et la chaleur dont chacun d'eux fait preuve à mon égard.

Ces derniers mots seront finalement adressés à mon soleil quotidien, cette douce turbulence qui dessine les sons et fait chanter les couleurs, à ma compagne Emilie V. Après tant d'années partagées sur notre navire, je pourrais prétendre prendre pied, prédire, prévoir. Mais ce serait mal connaître son énergie instinctive, créative, ce vif qui me pousse en avant et me tient en haleine. Je te remercie pour les rires et les pleurs, pour cette complicité simple et vraie, pour ce soutien inépuisable que tu m'offres comme une évidence. Chère partenaire, la coque est solide et les soutes sont pleines, attrapons ensemble le gouvernail et voguons vers de nouvelles aventures.

# Publications

## Journal Papers

- Rivier, M., Congedo, P. M. (2019). Surrogate-assisted Bounding-Box approach for optimization problems with tunable objectives fidelity. *Journal of Global Optimization*, 75(4), 1079-1109.

- Rivier, M., Congedo, P. M. (2018). Surrogate-Assisted Bounding-Box Approach Applied to Constrained Multi-Objective Optimisation Under Uncertainty. [Research Report] RR-9214, Inria Saclay Ile de France. 2018, pp.1-37. hal-01897399v2 *Submitted*

- Rivier, M., Razaaly, N., Congedo, P. M. (2020). Non-Parametric Measure Approximations for Constrained Multi-Objective Optimisation under Uncertainty. *In preparation*

## International Conferences

- Rivier, M., Congedo, P. M. (2017). A novel framework for multi-objective optimization under uncertainties: An imprecise Pareto front and evolutive metamodel coupling. *UNCECOMP 2017 - 2nd International Conference on Uncertainty Quantification in Computational Sciences and Engineering*, Jun 2017, Rhodes, Greece. hal-01670998

- Rivier, M., Congedo, P. M. (2017). Surrogate-Assisted Bounding-Box Approach for Uncertainty-Based Multi-Objective Optimization Problems. *EUROGEN 2017 - International Conference on Evolutionary and Deterministic Methods for Design Optimization and Control*, Sep 2017, Madrid, Spain. hal-01671024

- Rivier, M., Congedo, P. M. (2018). Low-cost optimization under uncertainty through box representation of robustness measures. *BIOMA 2018 - 8th International Conference on Bioinspired Optimization Methods and their Applications*, May 2018, Paris, France. hal-01841372

- Rivier, M., Congedo, P. M. (2018). Efficient optimization under uncertainty within the SABBa framework. *ECCM-ECFD 2018 - 6th European Conference on Computational Mechanics - 7th European Conference on Computational Fluid Dynamics*, Jun 2018, Glasgow, United Kingdom. hal-01841360

- Rivier, M., Congedo, P. M. (2019). Different Measure Approximations for Efficient Constrained Multi-Objective Optimization under Uncertainty. *UQOP - Uncertainty Quantification and OPtimization*, Mar 2019, Paris, France. hal-02286145

- Rivier, M., Congedo, P. M. (2019). A Surrogate-Assisted Multi-fidelity Measure Approximation Framework for Efficient Constrained Multiobjective Optimization Under Uncertainty. *UNCECOMP 2019 - 3rd International Conference on Uncertainty Quantification in Computational Sciences and Engineering*, Jun 2019, Hersonissos, Greece. hal-02286156

## Seminar

- Rivier, M., Robidou, S., Congedo, P. M. (2017). Innovative methods for an efficient and low-cost optimization under uncertainties for industrial problems. *ArianeGroup PhDDays 2017*, Dec 2017, Issac, France.

- Rivier, M., Robidou, S., Congedo, P. M. (2018). Different Measure Approximations for Efficient Constrained Multi-Objective Optimization under Uncertainty. *ArianeGroup PhDDays 2018*, Dec 2018, Issac, France.

- Rivier, M., Congedo, P. M. (2018). Low-cost optimization under uncertainty through box representation of robustness measures. *Mascot-num 2018*, Mar 2018, Nantes, France. hal-01841352

- Rivier, M., Congedo, P. M. (2019). Different Measure Approximations for Efficient Constrained Multi-Objective Optimization under Uncertainty. *SMAI 2019 - 9e Biennale Française des Mathématiques Appliquées et Industrielles*, May 2019, Guidel, France. hal-02286141

## Poster

- Rivier, M., Congedo, P. M. (2019). Different measure approximations for efficient constrained multi-objective optimization under uncertainty. *SIAM CSE19 - Conference on Computational Science and Engineering*, Feb 2019, Spokane, United States. hal-02286159

## Training school

- UTOPIAE Opening Training School. Nov 2017, Scottish Universities Insight Institute, University of Strathclyde, Glasgow.

# Contents

# List of Figures

# List of Tables

# Part A

# Context and theoretical tools

# Introduction en français



## –1 Contexte et objectifs

Qu'est-ce que le savoir ?

Bien que ce travail ne vise pas à répondre à cette question, il peut être très instructif d'y réfléchir.

À la croisée de la physique théorique et de la psychologie, les experts semblent s'accorder sur le fait que le cerveau humain serait l'objet le plus complexe de l'univers, avec des centaines de milliards de neurones interconnectés. La programmation du cerveau à travers les générations guide les développements et l'adaptation des espèces pour leur survie. Celui des humains se concentre particulièrement sur la recherche et la reconnaissance de motifs, dans le but de faire des prédictions sur leur environnement. Cette faculté prédictive, habituellement appelée expérience ou intuition, se construit et s'affine à travers de nombreux essais et une adaptation aux résultats.

Cette méthodologie par essais et apprentissage est au cœur du développement de l'humanité et détermine encore aujourd'hui la majorité de nos choix et de nos actions. Elle montre cependant ses limites quand les expériences se font trop rares. La théorie se révèle alors un outil puissant et complémentaire dans l'appréhension du réel, permettant de prédire le résultat d'une expérience sans la réaliser. La modélisation intuitive empirique est alors remplacée par un modèle théorique, qui est en capacité de produire des prédictions arbitrairement complexes, précises et reproductibles. En suivant la méthode scientifique, ces modèles sont susceptibles d'être réfutés par les prochaines générations et remplacés par des théories plus représentatives et robustes. D'après la théorie du savoir de Popper, l'induction n'est en effet pas réalisable en pratique et ne devrait être considérée que comme une corroboration. Les théories les plus crédibles sont alors celles qui résistent le mieux à la falsification, c'est-à-dire dont aucun contre-exemple n'a encore pu être trouvé. Une théorie particulièrement robuste est la relativité générale d'Einstein, qui s'est révélée plus représentative que la mécanique Newtonienne en pratique, notamment pour prédire l'orbite de Mercure. La théorie de Newton n'en demeure pas moins très utile pour sa simplicité et sa grande

précision dans la majorité des situations. À l'instar de la modélisation empirique basée sur l'expérience, l'intérêt d'une théorie repose ainsi dans le compromis qu'elle propose entre précision et simplicité, ce qui n'est pas sans rappeler le rasoir d'Ockham.

L'une des principales avancées du siècle passé est l'émergence des machines de Turing, un modèle abstrait permettant l'utilisation d'algorithmes sur des machines de calcul. Alliées à la puissance de calcul toujours grandissante, décrite par la fameuse loi de Moore, la capacité d'analyse et de calcul de l'humanité est plus grande que jamais, ouvrant de nombreuses opportunités pour la recherche et les avancées technologiques.

Dans ce contexte, la simulation numérique a profondément impacté le monde de l'ingénierie. Ainsi, bien que les différences finies remontent au 17ème et 18ème siècle avec Newton et Euler, les ordinateurs actuels procurent un cadre idéal pour leur mise en application sur des problèmes complexes de modélisation 2D et 3D. Le 20ème siècle a vu l'émergence de nombreuses techniques pour la simulation numérique telles que la méthode aux éléments finis, qui permettent d'atteindre une précision toujours plus grande.

La modélisation et la prédiction ne sont cependant que rarement l'objectif final d'une étude, et sont souvent intégrées à un processus de prise de décision. En pratique, la qualité d'un produit, sa fiabilité et l'efficience des moyens de production sont ainsi des facteurs clés de succès pour toute entreprise. La majorité de l'ingénierie profite donc maintenant de l'optimisation par simulation, où le modèle numérique est supposé représentatif de la réalité et est optimisé à la place du système physique. Cela soulève bien-sûr la question centrale de la représentativité. Une grande quantité de travaux a été allouée à la question, et de nombreux schémas numériques sont disponibles, avec différents niveaux de simplification. Ce besoin de représentativité, malgré la grande complexité des systèmes étudiés, peut porter le temps de simulation à plusieurs heures, jours, ou même semaines. En réponse, un effort important a été investi dans le domaine de l'optimisation pour limiter au maximum le nombre d'itérations nécessaires pour atteindre le système optimal. L'objectif est donc de minimiser le nombre d'appels au code de simulation.

L'optimisation directe de la performance d'un système n'est cependant pas toujours suffisante lors de la phase de conception. Des choix suboptimaux sont en effet très souvent préférés afin d'assurer une certaine robustesse et fiabilité.

Ce besoin de robustesse et de fiabilité apparaît lorsque des décisions sont prises dans un environnement incertain, où certaines variables peuvent différer de leur valeur nominale. La définition de robustesse par le dictionnaire d'Oxford est "the ability to withstand or overcome adverse conditions or rigorous testing", c'est-à-dire la capacité de maintenir une bonne performance quelles que soient les conditions. La fiabilité est quant à elle définie comme "the quality of being trustworthy or of performing consistently well", ce qui signifie qu'un intérêt tout particulier est alors porté au contrôle du risque d'échec.

La prise en compte de ces incertitudes est une étape cruciale de conception. Elle consiste notamment à caractériser les différentes sources d'incertitudes et à les propager à travers un modèle d'intérêt. Ces analyses sont souvent regroupées dans sous le nom de quantification d'incertitude.

De même que pour la procédure d'optimisation, réaliser une propagation d'incertitude représente un coût de calcul significatif. Dans un contexte où chaque évaluation est une simulation numérique complète, le coût global de ces analyses statistiques devient rapidement ingérable.

Pour pallier ce problème, la plupart des méthodes de propagation d'incertitude reposent sur la construction d'un modèle de la simulation numérique, gratuit à évaluer. Cette technique s'appelle la métamodélisation, car il s'agit de construire un modèle du modèle numérique. Le point central de ces méthodes est le très faible coût de calcul associé à l'évaluation de ces modèles de substitution comparé à la simulation numérique complète. Cela permet de réaliser les nombreuses évaluations que necessitent des processus d'optimisation ou d'échantillonnage. De manière similaire au cerveau humain, la construction de ces modèles suit généralement une procédure d'essais et d'apprentissage. Il existe donc un très grand nombre de techniques dans la littérature visant à donner à la machine une "intuition" du résultat du code de simulation pour un certain jeu de paramètres d'entrée.

Finalement, dans un contexte incertain, optimiser un système dont les performances sont aléatoires devient une tâche plus ambigüe. Cette opération rentre alors dans le domaine de l'optimisation sous incertitude, qui est le sujet principal de cette thèse. Nous choisissons spécifiquement de minimiser (ou maximiser) des valeurs statistiques, qui peuvent être calculées par propagation d'incertitude. Cela requiert cependant la réalisation d'une propagation d'incertitude à chaque itération d'optimisation, augmentant considérablement le coût total de l'étude.

Bien que les termes de robustesse et de fiabilité n'ont pas encore atteint un réel consensus au sein de la communauté, nous proposons de suivre la tendance principale, qui appelle robustesse la prise en compte de paramètres incertain dans les objectifs d'optimisation (*e.g.* maximisation de la performance moyenne), et fiabilité la considération de ces incertitudes dans les contraintes d'optimisation (*e.g.* la probabilité d'échec doit être inférieure à 1%).

Dans ce travail, nous proposons plusieurs approches qui visent à réduire le coût de calcul global associé à cette procédure d'optimisation sous incertitude. Les stratégies présentées visent notamment à être applicables à des problèmes d'optimisation multi-objectif sous contraintes et à pouvoir traiter différentes métriques de robustesse et de fiabilité simultanément.

## –2 Plan et principales contributions

La structure du manuscrit est présentée ici, et les principales contributions de la thèse sont explicitées.

**Le Chapitre 1** donne une introduction aux méthodes à noyaux à travers la théorie des espaces de Hilbert à noyau reproduisant. Ce chapitre décrit notamment de manière intuitive les processus Gaussiens, qui sont utilisés à de multiples reprises dans le reste du manuscrit.

**Le Chapitre 2** introduit formellement le problème d'optimisation sous incertitudes dans un cadre multi-objectif sous contraintes. Les relations de domination de Pareto sont notamment définies dans un contexte probabiliste, où les valeurs des objectifs et contraintes sont considérées aléatoires. Ce cadre nous permet de proposer la première contribution de cette thèse, qui consiste à reformuler le problème d'optimisation sous incertitude comme un problème d'optimisation bruitée avec précision adaptable. En pratique, un premier algorithme nommé Measure Approximation with Tunable Accuracy (MATA) permet de réaliser l'optimisation bruitée en améliorant uniquement

la précision des résultats prometteurs. Cette approche est couplée à une stratégie d'assistance par métamodèle, souvent appelée Surrogate-Assisting (SA), permettant de réduire le coût global d'une optimisation grâce à l'apprentissage en parallèle d'un modèle de substitution. La stratégie globale proposée, nommée SAMATA, est au cœur des développements proposés dans les chapitres suivant.

**Le Chapitre 3** vise à mettre en application SAMATA en supposant spécifiquement que les erreurs d'approximation, représentées comme des variables aléatoires, suivent des distributions uniformes indépendantes les unes des autres. La première contribution de ce chapitre est la description formelle de l'algorithme SABBa, cas particulier de SAMATA sous l'hypothèse de distributions uniformes, et la démonstration de sa convergence sous certaines hypothèses. La seconde contribution concerne son application à l'optimisation sous incertitude et la proposition de formulations spécifiques pour le calcul des mesures de robustesse et fiabilité, ainsi que de l'erreur associée. L'application de SABBa sur différents cas-tests analytiques le révèle particulièrement robuste et parcimonieux par rapport aux approches plus conventionnelles.

**Le Chapitre 4** propose de ne plus supposer la forme de la distribution d'erreur *a priori*. Il est notamment proposé d'échantillonner les mesures de robustesse et fiabilité, avec des réalisations tirées de manière jointe ou disjointe. Un premier algorithme est développé dans le but de tirer des réalisations jointes à partir d'un processus Gaussien dans l'espace couplé. Un second algorithme est proposé dans le contexte de réalisations disjointes afin de construire un modèle de substitution non-Gaussien pour l'utilisation de la stratégie SA. L'approche SAMATA par échantillonnages est comparée à SABBa et à des méthodes plus classiques sur des cas-tests analytiques impliquant des incertitudes paramétriques et non-paramétriques. Elle se révèle particulièrement efficace et parcimonieuse sur l'ensemble des comparaisons réalisées.

**Le Chapitre 5** applique les deux algorithmes développés (SABBa et SAMATA par échantillonnages) sur trois applications d'ingénierie. Différentes mesures de robustesse et fiabilité sont traitées et les stratégies proposées donnent des résultats très satisfaisants sur chacune de ces applications.

**Le Chapitre 6** donne quelques conclusions et perspectives sur le travail réalisé. Les différentes contributions sont rappelées et leur limitations sont explicitées. Quelques idées d'amélioration sont finalement proposées en restant dans le cadre de SAMATA mais aussi en appliquant certains concepts de ce manuscrit dans un nouveau contexte.

# Introduction



## –1 Context and main objectives

What is knowledge?

Though this work does not intend to answer this question, much can be gained reasoning around it.

Theoretical physicists and psychologists agree that the human brain may very well be the most complicated thing in the known universe, with hundreds of billions of neurons, each of them connected to thousands of others. The programming of the brain guides the developments and adaptations for the survival of the species. The human brain is notably driven by the prime directive to seek patterns, which can then be exploited to make predictions of the world. Such ability usually referred to as experience or intuition, originates from many trials and errors.

This *try and learn* methodology is at the core of human development and determines most of our everyday actions, behavior and choices. However, this strategy shows its limit when experiments are difficult to perform or even unfeasible. In this context, theory complements practice very efficiently by giving a way to predict the outcome of an experiment without actually performing it. The empirical human modeling of reality is replaced with a theoretical model, that can grasp much more complex and fine behavior, with perfect reproducibility. Following the scientific method, this theoretical model may eventually be refuted by the next generations and replaced in turns with an even more representative theory. Indeed, following Popper's knowledge

theory, induction is unfeasible empirically and should only be considered as corroboration. Credible theories are then the ones that show high resilience to falsifiability, that is to says when no empirical counter-example has yet been found. A famous resilient theory is Einstein's general relativity that has shown more representative than Newton's mechanics in practice, notably on Mercury's orbit. Yet, the latter remains very attractive for its accuracy in most situation and its simplicity. Just like one's empirical modeling of reality, everything is a balance between accuracy and simplicity, under Ockham's razor.

One of the breakthroughs in the past century is the rise of Turing machines, which is an abstract model for computational machines that allow defining algorithms properly. Combined with the ever-growing computational power described by the famous Moore's law, the current computational power of our computers is unprecedented and opens many doors for research and technological breakthrough.

Notably, numerical simulation has profoundly transformed the engineering world. Although finite differences can be traced back to Newton and Euler in the 17th and 18th century, computers give an ideal framework for applying such methods to complex 2D and 3D systems. The 20th century saw the rise of multiple numerical simulation techniques, such as the finite element method, that allowed to make extremely accurate predictions.

Usually, modeling and predicting is not the sole purpose, but is part of a decision-making problem. Notably, in the industry, quality, cost-efficiency and resilience are critical factors of success. In most engineering companies arose the concept of Simulation-Based Optimization, where one assumes the numerical model to be representative of reality and optimizes it in place of the physical system. This concept raises the major issue of representativeness. An extensive amount of work has been allocated to such problematics, with numerical schemes that may feature a wide range of simplifying assumptions. Due to the high complexity of the systems of interest, accurate simulations can take hours, days or even weeks to complete. In the optimization literature, much effort has been devoted to limiting the number of trials needed to reach the optimal design. In essence, one aims to reduce the overall number of calls to the simulation code.

However, direct performance optimization is not the whole picture when designing complex systems. Sub-optimal behaviors are nearly always favored in nature for ensuring some level of robustness or reliability.

This necessity arises when making a decision in an uncertain environment, where variables differ from their usual nominal state. Robustness definition by Oxford dictionary is "the ability to withstand or overcome adverse conditions or rigorous testing", in other words, the ability to maintain a high-performance level even in non-nominal conditions. Reliability is defined as "the quality of being trustworthy or of performing consistently well", meaning that controlling the risk of failure associated with uncertain events is of significant importance.

Managing these uncertainties is a crucial step in any design process. It notably consists in characterizing the different sources of uncertainty (Uncertainty Characterization) and propagating them through a model of interest (Uncertainty Propagation). These analyses are usually gathered under the name of Uncertainty Quantification (UQ).

In a simulation-based setting, similarly to the optimization process, Uncertainty Propagation has a high computational cost. Accurately characterizing the output variability of a simulation code requires a high number of trials, each of which requiring a full simulation run. The overall cost of this statistical analysis can quickly become unmanageable.

This strong constraint may require the construction of a free-to-evaluate model of the numerical simulation. This approach is referred to as metamodeling, making a model of the model, or equivalently surrogate modeling. The main interest lies in the fact that these surrogates are much cheaper to evaluate than the full numerical simulation. It permits to carry out optimization or sampling procedures that require a lot of evaluations in a manageable time. The learning procedure of theses surrogates follows again a *try and learn* methodology. There is an extensive amount of techniques in the literature, which all aim at giving the machine an "intuition" of the simulation output for a given set of input variables.

In an uncertain context, optimizing a system becomes an unclear task as it requires to compare aleatory performances. This topic is referred to as Optimization Under Uncertainty and is the core subject of this thesis. It is notably chosen to perform this uncertainty-based optimization by maximizing (or minimizing) a given statistical value, that can be computed using Uncertainty Propagation techniques. This approach, however, requires to propagate the input uncertainties at each step of the optimization process, which further increases the associated computational cost.

Although the use of the words robustness and reliability seems not to have reached a true consensus yet, we propose to follow the standard practice that names robustness the consideration of uncertain parameters in the objectives (*e.g.* Maximizing the mean performance of a system), and reliability when considering them in the constraints (*e.g.* Probability of failure must be below 1%).

In this work, we propose several approaches that aim to lower the overall computational cost associated with such uncertainty-based optimization. Notably, the proposed strategies aim to apply to constrained multi-objective optimization problems and to treat simultaneously different metrics for robustness and reliability.

## –2 State of the art

An extensive amount of work has been devoted to the problem of Optimization Under Uncertainty in recent years. We propose here a non-exhaustive review of the associated literature, focusing first on the challenge of Uncertainty Quantification (UQ), with emphasis on low-cost Uncertainty Propagation (UP). The broad subject of optimization is then tackled, narrowed down to our constrained multi-objective setting, with specific attention on cost-efficient and derivative-free strategies. We finally deal with Optimization Under Uncertainty, divided into two sections: Reliability-Based Design Optimization (RBDO), that deals with probabilistic and worst-case feasibility constraints, and Robust Design Optimization (RDO), where the deterministic objectives are replaced with averaged or worst-case ones, possibly in a multi-objective context such as the classical Taguchi optimization.

## –2.1 Uncertainty Quantification

The field of Uncertainty Quantification (UQ) deals with characterizing how likely certain outcomes are, given that some aspects of the system at hand are unknown.

**Epistemic uncertainties and imprecise probability** Uncertainties can come from different sources and are usually considered as aleatoric, epistemic or a combination of the two. Epistemic uncertainty is also referred to as systematic uncertainty as it arises when some parameters are not stochastic by nature but not precisely known in practice.

Numerical approaches have been proposed in the literature to cope with epistemic uncertainties, in both UQ [Chen et al., 2013] and Optimization [Limbourg, 2005]. This type of uncertainty is often considered non-probabilistic, contrarily to aleatoric uncertainties. The field of imprecise probabilities can also be exploited to give imprecise aleatoric input distribution to tackle epistemic uncertainties. Numerical applications of such techniques can notably be found in [Schöbi, 2019] and [Adje et al., 2013].

**Probabilistic UQ methodologies** In this thesis, we focus on probabilistic uncertainty descriptions, which usually refer to aleatoric uncertainties. Note, however, that it is common practice to translate epistemic uncertainties in probabilistic terms. We focus on the following problem:

$$\text{Characterize the Probability Density Function } \phi_Y \text{ of } Y = f(\boldsymbol{\xi}) \tag{1}$$

where $f : \mathbb{R}^p \to \mathbb{R}^m$ is a real-valued function and $\boldsymbol{\xi} \in \Xi \subset \mathbb{R}^p$ is a random vector with known Probability Density Function (PDF) $\phi_{\boldsymbol{\xi}}$. We notably focus on the case where $f$ is costly to evaluate and considered as a black-box, meaning that no analytical formula is available and direct Monte Carlo Simulation (MCS) strategies are unfeasible.

The general Uncertainty Quantification (UQ) question is tackled in several reviews that compare different methodologies and software. Namely, [Marelli and Sudret, 2014] proposes the UQLab framework for performing all kind of statistical analysis, [Lin et al., 2012] compares the performance of numerous software on the most classical UQ tasks and [Rocquigny, 2009] gives a generic methodology for performing UQ in an industrial context.

**Surrogate models** Although direct statistical estimations, such as the method of moments [Wang and Peng, 2014], can be employed, it is common practice for performing Uncertainty Propagation (UP) to construct a surrogate model of the Quantities of Interest (QoI) to perform statistical analyses on this low-cost approximation of the simulation outputs. This approach is notably compared with the gradient-based Method of Moments in [Martinelli and Duvigneau, 2010]. An overview of the different surrogate modeling strategies for Uncertainty Propagation is proposed in [Sudret et al., 2017], with a focus on Low-Rank Approximation (LRA) and Polynomial Chaos Expansion (PCE). The latter is discussed in [Knio and Le Maitre, 2006] in the context of Computational Fluid Dynamics (CFD), and more deeply in [Le Maître and Knio, 2010] that gives theoretical background on spectral methods. An application of this approach is proposed in [Rivier et al., 2019] alongside a dimension reduction technique. However, lately, Kriging surrogate models, also called Gaussian Processes (GP), are increasingly

exploited in the UQ community and have met great success. Kriging-based methods can be exploited for performing forward [Razaaly et al., 2017] as well as backward [Cortesi, 2018] Uncertainty Propagation. The latter roughly consists in performing parameter calibration in the presence of probabilistic outputs. Bayesian calibration [Kennedy and O'Hagan, 2001] is a conventional approach within this field.

One of the most insightful and challenging Uncertainty Propagation problem is called reliability estimation, which often refers to low-probability or extreme quantile estimation. It can be accelerated through the use of advanced sampling techniques such as quasi MCS and Adaptive Importance Sampling (AIS). The former is exploited in [Xu et al., 2016] with unequal weights. The latter is used [Morio, 2012] in a non-parametric way, or [Pastel, 2012] in combination with the Minimum Volume Set (MVS) approach. Reliability estimation is also performed in [Picheny et al., 2010b] with a bootstrap strategy and in [Au and Beck, 2001] for high-dimensional problems by introducing Subset Simulation (SS), that is now a state-of-the-art low probability estimation technique based on Markov-Chain-Monte-Carlo (MCMC). Kriging-based approaches are also very popular for performing reliability estimation, notably since 2010, when [Echard et al., 2011] introduced the famous Adaptive-Kriging Monte Carlo Simulation (AK-MCS). This technique has notably been exploited in [Schöbi et al., 2016], which proposed to use Polynomial-Chaos-Kriging (PC-Kriging) surrogate models and to perform multi-point refinement, as well as in [Razaaly and Congedo, 2018], that performs well on multiple failure regions through Gaussian mixture-based Importance Sampling (IS). Further parsimony is allowed in the context of multiple failure modes with different computational costs in [Perrin, 2016] by ordering the performance functions within a nested Gaussian Process (GP) surrogate strategy. In 2010, the Sequential Uncertainty Reduction (SUR) formulation of the low-probability estimation was also introduced by [Bect et al., 2012], and led to numerous developments, *e.g.* regarding theoretical consistency proofs [Bect et al., 2019] or fast numerical computations [Chevalier et al., 2014a]. It has also been applied to real-world cases with great success [Jala, 2013]. This approach is notably available in the Kriginv R package [Chevalier et al., 2014b]. More generally, Kriging-based reliability estimation techniques have proven very cost-efficient in multiple applicative domains, such as structural analysis [Kaymaz, 2005] or space debris reentry prediction [Sanson, 2019]. Finally, some other machine learning techniques have been exploited to lower the overall cost, such as Support Vector Machines (SVM), either through regression [Dai et al., 2012] or classification [Basudhar et al., 2008, Li et al., 2006, Most, 2007], in combination with adaptive sampling strategies.

Another classical analysis, deeply related to Uncertainty Quantification, is Sensitivity Analysis (SA). Notably, global SA, reviewed in [Saltelli et al., 2008], is usually achieved with approaches similar to UP techniques, such as the use of Polynomial Chaos Expansion (PCE) [Sudret, 2008] or Gaussian Process (GP) surrogate models. The latter has been exploited both in the parametric [Marrel et al., 2012] and functional [Marrel et al., 2015] settings, with the use of Generalized Additive Models (GAM) and Proper Orthogonal Decomposition (POD). Approaches have also been proposed for dealing with correlated inputs [Da Veiga et al., 2009] and for replacing the classical variance-based sensitivity measures with more general dependence measures [Da Veiga, 2015]. For practical applications, efficient formulations for sensitivity estimation have been presented in [Da Veiga and Gamboa, 2013] within a metamodel-free framework, in [Chastaing et al., 2015] for dependent variables and in [Perrin and Defaux, 2019] in the context of reliability estimation.

**Remarks** *Note that several works have been dedicated to deriving error bounds on UQ analysis: p-boxes and robust Bayes techniques [Ferson et al., 2010]; error-based bounds on statistical moments [Barth, 2016]. Similarly, uncertainties associated with sensitivity indices are assessed in [Janon et al., 2014] with the use of reduced-basis surrogate models.*
*Not all Uncertainty Quantifications can be tackled through the use of surrogate modeling, notably when the problem features non-parametric uncertainties [Soize, 2005].*
*This thesis notably proposes error-based adaptive strategies, that can be applied in the context of non-parametric uncertainties.*

## –2.2 Optimization

As raised in the introduction, theoretical and numerical optimizations have enabled an extensive amount of developments and applications. Within these real-world problems, several objectives are often conflicting, raising the need for multi-objective optimization, that returns a set of compromises between the different objectives. These compromises represent the Pareto front of the optimization problem. Moreover, several constraints are usually imposed to ensure that the found optima are feasible.

The optimization problem classically writes as follows:

$$\begin{aligned} \text{minimize:} \quad & \boldsymbol{f}(\boldsymbol{x}), \\ \text{satisfying:} \quad & \boldsymbol{g}(\boldsymbol{x}) \leq \boldsymbol{0}, \\ \text{by changing:} \quad & \boldsymbol{x} \in \mathcal{X}, \end{aligned} \tag{2}$$

with $m_1$ objectives $\boldsymbol{f}(\boldsymbol{x}) \in \mathbb{R}^{m_1}$, $m_2$ constraints $\boldsymbol{g}(\boldsymbol{x}) \in \mathbb{R}^{m_2}$, and design vector $\boldsymbol{x} \in \mathcal{X} \subset \mathbb{R}^n$.

**Multi-objective optimization** In this context, the most popular derivative-free constrained multi-objective optimization strategies are Evolutionary Algorithms (EA), which regroup all population-based optimization metaheuristics that use some form of reproduction, recombination and mutation, alongside a selection scheme based on a chosen fitness function. Pushing the convergence through the number of generations should permit to enhance the value of the fitness function locally, while the mutation operator allows us to explore new areas. A comparison of multi-objective EA strategies is proposed in [Zitzler et al., 2000], in the classical bi- or tri-objective setting. When dealing with more objectives ($\geq 4$), one should refer to the many-objective optimization literature, notably reviewed in [Ishibuchi et al., 2008]. For more details, [Yang, 2010] gives a thorough description of both the theoretical foundations and practical metaheuristics for application to engineering test-cases. Finally, we highlight the Non-linear Optimization with Mesh-Adaptive Direct Search (NOMAD) [Le Digabel, 2011], that enjoys rigorous convergence theory and performs constrained multi-objective derivative-free non-linear optimization with great efficiency.

**Surrogate-Assisted optimization** The construction of a surrogate model of the fitness function in the design space is a common way to solve the optimization problem. Any optimization algorithm can then be employed on this surrogate with negligible cost. This approach is referred to as Surrogate-Assisted (SA) optimization. It has been

extensively exploited in EA setting, as reviewed in [Jin, 2011] and [Emmerich et al., 2002]. Namely, two main strategies emerged for Evolutionary Algorithms: evolution control and surrogate approach. These two techniques are presented and compared in [Buche et al., 2005], with the use of GP surrogate models. Evolution control, also referred to as pre-evaluations, only evaluates the fitness function on a controlled percentage of the population, which shows most promising. This strategy has been proposed in [Jin et al., 2002] and [Emmerich et al., 2006] coupled with GP surrogate models and in [Giotis et al., 2001] and [Giannakoglou, 2002] with Neural Networks (NN), notably Radial Basis Function Networks (RBFN), surrogate models. The so-called surrogate approach, more widely called Sequential Approximate Optimization (SAO), consists in running the whole optimization process on the approximate surrogate model, evaluating the fitness function at well-chosen locations, and refining the surrogate model with these data to rerun optimization.

SAO belongs to the broader field of adaptive refinements for surrogate models. It has been exploited in [Contal, 2016] with a Bayesian view and in [Buyukada, 2017] with classical Response Surface Methodology (RSM) techniques. For more details, [Nakayama et al., 2009] gives a thorough review of SAO theory and applications. Finally, [Torczon and Trosset, 1998] also adopts a Bayesian strategy and highlights that a well-chosen merit function must be chosen to perform adaptive refinement to reach the well-known exploration/exploitation balance. These concepts come from the Bayesian Optimization (BO) community, that is presented hereafter.

**Bayesian Optimization** Bayesian Optimization (BO) can be regarded as the intersection between Sequential Approximate Optimization and statistical modeling. Note that BO is usually performed with the most classical statistical model, namely Gaussian Processes, but can be adapted to any other random process, such as the recently proposed Student-t processes [Shah et al., 2014]. These approaches rely on well-chosen and easy-to-compute merit functions, usually called acquisition functions. In essence, the problem of optimizing a costly fitness function is solved by iteratively maximizing the current acquisition function for choosing the location of the next evaluation. [Brochu et al., 2010] gives a straightforward tutorial for understanding and applying such optimization technique. The most classical acquisition function is Expected Improvement, which adds the subsequent evaluation at the location that gives the most significant expected improvement of the optimal value. It was proposed by [Jones et al., 1998] under the name of Efficient Global Optimization (EGO). Many new acquisition functions have then been introduced, as presented in [Shahriari et al., 2015]. The amount of literature in this field is extensive, and we only highlight here a few of these works.

A parallel extension of the Upper Confidence Bound (UCB) acquisition function has been proposed in [Contal et al., 2013] with local Pure Exploration (PE). A strategy for managing hyperparameter updating has been introduced in [Gano et al., 2006] in the context of variable fidelity optimization. The hyperparameters are further studied in [Benassi et al., 2011], that proposes a fully Bayesian EI with hyperparameter prior distributions, also called hyperpriors. Practical computations associated with such an approach are presented in [Benassi, 2013] with a Sequential MCS technique. Recently, [Toscano-Palmerin and Frazier, 2018] has proposed a very efficient strategy for optimizing sums or integral quantities within the BO setting. Because GP naturally accounts for noisy training data, acquisition functions have been developed when dealing with noisy fitness functions, such as [Huang et al., 2006] with an aug-

mented EI criterion, compared in [Picheny et al., 2013] that benchmarks several other approaches such as EI with plugin and Approximate Knowledge Gradient (AKG). Note that insights can be found in [Wilson et al., 2018] on how to maximize these acquisition functions in practice. Most of the above strategies are unconstrained mono-objective optimization techniques, which can be quite limiting for real-world applications. Hence, developments have been proposed to account for expensive-to-evaluate constraints [Gardner et al., 2014], by constructing a GP model of the constraint function and evaluating the Probability of Feasibility (PF) associated to all locations in the input space. A significant amount of work has also been devoted to extending BO to the multi-objective setting. As presented in [Wagner et al., 2010], this is either performed through scalarization of the multi-objective problem, aggregation of the acquisition functions [Knowles, 2006], domination considerations or hypervolume computation. The latter is notably proposed in [Emmerich and Klinkenberg, 2008] through Expected Hypervolume Improvement (EHI) and in [Picheny, 2015] within the SUR setting. It has also been exploited in [Feliot et al., 2017, Feliot, 2017] with a suitable sequential MCS approximation of the EHI indicator and domination-based management of the constraints. Finally, a PF-based computation of the EHI acquisition function in a constrained setting is proposed in [Abdolshah et al., 2018].

Recently, informational approaches for Bayesian approaches have gained a lot of attention. This strategy was first proposed in [Villemonteix et al., 2009b, Villemonteix et al., 2009a, Villemonteix, 2008] that highlighted the associated computational difficulties. [Hennig and Schuler, 2012] proposed the term of Entropy Search (ES) that is now classically used. Intuitively, it consists in selecting the next evaluation location to maximize the information gained on the optimum location, that is to say by minimizing the associated entropy. Several works have tackled the computational issue, giving rise to the Predictive Entropy Search (PES) [Hernández-Lobato et al., 2014] and Max-value Entropy Search (MES) [Wang and Jegelka, 2017] that yield impressive performance. It is interesting to note that this informational approach naturally extends to the constrained [Hernández-Lobato et al., 2015, Hernández-Lobato et al., 2016b] and multi-objective [Hernández-Lobato et al., 2016a, Garrido-Merchán and Hernández-Lobato, 2019] settings. A strategy has also been introduced for performing multi-point enrichment in [Shah and Ghahramani, 2015].

**Noisy optimization** In this thesis, we focus on performing optimization in the presence of uncertainties. Such setting has received significant attention from the EA community, as can be seen in several reviews [Jin et al., 2005, Goh and Tan, 2009, Tan and Goh, 2008]. Notably, [Jin et al., 2005] proposed four types of uncertainties: (i) noise, which is an intrinsic bias on the evaluation of the fitness function, coming for example from imperfect sensors; (ii) robustness, which refers to specific parameters that may vary from their nominal value between the optimization procedure and the real-world application; (iii) fitness approximation, when the true fitness value is not computable or very expensive, and less accurate but cheaper representation is used instead; (iv) time-varying fitness function, meaning that the optimum changes through time and that the optimization algorithm should tackle this variability.

**Remark** *Note that in this work, although some techniques are proposed in the general noisy case, the aim is to tackle robustness through the use of surrogate-based fitness approximations, which accuracy can be efficiently estimated.*

A lot of work has been devoted to tackling noise within the optimization process. This field is usually referred to as Noisy Optimization (NO). It has been proposed in [Syberfeldt et al., 2010] to perform confidence-based dynamic resampling for tackling noise in Evolutionary Algorithm. Similarly, through an improved acceptance condition, Pure Random Search (PRS) can be extended to a noisy environment. Tackling noise is also performed in [Žilinskas, 2010] by replacing the RBF surrogate model with a noisy GP in the P-algorithm, and in [Gutjahr and Pflug, 1996] with a noise reduction assumption throughout the optimization process. More generally, some parsimonious techniques for Noisy Optimization rely on the assumption that there exist either different levels of accuracy for computing the Quantities of Interest or a way to refine a given accuracy so that noise can be reduced on-the-fly on the most promising designs. The former technique is called multi-fidelity or multi-level. Note that multi-level usually refers to different mesh refinements for numerical simulation, while multi-fidelity involves different modeling strategies. An extensive amount of work is devoted to these approaches, such as [Ho and Parpas, 2016] and [El-Beltagy and Keane, 1998] for the multi-level strategy and [March and Willcox, 2012, Zahir and Gao, 2013] for multi-fidelity approaches. As raised earlier, recent works have also proposed adaptive approaches for tackling problems where fidelity can be tuned online. This approach is particularly adapted when the fitness function is the limit of intermediate approximation of increasing fidelities, such as iterative solvers or Monte Carlo estimations. This concept has been proposed in [Picheny et al., 2010a] with online computation time allocation in a BO context.

In a multi-objective setting, the existence of such computations with variable fidelities raises the need for extending the classical Pareto dominance to an uncertain environment. Again, the EA community has quickly proposed techniques for problems featuring uniform noises [Teich, 2001] or Gaussian ones [Hughes, 2001]. The latter is extended in [Limbourg and Aponte, 2005], integrating a Bayesian treatment of the heteroscedastic noises associated with each individual. Similarly, in [Eskandari et al., 2007], the efficient FastPGA algorithm is extended to a stochastic context under the assumption of Gaussian noise. In [Fieldsend and Everson, 2005], a non-probabilistic approach is pursued and compares the so-called Imprecision-Propagating approach, featuring only partial ordering, with the Distribution-Assuming techniques. Interval and worst-case approaches are tackled in [Gong et al., 2010, Soares et al., 2009, Li et al., 2015], within the classical NSGA-II and MOEA frameworks. [Barrico and Antunes, 2006] proposes to add a degree of robustness, which permits to classify different Pareto fronts according to their robustness to input variability. Finally, [Khosravi et al., 2015, Khosravi et al., 2018, Khosravi et al., 2019] extends these computations to non-parametric noise distributions through the construction of histogram-based approximate densities, which permits fast computations of the probability of dominance. Another approach for lowering this computational cost is to perform First-Order Reliability Method (FORM) estimation of these probabilities, as presented in [Coelho, 2014]. With a more theoretical view, [Ide and Schöbel, 2016] and [Kuhn et al., 2016] review different concepts of Pareto efficiency and point out their relations.

**Remark** *A straightforward extension of the Pareto dominance rule to interval uncertainty in the presence of constraints is proposed in [Mlakar et al., 2014] under the name of Bounding-Boxes. This extension has been exploited in [Fusi and Congedo, 2016] within an adaptive strategy for performing multi-objective optimization on objective functions computed with tunable accuracy. This work has served as a basis for the present thesis.*

## –2.3 Reliability-Based Design Optimization (RBDO)

As raised earlier, reliability estimation is a key analysis throughout the designing process. Several techniques, either aleatoric or non-probabilistic permit to take into account input uncertainty and to propagate them efficiently through expensive black-box functions such as numerical simulation. By defining design parameters that can be controlled in addition to the random variables, Reliability-Based Design Optimization (RBDO) solves the optimization problem under the constraint that a prescribed reliability is satisfied. In this thesis, considering a constraint function $\boldsymbol{g}$ that depends both on design $\boldsymbol{x} \in \mathcal{X} \subset \mathbb{R}^n$ and uncertain parameters $\boldsymbol{\xi} \in \Xi \subset \mathbb{R}^p$, we write the RBDO problem with so-called reliability measures $\boldsymbol{\rho}_{\boldsymbol{g}}$ as follows:

$$\begin{aligned} \text{minimize:} \quad & \boldsymbol{f}(\boldsymbol{x}), \\ \text{satisfying:} \quad & \boldsymbol{\rho}_{\boldsymbol{g}}(\boldsymbol{x}) \leq \boldsymbol{0}, \\ \text{by changing:} \quad & \boldsymbol{x} \in \mathcal{X}, \end{aligned} \tag{3}$$

with $\boldsymbol{f} \in \mathbb{R}^{m_1}$ and $\boldsymbol{\rho}_{\boldsymbol{g}} \in \mathbb{R}^{m_2}$. As reviewed in the following, reliability measures $\boldsymbol{\rho}_{\boldsymbol{g}}$ usually refer to worst-case values $\rho_g(\boldsymbol{x}) = \max_{\xi}[g(\boldsymbol{x}, \boldsymbol{\xi})]$, where $\boldsymbol{\xi}$ might be either random or non-probabilistic. Other very popular reliability measures are quantiles $\rho_g(\boldsymbol{x}) = \mathrm{q}_{\xi}^{p}[g(\boldsymbol{x}, \boldsymbol{\xi})]$ and superquantiles $\rho_g(\boldsymbol{x}) = \mathbb{E}_{\xi}\big[g(\boldsymbol{x}, \boldsymbol{\xi}) \mid g(\boldsymbol{x}, \boldsymbol{\xi}) \geq \mathrm{q}_{\xi}^{p}[g(\boldsymbol{x}, \boldsymbol{\xi})]\big]$, computed on aleatory variables $\boldsymbol{\xi}$.

**Non-probabilistic uncertainties** Similarly to reliability estimation, RBDO problems can be tackled with hybrid random, interval and fuzzy uncertain parameters, as presented in [Wang et al., 2017]. Notably, interval uncertain inputs are addressed in [Hao et al., 2017] in the context of convex uncertainties, where the reliability estimation is carried out by determining the Most Probable Failure Point (MPFP). The same problem has been tackled in [Kang et al., 2011], which proposes the Concerned Performance Approach (CPA), that follows the same idea as the Performance Measure Approach (PMA), by finding the worst-case value of the limit-state function over the frontier of the convex shape. This CPA is performed in a further parsimonious way in [Meng et al., 2016] through the use of a Sequential Optimization Approach (SOA) that avoids playing the inner loop at all optimization iterations. [Luo et al., 2011] also exploits a sequential strategy in the context of hybrid random and interval parameters. Finally, [Li et al., 2017] computes the Reliability Measurement Index (RMI) as the failure area over the total one, calculated through interval perturbation in the context of linear systems of equations.

**First-Order approaches for random uncertainties** In the context of probabilistic uncertainties, the most classical method is the Reliability-Index (RI) approach, where a reliability index is computed using the Most Probable Failure Point (MPFP) search,

and the associated derivative is calculated. This technique is compared in [Lee et al., 2002] with the Performance Measure Approach (PMA), with the addition of a convergence smoothing technique. The PMA is further studied: [Youn et al., 2005] with a fast strategy in the neighborhood of the deterministic optimum, [Liang et al., 2008] which determines the angle of search at each iteration to avoid the inner PMA loop, and [Du et al., 2004] that performs Taguchi optimization, namely mean optimization and variance minimization. Also, [Youn et al., 2003] has introduced a Hybrid Mean Values (HMV) inner optimization technique, that merges the efficiency of the Advanced Mean Value (AMV) in convex settings and of the Conditioned Mean Value (CMV) in concave ones. In [Jensen et al., 2009], the RBDO problem is solved in high dimensional problems through an efficient line-search optimization algorithm based on polynomial approximations. Similarly, [Ching and Hsu, 2008] computes a factor of safety that gathers the information when many reliability constraints are to be satisfied simultaneously. The use of local approximations is exploited in [Cheng et al., 2006] with a Sequential Approximate Programming (SAP) technique, which corresponds to the SAO raised earlier. FORM has also been reviewed in combination with surrogate models in [Chu, 2016], and in [Lagaros et al., 2008], which focuses on Neural Networks (NN).

**Surrogate models** Recently, surrogate models have been exploited with great success to solve RBDO problems. Their use has been reviewed in [Bourinet, 2018] and [Moustapha and Sudret, 2019] with an emphasis on adaptive surrogate-modeling strategies, and in [Valdebenito and Schuëller, 2010b], that compares First-Order Reliability Method (FORM), decoupling approaches and surrogate-based strategies. Derivation of conservative reliability estimations throughout the optimization iteration has been studied in [Picheny, 2009], that aims to manage the accuracy of these approximate intermediate models. Some works have proposed the use of Neural Networks (NN) models [Papadrakakis and Lagaros, 2002] or PCE-Kriging representations [Coelho and Bouillard, 2011] for lowering the overall cost of RBDO. Local surrogate models have also been exploited in [Valdebenito and Schuëller, 2010a, Valdebenito and Schuëller, 2011], combined with line-search optimization, and in [Youn and Choi, 2004] for accelerating the HMV optimization of the inner PMA loop. [Dubourg et al., 2011, Dubourg, 2011] builds adaptive Kriging surrogate models in the so-called augmented space that encompasses the whole possible variations when dealing with uncertain design variables. The strategy is based on a K-means clustering of samples drawn from an accept-reject algorithm, with control on pseudo-confidence bounds of the failure probability. A GP surrogate model is also constructed in the augmented space in [Moustapha et al., 2016]. However, it proposes to build this model in a hyper-rectangle corresponding to the multi-dimensional product of the $\alpha$ to $1-\alpha$ quantiles intervals. Adaptive refinement is then performed similarly to the work of [Echard et al., 2011] with the AK-MCS strategy. Such adaptive refinement strategy is also exploited in [Razaaly, 2019] with further improved parsimony through the use of a Gaussian mixture Importance Sampling (IS) density for very low probability estimation, and the availability of unbiased estimation at the cost of few additional evaluations of the fitness function. An extreme quantile estimation technique is then derived, with parallel surrogate model refinement.

**Topology Optimization** Finally, note that RBDO techniques can be extended to Topology Optimization problems, with several formulations. FORM approaches mentioned above, computed with either RI or PMA method, are exploited in [Maute and Frangopol, 2003, Jung and Cho, 2004] for TO, and combined in [Zhao et al., 2016] with RSM techniques. Non-probabilistic topology RBDO is also achievable, as presented in [Kang and Luo, 2009] with convex uncertainty modeling and non-probabilistic Reliability Index (RI).

## –2.4 Robust Design Optimization (RDO)

Accounting for reliability-based probabilistic constraint is one of the main concerns in Optimization Under Uncertainty (OUU). The other pillar is the so-called Robust Design Optimization, which is classically defined as follows:

$$\begin{aligned} \text{minimize:} \quad & \boldsymbol{\rho}_f(\boldsymbol{x}), \\ \text{satisfying:} \quad & \boldsymbol{g}(\boldsymbol{x}) \leq \mathbf{0}, \\ \text{by changing:} \quad & \boldsymbol{x} \in \mathcal{X}, \end{aligned} \tag{4}$$

where $\boldsymbol{\rho}_f \in \mathbb{R}^{m_1}$, $\boldsymbol{g} \in \mathbb{R}^{m_2}$ and $\boldsymbol{x} \in \mathcal{X} \subset \mathbb{R}^n$. Robustness measures $\boldsymbol{\rho}_f$ most classically refer to either worst-case $\rho_f(\boldsymbol{x}) = \max_{\xi}[f(\boldsymbol{x}, \boldsymbol{\xi})]$ or average value $\rho_f(\boldsymbol{x}) = \mathbb{E}_{\xi}[f(\boldsymbol{x}, \boldsymbol{\xi})]$, where $\boldsymbol{\xi} \in \Xi \subset \mathbb{R}^p$ is a random vector. In addition, a popular strategy is Taguchi optimization, which is a bi-objective optimization problem that aims to optimize the average value and minimize the associated variance $\rho_f(\boldsymbol{x}) = \mathbb{V}_{\xi}[f(\boldsymbol{x}, \boldsymbol{\xi})]$.

In this section, focus will be on Robust Design Optimization (RDO) as well as its combination with Reliability-Based Design Optimization (RBDO), resulting in a complete Optimization Under Uncertainty (OUU) problem defined as follows:

$$\begin{aligned} \text{minimize:} \quad & \boldsymbol{\rho}_f(\boldsymbol{x}), \\ \text{satisfying:} \quad & \boldsymbol{\rho}_g(\boldsymbol{x}) \leq \mathbf{0}, \\ \text{by changing:} \quad & \boldsymbol{x} \in \mathcal{X}, \end{aligned} \tag{5}$$

**Robust counterpart** RDO originated with a worst-case paradigm. The associated theory of robust counterpart, applied to Linear, Conic, and Multi-Stage Optimization is notably reviewed in [Ben-Tal et al., 2009]. It has also been studied in [Kuroiwa and Lee, 2012] within a multi-objective setting. [Beyer and Sendhoff, 2007] proposes an overview of Robust Optimization (RO), starting with Taguchi's view and introducing robustness measures. Namely, it presents the worst-case measure, or robust counterpart, the expectancy measure and the probabilistic, or quantile, measure. Finally, [Schuëller and Jensen, 2008] presents these robustness measures and combines them with reliability measures. The resulting Reliability-Based Robust Design Optimization (RBRDO) will simply be referred to as Optimization Under Uncertainty (OUU) problem in the following. Note that the computation of robust counterpart in the context of Linear, Quadratic, Polyhedral, *etc* Programming has been extensively studied, notably in [Gabrel et al., 2014, Gorissen et al., 2015, Bertsimas et al., 2011, Klamroth et al., 2013].

**Anti-optimization** Under the robust counterpart paradigm, a significant amount of work has been dedicated to non-probabilistic approaches, also called anti-optimization. This name denotes the two-layer structure of the optimization problem that optimizes

the worst outcome. [Elishakoff et al., 1994] is an early application of such anti-optimization paradigm to structural mechanics. Notably, it takes advantage of the intrinsic linearity to perform efficient Sequential Linear Programming (SLP). The same problem is also tackled in [Lombardi, 1998] by cycling over the two anti-optimizations steps. Structural anti-optimization is also carried out in [McWilliam, 2001], that compares the Interval Perturbation (IP) method, which performs a linear approximation of the displacement with respect to the uncertain parameters, and the monotonic method, that only assumes that no turning point exists on the displacement surface, thus allowing the direct computation of the extrema. The Interval Perturbation (IP) method is also extended to wide parameter intervals with the Subinterval Perturbation (SP) method in [Qiu and Elishakoff, 1998]. These anti-optimization techniques have also been exploited in [Moens and Vandepitte, 2005] in the context of the Finite Element Method (FEM) through Interval and Fuzzy FEM techniques. As raised in the previous section, interval analysis can be compared with convex models, generally making use of ellipsoid shapes [Hu and Qiu, 2010, Qiu and Wang, 2005b]. Additionally, the use of surrogate models for alleviating the cost of the double optimization loop is proposed in [Wu et al., 2014] within the interval arithmetic setting. Finally, [Qiu and Wang, 2003, Qiu and Wang, 2005a, Zhu et al., 2017] compare the interval analysis setting with hybrid or only probabilistic techniques.

**Robust Evolutionary Algorithms** Robustness has also been extensively tackled in the EA setting to avoid convergence toward spurious optima. Notably, [Deb and Gupta, 2006] has proposed two types of robust solutions, under design uncertainty, through the use of expectation robustness measure or constrained variability computed with sampling techniques. In [Li et al., 2005], the authors propose to use a robustness index as an additional objective, which permits to draw all compromises between performance and robustness. A strategy is presented in [Basseur and Zitzler, 2006] for computing expectation robustness measures in Indicator-Based Multi-Objective Optimization, based on arbitrary probability distribution in the objective space. Finally, expectation and variance robustness measure are proposed and compared in [Jin and Sendhoff, 2003] with a Taguchi optimization strategy. Note that a decoupling approach has been introduced in [Meneghini et al., 2016] for performing worst-case optimization. This strategy is called co-evolutionary optimization and evolves the algorithm population iteratively based on performance and robustness.

**Sampling-based strategies** To limit the computational cost associated with RDO, and notably to alleviate the burden of the nested optimization loop, advanced sampling-based strategies have been proposed in the literature. For example, several advanced Monte Carlo methods are presented in [Hu et al., 2016] with application to aerospace vehicle design. Similarly, a Stochastic Subset Optimization (SSO) technique has been introduced in [Taflanidis and Beck, 2008] to iteratively sample the design space in the area that yields high plausibility of containing the optimum. Local optimization is then carried out through stochastic optimization, based on simultaneous perturbation derivative estimation and Common Random Number (CRN) robustness measure computation. The simultaneous perturbation stochastic optimization is also exploited in [Medina and Taflanidis, 2014], in combination with an adaptive Importance Sampling (IS) strategy. The latter permits to estimate robustness measure with a significant reduction of the estimator variance, by retaining information between optimization

steps. In [Congedo et al., 2013b], the Uncertainty Propagation (UP) step required for computing robustness measures is carried out through Simplex Stochastic Collocation (SSC), with a stopping criterion associated with the Nelder-Mead optimization process. Similarly, [Pisaroni et al., 2019] introduces a continuation Multi-Level Monte Carlo (MLMC) technique, that allows performing MLMC approximations of the chosen robustness measures up with a self-tuning of the accuracy according to the performance of the current design. An adaptive Monte Carlo computation of robustness measures, with increasing efficiency throughout the optimization process is also proposed in [Cheng et al., 2017].

**Remark** *The strategies presented above have been very influential in this thesis. Notably, the idea of linking the accuracy of the Uncertainty Propagation (UP) step (robustness and reliability measure approximation) with the optimization process is key in our work. Intuitively, starting with a rough estimation of the optimal area, which is then followed by an accurate computation of the optimal design.*

**Surrogate models** Surrogate models are very efficient for reaching great parsimony throughout the RDO process and have been compared with Taylor series expansion in [Duvigneau, 2007, Duvigneau et al., 2013] for aerodynamic shape robust optimization. The use of metamodels in OUU is notably reviewed in [Jin et al., 2003], with a comparison of Polynomial Regression (PR), Kriging and Radial Basis Functions (RBF). [Chen et al., 1996] was one of the first work to propose the use of Response Surface Methodology (RSM) for Taguchi Robust Design. Practically, this work considers both uncertain environmental parameters and variations in design variables. Polynomial Chaos Expansion (PCE) has notably been extensively exploited in for efficiently solving RDO problems. A double-loop RDO strategy is adopted in [Lucor et al., 2007] with the construction of a generalized PCE at each optimization iteration, embodied in a stochastic optimization strategy. A similar procedure is adopted in [Keshavarzzadeh et al., 2016], using both intrusive and non-intrusive formulations in the context of structural mechanics. The approach is extended to reliability measures through a relaxed Heavyside function, permitting analytical computation of the probability of failure. A PCE surrogate model is also constructed in [Xie et al., 2016] in the augmented space, and incorporated in a nested MCS optimization procedure. Finally, in [Dubreuil et al., 2018], robustness is achieved by the computation of an adapted Expected Improvement (EI) criterion for Random-Field (RF) Quantities of Interest (QoI). This RF is constructed with a Kriging surrogate model of PCE coefficients based on some UQ training data. A Karhunen-Loeve Expansion (KLE) representation can be used and truncated to manage the number of coefficients.

**Gaussian Processes** Kriging surrogate models have also shown particularly efficient in the context of uncertainty-based optimization. Taguchi optimization is notably carried out in [Dellino et al., 2009, Dellino et al., 2012] with Kriging surrogate models. Practically, these works compare the use of GP surrogate models applied to the QoI functions, the robustness and reliability measure, and both the QoI and the statistical measures. These 1-layer and 2-layer kriging models have also been studied in [Eldred et al., 2002] within the DAKOTA software. In the context of Taguchi optimization, a 2-layer kriging model is also exploited in [Lee and Park, 2006] with a simulated annealing optimization process. Note that in this work, the inner kriging layer is constructed

in the coupled-space, containing both design and uncertain parameters. Similarly, 2-layer kriging-based RDO is performed in [Bufi, 2016, Bufi and Cinnella, 2017] with Multi-Objective Expected Improvement (MOEI) infill strategy within the NSGA optimization process. The EI criterion is also exploited in [Guerra, 2016] in combination with the NSGA-II optimization process with a parallel infill criterion spread over the Pareto front. Between each refinement procedure, a full NSGA-II process is carried out on the current GP surrogate model. This Multi-Objective Efficient Global Optimization (MOEGO) has been applied to worst-case and superquantile RDO problems. A similar approach has been presented in [Baudoui, 2012] for Robust Design Optimization (RDO) based on the NSGA-II algorithm. In this work, the ranking procedure takes into account the variability associated with the GP model on which optimization is performed. Refinement is then performed through a multiple Maximum Mean Square Error (MMSE) procedure on several objective functions. This strategy is then extended to uncertainty-based optimization problem through linearization and Gaussian approximation of the robustness measures, combined with a reformulation of the infill criterion in the coupled space. This strategy, called PareBRO (Pareto Band Robust Optimization), is exploited in [Ammar, 2014] for performing OUU in nuclear applications. Linearization allows deriving a GP surrogate model of the robustness and reliability measures from a coupled-space kriging model. These are usually referred to as projected processes and are easy to compute for mean measure and can be approximated for variance measure. This technique has been exploited in [Janusevskis and Le Riche, 2013] with a specific infill strategy, that determines the new design parameters first through EI criterion and computes the best uncertain location by minimizing the output measure variance. A similar approach has been proposed in [Williams et al., 2000] with an MCS-computed noisy EI criterion that aims to apply a fully Bayesian infill strategy. This technique is also employed in [Da Veiga and Delbos, 2013] for multi-objective problems in combination with several adapted Bayesian Optimization techniques. To avoid constructing a GP surrogate model on the whole coupled space, [Marzat et al., 2013] proposes to solve worst-case RBO problems through the relaxation of the minimax formulation, by iteratively minimizing the robustness measure and maximizing the QoI over the uncertain parameters. This procedure is performed through a nested EI strategy, with kriging surrogate in separated design and uncertain spaces. A Sequential Approximate Optimization (SAO) strategy is proposed in [Zhang et al., 2017] for performing optimization on mean robustness measure. To this extent, the kriging surrogate model is constructed in the augmented space, but only in a small neighborhood of the design space. When the gradient-based optimization process reaches the neighborhood boundary, the window is translated if the accuracy criterion is met. When only design parameters are subject to uncertainty, and the robustness measure at hand is worst-case, an adapted EI criterion has been derived in [ur Rehman et al., 2014]. Finally, [Le Riche et al., 2009] tackles 95%-quantiles as robustness measure and computes them with tunable accuracy within an EI-based optimization procedure. RDO is also solved in the context of non-parametric uncertainties in [Sabater et al., 2020] through the use of a Bayesian treatment of the classical quantile regression surrogate. Very recently, another robustness measure that depicts the "flatness" of the solution around the optimum has been proposed in [Ribaud et al., 2020], which employs Gradient Enhanced Kriging (GEK) for efficient computation of this objective. The robust optimization problem is then solved in a multi-objective context using the NSGA-II evolutionary algorithm.

**Multidisciplinary problems** When the black-box Quantities of Interest (QoI) consists in a set of solvers chained together, specific Uncertainty Quantification (UQ) techniques have been proposed in the literature. As a consequence, Optimization Under Uncertainty problems applied in such context can benefit from these techniques. This possibility is notably discussed in [Brevault, 2015] that gives a broad overview of the subject and [Balesdent et al., 2016] that focuses on epistemic uncertainties. A dedicated algorithm named LOcal Uncertainty Processing (LOUP) has been proposed in [Baudoui et al., 2012], through forward Uncertainty Propagation (UP) only to downstream solvers and a posteriori validation.

**Topology Optimization** Similarly to RBDO, Robust Topology Optimization (RTO) has also been extensively studied in the literature. Notably, interval uncertainty and worst-case RDO are tackled with the perturbation method [Asadpoure et al., 2011], with Chebyshev interval analysis [Wu et al., 2016] and by reformulating RDO as a Semi-Definite Optimization (SDO) optimization [Thore et al., 2017]. Probabilistic robustness measures have also been applied to the context of Topology Optimization (TO), such as in [Amir et al., 2012], where a pre-conditioned gradient reanalysis permits to tackle both worst-case and Taguchi-like robustness measures. Taguchi optimization is also performed in [Chen et al., 2016] through a linearized structural formulation under interval uncertainty, and in [Guest and Igusa, 2008], that deals with uncertainties in node location through an equivalent formulation with uncertain loads. Several works have tackled geometrical uncertainties within a level-set framework, which permits to move the structure boundaries naturally. This approach has been exploited in [Wu et al., 2017], under hybrid interval and random uncertainties, with the use of Polynomial Chaos Chebyshev Intervals. Similarly, a PCE-based random density field is used in [Keshavarzzadeh et al., 2017], and a Stochastic Collocation FEM based on KLE uncertainty modeling is proposed in [Martinez-Frutos et al., 2016]. Polynomial Chaos Expansion (PCE) has also been exploited in an intrusive manner in [Tootkaboni et al., 2012] with the classical Stochastic Galerkin setting.

**Innovative formulations** Note that some works propose innovative techniques with an original view and formulation for Optimization Under Uncertainty (OUU). Namely, [Seshadri et al., 2016] suggests scalarizing multi-objective RDO such as Taguchi formulation through a density matching strategy. Taguchi optimization is also tackled in [Tang and Périaux, 2012], that exploits adjoint solver for obtaining gradients, and iteratively optimize one of the objectives until a fixed point is attained, in a Nash equilibrium setting. Finally, [Trappler et al., 2020] introduces a new robustness measure similar to quantiles and Values-at-Risk in the name of a family of Relative Regret Estimators, for which the acceptable probability to worsen the optimal value can be prescribed.

# –3 Industrial context and contributions

## –3.1 Context

This thesis has been carried out at ArianeGroup, European leader in access to space through the reliable Ariane 5 vehicle, that has performed over 100 successful launches, and the oncoming Ariane 6, that aims for very high flexibility both in its payload and its

target orbit. ArianeGroup designs propulsive systems for civil and strategic purposes, which may involve liquid and solid propellant. The lattest is mostly tackled in the Aquitaine sites, and notably at Le Haillan center where this thesis has been completed. It specializes in the conception of solid propulsive systems, so-called boosters, from the igniter to the nozzle and rear sets. These boosters are strapped on the launch vehicle and deliver a massive thrust from the ignition onward. They only burn for a few minutes and aim to tear the vehicle off the ground at the beginning of the flight. The nozzle and rear sets are also produced on-site.

Of course, real-world trials are costly and should be avoided as much as possible. To this extent, the Analysis department, in which this thesis was carried out, gathers vast expertise on numerical simulation. Specifically, fine modelisation of aerodynamical, thermochemical, thermomechanical or mechanical phenomena is required for performing simulation-based engineering and carrying out reliability analyses in collaboration with the different Engineering departments.

As raised above, there is a need for both reliability and optimality. In its recent history, Ariane has shown steady and robust reliability levels, with an unprecedented number of consecutive successes. Given the price associated with each launch, a failure would indeed have dreadful repercussions on the company. However, performance and reliability are often conflicting, and a proper compromise must be chosen. Overconservativeness may ensure extreme reliability, but impacts severely the overall efficiency of the vehicle.

In this context, a precise uncertainty quantification must be carried out, and a reliability level must be imposed. Similarly, to ensure good performance under uncertain conditions, adapted robustness measures can be chosen and optimized in place of the original nominal objective function.

In practice, such an uncertainty-based optimization strategy could be applied to ensure that, in an uncertain environment, the highest pressure inside the booster remains under a given threshold while maximizing the average overall performance. Similarly, a security coefficient is usually associated with each design, denoting whether the material capacity is exceeded at some location during the numerical simulation. Ensuring a probability level for this coefficient to remain in the admissible domain while optimizing the geometry of the rear set is another possible application of Optimization Under Uncertainty. Finally, with a more general view, trade-offs between cost and performance of the systems at hand should be compared in a multi-objective setting, either with average or quantile robustness measures. Note that the latter corresponds roughly to a relaxed worst-case strategy.

In summary, here are the main objectives of the methods developed in present work:

- Be capable of tackling a general measure-based formulation for Optimization Under Uncertainty with most classical statistics (mean, variance, quantile, worst-case).
- Deal with multiple objectives and constraints.
- Achieve very high parsimony, to perform uncertainty-based optimization on computationally expensive numerical models.
- Limit the computational impact of the proposed algorithm, to also efficiently tackle inexpensive functions.

- Be flexible concerning the optimization and Uncertainty Propagation (UP) procedures and propose an efficient coupling strategy.

The last objective permits to naturally benefit from future improvement in the optimization and UQ community, exploit case-specific optimization strategies and tackle atypical settings such as non-parametric uncertainties.

## –3.2 Contributions

In light of the above objectives, the main contributions of this thesis to the field of Optimization Under Uncertainty (OUU) are highlighted here.

### –3.2.1 Tunable accuracy formulation of OUU problems

Robustness and reliability measures are integrands that are approximated throughout the optimization process, with refinable accuracy. We assume that these measures can be regarded as random variables at a given design, and more generally as random fields over the design space. The associated distributions must then be estimated.

**Probabilistic multi-objective setting** We introduce a probabilistic framework for constrained multi-objective optimization problems. Such setting has been exploited in [Khosravi et al., 2018, Khosravi et al., 2019, Coelho, 2014], which proposed probabilistic Pareto dominance rules without any assumption on the distribution shapes. We extend these concepts, notably to dependent random variables.

Besides, a quantitative indicator is proposed for ranking Pareto-optimal designs. This indicator is called Pareto-Optimal Probability (POP) and naturally tackles dependent variables with non-parametric distributions. Similarly to the PUI in [Selçuklu et al., 2020], several metrics are compared, in terms of interpretability and computational burden.

**Tunable accuracy optimization** An efficient strategy is then proposed for solving optimization problems where objectives and constraints are computed with tunable accuracy. It consists in coupling two primary techniques. First, a Measure Approximation with Tunable Accuracy (MATA) approach, such as in [Fusi and Congedo, 2016], that only refines promising individuals up to a given threshold. Second, a Surrogate-Assisting (SA) strategy, that constructs a surrogate model of the objective and constraint functions in the design space to bypass further evaluations when the surrogate is sufficiently accurate.

Both these techniques are proposed in a very general form. We investigate in the following the influence of the error distribution on the Surrogate-Assisted Measure Approximation with Tunable Accuracy (SAMATA) strategy and we propose efficient and suitable algorithms.

### –3.2.2 Bounding-Box approximations

We first assume the error to be uniformly distributed. The estimation error can thus be regarded as an interval (in one-dimensional problems) or a product of intervals (in multi-dimensional problems).

The restriction of the above tunable accuracy setting to such independent uniform distributions has notably been tackled in [Mlakar et al., 2014] and [Fusi and Congedo, 2016], with the name of Bounding-Box approach (BBa). The coupling with the Surrogate-Assisting (SA) strategy will be referred to as SABBa.

**Theoretical considerations** Bounding-Box Pareto dominance is shown to be a particular case of the general probabilistic Pareto dominance rule. Explicit formulations are proposed and notably different from the literature on the management of unfeasible individuals.

Then, under some conservativeness assumptions, we demonstrate the convergence of the proposed approach toward the Pareto-optimal designs. To this extent, the specific set sequence of Pareto-optimal individuals is introduced and shown to yield robust estimations of the Pareto front.

**Application to noisy optimization** We illustrate the performance of SABBa on noisy optimization problems with tunable fidelity. Specifically, we illustrate some expected trends concerning tuning parameters on some analytical test-cases, with specific optimization and surrogate modeling strategies.

**Application to Optimization Under Uncertainty** Finally, efficient formulations are proposed for applying SABBa to OUU problems. Both the SA model and the BB computation and refinement are based on Gaussian Process (GP) surrogate models. Similarly to [Mlakar et al., 2014] and [Baudoui, 2012], a $\pm 3\sigma$ paradigm is exploited to determine the bounds associated with the performance of each design. Refinement is carried out on multiple robustness and reliability measures simultaneously through an adaptive weighting of statistics-dependent criteria.

This strategy is applied to analytical and engineering-based test-cases, with excellent efficiency and parsimony compared to classical approaches. It notably shows much higher robustness in its convergence rate compared to non-adaptive strategies, while permitting a flexible coupling with any optimization and UQ processes of choice.

### –3.2.3 Sampling-based approximations

Then, we propose to compute non-parametric approximations of the error distributions with a sampling-based technique. For a given design, the robustness and reliability measures are represented as a set of realizations, that may exhibit complex shapes and some dependency, contrarily to all Distribution-Assuming (DA) strategies, such as projected processes [Da Veiga and Delbos, 2013, Baudoui, 2012, Janusevskis and Le Riche, 2013, Williams et al., 2000]. These realizations usually reveal more representative of the true measure distributions, and they can be jointly drawn between different designs if a coupled-space surrogate model is available, which has not been proposed in other probabilistic dominance frameworks [Khosravi et al., 2018, Khosravi et al., 2019, Coelho, 2014].

**Low-cost strategy for drawing joint realizations** Issues associated with drawing such joint realizations are highlighted, and an adaptive algorithm is proposed for keeping the associated cost manageable. This strategy show similarities with Nyström approximation for Karhunen-Loeve Expansion (KLE) but allows for intuitive control

of the overall accuracy and adaptive sampling. A Surrogate-Assisting (SA) model that takes advantage of these joint realizations is then introduced to apply the SAMATA strategy. Quantitative comparisons with SABBa reveal a drastic increase in the overall parsimony and show that the non-parametric characteristic permits better representativeness.

**Surrogate-Assisting model for disjoint realizations** When coupled-space surrogate models cannot be constructed, the above strategy is not applicable. In this case, we propose a SA model based on Kernel Density Estimation (KDE) methods. From only disjoint realizations, it aims at recovering a non-parametric random field over the whole design space. This construction is achieved through a rejection sampling technique that relapses to heteroscedastic GP in a Gaussian context. This approach shows both great parsimony and high efficiency in capturing complex distribution shapes.

## –4 Outline of this work

The manuscript is organized as follows.

**Chapter 1** gives a brief introduction to kernel methods with the Reproducible Kernel Hilbert Space (RKHS) theory. It notably aims to present Gaussian Processes intuitively so that the reader can fully understand their use in the following chapters.

**Chapter 2** introduces the Optimization Under Uncertainty (OUU) problem in a constrained multi-objective context. The notion of Pareto efficiency is formally defined in a probabilistic framework. Finally, the Surrogate-Assisted Measure Approximation with Tunable Accuracy (SAMATA) strategy is illustrated in its most general formulation.

**Chapter 3** deals with the specific assumption of independent uniform distribution of the error. We illustrate how the concept of Pareto dominance is simplified in this case and make the formulation of SABBa explicit. Theoretical convergence properties can then be demonstrated under some assumptions. SABBa is then applied to arbitrarily noised test-case, and the impact of some user-defined parameters is illustrated. Secondly, specific formulations are proposed for using SABBa on Optimization Under Uncertainty (OUU) problems. Analytical test-cases reveal very high robustness and parsimony for the proposed approach compared to more conventional strategies.

**Chapter 4** proposes sampling-based formulations for applying SAMATA with joint or disjoint measure realizations. The proposed algorithms for drawing joint measure realizations from a coupled-space Gaussian Process and reconstructing a non-parametric random field from disjoint measure realizations are presented in details and illustrated on simple test-cases. Sampling-based SAMATA is then quantitatively compared to SABBa and classical approaches on analytical applications featuring parametric and non-parametric uncertainties.

**Chapter 5** presents three engineering-based OUU problems solved with SABBa and sampling-based SAMATA. Several robustness and reliability measures are tackled throughout these test-cases, namely mean, quantile and worst-case.

**Chapter 6** gives some conclusions and perspectives of the present work. The contributions are highlighted, and the limitations of the proposed strategies are discussed. Some ideas are finally proposed for improving current algorithms and using some concepts introduced in this thesis in another context.

CHAPTER *I*

# Introduction to kernel methods

**Overview** *In this chapter, we aim to give an overview of kernel methods such as Gaussian Processes (GP) and Kernel Mean Embedding (KME). The former is of particular importance for constructing predictive models on scalar training data and are notably exploited in Chapters III and IV. The latter allows for generalizing these predictive models to probability densities as input data.*
*We propose an intuitive introduction to GP starting from linear and feature regression, and combining them with Bayesian regression. The Reproducing Kernel Hilbert Space (RKHS) theory is then quickly introduced and extended to the recently developed Kernel Mean Embeddings (KME).*

**Outline** *In section I–1, we introduce Gaussian Processes (GP) and highlight their links with other classical statistical models. The Bayesian regression on features is first developed, and a functional view of the random process is then drawn.*
*Section I–2 gives a quick overview of the RKHS theory. Kernel Mean Embeddings (KME) and particularly Conditional Mean Embeddings (CME) are then presented for their ability to extend kernel methods to distributions.*

## I–1 An intuitive introduction to Gaussian processes

In this section, several statistical models are presented for interpreting a cloud of points. The purpose of each of these data-driven approaches is made explicit and ultimately lead to the introduction of Gaussian Processes. Although we favor a weight-space view, we also give a functional perspective. Both of these are notably presented in [Williams and Rasmussen, 2006], in a very didactic way.

Throughout this section, we deal with multi-dimensional training data in $\mathbb{R}^d$ at $n$ training locations, denoted $\{\boldsymbol{x}_i\}_{i=1}^n$, and 1D training outputs $\{y_i\}_{i=1}^n$. We notably aim to construct an approximate representation of the output variable at any location $\boldsymbol{x}_*$ of interest.

### I–1.1 The basics: Linear Regression

Linear Regression aims at finding a linear tendency within training data. For any input dimension $d$, linear relationships between an input parameter $x_* \in \mathbb{R}^d$ and an output value $y_* \in \mathbb{R}$ formally writes:

$$y_* = \sum_{j=1}^{d} w_j x_{*_j} + w_0, \tag{I.1}$$

which can be simplified to:

$$y_* = \boldsymbol{x}_*^\intercal \boldsymbol{w}, \tag{I.2}$$

where $\boldsymbol{w} = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_d \end{bmatrix}$ and $\boldsymbol{x}_* = \begin{bmatrix} 1 \\ x_{*_1} \\ \vdots \\ x_{*_d} \end{bmatrix}$.

Thus, performing linear regression consists in determining the weights $\boldsymbol{w}$. The available information is the set of training data and can be exploited for finding the most appropriate weights. To this extent, training locations are gathered in the so-called design matrix $X^\intercal$ as follows:

$$X^\intercal = \begin{bmatrix} \boldsymbol{x}_1^\intercal \\ \boldsymbol{x}_2^\intercal \\ \vdots \\ \boldsymbol{x}_n^\intercal \end{bmatrix}, \tag{I.3}$$

which features $n$ rows and $d+1$ columns. Note that each training element $\boldsymbol{x}_i$ above starts with an element of value 1.

Based on the $n$ training values $\boldsymbol{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}$, the weights can be found by solving the linear system:

$$\boldsymbol{y} = X^\intercal \boldsymbol{w}. \tag{I.4}$$

Three situations can occur at this point:

- $n < d+1$, the system is underdetermined, and an infinite number of solutions exist. For example, in a monodimensional problem, an infinite number of straight lines passes through a given training point.

- $n = d+1$, the system is determined, one unique solution exists. Similarly, only one straight line passes through two distinct training points. If training points are not distinct, the system relapses to an underdetermined one.

- $n > d+1$, the system is overdetermined, no solution exists a priori, unless the matrix rank is actually $d+1$. No straight line can pass through three non-aligned points.

In the statistical and data-driven community, the focus is usually put on the overdetermined situation. One aims to draw a model of low complexity from a sufficient number of data, thus performing a regression.

Intuitively, although no linear model could be able to reproduce the training data, one may try to fit them as close as possible. With $\hat{\boldsymbol{y}}$ a linear model, linear regression is performed by considering the following formulation:

$$\boldsymbol{y} = \hat{\boldsymbol{y}} + \boldsymbol{\varepsilon} = X^\intercal \boldsymbol{w} + \boldsymbol{\varepsilon}, \tag{I.5}$$

and by minimizing the regression error vector $\boldsymbol{\varepsilon}$.

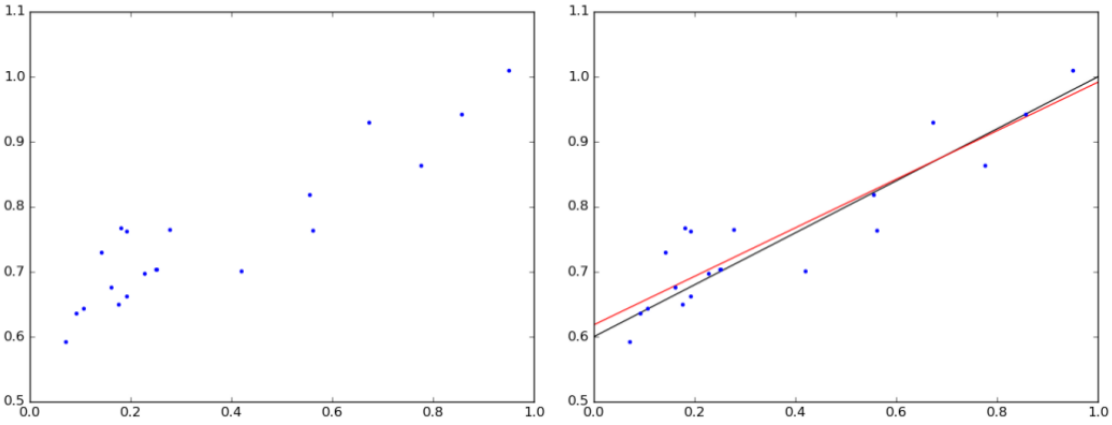
Notably, the $\ell 2$ norm $\|\boldsymbol{\varepsilon}\|_2$ is minimized by solving the so-called normal equation:

$$\begin{aligned} XX^\intercal \boldsymbol{w} &= X\boldsymbol{y}, \\ \boldsymbol{w} &= (XX^\intercal)^{-1} X\boldsymbol{y}. \end{aligned} \tag{I.6}$$

This very classical technique is referred too as the Least-Square Method (LSM). A key computational point of this technique is that it requires to inverse the $(d+1) \times (d+1)$ matrix $XX^\intercal$.

For illustration, Linear Regression is performed on the data points represented in Figure I.1(a). These training points are drawn from the noisy function $x_* \mapsto y_* + \varepsilon$, where $y_* = 0.6 + 0.4x_*$ is the true noiseless trend and $\varepsilon$ is a small centered Gaussian noise.

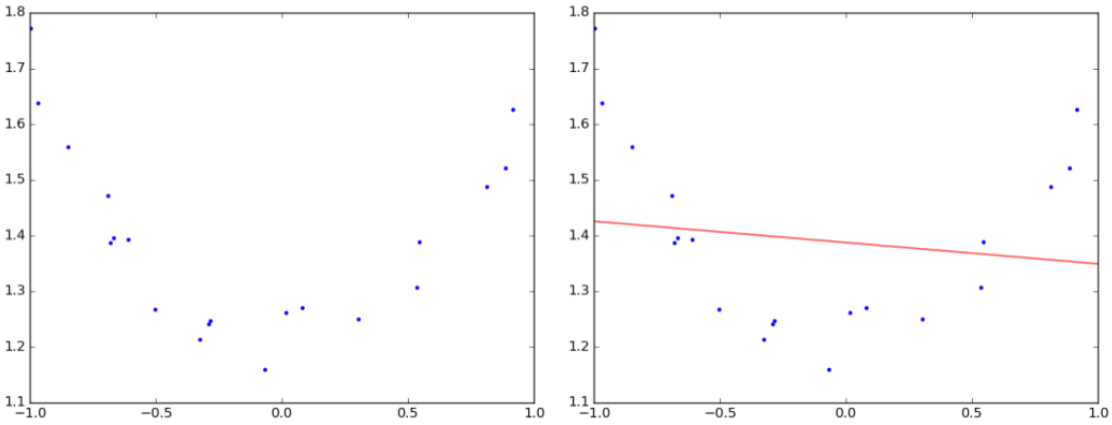
The Least-Square fit $\hat{y}_* = \begin{bmatrix} 1 \\ x_* \end{bmatrix}^\intercal \begin{bmatrix} 0.618 \\ 0.373 \end{bmatrix}$ gives a good approximation of the underlying noiseless function $y_* = 0.6 + 0.4x_*$. This regression is drawn with a red line in Figure I.1(b), alongside the true noiseless function drawn in black.



(a) (b)

Figure I.1 (a) Training data. (b) True noiseless function in black and Linear Regression in red.

However, Linear Regression only aims to retrieve the linear relationship between the input and output variables. Although it permits to sense the overall impact of each input parameter, it usually shows unable to yield an accurate model of more complex data. Such limitation is notably represented in Figure I.2, where Linear Regression fails at grasping a quadratic relationship.



(a) (b)

Figure I.2 Limitation of Linear Regression. (a) Training data. (b) Linear model in red.

## I–1.2 Adding complexity: Feature Regression

The most straightforward approach to tackle non-linearity is to incorporate features within the regression problem. In practice, this permits the regression model to accurately represent functions of arbitrary complexity, given that the appropriate features are chosen.

In practice, Linear Regression aims to represent the output as a linear combination of the constant function $f(\boldsymbol{x}_*) = 1$ and the $d$ projection functions $f(\boldsymbol{x}_*) = x_{*_i}$. Although natural, this choice is quite arbitrary and any basis function could be used instead. Hence, Feature Regression is an extension of Linear Regression to non-linear basis functions. Note that the application of feature regression is strictly similar to linear regression, as depicted in the following formulas. The $m$ features are denoted $\{\phi_i\}_{i=1}^m$ and the design matrix writes:

$$\Phi^\intercal = \begin{bmatrix} \boldsymbol{\phi}(\boldsymbol{x}_1)^\intercal \\ \boldsymbol{\phi}(\boldsymbol{x}_2)^\intercal \\ \vdots \\ \boldsymbol{\phi}(\boldsymbol{x}_n)^\intercal \end{bmatrix}. \tag{I.7}$$

Because each row contains $m$ elements, with $\boldsymbol{\phi}(\boldsymbol{x}_i) = \begin{bmatrix} \phi_1(\boldsymbol{x}_i) \\ \phi_2(\boldsymbol{x}_i) \\ \vdots \\ \phi_m(\boldsymbol{x}_i) \end{bmatrix}$, the design matrix $\Phi^\intercal$ is now of size $n \times m$.

**Remark** *As expected, Feature Regression relapses to Linear Regression by fixing $m = d+1$, $\phi_1(\boldsymbol{x}_*) = 1$ and $\phi_{i+1}(\boldsymbol{x}_*) = x_{*_i}$ for $i \in [\![1, d]\!]$.*

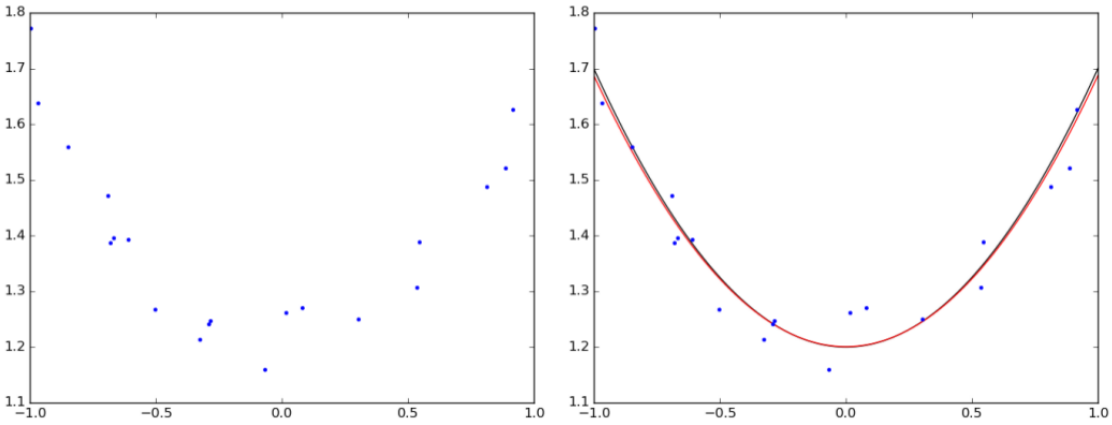
Optimal weights can be derived from the normal equations as before:

$$\begin{aligned} \Phi\Phi^\intercal \boldsymbol{w} &= \Phi \boldsymbol{y}, \\ \boldsymbol{w} &= (\Phi\Phi^\intercal)^{-1}\Phi\boldsymbol{y}, \end{aligned} \tag{I.8}$$

and requires the inversion of a $m \times m$ matrix.

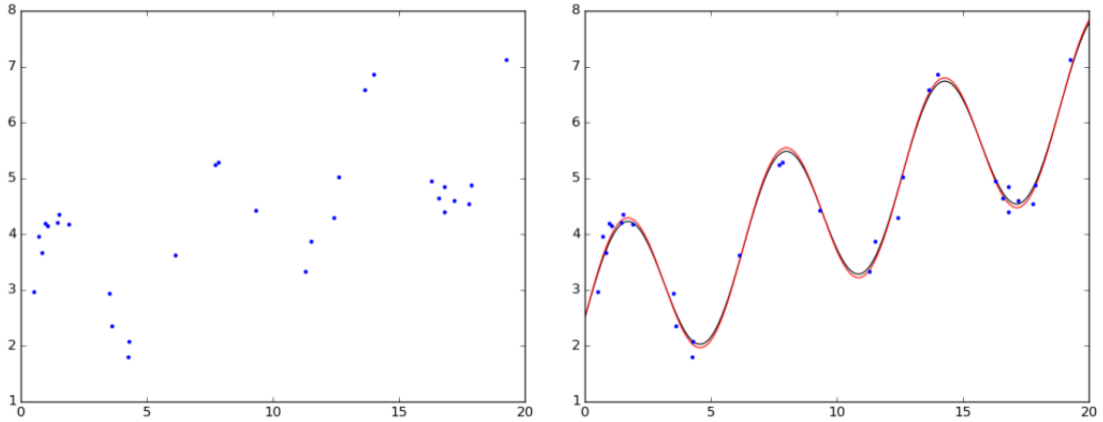
Feature Regression is pictured in Figure I.3 on a quadratic test-case, with excellent agreement between the true noiseless function $y_* = 1.2 + 0.5x_*^2$ and the Least-Square fit $\hat{y}_* = \begin{bmatrix} 1 \\ x_*^2 \end{bmatrix}^\intercal \begin{bmatrix} 1.2 \\ 0.487 \end{bmatrix}$. Similarly, in Figure I.4, the combination of linear and sine functions $y_* = 2.5 + 0.2x_* + 1.4\sin(x_*)$ is well retrieved with $\hat{y}_* = \begin{bmatrix} 1 \\ x_* \\ \sin(x_*) \end{bmatrix}^\intercal \begin{bmatrix} 2.502 \\ 0.2 \\ 1.463 \end{bmatrix}$.

**Remark** *Feature Regression can be viewed in two equally valid manners. As raised above, it can be seen as a linear combination of arbitrary basis functions $\phi_i$, with weights optimized on the available training data. Equivalently, it can be regarded as a simple Linear Regression performed in the so-called feature space. The latter is of dimension m, where each one corresponds to a basis function.*
*In essence, one may either fit a quadratic curve with a Least-Square fit of a quadratic function or by performing Linear Regression in the $x^2$ space.*




(a) (b)

Figure I.3 Quadratic Feature Regression example. (a) Training data. (b) True noiseless function in black and Linear Regression in red.




(a) (b)

Figure I.4 Sine Feature Regression example. (a) Training data. (b) True noiseless function in black and Linear Regression in red.

Feature regression is a powerful and natural extension of Linear Regression to the non-linear case, permitting to tackle complex functions and to construct very accurate models. However, the obvious downside is that all features must be provided by hand. In practice, either an expert must be able to give a list of functions that should represent the data, or these basis functions should be guessed from the landscape of the training points.

This issue considerably limits the practical use of this technique. Fortunately, an extensive amount of work has been dedicated to overcoming this issue, notably through the use of non-parametric regression.

## I–1.3 Adding generality: Non-parametric regression

Notably, we focus here on the kernel-based non-parametric regression techniques, such as the Radial Basis Functions (RBF) model. The basic idea of these methods is to construct the basis functions based on the training data. In this context, an intu-

itive strategy is to put information where training points are located and to diffuse it in their neighborhood smoothly. Thus the RBF technique roughly consists in placing local functions, called kernel functions, on each training point and considering the global predictor as a weighted sum of these local basis functions.

By construction, this approach constructs a number $m$ of basis functions equal to the number of training data $n$. The linear system governing the interpolation of training data is thus squared and admits a unique solution given by:

$$\boldsymbol{w} = \Phi^{-\intercal}\boldsymbol{y} \tag{I.9}$$

**Remark** *For the above equation to be well defined, the design matrix $\Phi^{\intercal}$, and thus the chosen kernel function, has to be invertible. Note that the technique is sometimes employed with non-positive definite functions when a polynomial regression is used in combination. The associated basis functions are then said to be conditionally positive definite.*

Because the system is determined, the RBF model is interpolating. An example is given in Figure I.5, where a Gaussian-shaped function is placed at each training location. Formally, we set:

$$\forall i \in [\![1, n]\!], \phi_i(\boldsymbol{x}_*) = \exp\left(-\frac{(\boldsymbol{x}_* - \boldsymbol{x}_i)^2}{2\sigma_1^2}\right), \tag{I.10}$$

with a given value $\sigma_1$ for the standard deviation.

The associated weight vector is the solution of system (I.9).

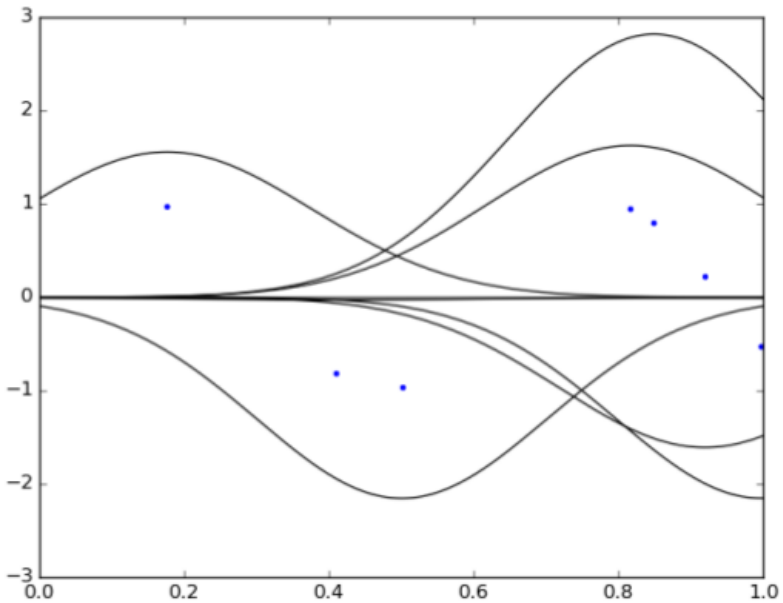



Figure I.5 RBF model example. Training points in blue and weighted basis functions in black.

Summing these weighted basis functions gives a smooth interpolating model, that adapts to the training data to capture non-linearities. This model reads:

$$\hat{y}_* = \boldsymbol{\phi}(\boldsymbol{x}_*)^{\intercal}\Phi^{-\intercal}\boldsymbol{y}. \tag{I.11}$$

One can note that the RBF model is both a linear combination of the basis functions, $\hat{y}_* = \boldsymbol{\phi}(\boldsymbol{x}_*)^\intercal \boldsymbol{w}$ and a linear combination of the training values $\hat{y}_* = \boldsymbol{\alpha}^\intercal \boldsymbol{y}$.

The final RBF model associated with the basis functions and weights represented in Figure I.5 is drawn in Figure I.6. It shows excellent qualitative behavior and smoothly connects the training data.

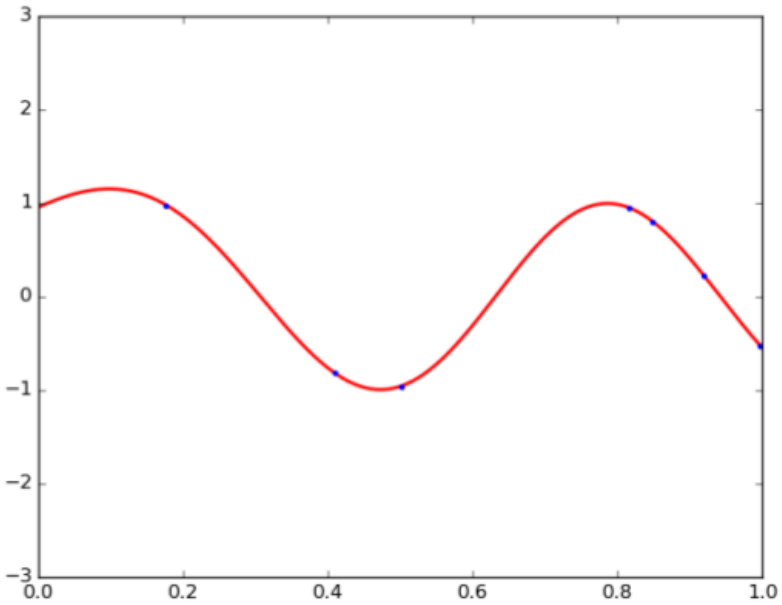
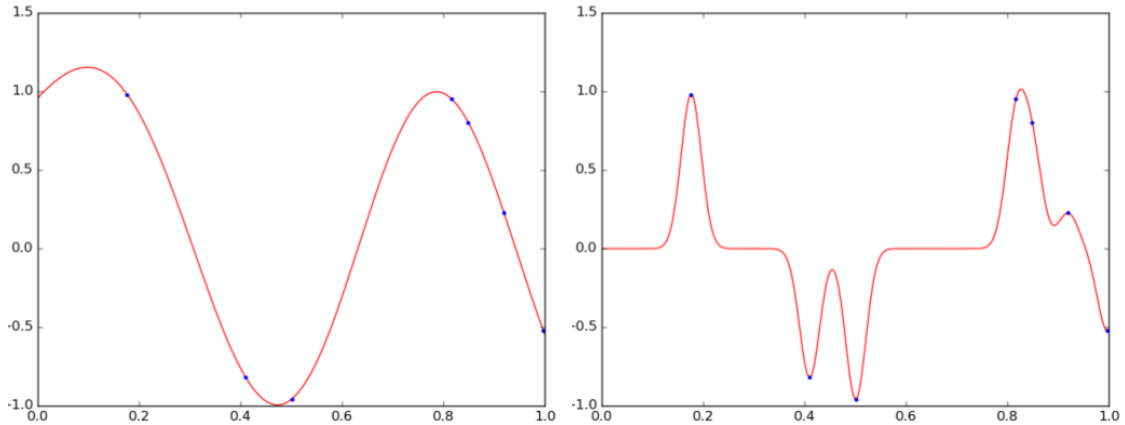



Figure I.6 RBF model example. Training points in blue and RBF model in red.

Note that the choice of the kernel function is critical for the overall shape of the regression model. Figure I.7 shows two RBF models constructed with Gaussian-shaped kernel function. However, Figure I.7(a) features a medium standard deviation $\sigma_1$ within the kernel function whereas Figure I.7(b) is constructed on basis functions of much smaller standard deviation $\sigma_2$.

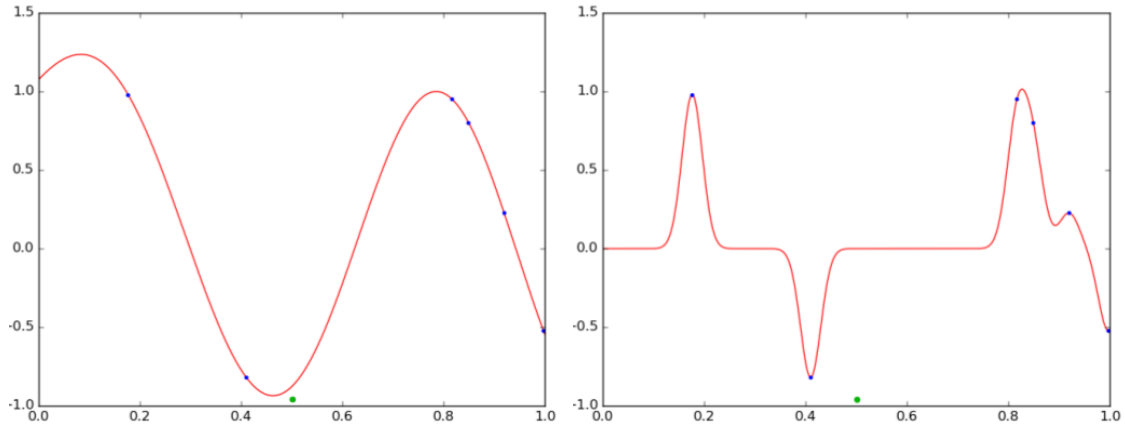



(a) (b)

Figure I.7 RBF models. (a) With standard deviation $\sigma_1$, (b) with standard deviation $\sigma_2 < \sigma_1$.

Note, however, that a missing training data has a more severe impact on the second model, Fig. I.8(b), than on the first one, Fig. I.8(a). This remark suggests a framework for selecting the standard deviation parameter, usually referred to as a hyperparame-

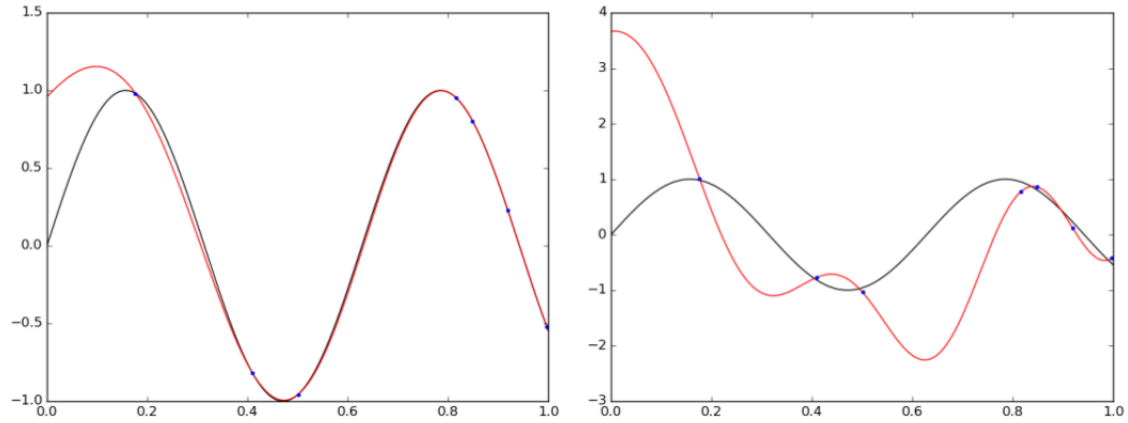
ter. RBF models are constructed on subsets of the training data, and errors are read at the missing training points. Such methods are referred to as Cross-Validation (CV) techniques. The hyperparameters can then be modified to minimize the mean CV error. This strategy aims to provide good generalization properties as the model can offer.



(a) (b)

Figure I.8 RBF models with one missing training data in green. (a) Good generalization, (b) Poor generalization.

Two major issues remain on RBF models. First, because they inherently are interpolation models, noise on training data could have a significant impact on the overall model quality. An example of such sensitivity is pictured in Figure I.9, where small noise on training values in Figure I.9(b) induces a dramatic change of the RBF model compared to the noiseless case in Figure I.9(a).



(a) (b)

Figure I.9 Impact of training noise. Original function in black, training data in blue and RBF model in red. (a) Noiseless training data, (b) Noisy training data.

Several techniques have been proposed in the literature for overcoming this issue, notably under the general concept of regularization. Intuitively, instead of minimizing the classical Least-Square loss function $L(\boldsymbol{w}) = \|\boldsymbol{y} - \Phi^\intercal \boldsymbol{w}\|_2$, thus making sure that the model interpolates the training data, the regularized problem aims to minimize the combined loss function $L_k(\boldsymbol{w}) = \|\boldsymbol{y} - \Phi^\intercal \boldsymbol{w}\|_2 + \lambda \|\boldsymbol{w}\|_k$. Using $\ell 1$ norm

regularization is usually called LASSO, and is an efficient dimension reduction technique by setting many coefficients $w_i$ to zero. As for $\ell 2$ regularization, it is either called Ridge or Tikhonov regularization and the solution can be derived analytically $\boldsymbol{w} = (\Phi\Phi^\intercal + \lambda I)^{-1}\Phi\boldsymbol{y}$. Another regularization technique consists in fixing an arbitrary number of basis functions beforehand and computing the associated best Least Square approximation of the training data. This approach is usually represented with a Neural Network view and is called Radial Basis Function Network (RBFN).

Secondly, although RBF models give a prediction of the Quantity of Interest (QoI) at any location $\boldsymbol{x}_*$, it does not express how reliable these predictions are. Intuitively, the model should be very accurate close to training points, but less reliable far from them.

As a potential answer to both the regularization and error estimation issues, Bayesian regression is presented in the following section.

## I–1.4 A probabilistic view: Bayesian Regression

### I–1.4.1 The Bayesian setting

Bayes theorem may be one of the most elegant mathematical construction and can be viewed as a mathematical definition of knowledge. In essence, Bayes rule allows updating the plausibility of a theory given some information. Formally, this theorem writes:

$$\mathbb{P}[A \mid B] = \frac{\mathbb{P}[B \mid A]\mathbb{P}[A]}{\mathbb{P}[B]}. \tag{I.12}$$

In plain English, this formula reads as such: The posterior probability of event A knowing B equals the probability of B knowing A, multiplied by the prior probability of A, divided by the marginal probability of B. In the above, the probability $\mathbb{P}[B \mid A]$ of B knowing A is usually called the likelihood of A knowing B. Essentially:

$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{marginal}}, \tag{I.13}$$

and because the denominator is roughly only a normalization constant, making sure that the probability sums to one, one often uses:

$$\text{posterior} \propto \text{likelihood} \times \text{prior}. \tag{I.14}$$

**Remark** *Bayesian reasoning plays a central role in epistemology, by giving a "formal way of reasoning". If one believes in a theory with a certain probability, each event should update this probability in a Bayesian manner.*

### I–1.4.2 Bayesian Regression

In a regression context, the problem is to find the appropriate weights $\boldsymbol{w}$ given the training data. In other words, in the context of Linear Regression, a Bayesian approach aims to compute:

$$\mathbb{P}[\boldsymbol{w} \mid X, \boldsymbol{y}] \propto \mathbb{P}[\boldsymbol{y} \mid X, \boldsymbol{w}]\mathbb{P}[\boldsymbol{w}]. \tag{I.15}$$

We assume independent input noises following Gaussian distributions with zero mean and $\sigma_n^2$ variance. As an answer to the need for regularization, the weights $\boldsymbol{w}$ can be forced to remain small through the incorporation of prior knowledge. The

use of another Gaussian distribution is much recommended to make computations tractable: $\boldsymbol{w}$ is thus assumed to follow a normal distribution with zero mean and $\Sigma_w$ covariance matrix.

Combining the likelihood and prior distributions:

$$\mathbb{P}[\boldsymbol{y} \mid X, \boldsymbol{w}] = \mathcal{N}\big(X^\intercal \boldsymbol{w},\ \sigma_n^2 I\big),$$
$$\mathbb{P}[\boldsymbol{w}] = \mathcal{N}\big(0,\ \Sigma_w\big),$$

permits to derive the posterior as follows:

$$\mathbb{P}[\boldsymbol{w} \mid X, \boldsymbol{y}] = \mathcal{N}\big(A_X^{-1} X \boldsymbol{y},\ \sigma_n^2 A_X^{-1}\big), \tag{I.16}$$

where $A_X = XX^\intercal + \sigma_n^2 \Sigma_w^{-1}$.

Thus, the Bayesian Linear Regression model $\hat{y}_* = \boldsymbol{x}_*^\intercal \boldsymbol{w}$ yields:

$$\mathbb{P}[\hat{y}_* \mid \boldsymbol{x}_*, X, \boldsymbol{y}] = \mathcal{N}\big(\boldsymbol{x}_*^\intercal A_X^{-1} X \boldsymbol{y},\ \sigma_n^2 \boldsymbol{x}_*^\intercal A_X^{-1} \boldsymbol{x}_*\big). \tag{I.17}$$

Of course, this Bayesian Linear Regression can be performed in feature space. The Bayesian Feature Regression weights simply write:

$$\mathbb{P}[\boldsymbol{w} \mid \Phi, \boldsymbol{y}] = \mathcal{N}\big(A_\Phi^{-1} \Phi \boldsymbol{y},\ \sigma_n^2 A_\Phi^{-1}\big), \tag{I.18}$$

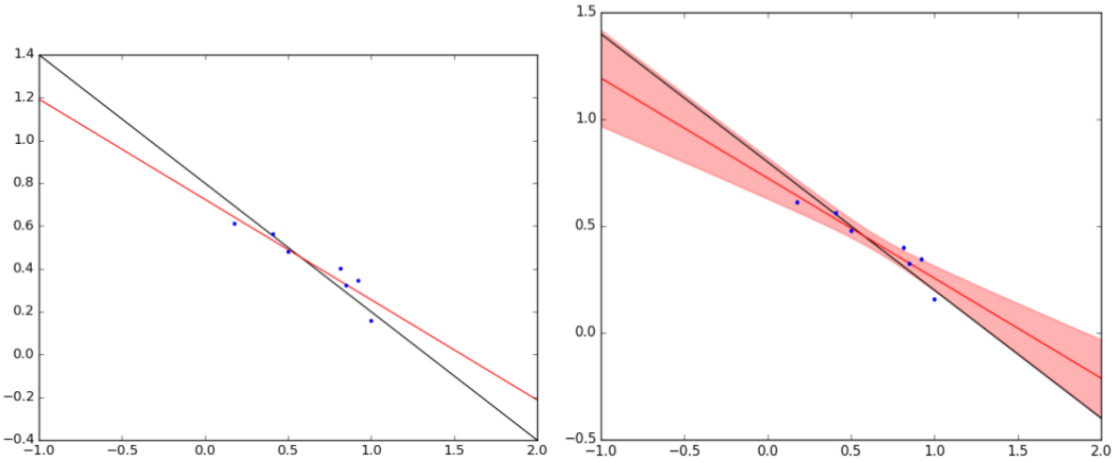
where $A_\Phi = \Phi\Phi^\intercal + \sigma_n^2 \Sigma_w^{-1}$, which gives the following statistical model:

$$\mathbb{P}[\hat{y}_* \mid \boldsymbol{x}_*, \Phi, \boldsymbol{y}] = \mathcal{N}\big(\boldsymbol{\phi}(\boldsymbol{x}_*)^\intercal A_\Phi^{-1} \Phi \boldsymbol{y},\ \sigma_n^2 \boldsymbol{\phi}(\boldsymbol{x}_*)^\intercal A_\Phi^{-1} \boldsymbol{\phi}(\boldsymbol{x}_*)\big). \tag{I.19}$$

**Remark** *One may recognize here the weights computed with Tikhonov regularization but with a regularization matrix $\sigma_n^2 \Sigma_w^{-1}$ instead of $\lambda I$. Hence, stronger regularization is achieved either by increasing the assumed variance of the input noise $\sigma_n^2$ or by reducing the a priori allowed range for weights described by $\Sigma_w$. Intuitively, the former allows the model to pass far from training data without penalizing the loss function too much, and the latter only allows small weights to be used.*
*If one aims for an unregularized solution, for example with an interpolating RBF model, the regularizing term $\sigma_n^2 \Sigma_w^{-1}$ should be set to zero. This fact can be regarded as an assumption of noiseless training data or as a process letting the weights grow as much as desired, with a non-informative prior.*

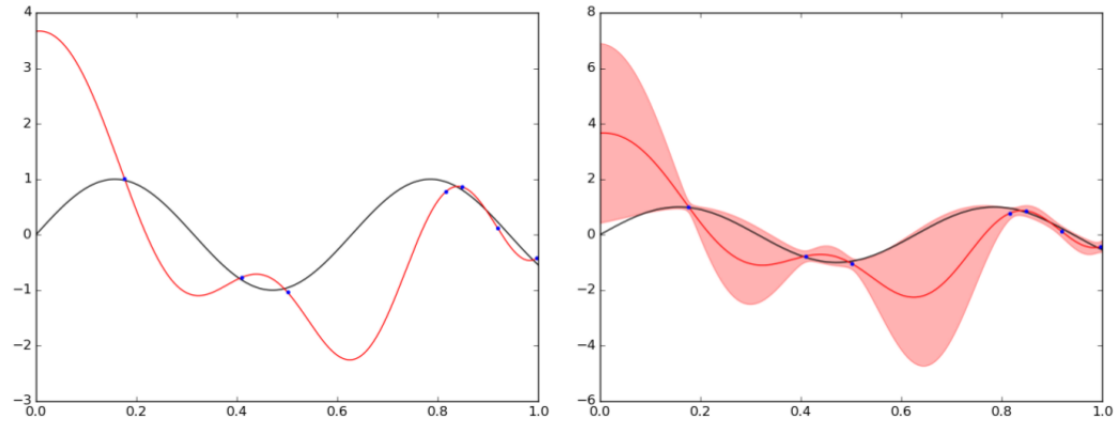
In the unregularized case ($\sigma_n^2 \Sigma_w^{-1} = 0$), two examples are given in Figure I.10 and I.11 with linear and RBF features respectively. A translucent band is added around the mean curve to plot the statistical model, corresponding to the $\pm 2\sigma$ area. Equivalently, these bounds can be regarded as the 2.5% and 97.5% quantiles of the Gaussian distribution at each location $x$. In both Figures, the true underlying function is drawn in black, and the model is represented in red.

Using the same training data and RBF basis functions as in Figure I.11, we finally plot in Figure I.12 the output of regularized Bayesian Regression. For simplicity, $\Sigma_w$ is set to $I$ and $\sigma_n^2$ is set to the variance that was used to generate the training data. Figure I.12(a) shows that the original function can be efficiently retrieved through regularization, and more generally that Bayesian Regression is a very efficient framework for doing so, by returning not only a prediction but a whole Probability Density Function for the output value $y_*$ at a new location $\boldsymbol{x}_*$.



(a) (b)

Figure I.10 (a) Linear Regression. (b) Bayesian Linear Regression with no regularization, mean and $\pm 2\sigma$ band in red.



(a) (b)

Figure I.11 (a) RBF model. (b) Bayesian Regression on RBF basis functions with no regularization, mean and $\pm 2\sigma$ band in red.

However, Figure I.12(b) reveals that a Bayesian treatment of the weights associated with the RBF basis functions is not sufficient to capture the uncertainty due to the distance from training points. Because all basis functions have a zero value far from the training points, whatever the weight values, the model returns zero.

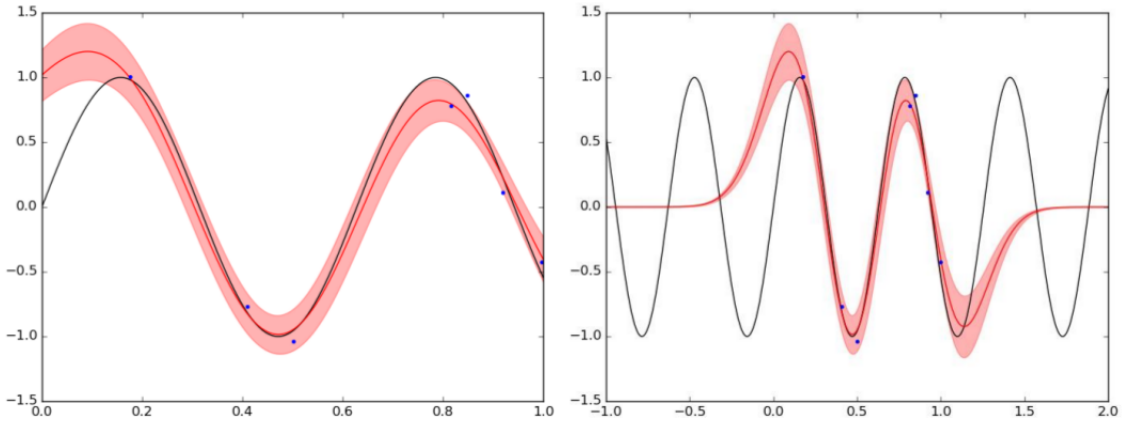
To tackle this issue, we introduce in the following section an alternative formulation that is suitable for employing the so-called *kernel trick*.

## I–1.5 Gaussian processes

### I–1.5.1 Alternative formulation and kernel trick

In the above, the Bayesian non-parametric model writes:

$$\mathbb{P}[\hat{y}_* \mid \boldsymbol{x}_*, \Phi, \boldsymbol{y}] = \mathcal{N}\big(\boldsymbol{\phi}(\boldsymbol{x}_*)^\intercal A_\Phi^{-1} \Phi \boldsymbol{y},\ \sigma_n^2 \boldsymbol{\phi}(\boldsymbol{x}_*)^\intercal A_\Phi^{-1} \boldsymbol{\phi}(\boldsymbol{x}_*)\big), \tag{I.20}$$



(a) (b)

Figure I.12 Regularized Bayesian Regression on RBF basis functions, mean prediction and $\pm 2\sigma$ band in red, true function in black. (a) Zoomed-in view, (b) zoomed-out view.

with

$$A_\Phi = \Phi\Phi^\intercal + \sigma_n^2 \Sigma_w^{-1}. \tag{I.21}$$

However, using the following equality, where $U$ and $V$ are matrices and $V$ positive definite:

$$\begin{aligned} U(U^\intercal V U + I) &= UU^\intercal V U + U = (UU^\intercal + V^{-1})VU \\ U(U^\intercal V U + I) &= (UU^\intercal + V^{-1})VU \\ (UU^\intercal + V^{-1})^{-1}U &= VU(U^\intercal V U + I)^{-1}, \end{aligned} \tag{I.22}$$

one can derive an alternate formulation for the mean prediction. Setting $U = \Phi$ and $V = \sigma_n^{-2}\Sigma_w$, we obtain:

$$\begin{aligned} (\Phi\Phi^\intercal + \sigma_n^2\Sigma_w^{-1})^{-1}\Phi &= \sigma_n^{-2}\Sigma_w\Phi(\Phi^\intercal\sigma_n^{-2}\Sigma_w\Phi + I)^{-1} \\ A_\Phi^{-1}\Phi &= \Sigma_w\Phi(\Phi^\intercal\Sigma_w\Phi + \sigma_n^2 I)^{-1}. \end{aligned} \tag{I.23}$$

Similarly, using the matrix inversion lemma, that states:

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(C^{-1} + DA^{-1}B)^{-1}DA^{-1}, \tag{I.24}$$

with $A = \sigma_n^2\Sigma_w^{-1}$, $B = \Phi$, $C = I$ and $D = \Phi^\intercal$, one can reformulate the variance as follows:

$$\begin{aligned} \sigma_n^2\boldsymbol{\phi}(\boldsymbol{x}_*)^\intercal A_\Phi^{-1}\boldsymbol{\phi}(\boldsymbol{x}_*) &= \sigma_n^2\boldsymbol{\phi}(\boldsymbol{x}_*)^\intercal A_\Phi^{-1}\boldsymbol{\phi}(\boldsymbol{x}_*) \\ &= \sigma_n^2\boldsymbol{\phi}(\boldsymbol{x}_*)^\intercal(\Phi\Phi^\intercal + \sigma_n^2\Sigma_w^{-1})^{-1}\boldsymbol{\phi}(\boldsymbol{x}_*) \\ &= \sigma_n^2\boldsymbol{\phi}(\boldsymbol{x}_*)^\intercal\big(\sigma_n^{-2}\Sigma_w - \sigma_n^{-2}\Sigma_w\Phi(I + \Phi^\intercal\sigma_n^{-2}\Sigma_w\Phi)^{-1}\Phi^\intercal\sigma_n^{-2}\Sigma_w\big)\boldsymbol{\phi}(\boldsymbol{x}_*) \\ &= \boldsymbol{\phi}(\boldsymbol{x}_*)^\intercal\big(\Sigma_w - \Sigma_w\Phi(\sigma_n^2 I + \Phi^\intercal\Sigma_w\Phi)^{-1}\Phi^\intercal\Sigma_w\big)\boldsymbol{\phi}(\boldsymbol{x}_*) \\ &= \boldsymbol{\phi}(\boldsymbol{x}_*)^\intercal\Sigma_w\boldsymbol{\phi}(\boldsymbol{x}_*) - \boldsymbol{\phi}(\boldsymbol{x}_*)^\intercal\Sigma_w\Phi(\Phi^\intercal\Sigma_w\Phi + \sigma_n^2 I)^{-1}\Phi^\intercal\Sigma_w\boldsymbol{\phi}(\boldsymbol{x}_*). \end{aligned} \tag{I.25}$$

As seen before, regularization can be controlled with either $\sigma_n^2$ or $\Sigma_w$. Thus for simplicity, $\Sigma_w = I$ is used in the following. Denoting $K = \Phi^\intercal\Phi$, $\boldsymbol{k}_* = \Phi^\intercal\boldsymbol{\phi}(\boldsymbol{x}_*)$ and $k_{**} = \boldsymbol{\phi}(\boldsymbol{x}_*)^\intercal\boldsymbol{\phi}(\boldsymbol{x}_*)$, the statistical model reads:

$$\mathbb{P}[\hat{y}_* \mid \boldsymbol{x}_*, \Phi, \boldsymbol{y}] = \mathcal{N}\big(\boldsymbol{k}_*(K + \sigma_n^2 I)^{-1}\boldsymbol{y},\ k_{**} - \boldsymbol{k}_*^\intercal(K + \sigma_n^2 I)^{-1}\boldsymbol{k}_*\big). \tag{I.26}$$

**Remark** *Note that, if matrix $\Sigma_w$ were to be retained, its square root $\Sigma_w^{1/2}$ could be computed and the different matrices $K$, $\boldsymbol{k}_*$ and $k_{**}$ could be defined as the dot product between basis functions $\boldsymbol{\psi}(\boldsymbol{x}_*) = \Sigma_w^{1/2}\boldsymbol{\phi}(\boldsymbol{x}_*)$.*

The crucial point with this new formulation is that the matrix to invert $K + \sigma_n^2 I$ is now of size $n \times n$ with $n$ the number of training samples instead of size $m \times m$ with $m$ the number of features in Feature Regression. This comes from the term $\Phi\Phi^\intercal$ that has been replaced with $\Phi^\intercal\Phi$. An important point here is to lower the computational cost when $n < m$, although this is rarely the case and the RBF approach already tackles this issue by setting $m = n$. The second and most significant point is to make the statistical model independent from the number of features. This notably permits to make use of an infinite number of features, implicitly defined through the famous *kernel trick*. In practice, it allows to lift any algorithm relying solely on inner products to an infinite dimensional feature space by replacing all inner products with a kernel function $k$. For example, in our case, an infinite number of features are defined with:

$$\begin{bmatrix} \phi_1(\boldsymbol{x}) \\ \phi_2(\boldsymbol{x}) \\ \vdots \end{bmatrix}^\intercal \begin{bmatrix} \phi_1(\boldsymbol{x}_*) \\ \phi_2(\boldsymbol{x}_*) \\ \vdots \end{bmatrix} = \boldsymbol{\phi}(\boldsymbol{x})^\intercal \boldsymbol{\phi}(\boldsymbol{x}_*) = k(\boldsymbol{x}, \boldsymbol{x}_*). \tag{I.27}$$

We fix $k_{**} = k(\boldsymbol{x}_*, \boldsymbol{x}_*)$, $\boldsymbol{k}_* = \begin{bmatrix} k(\boldsymbol{x}_1, \boldsymbol{x}_*) \\ \vdots \\ k(\boldsymbol{x}_n, \boldsymbol{x}_*) \end{bmatrix}$ and $K = \begin{bmatrix} k(\boldsymbol{x}_1, \boldsymbol{x}_1) & \dots & k(\boldsymbol{x}_1, \boldsymbol{x}_n) \\ \vdots & \ddots & \vdots \\ k(\boldsymbol{x}_n, \boldsymbol{x}_1) & \dots & k(\boldsymbol{x}_n, \boldsymbol{x}_n) \end{bmatrix}$ with kernel function:

$$k(\boldsymbol{a}, \boldsymbol{b}) = \exp\left( -\frac{(\boldsymbol{a} - \boldsymbol{b})^2}{2\sigma_1^2} \right), \tag{I.28}$$

corresponding to the RBF basis functions from the preceding sections. The non-parametric statistical model:

$$\mathbb{P}[\hat{y}_* \mid \boldsymbol{x}_*, \Phi, \boldsymbol{y}] = \mathcal{N}\left( \boldsymbol{k}_*(K + \sigma_n^2 I)^{-1}\boldsymbol{y},\ k_{**} - \boldsymbol{k}_*^\intercal(K + \sigma_n^2 I)^{-1}\boldsymbol{k}_* \right) \tag{I.29}$$

yields the results shown in Figure I.13. The hyperparameters $\sigma_1$ and $\sigma_n$ are chosen equal to the preceding test-cases. The mean prediction drawn in dark blue is very similar to the regularized RBF, but the model uncertainty is appropriately high in areas far from training points.

This Bayesian non-parametric regression with kernel trick corresponds to the famous Gaussian Process(GP) model.

**Remark** *Note that the mean prediction $\mu_{GP} = \boldsymbol{k}_*(K + \sigma_n^2 I)^{-1}\boldsymbol{y}$ is a linear combination of the kernel function evaluated between $\boldsymbol{x}_*$ and each training location $\boldsymbol{x}_i$. Similarly to the RBF model, the prediction in Figure I.13 is a combination of Gaussian-shaped functions. Notably, the representer theorem from the RKHS theory presented in the next section shows that this finite combination is the best approximation in the infinite-dimensional feature space.*

**Remark** *Intuitively, the variance associated with the GP model accurately captures the lack of knowledge because, contrarily to the RBF model, there is an infinite number of features, namely Gaussian-shaped functions placed at each location of the input space. A Bayesian approach would detect that the weight associated with features far from the training data does not deteriorate the fit of the model. The likelihood being non-informative, weights would remain at their prior distributions.*

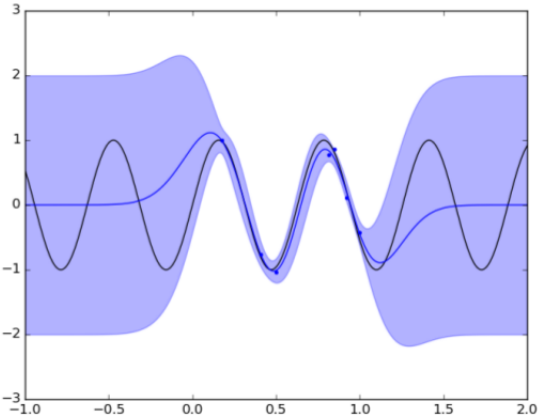



Figure I.13 Gaussian Process model. True function in black, mean prediction and $\pm 2\sigma$ band in blue.

### I–1.5.2 Function-space view

A Gaussian Process can be regarded, in a much more natural way, as a random process with Gaussian distribution conditioned on some training values, with a prior covariance function $k$.

Such a random process is an infinite-dimensional collection of random variables $y \mid \boldsymbol{x}$, spanning the input space, such that any finite subset $\left\{y \mid \boldsymbol{x}_i\right\}_i$ shows a Gaussian joint distribution. A Gaussian Process is thus entirely specified by its mean and covariance functions. Without loss of generality, we consider here the mean function to be zero and focus on the covariance function $k$.

By definition, with $\boldsymbol{x}$ the set of training points and $\boldsymbol{x}_*$ a set of prediction locations, a Gaussian Process is defined as follows:

$$\begin{bmatrix} \boldsymbol{y} \\ \hat{\boldsymbol{y}}_* \end{bmatrix} \sim \mathcal{N}\left(\boldsymbol{0}, \begin{bmatrix} K + \sigma_n^2 I & K_* \\ K_*^\intercal & K_{**} \end{bmatrix}\right), \tag{I.30}$$

where $K$ contains $k(\boldsymbol{x}_i, \boldsymbol{x}_j)$, $K_*$ contains $k(\boldsymbol{x}_i, \boldsymbol{x}_{*_j})$ and $K_{**}$ contains $k(\boldsymbol{x}_{*_i}, \boldsymbol{x}_{*_j})$.

Given this joint normal distribution, one can incorporate the known training values $\boldsymbol{y}$ and derive the model $\hat{\boldsymbol{y}}_*$ through the formula of conditional probabilities. For two random vectors $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ with joint distribution $\mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{bmatrix}, \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}\right)$, conditional probability can be computed as follows:

$$\begin{aligned} \mu_{\boldsymbol{A}_2 | \boldsymbol{A}_1 = \boldsymbol{a}_1} &= \mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(\boldsymbol{a}_1 - \boldsymbol{\mu}_1) \\ \Sigma_{\boldsymbol{A}_2 | \boldsymbol{A}_1 = \boldsymbol{a}_1} &= \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}. \end{aligned} \tag{I.31}$$

Applied to the GP joint distributions from Equation (I.30), it permits to retrieve the GP model defined in Equation (I.29):

$$\begin{aligned} \hat{\boldsymbol{y}}_* &\sim \mathcal{N}(\mu_{GP}, \Sigma_{GP}) \\ \mu_{GP} &= K_*^\intercal (K + \sigma_n^2 I)^{-1} \boldsymbol{y} \\ \Sigma_{GP} &= K_{**} - K_*^\intercal (K + \sigma_n^2 I)^{-1} K_*. \end{aligned} \tag{I.32}$$

Note that contrarily to Equation (I.29), the above formula directly deals with several test points $\boldsymbol{x}_*$ simultaneously and returns a conditioned covariance matrix.

**Remark** *The above formula tackles homoscedastic input noises, meaning that all noises are of variance $\sigma_n^2$. In a heteroscedastic setting, where each training value has its own noise variance $\sigma_i^2$, one should simply replace the term $\sigma_n^2 I$ with a diagonal matrix $\Delta = diag\left(\{\sigma_i^2\}_i\right)$.*

This function-space view gives a very explicit way to draw realizations of the GP model. Because this statistical model is essentially one infinite-dimensional distribution, any realization of this distribution (on a finite subset of the input space) is a functional realization of the Quantity of Interest, drawn at a finite number of locations. Figure I.14 gives some examples of functional GP realizations. The eight training points are drawn in blue, and the five realizations are represented with gray curves. Note that these curves are a collections of 1000 equally distributed points along the x-axis.

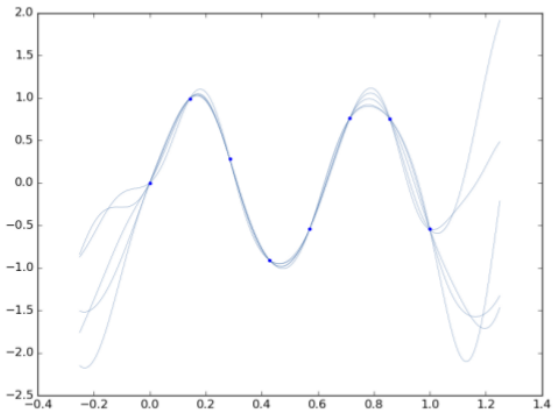



Figure I.14 Five GP realizations, represented in gray, based on the blue training data.

We recall that formally, drawing realizations from a multivariate Gaussian distribution requires the computation of a square-root matrix. Similarly to the univariate case:

$$X^{(i)} = \mu_X + \sqrt{\sigma_X^2} Z, \quad Z \sim \mathcal{N}(0,1), \tag{I.33}$$

multivariate realizations are drawn as follows:

$$\hat{\boldsymbol{y}}_*^{(i)} = \boldsymbol{\mu}_{GP} + L_{GP}\boldsymbol{Z}, \quad \boldsymbol{Z} \sim \mathcal{N}(\boldsymbol{0}, I), \tag{I.34}$$

where $L_{GP}$ refers to the Cholesky decomposition of $\Sigma_{GP}$, so that $\Sigma_{GP} = L_{GP} L_{GP}^\intercal$.

### I–1.5.3 Kernel and hyperparameters

One may notice in Figure I.13 that the $\pm 2\sigma$ band exactly spans four units. This behavior comes from the fact that the infinity of basis functions defined in Equation (I.28) for each $a$ as a function of $b$ all have a height of 1 at $a = b$. The prior $\Sigma_w = I$ on the weights thus yields a prior variance of 1, revealed far from training data.

This variance may, however, be highly inappropriate when the QoI varies in much smaller or broader ranges. Similarly to the RBF, GP hyperparameters must be determined to best fit the training data. Practically, this signifies that the prior, namely the basis functions, must be optimized for the GP model to yield an accurate prediction of the QoI. Several kernel functions can be used for constructing GP models. The most classical ones are the squared-exponential (Gaussian) kernel, the exponential kernel,

the family of Matern kernels, the polynomial kernels, *etc*. GP models entirely rely on the chosen kernel function, and many choices have been proposed in the literature, for treating *e.g* discrete variables, multiple outputs [Chan, 2013, Boyle and Frean, 2005, Hong et al., 2017], additive functions [Duvenaud et al., 2011] and high dimension [Lawrence, 2004, Jiang et al., 2014]. For more insights on these kernels and many other ones, we advise the reader to look at [Williams and Rasmussen, 2006] and [Duvenaud, 2014].

The one-dimensional squared exponential kernel writes:

$$k_{SE}(x,x') = \sigma_p^2 \exp(-\frac{(x-x')^2}{2l^2}). \tag{I.35}$$

Standard practice for fitting the Gaussian Process model is thus to find the most appropriate values of three hyperparameters, namely $\sigma_p^2$, the prior variance, $l$ the characteristic lengthscale and $\sigma_n^2$ the noise variance.

Such hyperparameter optimization can be carried out through Cross-Validation, similarly to the RBF model, but is more classically performed with a Maximum Likelihood (ML) paradigm. The latter approach finds the hyperparameters that maximize the marginal likelihood of obtaining the training data under the GP prior $\boldsymbol{y} \sim \mathcal{N}(\mathbf{0}, K + \sigma_n^2 I)$. In practice, one often maximizes the associated log-likelihood, that formally writes:

$$\log \mathbb{P}[\boldsymbol{y} \mid \boldsymbol{x}] = -\frac{1}{2}\boldsymbol{y}^\intercal (K + \sigma_n^2 I)^{-1}\boldsymbol{y} - \frac{1}{2}\log|K + \sigma_n^2 I| - \frac{n}{2}\log 2\pi, \tag{I.36}$$

where the first term refers to the fitting error, the second to the model complexity and the third is a normalizing constant. In summary, the ML approach naturally balances accuracy and simplicity.

Such maximization is usually carried out with multi-start gradient methods (*e.g.* L-BFGS), or with Evolutionary Algorithms (EA) (*e.g.* Differential Evolution).

**Remark** *Note that a fully Bayesian approach would require to formulate hyper-priors, that is to say, prior distributions for the hyperparameters, to tract the full posterior distribution for each of them. The ML technique only accounts for the values maximizing the likelihood value, which corresponds to the Maximum A Posteriori (MAP) when the prior is uninformative.*

## I–2 Reproducing Kernel Hilbert Spaces

The preceding section has shown the crucial role of kernel functions within the Gaussian Process models. While they can naturally be regarded as the spatial covariance function of the random process, RKHS theory shows that they can be viewed as evaluation functional in a particular Hilbert space.

In the following, we do not propose a detailed description of the RKHS theory, which can be found in the literature, notably in [Berlinet and Thomas-Agnan, 2011]. A quick and intuitive overview is proposed following [Gretton, 2013] and [Sejdinovic and Gretton, 2012]. A special interest here is devoted to the link between RKHS and statistical modeling. Note that such a link has already been pointed out, for example in [Anjyo and Lewis, 2011] and [Seeger, 2000].

## I–2.1 RKHS theory fundamentals

Let us recall the definition of a Hilbert space.

**Definition 1.** A Hilbert space $\mathcal{H}$ is a Banach space which norm $\|\cdot\|$ comes from an inner product $\langle\cdot,\cdot\rangle$. Formally, for any $f \in \mathcal{H}$:

$$\|f\| = \sqrt{\langle f, f\rangle}. \tag{I.37}$$

Kernel functions can then be defined as inner products between feature maps in a Hilbert space.

**Definition 2.** With $\mathcal{X}$ a non-empty set, a function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called a kernel if there exists a Hilbert space $\mathcal{H}$ and a feature map $\phi : \mathcal{X} \to \mathcal{H}$ so that for any $(x, x') \in \mathcal{X}^2$,

$$k(x, x') = \langle \phi(x), \phi(x')\rangle_{\mathcal{H}}. \tag{I.38}$$

**Remark** *Note that in the previous section, kernels were indeed defined as inner products between features in Equation I.27.*

Kernel functions are symmetric by definition. Their positive definiteness can be easily derived through bilinearity as, for any $(a_1, ..., a_n) \in \mathbb{R}^n$ and $(x_1, ..., x_n) \in \mathcal{X}^n$,

$$\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j k(x_i, x_j) = \left\langle \sum_{i=1}^{n} a_i \phi(x_i), \sum_{j=1}^{n} a_j \phi(x_j) \right\rangle = \left\| \sum_{i=1}^{n} a_i \phi(x_i) \right\|^2 \geq 0. \tag{I.39}$$

A first definition for Reproducing Kernel Hilbert Space (RKHS) is then proposed in Definition 3 based on the reproducing property.

**Definition 3.** A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called a reproducing kernel, and the Hilbert space $\mathcal{H}$ of $\mathbb{R}$-valued functions defined over $\mathcal{X}$ is called a Reproducing Kernel Hilbert Space (RKHS) if:

$$\begin{aligned} &\forall x \in \mathcal{X},\ k(\cdot, x) \in \mathcal{H} \\ \text{and}\quad &\forall x \in \mathcal{X}, \forall f \in \mathcal{H},\ \langle f, k(\cdot, x)\rangle_{\mathcal{H}} = f(x), \end{aligned} \tag{I.40}$$

where the second line corresponds to the reproducing property. This property directly permits to derive the uniqueness of the reproducing kernel in $\mathcal{H}$.

In particular, for any couple $(x, x') \in \mathcal{X}^2$, a reproducing kernel $k$ yields $k(x, x') = \langle k(\cdot, x), k(\cdot, x')\rangle_{\mathcal{H}}$. Hence, the feature map $\phi(x) = k(\cdot, x)$ shows that a reproducing kernel is indeed a kernel as defined in Definition 2.

**Definition 4.** Within a Hilbert space $\mathcal{H}$ of functions $f : \mathcal{X} \to \mathbb{R}$, an evaluation functional $\delta_x : \mathcal{H} \to \mathbb{R}$ is defined by $\delta_x(f) = f(x)$.

Note that, based on Definition 3, the inner product with the reproducing kernel function $k(\cdot, x)$ takes the role of evaluation functional in $\mathcal{H}$:

$$\forall f \in \mathcal{H},\ \delta_x(f) = \langle f, k(\cdot, x)\rangle_{\mathcal{H}}. \tag{I.41}$$

An alternative definition of Reproducing Kernel Hilbert Space can be proposed in terms of evaluation functional.

**Definition 5.** An RKHS can also be defined in terms of evaluation functional. Namely, $\mathcal{H}$ is an RKHS if $\delta_x$ is continuous (thus bounded) over $\mathcal{X}$. That is to say if $\forall x \in \mathcal{X}$, $\exists \lambda_x \geq 0$ such that:

$$\forall f \in \mathcal{H},\ |f(x)| = |\delta_x(f)| \leq \lambda_x \|f\|_{\mathcal{H}}. \tag{I.42}$$

This definition notably shows that in an RKHS convergence in norm implies pointwise convergence.

Equivalence between reproducing kernels (Def. 3) and bounded evaluation functionals (Def. 5) can be shown using Cauchy-Schwarz inequality for the forward implication, and through the use of Riesz representer theorem in the other direction. More details can be found in [Sejdinovic and Gretton, 2012].

Given an RKHS, we have shown that there exists a unique reproducing kernel $k$ and that it is a kernel, meaning that it is a positive definite function that can be written as an inner product in a feature space. A significant result in the RKHS theory is the Moore-Aronszajn theorem [Aronszajn, 1950] that gives the third implication and thus the equivalence between these concepts.

**Theorem 1.** *For any positive definite function $k(x, x')$, there exists a unique RKHS $\mathcal{H}$ for which $k$ is a reproducing kernel.*

This constructive theorem shows that the completion of $Span\Big(\{k(\cdot, x)\}_{x\in\mathcal{X}}\Big)$ forms a unique Hilbert Space $\mathcal{H}$ in which $k$ verifies the reproducing properties.

In summary, the notions of positive definite functions, kernels (inner product between feature maps) and reproducing kernels (associated with an RKHS) are all equivalent.

In a regression context, one aims to find the function $f$ that best fits some training data. In practice, reducing this search to an RKHS $\mathcal{H}$ with reproducing kernel $k$, this corresponds to determining the infinite number of weights $w_i$ that characterize $f = \sum_{i=1}^{\infty} w_i k(\cdot, x_i)$.

However, the representer theorem [Schölkopf et al., 2001] shows that a regularized regression problem in an RKHS only requires as many weights $w_i$ as the number of training data $y_i$.

**Theorem 2.** *Considering an RKHS $\mathcal{H}$ with reproducing kernel $k$, the solution to the regularized regression problem:*

$$\begin{aligned} &\text{Find } f \in \mathcal{H} \text{ that minimize } L_\lambda(f), \\ &\text{with } L_\lambda(f) = \sum_{i=1}^{n} (y_i - f(x_i))^2 + \lambda \|f\|_{\mathcal{H}}^2, \end{aligned} \tag{I.43}$$

*can be found in the form:*

$$f(x) = \sum_{i=1}^{n} w_i k(x, x_i). \tag{I.44}$$

This result can be derived by applying the reproducing property on an orthogonal decomposition of $f$.

**Remark** *The representer theorem explains how the kernel trick, that elevates the regression to an infinite-dimensional feature space is solved using only an $n$-dimensional system.*

## I–2.2 Kernel Mean Embeddings

The RKHS theory allows creating a general setting that formalizes the use of smoothing kernels in a regression context. However, as presented in [Berlinet and Thomas-Agnan, 2011] and [Muandet et al., 2017], RKHS are also capable of embedding probability distributions by constructing a feature map from the space of probability measures to specific function space.

In short, this embedding is performed by deriving the representer of the expectation functional in a given RKHS $\mathcal{H}$ with reproducing kernel $k$. Making use of the reproducing property, for any $f \in \mathcal{H}$ and any probability measure $\mathbf{P}$, the following can be derived:

$$\begin{aligned} \int f(x)\mathrm{d}\mathbf{P}(x) &= \int \langle f, k(\cdot, x)\rangle_{\mathcal{H}} \mathrm{d}\mathbf{P}(x) \\ &= \left\langle f, \int k(\cdot, x)\mathrm{d}\mathbf{P}(x) \right\rangle_{\mathcal{H}} \\ &= \langle f, \mu_X \rangle_{\mathcal{H}}. \end{aligned} \tag{I.45}$$

Equation I.45 yields two important results. First, for any function $f$ of $\mathcal{H}$, computing its expectation with respect to a probability measure $\mathbf{P}$ only requires to compute the inner product between $f$ and the mean embedding $\mu_X$, which greatly reduces the associated burden. Secondly, the Kernel Mean Embedding (KME) $\mu_X$ directly serves as a representer of the probability measure $\mathbf{P}$ in $\mathcal{H}$. Note that this mapping $\mathbf{P} \mapsto \mu_X = \int k(\cdot, t)\mathrm{d}\mathbf{P}(t) = \mathbb{E}_{X\sim\mathbf{P}}[k(\cdot, X)]$ corresponds to the convolution which is carried out when performing Kernel Density Estimation (KDE). However, the RKHS setting imposes the smoothing kernel to be positive definite and embeds the probability measure into a fixed feature space.

As expected the empirical KME with uniform weights writes:

$$\hat{\mu}_X = \frac{1}{n}\sum_{i=1}^{n} k(\cdot, \boldsymbol{x}_i), \tag{I.46}$$

which corresponds to the finite sample KDE with kernel $k$. The RKHS view notably allows for computing distances between approximated distributions, through the Maximum Mean Discrepancy (MMD) metric.

**Remark** *One may wonder how an expectation can be the representer of a probability measure. Intuitively, a distribution can be uniquely determined by its moments, each of which is the expectation of a polynomial function w.r.t. the probability measure at hand. Hence, when the reproducing kernel is characteristic, the KME uniquely characterizes the probability measure. Great illustrations on how moments and densities can be recovered from the RKHS embedding can be found in [Kanagawa and Fukumizu, 2014].*

KME can be readily applied to joint distributions using tensor product feature spaces, that is to say with a multivariate kernel function that corresponds to the product of univariate kernels. Following the developments in [Song et al., 2013], covariance embeddings $C_{XY}$ can be obtained and empirically estimated.

A conditional operator $C_{Y|X} = C_{YX}C_{XX}^{-1}$ can finally be proposed in order to estimate a conditional distribution. Given a data set $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$, the Conditional Mean Embedding (CME) can finally be derived as follows:

$$\hat{\mu}_{Y|\boldsymbol{x}} = \beta_x^\intercal \boldsymbol{k}_y, \tag{I.47}$$

where $\boldsymbol{k}_y = \begin{bmatrix} k_y(\cdot, y_1) \\ \vdots \\ k_y(\cdot, y_n) \end{bmatrix}$ are functions of $y$ and $\beta_x$ are weights. Note that if these weights were uniform, this formula would relapse to the KME embedding I.45. In a conditional context, these weights can be written as follows:

$$\beta_x = \boldsymbol{k}_x^\intercal (K + \sigma I)^{-1}. \tag{I.48}$$

Similarly, $\boldsymbol{k}_x = \begin{bmatrix} k_x(\cdot, \boldsymbol{x}_1) \\ \vdots \\ k_x(\cdot, \boldsymbol{x}_n) \end{bmatrix}$, and $K$ is filled with elements $k_x(\boldsymbol{x}_i, \boldsymbol{x}_j)$. Note that the regularized inverse of $K = C_{XX}$ was used here.

**Remark** *Because normalization is intrinsic to this representation, it is much more efficient than classical KDE-based conditional approximations.*

**Remark** *One may notice that the above equation looks a lot like regression formulas. This similarity is more deeply studied in [Grünewälder et al., 2012], which demonstrates that CME is solution to a vector-valued regression problem with Tikhonov-like regularization.*

**Conclusion of the chapter** *In this chapter, an intuitive introduction to kernel methods is given, with a strong emphasis on Gaussian Process models. Their construction starts with a good understanding of Linear and Feature Regression, coupled with Bayesian reasoning. The regression model can be written only in terms of inner products, which permits to exploit the kernel trick and easily work in an infinite-dimensional feature space.*
*These developments are built on the RKHS theory, that studies the link between positive definite functions, feature maps and space of functions. In this context, functional regression with regularization shows how GP models can deal with an infinite number of features in practice.*
*Finally, Kernel Mean Embeddings are introduced and permit to represent probability measures in an RKHS. Key elements are presented, and major works are referenced. The recent development of Conditional Mean Embeddings, which formulation highly resemble a regression problem, is also introduced.*
*These kernel methods, notably Gaussian Processes, are used in the following of the thesis for their capacity to give both an accurate prediction and an estimation of the associated uncertainty. Their capability to sample functional realizations is also extensively exploited in Chapter IV.*

# Part B

# Methods and algorithms

CHAPTER *II*

# A general approach for constrained multi-objective optimization under uncertainty



**Overview** *In this chapter, the uncertainty-based optimization problem is presented in details. It is reformulated as a tunable accuracy optimization problem and addressed with a probabilistic interpretation. A specific algorithm (SAMATA), is then proposed in the context of Optimization Under Uncertainty (OUU).*

**Outline** *The OUU problem is formally written in Section II–1 and different robustness and reliability measures are illustrated.*
*In Section II–2, We assume that the approximation error for computing objectives and constraints with a given accuracy can be interpreted as a random variable. We then propose a probabilistic setting for performing constrained multi-objective optimization. Notably, we introduce the concepts of probabilistic Pareto dominance and Pareto Optimal Probability (POP).*
*We finally depict the SAMATA algorithm in Section II–3.*

**Contributions** *A probabilistic setting for dealing with uncertain outputs in a multi-objective context has already been used in the literature. Though, to the author knowledge, II–2 gives the most extensive presentation of such approaches. Furthermore, the concept of Pareto-Optimal Probability (POP) is proposed, with original approximation formulas for computational efficiency.*
*Then, Section II–3 introduces the general framework that allows for cost-efficient optimization under uncertainty. This strategy is derived from the Bounding-Box approach, introduced in [Fusi and Congedo, 2016], and is here exploited in the context of random approximations with unknown distribution shape. Additionally, a Surrogate-Assisting strategy is proposed to lower the computational cost further.*

# II–1 Formulation of the problem

In this Section, the problem of constrained multi-objective Optimization Under Uncertainty (OUU) is formally defined. First, Section II–1.1 presents the Pareto dominance setting for multi-objective optimization. Then, uncertainty is incorporated within the optimization process in Section II–1.2. Finally, the measure-based setting for OUU is made explicit in Section II–1.3.

## II–1.1 Multi-objective optimization

Contrarily to mono-objective optimization problems, where the performance is a value in the totally ordered set $\mathbb{R}$, comparing multiple objectives at once requires to choose an order on $\mathbb{R}^m$, with $m$ the number of objectives. To this extent, the most classical choices are the lexicographic order and the product order, that are defined as follows.

The lexicographic order, as its name suggests, is the order in which words are sorted in a dictionary. The different letters (similarly different objectives) are compared one after another, thus putting a lot of importance on the first ones. Formally, the ordering on $\mathbb{R}^2$ is defined as such, with $\boldsymbol{a}, \boldsymbol{b} \equiv (a_1, a_2), (b_1, b_2) \in \mathbb{R}^2$:

$$\boldsymbol{a} \leq \boldsymbol{b} \iff a_1 < b_1 \text{ or } (a_1 = b_1 \text{ and } a_2 \leq b_2). \tag{II.1}$$

Likewise, the product order on $\mathbb{R}^2$, with again $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^2$, reads:

$$\boldsymbol{a} \leq \boldsymbol{b} \iff a_1 \leq b_1 \text{ and } a_2 \leq b_2. \tag{II.2}$$

One can see that the product order avoids the asymmetric behavior of the lexicographic order and treats multiple dimensions interchangeably. However, it does not give a total ordering, meaning that two individuals may be incomparable.

Figure II.1 gives an example of individuals compared in $\mathbb{R}^2$. The relations between them with respect to the lexicographic and product order are then given in Table II.1. One can note that, because the product order is only a partial ordering, points $a$ and $b$ are incomparable, meaning that $\boldsymbol{a} \not\leq \boldsymbol{b}$ and $\boldsymbol{b} \not\leq \boldsymbol{a}$. Note that the lexicographic order compares abscissa before ordinate.

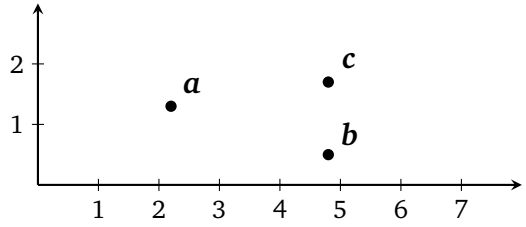



Figure II.1 Three individuals in $\mathbb{R}^2$. Relations reported in Table II.1.

| | Lexicographic order | Product order |
|---|---|---|
| $\boldsymbol{a} \leq \boldsymbol{b}$ | ✓ | ✗ |
| $\boldsymbol{b} \leq \boldsymbol{a}$ | ✗ | ✗ |
| $\boldsymbol{a} \leq \boldsymbol{c}$ | ✓ | ✓ |
| $\boldsymbol{c} \leq \boldsymbol{a}$ | ✗ | ✗ |
| $\boldsymbol{b} \leq \boldsymbol{c}$ | ✓ | ✓ |
| $\boldsymbol{c} \leq \boldsymbol{b}$ | ✗ | ✗ |

Table II.1 All relations using lexicographic and product order on individuals from Figure II.1.

**Remark** *Unless the user can rank all the objectives of its optimization problem by importance, the lexicographic order is to be avoided. In this context, the use of the product order to compare conflicting objectives is often traced back to Vilfredo Pareto in the late $XIX^{th}$ century. He defined the efficiency of a solution as the inability to improve an objective without deteriorating another one. In the following, the Pareto dominance rules are defined without loss of generality in the multi-minimization case. Hence, Pareto dominance $\boldsymbol{a} \succ \boldsymbol{b}$ corresponds to the product order $\boldsymbol{a} \leq \boldsymbol{b}$, with $\boldsymbol{a} \neq \boldsymbol{b}$.*

For two vectors $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^m$, the classical Pareto dominance reads as follows:

$$\begin{aligned}
\boldsymbol{a} \succ \boldsymbol{b} \ (\boldsymbol{a} \text{ dominates } \boldsymbol{b}) &\iff \forall j \in [\![1, m]\!],\ a_j \leq b_j \quad \text{and} \\
&\quad\ \exists j \in [\![1, m]\!],\ a_j < b_j, \\
\boldsymbol{a} \succ\!\!\succ \boldsymbol{b} \ (\boldsymbol{a} \text{ strictly dominates } \boldsymbol{b}) &\iff \forall j \in [\![1, m]\!],\ a_j < b_j, \\
\boldsymbol{a} \sim \boldsymbol{b} \ (\boldsymbol{a} \text{ is indifferent to } \boldsymbol{b}) &\iff \boldsymbol{a} \not\succ \boldsymbol{b} \text{ and } \boldsymbol{b} \not\succ \boldsymbol{a}.
\end{aligned} \tag{II.3}$$

This dominance rule can be extended in several ways to account for the presence of constraints in the optimization problem. Notably, between two unfeasible individuals, it is proposed in [Mlakar et al., 2014] and [Feliot et al., 2017] to consider the one with the smallest constraint violation as dominating the other one. In this work, to automatically discard all unfeasible individuals, we propose to consider all of them to dominate each other mutually.

**Remark** *We stress here that* $\boldsymbol{a}$ *and* $\boldsymbol{b}$ *live in the objective and constraint space, meaning that in an optimization context, they would correspond to the image of specific designs by the objective and constraint functions* $\boldsymbol{f}$ *and* $\boldsymbol{g}$*. Formally:*

$$\boldsymbol{a} = \begin{pmatrix} \boldsymbol{f}(\boldsymbol{x}_a) \\ \boldsymbol{g}(\boldsymbol{x}_a) \end{pmatrix},$$
$$\boldsymbol{b} = \begin{pmatrix} \boldsymbol{f}(\boldsymbol{x}_b) \\ \boldsymbol{g}(\boldsymbol{x}_b) \end{pmatrix}.$$

With constraints of the form $\boldsymbol{g}(\boldsymbol{x}) \leq \boldsymbol{0}$, and denoting by $\cdot_f$ and $\cdot_g$ the projection of any point on the objective and constraint dimensions, the constrained Pareto dominance is defined as follows. For two vectors $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^m$,

$$\begin{aligned} \boldsymbol{a} \succ_c \boldsymbol{b} &\iff \boldsymbol{a} \in \mathcal{A} \text{ and } \boldsymbol{a}_f \succ \boldsymbol{b}_f \text{ or } \boldsymbol{b} \in \mathcal{F}, \\ \boldsymbol{a} \succcurlyeq_c \boldsymbol{b} &\iff \boldsymbol{a} \in \mathcal{A} \text{ and } \boldsymbol{a}_f \succcurlyeq \boldsymbol{b}_f \text{ or } \boldsymbol{b} \in \mathcal{F}, \\ \boldsymbol{a} \sim_c \boldsymbol{b} &\iff \boldsymbol{a} \not\succ_c \boldsymbol{b} \text{ and } \boldsymbol{b} \not\succ_c \boldsymbol{a}, \end{aligned} \tag{II.4}$$

where the admissible and failure sets $\mathcal{A}$ and $\mathcal{F}$ are defined as such:

$$\begin{aligned} \mathcal{A} &= \left\{ \boldsymbol{c} \in \mathbb{R}^m \mid \boldsymbol{c}_g \leq \boldsymbol{0} \right\}, \\ \mathcal{F} &= \left\{ \boldsymbol{c} \in \mathbb{R}^m \mid \boldsymbol{c}_g \not\leq \boldsymbol{0} \right\}. \end{aligned} \tag{II.5}$$

In Eq. (II.5), $\boldsymbol{c}_g \leq \boldsymbol{0}$ means $\forall i, c_{g_i} \leq 0$ and signifies that $\boldsymbol{c}$ satisfies all the constraints. Contrarily, $\boldsymbol{c}_g \not\leq \boldsymbol{0}$ means that $\boldsymbol{c}$ does not comply with at least one of the constraints.

**Remark** *Note that the constrained Pareto dominance* (II.4) *properly reduces to the classical Pareto dominance* (II.3) *in the absence of constraint.*

For any set $\mathcal{Y} \subset \mathbb{R}^m$ containing points in the objectives and constraints space, the set $\mathcal{P}(\mathcal{Y})$ of all Pareto-optimal points in $\mathcal{Y}$ is called the Pareto front of the problem with regard to $\mathcal{Y}$, and can be formally written as follows in a constrained setting:

$$\mathcal{P}(\mathcal{Y}) = \left\{ \boldsymbol{a} \in \mathcal{Y} \mid \nexists \boldsymbol{b} \in \mathcal{Y},\ \boldsymbol{b} \succ_c \boldsymbol{a} \right\}. \tag{II.6}$$

For example, the Pareto front in Figure II.1 is $\{a, b\}$.

## II–1.2 Uncertainty-based optimization

A deterministic constrained multi-objective optimization problem can be written as follows, without loss of generality:

$$\begin{aligned} \text{minimize:} \quad & \boldsymbol{f}(\boldsymbol{x}), \\ \text{satisfying:} \quad & \boldsymbol{g}(\boldsymbol{x}) \leq \boldsymbol{0}, \\ \text{by changing:} \quad & \boldsymbol{x} \in \mathcal{X}, \end{aligned} \tag{II.7}$$

with objectives $\boldsymbol{f}(\boldsymbol{x}) \in \mathbb{R}^{m_1}$ and constraints $\boldsymbol{g}(\boldsymbol{x}) \in \mathbb{R}^{m_2}$, and where the design vector $\boldsymbol{x}$ lives in $\mathcal{X} \subset \mathbb{R}^n$. Note that, in the above and in the following, $\boldsymbol{g}(\boldsymbol{x}) \leq \boldsymbol{0}$ stands for $\forall i, g_i(\boldsymbol{x}) \leq 0$. More information on the product order is given in Section II–1.1.

The computation of $\boldsymbol{f}$ and $\boldsymbol{g}$ can also depends on a multitude of uncontrollable parameters (*e.g.* environmental, material, geometrical, ...), here denoted by $\boldsymbol{\xi} \in \Xi$.

These parameters can be considered uncertain, and the most straightforward setting is to consider them as aleatoric. This thesis is limited to this scope and does not deal with epistemic uncertainties, presented in the introductory chapter. In this context, $\boldsymbol{\xi}$, $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi})$ and $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\xi})$ are random variables. Equation (II.7) is not suitable anymore to represent the problem.

**Remark** *It is possible to relapse to a deterministic optimization problem* (II.7) *by fixing uncertainties* $\boldsymbol{\xi}$ *at some nominal values* $\boldsymbol{\xi}_0$*, namely:*

$$\begin{aligned} \text{minimize:} \quad & \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi}_0), \\ \text{satisfying:} \quad & \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\xi}_0) \leq \mathbf{0}, \\ \text{by changing:} \quad & \boldsymbol{x} \in \mathcal{X}. \end{aligned} \tag{II.8}$$

*In other words, by solving the deterministic optimization problem* (II.7)*, one implicitly assumes the parameters* $\boldsymbol{\xi}$ *to follow a Dirac distribution at* $\boldsymbol{\xi}_0$*.*

This thesis tackles uncertainty-based optimization problems in a measure-based setting, where the problem is solved by choosing specific statistics of the random outputs $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi})$ and $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\xi})$ to be optimized.

We denote in the following the uncertainty-driven objectives and constraints as robustness and reliability measures.

## II–1.3 Robustness and reliability measures

Writing $\boldsymbol{\rho}_f$ and $\boldsymbol{\rho}_g$ as some statistical measures on the objective function f and constraint g, respectively, the OUU problem can be formulated as follows:

$$\begin{aligned} \text{minimize:} \quad & \boldsymbol{\rho}_f(\boldsymbol{x}), \\ \text{satisfying:} \quad & \boldsymbol{\rho}_g(\boldsymbol{x}) \leq \mathbf{0}, \\ \text{by changing:} \quad & \boldsymbol{x} \in \mathcal{X}, \end{aligned} \tag{II.9}$$

As raised in the introductory chapter, using statistical measures as objectives is usually referred to as Robust Design Optimization (RDO), and these statistics ($\boldsymbol{\rho}_f$) are called robustness measures. Similarly, Reliability-Based Design Optimization (RBDO) deals with so-called reliability measures $\boldsymbol{\rho}_g$ as constraints.

In the following, the original functions $\boldsymbol{f}$ and $\boldsymbol{g}$ are gathered in a vector $\boldsymbol{q} \in \mathbb{R}^{m_1+m_2}$, that refers to Quantities of Interest (QoI). Namely,

$$\forall (\boldsymbol{x}, \boldsymbol{\xi}) \in \mathcal{X} \times \Xi, \ \boldsymbol{q}(\boldsymbol{x}, \boldsymbol{\xi}) = \begin{pmatrix} \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi}) \\ \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\xi}) \end{pmatrix}. \tag{II.10}$$

Similarly, $\boldsymbol{\rho}_f$ and $\boldsymbol{\rho}_g$ are gathered in a single vector $\boldsymbol{\rho} \in \mathbb{R}^{m_1+m_2}$,

$$\forall \boldsymbol{x} \in \mathcal{X}, \ \boldsymbol{\rho}(\boldsymbol{x}) = \begin{pmatrix} \boldsymbol{\rho}_f(\boldsymbol{x}) \\ \boldsymbol{\rho}_g(\boldsymbol{x}) \end{pmatrix}. \tag{II.11}$$

The most classical robustness and reliability measures for problem (II.9) are the following:

- Expectation $\rho(\boldsymbol{x}) = \mathbb{E}_{\xi}[q(\boldsymbol{x}, \boldsymbol{\xi})]$,
- Variance $\rho(\boldsymbol{x}) = \mathbb{V}_{\xi}[q(\boldsymbol{x}, \boldsymbol{\xi})]$,
- Worst case $\rho(\boldsymbol{x}) = \max_{\xi}[q(\boldsymbol{x}, \boldsymbol{\xi})]$
- Quantile $\rho(\boldsymbol{x}) = \mathrm{q}^{p}_{\xi}[q(\boldsymbol{x}, \boldsymbol{\xi})]$ (also called Value at Risk, VaR) ,
- Superquantile $\rho(\boldsymbol{x}) = \mathbb{E}_{\xi}\left[q(\boldsymbol{x}, \boldsymbol{\xi}) \mid q(\boldsymbol{x}, \boldsymbol{\xi}) \geq \mathrm{q}^{p}_{\xi}[q(\boldsymbol{x}, \boldsymbol{\xi})]\right]$ (also called Expected Value at Risk, EVaR).

Note that any combination of these may be used, and one has to choose which ones are most relevant for a given application.

In practice, robust objectives usually are either expectation measures, making sure that performance is optimized in average over the random input, or worst-case measures, that ensure the worst outcome to be the least detrimental possible. Note that worst-case measures historically are the first definition of robustness. They can be relaxed to quantile or superquantile measures, that take into account the probability distribution, contrarily to the worst case. Taguchi's robust optimization is a different approach that optimizes the mean performance (expectation measure) while simultaneously minimizing the associated variance.

Reliability constraints can impose hard constraint, where inequality $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\xi}) \leq \mathbf{0}$ must be verified for all values of $\boldsymbol{\xi} \in \Xi$, in which case the worst-case measure should be used. They are usually relaxed into probabilistic constraints, that ensures $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\xi}) \leq \mathbf{0}$ with a given probability $p$. This relation can be readily obtained by using the $p$-quantile as a reliability measure.

**Remark** *Note that the scope of this thesis is limited to the multi-objective case, $m_1 \leq 3$, so that the classical Pareto dominance rule remains relevant. The many-objective setting, $m_1 > 3$, requires some adjustment in the dominance criterion or some scalarization of the optimization problem, as presented in review [Ishibuchi et al., 2008].*



**Discussion** *Given the distribution of the random input variable $\boldsymbol{\xi}$, and some measures $\boldsymbol{\rho}$, optimization problem (II.9) can be solved with classical techniques. However, if the unitary evaluation of $\boldsymbol{q}$ is expensive, a compromise between accuracy and computational cost for evaluating $\boldsymbol{\rho}$ is required. How the approximation error on $\boldsymbol{\rho}$ can be integrated into the Pareto front computation is the subject of the next Section.*

## II–2 A probabilistic setting

As raised in the preceding Section, the objective and constraints values $\boldsymbol{\rho}$ can only be computed with a given level of accuracy. In this context, we propose to regard the unknown exact value $\boldsymbol{\rho}(\boldsymbol{x}_0)$ at any design $\boldsymbol{x}_0$ as a random variable $\mathbf{P}(\boldsymbol{x}_0)$.

By definition, given a probability space $(\Omega, \mathcal{F}, P)$, with $\Omega$ the set of outcomes, $\mathcal{F}$ the set of events and $P$ the probability function that assigns a probability to each event, a random field $F$ is defined as a collection of random variables $F(t)$ indexed on a topological space $T$:

$$F \equiv \big\{F(t) \mid t \in T\big\}. \tag{II.12}$$

We thus propose to approximate the measures $\boldsymbol{\rho}$ with a random field $\mathbf{P}$. We show here the extension of the classical Pareto dominance rule under a probabilistic perspective.

Several works have proposed to treat Pareto dominance in a probabilistic way, for example with uniform [Teich, 2001] or Gaussian [Hughes, 2001] assumptions. Some studies have notably been devoted to getting rid of such shape assumptions, through First-Order Reliability Method probability approximations [Coelho, 2014] or histogram reconstruction [Khosravi et al., 2018, Khosravi et al., 2019].

Section II–2.1 introduces the so-called probabilistic Pareto dominance, that can be relaxed for a practical purpose. Then in Section II–2.2, we propose the Pareto Optimal Probability (POP) metric and its numerical approximations. The POP notably permits to compare non-dominated elements quantitatively.

## II–2.1 Probabilistic Pareto dominance

Following ideas from [Coelho, 2014, Khosravi et al., 2018, Khosravi et al., 2019], we define a probabilistic Pareto dominance between two random vectors $\mathbf{A}$ and $\mathbf{B}$ in a constrained multi-objective setting as the probability $\mathbb{P}_{\mathbf{A},\mathbf{B}}\big[\mathbf{A} \succ_c \mathbf{B}\big]$, computed over the joint distribution of $\mathbf{A}$ and $\mathbf{B}$.

**Remark** *Again,* $\mathbf{A}$ *and* $\mathbf{B}$ *are random vectors in the objective and constraint space. They could be written as explicit images of some specific designs by the objective and constraint random field* $\mathbf{P}$*:* $\mathbf{A} = \mathbf{P}(\boldsymbol{x}_a)$ *and* $\mathbf{B} = \mathbf{P}(\boldsymbol{x}_b)$.

Formally, the unconstrained probability is simply probability for $\mathbf{A}$ to be lower than $\mathbf{B}$ in all dimensions, or equivalently the probability of the difference between $\mathbf{A}$ and $\mathbf{B}$ to be negative on all dimensions. This writes:

$$\mathbb{P}_{\mathbf{A},\mathbf{B}}\big[\mathbf{A} \succ \mathbf{B}\big] = \int_{\mathbb{R}^m_-} \phi_{\mathbf{A}-\mathbf{B}}(\boldsymbol{y})\mathrm{d}\boldsymbol{y} = \Phi_{\mathbf{A}-\mathbf{B}}(\mathbf{0}), \tag{II.13}$$

where $\phi_{\mathbf{A}-\mathbf{B}}$ is the Probability Density Function (PDF) of $\mathbf{A}-\mathbf{B}$ and $\Phi_{\mathbf{A}-\mathbf{B}}$ is the associated Cumulative Density Function (CDF). Formulation (II.13) is short and straightforward, by dealing with the distribution of the difference between $\mathbf{A}$ and $\mathbf{B}$. An equivalent formula can be written using the joint PDF $\phi_{\mathbf{A},\mathbf{B}}$ of $\mathbf{A}$ and $\mathbf{B}$:

$$\mathbb{P}_{\mathbf{A},\mathbf{B}}\big[\mathbf{A} \succ \mathbf{B}\big] = \iint_{\boldsymbol{a} \succ \boldsymbol{b}} \phi_{\mathbf{A},\mathbf{B}}(\boldsymbol{a},\boldsymbol{b})\mathrm{d}\boldsymbol{a}\mathrm{d}\boldsymbol{b}. \tag{II.14}$$

As for the constrained case, definition (II.4) shows that for $\boldsymbol{a}$ to dominate $\boldsymbol{b}$, either $\boldsymbol{b}$ must be in the failure domain, or $\boldsymbol{a}$ must be in the admissible domain and dominate $\boldsymbol{b}$ according to the classical Pareto dominance rule. The integral in (II.14) can be

split up depending on $\boldsymbol{b}$ belonging to the failure or admissible domain, and additional dominance constraint can be added in the latest case. This can be written as follows:

$$\mathbb{P}_{\mathbf{A},\mathbf{B}}\big[\mathbf{A} \succ_c \mathbf{B}\big] = \int_{\boldsymbol{b}\in\mathcal{F}} \phi_{\mathbf{B}}(\boldsymbol{b})\mathrm{d}\boldsymbol{b} + \iint_{\substack{\boldsymbol{a}\in\mathcal{A}\\ \boldsymbol{b}\in\mathcal{A}\\ \boldsymbol{a}\succ\boldsymbol{b}}} \phi_{\mathbf{A},\mathbf{B}}(\boldsymbol{a},\boldsymbol{b})\mathrm{d}\boldsymbol{a}\mathrm{d}\boldsymbol{b}. \tag{II.15}$$

**Remark** *Note that these formulations rely on the joint PDF $\phi_{\mathbf{A},\mathbf{B}}$, which intrinsically takes into account any dependency between* $\mathbf{A}$ *and* $\mathbf{B}$*. This notably differ from the independence assumptions in [Khosravi et al., 2018, Khosravi et al., 2019].*

Basing on the above formulas, a probabilistic constrained Pareto dominance rule can be formulated between random vectors:

$$\begin{aligned}
\mathbf{A} \underset{\epsilon}{\succ}_c \mathbf{B} &\iff \mathbb{P}_{\mathbf{A},\mathbf{B}}\big[\mathbf{A} \succ_c \mathbf{B}\big] \geq 1-\epsilon,\\
\mathbf{A} \underset{\epsilon}{\succ\!\!\succ}_c \mathbf{B} &\iff \mathbb{P}_{\mathbf{A},\mathbf{B}}\big[\mathbf{A} \succ\!\!\succ_c \mathbf{B}\big] \geq 1-\epsilon,\\
\mathbf{A} \underset{\epsilon}{\sim}_c \mathbf{B} &\iff \mathbf{A} \underset{\epsilon}{\not\succ}_c \mathbf{B} \text{ and } \mathbf{B} \underset{\epsilon}{\not\succ}_c \mathbf{A}.
\end{aligned} \tag{II.16}$$

Note that the above formulation (II.16) is actually the $\epsilon$-relaxed probabilistic constrained Pareto dominance. It tackles the case of random vectors with infinite support (*e.g.* Gaussian vectors), that prevent the probability to reach exactly 1. Such a relaxation has been proposed in [Khosravi et al., 2018]. If relaxation is to be removed, (II.16) relapses to the following formulation:

$$\begin{aligned}
\mathbf{A} \underset{0}{\succ}_c \mathbf{B} &\iff \mathbb{P}_{\mathbf{A},\mathbf{B}}\big[\mathbf{A} \succ_c \mathbf{B}\big] = 1,\\
\mathbf{A} \underset{0}{\succ\!\!\succ}_c \mathbf{B} &\iff \mathbb{P}_{\mathbf{A},\mathbf{B}}\big[\mathbf{A} \succ\!\!\succ_c \mathbf{B}\big] = 1,\\
\mathbf{A} \underset{0}{\sim}_c \mathbf{B} &\iff \mathbf{A} \underset{0}{\not\succ}_c \mathbf{B} \text{ and } \mathbf{B} \underset{0}{\not\succ}_c \mathbf{A}.
\end{aligned} \tag{II.17}$$



**Discussion** *The dominance rule defined in (II.16) gives a general setting for comparing random vectors in the objective and constraint space, regardless of their distribution shape and mutual dependencies. Notably, because we proposed to regard the approximation error of robustness and reliability measures as random variables, this probabilistic Pareto dominance rule gives a proper setting for comparing them. Note however that this approximation error will "blur" the Pareto front, meaning that a much higher number of elements will be non-dominated. Section II–2.2 tackles this phenomenon and propose a quantitative approach to ranking all Pareto-optimal solutions.*

## II–2.2 Pareto-Optimal Probability

Using the Pareto front definition (II.6), one can extract all non-dominated measures from a given set of random vectors with respect to the proposed probabilistic Pareto dominance rule (II.16). However, depending on the level of accuracy, the Pareto front can be very wide and contain most of these random vectors.

Thus, we propose to give a quantitative ranking between them using the concept of Pareto-Optimal Probability (POP). This simply refers to the probability for one random vector to be non-dominated, given its joint distribution with all other ones. Formally:

$$POP(\cdot) = \mathbb{P}_{\mathbf{A},\mathbf{B},\mathbf{C}}\big[\cdot \in \mathcal{P}(\{\mathbf{A},\mathbf{B},\mathbf{C}\})\big] = \int_{\{+,\times\}^3} \phi_{\mathbf{A},\mathbf{B},\mathbf{C}}(\boldsymbol{x}) \mathbb{1}_{\mathcal{P}(\{\mathbf{A},\mathbf{B},\mathbf{C}\})(\boldsymbol{x})}(\cdot) \mathrm{d}(\boldsymbol{x}), \tag{II.18}$$

where $\cdot$ can refer to any individual and $\mathbb{1}_{\mathcal{S}}$ is the indicator function of set $\mathcal{S}$.

An introductive example of this indicator is given in Section II–2.2.1 on a discrete test-case. It notably illustrates the importance of taking into account the joint distribution of all individuals. Several approximation metrics are then proposed in Section II–2.2.2 to keep the computational budget manageable. Finally, these metrics are quantitatively compared in Section II–2.2.3 on three test-cases featuring random vectors with uniform and independent distributions.

### II–2.2.1 Introductive example

We illustrate the POP indicator with the example proposed in Figure II.2 with three discrete random vectors **A**, **B** and **C**, that can take two different values, drawn with "plus" signs and crosses.

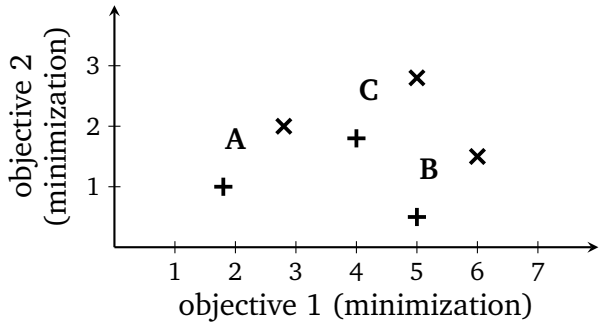



Figure II.2 Three random vectors, each with two possible outcomes + and ×.

The marginal probabilities are Bernoulli distribution with probability of 0.5. In other words, $\mathbb{P}[\mathbf{A} = +] = \mathbb{P}[\mathbf{B} = +] = \mathbb{P}[\mathbf{C} = +] = 0.5$, and same for ×.

With this test-case, we notably aim to highlight the impact of the joint Probability Density Function (PDF) on the POP indicator. To this extent, we introduce a first joint PDF $\phi^{(1)}_{\mathbf{A},\mathbf{B},\mathbf{C}}$ that considers **A**, **B** and **C** to be independent and a second one $\phi^{(2)}_{\mathbf{A},\mathbf{B},\mathbf{C}}$ that only allows two outcomes, $\mathbf{A} = \mathbf{B} = \mathbf{C} = +$ or $\mathbf{A} = \mathbf{B} = \mathbf{C} = \times$. The associated probability tables are depicted in Table II.2 and the Pareto front $\mathcal{P}(\{\mathbf{A},\mathbf{B},\mathbf{C}\})$ is made explicit for each scenario.

**Remark** *The notation* $\mathbf{A} = +$ *or* $\mathbf{A} = \times$ *used in the above is slightly abusive. However, because of its clarity and simplicity, it will be used for the rest of this example.*

The Pareto Optimal Probability (POP) of each individual can then be computed as the probability of belonging to the Pareto front $\mathcal{P}(\{\mathbf{A},\mathbf{B},\mathbf{C}\})$. These POP are reported in Table II.3. We denote $POP^{(1)}$ when $\phi^{(1)}_{\mathbf{A},\mathbf{B},\mathbf{C}}$ is considered and $POP^{(2)}$ when using $\phi^{(2)}_{\mathbf{A},\mathbf{B},\mathbf{C}}$.

| A | B | C | $\phi^{(1)}_{\mathbf{A},\mathbf{B},\mathbf{C}}$ | $\phi^{(2)}_{\mathbf{A},\mathbf{B},\mathbf{C}}$ | $\mathcal{P}(\{\mathbf{A},\mathbf{B},\mathbf{C}\})$ |
|---|---|---|---|---|---|
| + | + | + | 1/8 | 1/2 | $\{\mathbf{A},\mathbf{B}\}$ |
| + | + | × | 1/8 | 0 | $\{\mathbf{A},\mathbf{B}\}$ |
| + | × | + | 1/8 | 0 | $\{\mathbf{A}\}$ |
| + | × | × | 1/8 | 0 | $\{\mathbf{A}\}$ |
| × | + | + | 1/8 | 0 | $\{\mathbf{A},\mathbf{B},\mathbf{C}\}$ |
| × | + | × | 1/8 | 0 | $\{\mathbf{A},\mathbf{B}\}$ |
| × | × | + | 1/8 | 0 | $\{\mathbf{A},\mathbf{B},\mathbf{C}\}$ |
| × | × | × | 1/8 | 1/2 | $\{\mathbf{A},\mathbf{B}\}$ |

Table II.2 Probability tables of $\phi^{(1)}_{\mathbf{A},\mathbf{B},\mathbf{C}}$ and $\phi^{(2)}_{\mathbf{A},\mathbf{B},\mathbf{C}}$ and Pareto front of each scenario.

| Individual | $POP^{(1)}$ | $POP^{(2)}$ |
|---|---|---|
| **A** | 1 | 1 |
| **B** | 0.75 | 1 |
| **C** | 0.25 | 0 |

Table II.3 Pareto-Optimal Probabilities of **A**, **B** and **C** computed for both cases.

**Remark** *In the above example,* **A**, **B** *and* **C** *are compared in an unconstrained setting. However, computing of the Pareto front $\mathcal{P}(\cdot)$ using the constrained Pareto dominance rule naturally generalizes the Pareto Optimal Probability (POP) indicator to constrained optimization problems.*

### II–2.2.2 POP approximations

In the above, the POP computation was straightforward because of the low number of random vectors and their simple PDF. In real applications, one may have to compare hundreds of elements with unknown distribution shape. The computation of the integral (II.18) quickly becomes intractable and requires to be approximated numerically. Formally, with $\mathcal{I} = \{\mathbf{I}_k\}_{k=1}^{N}$ the set of all aleatory individuals, the POP of any element $\mathbf{I}_k$ is defined as such:

$$POP(\mathbf{I}_k) = \mathbb{P}_{\mathcal{I}}\big[\mathbf{I}_k \in \mathcal{P}(\mathcal{I})\big] = \mathbb{P}_{\mathcal{I}}\Big[\bigcap_{\substack{\mathbf{I}_j \in \mathcal{I} \\ j \neq k}} \mathbf{I}_j \not\prec_c \mathbf{I}_k\Big], \tag{II.19}$$

where $\mathbb{P}_{\mathcal{I}}$ is computed over the joint PDF of all elements of $\mathcal{I}$.

**Remark** *In Eq. II.19, the complexity of the computation comes not only from $\mathbb{P}_{\mathcal{I}}\big[\mathbf{I}_j \not\prec_c \mathbf{I}_k\big]$, that is intractable in the general case of unknown PDFs, but also from the intersection between the $N-1$ events.*
*The former issue is tackled in Chapters III and IV, by assuming a uniform distributions and proposing a non-parametric sampling-based approach respectively.*
*The latter one is managed through the computation of approximating POP metrics, that are presented hereafter.*

In the early 2000s, [Teich, 2001] and [Hughes, 2001] proposed the use of a probabilistic ranking as the fitness function in an Evolutionary Algorithm context. Notably, in [Teich, 2001], the score of an individual is computed by averaging the one-to-one domination probability with respect to all other ones. Using such an idea and replacing the domination probability with the POP, we write the $POP_{av}$ indicator:

$$POP_{av}(\mathbf{I}_k) = \frac{1}{N-1} \sum_{\substack{\mathbf{I}_j \in \mathcal{I} \\ j \neq k}} \mathbb{P}_{\mathbf{I}_j, \mathbf{I}_k}\left[\mathbf{I}_j \not\succ_c \mathbf{I}_k\right]. \tag{II.20}$$

By performing only one-to-one comparisons, the computational burden associated to (II.20) is very low. However, this formulation has some undesirable properties. Notably,

$$\begin{aligned} &\exists j, \mathbf{I}_j \underset{0}{\succ_c} \mathbf{I}_k \implies POP(\mathbf{I}_k) = 0, \\ \text{while} \quad &\exists j, \mathbf{I}_j \underset{0}{\succ_c} \mathbf{I}_k \not\Longrightarrow POP_{av}(\mathbf{I}_k) = 0. \end{aligned} \tag{II.21}$$

The $POP_{av}$ metric can be seen as the normalized $l_1$-norm of the vector which contains the one-to-one non-domination probability of an individual with respect to all other ones. Formally, as the events $\mathbf{A} \succ_c \mathbf{B}$ and $\mathbf{A} \not\succ_c \mathbf{B}$ are complementary, $POP_{av}$ can be written as follows:

$$POP_{av}(\mathbf{I}_k) = \frac{1}{N-1} \left\| \mathbf{1} - \boldsymbol{p}_{\mathcal{I}}(\mathbf{I}_k) \right\|_1, \tag{II.22}$$

where $\boldsymbol{p}_{\mathcal{I}}(\mathbf{I}_k)$ is the vector containing all elements $\left\{\mathbb{P}_{\mathbf{I}_j, \mathbf{I}_k}\left[\mathbf{I}_j \succ_c \mathbf{I}_k\right]\right\}_{j \neq k}$.

Following this interpretation, we propose a second metric, denoted as $POP_{min}$, that deals with the $l_\infty$-norm of $\boldsymbol{p}_{\mathcal{I}}(\mathbf{I}_k)$. Note that the norm is not computed on the same vector as the $\mathbf{1}$ is here taken out:

$$POP_{min}(\mathbf{I}_k) = 1 - \left\| \boldsymbol{p}_{\mathcal{I}}(\mathbf{I}_k) \right\|_\infty. \tag{II.23}$$

This metric simply writes as the following, taking the minimal POP value against all other individuals:

$$POP_{min}(\mathbf{I}_k) = \min_{\substack{\mathbf{I}_j \in \mathcal{I} \\ j \neq k}} \left( \mathbb{P}_{\mathbf{I}_j, \mathbf{I}_k}\left[\mathbf{I}_j \not\succ_c \mathbf{I}_k\right] \right), \tag{II.24}$$

which satisfies the desired implication:

$$\exists j, \mathbf{I}_j \underset{0}{\succ_c} \mathbf{I}_k \implies POP_{min}(\mathbf{I}_k) = 0. \tag{II.25}$$

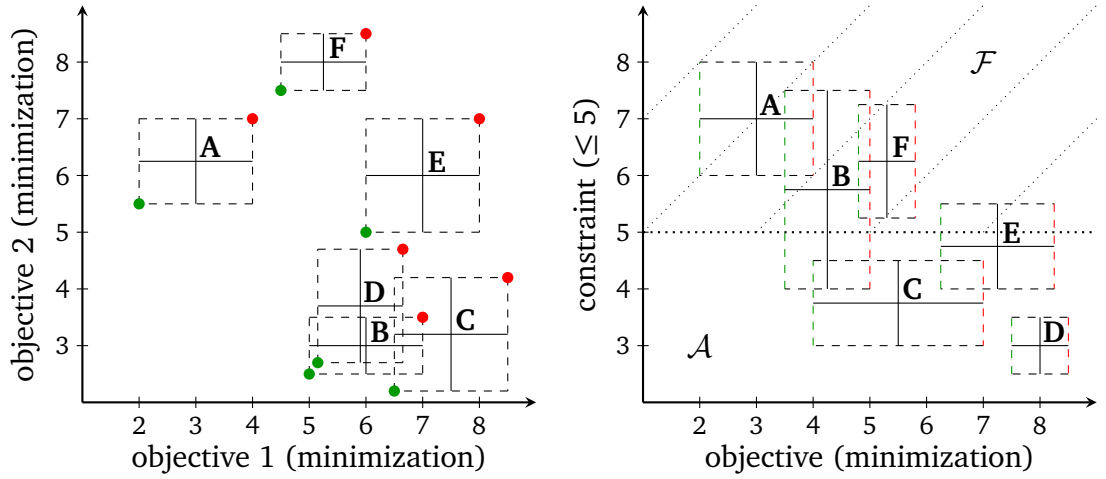
In summary, both $POP_{av}$ and $POP_{min}$ only require the computation of $\boldsymbol{p}_{\mathcal{I}}(\mathbf{I}_k)$, which lowers significantly the computational burden compared to the original POP, Eq. (II.19). However, $POP_{min}$ tends to behave as expected with dominated individuals, contrarily to $POP_{av}$.

### II–2.2.3 Application and comparison

These three metrics *i.e.* $POP$, $POP_{av}$ and $POP_{min}$ are compared on three simple examples to illustrate their behavior. Each example is constituted of a given set of random vectors and an optimization problem. Test-cases 1 and 2 are a bi-minimization problem (Figure II.3 (a)) and a constrained mono-objective minimization problem (Figure II.3 (b)) respectively. The third example is a bi-minimization problem with highly clustered boxes (Figure II.4), and highlights another good behavior of the $POP_{min}$ metric.

In these three examples, each random vector is assumed to follow a multi-dimensional uniform distribution and to be independent of all other individuals. Such an assumption will be at the core of the developments in Chapter III. Here it allows for a quick quantitative comparison of the proposed POP metrics.

**Examples 1 and 2** Figure II.3 depicts (a) a multi-objective and (b) a constrained optimization setting, with six random vectors, following independent uniform distributions. The different POP metrics are compared in Table II.4 for both examples. As raised in the preceding section, $POP_{av}$ is the only metric to return non-zero values for some strictly dominated individuals, such as **F** in Example 1 and **D** in Example 2. These individuals are even ranked above some non-dominated ones, which is very counter-intuitive and undesirable.




(a) (b)

Figure II.3 (a) Bi-minimization problem and (b) constrained mono-objective minimization. Best and worst outcomes in green and red respectively. Associated POP metrics in Table II.4. .

**Example 3** The last example features clustered individuals. As depicted in Figure II.4, individuals **C** to **F** share the same PDF. Intuitively, they are all equally close to the Pareto front. On the contrary, **B** is sensibly less efficient than **A**, hence further from the Pareto front. However, when computing the $POP$ of each individual, see Table II.5, it appears that **B** ranks higher that individuals **C** to **F**. The same applies for $POP_{av}$, but because $POP_{min}$ only keeps the minimal probability, clusters of individuals do not affect its value as much as the two other metrics. Hence, it gives a good and intuitive indicator of the closeness of each individual to the Pareto front.

| | Example 1 | | | Example 2 | | |
|---|---|---|---|---|---|---|
| | $POP$ | $POP_{av}$ | $POP_{min}$ | $POP$ | $POP_{av}$ | $POP_{min}$ |
| **A** | 1.0 | 1.0 | 1.0 | 0.0 | 0.0 | 0.0 |
| **B** | 0.902 | 0.98 | 0.912 | 0.254 | 0.279 | 0.254 |
| **C** | 0.39 | 0.828 | 0.419 | 0.724 | 0.943 | 0.746 |
| **D** | 0.621 | 0.924 | 0.622 | 0.0 | 0.628 | 0.0 |
| **E** | 0.017 | 0.508 | 0.07 | 0.022 | 0.483 | 0.031 |
| **F** | 0.0 | 0.735 | 0.0 | 0.0 | 0.0 | 0.0 |

Table II.4 POP comparison on examples 1 and 2 (Fig. fig:exPOP12)

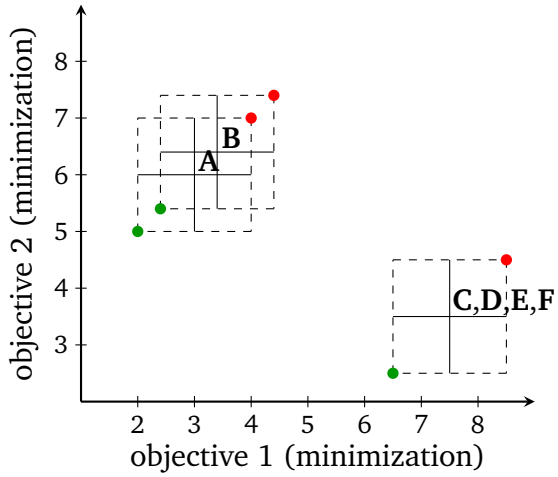



Figure II.4 Bi-minimization problem with boxes **C** to **F** superimposed. Best and worst outcomes in green and red respectively. Associated POP metrics in Table II.5.

| | Example 3 | | |
|---|---|---|---|
| | $POP$ | $POP_{av}$ | $POP_{min}$ |
| **A** | 0.898 | 0.98 | 0.898 |
| **B** | 0.538 | 0.908 | 0.538 |
| **C** | 0.521 | 0.85 | 0.75 |
| **D** | 0.521 | 0.85 | 0.75 |
| **E** | 0.521 | 0.85 | 0.75 |
| **F** | 0.521 | 0.85 | 0.75 |

Table II.5 POP comparison on example 3 (Fig. II.4)

**Discussion** *In this section, three metrics are presented for comparing random vectors in a constrained multi-objective setting. The exact POP computation is often intractable and sensitive to clustering of the individuals. The $POP_{av}$ indicator is much easier to compute but returns counter-intuitive, if not erroneous, ranking among the individuals. Hence, for its interpretability and low-computational burden, the $POP_{min}$ metric is systematically used in the following of this work.*

## II–3 The SAMATA algorithm

In this section, we propose a general technique for Optimization Under Uncertainty (OUU), that aims to integrate the concept of tunable accuracy within the OUU process. This concept has notably been presented in [Picheny et al., 2010a] and exploited in [Fusi and Congedo, 2016].

The core idea of the approach is to tune the accuracy of each computation of $\boldsymbol{\rho}$ to focus computational power on the most promising designs. Practically, robustness and reliability measures are computed with some evaluations of $\boldsymbol{q}$ with an estimation of their approximation error. These approximated measures are treated as a random vector and can be compared and ranked using the strategies proposed in Section II–2. The accuracy is then adaptively tuned by improving these approximations only on the most promising individuals. We refer to this approach as the Measure Approximation with Tunable Accuracy (MATA), as different levels of refinement are used to approximate the robustness and reliability measures, depending on their estimated performance.

Additionally, we construct a surrogate model on these measure approximations in the design space, denoted in the following as Surrogate-Assisting (SA) model. At any new design $\boldsymbol{x}$, this surrogate returns predictions of robustness and reliability measures, as well as an estimation of the prediction error. When this error is small enough, this prediction is returned to the optimization process bypassing the direct estimation of $\boldsymbol{\rho}$ through multiple evaluations of $\boldsymbol{q}$.

These two methods (SA and MATA) couple very efficiently to lower the cost of an uncertainty-based optimization process. The MATA technique allows focusing the refinements only on the promising designs while the SA strategy bypasses measure approximations and evaluations of $\boldsymbol{q}$ whenever the behavior of $\boldsymbol{\rho}$ is well captured.

Figure II.5 gives a flowchart of this strategy, which we refer to as the SAMATA strategy. The specific notations are presented in the following sections II–3.1 and II–3.2.

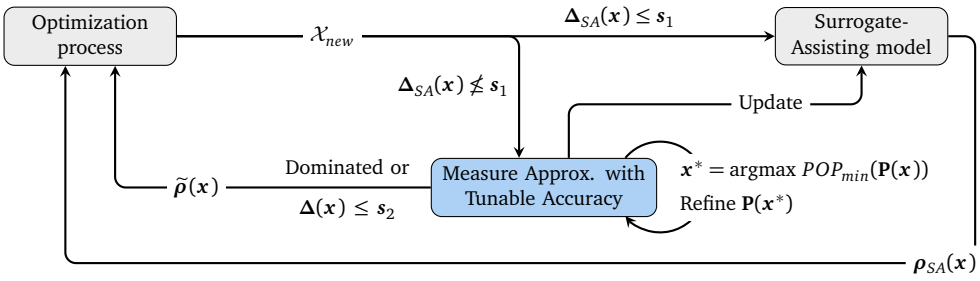



Figure II.5 SAMATA strategy flowchart.

### II–3.1 Surrogate-Assisting strategy

Surrogate-Assisting (SA) strategies are common practice for accelerating the optimization process. Such techniques are notably reviewed, within Evolutionary Algorithms (EA) in [Jin, 2011] and [Emmerich et al., 2002], and in a more general context under the name of Sequential Approximate Optimization (SAO) in [Nakayama et al., 2009]. The specific strategy chosen here is presented in details.

In the proposed approach, a surrogate model is constructed and updated throughout the optimization process directly on the robustness and reliability measures $\boldsymbol{\rho}$ in the design space. For each design $\boldsymbol{x}_i$ visited by the optimization process, we assume that an aleatoric approximation $\mathbf{P}(\boldsymbol{x}_i)$ is returned. The SA model then builds on the training set $\left\{\left(\boldsymbol{x}_i, \mathbf{P}(\boldsymbol{x}_i)\right)\right\}_i$. When enough information is gathered at a new visited design $\boldsymbol{x}_{new}$, the SA model bypasses the computation of the aleatoric approximation $\mathbf{P}(\boldsymbol{x}_{new})$ and returns the current prediction $\boldsymbol{\rho}_{SA}(\boldsymbol{x}_{new})$. One can expect this approach to be extensively exploited at the end of the optimization process when a lot of training points are available, and most new designs are in the optimal area.

The SA model must then be able to deal with aleatoric inputs, that can be considered affected by random noise. Notably, the model should take into account this noise for approximating its prediction error. One may recognize here an ideal application of the heteroscedastic Gaussian Processes presented in Section I. Specific SA models are proposed in Chapters III and IV to cope with different assumptions on the noise distribution shape.

We propose a general formulation, where the accuracy of the SA model at a new design $\boldsymbol{x}_{new}$ is defined as a vector $\boldsymbol{\Delta}_{SA}(\boldsymbol{x}_{new})$, so that the SA prediction $\boldsymbol{\rho}_{SA}(\boldsymbol{x}_{new})$ is returned to the optimization process whenever $\boldsymbol{\Delta}_{SA}(\boldsymbol{x}_{new}) \leq \boldsymbol{s}_1$, where $\boldsymbol{s}_1$ is a user-defined threshold chosen beforehand. Note that when the condition is not satisfied, an approximation $\mathbf{P}(\boldsymbol{x}_{new})$ must be computed, at the cost of some evaluation of $\boldsymbol{q}$. The couple $\left(\boldsymbol{x}_{new}, \mathbf{P}(\boldsymbol{x}_{new})\right)$ is then added to the training data and a chosen estimated value $\widetilde{\boldsymbol{\rho}}(\boldsymbol{x}_{new})$ (usually the mean of $\mathbf{P}(\boldsymbol{x}_{new})$) is returned to the optimizer.

To ensure the accuracy of the final results, we propose to carry out the whole optimization process several times with decreasing values of $\boldsymbol{s}_1$, by saving all training data $\left\{\left(\boldsymbol{x}_i, \mathbf{P}(\boldsymbol{x}_i)\right)\right\}_i$ from one iteration to another. This procedure allows getting several intermediate results, that compromise between computational cost and accuracy. With high $\boldsymbol{s}_1$ value, few training data are required, which keeps the number of evaluations of $\boldsymbol{q}$ quite low, and the SA model prediction $\boldsymbol{\rho}_{SA}$ is extensively exploited. On the contrary, with a small $\boldsymbol{s}_1$ value, many training data are required before employing the SA model, thus ensuring a better accuracy of $\boldsymbol{\rho}_{SA}$ anyway.

The structure of this approach is depicted in Algorithm II.1. The computation and refinement of the probabilistic approximations $\mathbf{P}(\boldsymbol{x})$ (lines 4 and 12 of the algorithm) are described in the following section II–3.2.

## II–3.2 Measure Approximation with Tunable Accuracy

At each optimization iteration, a probabilistic approximation $\mathbf{P}(\boldsymbol{x})$ of $\boldsymbol{\rho}(\boldsymbol{x})$ is computed for all designs $\boldsymbol{x}$ given by the optimization process. We propose to employ the iterative approach presented in [Fusi and Congedo, 2016], that put most of the computational effort on promising designs. Notably, we explicitly define a threshold $\boldsymbol{s}_2$ to control the accuracy of these Pareto-optima.

We denote $\mathcal{X}_c$ the set of visited designs throughout the optimization process. Namely, at each iteration, the new designs $\mathcal{X}_{new}$ explored by the optimization algorithm are added to $\mathcal{X}_c$ the set of designs to compute. Within this set, we denote by $\mathcal{X}_r$ the set of designs that must be further refined. These designs satisfy two criteria: (i) they belong to the current Pareto front $\mathcal{P}(\mathbf{P}(\mathcal{X}_c))$ and (ii) the user-defined accuracy threshold $\boldsymbol{s}_2$ is not reached. Formally:

$$\mathcal{X}_r = \left\{\boldsymbol{x} \in \mathcal{X}_c \mid \boldsymbol{\Delta}(\boldsymbol{x}) \not\leq \boldsymbol{s}_2 \;\&\; \mathbf{P}(\boldsymbol{x}) \in \mathcal{P}(\mathbf{P}(\mathcal{X}_c))\right\}, \tag{II.26}$$

**Algorithm II.1** Algorithm of the Surrogate-Assisting strategy

Set initial thresholds $\boldsymbol{s}_1$
Initialize the training set $\mathcal{T} = \left\{\left(\boldsymbol{x}_{train_i}, \mathbf{P}(\boldsymbol{x}_{train_i})\right)\right\}_i$ empty
**for** $k$ from 0 to $N_{threshold}$ **do**
    Refine approximations in $\mathcal{T}$ ▷ Alg. II.2
    Launch optimizer
    **while** Optimization running **do**
        Update the SA model on $\mathcal{T}$
        Get new designs $\mathcal{X}_{new}$ to visit
        $\mathcal{X}_{SA} = \left\{\boldsymbol{x} \in \mathcal{X}_{new} \mid \boldsymbol{\Delta}_{SA}(\boldsymbol{x}) \leq \boldsymbol{s}_1\right\}$
        $\mathcal{X}_c = \mathcal{X}_{new} \setminus \mathcal{X}_{SA}$
        $\forall \boldsymbol{x} \in \mathcal{X}_{SA}$, return $\boldsymbol{\rho}_{SA}(\boldsymbol{x})$ to the optimizer
        $\forall \boldsymbol{x} \in \mathcal{X}_c$, compute $\mathbf{P}(\boldsymbol{x})$ and return $\widetilde{\boldsymbol{\rho}}(\boldsymbol{x})$ to the optimizer ▷ Alg. II.2
        $\forall \boldsymbol{x} \in \mathcal{X}_c$, add $\left(\boldsymbol{x}, \mathbf{P}(\boldsymbol{x})\right)$ to $\mathcal{T}$
    **end while**
    Decrease $\boldsymbol{s}_1$
**end for**

where $\boldsymbol{\Delta}(\boldsymbol{x})$ is a chosen measure of the variability of $\mathbf{P}(\boldsymbol{x})$ (*e.g.* its standard deviation) and $\mathbf{P}(\mathcal{X}_c)$ corresponds to the image set $\left\{\mathbf{P}(\boldsymbol{x})\right\}_{\boldsymbol{x}\in\mathcal{X}_c}$. Note that if some designs in $\mathcal{X}_c$ have just been proposed by the optimization process, first approximations $\mathbf{P}$, and associated accuracy $\boldsymbol{\Delta}$, must be computed prior to assessing $\mathcal{X}_r$.

Finally, given this set of designs to refine, we propose to use the POP ranking presented in Section II–2.2 to select the very most promising design within $\mathcal{X}_r$. The variability of the associated approximation $\mathbf{P}(\boldsymbol{x})$ can then be reduced, at the cost of some evaluations of $\boldsymbol{q}$. In practice, the accuracy of the approximated robustness and reliability measures is hence improved at $\boldsymbol{x}^*$ where the $POP_{min}$ metric is maximized, as follows:

$$\boldsymbol{x}^* = \underset{\boldsymbol{x}\in\mathcal{X}_r}{\operatorname{argmax}}\ POP_{min}(\mathbf{P}(\boldsymbol{x})). \tag{II.27}$$

The MATA refinement technique, for a given set of designs $\mathcal{X}_c$ and a fixed threshold $\boldsymbol{s}_2$, is detailed in Algorithm II.2.

**Algorithm II.2** Algorithm of the Measure Approximation with Tunable Accuracy technique

Read threshold $\boldsymbol{s}_2$ and design set $\mathcal{X}_c$
Compute first approximations $\mathbf{P}(\boldsymbol{x})$ or retrieve the ones that already exist
Compute $\mathcal{P}(\mathbf{P}(\mathcal{X}_c))$ with formula (II.6)
Compute $\mathcal{X}_r$ with Equation (II.26)
**while** $\mathcal{X}_r$ is non-empty **do**
    Find $\boldsymbol{x}^*$ with Equation (II.27)
    Refine approximation $\mathbf{P}(\boldsymbol{x}^*)$ with some evaluations of $\boldsymbol{q}$
    Update $\mathcal{P}(\mathbf{P}(\mathcal{X}_c))$ and $\mathcal{X}_r$
**end while**
**return** $\mathbf{P}(\boldsymbol{x})$ for all $\boldsymbol{x}$ in $\mathcal{X}_c$

**Discussion** *In this section, the Surrogate-Assisting strategy and the Measure Approximation with Tunable Accuracy approach have been illustrated. The associated algorithm is made explicit, and the coupling is depicted in the flowchart Figure II.5. Note that throughout this section, SAMATA is presented in its most general form. More precisely, no computational details were given on:*

- *which optimization method is used,*
- *how to compute the probabilistic approximations* $\mathbf{P}(\boldsymbol{x})$*,*
- *how to refine these estimations,*
- *which metrics* $\widetilde{\boldsymbol{\rho}}(\boldsymbol{x})$ *and* $\boldsymbol{\Delta}(\boldsymbol{x})$ *are extracted from* $\mathbf{P}(\boldsymbol{x})$*,*
- *which SA model is constructed,*
- *and how* $\boldsymbol{\rho}_{SA}(\boldsymbol{x})$ *and* $\boldsymbol{\Delta}_{SA}(\boldsymbol{x})$ *are computed from the SA model.*

*All these computational choices will be described in the following Chapters III and IV.*

**Conclusion of the chapter** *This chapter has introduced the Optimization Under Uncertainty problem.*
*A probabilistic setting for constrained multi-objective optimization has been proposed to deal with approximation errors of the robustness and reliability measures. To this extent, the concept of Pareto dominance is introduced and extended to constrained optimization problems. The approximation errors on the objective and constraint values are regarded as random variables, for which a probabilistic Pareto dominance is proposed. This procedure allows for capturing the set of Pareto-optimal individuals. The Pareto-Optimal Probability is then proposed to rank all Pareto-optimal individuals. The associated computational burden is highlighted, and several approximate metrics are introduced. For its interpretability, the* $POP_{min}$ *metric is systematically used in the following of this work.*
*Finally, the Surrogate-Assisting Measure Approximation with Tunable Accuracy approach is proposed. Coupled with a multi-objective optimization technique, it permits to solve OUU problems where the objectives and constraints are numerically approximated integrands, which error is regarded as a random variable. In this context, high parsimony is achieved by only refining the most promising designs according to the POP, and by discarding all dominated individuals under the probabilistic Pareto dominance rule. Also, a Surrogate-Assisting strategy allows for eventually bypassing most of these integrand approximations.*
*Applicative formulations for applying SAMATA to real test-cases are proposed in the following chapters III and IV with different assumptions on the distribution of the error approximation.*

CHAPTER *III*

# Uniform approximations: Bounding Boxes

**Overview** *In this chapter, we apply the SAMATA strategy with a specific assumption on the probabilistic error approximations* $\mathbf{P}(\boldsymbol{x})$*. Specifically, these approximations are assumed to be independent of each other and to follow a uniform distribution on a Cartesian product of intervals. This choice is motivated by the fact that most of the time, only bounds are available to quantify the error of an uncertainty propagation process [Ferson et al., 2010, Barth, 2016]. In this context, the MATA strategy is simply referred to as the Bounding-Box approach (BBa), introduced in [Fusi and Congedo, 2016].*
*This approach is first employed in a noisy optimization context with tunable accuracy, where theoretical convergence is demonstrated under some assumptions and illustrated on artificially noised test-cases. Specific formulations are then proposed in the case of optimization under uncertainty. Their efficiency is assessed with respect to more conventional methods.*

**Outline** *Section III–1 introduces the basic definitions of Bounding-Boxes and Boxed Pareto dominance. These notions allow formulating a general framework for applying SAMATA within the Bounding-Box setting.*
*Then, Section III–2 studies the convergence properties of this Surrogate-Assisted Bounding-Box approach (SABBa) toward the Pareto-optimal area under some assumptions. To this extent, we highlight a specific set sequence that is proven to yield conservative approximations of the Pareto front.*
*In its most general form, SABBa allows tackling noisy optimization problems with tunable objectives and constraints accuracy. This kind of problem is studied in Section III–3, with artificially noised analytical test-cases, to picture the asymptotic behavior of SABBa against more classical techniques.*
*Finally, specific techniques are proposed in Section III–4 for computing robustness and reliability measures with tunable and refinable accuracy. These techniques permit to apply SABBa to Optimization Under Uncertainty (OUU) problems, allowing for analytical comparisons against more conventional methods.*

**Contribution** *The first contribution is a proper formalization and convergence study of the Bounding-Box approach, proposed in [Fusi and Congedo, 2016]. The Surrogate-Assisting strategy is made explicit and shown to naturally and efficiently couple with the Bounding-Box approach.*
*The second contribution is the use of GP-based formulations for applying SABBa to the uncertainty-based optimization setting. This strategy is quantitatively compared with classical techniques and shows both great parsimony and robustness.*

## III–1 Bounding-Box context

This section illustrates the concept of Bounding-Box for uniform error approximations, formulates the associated strategy called SABBa.

## III–1.1 Definition of a Bounding-Box

Let us denote by $\widetilde{\boldsymbol{\rho}}^l(\boldsymbol{x})$ the approximated value returned when computing $\boldsymbol{\rho}(\boldsymbol{x})$ for a given design $\boldsymbol{x}$ with an accuracy level $l$. Let us further assume that the approximation error $\boldsymbol{\varepsilon}^l(\boldsymbol{x}) = \boldsymbol{\rho}(\boldsymbol{x}) - \widetilde{\boldsymbol{\rho}}^l(\boldsymbol{x})$ can be estimated conservatively by $\overline{\boldsymbol{\varepsilon}}^l(\boldsymbol{x})$, meaning that $|\boldsymbol{\varepsilon}^l(\boldsymbol{x})| \leq \overline{\boldsymbol{\varepsilon}}^l(\boldsymbol{x})$. This corresponds to getting bounds on the approximated value, that ensures $\boldsymbol{\rho} \in \left[\widetilde{\boldsymbol{\rho}}^l - \overline{\boldsymbol{\varepsilon}}^l, \widetilde{\boldsymbol{\rho}}^l + \overline{\boldsymbol{\varepsilon}}^l\right]$. Figure III.1 represents this $m$-dimensional product of intervals containing $\boldsymbol{\rho}$, which is called a Bounding-Box. The approximation of $\boldsymbol{\rho}(\boldsymbol{x})$ defined in Section II–2 is here denoted by $\mathcal{B}\left(\widetilde{\boldsymbol{\rho}}^l, \overline{\boldsymbol{\varepsilon}}^l\right)$ using the box notation from Definition 6.

**Definition 6.** A $m$-dimensional box is defined by its center and half-width vectors as follows:

$$\mathcal{B}(\boldsymbol{a}, \boldsymbol{r}) = \left\{\boldsymbol{b} \in \mathbb{R}^m \mid \boldsymbol{b} \in [\boldsymbol{a} - \boldsymbol{r}, \boldsymbol{a} + \boldsymbol{r}]\right\} \in \wp_{\mathcal{B}}(\mathbb{R}^m) \subset \wp(\mathbb{R}^m) \tag{III.1}$$

$\forall (\boldsymbol{a}, \boldsymbol{r}) \in \mathbb{R}^m \times \mathbb{R}^m_+$, and where $\wp_{\mathcal{B}}(\mathbb{R}^m)$ represents the set of all Cartesian products of intervals in $\mathbb{R}^m$, which is a subset of $\wp(\mathbb{R}^m)$ the power set of $\mathbb{R}^m$.

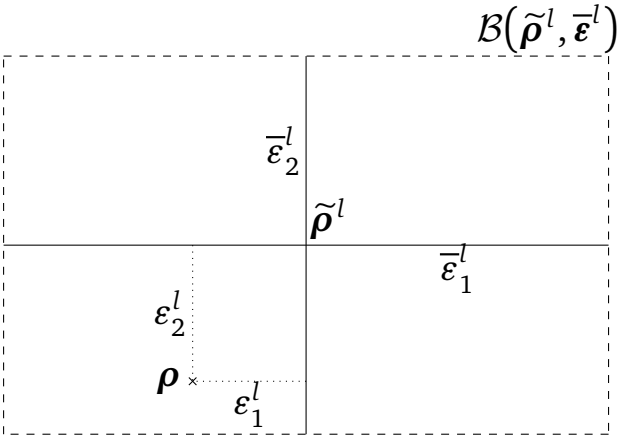



Figure III.1 Bounding-Box approximation

**Remark** *One can note that boxes relapse properly to points when their width goes to zero,*

$$\mathcal{B}(\boldsymbol{a}, \boldsymbol{0}) \equiv \boldsymbol{a}. \tag{III.2}$$

*In the following, to lighten the notation, for any $\boldsymbol{f} : \mathcal{X} \to \mathbb{R}^m$, we set $\forall (\boldsymbol{x}, \boldsymbol{r}) \in \mathcal{X} \times \mathbb{R}^m$,*

$$\mathcal{B}_{\boldsymbol{f}}(\boldsymbol{x}, \boldsymbol{r}) \equiv \mathcal{B}\left(\boldsymbol{f}(\boldsymbol{x}), \boldsymbol{r}\right). \tag{III.3}$$

## III–1.2 Boxed Pareto dominance

Following [Mlakar et al., 2014] and [Fusi and Congedo, 2016], the application of the non-relaxed probabilistic Pareto dominance rule defined in Eq. (II.17) to uniform distributions is referred to as the Boxed Pareto dominance. Note that relaxation is not required here because Bounding-Box approximations have a finite support.

**Remark** *In the following, all relations are given between elements $\boldsymbol{a}$ and $\boldsymbol{b}$ of the objective and constraint space.*

Instead of computing a probability of dominance over all outcomes of two boxes $\mathcal{B}(\boldsymbol{a},\boldsymbol{r})$ and $\mathcal{B}(\boldsymbol{b},\boldsymbol{r}')$, the Boxed Pareto dominance can be regarded as a classical dominance rule between the least performing point of $\mathcal{B}(\boldsymbol{a},\boldsymbol{r})$ and the best outcome of $\mathcal{B}(\boldsymbol{b},\boldsymbol{r}')$. The unconstrained dominance rule is made explicit in Definition 7.

**Definition 7.** For two boxes $\big(\mathcal{B}(\boldsymbol{a},\boldsymbol{r}),\mathcal{B}(\boldsymbol{b},\boldsymbol{r}')\big) \in \wp_{\mathcal{B}}(\mathbb{R}^m)^2$,

$$\begin{aligned}
\mathcal{B}(\boldsymbol{a},\boldsymbol{r}) \underset{0}{\succ} \mathcal{B}(\boldsymbol{b},\boldsymbol{r}') &\iff \forall j \in [\![1,m]\!],\ a_j + r_j \leq b_j - r'_j \quad \text{and} \\
&\qquad \exists j \in [\![1,m]\!],\ a_j + r_j < b_j - r'_j, \\
\mathcal{B}(\boldsymbol{a},\boldsymbol{r}) \underset{0}{\succcurlyeq} \mathcal{B}(\boldsymbol{b},\boldsymbol{r}') &\iff \forall j \in [\![1,m]\!],\ a_j + r_j < b_j - r'_j, \\
\mathcal{B}(\boldsymbol{a},\boldsymbol{r}) \underset{0}{\sim} \mathcal{B}(\boldsymbol{b},\boldsymbol{r}') &\iff \mathcal{B}(\boldsymbol{a},\boldsymbol{r}) \underset{0}{\not\succ} \mathcal{B}(\boldsymbol{b},\boldsymbol{r}') \text{ and } \mathcal{B}(\boldsymbol{b},\boldsymbol{r}') \underset{0}{\not\succ} \mathcal{B}(\boldsymbol{a},\boldsymbol{r}).
\end{aligned} \tag{III.4}$$

The notation $\underset{0}{\succ}$ specifies that the dominance rule is not relaxed ($\epsilon = 0$).

**Remark** *It can be noted that the relation between $\mathcal{B}(\boldsymbol{a},\boldsymbol{r})$ and $\mathcal{B}(\boldsymbol{b},\boldsymbol{r}')$ is the same as between $\mathcal{B}(\boldsymbol{a},\boldsymbol{r}')$ and $\mathcal{B}(\boldsymbol{b},\boldsymbol{r})$.*

This Boxed Pareto dominance can be extended to the constrained optimization setting, similarly to (II.4), by checking whether $\mathcal{B}(\boldsymbol{a},\boldsymbol{r})$ and $\mathcal{B}(\boldsymbol{b},\boldsymbol{r}')$ belong to the admissible or failure set.

Note that the non-relaxed Pareto dominance rule (II.17) requires any outcome of an individual to dominate every outcome of the other one. In this context, belonging to the admissible $\mathcal{A}$ or failure $\mathcal{F}$ sets must be verified for all outcomes of an individual. Thus, the boxed admissible and failure sets $\mathcal{A}_{\mathcal{B}}$ and $\mathcal{F}_{\mathcal{B}}$ are defined as follows:

$$\begin{aligned}
\mathcal{A}_{\mathcal{B}} &= \big\{\mathcal{B} \in \wp_{\mathcal{B}}(\mathbb{R}^m) \mid \forall \boldsymbol{a} \in \mathcal{B},\ \boldsymbol{a} \in \mathcal{A}\big\}, \\
\mathcal{F}_{\mathcal{B}} &= \big\{\mathcal{B} \in \wp_{\mathcal{B}}(\mathbb{R}^m) \mid \forall \boldsymbol{a} \in \mathcal{B},\ \boldsymbol{a} \in \mathcal{F}\big\}.
\end{aligned} \tag{III.5}$$

These sets can also be written as follows, using the best and worst outcomes:

$$\begin{aligned}
\mathcal{A}_{\mathcal{B}} &= \big\{\mathcal{B}(\boldsymbol{a},\boldsymbol{r}) \in \wp_{\mathcal{B}}(\mathbb{R}^m) \mid \boldsymbol{a} + \boldsymbol{r} \in \mathcal{A}\big\}, \\
\mathcal{F}_{\mathcal{B}} &= \big\{\mathcal{B}(\boldsymbol{a},\boldsymbol{r}) \in \wp_{\mathcal{B}}(\mathbb{R}^m) \mid \boldsymbol{a} - \boldsymbol{r} \in \mathcal{F}\big\}.
\end{aligned} \tag{III.6}$$

This formulation reveals that the failure set is not always complementary to the admissible set. Intuitively, when dealing with random objective and constraint values such as boxes, one can be only halfway in $\mathcal{A}$ and the other half in $\mathcal{F}$.

The constrained Boxed Pareto dominance is then provided in Definition 8.

**Definition 8.** For two boxes $\big(\mathcal{B}(\boldsymbol{a},\boldsymbol{r}),\mathcal{B}(\boldsymbol{b},\boldsymbol{r}')\big) \in \wp_{\mathcal{B}}(\mathbb{R}^m)^2$, and denoting by $\cdot_f$ the objective values,

$$\begin{aligned}
\mathcal{B}(\boldsymbol{a},\boldsymbol{r}) \underset{0}{\succ}_c \mathcal{B}(\boldsymbol{b},\boldsymbol{r}') &\iff \mathcal{B}(\boldsymbol{a}_f,\boldsymbol{r}_f) \underset{0}{\succ} \mathcal{B}(\boldsymbol{b}_f,\boldsymbol{r}'_f) \text{ and } \mathcal{B}(\boldsymbol{a},\boldsymbol{r}) \in \mathcal{A}_{\mathcal{B}} \quad \text{or} \\
&\qquad \mathcal{B}(\boldsymbol{b},\boldsymbol{r}') \in \mathcal{F}_{\mathcal{B}}, \\
\mathcal{B}(\boldsymbol{a},\boldsymbol{r}) \underset{0}{\succcurlyeq}_c \mathcal{B}(\boldsymbol{b},\boldsymbol{r}') &\iff \mathcal{B}(\boldsymbol{a}_f,\boldsymbol{r}_f) \underset{0}{\succcurlyeq} \mathcal{B}(\boldsymbol{b}_f,\boldsymbol{r}'_f) \text{ and } \mathcal{B}(\boldsymbol{a},\boldsymbol{r}) \in \mathcal{A}_{\mathcal{B}} \quad \text{or} \\
&\qquad \mathcal{B}(\boldsymbol{b},\boldsymbol{r}') \in \mathcal{F}_{\mathcal{B}}, \\
\mathcal{B}(\boldsymbol{a},\boldsymbol{r}) \underset{0}{\sim}_c \mathcal{B}(\boldsymbol{b},\boldsymbol{r}') &\iff \mathcal{B}(\boldsymbol{a},\boldsymbol{r}) \underset{0}{\not\succ}_c \mathcal{B}(\boldsymbol{b},\boldsymbol{r}') \text{ and } \mathcal{B}(\boldsymbol{b},\boldsymbol{r}') \underset{0}{\not\succ}_c \mathcal{B}(\boldsymbol{a},\boldsymbol{r}),
\end{aligned} \tag{III.7}$$

**Remark** *Again, the consistency between the two domination rules when the size of the boxes tends to zero is verified:*

$$\forall \boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^m,\ \mathcal{B}(\boldsymbol{a}, 0) \underset{0}{\succ_c} \mathcal{B}(\boldsymbol{b}, 0) \iff \boldsymbol{a} \succ_c \boldsymbol{b}, \tag{III.8}$$

Two examples are given in Figure III.2, with four boxes to be compared in (a) a bi-minimization problem and (b) a constrained mono-objective optimization problem. First, Table III.1 depicts which boxes of the constrained example III.2(b) belong to the boxed admissible and failure sets $\mathcal{A}_{\mathcal{B}}$ and $\mathcal{F}_{\mathcal{B}}$. Then, using the Boxed Pareto dominances from Definitions 7 and 8, relations between the four boxes are summarized in Table III.2.

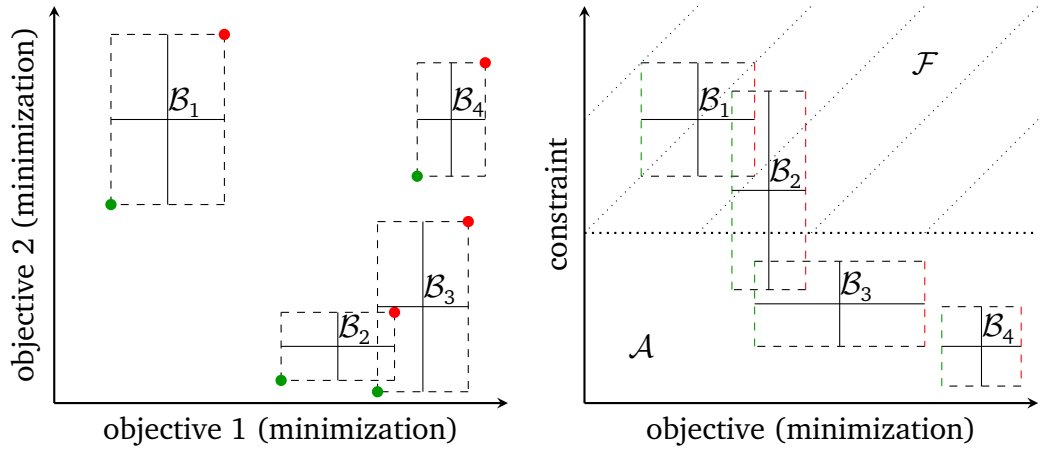



Figure III.2 Comparison of 4 boxes in (a) a bi-minimization case and (b) a constrained mono-minimization one. Best and worst outcomes in green and red, respectively.

| | $\mathcal{B}_1$ | $\mathcal{B}_2$ | $\mathcal{B}_3$ | $\mathcal{B}_4$ |
|---|---|---|---|---|
| $\in \mathcal{A}_{\mathcal{B}}$ | ✗ | ✗ | ✓ | ✓ |
| $\in \mathcal{F}_{\mathcal{B}}$ | ✓ | ✗ | ✗ | ✗ |

Table III.1 Membership in $\mathcal{A}_{\mathcal{B}}$ and $\mathcal{F}_{\mathcal{B}}$ for all boxes from Fig. III.2(b).

In Figure III.2, $\mathcal{B}_1$ is the only strictly dominated box, that lies entirely in $\mathcal{F}$. Hence, $\mathcal{B}_1 \in \mathcal{F}_{\mathcal{B}}$, as can be seen in Table III.1. On the contrary, $\mathcal{B}_3$ and $\mathcal{B}_4$ lie entirely in $\mathcal{A}$ and thus belong in $\mathcal{A}_{\mathcal{B}}$. Finally, $\mathcal{B}_2$ belong neither in $\mathcal{A}_{\mathcal{B}}$ nor $\mathcal{F}_{\mathcal{B}}$ because it overlaps both $\mathcal{A}$ and $\mathcal{F}$.

**Remark** *Note that here, Pareto fronts are sets of non-dominated boxes. They are defined using the non-relaxed probabilistic Pareto dominance, that has been derived in the boxed settings in Definition 8. Formally, with $\mathcal{Y}_{\mathcal{B}}$ a set of boxes, the Pareto front is defined as follows:*

$$\mathcal{P}(\mathcal{Y}_{\mathcal{B}}) = \left\{ \mathcal{B} \in \mathcal{Y}_{\mathcal{B}} \mid \nexists \mathcal{B}' \in \mathcal{Y}_{\mathcal{B}},\ \mathcal{B}' \underset{0}{\succ_c} \mathcal{B} \right\}. \tag{III.9}$$

*The boxed Pareto fronts are made explicit on the last line of Table III.2.*

| | Bi-minimization Fig. III.2(a) | Constrained minimization Fig. III.2(b) |
|---|---|---|
| $\mathcal{B}_1, \mathcal{B}_2$ | $\mathcal{B}_1 \underset{0}{\sim_c} \mathcal{B}_2$ | $\mathcal{B}_1 \underset{0}{\prec\!\!\prec_c} \mathcal{B}_2$ |
| $\mathcal{B}_1, \mathcal{B}_3$ | $\mathcal{B}_1 \underset{0}{\sim_c} \mathcal{B}_3$ | $\mathcal{B}_1 \underset{0}{\prec\!\!\prec_c} \mathcal{B}_3$ |
| $\mathcal{B}_1, \mathcal{B}_4$ | $\mathcal{B}_1 \underset{0}{\sim_c} \mathcal{B}_4$ | $\mathcal{B}_1 \underset{0}{\prec\!\!\prec_c} \mathcal{B}_4$ |
| $\mathcal{B}_2, \mathcal{B}_3$ | $\mathcal{B}_2 \underset{0}{\sim_c} \mathcal{B}_3$ | $\mathcal{B}_2 \underset{0}{\sim_c} \mathcal{B}_3$ |
| $\mathcal{B}_2, \mathcal{B}_4$ | $\mathcal{B}_2 \underset{0}{\succ\!\!\succ_c} \mathcal{B}_4$ | $\mathcal{B}_2 \underset{0}{\sim_c} \mathcal{B}_4$ |
| $\mathcal{B}_3, \mathcal{B}_4$ | $\mathcal{B}_3 \underset{0}{\sim_c} \mathcal{B}_4$ | $\mathcal{B}_3 \underset{0}{\succ\!\!\succ_c} \mathcal{B}_4$ |
| $\mathcal{P}\left(\{\mathcal{B}_i\}_{i=1}^4\right)$ | $\{\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3\}$ | $\{\mathcal{B}_2, \mathcal{B}_3\}$ |

Table III.2 All relations between the four boxes in both examples from Figure III.2. Pareto fronts are made explicit on the last line.

## III–1.3 SABBa general flowchart

From the Surrogate-Assisting Measure Approximation with Tunable Accuracy (SAMATA) introduced in Section II–3, we formulate here the particular case of independent uniform approximations, referred to as the Surrogate-Assisting Bounding-Box approach (SABBa).

This approach can be summarized in few bullet points. When a new set of designs $\mathcal{X}_{new}$ is proposed by the optimization algorithm:

- The error approximation $\overline{\boldsymbol{\varepsilon}}_{SA}(\boldsymbol{x})$ of the current SA model is compared to a user-defined threshold $\boldsymbol{s}_1$ at each design $\boldsymbol{x}$ in $\mathcal{X}_{new}$. For all designs satisfying $\overline{\boldsymbol{\varepsilon}}_{SA}(\boldsymbol{x}) \leq \boldsymbol{s}_1$, the prediction provided by the SA surrogate $\boldsymbol{\rho}_{SA}(\boldsymbol{x})$ is returned to the optimization process.
- For all designs that did not satisfy the $\boldsymbol{s}_1$ threshold, a first estimation $\widetilde{\boldsymbol{\rho}}^0(\boldsymbol{x})$ is computed, alongside an error approximation $\overline{\boldsymbol{\varepsilon}}^0(\boldsymbol{x})$. We set here $l = 0$.
- For all boxes $\mathcal{B}(\widetilde{\boldsymbol{\rho}}^l(\boldsymbol{x}), \overline{\boldsymbol{\varepsilon}}^l(\boldsymbol{x}))$ that either (i) are dominated or (ii) satisfy $\overline{\boldsymbol{\varepsilon}}^l(\boldsymbol{x}) \leq \boldsymbol{s}_2$, the estimation $\widetilde{\boldsymbol{\rho}}^l(\boldsymbol{x})$ is returned to the optimizer.
- If one of the previous condition is not satisfied for some designs, finer approximations $\widetilde{\boldsymbol{\rho}}^{l+1}(\boldsymbol{x}^*)$ and $\overline{\boldsymbol{\varepsilon}}^{l+1}(\boldsymbol{x}^*)$ are computed on a set $\mathcal{X}^*$ of designs. The order and number of designs in $\mathcal{X}^*$ can be managed with different strategies. This step is repeated until all criteria from the third bullet point are met.

**Remark** *In the above, the notations are intentionally lightened. One should read the accuracy $l$ as $l(\boldsymbol{x})$, so that each design is computed with a given accuracy. When refinement is required at $\boldsymbol{x}^*$, $l(\boldsymbol{x}^*)$ is incremented by one and $\forall \boldsymbol{x} \neq \boldsymbol{x}^*$, $l(\boldsymbol{x})$ is kept constant.*

**Remark** *The accuracy metrics $\boldsymbol{\Delta}(\boldsymbol{x})$ and $\boldsymbol{\Delta}_{SA}(\boldsymbol{x})$ from Equation (II.26) are here written $\overline{\boldsymbol{\varepsilon}}^l(\boldsymbol{x})$ and $\overline{\boldsymbol{\varepsilon}}_{SA}(\boldsymbol{x})$ respectively.*

This strategy can be pictured using the flowchart from Section II–3 using Bounding-Box notations. The updated flowchart is represented in Figure III.3.

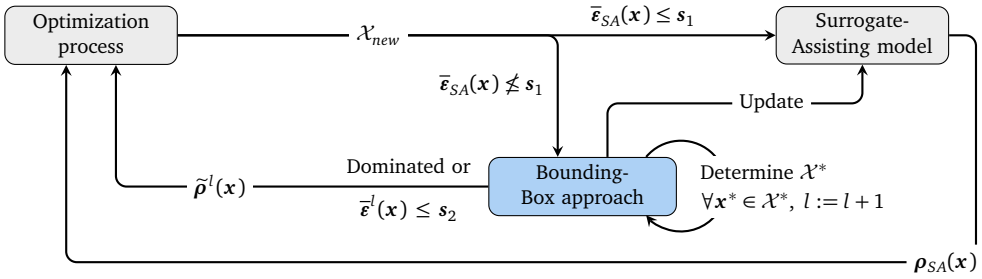



Figure III.3 SABBa flowchart.

**Discussion** *In this section, the Bounding-Box setting is formally introduced, and the Boxed Pareto dominance is derived from the general probabilistic Pareto dominance definition. Basing on the SAMATA algorithm presented in the preceding chapter, a general framework for applying SABBa on test-cases with objectives and constraints that can only be approximated with tunable and refinable accuracy is presented.*

# III–2 Theoretical considerations

This section provides some analyses over the convergence of the proposed strategy. In particular, SABBa is shown to converge towards the exact Pareto-optimal area under some assumptions.

Practically, Section III–2.1 defines the Pareto-optimal sets, both in the classical and in the Bounding-Box context. Section III–2.2 then formally introduces the so-called Bounding-Box approach, which consists in a recursive refinement scheme on the set of Pareto-optimal designs. Finally, Section III–2.3 illustrates the convergence of this approach under some assumptions and Section III–2.4 shows that coupling with the Surrogate-Assisting strategy does not affect the overall behavior.

## III–2.1 Pareto-optimal sets

The Pareto front of a problem has been defined in Equation (II.6) with the classical Pareto dominance rule and in Equation (III.9) with the constrained boxed Pareto dominance. Formal definitions of the pre-image of the Pareto front $\mathcal{X}_{\mathcal{P}}$ in the context of Bounding-Boxes are presented in this section. These designs are referred to as Pareto-optimal designs.

**Definition 9.** With $\boldsymbol{\rho}$ the objective and constraint functions, let us define $\forall i,\ \boldsymbol{x}_i \in \mathcal{X}$ and $\boldsymbol{r}_i \in \mathbb{R}^m$,

$$\begin{aligned}\mathcal{X}^{\boldsymbol{\rho}}_{\mathcal{P}}\Big(\{\boldsymbol{x}_i\}_{i=1}^N\Big) &= \Big\{\boldsymbol{x}_i,\ i \in [\![1,N]\!] \;\Big|\; \boldsymbol{\rho}(\boldsymbol{x}_i) \in \mathcal{P}\Big(\{\boldsymbol{\rho}(\boldsymbol{x}_i)\}_{i=1}^N\Big)\Big\},\\ \mathcal{X}^{\boldsymbol{\rho}}_{\mathcal{P}_{\mathcal{B}}}\Big(\{(\boldsymbol{x}_i,\boldsymbol{r}_i)\}_{i=1}^N\Big) &= \Big\{\boldsymbol{x}_i,\ i \in [\![1,N]\!] \;\Big|\; \mathcal{B}_{\boldsymbol{\rho}}(\boldsymbol{x}_i,\boldsymbol{r}_i) \in \mathcal{P}\Big(\{\mathcal{B}_{\boldsymbol{\rho}}(\boldsymbol{x}_i,\boldsymbol{r}_i)\}_{i=1}^N\Big)\Big\},\end{aligned} \tag{III.10}$$

the Pareto-optimal designs $\mathcal{X}^{\boldsymbol{\rho}}_{\mathcal{P}}$ in the classical and $\mathcal{X}^{\boldsymbol{\rho}}_{\mathcal{P}_{\mathcal{B}}}$ in the boxed settings respectively.

**Remark** $\mathcal{X}^{\rho}_{\mathcal{P}_{\mathcal{B}}}\left(\left\{\left(\boldsymbol{x}_i, \boldsymbol{r}_i\right)\right\}_{i=1}^{N}\right)$ *can be explicitly defined as follows:*

$$\boldsymbol{x}_i \in \mathcal{X}^{\rho}_{\mathcal{P}_{\mathcal{B}}}\left(\left\{\left(\boldsymbol{x}_i, \boldsymbol{r}_i\right)\right\}_{i=1}^{N}\right) \iff \mathcal{B}_{\rho}(\boldsymbol{x}_i, \boldsymbol{r}_i) \notin \mathcal{F}_{\mathcal{B}} \tag{III.11}$$
$$\text{and } \mathcal{B}_{\rho}(\boldsymbol{x}_k, \boldsymbol{r}_k) \in \mathcal{A}_{\mathcal{B}} \Rightarrow \boldsymbol{\rho}_f(\boldsymbol{x}_k) + \boldsymbol{r}_{f_k} \not\leq \boldsymbol{\rho}_f(\boldsymbol{x}_i) - \boldsymbol{r}_{f_i},$$

*where again* $\boldsymbol{\rho}_f$ *refers to the objective values.*

**Proposition 1.** *Note that* $\forall \left\{\left(\boldsymbol{x}_i, \boldsymbol{r}_i\right)\right\}_{i=1}^{N} \in \left(\mathcal{X} \times \mathbb{R}^m\right)^N$,

$$\mathcal{X}^{\rho}_{\mathcal{P}}\left(\left\{\boldsymbol{x}_i\right\}_{i=1}^{N}\right) \subseteq \left\{\boldsymbol{x}_i\right\}_{i=1}^{N} \text{ and } \mathcal{X}^{\rho}_{\mathcal{P}_{\mathcal{B}}}\left(\left\{\left(\boldsymbol{x}_i, \boldsymbol{r}_i\right)\right\}_{i=1}^{N}\right) \subseteq \left\{\boldsymbol{x}_i\right\}_{i=1}^{N}. \tag{III.12}$$

*Proof.* The proof here is trivial. By definition,

$$\mathcal{X}^{\rho}_{\mathcal{P}}\left(\left\{\boldsymbol{x}_i\right\}_{i=1}^{N}\right) = \left\{\boldsymbol{x}_i,\ i \in [\![1, N]\!] \mid \text{non-dominated}\right\} \subseteq \left\{\boldsymbol{x}_i,\ i \in [\![1, N]\!]\right\} \equiv \left\{\boldsymbol{x}_i\right\}_{i=1}^{N}.$$

The same can be said for the boxed Pareto optima. □

Note that the inclusion becomes strict whenever a design does not satisfy the non-domination condition.

The continuous extension of the Pareto front is also introduced in Definition 10. It contains all dominated but not strictly dominated points of $\mathbb{R}^m$ with respect to the Pareto front. This set corresponds to the usual representation of the Pareto front, as in Figure III.4.

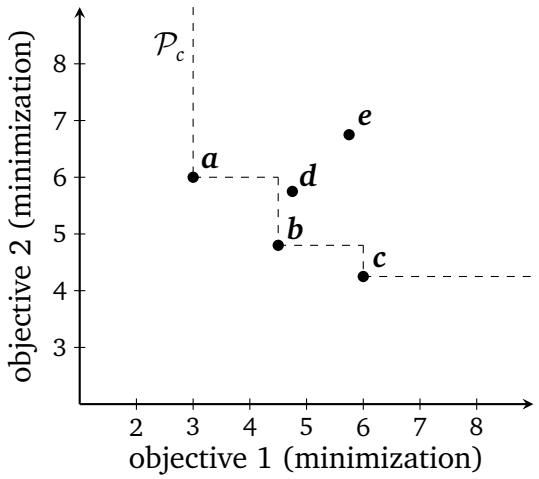



Figure III.4 Illustration of $\mathcal{P}_c$ on a simple case with five elements in the objective space.

**Definition 10.** With $\mathcal{A} \subset \mathbb{R}^m$, let us define $\mathcal{P}_c$ as follows

$$\mathcal{P}_c(\mathcal{A}) = \mathcal{P}(\mathcal{A}) \cup \left\{\boldsymbol{b} \in \mathbb{R}^m \mid \exists \boldsymbol{z} \in \mathcal{P}(\mathcal{A}),\ \boldsymbol{z} \succ_c \boldsymbol{b}\ \&\ \nexists \boldsymbol{z} \in \mathcal{P}(\mathcal{A}),\ \boldsymbol{z} \succ\!\!\succ_c \boldsymbol{b}\right\}. \tag{III.13}$$

Note that in Figure III.4, the set $\mathcal{A}$ only contains points $\boldsymbol{a}$ to $\boldsymbol{e}$.

## III–2.2 Bounding-Box approach

This section introduces the notations needed to describe the Bounding-Box approach, presented in Section III–1.3.

**Definition 11.** An intrinsic error is assumed on the computation of the objectives and constraints $\boldsymbol{\rho}$. The estimated value $\widetilde{\boldsymbol{\rho}}^l(\boldsymbol{x})$ differs from $\boldsymbol{\rho}(\boldsymbol{x})$ by an additive error that depends on the accuracy $l$, as follows:

$$\forall \boldsymbol{x} \in \mathcal{X},\ \boldsymbol{\rho}(\boldsymbol{x}) = \widetilde{\boldsymbol{\rho}}^l(\boldsymbol{x}) + \boldsymbol{\varepsilon}^l(\boldsymbol{x}). \tag{III.14}$$

**Remark** *Let us recall that $l$ implicitly refers to $l(\boldsymbol{x})$. The accuracy used for computing the objective functions can differ from one design $\boldsymbol{x}$ to another. The Bounding-Box approach aims to compute $\boldsymbol{\rho}$ with high accuracy (high values of $l$) only for efficient designs.*

Two assumptions are then necessary for demonstrating the convergence of the proposed approach. Namely, we assume that a conservative estimation of the true error is available and that infinite refinement leads to the exact estimate.

**Assumption 1.** We assume that a conservative estimation $\overline{\boldsymbol{\varepsilon}}^l$ of the true error $\boldsymbol{\varepsilon}^l$ can be computed, meaning that:

$$\forall \boldsymbol{x} \in \mathcal{X},\ |\boldsymbol{\varepsilon}^l(\boldsymbol{x})| \leq \overline{\boldsymbol{\varepsilon}}^l(\boldsymbol{x}). \tag{III.15}$$

In other words, $\forall \boldsymbol{x} \in \mathcal{X}, \boldsymbol{\rho}(\boldsymbol{x}) \in \mathcal{B}_{\widetilde{\boldsymbol{\rho}}^l}\big(\boldsymbol{x}, |\boldsymbol{\varepsilon}^l(\boldsymbol{x})|\big) \subseteq \mathcal{B}_{\widetilde{\boldsymbol{\rho}}^l}\big(\boldsymbol{x}, \overline{\boldsymbol{\varepsilon}}^l(\boldsymbol{x})\big)$.

**Assumption 2.** No constraint is imposed on the monotony of $\big(\overline{\boldsymbol{\varepsilon}}^l\big)_{l\in\mathbb{N}_+}$ but its convergence is assumed:

$$\lim_{l\to+\infty} \overline{\boldsymbol{\varepsilon}}^l = 0. \tag{III.16}$$

Note, however, that the computational cost increases with the accuracy.

In this work, the accuracy level $l$ is tuned adaptively for each design $\boldsymbol{x}$. Hence, we consider the accuracy $l$ at each $\boldsymbol{x}$ as an increasing sequence $\big(l_k\big)_{k\in\mathbb{N}}$ during the optimization process. Although Section III–1.3 proposes an iterative refinement based on the $POP_{min}$ metric, we propose to carry out the convergence analysis on a more general case, where a set of design $\mathcal{X}_k^*$ is getting refined at each iteration.

The output of interest is $\mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}$, the Pareto-optimal designs. The proposed Bounding-Box approach gives an approximation based on sequentially refined estimations $\widetilde{\boldsymbol{\rho}}^{l_k}$, where refinement is performed only on the most interesting designs. This assumption is formally defined in Definition 12. This formalism is given in the design space, but it can easily be transposed to the objective and constraint space.

**Definition 12.** For any design set $\{\boldsymbol{x}_i\}_{i=1}^N \in \mathcal{X}^N$, with $\widetilde{\boldsymbol{\rho}}^{l_k}$ the approximation of $\boldsymbol{\rho}$ and $\overline{\boldsymbol{\varepsilon}}^{l_k}$ the associated error estimation, the recursive Pareto-optimal sequences are defined as follows:

$$\begin{aligned}
&\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},0} = \{\boldsymbol{x}_i\}_{i=1}^N, \\
\text{Fixed:}\quad &\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k+1} = \mathcal{X}_{\mathcal{P}_{\mathcal{B}}}^{\widetilde{\boldsymbol{\rho}}^{l_k}}\Big(\Big\{\Big(\widetilde{\mathcal{X}}_{\mathcal{P}_i}^{\boldsymbol{\rho},0}, \overline{\boldsymbol{\varepsilon}}^{l_k}\Big(\widetilde{\mathcal{X}}_{\mathcal{P}_i}^{\boldsymbol{\rho},0}\Big)\Big)\Big\}_{i=1}^N\Big), \\
\text{Recursive:}\quad &\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k+1} = \mathcal{X}_{\mathcal{P}_{\mathcal{B}}}^{\widetilde{\boldsymbol{\rho}}^{l_k}}\Big(\Big\{\Big(\widetilde{\mathcal{X}}_{\mathcal{P}_i}^{\boldsymbol{\rho},k}, \overline{\boldsymbol{\varepsilon}}^{l_k}\Big(\widetilde{\mathcal{X}}_{\mathcal{P}_i}^{\boldsymbol{\rho},k}\Big)\Big)\Big\}_{i=1}^{N_k}\Big), \\
\text{where}\quad &l_k(\boldsymbol{x}) = \begin{cases} l_{k-1}(\boldsymbol{x})+1 & \text{if } \boldsymbol{x} \in \mathcal{X}_k^* \\ l_{k-1}(\boldsymbol{x}) & \text{else} \end{cases}, \\
&N_k = Card\big(\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}\big).
\end{aligned} \tag{III.17}$$

As raised earlier, $\mathcal{X}_k^*$ is the set of designs to refine. Following Sections II–3 and III–1.3, $\mathcal{X}_k^* = \Big\{\underset{\boldsymbol{x}\in\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}}{\text{argmax}}\ POP_{min}(\boldsymbol{x})\Big\}$ is a proper choice for such task when complemented with a threshold $\boldsymbol{s}_2$.

The *Fixed* strategy iteratively refines all elements from $\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},0} = \{\boldsymbol{x}_i\}_{i=1}^N$ whereas the *Recursive* strategy only refines elements from the current set $\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}$, which can be regarded as the approximation of $\mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}$ at the $k^{th}$ refinement iteration, using $\widetilde{\boldsymbol{\rho}}^{l_k}$ instead of $\boldsymbol{\rho}$. It represents the Pareto-optimal set associated to a given set of accuracies $\{l_k(\boldsymbol{x}_i)\}_i$.

Using the *Fixed* formulation with $\mathcal{X}_k^* = \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},0}$ corresponds to using a double-loop or nested optimization, where the objective and constraint functions are estimated with increasing accuracy on the whole design set $\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},0}$ with size $N$. As raised in Assumption 2, the most time-consuming step consists in decreasing the estimated error $\overline{\boldsymbol{\varepsilon}}^l$ through computations with higher accuracy.

However, using the *Recursive* Bounding-Box refinement strategy (III.17) with $\mathcal{X}_k^* = \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}$, Proposition 1 gives that $\forall k \in \mathbb{N}_+$, $\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k} \subseteq \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k-1} \subseteq \ldots \subseteq \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},0}$. Any unsatisfied non-domination condition implies a strict inclusion between two sets of the sequence. Hence, whenever a box gets dominated, the size $N_k$ of the Pareto-optimal set is decreased and further refinements are then performed only on $N_k \leq N$ designs.

**Remark** *In the following, $\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}$ will only refer to the recursive strategy, that is at the core of the Bounding-Box approach.*

A third strong assumption is required here to ensure the good behavior of the refinement sequence given by $\{\mathcal{X}_k^*\}_k$. It aims at avoiding to refine indefinitely only a subset of the current set $\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}$.

**Assumption 3.** We assume in the following that refinements are spread over the different designs, so that the error goes to zero for all designs that remain in $\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}$:

$$\lim_{k\to\infty} \max_{\boldsymbol{x}\in\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}} \overline{\boldsymbol{\varepsilon}}^{l_k}(\boldsymbol{x}) = \mathbf{0}. \tag{III.18}$$

**Remark** *This assumption shows that using* $\mathcal{X}_k^* = \underset{\boldsymbol{x}\in\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}}{\operatorname{argmax}}\, POP_{min}(\boldsymbol{x})$ *is not viable as it may refine the same design indefinitely. This is why SAMATA (Section II–3) and SABBa (Section III–1.3) require a threshold* $\boldsymbol{s}_2$*, so that refinements are only iterated on designs that still yield an estimated error above* $\boldsymbol{s}_2$*. This process can be carried out iteratively with decreasing values of* $\boldsymbol{s}_2$*, converging toward* 0*, to ensure that Assumption 3 is satisfied.*

## III–2.3 Convergence analysis

This section aims to demonstrate the convergence of the sequence $\left(\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}\right)_{k\in\mathbb{N}_+}$ toward the true Pareto-optimal set. More precisely, the convergence of the the continuous Pareto front $\mathcal{P}_c\left(\widetilde{\boldsymbol{\rho}}_f^{l_k}\left(\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}\right)\right)$ built on the approximated objectives toward the real continuous front $\mathcal{P}_c = \mathcal{P}_c\left(\boldsymbol{\rho}_f(\mathcal{X}_{\mathcal{P}})\right)$ in the objective space is demonstrated.

To this extent, another assumption is needed on the behavior of the optimization process.

**Assumption 4.** For a given multi-objective optimization, it is assumed that:

$$\forall \boldsymbol{y}\in\mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}, \exists \mathcal{D}\subseteq\mathcal{X} \text{ so that } \boldsymbol{y}\in\mathcal{D}, \mathring{\mathcal{D}}\neq\emptyset \text{ and } \forall j\in[\![1,m]\!],\ \rho_j\in\mathcal{C}^0(\mathcal{D}), \tag{III.19}$$

and that for any set $\mathcal{D}(\boldsymbol{y})$ satisfying the previous condition for some $\boldsymbol{y}\in\mathcal{X}$,

$$\forall \epsilon\in\mathbb{R}_+^*, \exists M\in\mathbb{N}_+, \forall \boldsymbol{y}\in\mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}, \exists k\in[\![1,M]\!],\ \boldsymbol{x}_k\in\mathcal{D}(\boldsymbol{y}), \left\|\boldsymbol{x}_k-\boldsymbol{y}\right\|\leq\epsilon. \tag{III.20}$$

In practice, (III.19) yields that there exists a non-empty part of $\mathcal{X}$ around each efficient design in which all $f_j$ are continuous. Equation (III.20) then states that the optimizer converges toward the whole Pareto front and covers the integrity of $\mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}$.

Assumption 4 simulates a "well-behaving" optimizer, that reaches the whole Pareto front thanks to the continuity assumption. We study the impact of the proposed strategy on the convergence of the optimization process in this context.

**Remark** *The following relation is verified for any design set* $\mathcal{Y}$*:*

$$\mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}\left(\mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}(\mathcal{Y})\right) = \mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}(\mathcal{Y}). \tag{III.21}$$

*More generally,* $\forall \mathcal{S}$ *s.t.* $\mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}(\mathcal{Y})\subseteq\mathcal{S}\subseteq\mathcal{Y}$*,*

$$\mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}(\mathcal{S}) = \mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}(\mathcal{Y}). \tag{III.22}$$

*This remark can be transposed to the objective space.*

**Proposition 2.** *In any set* $\mathcal{Y}$*, a design which is not Pareto-optimal is dominated by an element of the Pareto front:*

$$\forall \boldsymbol{y}\in\mathcal{Y},\ \ \boldsymbol{y}\notin\mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}(\mathcal{Y}) \iff \exists \boldsymbol{y}'\in\mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}(\mathcal{Y}),\ \boldsymbol{y}'\succ_c\boldsymbol{y}. \tag{III.23}$$

*Proof.* For $\boldsymbol{y}\notin\mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}(\mathcal{Y})$, let us assume that $\nexists \boldsymbol{y}'\in\mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}(\mathcal{Y}),\ \boldsymbol{y}'\succ_c\boldsymbol{y}$, then $\boldsymbol{y}\in\mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}\left(\mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}(\mathcal{Y})\cup\boldsymbol{y}\right) = \mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}(\mathcal{Y})$ from Equation (III.22), which proves the first implication by contradiction. The second implication is immediate from the definition of $\mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}$. □

**Lemma 1.** *Any Pareto-optimal design in the classical sense is also Pareto-optimal in the Boxed Pareto dominance sense. For any approximation $\widetilde{\boldsymbol{\rho}}$ with conservative error $\overline{\boldsymbol{\varepsilon}}$:*

$$\forall \left\{\left(\boldsymbol{x}_i, \overline{\boldsymbol{\varepsilon}}(\boldsymbol{x}_i)\right)\right\}_{i=1}^{N} \in \left(\mathcal{X} \times \mathbb{R}^m\right)^N,\ \mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}\left(\left\{\boldsymbol{x}_i\right\}_{i=1}^{N}\right) \subseteq \mathcal{X}_{\mathcal{P}_{\mathcal{B}}}^{\widetilde{\boldsymbol{\rho}}}\left(\left\{\left(\boldsymbol{x}_i, \overline{\boldsymbol{\varepsilon}}(\boldsymbol{x}_i)\right)\right\}_{i=1}^{N}\right). \tag{III.24}$$

*Proof.* By using the explicit definition of the Pareto front as in Equation (III.11), the proof is immediate as Assumption 1 gives $\forall \boldsymbol{x} \in \mathcal{X}, \forall j \in [\![1, m]\!],\ \rho_j(\boldsymbol{x}) \in \left[\widetilde{\rho}_j(\boldsymbol{x}) - \overline{\varepsilon}_j(\boldsymbol{x}),\ \widetilde{\rho}_j(\boldsymbol{x}) + \overline{\varepsilon}_j(\boldsymbol{x})\right]$. Hence, $\forall \boldsymbol{x}_i \in \mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}\left(\left\{\boldsymbol{x}_i\right\}_{i=1}^{N}\right)$,

$$\begin{aligned}
&\boldsymbol{\rho}(\boldsymbol{x}_i) \notin \mathcal{F} \text{ and } \boldsymbol{\rho}(\boldsymbol{x}_k) \in \mathcal{A} \Rightarrow \boldsymbol{\rho}_f(\boldsymbol{x}_k) \not\leq \boldsymbol{\rho}_f(\boldsymbol{x}_i),\\
&\widetilde{\boldsymbol{\rho}}(\boldsymbol{x}_i) - \overline{\boldsymbol{\varepsilon}}(\boldsymbol{x}_i) \notin \mathcal{F} \text{ and } \widetilde{\boldsymbol{\rho}}(\boldsymbol{x}_k) + \overline{\boldsymbol{\varepsilon}}(\boldsymbol{x}_k) \in \mathcal{A} \Rightarrow \widetilde{\boldsymbol{\rho}}_f(\boldsymbol{x}_k) + \overline{\boldsymbol{\varepsilon}}_f(\boldsymbol{x}_k) \not\leq \boldsymbol{\rho}_f(\boldsymbol{x}_i) - \overline{\boldsymbol{\varepsilon}}_f(\boldsymbol{x}_i),\\
&\mathcal{B}_{\widetilde{\boldsymbol{\rho}}}\left(\boldsymbol{x}_i, \overline{\boldsymbol{\varepsilon}}(\boldsymbol{x}_i)\right) \notin \mathcal{F}_{\mathcal{B}} \text{ and } \mathcal{B}_{\widetilde{\boldsymbol{\rho}}}\left(\boldsymbol{x}_k, \overline{\boldsymbol{\varepsilon}}(\boldsymbol{x}_k)\right) \in \mathcal{A}_{\mathcal{B}} \Rightarrow \widetilde{\boldsymbol{\rho}}_f(\boldsymbol{x}_k) + \overline{\boldsymbol{\varepsilon}}_f(\boldsymbol{x}_k) \not\leq \boldsymbol{\rho}_f(\boldsymbol{x}_i) - \overline{\boldsymbol{\varepsilon}}_f(\boldsymbol{x}_i),
\end{aligned}$$

Therefore, $\boldsymbol{x}_i \in \mathcal{X}_{\mathcal{P}_{\mathcal{B}}}^{\widetilde{\boldsymbol{\rho}}}\left(\left\{\left(\boldsymbol{x}_i, \overline{\boldsymbol{\varepsilon}}(\boldsymbol{x}_i)\right)\right\}_{i=1}^{N}\right)$, which ends the proof. □

Proposition 2 formalizes that there is at least one dominant design in a given set of points and Lemma 1 shows the robustness of the boxed dominance. Such robustness is also demonstrated for the recursive sequence of the Bounding-Box approach in Theorem 3.

**Theorem 3.** *For any initial set $\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},0}$, the true Pareto-optimal designs are retained in the recursive sequence* (III.17) *$\forall k \in \mathbb{N}_+$, i.e.:*

$$\mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}\left(\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},0}\right) \subseteq \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}. \tag{III.25}$$

*Proof.* For proving this, mathematical induction can be used.

Let us assume that $\exists k \in \mathbb{N}_+$, so that $\mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}\left(\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},0}\right) \subseteq \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}$. Then, because $\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k} \subseteq \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},0}$ is always verified, Equation (III.22) yields that:

$$\mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}\left(\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},0}\right) = \mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}\left(\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}\right). \tag{III.26}$$

Finally, Lemma 1 gives $\mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}\left(\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}\right) \subseteq \mathcal{X}_{\mathcal{P}_{\mathcal{B}}}^{\widetilde{\boldsymbol{\rho}}^{l_k}}\left(\left\{\left(\widetilde{\mathcal{X}}_{\mathcal{P}_i}^{\boldsymbol{f},k}, \overline{\boldsymbol{\varepsilon}}^{l_k}\left(\widetilde{\mathcal{X}}_{\mathcal{P}_i}^{\boldsymbol{f},k}\right)\right)\right\}_{i=1}^{N_k}\right) = \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k+1}$, which ends the inductive step of the proof, yielding $\mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}\left(\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},0}\right) \subseteq \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k+1}$.

Of course, $\mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}\left(\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},0}\right) \subseteq \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},0} = \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},0}$, therefore, the mathematical induction proves the inclusion between the Pareto-optimal sets. □

This first theorem extends the robust behavior presented in Lemma 1 to the recursive Bounding-Box strategy introduced in Definition 12. We raised earlier that this strategy allows reducing the number of refinements. Theorem 3 also ensures that all Pareto-optimal designs are retained for high-accuracy estimation.

Using Assumption 4, the convergence of the continuous front $\mathcal{P}_c$ can be proven in the objective space. Hence, in practice, we study the convergence of $\widetilde{\mathcal{P}}_{cf}^{\,k} = \mathcal{P}_c\left(\widetilde{\boldsymbol{\rho}}_f^{l_k}\left(\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}\right)\right)$ toward $\mathcal{P}_{cf} = \mathcal{P}_c\left(\boldsymbol{\rho}_f\left(\mathcal{X}_{\mathcal{P}}^{\boldsymbol{\rho}}\right)\right)$.

**Theorem 4.** *For any initial set $\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},0} \in \mathcal{X}^N$, approximation sequence $\left(\widetilde{\boldsymbol{\rho}}^{l_k}\right)_{k\in\mathbb{N}_+}$ and conservative error set sequence $\left(\overline{\boldsymbol{\varepsilon}}^{l_k}\right)_{k\in\mathbb{N}_+}$, the following convergence is verified:*

$$\lim_{(N,k)\to(+\infty,+\infty)} d_H\left(\widetilde{\mathcal{P}}_{cf}^{\,k}, \mathcal{P}_{cf}\right) = 0, \tag{III.27}$$
$$\text{where } \widetilde{\mathcal{P}}_{cf}^{\,k} = \mathcal{P}_c\left(\widetilde{\boldsymbol{\rho}}_{\boldsymbol{f}}^{l_k}\left(\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}\right)\right),$$
$$\text{and } \mathcal{P}_{cf} = \mathcal{P}_c\left(\boldsymbol{\rho}_f\left(\mathcal{X}_{\mathcal{P}}\right)\right),$$

*with the classical Hausdorff distance denoted by $d_H$.*

*For demonstrating this convergence, the constraints must take the form $\boldsymbol{\rho}_{\boldsymbol{g}}(\boldsymbol{x}) \leq \mathbf{0}$, so that the failure set $\mathcal{F}$ is an open set.*

*Proof.* We first denote $\mathcal{P}_{cf}^{\,k} = \mathcal{P}_c\left(\boldsymbol{\rho}_f\left(\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}\right)\right)$.

The triangle inequality then gives :

$$d_H\left(\widetilde{\mathcal{P}}_{cf}^{\,k}, \mathcal{P}_{cf}\right) \leq d_H\left(\widetilde{\mathcal{P}}_{cf}^{\,k}, \mathcal{P}_{cf}^{\,k}\right) + d_H\left(\mathcal{P}_{cf}^{\,k}, \mathcal{P}_{cf}\right). \tag{III.28}$$

Let us first focus on the first term of the right-hand side of (III.28).

Assumption 1 implies that

$$\forall \boldsymbol{x} \in \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k},\ d_\infty\left(\boldsymbol{\rho}(\boldsymbol{x}), \widetilde{\boldsymbol{\rho}}^{l_k}(\boldsymbol{x})\right) \leq \max_j \overline{\boldsymbol{\varepsilon}}_j^{l_k}(\boldsymbol{x}), \tag{III.29}$$

where $d_\infty$ denotes the distance associated to the $l_\infty$ norm, $d_\infty(\cdot,\cdot) = \|\cdot - \cdot\|_\infty$.

Now, let us suppose that $\exists \boldsymbol{a} \in \mathcal{P}_{cf}^{\,k}$, so that $d_\infty\left(\boldsymbol{a}, \mathcal{P}_{cf}^{\,k}\right) > \max_{(i,j)} \overline{\boldsymbol{\varepsilon}}_j^{l_k}\left(\widetilde{\mathcal{X}}_{\mathcal{P}_i}^{\boldsymbol{\rho},k}\right)$. Recall that the distance between a point and a set is the minimal distance between this point and each element of the set.

This means that $\widetilde{\mathcal{P}}_{cf}^{\,k} \cap \mathcal{B}\left(\boldsymbol{a}, \overline{\boldsymbol{\varepsilon}}_{max}\right) = \emptyset$ with $\overline{\boldsymbol{\varepsilon}}_{max}$ being the $m$-dimensional vector where each component is equal to $\max_{(i,j)} \overline{\boldsymbol{\varepsilon}}_j^{l_k}\left(\widetilde{\mathcal{X}}_{\mathcal{P}_i}^{\boldsymbol{\rho},k}\right)$.

Therefore, from Def. 10, either (i) $\exists \boldsymbol{x} \in \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}, \mathcal{B}\left(\widetilde{\boldsymbol{\rho}}_{\boldsymbol{f}}^{l_k}(\boldsymbol{x}), \mathbf{0}\right) \underset{0}{\succcurlyeq} \mathcal{B}\left(\boldsymbol{a}, \overline{\boldsymbol{\varepsilon}}_{max}\right)$ or (ii) $\forall \boldsymbol{a}' \in \mathcal{B}\left(\boldsymbol{a}, \overline{\boldsymbol{\varepsilon}}_{max}\right), \nexists \boldsymbol{x} \in \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}$, so that $\widetilde{\boldsymbol{\rho}}_{\boldsymbol{f}}^{l_k}(\boldsymbol{x}) \succ \boldsymbol{a}'$.

- In the first case (i), using the remark below Definition 7, the widths can be swaped, $\exists \boldsymbol{x} \in \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}, \mathcal{B}\left(\widetilde{\boldsymbol{\rho}}_{\boldsymbol{f}}^{l_k}(\boldsymbol{x}), \overline{\boldsymbol{\varepsilon}}_{max}\right) \underset{0}{\succcurlyeq} \mathcal{B}\left(\boldsymbol{a}, \mathbf{0}\right)$ and as $\boldsymbol{a} \in \mathcal{P}_{cf}^{\,k}, \nexists \boldsymbol{x}' \in \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}, \boldsymbol{\rho}(\boldsymbol{x}') \succ \succ \boldsymbol{a}$. Hence, $\exists \boldsymbol{x} \in \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}, \boldsymbol{\rho}(\boldsymbol{x}) \notin \mathcal{B}\left(\widetilde{\boldsymbol{\rho}}^{l_k}(\boldsymbol{x}), \overline{\boldsymbol{\varepsilon}}_{max}\right)$. This would mean $\exists \boldsymbol{x} \in \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}$, $d_\infty\left(\boldsymbol{\rho}(\boldsymbol{x}), \widetilde{\boldsymbol{\rho}}^{l_k}(\boldsymbol{x})\right) > \overline{\boldsymbol{\varepsilon}}_{max_i} \geq \max_j \overline{\boldsymbol{\varepsilon}}_j^{l_k}(\boldsymbol{x})$, which is contradictory with (III.29).

- The second case (ii) implies that $\exists \boldsymbol{b} \in \widetilde{\mathcal{P}}_{cf}^{\,k}, \mathcal{B}\left(\boldsymbol{a}, \overline{\boldsymbol{\varepsilon}}_{max}\right) \underset{0}{\succcurlyeq} \mathcal{B}\left(\boldsymbol{b}, \mathbf{0}\right)$. However, $\exists \boldsymbol{x} \in \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}$ so that $\boldsymbol{\rho}_f(\boldsymbol{x}) \succeq \boldsymbol{a}$, and thus $\mathcal{B}\left(\boldsymbol{\rho}_f(\boldsymbol{x}), \overline{\boldsymbol{\varepsilon}}_{max}\right) \underset{0}{\succcurlyeq} \mathcal{B}\left(\boldsymbol{b}, \mathbf{0}\right)$. As $\boldsymbol{b} \in \widetilde{\mathcal{P}}_{cf}^{\,k}$, from Def. 10, $\nexists \boldsymbol{x}' \in \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}$, so that $\widetilde{\boldsymbol{\rho}}_{\boldsymbol{f}}^{l_k}(\boldsymbol{x}') \succcurlyeq \boldsymbol{b}$. In summary, $\exists \boldsymbol{x} \in \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}, \widetilde{\boldsymbol{\rho}}_{\boldsymbol{f}}^{l_k}(\boldsymbol{x}) \notin \mathcal{B}\left(\boldsymbol{\rho}_f(\boldsymbol{x}), \overline{\boldsymbol{\varepsilon}}_{max}\right)$, which contradicts again (III.29).

Hence, by contradiction, $\forall \boldsymbol{a} \in \mathcal{P}_{cf}^{k}, d_\infty\left(\boldsymbol{a}, \widetilde{\mathcal{P}}_{cf}^{k}\right) \leq \max_{(i,j)} \overline{\boldsymbol{\varepsilon}}_j^{l_k}\left(\widetilde{\mathcal{X}}_{\mathcal{P}_i}^{\boldsymbol{\rho},k}\right)$. This statement holds also when inverting the Pareto front continuous sets (real and approximated), and can be proved in the same way. As a consequence, the Hausdorff distance can be written as follows:

$$d_H\left(\widetilde{\mathcal{P}}_{cf}^{k}, \mathcal{P}_{cf}^{k}\right) \leq \max_{(i,j)} \overline{\boldsymbol{\varepsilon}}_j^{l_k}\left(\widetilde{\mathcal{X}}_{\mathcal{P}_i}^{\boldsymbol{\rho},k}\right).$$

Hence, Assumption 3 gives the first half of Equation (III.28):

$$\lim_{k\to\infty} d_H\left(\widetilde{\mathcal{P}}_{cf}^{k}, \mathcal{P}_{cf}^{k}\right) = 0. \tag{III.30}$$

Let us focus now on the second part of the sum.

For all $\boldsymbol{a} \in \mathcal{P}_{cf}^{k}, \exists \boldsymbol{y} \in \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}, \boldsymbol{\rho}_f(\boldsymbol{y}) \succeq \boldsymbol{a}$.

If $\boldsymbol{\rho}(\boldsymbol{y}) \in \mathcal{F}$, as $\mathcal{F}$ is an open set, $\exists \epsilon, \forall \boldsymbol{y}' \in \mathcal{B}\left(\boldsymbol{\rho}(\boldsymbol{y}), \boldsymbol{\epsilon}\right), \boldsymbol{\rho}(\boldsymbol{y}') \in \mathcal{F}$, meaning $\mathcal{B}\left(\boldsymbol{\rho}(\boldsymbol{y}), \boldsymbol{\epsilon}\right) \in \mathcal{F}_{\mathcal{B}}$. Assumption 3 yields $\exists K, \forall k^* \geq K, 2\overline{\boldsymbol{\varepsilon}}^{l_{k^*}} \leq \boldsymbol{\epsilon}$, thus $\mathcal{B}\left(\widetilde{\boldsymbol{\rho}}^{l_{k^*}}(\boldsymbol{y}), \overline{\boldsymbol{\varepsilon}}^{l_{k^*}}(\boldsymbol{y})\right) \in \mathcal{F}_{\mathcal{B}}$. Hence, $\boldsymbol{y} \notin \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k^*}$. This gives that $\exists K, \forall k^* \geq K, \forall \boldsymbol{y} \in \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k^*}, \boldsymbol{\rho}(\boldsymbol{y}) \in \mathcal{A}$.

Thus, $\exists \boldsymbol{a}' \in \mathcal{P}_{cf}$, so that $\boldsymbol{a}' \succeq \boldsymbol{a}$. Moreover, $\forall \boldsymbol{b}$ such that $\exists \boldsymbol{a}' \in \mathcal{P}_{cf}$, $\boldsymbol{a}' \succcurlyeq \boldsymbol{b}$, Assumption 4 with Theorem 3 provides evidence that the recursive discrete efficient set converges toward the continuous real one and that this efficient set is included in $\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}$. In other words, $\forall k^* \geq K, \exists M \in \mathbb{N}^*, \exists \boldsymbol{x} \in \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k^*}$ so that $\left|\boldsymbol{a}' - \boldsymbol{\rho}_f(\boldsymbol{x})\right| \leq \left|\boldsymbol{a}' - \boldsymbol{b}\right|$. Thus, $\forall k^* \geq K, \exists M \in \mathbb{N}^*, \exists \boldsymbol{x} \in \widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k^*}, \boldsymbol{\rho}_f(\boldsymbol{x}) \succcurlyeq \boldsymbol{b}$.

Hence, when $k$ is high enough, $\mathcal{P}_{cf}^{k}$ is always dominated by $\mathcal{P}_{cf}$ and any element dominated by $\mathcal{P}_{cf}$ is also dominated by $\mathcal{P}_{cf}^{k}$ when $N$ grows. From Definition 10, it can be deduced that:

$$\lim_{(N,k)\to(+\infty,+\infty)} d_H\left(\mathcal{P}_{cf}^{k}, \mathcal{P}_{cf}\right) = 0. \tag{III.31}$$

Finally, by combining Equations (III.28), (III.30) and (III.31), it comes:

$$\lim_{(N,k)\to(+\infty,+\infty)} d_H\left(\widetilde{\mathcal{P}}_{cf}^{k}, \mathcal{P}_{cf}\right) = 0,$$

which ends the proof. □

The continuity assumption gives preimage convergence.

In summary, under some assumptions about conservative errors and overall good behaviors of the optimization process, the convergence of the approximated Pareto front toward the true $\mathcal{P}_c$ in the objective space is demonstrated. A representation of the Pareto-optimal set is drawn in Figure III.4.

## III–2.4 Surrogate-Assisting model

In this section, we study the convergence properties of the method when the Bounding-Box approach is coupled with the Surrogate-Assisting strategy.

The SA model predictive value is written $\boldsymbol{\rho}_{SA}(\boldsymbol{x})$ and the estimation of the error is denoted $\overline{\boldsymbol{\varepsilon}}_{SA}(\boldsymbol{x})$, which we assume as conservative, *i.e.* $\overline{\boldsymbol{\varepsilon}}_{SA}(\boldsymbol{x}) \geq \boldsymbol{\varepsilon}_{SA}(\boldsymbol{x}) = \boldsymbol{\rho}(\boldsymbol{x}) - \boldsymbol{\rho}_{SA}(\boldsymbol{x})$. The value returned to the optimizer is the SA model predictive value when accuracy $\boldsymbol{s}_1$ is reached, and the Bounding-Box estimation converged to $\boldsymbol{s}_2$ otherwise.

**Definition 13.** Using the SABBa notations from the previous section, the approximated value $\boldsymbol{\rho}_{SABBa}$ and associated conservative error $\overline{\boldsymbol{\varepsilon}}_{SABBa}$ are defined as follows:

$$\left(\boldsymbol{\rho}^{\boldsymbol{s}_1,k}_{SABBa}(\boldsymbol{x}_i), \overline{\boldsymbol{\varepsilon}}^{\boldsymbol{s}_1,k}_{SABBa}(\boldsymbol{x}_i)\right) = \begin{cases} \left(\boldsymbol{\rho}_{SA}(\boldsymbol{x}_i), \overline{\boldsymbol{\varepsilon}}_{SA}(\boldsymbol{x}_i)\right) & \text{if } \overline{\boldsymbol{\varepsilon}}_{SA}(\boldsymbol{x}_i) \leq \boldsymbol{s}_1 \\ \left(\widetilde{\boldsymbol{\rho}}^{l_k}(\boldsymbol{x}_i), \overline{\boldsymbol{\varepsilon}}^{l_k}(\boldsymbol{x}_i)\right) & \text{else} \end{cases}, \tag{III.32}$$

and the value returned to the optimizer is:

$$\left(\boldsymbol{\rho}^{\boldsymbol{s}_1,k(\boldsymbol{s}_2)}_{SABBa}(\boldsymbol{x}_i), \overline{\boldsymbol{\varepsilon}}^{\boldsymbol{s}_1,k(\boldsymbol{s}_2)}_{SABBa}(\boldsymbol{x}_i)\right),$$

where $k(\boldsymbol{s}_2)$ is the lower value of $k$ so that $\forall \boldsymbol{x} \in \widetilde{\mathcal{X}}^{\boldsymbol{\rho},k}_{\mathcal{P}}, \overline{\boldsymbol{\varepsilon}}^{l_k}(\boldsymbol{x}) \leq \boldsymbol{s}_2$.

The Pareto-optimal designs $\mathcal{X}^{\boldsymbol{\rho}}_{\mathcal{P}}$ are then approximated with a new recursive sequence, which takes into account SA-based approximations. To this extent, we give a few additional notations.

**Definition 14.** Given a set $\mathcal{X}_{new}$ of designs to study, let us define:

$$\begin{aligned} \mathcal{X}_{SA} &= \left\{\boldsymbol{x} \in \mathcal{X}_{new} \mid \overline{\boldsymbol{\varepsilon}}_{SA}(\boldsymbol{x}) \leq \boldsymbol{s}_1\right\} \\ \mathcal{X}_c &= \mathcal{X}_{new} \setminus \mathcal{X}_{SA} \end{aligned}$$

Namely, we denote with $\mathcal{X}_c$ the designs that do not meet the SA requirement $\overline{\boldsymbol{\varepsilon}}_{SA}(\boldsymbol{x}) \leq \boldsymbol{s}_1$ and $\mathcal{X}_{SA}$ the other ones, that can be approximated by the SA model.

**Definition 15.** For a given set $\mathcal{X}_{new}$, the SABBa recursive Pareto-optimal sequence is defined as follows:

$$\begin{aligned} &\widetilde{\mathcal{X}}^{\boldsymbol{\rho},0}_{\mathcal{P},\boldsymbol{s}_1} = \mathcal{X}_{new}, \\ &\widetilde{\mathcal{X}}^{\boldsymbol{\rho},k+1}_{\mathcal{P},\boldsymbol{s}_1} = \mathcal{X}^{\boldsymbol{\rho}^{\boldsymbol{s}_1,k}_{SABBa}}_{\mathcal{P}_{\mathcal{B}}}\left(\left\{\left(\widetilde{\mathcal{X}}^{\boldsymbol{\rho},k}_{\mathcal{P},\boldsymbol{s}_1 i}, \overline{\boldsymbol{\varepsilon}}^{\boldsymbol{s}_1,k}_{SABBa}\left(\widetilde{\mathcal{X}}^{\boldsymbol{\rho},k}_{\mathcal{P},\boldsymbol{s}_1 i}\right)\right)\right\}^{N_k}_{i=1}\right), \end{aligned} \tag{III.33}$$

The refinement sequence $l_k$ for designs in $\mathcal{X}_c$ is similar to Definition 12.

**Theorem 5.** *Given two user-defined thresholds $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$, the true discrete Pareto optima are included in the SABBa Pareto optima:*

$$\forall \left\{\boldsymbol{x}_i\right\}^N_{i=1} \in \mathcal{X}^N,\ \mathcal{X}^{\boldsymbol{\rho}}_{\mathcal{P}}\left(\left\{\boldsymbol{x}_i\right\}^N_{i=1}\right) \subseteq \mathcal{X}^{\boldsymbol{\rho}^{\boldsymbol{s}_1,k(\boldsymbol{s}_2)}_{SABBa}}_{\mathcal{P}_{\mathcal{B}}}\left(\left\{\left(\boldsymbol{x}_i, \overline{\boldsymbol{\varepsilon}}^{\boldsymbol{s}_1,k(\boldsymbol{s}_2)}_{SABBa}(\boldsymbol{x}_i)\right)\right\}^N_{i=1}\right), \tag{III.34}$$

*and equality holds when $\boldsymbol{s}_1 = \boldsymbol{s}_2 = \boldsymbol{0}$.*

*Proof.* The proof is straightforward and comes from the following inequalities:
If $\overline{\boldsymbol{\varepsilon}}_{SA}(\boldsymbol{x}_i) \not\leq \boldsymbol{s}_1$,

$$\left|\boldsymbol{\rho}_{SABBa}(\boldsymbol{x}_i) - \boldsymbol{\rho}(\boldsymbol{x}_i)\right| = \left|\widetilde{\boldsymbol{\rho}}^{l_{k(\boldsymbol{s}_2)}}(\boldsymbol{x}_i) - \boldsymbol{\rho}(\boldsymbol{x}_i)\right| \leq \overline{\boldsymbol{\varepsilon}}^{l_{k(\boldsymbol{s}_2)}}(\boldsymbol{x}_i).$$

Else,

$$\left|\boldsymbol{\rho}_{SABBa}(\boldsymbol{x}_i) - \boldsymbol{\rho}(\boldsymbol{x}_i)\right| = \left|\boldsymbol{\rho}_{SA}(\boldsymbol{x}_i) - \boldsymbol{\rho}(\boldsymbol{x}_i)\right| \leq \overline{\boldsymbol{\varepsilon}}_{SA}(\boldsymbol{x}_i),$$

from Definition 13. □

**Remark** *Note that the Surrogate-Assisting model is constructed on all previously computed boxes. That is to say, previous estimations $\widetilde{\boldsymbol{\rho}}^{l_{k(s_2)}}$ must have been computed to serve as training data for the SA model prediction. Hence, sequential optimization approaches will benefit the most from the Surrogate-Assisting strategy by permitting update of the SA model at each iteration.*

This Surrogate-Assisting strategy complements the Bounding-Box approach efficiently. If a pattern, linearity or simple behavior of the objective functions is detected, it can drastically reduce the overall optimization cost.

**Remark** *The convergence of the coupled approach can then be deduced from Theorems 4 and 5, by replacing $k \to +\infty$ with $(\boldsymbol{s}_1, \boldsymbol{s}_2) \to (\mathbf{0}, \mathbf{0})$. Intuitively, with a small $\boldsymbol{s}_1$, SA model will be built only if the prediction is very accurate. Choosing a small $\boldsymbol{s}_2$ allows a refinement $k$ that grows toward infinity.*

## III–2.5 Algorithm

Using all notations introduced in this section, we give here a detailed algorithm for applying SABBa in practice.

As depicted in Algorithm III.1, SABBa first checks at each iteration if some designs can be approximated with the SA model prediction. Refinements are then sequentially performed on the set $\mathcal{X}_k^*$ until all Pareto-optimal designs that were not estimated by the SA model have their estimated error below $\boldsymbol{s}_2$. Finally, all Pareto-optimal designs are gathered in $\mathcal{X}_{PO}$, and are compared to the new designs $\mathcal{X}_{new}$ during the next optimization iteration. Before this next iteration, the SA model is updated with all new estimated values $\widetilde{\boldsymbol{\rho}}^{l_{k(s_2)}}(\boldsymbol{x})$ and $\overline{\boldsymbol{\varepsilon}}^{l_{k(s_2)}}(\boldsymbol{x})$.

Following the SAMATA algorithm, Section II–3, one can loop on finer and finer values of the thresholds $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ to ensure proper convergence of the method. Here, the algorithm is presented with these parameters fixed at a given value.

**Remark** *To simplify the choice of the $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ thresholds, we define the normalized thresholds $\bar{\boldsymbol{s}}_1$ and $\bar{\boldsymbol{s}}_2$ as a percentage of the output ranges. Formally, for each output dimension $i$:*

$$\begin{aligned} s_{1_i} &= \bar{s}_{1_i} \times h_i, \\ s_{2_i} &= \bar{s}_{2_i} \times h_i, \\ \text{where } \; h_i &= \max_{\boldsymbol{x}}(\rho_i(\boldsymbol{x})) - \min_{\boldsymbol{x}}(\rho_i(\boldsymbol{x})). \end{aligned} \tag{III.35}$$

*As a reminder, $\bar{\boldsymbol{s}}_1$ and $\bar{\boldsymbol{s}}_2$ will be written in percentage.*
*Note that in practice, $h_i$ will be approximated with $\widetilde{\boldsymbol{\rho}}$ values.*

**Algorithm III.1** SABBa general algorithm

Set $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$
Initialize $\mathcal{X}_{PO}$ empty
**while** Optimization running **do**
  Get new designs $\mathcal{X}_{new}$
  Get $\mathcal{X}_c$ and $\mathcal{X}_{SA}$ from Definition 14
  Return $\boldsymbol{\rho}_{SA}(\boldsymbol{x})$ to the optimizer for each $\boldsymbol{x}$ in $\mathcal{X}_{SA}$
  Initialize $\widetilde{\boldsymbol{\rho}}^{l_0}(\boldsymbol{x})$ and $\overline{\boldsymbol{\varepsilon}}^{l_0}(\boldsymbol{x})$ for each $\boldsymbol{x}$ in $\mathcal{X}_c$
  Initialize $\widetilde{\mathcal{X}}^{\boldsymbol{\rho},0}_{\mathcal{P},\boldsymbol{s}_1} = \mathcal{X}_{new} \cup \mathcal{X}_{PO}$
  $k = 0$
  **while** $\exists \boldsymbol{x} \in \widetilde{\mathcal{X}}^{\boldsymbol{\rho},k}_{\mathcal{P},\boldsymbol{s}_1} \cap \mathcal{X}_c, \overline{\boldsymbol{\varepsilon}}^{l_k}(\boldsymbol{x}) \not\leq \boldsymbol{s}_2$ **do**
    Determine $\mathcal{X}^*_k$ to refine in $\widetilde{\mathcal{X}}^{\boldsymbol{\rho},k}_{\mathcal{P},\boldsymbol{s}_1} \cap \mathcal{X}_c$
    Compute new estimations $\widetilde{\boldsymbol{\rho}}^{l_{k+1}}(\boldsymbol{x})$ and $\overline{\boldsymbol{\varepsilon}}^{l_{k+1}}(\boldsymbol{x})$ for each $\boldsymbol{x}$ in $\mathcal{X}^*_k$
    Set $\widetilde{\boldsymbol{\rho}}^{l_{k+1}}(\boldsymbol{x}) = \widetilde{\boldsymbol{\rho}}^{l_k}(\boldsymbol{x})$ and $\overline{\boldsymbol{\varepsilon}}^{l_{k+1}}(\boldsymbol{x}) = \overline{\boldsymbol{\varepsilon}}^{l_k}(\boldsymbol{x})$ for all $\boldsymbol{x} \in \mathcal{X}_c \setminus \mathcal{X}^*_k$
    Compute new iterate $\widetilde{\mathcal{X}}^{\boldsymbol{\rho},k+1}_{\mathcal{P},\boldsymbol{s}_1}$ from Eq. (III.33)
    $k = k + 1$
  **end while**
  $k(\boldsymbol{s}_2) = k$
  Set $\mathcal{X}_{PO} = \widetilde{\mathcal{X}}^{\boldsymbol{\rho},k(\boldsymbol{s}_2)}_{\mathcal{P},\boldsymbol{s}_1}$
  Return $\widetilde{\boldsymbol{\rho}}^{l_{k(\boldsymbol{s}_2)}}(\boldsymbol{x})$ to the optimizer for each $\boldsymbol{x}$ in $\mathcal{X}_c$
  Update $\boldsymbol{\rho}_{SA}$ and $\overline{\boldsymbol{\varepsilon}}_{SA}$
**end while**
Return $\mathcal{X}_{PO}$

**Discussion** *This section provides a thorough theoretical presentation of SABBa, with proper definitions and clear assumptions that allow demonstrating convergence toward the Pareto-optimal designs. The main conclusions are as follows:*

- *SABBa is a particular case of SAMATA, that deals with uncorrelated uniform distributions as measure approximations. Notation-wise, the error distribution is here denoted by a box* $\mathcal{B}_{\widetilde{\boldsymbol{\rho}}}\big(\boldsymbol{x}, \overline{\boldsymbol{\varepsilon}}(\boldsymbol{x})\big)$.
- *In this context, Theorem 3 shows that iterating the Pareto-optimal set sequence (III.17) retains the true Pareto optima from the initial set.*
- *Using this theorem and several assumptions, the overall convergence of the Bounding-Box approach is demonstrated, by showing that the classical Pareto front representation of the Pareto optimal sequence, in the objective space, converges in the Hausdorff distance sense toward the true Pareto front of the problem when the number of designs and the refinement tend toward infinity.*
- *Finally, the coupling of this Bounding-Box approach with a Surrogate-Assisting strategy is presented and properly defined. Intuition is given on how to obtain overall convergence of the Surrogate-Assisting Bounding-Box approach.*

## III–3 Noisy optimization with tunable accuracy

The SABBa approach can be naturally applied in the more general context of noisy optimization with tunable accuracy. This section is devoted to the illustration of this application.

Notably, we assess the performance of SABBa on several analytical noisy optimization test-cases. Specifically, we aim to study the asymptotic behavior of SABBa compared to Fusi's Bounding-Box approach [Fusi and Congedo, 2016] and the Double-Loop approach. To this extent, specific formulations for the optimization problem with noisy objective functions with tunable accuracy are given in Section III–3.1.

Section III–3.2.1 tackles a classical bi-objective optimization problem with a comparison between the different strategies. The example provided in Section III–3.2.2 then highlights the increase of the computational cost raised in [Fusi and Congedo, 2016] when the optimizer converges toward the Pareto front. This behavior notably permits to emphasize the gains provided by the Surrogate-Assisting (SA) strategy. Finally, the robustness of SABBa to complex Pareto front is illustrated in a third test-case, presented in Section III–3.2.3.

### III–3.1 Numerical ingredients

In this section, we define the required elements (optimizer, surrogate model and computations with tunable accuracy) for applying SABBa. Note that tunable noises are artificially simulated to compare the different strategies asymptotically.

**Computations with tunable accuracy** A noise is arbitrarily added on the real value of the analytical test-cases, inversely proportional to the desired accuracy. In practice, the low-accuracy estimation is assumed to cost one evaluation and each refinement increases this value by one. The first low-accuracy computation is performed as follows for each objective and constraint function $\rho_i$:

$$\begin{aligned}
\widetilde{\rho}_i^0 &= \rho_i + \varepsilon_{bias_i}, \\
\overline{\varepsilon}_i^0 &= 2.\varepsilon_{noise_i}, \\
\varepsilon_{bias_i} &= \varepsilon_{noise_i} X_2 \quad \text{with } X_2 \sim \mathcal{U}[-1,1], \\
\varepsilon_{noise_i} &= \frac{\max(\rho_i) - \min(\rho_i)}{4 + 6X_1} \quad \text{with } X_1 \sim \mathcal{U}[0,1].
\end{aligned}$$

This estimation can be refined with a given refinement factor $r_f > 1$:

$$\begin{aligned}
\widetilde{\rho}_i^l &= \rho_i + \frac{\widetilde{\rho}_i^{l-1} - \rho_i}{r_f}, \\
\overline{\varepsilon}_i^l &= \frac{\overline{\varepsilon}_i^{l-1}}{r_f}.
\end{aligned}$$

**Optimizer** The optimization is performed with a sequence of Monte-Carlo samplings. At each iteration, the set of new designs is constituted by ten random designs, that can be arbitrarily forced to converge toward the optimal area. The effect of this convergence is studied in test-cases 2 and 3.

**Surrogate model** Similarly to the computations with tunable accuracy, the Surrogate-Assisting (SA) model $\boldsymbol{\rho}_{SA}$ is here an arbitrarily noised evaluation of the objective functions. We consider this noise to be proportional to the distance $d_{\min}$ to the closest training point. Practically, it is computed as a percentage of the range covered by the objective functions. For each dimension $i$:

$$\begin{aligned}
\rho_{SA_i} &= \rho_i + \varepsilon_{bias_i}, \\
\varepsilon_{bias_i} &= \varepsilon_{noise_i} X_2 \quad \text{with } X_2 \sim \mathcal{U}[-1,1], \\
\varepsilon_{noise_i} &= \frac{\overline{\epsilon}_{SA_i}}{2} X_1 \quad \text{with } X_1 \sim \mathcal{U}[0,1], \\
\overline{\epsilon}_{SA_i} &= \frac{d_{\min}}{\frac{1}{N_x}\sum_j \big(\max(x_j) - \min(x_j)\big)} . \frac{\max(\rho_i) - \min(\rho_i)}{2}.
\end{aligned}$$

**Refinements** Finally, the set of designs to refine at each iteration, denoted $\mathcal{X}_k^*$ in Definition 12 and Algorithm III.1, corresponds here to the whole current set of non-dominated designs $\widetilde{\mathcal{X}}_{\mathcal{P}}^{\boldsymbol{\rho},k}$. Such strategy notably corresponds to the Bounding-Box approach followed by Fusi in [Fusi and Congedo, 2016] and should allow for proper computational cost comparison.

In the following, SABBa is compared to three different approaches: i) the nested classical approach (denoted by *Class*), *i.e.* the non-recursive Bounding-Box approach defined in Def. 12, where each box is refined up to the threshold $\boldsymbol{s}_2$; ii) the recursive Bounding-Box approach, proposed by Fusi in [Fusi and Congedo, 2016], where only non-dominated boxes are refined up to $\boldsymbol{s}_2$; iii) a surrogate-assisted nested approach (denoted by *Class-acc*).

## III–3.2 Applications

### III–3.2.1 Test-case 1: The BNH problem

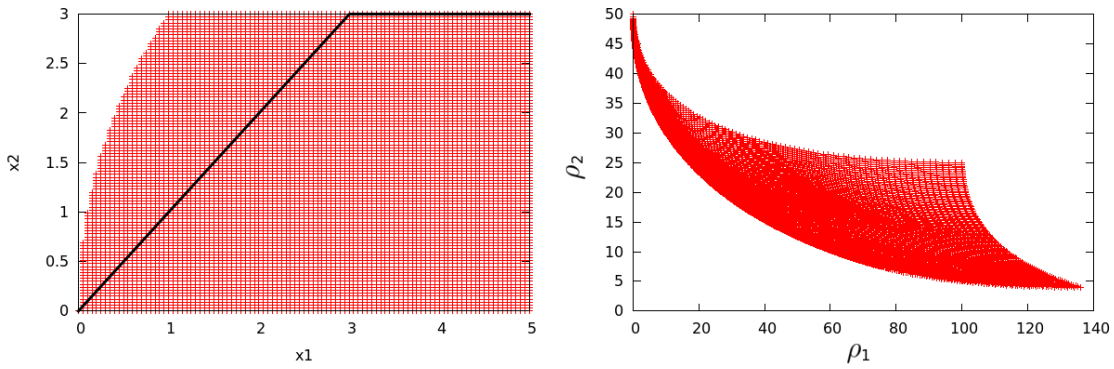
The first test-case deals with a bi-objective constrained problem, proposed by Binh and Korn (1997):

$$\begin{aligned}
\text{minimize:} \quad & \boldsymbol{\rho}(\boldsymbol{x}) = \begin{pmatrix} 4x_1^2 + 4x_2^2 \\ (x_1-5)^2 + (x_2-5)^2 \end{pmatrix} \\
\text{subject to:} \quad & (x_1-5)^2 + x_2^2 \leq 25 \\
& (x_1-8)^2 + (x_2+3)^2 \geq 7.7 \\
\text{by changing:} \quad & (x_1, x_2) \in [0,5] \times [0,3]
\end{aligned} \tag{III.36}$$

The acceptable design area and its counterpart in the objective space are represented in Figure III.5. The thick black curve represents the Pareto optimal set in the design space. New designs are chosen randomly in $[0,5] \times [0,3]$ under the optimization geometric constraints.

**Remark** *Note that here, constraints are assumed to be known, so that the tunable accuracy strategy is only performed on the objective functions.*

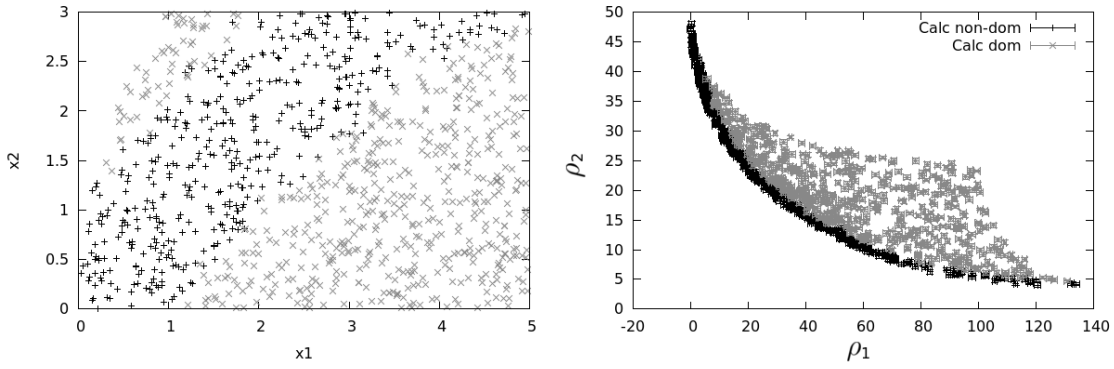
Figures III.6 and III.7 depict the results, in terms of optimal designs and objectives, for the classical and Fusi approaches, respectively. These optimizations are performed with 100 optimization iterations and $\bar{\boldsymbol{s}}_2 = 1\%$. This value means that the threshold




(a) (b)

Figure III.5 Test-case 1, a) feasible area in red and Pareto optima (thick line), b) image in the objective space.

equals 1% of the objective range in each dimension, as defined in Equation (III.35). Because of the low gradient around the optima, the non-dominated designs (in black) cover a quite wide area. Dominated designs and boxes are drawn in grey.



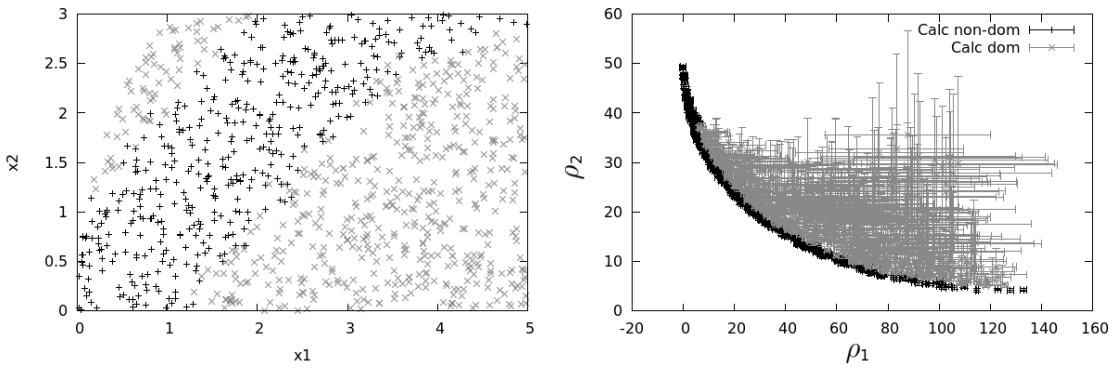

(a) (b)

Figure III.6 Test-case 1, *Class approach*, non-dominated designs in black and dominated ones in grey: a) design space, b) objective space.

The approximated Pareto front of these two strategies are qualitatively very similar, and both provided the same Pareto-optimal design area. However, significant portions of the boxes are not refined up to $s_2$ in the recursive approach (Figure III.7), thus reducing the number of function evaluations and the computational cost of the optimization.

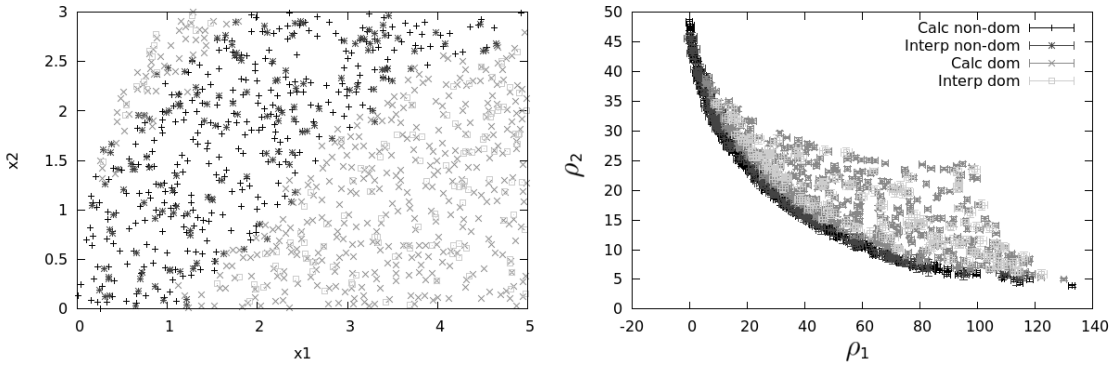
Figures III.8 and III.9 picture the impact of the Surrogate-Assisting (SA) strategy on the classical and recursive (Fusi) approaches, respectively. Figure III.9 performs the optimization with a Surrogate-Assisted recursive approach, which corresponds to SABBa. Thresholds $\bar{s}_2 = 1\%$ and $\bar{s}_1 = 1\%$ are used.

SA-based designs and boxes are drawn in dark grey when they are non-dominated and light grey otherwise. One may note that the Pareto-optimal design area and the approximated Pareto front are similar to the fully-computed ones from Figure III.6.



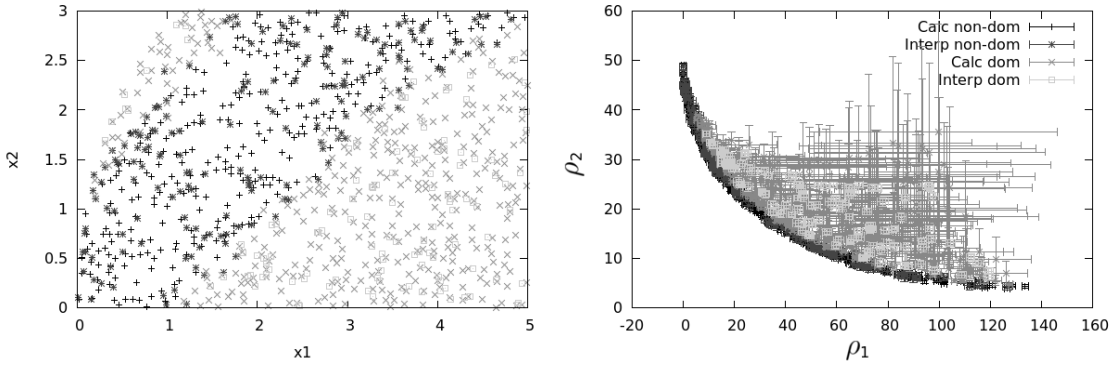

(a) (b)

Figure III.7 Test-case 1, *Fusi approach*, non-dominated designs in black and dominated one in grey: a) design space, b) objective space.




(a) (b)

Figure III.8 Test-case 1, *Class-acc approach*, non-dominated interpolated designs added in dark grey, dominated interpolated designs in light grey. a) design space, b) objective space.




(a) (b)

Figure III.9 Test-case 1, *SABBa*, non-dominated interpolated designs added in dark grey, dominated interpolated designs in light grey: a) design space, b) objective space.

We compare the number of function evaluations needed for each strategy. As raised before, because the tools are analytically simulated, the number of function evaluations only allows for asymptotic comparison and should not be confronted with real applications. The results are reported in Figure III.10 (a):

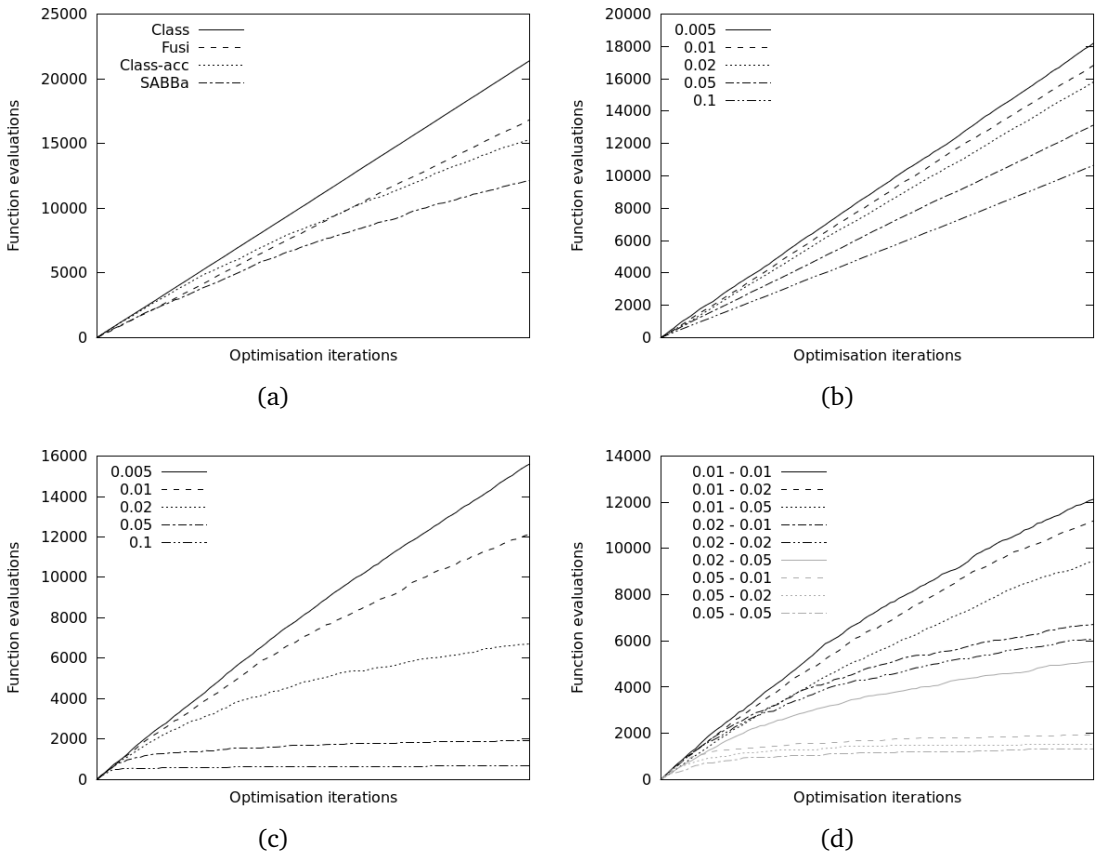



Figure III.10 Test-case 1: a) number of function evaluations for each approach ($\bar{s}_1 = \bar{s}_2 = 0.01$), b) Fusi's approach with varying $\bar{s}_2$, c) SABBa with varying $\bar{s}_1$ at $\bar{s}_2 = 0.01$ ; d) SABBa with varying $\bar{s}_1$ and $\bar{s}_2$.

- The recursive (Fusi [Fusi and Congedo, 2016]) strategy appears to be more parsimonious than the classical approach. The slope of the cost curve is reduced by a ratio that can be interpreted as the mean refinement level using the recursive strategy with respect to the full refinement. Figures III.6 (a) to III.9 (a) show that nearly half of the designs are non-dominated in this test-case. Each of these corresponds to a fully refined box, which limits the computational gains.

- The SA strategy is more and more effective throughout the optimization process. During the first iterations, the surrogate is not yet accurate, and the slope is similar to non-assisted strategies. But in the later iterations, surrogate-assisted strategies show significant cost reduction.

The influence of the user-defined thresholds $\bar{s}_1$ and $\bar{s}_2$ is also investigated:

- Figure III.10 (b) pictures the substantial impact of $\bar{s}_2$ on the computational cost. This trend comes from the high number of accurate boxes. Decreasing the threshold enforces higher accuracy for all these boxes but significantly rises the global cost. The critical repercussion on the accuracy of the Pareto-optimal set is qualitatively drawn in Fig. III.11.

- Figures III.10 (c) and III.10 (d) illustrate the momentous cost reduction that can be achieved with the SA strategy, at the expense of Pareto front accuracy. One should, however, be careful not to underestimate the surrogate error, *e.g.* when high local variations are not yet captured.

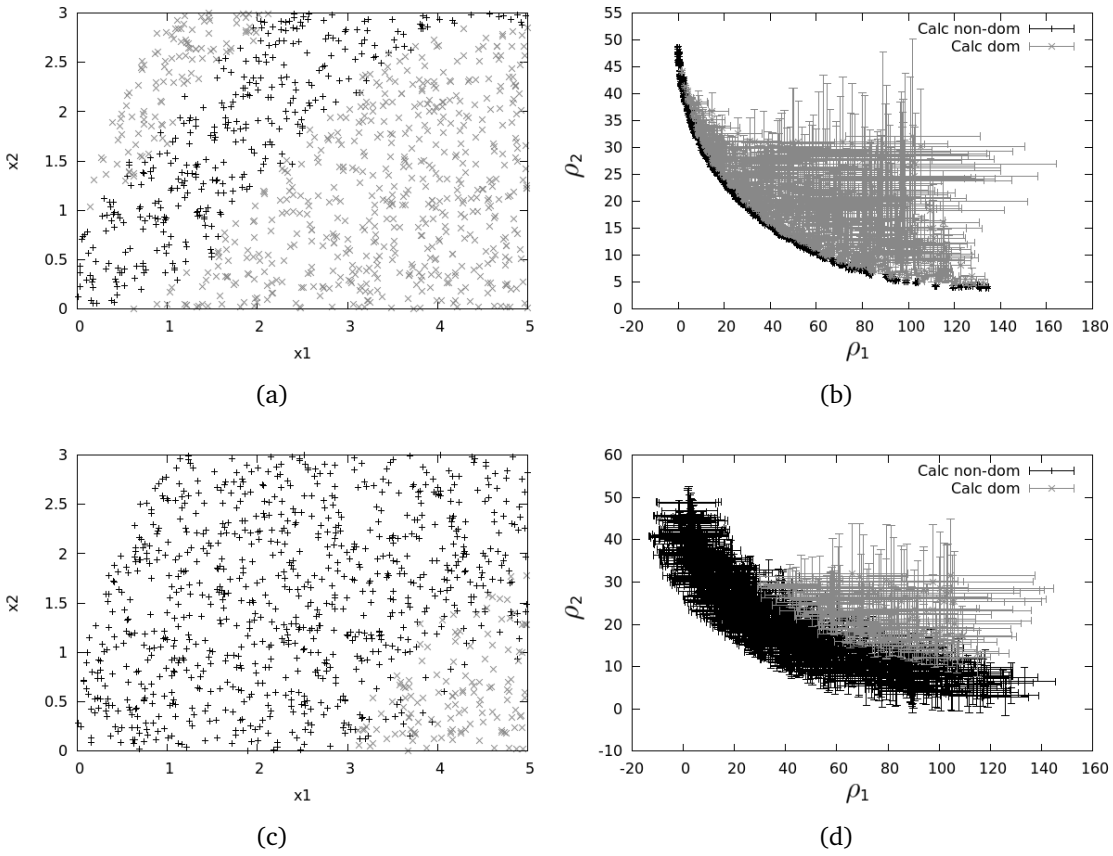



Figure III.11 Impact of $\bar{s}_2$: Outputs with $\bar{s}_2 = 0.005$ in a) design space and b) objective space; Outputs with $\bar{s}_2 = 0.1$ in c) design space and d) objective space.

### III–3.2.2 Test-case 2: The Triangle problem

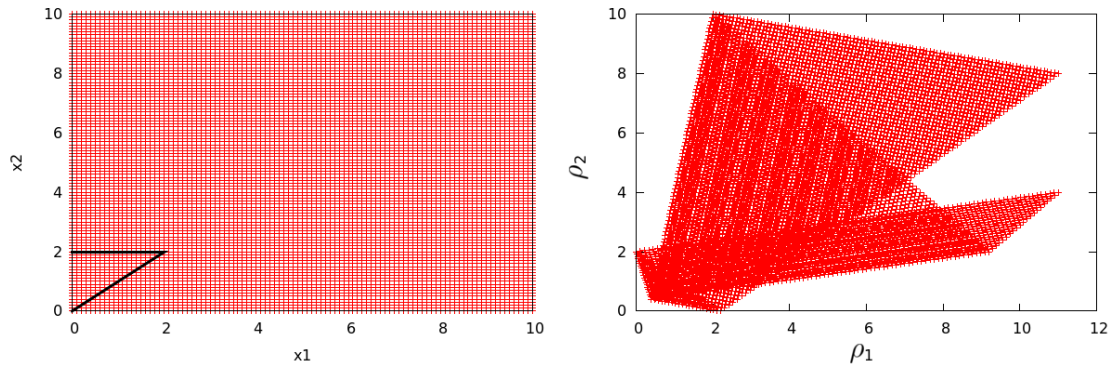
This problem is a bi-objective unconstrained optimization:

$$\begin{aligned} \text{minimize:} \quad & \boldsymbol{f}(\boldsymbol{x}) = \begin{pmatrix} \frac{x_1+x_2}{10} + |x_1 - x_2| \\ \frac{x_1}{5} + |x_2 - 2| \end{pmatrix} \\ \text{by changing:} \quad & (x_1, x_2) \in [0,10]^2 \end{aligned} \tag{III.37}$$

The design space with the optimal set (represented with a thick black curve) and the objective space counterpart are represented in Fig. III.12.

This small optimal area is well fit to study the behavior of SABBa when the optimizer converges toward the Pareto front. In practice, two variants are compared. At each iteration, 10 new random designs are given to SABBa: a) in $\mathcal{X} = [0,10]\times[0,10]$; b) in a domain $\mathcal{X}^k$ converging toward $[0,2]\times[0,2]$, with $\mathcal{X}^0 = \mathcal{X}$.

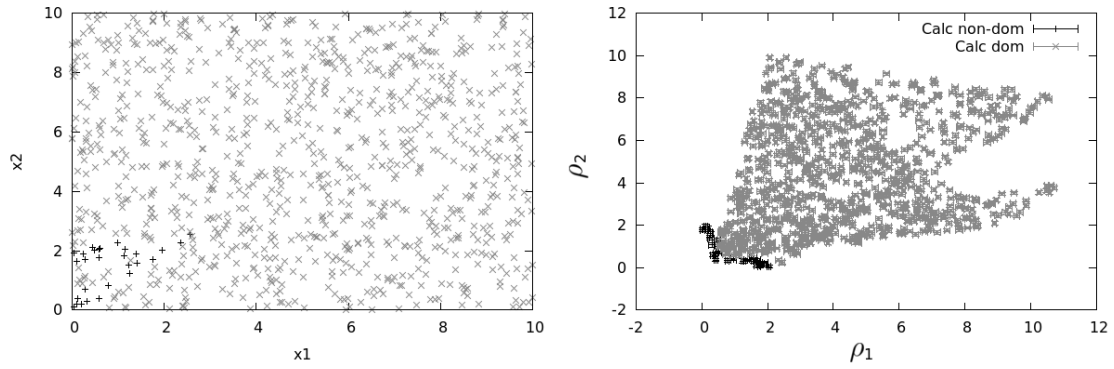
We aim at replicating the slow-down of the Bounding-Box approach raised in [Fusi and Congedo, 2016], and at quantifying how the SA strategy can address this issue.




(a) (b)

Figure III.12 Test-case 2: a) input space and Pareto optima (thick line), b) image in the objective space.

**No convergence of the designs** In this variant, the ten new designs are chosen randomly within the $[0,10]^2$ design space. The approximation of the Pareto front is very poor and highly inefficient, as one could expect. The result is pictured in Fig. III.13.




(a) (b)

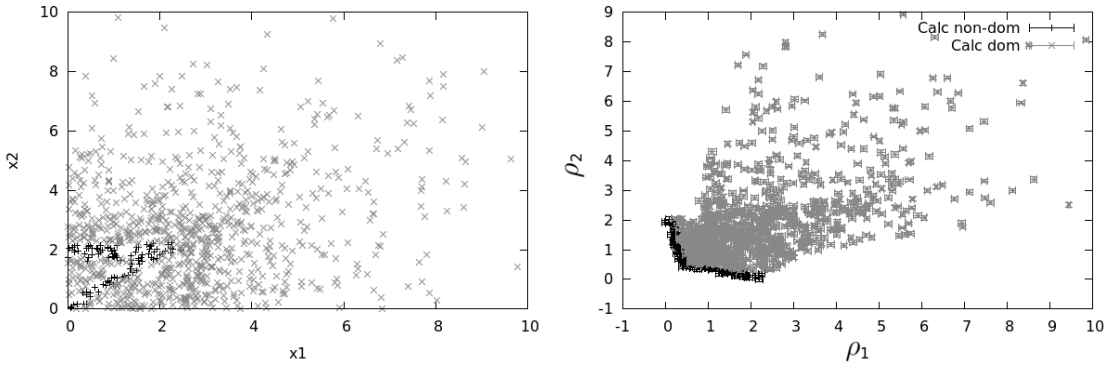
Figure III.13 Test-case 2: *Classical approach*, non-converging domain for new designs: a) design space, b) objective space.

**Convergence of the designs** Here, the 10 new designs are randomly chosen in a domain that converges toward the optimal area. In practice, with $j$ the optimization iteration, the new designs are drawn as follows, for $i$ in $\{1,2\}$,

$$x_i = X\Big(2+8\exp\Big(-\frac{j}{45}\Big)\Big), \quad \text{with } X \sim \mathcal{U}[0,1].$$

This gives a better refinement of the Pareto front, as can be seen comparing Figs. III.13 and III.14.

The number of function evaluations is depicted in Fig. III.15, where the different strategies are represented. As raised in [Fusi and Congedo, 2016], the convergence toward the optimal area induces a decreasing impact of the recursive strategy. In

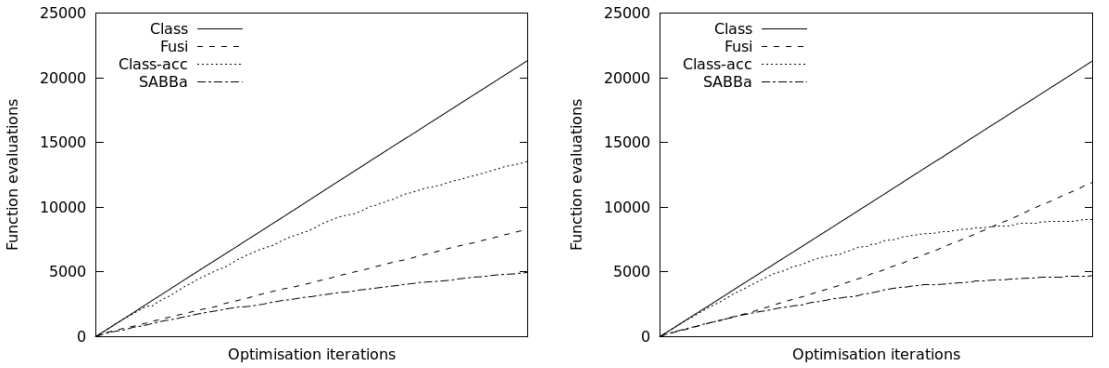



(a) (b)

Figure III.14 Test-case 2: *Classical approach*, converging domain for new designs: a) design space, b) objective space.

consequence, the slope of the associated cost curve (dashed) slowly relapses to the slope of the classical strategy (solid) in Fig. III.15 (b). Intuitively, this is due to the increasing percentage of non-dominated boxes among the new designs.

However, quite surprisingly, one can see in Fig. III.15 (b) that both surrogate-assisted strategies (*Class-acc* and *SABBa*) reach their cost plateau faster with designs converging toward the optimal area. Instead of being a drawback, good behavior of the optimization process gives even more parsimony to SA-based methods. Intuitively, this behavior comes from the fact that the surrogate model only needs to be highly refined in the optimal area to bypass future computations.

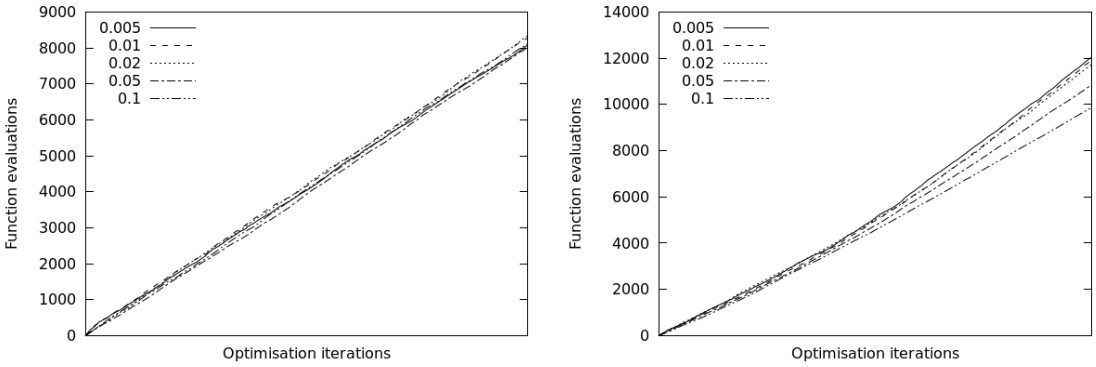



(a) (b)

Figure III.15 Test-case 2: number of function evaluations depending on the strategy for a) the non-converging case, b) the converging case.

Fig. III.15 (a) can also be compared with Fig. III.10 (a). Contrarily to the first test-case, Fig. III.13 has a small Pareto front. A majority of the boxes are hence dominated, which permits the recursive strategy to generate higher computational gains.

**Remark** *The increasing number of non-dominated designs in the converging case also induces an increasing impact of the threshold $\bar{\mathbf{s}}_2$ throughout the optimization process. This trend is depicted in Figure III.16.*




(a) (b)

Figure III.16 Test-case 2: number of function evaluations (*Fusi* strategy) depending on $\bar{s}_2$ for a) the non-converging case, b) the converging case.

### III–3.2.3 Test-case 3: The Kursawe problem

The Kursawe bi-objective optimization is performed here to study the behavior of SABBa on a more complex Pareto front shape. The problem is defined as follows:

$$\begin{aligned} \text{minimize:} \quad & \boldsymbol{f}(\boldsymbol{x}) = \begin{pmatrix} \sum_{i=1}^{2} \left[ -10 \exp\left( -0.2\sqrt{x_i^2 + x_{i+1}^2} \right) \right] \\ \sum_{i=1}^{3} \left[ |x_i|^{0.8} + 5\sin(x_i^3) \right] \end{pmatrix} \\ \text{by changing:} \quad & (x_1, x_2, x_3) \in [-5, 5]^3 \end{aligned} \tag{III.38}$$

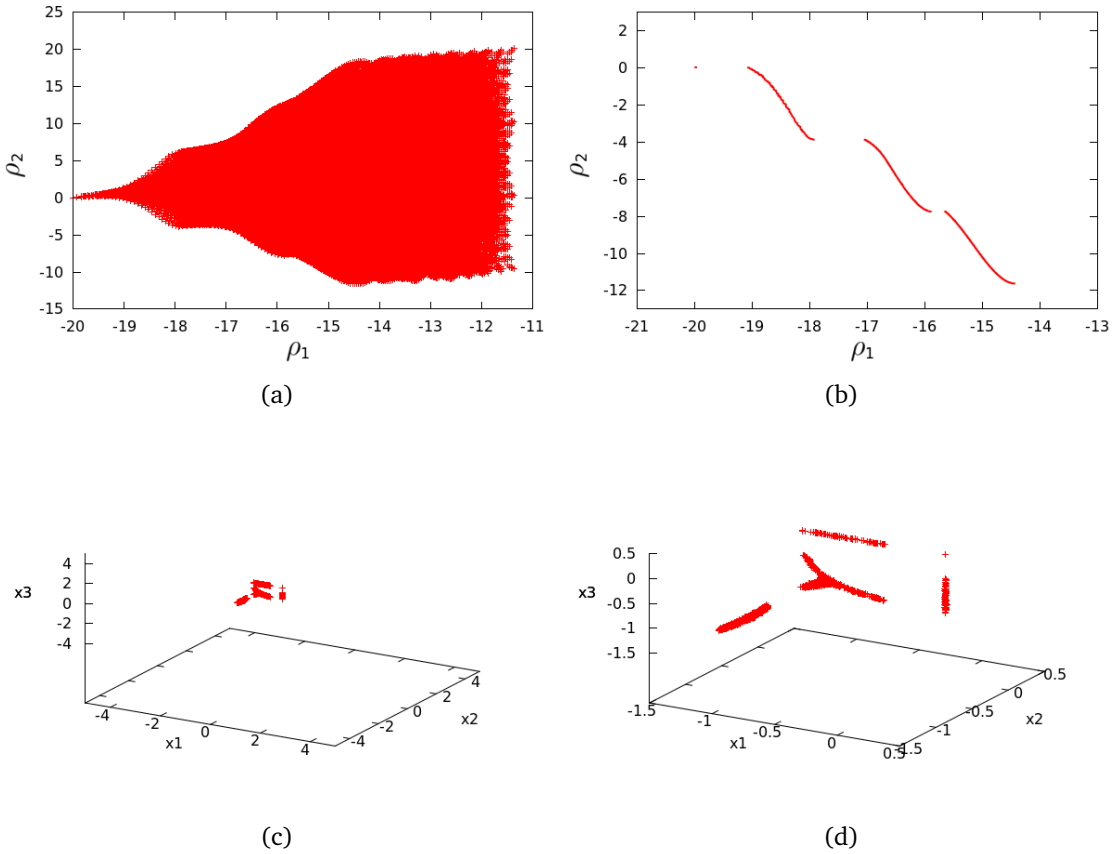
This problem is known to have a complex Pareto front with several discontinuities, associated with a small area of the design space. Convergence toward the optimal zone is again forced, similarly to test-case 2 b). We aim to show that SABBa is not influenced by the complexity or dimensionality of the optimization problem. In practice, these issues must be taken into account, and suitable choices for the optimizer and surrogate model must be made.

The highly discontinuous Pareto front associated with the problem is given in Fig. III.17. Because of the numerous dominated individuals in Fig. III.17 (a), the recursive strategy should have a high impact on the overall cost. The Pareto-optimal designs are represented in Fig. III.17 (c) in the whole $[-5, 5]^3$ domain and in Fig. III.17 (d) in a close-up view. These designs form complex disjoint sets, and their approximations will be qualitatively compared. SABBa is applied in the following with different thresholds.

First, Fig. III.18 gives the outputs obtained with a fine threshold. The results are highly comparable to Fig. III.17, with only 28510 function evaluations over 48913 for the classical approach (**gain = 42%**).

With larger thresholds, the optimal area is less accurately captured, as seen in Fig. III.19 (a), and the associated approximated Pareto front is depicted with higher imprecision in Fig. III.19 (b). Here, 8294 function evaluations were required (**gain = 83%**).

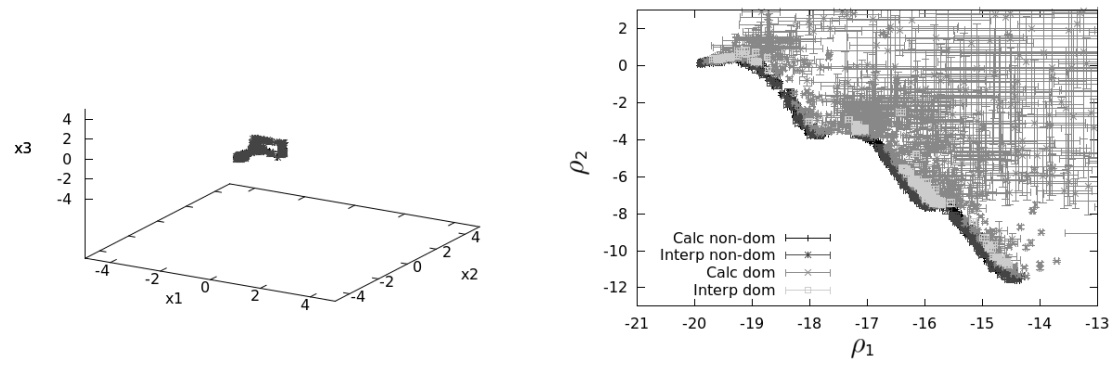
With very coarse thresholds, the local optimal shape is not captured, as depicted in Fig. III.20. However, it gives a first rough approximation of the optimal area with only a computational cost of 3546 evaluations (**gain = 93%**).




(a) (b)

(c) (d)

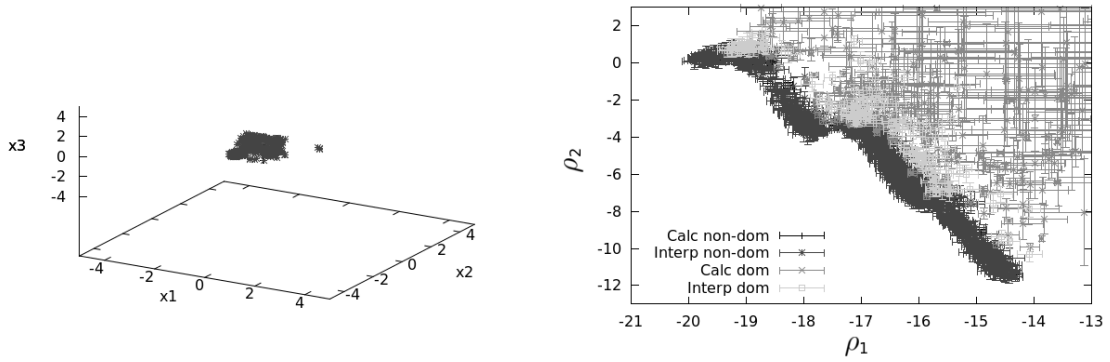
Figure III.17 Test-case 3: a) image in the objective space, b) associated Pareto front, c) Pareto optima in the design space, d) in a close-up view.




(a) (b)

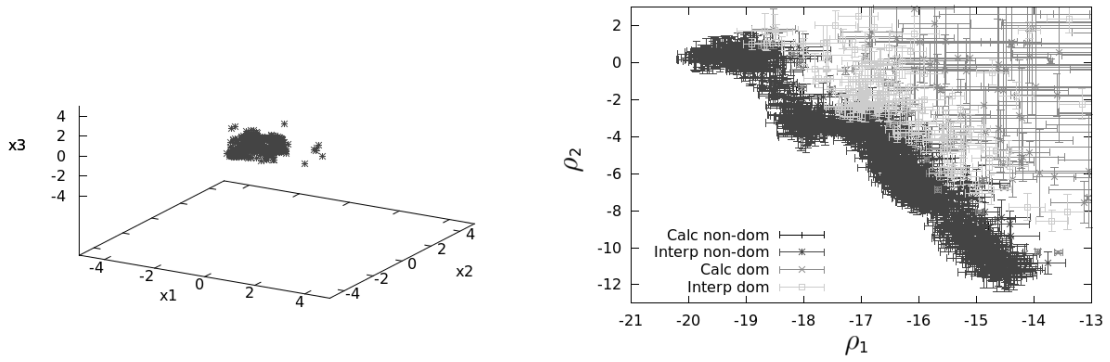
Figure III.18 Test-case 3: SABBa outputs with fine thresholds in a) the design space, b) the objective space. Computational cost saving: 42%.

The cost graphs are given in Fig. III.21. Similarly to Figs. III.15 (b) and III.16 (b), Figs. III.21 (a) and III.21 (b) reveal a decreasing impact of the recursive strategy alone and an increasing sensitivity to $\bar{s}_2$ throughout the optimization. Figure III.21 (c) and III.21 (d) show that the computational cost associated to SABBa is here highly influenced by the threshold $\bar{s}_1$, that controls the accuracy of the SA model.




(a) (b)

Figure III.19 Test-case 3: SABBa outputs with moderate thresholds in a) the design space, b) the objective space. Computational cost saving: 83%.




(a) (b)

Figure III.20 Test-case 3: SABBa outputs with coarse thresholds in a) the design space, b) the objective space. Computational cost saving: 93%.

SABBa yields a significant decrease in the computational cost. Note anyway that the user-defined thresholds heavily impact both the accuracy and the global computational cost, and they should be chosen with care in real-life applications.

The efficiency of the SABBa framework is not directly correlated with the complexity of the optimization problem. The user should choose the most relevant optimization and surrogate modeling techniques for tackling high dimensions, non-linearity or other difficulties. In this context, SABBa can bring consistent cost reduction through fine allocation of the high-accuracy estimations and robust use of the Surrogate-Assisting model.

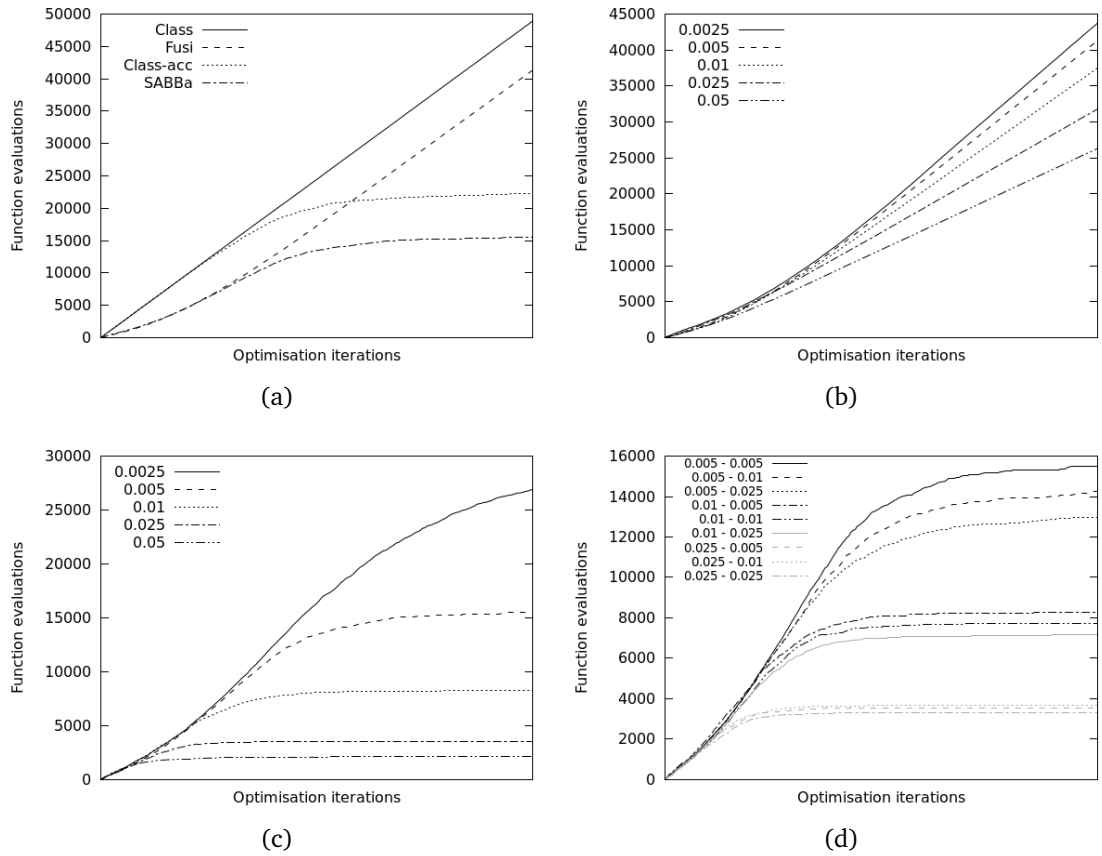



Figure III.21 Test-case 3: a) number of function evaluations for each method of interest ($\bar{s}_1 = \bar{s}_2 = 0.01$), b) Fusi approach with varying $\bar{s}_2$, c) SABBa with varying $\bar{s}_1$ at $\bar{s}_2 = 0.01$ ; d) SABBa with varying $\bar{s}_1$ and $\bar{s}_2$.

**Discussion** *The SABBa strategy is quantitatively analyzed in this section in the context of noisy optimization problems with tunable accuracy. In practice, analytical test-cases have been artificially noised, and both the optimization and surrogate modeling strategies are numerically simulated.*
*On these artificial test-cases, the asymptotic behavior of the different approaches has been compared. The Bounding-Box approach from [Fusi and Congedo, 2016] yields a consistent cost reduction compared to the classical nested strategy by allocating the refinements only on promising, i.e. non-dominated, designs. However, when the optimizer converges toward the optimal area and only proposes promising designs, this approach relapses to the costly classical one. Also, the Surrogate-Assisting strategy is shown to yield substantial cost reduction in the latest iterations, when the surrogate is converged to $\boldsymbol{s}_1$. Contrarily to the Bounding-Box approach, SABBa takes advantage of the convergence toward the optimal area to yield an accurate SA model quickly.*
*The coupling between the Surrogate-Assisting strategy and the Bounding-Box approach seems very appealing and parsimonious. In the next section, efficient formulations are proposed for applying SABBa to the problem of uncertainty-based optimization.*

# III–4 Optimization under uncertainty

In this section, we propose efficient formulations and strategies for applying SABBa to the problem of optimization under uncertainty. Robustness and reliability measures are approximated with tunable and refinable accuracy by constructing surrogate models on some of the training points. Boxes widths can then be estimated thanks to the predictive error of the chosen surrogate models.

The computation and refinement scheme of the Bounding-Boxes are first presented in Section III–4.1. Some computational details are then given in Section III–4.2 for the calculation of Pareto-Optimal Probability, and a quantitative quality indicator is proposed in Section III–4.3 for assessing the performance of SABBa. Finally, Section III–4.4 summarizes the algorithm for performing uncertainty-based optimization with SABBa.

Section III–4.5 is devoted to the assessment of the proposed method on two analytical problems.

## III–4.1 Measures computation and refinement

The construction of boxes requires the assessment of robustness and reliability measures and the associated errors. As raised earlier, we choose to focus on expectation, variance, extremum (minimum or maximum), and quantile measures.

The values returned to the optimization process are denoted as $\boldsymbol{\rho}_{SABBa}^{s_1,k(s_2)}(\boldsymbol{x})$ and $\overline{\boldsymbol{\varepsilon}}_{SABBa}^{s_1,k(s_2)}(\boldsymbol{x})$, as defined in Equation (III.32). By definition, they can be computed with tunable accuracy using the Bounding-Box approach or can be returned by the SA model. These two computations are presented in details in Sections III–4.1.1 and III–4.1.2, respectively.

Note that we use Gaussian Processes as surrogate models, both for the SA model and for the computation of the robustness and reliability measures. Practically, with $\boldsymbol{\sigma}^2(\boldsymbol{x})$ the predictive variance of a GP surrogate model, we propose to follow a $\pm 3\sigma$ paradigm and to return the following error:

$$\overline{\boldsymbol{\varepsilon}}(\boldsymbol{x}) = 3\boldsymbol{\sigma}(\boldsymbol{x}).$$

The use of this strategy in the context of SA-based and computed boxes is made explicit in the following two sections.

**Remark** *The $\pm 3\sigma$ paradigm does not imply $\overline{\boldsymbol{\varepsilon}}(\boldsymbol{z}) \geq |\boldsymbol{\varepsilon}(\boldsymbol{z})|$, where $\boldsymbol{\varepsilon}$ is the true error of the surrogate. However, the probability of dissatisfying this inequality is very low, which should allow for a good overall behavior of the approach.*

### III–4.1.1 Computed boxes

For each design $\boldsymbol{x}$ that does not meet the criterion $\overline{\boldsymbol{\varepsilon}}_{SA}(\boldsymbol{x}) \leq s_1$, the associated box must be computed and refined with the Bounding-Box approach. To perform such task, we propose to construct a surrogate model $\widehat{\boldsymbol{q}}$ of $\boldsymbol{q}$ based on a given number of training data, that may be increased for refinement. More precisely, at a given design $\boldsymbol{x}$, measure $\boldsymbol{\rho}(\boldsymbol{x})$ can be approximated from a surrogate model $\widehat{\boldsymbol{q}}(\boldsymbol{x},\boldsymbol{\xi})$ in the uncertain space $\Xi$. The true functions are denoted $\boldsymbol{q}(\boldsymbol{x},\cdot)$ and will be approximated with Gaussian Process (GP) surrogate models. Two approaches are considered:

- *Separated Spaces:* A GP is built in $\Xi$ at a given $\boldsymbol{x}$ using a set of samples $\left\{\boldsymbol{q}(\boldsymbol{x},\boldsymbol{\xi}_i)\right\}_i$. This GP directly corresponds to $\widehat{\boldsymbol{q}}(\boldsymbol{x},\cdot)$.
- *Coupled Space:* A GP is built in $\mathcal{X}\times\Xi$ using a set of samples $\left\{\boldsymbol{q}(\boldsymbol{x}_i,\boldsymbol{\xi}_i)\right\}_i$. This gives an approximate $\widehat{\boldsymbol{q}}$ of $\boldsymbol{q}$ in the whole coupled space. At any given $\boldsymbol{x}$, the cut $\widehat{\boldsymbol{q}}(\boldsymbol{x},\cdot)$ can then be returned.

**Remark** *While the Separated Spaces (SS) approach builds a GP in a space of lower dimension ($Card(\Xi)$) than the Coupled Space (CS) approach ($Card(\mathcal{X}\times\Xi)$), thus easing the surrogate modeling task, it does not use surrounding samples in the design space and restarts from scratch at each new design point $\boldsymbol{x}$. These strategies will be compared on analytical test-cases in Section III–4.5.*

Both SS and CS approaches give a predictive model $\widehat{\boldsymbol{q}}(\boldsymbol{x},\cdot)$, that will be written $\widehat{\boldsymbol{q}}_{\boldsymbol{x}}$, and a predictive variance $\boldsymbol{\sigma}_{\boldsymbol{q}}(\boldsymbol{x},\cdot)$. The accuracy $l$ of these models, that refers to $\widetilde{\boldsymbol{\rho}}^l(\boldsymbol{x})$ and $\overline{\boldsymbol{\varepsilon}}^l(\boldsymbol{x})$ from the previous sections, corresponds here to the number of training data used for constructing $\widehat{\boldsymbol{q}}$. The $\pm 3\sigma$ paradigm then provides the box widths, denoted by $\overline{\boldsymbol{\varepsilon}}_{\boldsymbol{q}_{\boldsymbol{x}}} = 3\boldsymbol{\sigma}_{\boldsymbol{q}}(\boldsymbol{x},\cdot)$.

For each output measure, indexed by $i$, the classical empirical estimators are exploited for the computation of $\widetilde{\rho}_i^l(\boldsymbol{x})$:

$$\widetilde{\rho}_i^l(\boldsymbol{x}) = \begin{cases} \widetilde{\mu}(\boldsymbol{x}) = \dfrac{1}{N_\xi}\sum_{j=1}^{N_\xi}\widehat{q}_{\boldsymbol{x}}(\xi_j) & \text{for expectation,} \\ \widetilde{\sigma^2}(\boldsymbol{x}) = \dfrac{1}{N_\xi-1}\sum_{j=1}^{N_\xi}\left(\widehat{q}_{\boldsymbol{x}}(\xi_j)-\widetilde{\mu}(\boldsymbol{x})\right)^2 & \text{for variance,} \\ \widetilde{m}(\boldsymbol{x}) = \min\limits_{j\in[\![1,N_\xi]\!]}\widehat{q}_{\boldsymbol{x}}(\xi_j) & \text{for minimum,} \\ \widetilde{M}(\boldsymbol{x}) = \max\limits_{j\in[\![1,N_\xi]\!]}\widehat{q}_{\boldsymbol{x}}(\xi_j) & \text{for maximum,} \\ \widetilde{\mathrm{q}^p}(\boldsymbol{x}) = q^{ord}_{\lfloor pN_\xi\rfloor} & \text{for quantile,} \end{cases} \tag{III.39}$$

where $\left(q_j^{ord}\right)_j$ is the ordered list of all values $\left\{\widehat{q}_{\boldsymbol{x}}(\xi_j)\right\}_{j=1}^{N_\xi}$ in increasing order. The proposed associated width $\overline{\varepsilon}_i^l(\boldsymbol{x})$ are given in Eq. (III.40):

$$\overline{\varepsilon}_i^l(\boldsymbol{x}) = \begin{cases} \overline{\varepsilon}_\mu(\boldsymbol{x}) = \mathbb{E}_\xi\left[\overline{\varepsilon}_{q_{\boldsymbol{x}}}(\xi)\right], \\ \overline{\varepsilon}_{\sigma^2}(\boldsymbol{x}) = \mathbb{E}_\xi\left[\left(\overline{\varepsilon}_\mu(\boldsymbol{x})+\overline{\varepsilon}_{q_{\boldsymbol{x}}}(\xi)\right)^2 + 2\left|\widehat{q}_{\boldsymbol{x}}(\xi)-\widetilde{\mu}(\boldsymbol{x})\right|\left(\overline{\varepsilon}_\mu(\boldsymbol{x})+\overline{\varepsilon}_{q_{\boldsymbol{x}}}(\xi)\right)\right], \\ \overline{\varepsilon}_m(\boldsymbol{x}) = \max\left(\left|\widetilde{m}(\boldsymbol{x})-\min_\xi[\widehat{q}_{\boldsymbol{x}}^-(\xi)]\right|, \left|\widetilde{m}(\boldsymbol{x})-\min_\xi[\widehat{q}_{\boldsymbol{x}}^+(\xi)]\right|\right), \\ \overline{\varepsilon}_M(\boldsymbol{x}) = \max\left(\left|\widetilde{M}(\boldsymbol{x})-\max_\xi[\widehat{q}_{\boldsymbol{x}}^-(\xi)]\right|, \left|\widetilde{M}(\boldsymbol{x})-\max_\xi[\widehat{q}_{\boldsymbol{x}}^+(\xi)]\right|\right), \\ \overline{\varepsilon}_{\mathrm{q}^p}(\boldsymbol{x}) = \max\left(\left|\widetilde{\mathrm{q}^p}(\boldsymbol{x})-\mathrm{q}^p_\xi[\widehat{q}_{\boldsymbol{x}}^-(\xi)]\right|, \left|\widetilde{\mathrm{q}^p}(\boldsymbol{x})-\mathrm{q}^p_\xi[\widehat{q}_{\boldsymbol{x}}^+(\xi)]\right|\right). \end{cases} \tag{III.40}$$

In the above, $\widehat{q}_{\boldsymbol{x}}^+(\xi) = \widehat{q}_{\boldsymbol{x}}(\xi)+\overline{\varepsilon}_{q_{\boldsymbol{x}}}(\xi)$ and $\widehat{q}_{\boldsymbol{x}}^-(\xi) = \widehat{q}_{\boldsymbol{x}}(\xi)-\overline{\varepsilon}_{q_{\boldsymbol{x}}}(\xi)$. In practice, the statistics above, such as expectations and quantiles, are approximated by means of Monte Carlo Sampling (MCS) on surrogate model, similarly to (III.39). A formal justification for getting these box sizes is given in Section III–4.1.3.

We also provide guidance for refining the surrogate model $\widehat{\boldsymbol{q}}_{\boldsymbol{x}}(\xi)$. A valid option could be a space-filling paradigm, geometrically filling the uncertain space, but we choose here to use GP-based refinement criteria, that are measure-dependent. A par-

tial criterion is first computed for each statistical measure, using the following formulas:

$$c_i(\xi) = \begin{cases} \overline{\varepsilon}_{q_x}(\xi)\phi(\xi) & \text{for expectation,} \\ \overline{\varepsilon}_{q_x}(\xi)\phi(\xi) & \text{for variance,} \\ \left[\dfrac{\widetilde{m} - \widehat{q_x^-}(\xi)}{2\overline{\varepsilon}_{q_x}(\xi)}\right]_+ & \text{for minimum,} \\ \left[\dfrac{\widehat{q_x^+}(\xi) - \widetilde{M}}{2\overline{\varepsilon}_{q_x}(\xi)}\right]_+ & \text{for maximum,} \\ \left[\dfrac{\widetilde{q^p} - \widehat{q_x^-}(\xi)}{2\overline{\varepsilon}_{q_x}(\xi)}\right]_+ \left[\dfrac{\widehat{q_x^+}(\xi) - \widetilde{q^p}}{2\overline{\varepsilon}_{q_x}(\xi)}\right]_+ \phi(\xi) & \text{for quantile,} \end{cases} \tag{III.41}$$

with the same definition of $\widehat{q_x^+}(\xi)$ and $\widehat{q_x^-}(\xi)$ as before and where $\left[\,\cdot\,\right]_+ = \max(0,\cdot)$ is the positive part function. Note that we heuristically choose here to multiply the criteria for the mean, variance and quantile measures by the input Probability Density Function (PDF) $\phi(\xi)$ in order to put more weight on the most likely area. Justifications for these formulas are provided in III–4.1.3.

These partial criteria are then scalarized into the global refinement criterion through the following weighted sum:

$$c(\xi) = \sum_{i=1}^{m} w_i \bar{c}_i(\xi) \tag{III.42}$$

where $\bar{c}_i(\xi)$ are the normalized partial criteria, $m$ is the number of measures and $w_i$ are weighting coefficients. The normalized partial criteria are computed as follows:

$$\bar{c}_i(\xi) = \frac{c_i(\xi) - \min\limits_{\xi}[c_i(\xi)]}{\max\limits_{\xi}[c_i(\xi)] - \min\limits_{\xi}[c_i(\xi)]} \in [0,1]. \tag{III.43}$$

Finally, we propose here to compute the weights as the ratio between the conservative error $\overline{\varepsilon}_i^l$ and the target accuracy $\boldsymbol{s}_{2_i}$. In this manner, any partial criterion associated with high error compared to the target accuracy will heavily influence the final criterion. Practically, to emphasize this behavior, the dependence is chosen to be quadratic:

$$w_i = \left(\frac{\overline{\varepsilon}_i^l}{s_{2_i}}\right)^2. \tag{III.44}$$

**Remark** *Note that in the case of multi-point refinement, a greedy sequential approach can be applied by assuming that previous refinements are performed, fixing the predictive values of the GP surrogate model and recomputing the predictive variance to obtain the updated refinement criterion. We can then perform Black-box evaluations in parallel on these points. This approach is usually called the Kriging Believer strategy and allows performing multi-point refinement efficiently without the need of any clustering or local penalization heuristics (see review [Haftka et al., 2016] for more insights). However, this strategy makes a lot of assumptions and is usually sub-optimal.*

### III–4.1.2 SA-based boxes

When the SA model error is below $\boldsymbol{s}_1$ in all output dimension at a new design $\boldsymbol{x}$, the predictive value $\boldsymbol{\rho}_{SA}(\boldsymbol{x})$ is directly returned to the optimizer. The SA model is systematically built using a Gaussian Process.

In practice, this model is constructed on the designs for which boxes have been already computed using formulas from the previous section. These training data $\widetilde{\boldsymbol{\rho}}^l(\boldsymbol{x})$ are associated with different box sizes $\overline{\boldsymbol{\varepsilon}}^l(\boldsymbol{x})$. Specifically, we use this last error to build an heteroscedastic GP with heterogeneous variance computed as $\boldsymbol{\sigma}_i^2 = \left(\frac{\overline{\boldsymbol{\varepsilon}}^l(\boldsymbol{x}_i)}{3}\right)^2$ using again the $\pm 3\sigma$ paradigm.

The measure estimations $\boldsymbol{\rho}_{SA}$ and associated error $\overline{\boldsymbol{\varepsilon}}_{SA}$ at a new design $\boldsymbol{x}$ are computed as follows:

$$\begin{aligned}
\boldsymbol{\rho}_{SA}(\boldsymbol{x}) &= \boldsymbol{k}_*^T (K+\Delta)^{-1} \widetilde{\boldsymbol{\rho}}^l \\
\overline{\boldsymbol{\varepsilon}}_{SA}(\boldsymbol{x}) &= 3\boldsymbol{\sigma}_{SA}(\boldsymbol{x}) = 3\big(k_{**} - \boldsymbol{k}_*^T (K+\Delta)^{-1} \boldsymbol{k}_*\big).
\end{aligned} \tag{III.45}$$

Matrix $K = K\big(\{\boldsymbol{x}_i\}_i, \{\boldsymbol{x}_i\}_i\big)$ represents the autocovariance matrix between training points, $k_{**} = k\big(\boldsymbol{x}, \boldsymbol{x}\big)$ is the current variance at $\boldsymbol{x}$ and $\boldsymbol{k}_* = \boldsymbol{k}\big(\boldsymbol{x}, \{\boldsymbol{x}_i\}_i\big)$ is the covariance vector between $\boldsymbol{x}$ and the training points. The diagonal matrix $\Delta = diag\big(\{\sigma_i^2\}_i\big)$ represents the heterogeneous gaussian noises presented previously. More details on GP surrogate models can be found in [Williams and Rasmussen, 2006].

### III–4.1.3 Computational details

We justify here the formulas proposed in the preceding sections for error computation and refinement strategy.

**Justification for box sizes** We explain here the choice of the conservative errors proposed in Equation (III.40).

**Remark** *Note that in the following, empirical estimators are considered well converged, so that all impreciseness comes from the surrogate model $\widehat{q}_x$ of $q(\boldsymbol{x}, \cdot)$. For example, $\widetilde{\mu}(\boldsymbol{x})$ is considered equal to $\mathbb{E}_\xi[\widehat{q}_x(\xi)]$ in the following formula although it is defined as an empirical mean in Equation* (III.39).

The expectation approximation conservative error $\overline{\varepsilon}_\mu$ is quite straightforward:

$$\begin{aligned}
\big|\varepsilon_\mu(\boldsymbol{x})\big| = \big|\mu(\boldsymbol{x}) - \widetilde{\mu}(\boldsymbol{x})\big| &= \big|\mathbb{E}_\xi[q(\boldsymbol{x},\xi)] - \mathbb{E}_\xi[\widehat{q}_x(\xi)]\big| \\
&= \big|\mathbb{E}_\xi[\widehat{q}_x(\xi) + \varepsilon_{q_x}(\xi)] - \mathbb{E}_\xi[\widehat{q}_x(\xi)]\big| \\
&= \big|\mathbb{E}_\xi[\varepsilon_{q_x}(\xi)]\big| \\
&\leq \mathbb{E}_\xi[\overline{\varepsilon}_{q_x}(\xi)] \\
&= \overline{\varepsilon}_\mu(\boldsymbol{x}),
\end{aligned}$$

with $\mu(\boldsymbol{x})$ the true statistical moment on $q$ at a given $\boldsymbol{x}$ and $\widetilde{\mu}(\boldsymbol{x})$ the approximated one, computed on the surrogate model $\widehat{q}_x$. The true error $\varepsilon_{q_x}$ is defined from $q(\boldsymbol{x},\xi) = \widehat{q}_x(\xi) + \varepsilon_{q_x}(\xi)$ and is conservatively approximated by $\overline{\varepsilon}_{q_x}(\xi) \geq |\varepsilon_{q_x}(\xi)|$.

The same reasoning can be done for the variance measure, with a higher risk of overestimation:

$$\begin{aligned}\left|\varepsilon_{\sigma^2}(\boldsymbol{x})\right| = \left|\sigma^2(\boldsymbol{x}) - \widetilde{\sigma^2}(\boldsymbol{x})\right| &= \left|\mathbb{E}_\xi[(q(\boldsymbol{x},\xi)-\mu(\boldsymbol{x}))^2] - \mathbb{E}_\xi[(\widehat{q}_{\boldsymbol{x}}(\xi)-\widetilde{\mu}(\boldsymbol{x}))^2]\right| \\ &= \left|\mathbb{E}_\xi[(\widehat{q}_{\boldsymbol{x}}(\xi)-\widetilde{\mu}(\boldsymbol{x})+\varepsilon_{q_{\boldsymbol{x}}}(\xi)-\varepsilon_\mu(\boldsymbol{x}))^2] - \mathbb{E}_\xi[(\widehat{q}_{\boldsymbol{x}}(\xi)-\widetilde{\mu}(\boldsymbol{x}))^2]\right| \\ &= \left|\mathbb{E}_\xi[(\varepsilon_{q_{\boldsymbol{x}}}(\xi)-\varepsilon_\mu(\boldsymbol{x}))^2 + 2(\widehat{q}_{\boldsymbol{x}}(\xi)-\widetilde{\mu}(\boldsymbol{x}))(\varepsilon_{q_{\boldsymbol{x}}}(\xi)-\varepsilon_\mu(\boldsymbol{x}))]\right| \\ &\leq \mathbb{E}_\xi[(\overline{\varepsilon}_{q_{\boldsymbol{x}}}(\xi)+\overline{\varepsilon}_\mu(\boldsymbol{x}))^2 + 2|\widehat{q}_{\boldsymbol{x}}(\xi)-\widetilde{\mu}(\boldsymbol{x})|(\overline{\varepsilon}_{q_{\boldsymbol{x}}}(\xi)+\overline{\varepsilon}_\mu(\boldsymbol{x}))] \\ &= \overline{\varepsilon}_{\sigma^2}(\boldsymbol{x})\end{aligned}$$

**Remark** *The proposed conservative error associated with the expectation approximation could also be justified based on the monotonicity of the expectation operator. By definition* $q(\boldsymbol{x},\cdot) \in [\widehat{q}_{\boldsymbol{x}} - \overline{\varepsilon}_{q_{\boldsymbol{x}}}, \widehat{q}_{\boldsymbol{x}} + \overline{\varepsilon}_{q_{\boldsymbol{x}}}]$*, where* $f \leq g \iff \forall \boldsymbol{x}, f(\boldsymbol{x}) \leq g(\boldsymbol{x})$*. The monotonicity of the expectation operator then gives* $\mathbb{E}_\xi[q(\boldsymbol{x},\xi)] \in \left[\mathbb{E}_\xi[\widehat{q}_{\boldsymbol{x}}(\xi) - \overline{\varepsilon}_{q_{\boldsymbol{x}}}(\xi)], \mathbb{E}_\xi[\widehat{q}_{\boldsymbol{x}}(\xi) + \overline{\varepsilon}_{q_{\boldsymbol{x}}}(\xi)]\right]$*, or equivalently* $\mu(\boldsymbol{x}) \in \left[\widetilde{\mu}(\boldsymbol{x}) - \mathbb{E}_\xi[\overline{\varepsilon}_{q_{\boldsymbol{x}}}(\xi)], \widetilde{\mu}(\boldsymbol{x}) + \mathbb{E}_\xi[\overline{\varepsilon}_{q_{\boldsymbol{x}}}(\xi)]\right]$*, thus giving the conservative error found in the above* $\overline{\varepsilon}_\mu(\boldsymbol{x}) = \mathbb{E}_\xi[\overline{\varepsilon}_{q_{\boldsymbol{x}}}(\xi)]$.

By defining $\widehat{q}_{\boldsymbol{x}}^+(\xi) = \widehat{q}_{\boldsymbol{x}}(\xi) + \overline{\varepsilon}_{q\boldsymbol{x}}(\xi)$ and $\widehat{q}_{\boldsymbol{x}}^-(\xi) = \widehat{q}_{\boldsymbol{x}}(\xi) - \overline{\varepsilon}_{q\boldsymbol{x}}(\xi)$, the monotonicity of the minimum, maximum and quantile operators can also be exploited to obtain conservative error approximations. For the case of the minimum measure, with $\widetilde{m} = \min_\xi[\widehat{q}_{\boldsymbol{x}}(\xi)]$,

$$\begin{aligned}m(\boldsymbol{x}) = \min_\xi[q(\boldsymbol{x},\xi)] &\in \left[\min_\xi[\widehat{q}_{\boldsymbol{x}}^-(\xi)], \min_\xi[\widehat{q}_{\boldsymbol{x}}^+(\xi)]\right] \\ &= \left[\widetilde{m}(\boldsymbol{x}) - \left|\widetilde{m}(\boldsymbol{x}) - \min_\xi[\widehat{q}_{\boldsymbol{x}}^-(\xi)]\right|, \widetilde{m}(\boldsymbol{x}) + \left|\widetilde{m}(\boldsymbol{x}) - \min_\xi[\widehat{q}_{\boldsymbol{x}}^+(\xi)]\right|\right] \\ &\in \left[\widetilde{m}(\boldsymbol{x}) - \overline{\varepsilon}_m(\boldsymbol{x}), \widetilde{m}(\boldsymbol{x}) + \overline{\varepsilon}_m(\boldsymbol{x})\right]\end{aligned}$$

where $\overline{\varepsilon}_m(\boldsymbol{x}) = \max\left(\left|\widetilde{m}(\boldsymbol{x}) - \min_\xi[\widehat{q}_{\boldsymbol{x}}^-(\xi)]\right|, \left|\widetilde{m}(\boldsymbol{x}) - \min_\xi[\widehat{q}_{\boldsymbol{x}}^+(\xi)]\right|\right)$.

The same idea can be followed for the maximum and quantile approximations, resulting in the following errors:

$$\begin{aligned}\overline{\varepsilon}_M(\boldsymbol{x}) &= \max\left(\left|\widetilde{M}(\boldsymbol{x}) - \max_\xi[\widehat{q}_{\boldsymbol{x}}^-(\xi)]\right|, \left|\widetilde{M}(\boldsymbol{x}) - \max_\xi[\widehat{q}_{\boldsymbol{x}}^+(\xi)]\right|\right) \\ \overline{\varepsilon}_{\mathrm{q}^p}(\boldsymbol{x}) &= \max\left(\left|\widetilde{\mathrm{q}^p}(\boldsymbol{x}) - \mathrm{q}_\xi^p[\widehat{q}_{\boldsymbol{x}}^-(\xi)]\right|, \left|\widetilde{\mathrm{q}^p}(\boldsymbol{x}) - \mathrm{q}_\xi^p[\widehat{q}_{\boldsymbol{x}}^+(\xi)]\right|\right)\end{aligned}$$

**Justification for refinement criteria** The chosen partial criteria for GP model refinement given in Equation (III.41) are explained here.

Both the expectation and variance are global measures over the whole domain. For this reason, it has been chosen to iteratively add a point at the maximum predictive conservative error, to converge the model on the entire space. This strategy is usually called Maximum Mean Square Predictive Error or MMSPE (see [Santner et al., 2003]. Note that the partial criteria are multiplied by the input pdf $\phi(\xi)$ to weight the predictive error according to the probability of occurrence. Thus, the final partial criteria are:

$$\begin{aligned}c_\mu(\xi) &= \overline{\varepsilon}_{q_{\boldsymbol{x}}}(\xi)\phi(\xi) \\ c_{\sigma^2}(\xi) &= \overline{\varepsilon}_{q_{\boldsymbol{x}}}(\xi)\phi(\xi)\end{aligned}$$

The minimum (resp. maximum) measure partial criteria is simply the probability of exceeding the current minimal (resp. maximal) value $\widetilde{m}$ (resp. $\widetilde{M}$). The conservative error boxes are considered as uniform distributions, which gives the following criteria:

$$c_m(\xi) = \left[\frac{\widetilde{m} - \widehat{q}_{\boldsymbol{x}}^-(\xi)}{2\overline{\varepsilon}_{q_{\boldsymbol{x}}}(\xi)}\right]_+$$
$$c_M(\xi) = \left[\frac{\widehat{q}_{\boldsymbol{x}}^+(\xi) - \widetilde{M}}{2\overline{\varepsilon}_{q_{\boldsymbol{x}}}(\xi)}\right]_+$$

where $\widehat{q}_{\boldsymbol{x}}^+(\xi) = \widehat{q}_{\boldsymbol{x}}(\xi) + \overline{\varepsilon}_{q\boldsymbol{x}}(\xi)$, $\widehat{q}_{\boldsymbol{x}}^-(\xi) = \widehat{q}_{\boldsymbol{x}}(\xi) - \overline{\varepsilon}_{q\boldsymbol{x}}(\xi)$ and again $\left[\,\cdot\,\right]_+ = \max(0,\cdot)$ the positive part function.

Finally, for the case of the quantile measure, it has been chosen to compute the product of the aforementioned probability of exceeding the quantile value, multiplied by the input density:

$$c_{\mathrm{q}^p}(\xi) = \left[\frac{\widetilde{\mathrm{q}^p} - \widehat{q}_{\boldsymbol{x}}^-(\xi)}{2\overline{\varepsilon}_{q_{\boldsymbol{x}}}(\xi)}\right]_+ \left[\frac{\widehat{q}_{\boldsymbol{x}}^+(\xi) - \widetilde{\mathrm{q}^p}}{2\overline{\varepsilon}_{q_{\boldsymbol{x}}}(\xi)}\right]_+ p(\xi)$$

This product is maximized on the hyperplane $\widehat{q}_{\boldsymbol{x}} = \mathrm{q}^p$. Multiplying the criteria by $\phi(\xi)$ puts more weight according to the probability of occurrence. The spread of the refinement points is very likely to be sub-optimal. However, acceptable behavior is obtained by performing the optimization by Monte Carlo sampling. With this approach, previous refinements create a surrounding penalization area by reducing the local predictive variance $\overline{\varepsilon}_{q_{\boldsymbol{x}}}$.

Much more complex refinement criteria could be chosen here, to improve the parsimony of the whole approach further. However, the proposed choices give a fast determination of the next point, and the chosen refinement strategy usually has a lower impact than the surrogate modeling strategy. Section III–4.5 shows that SABBa with the proposed refinements yields excellent performances.

## III–4.2 POP computation

**Remark** *Throughout Sections III–2 and III–3, the Pareto Optimal Probability introduced in II–2.2 and $POP_{min}$ sequential refinement strategy proposed in II–3 have not been used. We give here the associated formulations for applying this technique in the Bounding-Box context.*

A set of boxes $\mathcal{B} = \left\{\mathcal{B}_i\right\}_i$, with $\mathcal{B}_i = \mathcal{B}\left(\boldsymbol{a}_i, \boldsymbol{r}_i\right)$, corresponds to a set of uniform densities that aims to approximate the true robustness and reliability measures. Every realization $\mathbf{P}_i \sim \mathcal{U}\left(\mathcal{B}_i\right)$ is of dimension $m = m_1 + m_2$ where $m_1$ is the number of objectives, denoted $\mathbf{P}_{i_f}$, and $m_2$ the number of constraints, denoted $\mathbf{P}_{i_g}$.

The probability in Equation (II.20) and (II.24) is computed as follows, using the independence assumptions between boxes and between dimensions:

$$\begin{aligned}
\mathbb{P}_{\mathbf{P}_j,\mathbf{P}_i}\left[\mathbf{P}_j \nprec_c \mathbf{P}_i\right] &= 1 - \mathbb{P}_{\mathbf{P}_j,\mathbf{P}_i}\left[\mathbf{P}_j \prec_c \mathbf{P}_i\right] \\
&= 1 - \Big(\mathbb{P}_{\mathbf{P}_{i_g}}\left[\mathbf{P}_{i_g} \not\leq \mathbf{0}\right] \\
&\quad + \mathbb{P}_{\mathbf{P}_{i_g}}\left[\mathbf{P}_{i_g} \leq \mathbf{0}\right]\mathbb{P}_{\mathbf{P}_{j_g}}\left[\mathbf{P}_{j_g} \leq \mathbf{0}\right]\mathbb{P}_{\mathbf{P}_{j_f},\mathbf{P}_{i_f}}\left[\mathbf{P}_{j_f} \succ \mathbf{P}_{i_f}\right]\Big) \\
&= \mathbb{P}_{\mathbf{P}_{i_g}}\left[\mathbf{P}_{i_g} \leq \mathbf{0}\right]\Big(1 - \mathbb{P}_{\mathbf{P}_{j_g}}\left[\mathbf{P}_{j_g} \leq \mathbf{0}\right]\mathbb{P}_{\mathbf{P}_{j_f},\mathbf{P}_{i_f}}\left[\mathbf{P}_{j_f} \succ \mathbf{P}_{i_f}\right]\Big)
\end{aligned} \tag{III.46}$$

where

$$\mathbb{P}_{\mathbf{P}_{i_g}}\left[\mathbf{P}_{i_g} \leq \mathbf{0}\right] = \prod_{k=m_1+1}^{m} \max\left(0, \min\left(1, \frac{-\mathcal{B}_{i_k}^-}{2r_{i_k}}\right)\right) = \prod_{k=m_1+1}^{m} \left[\frac{-\mathcal{B}_{i_k}^-}{2r_{i_k}}\right]_0^1,$$

and $\left[\,\cdot\,\right]_0^1$ means that values are taken between 0 and 1. The lower bound of the box is written as $\mathcal{B}_i^- = \boldsymbol{a}_i - \boldsymbol{r}_i$. The second probability involved in the last line of Equation (III.46) is computed as follows:

$$\begin{aligned}\mathbb{P}_{\mathbf{P}_{j_f},\mathbf{P}_{i_f}}\left[\mathbf{P}_{j_f} \succ \mathbf{P}_{i_f}\right] &= \prod_{k=1}^{m_1} \mathbb{P}_{\mathrm{P}_{j_k},\mathrm{P}_{i_k}}\left[\mathrm{P}_{j_k} \leq \mathrm{P}_{i_k}\right] \\ &= \prod_{k=1}^{m_1} \left(\left[\frac{L_{1_k}}{2r_{j_k}}\right]_0^1 + \left[\frac{L_{2_k}}{2r_{j_k}}\right]_0^1 \left(\frac{1}{2}\left[\frac{L_{2_k}}{2r_{i_k}}\right]_0^1 + \left[\frac{L_{3_k}}{2r_{i_k}}\right]_0^1\right)\right),\end{aligned}$$

where

$$\begin{aligned}L_{1_k} &= \mathcal{B}_{i_k}^- - \mathcal{B}_{j_k}^-, \\ L_{2_k} &= \min(\mathcal{B}_{i_k}^+, \mathcal{B}_{j_k}^+) - \max(\mathcal{B}_{i_k}^-, \mathcal{B}_{j_k}^-), \\ L_{3_k} &= \mathcal{B}_{i_k}^+ - \mathcal{B}_{j_k}^+.\end{aligned}$$

In the above, $\mathcal{B}_i^+ = \boldsymbol{a}_i + \boldsymbol{r}_i$ is the upper bound of the box.

In each dimension, $L_1$ can be interpreted as the portion of $\mathcal{B}_j$ dominating $\mathcal{B}_i$, $L_3$ the portion of $\mathcal{B}_i$ dominated by $\mathcal{B}_j$ and $L_2$ the overlapping area. We depict these lengths in Fig. III.22. Note that these values can be negative and that the computation is not symmetric between $\mathcal{B}_j$ and $\mathcal{B}_i$.

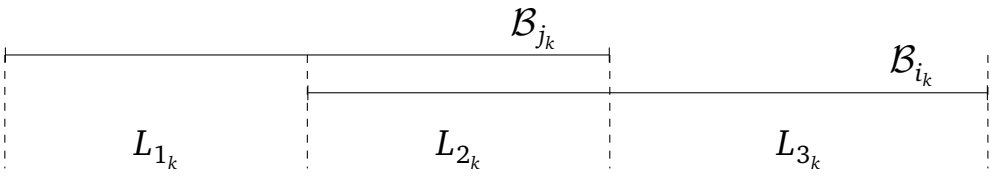



Figure III.22 Computational details

## III–4.3 Quality indicator

Assessing and comparing the performance of the proposed method requires the computation of a quantitative quality indicator. To this extent, the Hausdorff distance $d_H$ is a classical choice for computing the closeness of the found optimal set to the true one. This distance takes the general form:

$$d_H(\mathcal{A}, \mathcal{B}) = f\left(d_1(\mathcal{A}, \mathcal{B}), d_1(\mathcal{B}, \mathcal{A})\right),$$

where $f$ is usually chosen to be the max function $f(u, v) = \max(u, v)$. In the classical Hausdorff distance, distance $d_1$ is the maximal distance from a point of the first set to the second set:

$$d_1(\mathcal{A}, \mathcal{B}) = \max_{a \in \mathcal{A}}[d_2(a, \mathcal{B})],$$

and $d_2$ is the point to set distance, which is the minimal distance defined hereafter:

$$d_2(a, \mathcal{B}) = \min_{b \in \mathcal{B}}[d(a, b)].$$

In this last formula, $d$ refers to the classical Euclidean distance $d(a,b) = \|a-b\|_2$.

However, because $d_1$ takes the maximal distance from $\mathcal{A}$ to $\mathcal{B}$, it may not capture the overall similarities if one element is away from the other set. In practice, the modified Hausdorff distance proposed in [Dubuisson and Jain, 1994] returns much more intuitive values. This modified Hausdorff distance $d'_H$ replaces $d_1$ with the following average distance $d'_1$:

$$d'_1(\mathcal{A},\mathcal{B}) = \frac{1}{N}\sum_{a\in\mathcal{A}} d_2(a,\mathcal{B}).$$

A good quality indicator for an approximation $\widetilde{\mathcal{X}}^{\rho}_{\mathcal{P}}$ of the Pareto-optimal set is then:

$$Q = d'_H\big(\mathcal{X}^{\rho}_{\mathcal{P}}, \widetilde{\mathcal{X}}^{\rho}_{\mathcal{P}}\big).$$

With Bounding-Boxes, there is no Pareto Optimal set but rather a set of non-dominated designs, each of them having a Pareto Optimal Probability. The approximated Pareto front $\widetilde{\mathcal{P}}$ and Pareto optima $\widetilde{\mathcal{X}}_{\mathcal{P}}$ are hence aleatory, and realizations can be drawn straightforwardly. A realization $\widetilde{\mathcal{P}}_k$ from $\widetilde{\mathcal{P}}$ built from a set of boxes $\{\mathcal{B}_i\}_i$ corresponds to drawing a realization $\{\boldsymbol{\rho}_i\}_i$ of all boxes and to compute the associated Pareto front $\mathcal{P}\big(\{\boldsymbol{\rho}_i\}_i\big)$. The realization $\widetilde{\mathcal{X}}_{\mathcal{P}k}$ is simply the preimage.

In light of this, we propose to compute the expected value of the modified Hausdorff distance with respect to the realizations of $\widetilde{\mathcal{X}}_{\mathcal{P}}$:

$$Q_{\mathcal{B}} = \mathbb{E}_{\widetilde{\mathcal{X}}_{\mathcal{P}}}\big[d'_H(\mathcal{X}_{\mathcal{P}}, \widetilde{\mathcal{X}}_{\mathcal{P}})\big] \tag{III.47}$$

**Remark** *Recall that under the box independence and uniformity assumptions, as in Section III–4.2, we have* $\forall \boldsymbol{x}$*,*

$$\mathbf{P}(\boldsymbol{x}) \sim \mathcal{U}\Big(\mathcal{B}\big(\boldsymbol{\rho}_{SABBa}(\boldsymbol{x}), \overline{\boldsymbol{\varepsilon}}_{SABBa}(\boldsymbol{x})\big)\Big). \tag{III.48}$$

Intuitively, $Q_{\mathcal{B}}$ can be seen as an averaged distance between the real Pareto-optimal area and the preimage of a Pareto front realization. In a normalized input space such as the $[0,1]^2$ square, a value of $Q_{\mathcal{B}} = 1$ corresponds to a terrible score, with approximated and real Pareto optima on opposite sides of the input space. On the contrary, a score of $Q_{\mathcal{B}} = 10^{-2}$ reveals an excellent agreement between the approximated and the real optimal area.

## III–4.4 Uncertainty-based SABBa algorithm

In this section, the particular SABBa algorithm for performing Optimization Under Uncertainty (OUU) is presented. It is based on the general structure introduced in Algorithm III.1 and incorporates the developments from Sections III–4.1 to III–4.3.

Following the strategy proposed in Section II–3, SABBa will loop on decreasing values for the thresholds $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$. This approach should allow to (i) get low-cost coarse Pareto front approximations and (ii) slightly lower the overall computational cost by avoiding expensive refinements during the first iterations. Additionally, we also make use of the $POP_{min}$ refinement strategy, defined in Eqs. (II.26) and (II.27), with formulas introduced in Section III–4.2.

Then, computations and refinements of the robustness and reliability measures $\mathbf{P}(\boldsymbol{x}) \equiv \mathcal{B}_{\tilde{\rho}}\big(\boldsymbol{x}, \overline{\boldsymbol{\varepsilon}}(\boldsymbol{x})\big)$ are performed using formulations proposed in Section III–4.1.1. As for the Surrogate-Assisting model, it is built and updated as proposed in Section III–4.1.2.

**Remark** *Using this strategy, if several designs cannot be approximated with the Surrogate-Assisting model but all lie roughly in the same area, only one will get computed, and the updated SA model will be able to approximate the rest.*

The specific SABBa flowchart for solving Optimization Under Uncertainty (OUU) problems is given in Figure III.23, basing on Figure III.3. The complete algorithm, with more details on the refinement scheme and the differences between the use of Coupled Space and Separated Spaces, is then presented in Algorithm III.2.

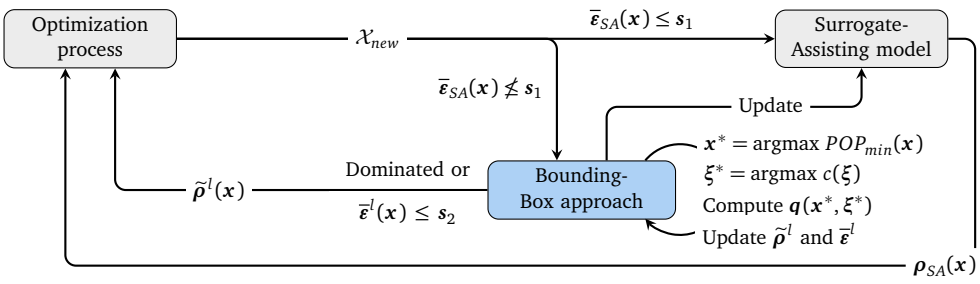



Figure III.23 SABBa flowchart for OUU problems.

Note that a SA approximation is computed for each $x \in \mathcal{X}_{new}$ at each new iteration. The associated Pareto Optimal Probability is computed, and the sets $\mathcal{X}_{SA}^k$ and $\mathcal{X}_c^k$ are determined. The set $\mathcal{X}_r^k$ of designs to refine is then computed and $x^*$ from Eq. (II.27) receives one refinement and is added to $\mathcal{X}_c$. All SA approximations for $x \in \mathcal{X}_{new} \setminus \mathcal{X}_c$ are updated and the operation is repeated.

**Algorithm III.2** General algorithm of SABBa

1: Loop over values of $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$
2: **while** Optimization running **do**
3: Get new designs $\mathcal{X}_{new}$
4: Initialize $\mathcal{X}_c$ empty
5: Compute all $\left(\boldsymbol{\rho}_{SA}(\boldsymbol{x}), \overline{\boldsymbol{\varepsilon}}_{SA}(\boldsymbol{x})\right)$ for $\boldsymbol{x} \in \mathcal{X}_{new}$ from (III.45)
6: Get $\mathcal{X}_c^0$ and $\mathcal{X}_{SA}^0$ from Definition 14
7: Deduce $\mathcal{X}_r^0$ from $\mathcal{X}_c^0$ using (II.26)
8: $k = 0$
9: **while** $\mathcal{X}_r^k \neq \emptyset$ **do**
10: Compute $POP_{min}(\boldsymbol{x})$ for all $\boldsymbol{x} \in \mathcal{X}_{new}$, with respect to all visited designs
11: Find $\boldsymbol{x}^*$ from $\mathcal{X}_r^k$ using (II.27)
12: **if** $\boldsymbol{x}^* \in \mathcal{X}_c$ **then**
13: Find $\boldsymbol{\xi}^*$ that maximizes the refinement criterion $c(\boldsymbol{\xi})$ defined in (III.42)
14: Launch evaluation of $\boldsymbol{q}(\boldsymbol{x}^*, \boldsymbol{\xi}^*)$
15: Update $\widehat{\boldsymbol{q}}_{\boldsymbol{x}}$ (Separated Space) or $\widehat{\boldsymbol{q}}$ (Coupled Spaces)
16: Compute $\widetilde{\boldsymbol{\rho}}^{l_k}$ and $\overline{\boldsymbol{\varepsilon}}^{l_k}$ at $\boldsymbol{x}_*$ (Separated Spaces) or at $\mathcal{X}_c$ (Coupled Space)
17: **else**
18: Add $\boldsymbol{x}^*$ in $\mathcal{X}_c$
19: Compute first approximation $\widetilde{\boldsymbol{\rho}}^0(\boldsymbol{x}^*)$ and $\overline{\boldsymbol{\varepsilon}}^0(\boldsymbol{x}^*)$
20: **end if**
21: Update $\left(\boldsymbol{\rho}_{SA}(\boldsymbol{x}), \overline{\boldsymbol{\varepsilon}}_{SA}(\boldsymbol{x})\right)$ for all $\boldsymbol{x} \in \mathcal{X}_{new} \setminus \mathcal{X}_c$
22: Get $\mathcal{X}_c^{k+1}$, $\mathcal{X}_{SA}^{k+1}$ and $\mathcal{X}_r^{k+1}$
23: $k = k + 1$
24: **end while**
25: Return $\boldsymbol{\rho}_{SABBa}(\boldsymbol{x})$ to the optimizer for all $\boldsymbol{x} \in \mathcal{X}_{new}$
26: **end while**
27: Return $\mathcal{X}_{\mathcal{P}_{\mathcal{B}}}^{\boldsymbol{\rho}_{SABBa}}\left(\mathcal{X}_{all}, \overline{\boldsymbol{\varepsilon}}_{SABBa}(\mathcal{X}_{all})\right)$ with $\mathcal{X}_{all}$ the set of all visited designs

SABBa can be coupled to any optimization algorithm and surrogate model for the SA strategy. This feature allows the approach to be very readily applicable and to benefit from new optimization or metamodeling techniques. In this chapter, the chosen optimization algorithm is NOMAD [Le Digabel, 2011] for its reliable management of multiple objectives and constraints within a derivative-free framework (version 3.6.2). All surrogate models are handled with the python Gaussian Process package GPy [GPy, 2012] (version 1.8.5).

**Remark** *In Algorithm III.2, the measure computation depends whether one makes a first approximation (line 17) or refines an existing one (line 12). Refinement is performed similarly when using separated spaces or coupled space surrogate model. However, coupled space surrogate model allows to get a first approximation* $\left(\widetilde{\boldsymbol{\rho}}^0(\boldsymbol{x}^*), \overline{\boldsymbol{\varepsilon}}^0(\boldsymbol{x}^*)\right)$ *for free, based only on the surrounding evaluations. On the contrary, a separated space strategy implies building a new surrogate from scratch at each new design, and hence performing some initial evaluations.*

**Remark** *Note also that similarly to Algorithm II.1, each time the thresholds* $\boldsymbol{s}_1$ *and* $\boldsymbol{s}_2$ *are lowered, the non-dominated computed boxes from the previous iterations must be refined before performing the optimization process. We simply propose to include these designs in the first studied set* $\mathcal{X}_{new}$.

Finally, one can see that SABBa relies on a quite small set of parameters, namely:

- The predefined sequence of pair of thresholds $(\boldsymbol{s}_1, \boldsymbol{s}_2)$;
- The number of function evaluations $N_{first}$ for the first approximations $\widetilde{\boldsymbol{\rho}}^0$ when using separated spaces surrogate models, and the number of evaluations $N_{ref}$ performed for each refinement;
- The number of new designs $N_{new}$ at each optimization iteration and $N_{init}$ at the first iteration. The optimizer may impose these parameters.



**Discussion** *This section provides all necessary numerical insights for applying SABBa to the problem of Optimization Under Uncertainty. We propose to compare the performance of the proposed algorithm against a direct nested approach and the A Priori MetaModel approach. Additionally, the next section quantitatively shows the relevance of proposed techniques by comparing several variants of SABBa.*

## III–4.5 Numerical tests

We apply here SABBa to two analytical test-cases. The first one is low-dimensional and deals with a Taguchi multi-objective robustness formulation (mean optimization and variance minimization). The second test-case features a higher number of dimensions and consists of a mean optimization under quantile constraint.

We systematically compare the performance of the proposed method to an A Priori MetaModel (APMM) strategy, where a surrogate model of the output is built in the coupled space (CS) before conducting the optimization process. The double loop approach mentioned in Section III–2.2 that performs a full Uncertainty Propagation at each optimization iteration can then be undertaken on the surrogate at a very low cost. The overall computational cost of building the surrogate model is chosen *a priori* based on the available budget, with a substantial impact on the accuracy of the results.

We show in the first test-case the performance of the direct Double Loop approach with respect to the APMM strategy. The former performs $N_{UP}$ evaluations for each design (with a total number of $N_{opt}$ designs) to compute the objective and constraint functions, which is shown to be extremely inefficient.

We compare here three variants of SABBa, denoted as follows:

- SA-SS: when using the Surrogate-Assisting strategy and separated-spaces models for constructing $\widetilde{\boldsymbol{q}}$ (see Section III–4.1.1);
- SA-CS: when using the Surrogate-Assisting strategy and a coupled-space model for constructing $\widetilde{\boldsymbol{q}}$ (see Section III–4.1.1).
- CS: when using the Coupled-Space model without the Surrogate-Assisting strategy;

Among the above variants, one can expect SA-CS to give better results than SA-SS and CS, taking advantage of both the low-dimensional Surrogate-Assisting model and the coupled space correlation to speed up box refinement.

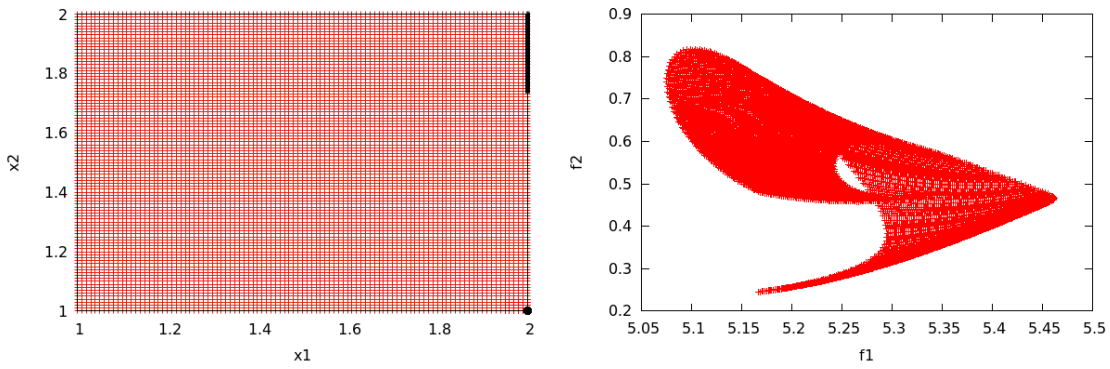
**Remark** *Ten runs are performed for each strategy to capture the mean convergence curve and the associated variability, represented as a translucent band around the mean. The computed distance between the approximated Pareto front and the true one is assumed to follow a log-normal distribution over the repeated runs. Note that high variability of the convergence curves reveals a lack of reliability of the associated approaches.*

### III–4.5.1 Test-case 1: Unconstrained Taguchi optimization

This problem is a bi-objective robust optimization proposed in [Fusi and Congedo, 2016]. There are two design variables $\boldsymbol{x}$ and one uncertain parameter $\xi$. The problem reads:

$$\begin{aligned} \text{minimize:} \quad & \boldsymbol{\rho}_f(\boldsymbol{x}) = \begin{pmatrix} \mu(\boldsymbol{x}) \\ \sigma^2(\boldsymbol{x}) \end{pmatrix} \\ \text{where:} \quad & \mu(\boldsymbol{x}) = \mathbb{E}_\xi[q(\boldsymbol{x},\xi)] \\ & \sigma^2(\boldsymbol{x}) = \mathbb{V}_\xi[q(\boldsymbol{x},\xi)] \\ \text{with:} \quad & q(\boldsymbol{x},\xi) = \xi - x_1\xi^5 + \cos(2\pi x_2 \xi) + 5 \\ & \xi \sim \mathcal{U}([0,1]) \\ \text{by changing:} \quad & (x_1,x_2) \in [1,2]^2 \end{aligned} \tag{III.49}$$

The Pareto front associated with this problem is discontinuous, and the optimal set in the design space is the union of a segment and a point (see Fig. III.24).




(a) (b)

Figure III.24 Test-case 1: a) Discretization of the design space in red and Pareto optimal sets in black. b) Image of the discretized points in the objective ($\mu$,$\sigma^2$) space.

The Double Loop and A Priori MetaModel strategies are quantitatively compared in Figure III.25. Each Uncertainty Propagation within the Double Loop is performed up to a relative accuracy of 5%. As expected, the Double Loop approach yields a very slow convergence toward the true Pareto front.

For this first test-case, we illustrate SABBa performance both when the surrogate can represent the underlying functions accurately and when it shows poor accuracy. The accuracy of the surrogate strongly influences the APMM $\widehat{\boldsymbol{q}}$, the Surrogate-Assisting model $\boldsymbol{\rho}_{SA}(\boldsymbol{x})$ and the separated or coupled-space models $\widehat{\boldsymbol{q}}(\boldsymbol{x},\cdot)$.

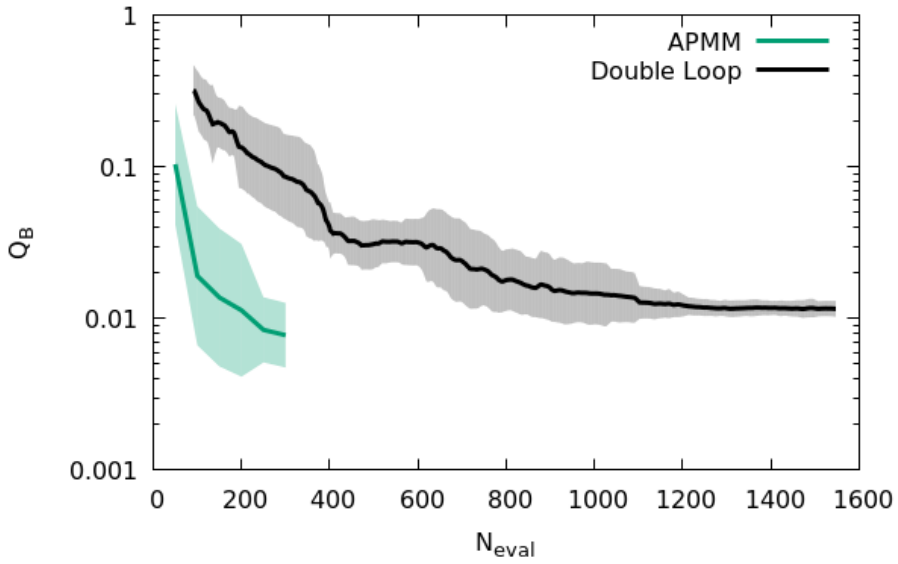



Figure III.25 Test-case 1: Cost comparison between the A Priori MetaModel and Double loop approach.

**With high-quality surrogate model** The unconstrained Taguchi optimization problem is first solved using a high-quality surrogate approach. Practically, we build an anisotropic GP surrogate model with a classical RBF kernel (also called squared-exponential, exponential quadratic or Gaussian), and Automatic Relevance Determination (ARD):

$$k(\boldsymbol{x},\boldsymbol{x}') = \sigma^2 \exp\left(-\frac{r_l^2}{2}\right),$$
$$\text{with } r_l = \sqrt{\sum_{i=1}^{m}\left(\frac{x_i - x_i'}{l_i}\right)^2}.$$

This model requires $m+1$ hyperparameters $\{\sigma^2, l_1, \ldots, l_m\}$ to be optimized but captures the characteristic lengthscale associated with each input dimension.

Note that here, the thresholds $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ are sequentially refined five times. As proposed in (III.35), to alleviate the tuning of these thresholds, only normalized thresholds $\bar{\boldsymbol{s}}_1$ and $\bar{\boldsymbol{s}}_2$ must be chosen beforehand. Note that the range $h_i$ covered in the $i^{th}$-dimension is updated at each iteration.

Here, $\bar{\boldsymbol{s}}_1$ and $\bar{\boldsymbol{s}}_2$ are both sequentially taken as 50%, 40%, 30%, 20%, 10% and 5% in all dimensions. The chosen parameters of SABBa are as follows: $N_{init} = 10$, $N_{new} = 1$ (sequential optimizer), $N_{first} = 5$ and $N_{ref} = 1$ (sequential refinement).

Figure III.26 pictures the convergence curves of the APMM strategy and the three studied SABBa variants (SA-CS, SA-SS and CS) in terms of the indicator defined in (III.47). The mean curves are mostly comparable, with a slight advantage for SA-CS. However, the associated variability is much smaller, with any SABBa variant compared to the APMM strategy. As raised earlier, the indicator is assumed log-normal. The high variance associated with the APMM strategy coupled with the heavy tail of the log-normal distribution makes the approach very unreliable. With this perspective, the worst results out of ten repeated runs of SA-CS and the APMM strategy are plotted in Figure III.27. These outputs correspond to $N_{eval} \approx 100$.

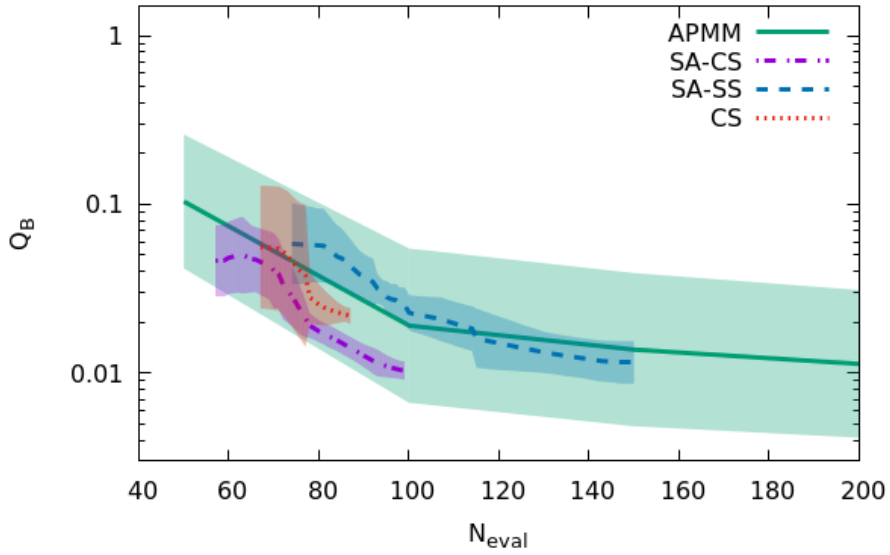
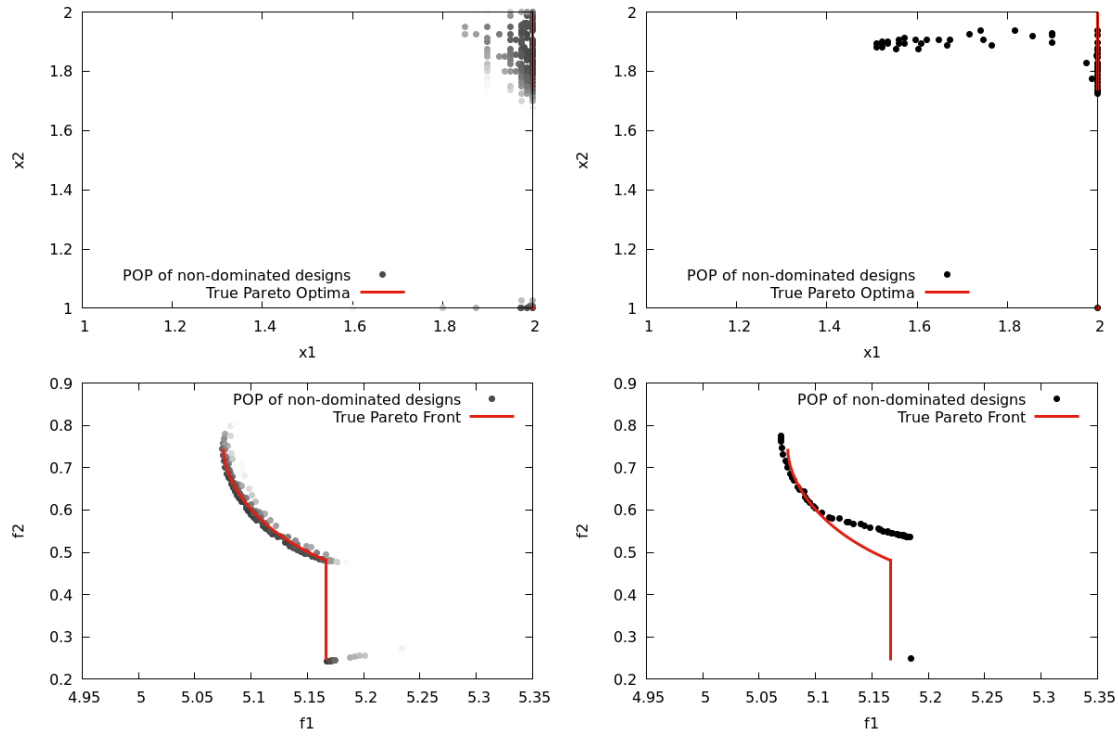



Figure III.26 Test-case 1-a: Cost comparison between APMM and three SABBa variants.

In these figures, optimal designs returned by SABBa are plotted in greyscale, which refers to the Pareto Optimal Probability (POP) of each design, ranging continuously from 0 (white) to 1 (black). Because the APMM strategy does not construct boxes, POP is either equal to 0 (dominated individual) or 1 (Pareto-optimal individual).




(a) (b)

Figure III.27 Worst optimization results: (a) SABBa SA-CS, 93 evaluations , $Q_{\mathcal{B}} = 1.29 \times 10^{-2}$. (b) APMM, 100 evaluations , $Q_{\mathcal{B}} = 1.91 \times 10^{-1}$.

In line with Figure III.26, Figure III.27 reveals that the APMM strategy may return very inaccurate results when the DoE that was used for constructing the coupled-space model is not well sampled in the optimal area. On the contrary, the adaptive surrogate model construction permits SABBa to return very stable results.

**With a low-quality surrogate model** In the following, the same optimization problem is solved using a low-quality surrogate model both within SABBa and the APMM strategy. Contrarily to the previous formulation, we build a isotropic GP surrogate model with only one lengthscale, that has to account for all dimensions. Practically,

$$k(\boldsymbol{x},\boldsymbol{x}') = \sigma^2 \exp\left(-\frac{r_l^2}{2}\right),$$
$$\text{with } r_l = \frac{1}{l}\sqrt{\sum_{i=1}^{m}\left(x_i - x_i'\right)^2}.$$

Only two hyperparameters $\{\sigma^2, l\}$ must be optimized. However, this model will notably fail when the characteristic lengths of the function in the different dimensions are very different. This test-case aims at simulating problems where the surrogate model is challenging to build. This problem can notably happen when the number of dimensions is significant.

In this context, the SA strategy should allow yielding a significant cost improvement through direct surrogate modeling of the statistical measures in the design space. Such improvement is indeed revealed in Figure III.28, where two behaviors can be found: (i) non-Surrogate-Assisted approaches (APMM and SABBa CS) show poor performance compared to the other strategies and (ii) the convergence curve of SA-CS and SA-SS are much more similar than in Figure III.26. Indeed, here, they both heavily rely on the low-dimensional SA model to speed-up the convergence.

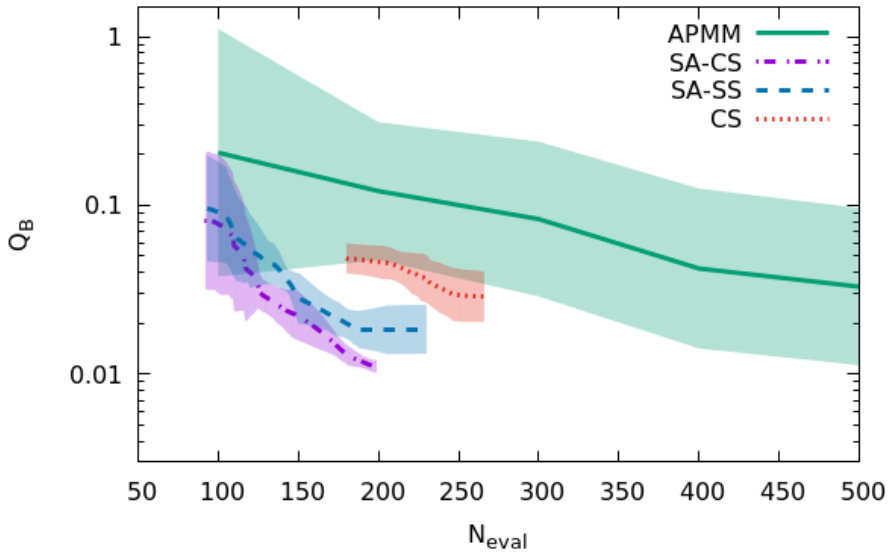
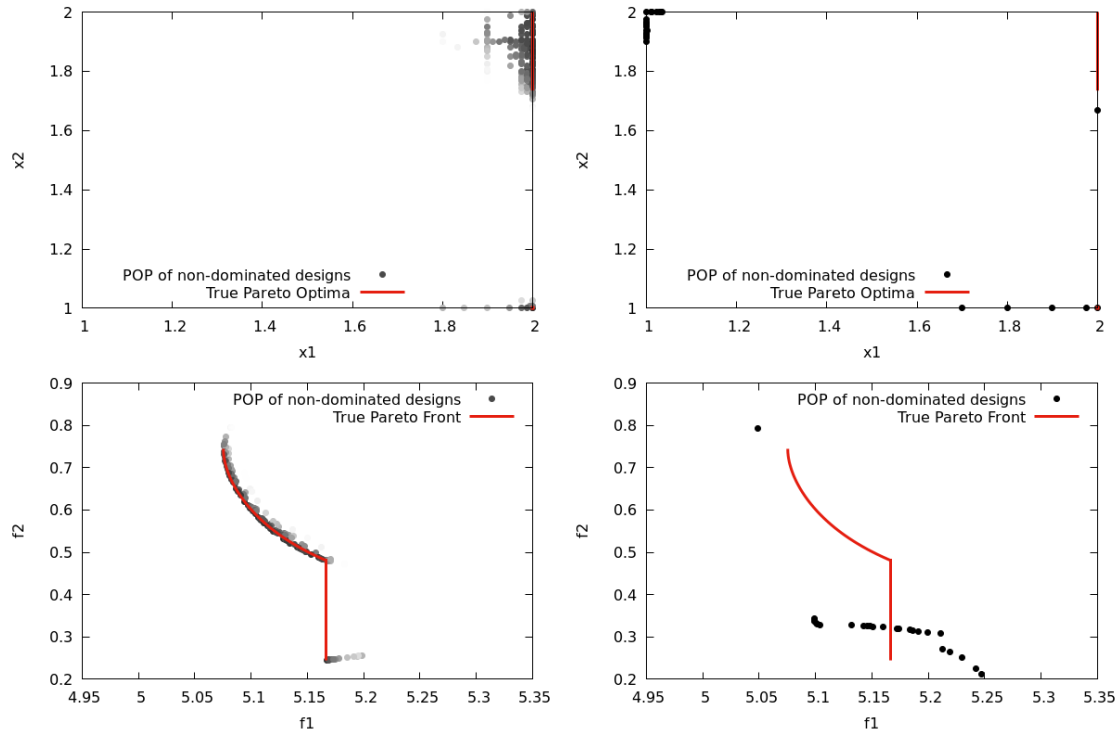



Figure III.28 Test-case 1-b: Cost comparison between APMM and three SABBa variants.

As previously, the worst optimization outputs are plotted for SABBa SA-CS and the APMM approach. As expected, in Figure III.29(a), SABBa shows high consistency and accuracy for capturing the Pareto-optimal area. On the contrary, the APMM result plotted in Figure III.29(b) with 200 evaluations is completely wrong, both in the design space and in the objective space.




(a) (b)

Figure III.29 Worst optimization results: (a) SABBa SA-CS, 196 evaluations , $Q_{\mathcal{B}} = 1.23 \times 10^{-2}$. (b) APMM, 200 evaluations , $Q_{\mathcal{B}} = 7.94 \times 10^{-1}$.

The use of coupled space surrogate models and Surrogate-Assisting strategy has shown to bring a significant cost reduction for SABBa SA-CS. It performs better on average than the APMM strategy and shows far greater consistency and robustness.

### III–4.5.2 Test-case 2: Quantile-constrained mean optimization

We propose this second test-case to assess the performance of SABBa in a higher dimensional case and in the presence of a reliability-based constraint (derived from the Six-Hump Camel function). The objective is a robustness measure derived from a simplified Rosenbrock function. We consider here four design variables and three

uncertain parameters. The problem is stated as follows:

$$
\begin{aligned}
\text{minimize:}\quad & \boldsymbol{\rho}_f(\boldsymbol{x}) = \mu(\boldsymbol{x}) \\
\text{satisfying:}\quad & \boldsymbol{\rho}_g(\boldsymbol{x}) = \mathrm{q}^{95\%}(\boldsymbol{x}) \leq 1 \\
\text{where:}\quad & \mu(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{\xi}}[q_1(\boldsymbol{x},\boldsymbol{\xi})] \\
& \mathrm{q}^{95\%}(\boldsymbol{x}) = \mathrm{q}^{95\%}_{\boldsymbol{\xi}}[q_2(\boldsymbol{x},\boldsymbol{\xi})] \\
\text{with:}\quad & q_1(\boldsymbol{x},\boldsymbol{\xi}) = \sum_{i=1}^{3}\left[(1-x_i) + 3\left(1 + \frac{\arctan\big(5(\xi_i - 0.5)\big)}{2}\right)\left(x_{i+1} - x_i^2\right)^2\right] \\
& q_2(\boldsymbol{x},\boldsymbol{\xi}) = \left(4 - 2.1x_1^2 + \frac{x_1^4}{3}\right)x_1^2 + x_1x_2 + \left(-4 + 4x_2^2\right)x_2^2 \\
& \qquad + \frac{\cos(2\pi\xi_1) - \sin(\frac{\pi}{2}\xi_1) - \xi_1 - \left(\cos(2\pi 0.05) - \sin(\frac{\pi}{2}0.05) - 0.05\right)}{5} \\
& \boldsymbol{\xi} \sim \mathcal{U}([0,1]^3) \\
\text{by changing:}\quad & \boldsymbol{x} \in [-0.2, 1.2]^4
\end{aligned}
\tag{III.50}
$$

One can note that the robustness measure reduces to the classical formulation of the 4D Rosenbrock function and the reliability measures to the Six-Hump Camel function. Analytically, it holds that:

$$
\begin{aligned}
\mu(\boldsymbol{x}) &= \sum_{i=1}^{3}\left[(1-x_i) + 3\left(x_{i+1} - x_i^2\right)^2\right], \\
\mathrm{q}^{95\%}(\boldsymbol{x}) &= \left(4 - 2.1x_1^2 + \frac{x_1^4}{3}\right)x_1^2 + x_1x_2 + \left(-4 + 4x_2^2\right)x_2^2.
\end{aligned}
\tag{III.51}
$$

The optimum of this deterministic problem is found at $\boldsymbol{x}_* \approx (0.7033, 0.7035, 0.6212, 0.3859)$ with $\mathrm{q}^{95\%}(\boldsymbol{x}_*) = 1$ and $\mu(\boldsymbol{x}_*) \approx 0.4981$.

For this test case, both $\bar{\boldsymbol{s}}_1$ and $\bar{\boldsymbol{s}}_2$ are sequentially taken as 50%, 40%, 30%, 20%, 10%, 5%, 3%, 2%, 1% and 0.5%. As for the other parameters, $N_{init} = 10$, $N_{new} = 1$ (sequential optimiser), $N_{first} = 5$ and $N_{ref} = 1$ (sequential refinement).

Figure III.30 shows a slight mean improvement when using SA-CS compared to the APMM strategy, roughly by a factor 2. The variability of the output is also halved with all SABBa variants, which is critical for real-world applications.

**Remark** *Note that the plateau that is reached by the SABBa curves is partially a plotting artifact. All runs have a different final number of evaluations, and the indicator $Q_{\mathcal{B}}$ is considered constant when the optimum is reached. Therefore, the average of these curves tends to flatten at the end.*

Again, we plot the worst optimization outputs in Figures III.31. They are depicted in parallel coordinates plots, with a $POP_{\text{min}}$ greyscale. A grey curve represents each efficient individual, and the exact optimum is depicted in red. This comparison reveals the better performance of SABBa again with respect to the APMM strategy.

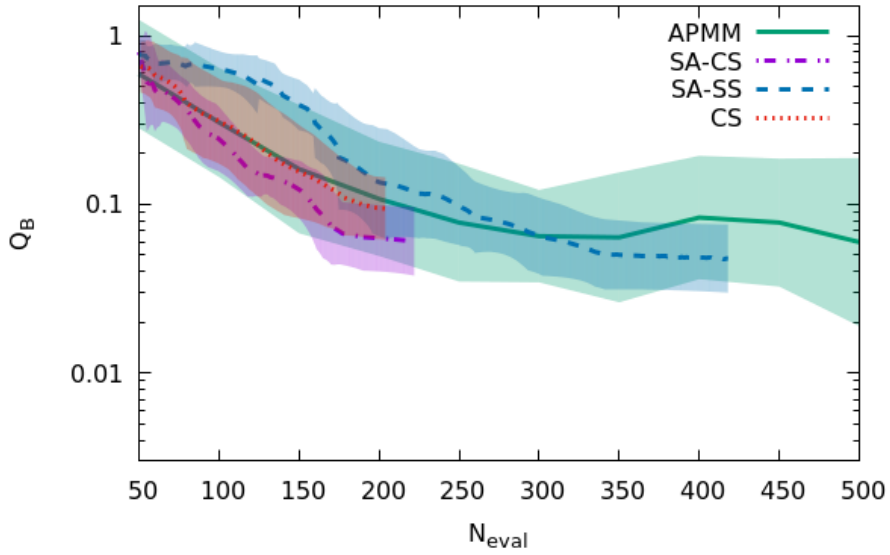



Figure III.30 Test-case 2: Cost comparison between APMM and three SABBa variants.

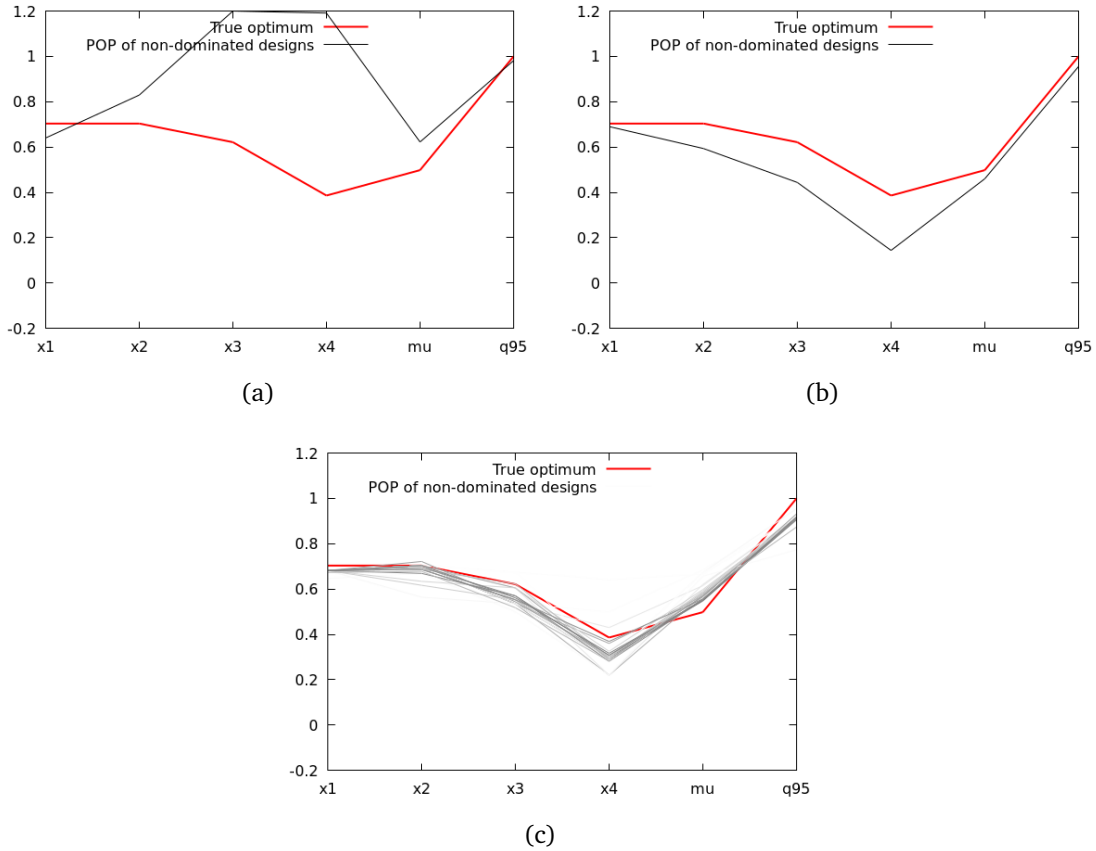



Figure III.31 Worst optimization results: (a) APMM, 150 evaluations, $Q_{\mathcal{B}} = 1.0 \times 10^{0}$. (b) SABBa SA-CS, 161 evaluations, $Q_{\mathcal{B}} = 1.12 \times 10^{-1}$. (c) APMM, 200 evaluations, $Q_{\mathcal{B}} = 3.24 \times 10^{-1}$.

**Discussion** *This section has provided all necessary numerical insights for applying SABBa to the problem of Optimization Under Uncertainty. The proposed strategies for computing and refining the approximations within the Bounding-Box approach aim at increasing the overall efficiency and lowering the number of function evaluations.*
*Analytical test-cases against a Double-Loop (DL) optimization strategy and an A Priori MetaModel (APMM) approach have shown that SABBa yields better convergence rate and robustness. This improvement is even more significant when the surrogate model is not able to predict $\boldsymbol{q}$ accurately. Thanks to the low-dimensional SA model and the adaptive strategy of the Bounding-Box approach that performs local refinements on the Pareto front, SABBa gains a factor 2 to 100 on the convergence toward the Pareto front. Compared to the APMM strategy, another advantage of SABBa is that the choice of the number of evaluations of $\boldsymbol{q}$ needs not be determined a priori, but is determined online, when the output is considered accurate enough.*
*Overall, these analyses validate the proposed strategy. The application of SABBa to uncertainty-based engineering problems is presented in Chapter V.*

**Conclusion of the chapter** *In summary, this chapter proposes a specific formulation of the SAMATA algorithm from Section II–3, characterized by independent uniform measure approximations. The notations are properly adapted to these so-called Bounding-Boxes [Fusi and Congedo, 2016]. Thorough theoretical analysis about the convergence of the Bounding-Box approach and its coupling with the Surrogate-Assisting strategy is performed.*
*A proof-of-concept of this approach has been proposed on a noisy optimization problem, to analyze the asymptotic effect of this strategy within a controlled artificial test-case. The results are in line with the expectations and provide some insights on the behavior of SABBa.*
*Efficient GP-based formulations have then been introduced for applying SABBa to optimization under uncertainty problems.*
*Finally, this algorithm has been applied to two analytical test-cases to analyze the performance of SABBa quantitatively. Three variants are proposed and compared to the A Priori MetaModel strategy, showing both quicker convergence and much higher robustness.*

CHAPTER *IV*

# Sampling-based measure approximations

**Overview** *In this section, we aim to avoid the restriction to independent uniform distributions proposed in SABBa. For example, the error could be approximated with Gaussian distributions, which often rely on the so-called projected process technique [Da Veiga and Delbos, 2013, Baudoui, 2012, Janusevskis and Le Riche, 2013, Williams et al., 2000]. Here, we propose a non-parametric approach that relies on sampling-based distribution approximations. This approach should allow for more representativeness of the error estimation given the current information, and thus for quicker convergence. Two original approaches are introduced in the following for constructing these sampling-based measure approximations. The first one relies on the construction of a coupled space surrogate model for drawing joint realizations of* $\mathbf{P}$ *over the whole design space. The second one exploits disjoint realizations, that can be drawn without the construction of a coupled-space surrogate model.*

**Outline** *Section IV–1 introduces a sampling-based setting which can be applied within the SAMATA strategy. Dominance criteria and POP computations are formulated in this setting.*
*Sections IV–2 and IV–3 then present two different techniques for computing such sampling-based approximations. To this extent, an inducing point strategy and a KDE-based surrogate model are proposed.*
*These strategies are applied to several analytical test-cases in Section IV–4 to assess their performance, notably against SABBa. The impact of the sampling-based approach against Gaussian assumptions is also pictured.*
*Finally, Section IV–5 is devoted to the application of the sampling-based SAMATA strategy to optimization problem featuring non-parametric uncertainties. The efficiency and representativeness of the proposed approach are illustrated in two analytical test-cases against more classical Distribution-Assuming techniques.*

**Contribution** *The sampling strategy proposed in Section IV–2 permits to control the accuracy of the representation through adaptive refinements, which is, to the author's knowledge, a novelty in the uncertainty-based optimization context. As for the final formulation, it is shown to be equivalent to a Nyström approximation on a Karhunen-Loève Expansion (KLE) representation of the GP random field in the coupled space.*
*The second proposed strategy is a novel technique for drawing realizations from a random field based on disjoint data drawn at some training location. It relies heavily on an underlying Gaussian Process surrogate model, that is generalized to non-parametric distribution shapes through a rejection sampling strategy.*
*Both proposed strategies permit to escape the context of Gaussian noises and incorporate more information, which is shown to perform slightly better than using a more conventional Gaussianity assumption.*
*Finally, the sampling-based SAMATA strategy performs very well on OUU problems featuring either parametric or non-parametric uncertainties.*

# IV–1 Sampling-based setting

SAMATA performs low-cost optimization under uncertainty by treating robustness and reliability measures as a random field $\mathbf{P}$ over the design space $\mathcal{X}$. In this section, the aim is to make as few assumptions as possible on the distribution of these random variables. To this extent, we assume that realizations $\mathbf{P}^{(i)}(\boldsymbol{x})$ can be drawn and exploit this empirical set of samples to characterize $\mathbf{P}(\boldsymbol{x})$.

Because $\mathbf{P}$ is essentially a random field over the design space $\mathcal{X}$, one can either draw realizations $\mathbf{P}^{(i)}$ of the whole random field or draw realizations $\mathbf{P}^{(i)}(\boldsymbol{x}_j)$ from distribution $\mathbf{P}(\boldsymbol{x}_j)$ at each location of interest, independently from the other ones. These two approaches are denoted as joint and disjoint realizations in the following, respectively. Specific insights for computing probability of dominance are given in the following paragraphs.

**Joint realizations** If realizations are drawn jointly over all designs $\boldsymbol{x}$, dominance must only be verified between joint samples. With a number $N_{samples}$ of realizations for each location, the probability of dominance between two individuals can be empirically estimated as follows:

$$\mathbb{P}_{\mathbf{P}}\Big[\mathbf{P}(\boldsymbol{x}_j) \not\succ_c \mathbf{P}(\boldsymbol{x}_k)\Big] \simeq \frac{1}{N_{samples}} \sum_{i=1}^{N_{samples}} \mathbb{1}_{\{\mathbf{P}^{(i)}(\boldsymbol{x}_j) \not\succ_c \mathbf{P}^{(i)}(\boldsymbol{x}_k)\}}, \tag{IV.1}$$

In order to avoid the computational time associated to POP computation to become too large, we propose a Monte Carlo approximation of the above estimation, with constant computational burden that can be chosen according to the available budget. More precisely, we estimate the probability by:

$$\mathbb{P}_{\mathbf{P}}\Big[\mathbf{P}(\boldsymbol{x}_j) \not\succ_c \mathbf{P}(\boldsymbol{x}_k)\Big] \simeq \frac{1}{N_{MC}} \sum_{i=1}^{N_{MC}} \mathbb{1}_{\{\mathbf{P}^{(I^{(i)})}(\boldsymbol{x}_j) \not\succ_c \mathbf{P}^{(I^{(i)})}(\boldsymbol{x}_k)\}}, \tag{IV.2}$$

where $I^{(i)}$ are realizations of the random index $I \sim \mathcal{U}(\llbracket 1, N_{samples} \rrbracket)$.

**Disjoint realizations** Here we consider that samples $\mathbf{P}^{(i)}(\boldsymbol{x})$ are drawn independently between the different $\boldsymbol{x}$. The dominance must be checked between all pairs $(i, j)$, as depicted in the following formula:

$$\mathbb{P}_{\mathbf{P}}\Big[\mathbf{P}(\boldsymbol{x}_j) \not\succ_c \mathbf{P}(\boldsymbol{x}_k)\Big] \simeq \frac{1}{N_{samples}^2} \sum_{i=1}^{N_{samples}} \sum_{i'=1}^{N_{samples}} \mathbb{1}_{\{\mathbf{P}^{(i)}(\boldsymbol{x}_j) \not\succ_c \mathbf{P}^{(i')}(\boldsymbol{x}_k)\}}. \tag{IV.3}$$

and, similarly to the joint case, a Monte Carlo approximation can be computed as follows:

$$\mathbb{P}_{\mathbf{P}}\Big[\mathbf{P}(\boldsymbol{x}_j) \not\succ_c \mathbf{P}(\boldsymbol{x}_k)\Big] \simeq \frac{1}{N_{MC}} \sum_{i=1}^{N_{MC}} \mathbb{1}_{\{\mathbf{P}^{(I_1^{(i)})}(\boldsymbol{x}_j) \not\succ_c \mathbf{P}^{(I_2^{(i)})}(\boldsymbol{x}_k)\}}, \tag{IV.4}$$

where we take realizations of $I_1$ and $I_2$, with $I_1, I_2 \sim \mathcal{U}(\llbracket 1, N_{samples} \rrbracket)$.

**Remark** *Several works have proposed techniques for computing probabilistic Pareto dominance between disjoint non-parametric density estimations [Coelho, 2014, Khosravi et al., 2018, Khosravi et al., 2019], notably with the aim to reduce the associated computational burden.*

The sampling-based setting requires some specific formulations to be integrated within SAMATA. Specifically, referring to the notations from Figure II.5, the following points should be tackled:

- how to construct the probabilistic approximation $\mathbf{P}(\boldsymbol{x})$, which is a set of realizations here;
- which metric $\widetilde{\boldsymbol{\rho}}$ and $\Delta$ are extracted from $\mathbf{P}(\boldsymbol{x})$;
- how to refines current approximations of the robustness and reliability measures;
- which technique is used for constructing the Surrogate-Assisting model.

The following sections are devoted to the illustration of these methodologies.

Notably, based on formulations (IV.1) to (IV.4), the $POP_{min}$ metric is computed as follows:

$$POP_{min}(\mathbf{P}(\boldsymbol{x}_k)) = \min_{\substack{\boldsymbol{x}_j \\ j\neq k}} \Big( \mathbb{P}_{\mathbf{P}}\big[\mathbf{P}(\boldsymbol{x}_j) \not\succ_c \mathbf{P}(\boldsymbol{x}_k)\big]\Big). \tag{IV.5}$$

**Remark** *Contrarily to Chapter III, the support of sampling-based measure approximations may not be finite. The Boxed Pareto dominance that corresponds to the non-relaxed probabilistic dominance introduced in Section II–2.1, is hence not adapted to discriminate promising individuals from less interesting ones. Thus, in the following of this chapter, a relaxed probabilistic dominance rule II.16 is adopted.*

**Discussion** *This short section aimed to introduce the sampling-based for multi-objective optimization problems. The proposed probabilistic dominance formulas can be readily incorporated within SAMATA to guide the refinements toward the most promising area.*
*In the next two sections, numerical procedures are proposed for generating realizations of the robustness and reliability measures. In Section IV–2, we assume that a surrogate model $\hat{\boldsymbol{q}}$ of $\boldsymbol{q}$ can be constructed as a function of both design parameters $\boldsymbol{x}$ and uncertain variables $\boldsymbol{\xi}$. On the contrary, Section IV–3 aims at drawing realizations $\mathbf{P}^{(i)}(\boldsymbol{x})$ at any design $\boldsymbol{x} \in \mathcal{X}$ without coupled space surrogate model, based on few independent sets of samples $\big\{\{\mathbf{P}^{(i)}(\boldsymbol{x}_j)\}_i\big\}_j$.*

# IV–2 Coupled-space realizations for joint samplings

## IV–2.1 Formulation and challenge

Here, we assume that a GP surrogate model can be built on $\boldsymbol{q}$ in the coupled design/uncertainty space, similarly to the *Coupled Space* approach from Section III–4.1.1. Based on training data $\big\{\boldsymbol{q}(\boldsymbol{x}_i, \boldsymbol{\xi}_i)\big\}_i$, the surrogate $\hat{\boldsymbol{q}}$ is constructed and yields an estimation of $\boldsymbol{q}$ at any location $(\boldsymbol{x}, \boldsymbol{\xi})$ in $\mathcal{X} \times \Xi$.

This coupled space surrogate model can be exploited to obtain an estimation of the robustness and reliability measure through Monte Carlo integration. This integration is directly performed on the predictive mean $\hat{\boldsymbol{q}}$ of the GP. For example, with the expectation measure $\boldsymbol{\rho}(\boldsymbol{x}) = \mathbb{E}_{\xi}[\boldsymbol{q}(\boldsymbol{x}, \xi)]$, one can use the estimator :

$$\widetilde{\boldsymbol{\rho}}(\boldsymbol{x}) = \frac{1}{N_{\xi}} \sum_{j=1}^{N_{\xi}} \hat{\boldsymbol{q}}(\boldsymbol{x}, \xi_j). \tag{IV.6}$$

More interestingly, one can draw realizations of the GP built on $\boldsymbol{q}$, which directly translates into realizations of $\mathbf{P}$ by integrating over the uncertain dimension. In short, this permits to generate samples from $\mathbf{P}$ without making any assumption on its distribution shape. The base assumption for constructing this representation is that $\boldsymbol{q}$ can be modeled as a stochastic process, chosen here to be a GP.

For example, in the case of the expectation measure $\boldsymbol{\rho}(\boldsymbol{x}) = \mathbb{E}_{\xi}[\boldsymbol{q}(\boldsymbol{x}, \xi)]$, by evaluating $\boldsymbol{q}$ at $N_{\xi}$ locations in the uncertain space for each of the $N_x$ designs, the joint realizations of $\mathbf{P}$ over all these designs write as follows:

$$\forall k \in [\![1, N_x]\!], \quad \mathbf{P}^{(i)}(\boldsymbol{x}_k) = \frac{1}{N_{\xi}} \sum_{j=1}^{N_{\xi}} \boldsymbol{q}^{(i)}(\boldsymbol{x}_k, \xi_j). \tag{IV.7}$$

The same can be carried out with empirical estimators of other robustness and reliability measures.

**Remark** *Note that $\boldsymbol{q}^{(i)}$ are fonctional realizations of $\boldsymbol{q}$ in the coupled space drawn from the Gaussian process $\hat{\boldsymbol{q}}$. These functional realizations give access to realizations $\mathbf{P}^{(i)}$ of $\mathbf{P}$ at different designs simultaneously.*

One can notice that the realization $\boldsymbol{q}^{(i)}$ is drawn at $N_x \times N_{\xi}$ locations, which can quickly reach a value of more than $10^5$. Looking back at Equation (I.34), it reveals that Cholesky decomposition should hence be performed on a matrix of size $10^5 \times 10^5$. The computational cost associated with such a task is prohibitive. This problem is quite common in the literature, and several approximations techniques have been proposed, such as a circulant embedding of the covariance matrix [Powell et al., 2014], regularization [Pourahmadi, 2011] or the use of a Gibbs sampler for Gaussian Markov Random Field [Lantuéjoul and Desassis, 2012]. However, some of these approaches require equispaced points, and the others remain too expensive to allow substantial sampling of the $\boldsymbol{q}$ random field.

As raised in [Panunzio et al., 2018], the complexity is particularly high when the domain is much larger than the characteristic correlation length of the random field. Intuitively, when a value is drawn, it spreads information in its surrounding at a distance proportional to the correlation length.

In the next section, a strategy is proposed for drawing realizations $\boldsymbol{q}^{(i)}$ of the Quantities of Interest, based on the GP assumption, at a vast number of locations simultaneously.

## IV–2.2 Inducing points strategy

### IV–2.2.1 General idea

Here, we propose to draw realizations $\boldsymbol{q}^{(i)}$ of the Gaussian random field on few well-spread points and from which a smooth functional realization can be constructed with zero computational cost on the whole input space. These well-chosen points will be referred to as inducing points.

In the following, for conciseness, coordinates $(\boldsymbol{x}, \boldsymbol{\xi})$ in the coupled space are denoted with $\boldsymbol{z}$. With a set of training designs $\boldsymbol{z}_{train} = (\boldsymbol{x}_{train}, \boldsymbol{\xi}_{train})$ and a set of inducing points $\boldsymbol{z}_{ind} = (\boldsymbol{x}_{ind}, \boldsymbol{\xi}_{ind})$, we denote $\boldsymbol{z}_{tot} = \boldsymbol{z}_{train} \cup \boldsymbol{z}_{ind}$.

We assume that a GP has been constructed on $\boldsymbol{z}_{train}$, as plotted in Figure IV.1, on values $\boldsymbol{q}_{train}$ and with an optimized kernel $k_{train}$. Its predictive mean and covariance are written $\boldsymbol{\mu}_{train}$ and $\Sigma_{train}$. This surrogate model is the current estimation of $\boldsymbol{q}$ over the whole coupled space.

We aim to generate random realizations of this Gaussian field on an extensive set of points $\boldsymbol{z}_{draw}$. Direct Gaussian sampling would perform Cholesky decomposition on the matrix $K_{draw} = \Sigma_{train}(\boldsymbol{z}_{draw})$ of size $N_{draw}^2$, which is shown in the previous section to be unfeasible quickly.

We propose to only draw realizations on the inducing points:

$$\boldsymbol{q}_{ind}^{(i)} = \boldsymbol{\mu}_{train}(\boldsymbol{z}_{ind}) + L_{ind}\boldsymbol{Y}^{(i)}, \quad \boldsymbol{Y} \sim \mathcal{N}(\boldsymbol{0}, I), \tag{IV.8}$$

with $L_{ind}$ the Cholesky decomposition of matrix $K_{ind} = \Sigma_{train}(\boldsymbol{z}_{ind})$ of much smaller size $N_{ind}^2$. Figure IV.2 gives an example of such drawn values at the inducing points. We then propose to estimate the realizations on $\boldsymbol{z}_{draw}$ as the following new GP predictive mean:

$$\boldsymbol{q}_{draw}^{(i)} = \boldsymbol{\mu}_{tot}^{(i)}(\boldsymbol{z}_{draw}) = \boldsymbol{k}_*^\intercal K_{tot}^{-1} \boldsymbol{q}_{tot}^{(i)}, \tag{IV.9}$$

where $\boldsymbol{k}_* = k_{train}(\boldsymbol{z}_{tot}, \boldsymbol{z}_{draw})$ and $K_{tot}^{-1}$ only need to be computed once as they do not depend on $i$, and where $\boldsymbol{q}_{tot}^{(i)}$ simply concatenates $\boldsymbol{q}_{train}$ and $\boldsymbol{q}_{ind}^{(i)}$. These realizations are represented in Figure IV.3, and the empirical 2-$\sigma$ band of the random field is computed on $10^5$ samples. One can note that they fit well the real confidence band of the underlying GP.

**Remark** *Intuitively, this approach corresponds to parametrizing the random field as a sum of kernels located at all training and inducing points. This is noticeable in the above Equation (IV.9). By denoting $K_{tot}^{-1}\boldsymbol{q}_{tot}^{(i)}$ as a vector $\boldsymbol{\beta}^{(i)}$ of coefficients, the proposed realizations $\boldsymbol{q}_{draw}^{(i)}$ are linear combinations of the the kernel function $k_{train}$ located at each training and inducing point. Coefficients $\boldsymbol{\beta}^{(i)}$ manage the randomness and kernel functions the spatial characteristic. The functional realizations write $k_{train}(\boldsymbol{z}_{tot}, \cdot)^\intercal \boldsymbol{\beta}^{(i)}$.*

### IV–2.2.2 Similarity with Nyström method for KLE

In practice, the proposed approach is equivalent to constructing a Karhunen-Loève Expansion (KLE) of a Gaussian random field, when computing the basis functions with Nyström approximation. Computational details are given hereafter to show the equivalence.

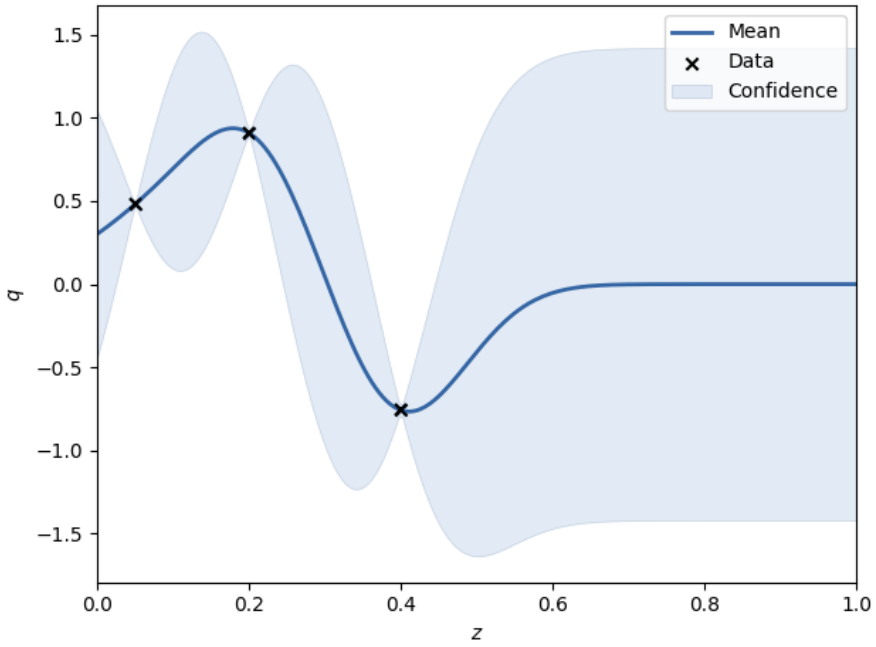



Figure IV.1 Gaussian process model built on the training data.

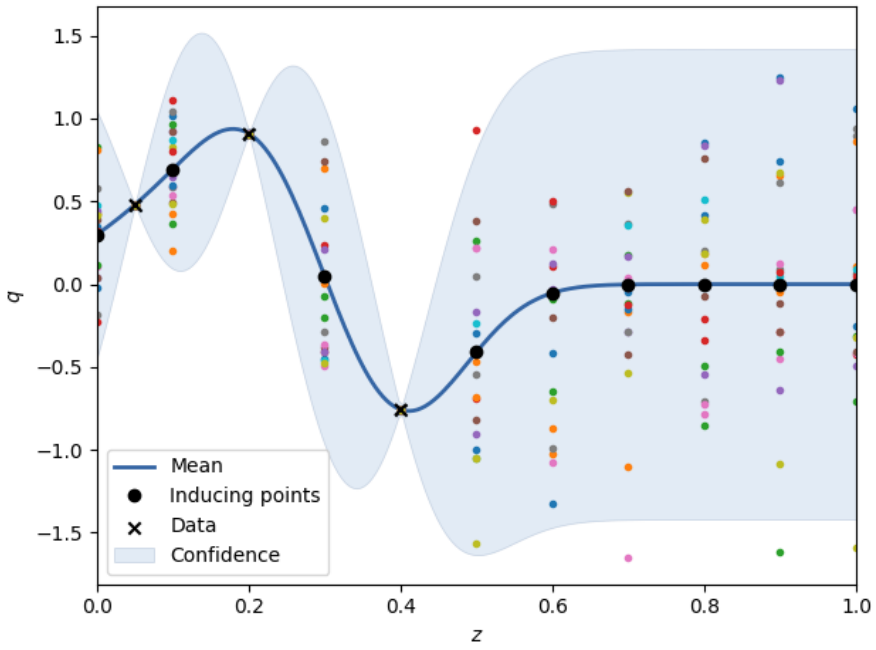



Figure IV.2 Realizations drawn at the inducing points.

To this extent, note that we can write the eigendecomposition (also called spectral decomposition) of the autocovariance matrix $K_{tot}$ from Equation (IV.9), which is Symmetric Positive Definite (SPD) by definition:

$$K_{tot} = U_{tot} \Lambda_{tot} U_{tot}^{\intercal}, \tag{IV.10}$$

where $\Lambda_{tot}$ is the diagonal matrix containing the eigenvalues associated with the eigenvectors stored in the columns of $U_{tot}$. To make this decomposition unique, the eigenvalues are sorted in decreasing order, $\Lambda_{tot_{11}} \geq \Lambda_{tot_{22}} \geq ... \geq 0$, and the eigenvectors

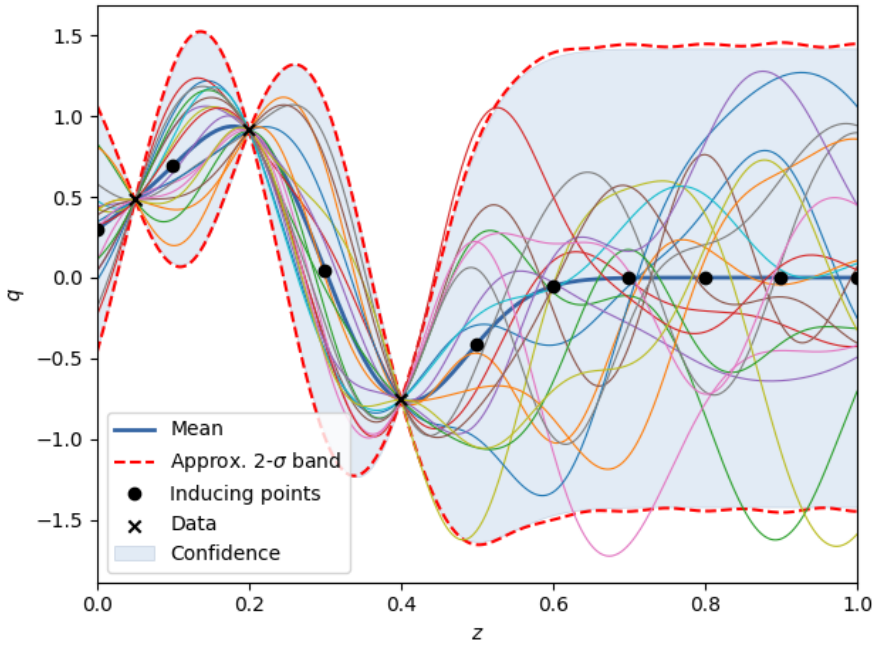



Figure IV.3 Extension to the whole input space with (IV.9). The approximated 2-$\sigma$ band fits well the confidence band.

are orthonormal, so that $U_{tot}^{\intercal} = U_{tot}^{-1}$, which gives:

$$U_{tot_{\cdot i}}^{\intercal} U_{tot_{\cdot j}} = \sum_k U_{tot_{ki}} U_{tot_{kj}} = \delta_{ij}, \tag{IV.11}$$

with $\delta_{ij}$ the Kronecker delta.

The inverse of $K_{tot}$ can thus be written:

$$K_{tot}^{-1} = U_{tot} \Lambda_{tot}^{-1} U_{tot}^{\intercal}, \tag{IV.12}$$

and the square root $K_{tot}^{-1/2}$ of this matrix is:

$$K_{tot}^{-1/2} = U_{tot} \Lambda_{tot}^{-1/2}, \tag{IV.13}$$

which verifies $K_{tot}^{-1} = K_{tot}^{-1/2} K_{tot}^{-1/2\intercal}$.

**KLE and Nyström approximation** A realization of a random field described with a truncated Karhunen-Loève Expansion reads as follows:

$$\boldsymbol{q}_{KLE}^{(i)}(\boldsymbol{z}) = \sum_{j=1}^{N_{KLE}} \sqrt{\lambda_j} \phi_j(\boldsymbol{z}) \theta_j^{(i)}, \tag{IV.14}$$

where the eigenvalues $\lambda_j$ and eigenvectors $\phi_j$ are solution of the following homogeneous Fredholm integral equation of the second kind:

$$\int k(\boldsymbol{z}, \boldsymbol{z}') \phi_j(\boldsymbol{z}) \mathrm{d}\boldsymbol{z} = \lambda_j \phi_j(\boldsymbol{z}') \tag{IV.15}$$

$$\text{with} \quad \int \phi_i(\boldsymbol{z}) \phi_j(\boldsymbol{z}) \mathrm{d}\boldsymbol{z} = \delta_{ij}.$$

As presented in [Williams and Seeger, 2001], the Nyström approximation consists in computing an empirical estimation of the above integral. This Monte Carlo approximation based on points $\boldsymbol{x}_k$ gives:

$$\frac{1}{N_{MC}}\sum_{k=1}^{N_{MC}} k(\boldsymbol{z}_k,\boldsymbol{z}')\phi_j(\boldsymbol{z}_k)\mathrm{d}\boldsymbol{z} \approx \lambda_j\phi_j(\boldsymbol{z}') \tag{IV.16}$$
$$\text{with}\quad \frac{1}{N_{MC}}\sum_{k=1}^{N_{MC}} \phi_i(\boldsymbol{z}_k)\phi_j(\boldsymbol{z}_k)\mathrm{d}\boldsymbol{z} \approx \delta_{ij}.$$

Replacing $\boldsymbol{z}'$ with all $\boldsymbol{z}_{tot_k}$, and setting $N_{MC} = N_{tot}$ so that the Monte Carlo estimation is performed on locations $\boldsymbol{z}_{tot}$, Equation (IV.16) yields:

$$\frac{1}{N_{tot}}K_{tot}\Phi \approx \Lambda\Phi \tag{IV.17}$$
$$\text{with}\quad \frac{1}{N_{tot}}\Phi_{\cdot i}^{\intercal}\Phi_{\cdot j} \approx \delta_{ij}.$$

Matching (IV.17) against (IV.10) and (IV.11) permits to derive:

$$\Phi \approx \sqrt{N_{tot}}U_{tot},$$
$$\Lambda \approx \frac{\Lambda_{tot}}{N_{tot}},$$

which gives the Nyström approximation of the $j$-th eigenfunction from Eq. (IV.16):

$$\phi_j(\boldsymbol{z}) \approx \frac{\sqrt{N_{tot}}}{\Lambda_{tot_{jj}}}\sum_{k=1}^{N_{tot}} k(\boldsymbol{z},\boldsymbol{z}_{tot_k})U_{tot_{kj}}. \tag{IV.18}$$

Using these approximations, KLE-based realizations write:

$$\begin{aligned}\boldsymbol{q}_{KLE}^{(i)}(\boldsymbol{z}) &= \sum_{j=1}^{N_{tot}}\left(\sqrt{\frac{\Lambda_{tot_{jj}}}{N_{tot}}}\frac{\sqrt{N_{tot}}}{\Lambda_{tot_{jj}}}\sum_{k=1}^{N_{tot}}\left(k(\boldsymbol{z},\boldsymbol{z}_{tot_k})U_{tot_{kj}}\right)\theta_{tot_j}^{(i)}\right)\\ &= \sum_{j=1}^{N_{tot}}\left(\frac{\theta_{tot_j}^{(i)}}{\sqrt{\Lambda_{tot_{jj}}}}\sum_{k=1}^{N_{tot}}\left(k(\boldsymbol{z},\boldsymbol{z}_{tot_k})U_{tot_{kj}}\right)\right).\end{aligned} \tag{IV.19}$$

**Inducing points** We proposed to draw realizations using Equation (IV.9). To simplify the following, we assume here that $N_{train} = 0$ and $\boldsymbol{q}_{tot} = \boldsymbol{q}_{ind}$. This implies that matrix $L_{ind}$ corresponds to the square root matrix $K_{tot}^{1/2}$.

By definition, in this context, $\boldsymbol{q}_{tot}^{(i)} = K_{tot}^{1/2}\boldsymbol{\theta}_{tot}^{(i)}$, with $\boldsymbol{\theta}_{tot}^{(i)} \sim \mathcal{N}(\boldsymbol{0}, I)$. A realization at location $\boldsymbol{z}$ then writes:

$$\begin{aligned}\boldsymbol{q}^{(i)}(\boldsymbol{z}) &= k(\boldsymbol{z},\boldsymbol{z}_{tot})K_{tot}^{-1}\boldsymbol{q}_{tot}^{(i)}\\ &= k(\boldsymbol{z},\boldsymbol{z}_{tot})K_{tot}^{-1/2}\boldsymbol{\theta}_{tot}^{(i)}\\ &= k(\boldsymbol{z},\boldsymbol{z}_{tot})U_{tot}\Lambda_{tot}^{-1/2}\boldsymbol{\theta}_{tot}^{(i)}\\ &= \sum_{j=1}^{N_{tot}}\left(\frac{\theta_{tot_j}^{(i)}}{\sqrt{\Lambda_{tot_{jj}}}}\sum_{k=1}^{N_{tot}}\left(k(\boldsymbol{z},\boldsymbol{z}_{tot_k})U_{tot_{kj}}\right)\right),\end{aligned} \tag{IV.20}$$

which corresponds exactly to Equation (IV.19).

**Remark** *Note that in the case $\boldsymbol{q}_{tot} \neq \boldsymbol{q}_{ind}$, which permits to condition the realizations on some training data, the same equivalence can be shown by comparing (IV.9) to a conditional Karhunen-Loève model [Tipireddy et al., 2019].*

### IV–2.2.3 Quality indicator

To attain a good accuracy of the random field approximation, a key element is the number of inducing points. We propose in this section an indicator to assess the approximation quality. Recall that we proposed to draw realizations as specific GP predictive means, following Equation (IV.9). Intuitively, if the density of inducing points is high enough with respect to the lengthscales, the predictive means $\boldsymbol{q}_{draw}^{(i)}$ will be able to represent all scales of the random process. In such cases, the associated predictive variance remains very low over the whole design space $\mathcal{X}$. The GP covariance reads:

$$\Sigma_{tot}(\boldsymbol{z}_{draw}) = K_{**} - \boldsymbol{k}_*^\intercal K_{tot}^{-1} \boldsymbol{k}_*, \tag{IV.21}$$

with $K_{**} = k_{train}(\boldsymbol{z}_{draw}, \boldsymbol{z}_{draw})$. The predictive variance vector $\sigma_{tot}^2(\boldsymbol{z}_{draw})$ is the diagonal of $\Sigma_{tot}(\boldsymbol{z}_{draw})$. We propose to base the quality indicator on the mean predictive variance over all points $\boldsymbol{z}_{draw}$. More precisely, as by definition $\forall i, \sigma_{tot\,i}^2(\boldsymbol{z}_{draw}) \in [0, K_{**ii}]$, we propose the following normalized indicator:

$$\alpha = 1 - \frac{1}{N_{draw}} \sum_{i=1}^{N_{draw}} \frac{\sigma_{tot\,i}^2(\boldsymbol{z}_{draw})}{K_{**ii}}, \tag{IV.22}$$

where $\alpha \in [0,1]$ and $\alpha \simeq 1$ denotes a low predictive variance, and thus a good approximation quality.

Note that for most of the usual kernel functions, the diagonal of $K_{**}$ is filled with $\sigma^2$, the variance hyperparameter of the chosen kernel. This is notably the case with the exponential, squared exponential and Matern kernels. In these cases, using $\sigma^2 N_{draw} = \mathrm{Tr}\big(K_{**}\big)$ and linearity of the trace, the indicator reads:

$$\alpha = \frac{\mathrm{Tr}\big(\boldsymbol{k}_*^\intercal K_{tot}^{-1} \boldsymbol{k}_*\big)}{\sigma^2 N_{draw}}. \tag{IV.23}$$

**Remark** *Note that one may recognize a Frobenius norm at the numerator of the proposed indicator:*

$$\begin{aligned} \mathrm{Tr}\big(\boldsymbol{k}_*^\intercal K_{tot}^{-1} \boldsymbol{k}_*\big) &= \mathrm{Tr}\big(\boldsymbol{k}_*^\intercal (L_{tot} L_{tot}^\intercal)^{-1} \boldsymbol{k}_*\big), \\ &= \mathrm{Tr}\big((L_{tot}^{-1} \boldsymbol{k}_*)^\intercal L_{tot}^{-1} \boldsymbol{k}_*\big), \\ &= ||L_{tot}^{-1} \boldsymbol{k}_*||_F^2, \end{aligned} \tag{IV.24}$$

*where $L_{tot}$ refers again to the Cholesky decomposition of $K_{tot}$.*

The above remark gives some insights into the proposed indicator. Multiplying by the matrix $L_{tot}$ permits to scale and correlate data according to the covariance matrix $K_{tot}$. Hence, in the above, $L_{tot}^{-1}$ decorrelates $\boldsymbol{k}_*$ and shrinks the values by $\sigma$. Because $\boldsymbol{k}_*$ contains variances and the numerator is squared, the $\sigma^2$ in the denominator shrinks the results again to reach the $[0,1]$ range. This returns the decorrelated correlations

between $\boldsymbol{z}_{tot}$ and $\boldsymbol{z}_{draw}$ for each column of $L_{tot}^{-1}\boldsymbol{k}_*$, that can at most sum to 1. This sum represents how much information is known at a given point in $\boldsymbol{z}_{draw}$ thanks to $\boldsymbol{z}_{tot}$. Finally, the mean over all columns explains the presence of $N_{draw}$ in Equation (IV.22).

The proposed indicator is pictured in Figures IV.4 and IV.5 with nine and three inducing points, respectively. The approach gives an empirical 2-$\sigma$ band very close to the real one in Figure IV.4. The quality indicator is computed and returns a value of 0.995. On the contrary, the approximated random field in Figure IV.5 is very dissimilar from the underlying GP, and the quality indicator has only a value of 0.825.

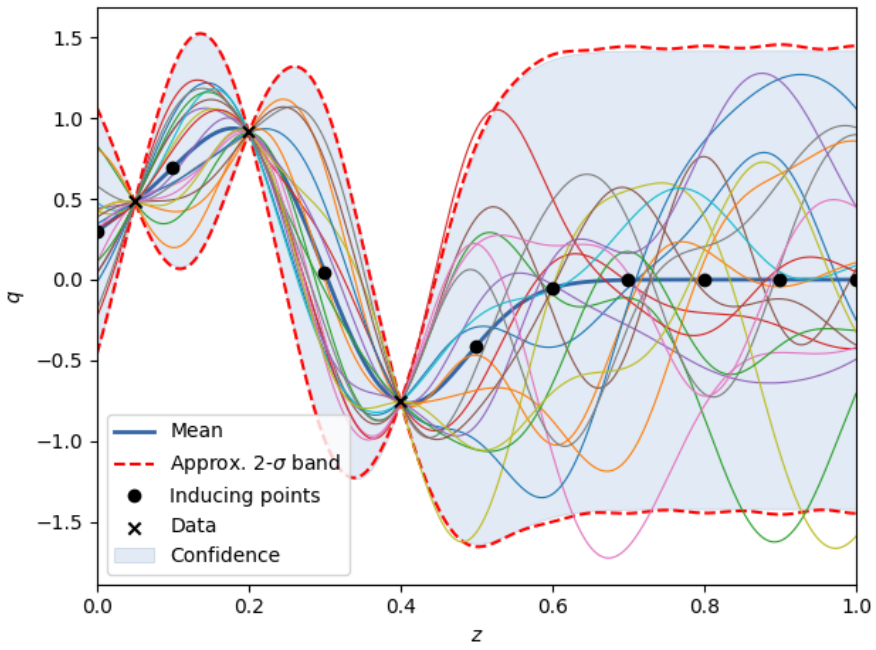



Figure IV.4 Approximated random field with 9 inducing points, $\alpha = 0.995$.

We give here an illustration of this indicator $\alpha$ in terms of distance between an approximated empirical covariance and the true matrix. More specifically, with $D$ the dimension of the input parameters $\boldsymbol{z}$, we define $\boldsymbol{q}$ as:

$$q(\boldsymbol{z}) = \sin(\boldsymbol{L}^\intercal \boldsymbol{z}) = \sin\left(\sum_{i=1}^{D} L_i z_i\right). \tag{IV.25}$$

Parameters $D$ and $\boldsymbol{L}$ are random variables, which yields a random function $q_{D,\boldsymbol{L}}$. In practice, $D \sim \mathcal{U}([\![1,4]\!])$ and $\boldsymbol{L} \sim \mathcal{U}([3,20])^D$.

For each sampled function $q_{D^{(k)},\boldsymbol{L}^{(k)}}$, a GP is built on 10 training data $q_{D^{(k)},\boldsymbol{L}^{(k)}}(\boldsymbol{z}_{train})$. A set of points $\boldsymbol{z}_{draw}$ is drawn uniformly within the input space. The predictive covariance $K$ on these points is given by Equation (I.32). A set of well spread inducing points of size $N_{ind} \sim \mathcal{U}([\![10,60]\!])$ is also drawn (*e.g.* with a maximin Latin Hypercube Sampling (LHS) strategy) and $10^5$ realizations $q^{(i)}_{D^{(k)},\boldsymbol{L}^{(k)}}(\boldsymbol{z}_{draw})$ are computed at $10^3$ locations $\boldsymbol{z}_{draw}$. The empirical covariance between all these realizations gives an approximation $\widetilde{K}$ of $K$ and the indicator $\alpha$ from Equation (IV.22) is computed.

All this process is repeated 300 times, and we plot in Figure IV.6 the evolution of the normalized Frobenius distance between $K$ and $\widetilde{K}$ with respect to $\alpha$.

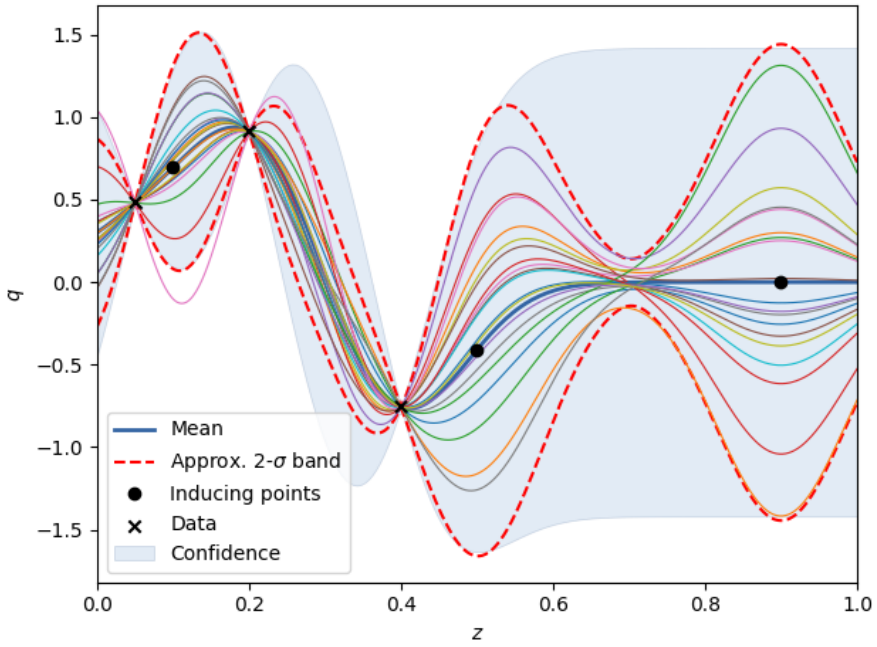



Figure IV.5 Approximated random field with 3 inducing points, $\alpha = 0.825$.

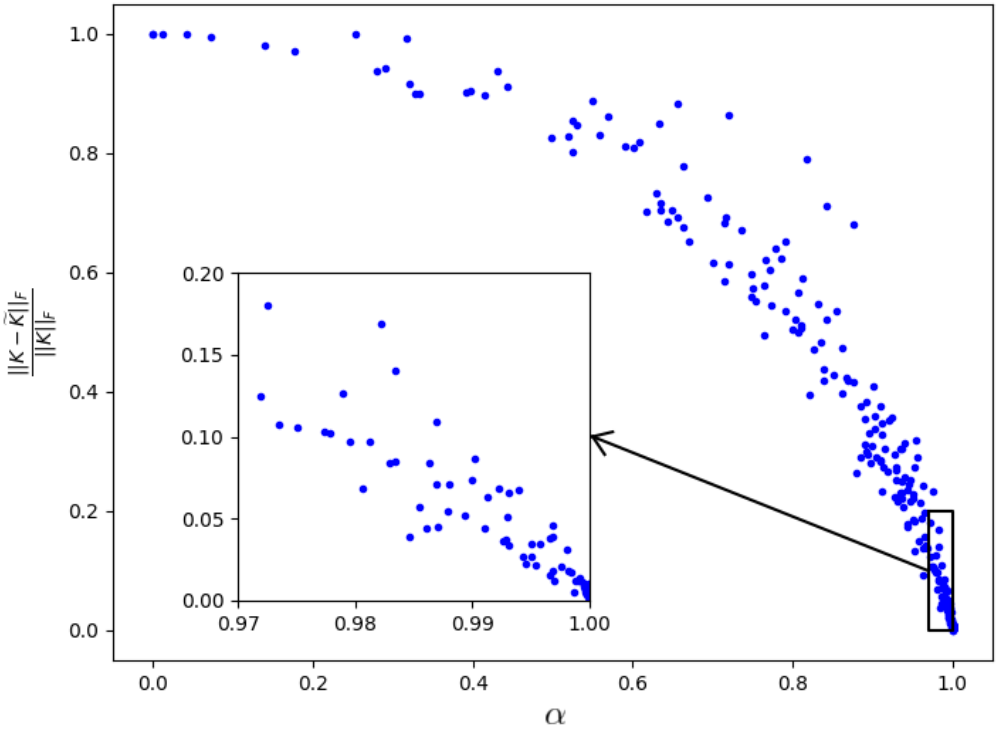



Figure IV.6 Accuracy of the empirical covariance $\widetilde{K}$ approximated with inducing points with respect to the proposed indicator.

Figure IV.6 shows that a strong correlation exists between the proposed indicator $\alpha$ and the accuracy of the empirical covariance. A value of the indicator very close to 1 seems to ensure an accurate representation of the random field. Notably, the zoomed window permits to see that when $\alpha$ is greater than 0.99 or 0.995, the worst recorded empirical covariances only yield 5 to 10% relative error.

**Remark** *Note that several appropriate norms for covariance matrices exist. Namely, the logarithmic, Cholesky-based, and Riemannian distances between covariance matrices allow to preserve positive definiteness and have many desirable properties [Herdin et al., 2005]. However, because we draw very badly conditioned empirical covariances, very small negative eigenvalues may arise, that prohibit the use of such distances. For its simplicity and interpretability, the Frobenius distance is the most reliable choice.*

### IV–2.2.4 Adaptive strategy

To take advantage of the proposed quality indicator, we propose an iterative algorithm ensuring that good value is attained before returning the approximated realizations. The number of inducing points is hence increased as long as a prescribed value for $\alpha$ has not been reached. At each iteration, these points are spread in the input space using the classical maximin criterion on a Latin Hypercube Sampling (LHS). The algorithm is given in Algorithm IV.1.

**Algorithm IV.1** Algorithm for choosing the number of inducing points

Choose threshold $\alpha_* \in [0,1]$
Choose initial number of inducing points $N_{init}$ and added per iteration $N_{add}$
Build a GP on the training data
$N_{ind} = N_{init}$
**while** True **do**
    Construct a DoE of $N_{ind}$ points using a maximin LHS
    Compute $\alpha$ from Equation (IV.22)
    **if** $\alpha \geq \alpha_*$ **then**
        Draw realizations with Equation (IV.9) using the DoE of inducing points
        Return realizations
    **else**
        $N_{ind}$ += $N_{add}$
    **end if**
**end while**

In the following, the threshold $\alpha_*$ will take values of 0.99 or 0.995, basing on Figure IV.6.

The simple approach proposed here is very intuitive and reveals very cost-efficient.

**Remark** *Note that the maximin LHS sampling technique could be replaced by any state of the art Design of Experiments (DoE) generation method. More interestingly, the indicator $\alpha$ itself could be used to optimize the sampling, depending on the location of the points $\mathbf{z}_{draw}$ and of the chosen kernel. By maximizing $\alpha$, one could notably take advantage of periodic kernels to spread information efficiently or focus the inducing points in regions where $\mathbf{z}_{draw}$ are more present.*

## IV–2.3 Implementation in SAMATA

As raised in Section IV–1, we need to provide specific numerical ingredients to apply the SAMATA strategy.

This sampling technique allows generating realizations of $\boldsymbol{q}$ on an extensive set of points $\boldsymbol{z}_{draw}$, to retrieve realizations of the robustness and reliability measures $\mathbf{P}$ with Equation (IV.7). In this manner, the Gaussian assumption of the GP surrogate model built on $\boldsymbol{q}$ directly translates into a distribution of $\mathbf{P}$ in the form of its empirical PDF.

Similarly to the approach proposed in Chapter III, the mean of this approximated distribution is returned to the optimization process as $\widetilde{\boldsymbol{\rho}}$, and three times the standard deviation is used as an accuracy metric $\boldsymbol{\Delta}$ from Section II–3. Concerning the refinements, we make use of the strategy presented in Section III–4.1.1 with the linear combination of weighted partial criteria.

Finally, the Surrogate-Assisting (SA) model is here constructed in a specific way in order to best exploit all the available information. SA-based measure approximations must also take the form of samplings, and should be drawn jointly with the realizations computed with Equation (IV.7). To this extent, we propose to define the SA-based realizations as random draws of a GP surrogate model conditioned on the computed joint realizations $\mathbf{P}_c^{(i)}$. In practice, the SA-based measure approximation on any set of design $\boldsymbol{x}_{SA}$ is a collection of realizations $\mathbf{P}_{SA}^{(i)}(\boldsymbol{x}_{SA})$ computed as such:

$$\mathbf{P}_{SA}^{(i)}(\boldsymbol{x}_{SA}) = \boldsymbol{\mu}_{SA}^{(i)}(\boldsymbol{x}_{SA}) + L_{SA}\boldsymbol{Y}^{(i)}, \quad \boldsymbol{Y} \sim \mathcal{N}(\mathbf{0}, I), \tag{IV.26}$$

where the predictive mean write:

$$\boldsymbol{\mu}_{SA}^{(i)}(\boldsymbol{x}_{SA}) = \boldsymbol{k}_*^\intercal K_c^{-1} \mathbf{P}_c^{(i)}, \tag{IV.27}$$

and $L_{SA}$ results from a Cholesky decomposition performed on $\Sigma_{SA}$ defined as the predictive covariance $\Sigma_{GP}(\boldsymbol{x}_{SA})$ defined in Equation (I.32). In the above, $\mathbf{P}_c^{(i)}$ refers to realizations drawn from Equation (IV.7), on the set designs $\boldsymbol{x}_c$. $K_c$ corresponds to the autocovariance matrix $k(\boldsymbol{x}_c, \boldsymbol{x}_c)$ and $\boldsymbol{k}_*$ to the covariance $k(\boldsymbol{x}_c, \boldsymbol{x}_{SA})$.

This SA model construction is illustrated on an example set of samples in Figure IV.7, where realizations of $\boldsymbol{\rho}$ are jointly drawn in ascending order. Two bimodal sets of samples are notably drawn at $x = 0.15$ and $x = 0.45$ in order to emphasize the need for non-parametric reconstruction. Some SA-based functions $\mathbf{P}_{SA}^{(i)}$ form Equation (IV.26) are represented over the input space. One can qualitatively see that such SA model naturally captures the two bi-modal densities.

**Remark** *All measure approximations from Equation (IV.7) are jointly computed. When a refinement is carried out on one individual, realizations $\mathbf{P}_c^{(i)}$ are then approximated for all designs in $\boldsymbol{x}_c$.*
*Similarly, the SA-based realizations must all be jointly drawn. This is key for being able to use the joint POP computation from Equations (IV.1) and (IV.2).*

These formulations can be incorporated within the SAMATA strategy, presented in Section II–3 and pictured in Figure II.5. The specific algorithm is given in Algorithm IV.2.

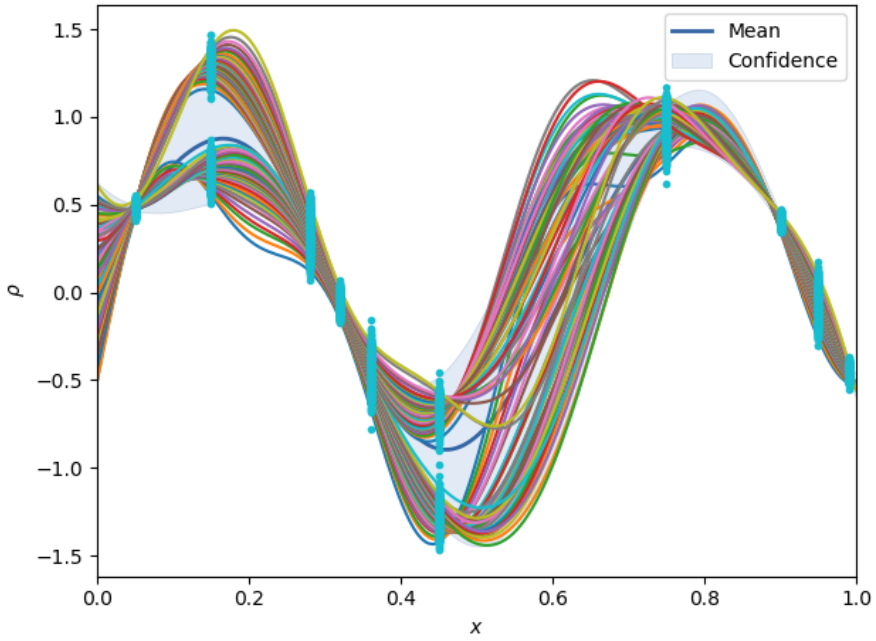



Figure IV.7 Realizations of the joint Surrogate-Assisted model. Training data in light blue and underlying GP model in dark blue, in the background.

**Algorithm IV.2** SAMATA algorithm with joint measure approximations

Launch $N_{init}$ evaluations $\boldsymbol{q}(\boldsymbol{x}_i, \boldsymbol{\xi}_i)$
Generate a first coupled space surrogate model $\widehat{\boldsymbol{q}}$
Loop over values of $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$
**while** Optimization running **do**
    Get new designs $\mathcal{X}_{new}$
    Initialize $\mathcal{X}_c$ empty
    Compute $\mathbf{P}_{SA}$ for all $\boldsymbol{x} \in \mathcal{X}_{new}$ from (IV.26)
    Compute $\mathcal{X}_{SA}^0 = \left\{\boldsymbol{x} \in \mathcal{X}_{new} \mid \boldsymbol{\Delta}_{SA}(\boldsymbol{x}) \leq \boldsymbol{s}_1\right\}$ and $\mathcal{X}_c^0 = \mathcal{X}_{new} \setminus \mathcal{X}_{SA}^0$
    Deduce $\mathcal{X}_r^0$ from $\mathcal{X}_c^0$ using (II.26)
    $k = 0$
    **while** $\mathcal{X}_r^k \neq \emptyset$ **do**
        Compute $\forall \boldsymbol{x} \in \mathcal{X}_{new}, POP_{min}(\boldsymbol{x})$ w.r.t. all designs using (II.24) and (IV.2)
        Find $\boldsymbol{x}^*$ from $\mathcal{X}_r^k$ using (II.27)
        **if** $\boldsymbol{x}^* \notin \mathcal{X}_c$ **then**
            Add $\boldsymbol{x}^*$ in $\mathcal{X}_c$
        **end if**
        Find $\boldsymbol{\xi}^*$ that maximizes the refinement criterion $c(\boldsymbol{\xi})$ defined in (III.42)
        Launch evaluation of $\boldsymbol{q}(\boldsymbol{x}^*, \boldsymbol{\xi}^*)$
        Update the coupled space surrogate model $\widehat{\boldsymbol{q}}$
        Update all approximations with $\mathbf{P}_c^{(i)}(\boldsymbol{x})$ for $\boldsymbol{x} \in \mathcal{X}_c$ using Equation (IV.7)
        Update $\mathbf{P}_{SA}$ for all $\boldsymbol{x} \in \mathcal{X}_{new} \setminus \mathcal{X}_c$
        Compute $\mathcal{X}_c^{k+1}$, $\mathcal{X}_{SA}^{k+1}$ and $\mathcal{X}_r^{k+1}$
        $k = k + 1$
    **end while**
    Return $\mathbb{E}\left[\mathbf{P}(\boldsymbol{x})\right]$ to the optimizer for all $\boldsymbol{x} \in \mathcal{X}_{new}$
**end while**
Return all measure approximations $\mathbf{P}(\boldsymbol{x})$ and their associated POP

**Discussion** *In this section, we illustrate a strategy for performing sampling-based approximations of the distribution* $\mathbf{P}$ *within SAMATA. By assuming that a coupled-space GP surrogate model* $\hat{q}$ *can be constructed in* $\mathcal{X} \times \Xi$*, we can propose joint realizations* $\mathbf{P}^{(i)}$ *at all training locations* $\boldsymbol{x}_{train}$ *simultaneously from realizations of the GP* $\hat{q}$*. The associated computational burden could quickly become unbearable when numerically approximating the robustness and reliability measures of interest. We thus proposed an approximating strategy that allows for drawing realizations of* $\hat{q}$ *on a huge number of points in the coupled space with a limited computational budget. The associated accuracy can be monitored through a specific indicator, and an adaptive strategy is proposed for ensuring good behavior of the technique.*
*An important remark here concerns the computational burden associated with drawing realizations of the random robustness and reliability measures. Also only requiring several seconds to several minutes, this computational burden is much higher than SABBa over-conservative intervals from Chapter III. Thus, sampling-based SAMATA is particularly adapted to tackling expensive functions* $\boldsymbol{q}$ *while SABBa may reveal more efficient on inexpensive functions (e.g. analytical).*

## IV–3 KDE-based random field surrogate

In this section, we assume that the construction of the aforementioned coupled space surrogate model is unfeasible. This problem can arise in many settings. Notably, the uncertain space may feature too many dimensions or have non-parametric inputs, such as [Soize, 2005] with random matrices, and [Sabater et al., 2020] where uncertain parameters are considered uncontrollable and are disregarded. In all these cases, specific methods for Uncertainty Propagation (UP) exist and can be used, but the strategy presented in the previous section that requires surrogate modeling in the uncertain dimension becomes inapplicable when uncertainties cannot be parametrized.

Many UP techniques have built-in methods for estimating their accuracy. For example, the variance of the Monte Carlo estimator which assumes a Gaussian density and error bounds on the statistics of interest are proposed in [Ferson et al., 2010, Barth, 2016]. They can be interpreted as uniform distributions under the maximum entropy principle. In this section, we propose sampling-based estimations of the statistics distributions through probabilistic metamodeling or bootstrap strategies.

Within the SAMATA strategy, the Surrogate-Assisting model must approximate the true function $\boldsymbol{\rho}$ based on the estimated random field $\mathbf{P}$ discretized at some design locations. We thus propose to construct a probabilistic surrogate model that builds on random realizations of the robustness and reliability measures $\mathbf{P}^{(i)}(\boldsymbol{x})$ at some given designs and on a Kernel Density Estimation (KDE) strategy.

Obtaining these disjoint realizations $\mathbf{P}^{(i)}(\boldsymbol{x})$ at different locations can be achieved with several strategies, depending on the context:

- With parametric uncertainties, a GP surrogate model of $\boldsymbol{q}$ can be constructed in the uncertain dimensions for a given design $\boldsymbol{x}_k$. Equation (IV.7) can then be readily exploited for drawing disjoint realizations of $\mathbf{P}$. We slightly modify the

notations for more clarity:

$$\forall k \in [\![1, N_x]\!], \quad \mathbf{P}^{(i)}(\boldsymbol{x}_k) = \frac{1}{N_\xi} \sum_{j=1}^{N_\xi} \boldsymbol{q}_{\boldsymbol{x}_k}^{(i)}(\boldsymbol{\xi}_j). \tag{IV.28}$$

Note that contrarily to the previous section, this formula is employed separatedly for each design point $\boldsymbol{x}_k$. This disjoint characteristic alleviates the computational bottleneck as we only require here $N_x$ Cholesky decompositions on $N_\xi \times N_\xi$ matrices, *i.e.* one for each function $\boldsymbol{q}_{\boldsymbol{x}_k}$.

- When dealing with non-parametric uncertainties, different strategies must be employed. Notably, we propose to draw realizations $\mathbf{P}^{(i)}$ of the statistical estimator at each design $\boldsymbol{x}_k$ with a bootstrapping strategy. This technique is made explicit in Section IV–5.

Whatever the chosen approach, a prediction $\mathbf{P}_{SA}(\boldsymbol{x})$ of $\mathbf{P}$ is then constructed over all $\mathcal{X}$ based on these empirical training distributions $\mathbf{P}^{(i)}(\boldsymbol{x}_{train})$ at some training locations. Note that if we were to assume Gaussian distribution for each $\mathbf{P}(\boldsymbol{x}_{train})$, the surrogate model $\mathbf{P}_{SA}(\boldsymbol{x})$ could be a Gaussian Process. However, we aim here for non-parametric techniques. Training data $\mathbf{P}(\boldsymbol{x}_{train})$ are sets of samples, on which a KDE distribution can be computed. We propose to construct the Surrogate-Assisting model so that it agrees with this training KDE distribution at each training point.

Also, similarly to simple Gaussian Processes, we aim to enforce smoothness on $\mathbf{P}$ over $\mathcal{X}$ through spatial correlation. In the following, we propose to ensure smoothness and KDE agreement at the training points through the use of a Markov-Chain Monte Carlo approach.

Section IV–3.1 first introduces the Kernel Density Estimation (KDE) technique and the chosen bandwidth, which are then exploited in Section IV–3.2 within a rejection sampling technique for obtaining an empirical estimation $\mathbf{P}_{SA}(\boldsymbol{x})$ of $\mathbf{P}$.

## IV–3.1 Kernel Density Estimation

Given a set of samples $\mathbf{P}^{(i)}(\boldsymbol{x}_{train})$ at a training location $\boldsymbol{x}_{train}$ and a specific smoothing kernel, Kernel Density Estimation (KDE) yields an approximation of the underlying continuous density $\phi_{\boldsymbol{x}_{train}}$. The smoothing kernel permits to diffuse the discrete information locally and return a non-parametric approximation of the Probability Density Function (PDF) of $\mathbf{P}$ at a given $\boldsymbol{x}_{train}$. Although approaches with varying weights exist, we use here the equally weighted KDE equations:

$$\phi_{\boldsymbol{x}_{train}}^{KDE}(\boldsymbol{\rho}) = \frac{1}{N_{samples}} \sum_{i=1}^{N_{samples}} k\big(\boldsymbol{\rho}, \mathbf{P}^{(i)}(\boldsymbol{x}_{train})\big), \tag{IV.29}$$

where $k$ is the aforementioned kernel, verifying $\forall \boldsymbol{\rho}', \int_{\mathbb{R}^n} k(\boldsymbol{\rho}, \boldsymbol{\rho}')\mathrm{d}\boldsymbol{\rho} = 1$. This yields directly that $\phi_{\boldsymbol{x}_{train}}^{KDE}(\boldsymbol{\rho})$ also sums to 1. Recall that $\boldsymbol{\rho}$ is a dummy variable here.

In the following, we choose to use the Gaussian kernel. For simplicity, we only use non-correlated Gaussian distributions. The standard deviations must be determined, and are usually referred to as the kernel bandwidths $\boldsymbol{h}$. Several strategies can be applied to calibrate $\boldsymbol{h}$, either using a rule of thumb or cross-validation approaches. In this work, we use the Silverman's bandwidth [Silverman, 1986] on normalized data.

In practice, the values $\boldsymbol{\rho}$ are normalized to $\bar{\boldsymbol{\rho}}$, so that:

$$\forall \boldsymbol{x}_{train}, \forall k, \bar{\rho}_k = \frac{\rho_k - \min_i P_k^{(i)}(\boldsymbol{x}_{train})}{\max_i P_k^{(i)}(\boldsymbol{x}_{train}) - \min_i P_k^{(i)}(\boldsymbol{x}_{train})}. \tag{IV.30}$$

We denote with $k_h$ the Gaussian kernel of standard deviation $h$,

$$k_h(\boldsymbol{\rho}, \boldsymbol{\rho}') = \frac{1}{h\sqrt{2\pi}} \exp\left( -\frac{||\boldsymbol{\rho} - \boldsymbol{\rho}'||_2^2}{2h^2} \right). \tag{IV.31}$$

The kernel used in Equation (IV.29) is $k^*(\boldsymbol{\rho}, \boldsymbol{\rho}') = k_{h^*}(\bar{\boldsymbol{\rho}}, \bar{\boldsymbol{\rho}}')$, where the chosen bandwidth comes from Silverman's rule of thumb:

$$h^* = \left( \frac{4}{n \times (d+2)} \right)^{\frac{1}{d+4}}. \tag{IV.32}$$

Note that the kernel $k^*$ is computed for each $\boldsymbol{x}_{train}$, thus for each estimated density $\phi^{KDE}_{\boldsymbol{x}_{train}}(\boldsymbol{\rho})$.

**Remark** *The bandwidth selection could be performed by maximizing the average Leave-One-Out likelihood. This strategy would more precisely capture rough distribution shapes. However, to avoid making the overall approach too expensive, we chose to use the Silverman's bandwidth, that shows good performance on the following test-cases.*

## IV–3.2 Surrogate-model construction

Using the KDE technique, we now have a set of non-parametric density estimations $\phi^{KDE}_{\boldsymbol{x}_{train}}(\boldsymbol{\rho})$ at each training locations $\boldsymbol{x}_{train}$, constructed from the disjoint measure realizations $\mathbf{P}^{(i)}(\boldsymbol{x}_{train})$.

Based on these densities, we aim to construct an approximation of the random field $\mathbf{P}$ over the whole design space $\mathcal{X}$.

We propose to make use of the Metropolis-Hasting (MH) algorithm to impose these computed KDE densities on realizations of a heteroscedastic GP, which will provide the desired smoothness for $\boldsymbol{\rho}$. In practice, similarly to the approach proposed in Section III–4, we build a GP on the empirical mean of our samplings at each $\boldsymbol{x}_{train}$, with heteroscedastic noises corresponding to the empirical standard deviation. The MH algorithm then rejects some realizations of this GP depending on their accordance with the KDE densities.

In the MH algorithm, the probability of acceptance for going from realization $\mathbf{P}^{(i)}$ to $\mathbf{P}^{(j)}$ reads:

$$\alpha = \frac{\phi^{(j)}_{target} \phi^{(i)}_{proposal}}{\phi^{(i)}_{target} \phi^{(j)}_{proposal}}. \tag{IV.33}$$

Probability $\alpha$ can be seen as the ratio between $p^{(j)} = \dfrac{\phi^{(j)}_{target}}{\phi^{(j)}_{proposal}}$ and $p^{(i)}$. Each value $p^{(j)}$ denotes how interesting a realization $\mathbf{P}^{(j)}$ is.

**Remark** *Note that the proposal distribution is usually a local deviation from the previous location. Here the proposal distribution is global and not conditioned on the previous draw. Indeed, realizations are drawn from the heteroscedastic GP. The KDE densities are then taken into account for computing the probability of acceptance.*

As raised above, the proposal distribution comes from the heteroscedastic GP and the target density refers to the computed KDE. Value $\phi_{KDE}^{(i)}$ defined hereafter is thus used as target density $\phi_{target}^{(i)}$:

$$\phi_{KDE}^{(i)} = \prod_{k=1}^{N_{train}} \phi_{\boldsymbol{x}_{train_k}}^{KDE}(\mathbf{P}^{(i)}(\boldsymbol{x}_{train_k})), \tag{IV.34}$$

and $\phi_{proposal}^{(i)}$ corresponds to $\phi_{GP}^{(i)}$, defined as follows:

$$\phi_{GP}^{(i)} = \phi_{Gauss}\big(\mathbf{P}^{(i)}(\mathcal{X}_{train}), \mu_{GP}(\mathcal{X}_{train}), \Sigma_{GP}(\mathcal{X}_{train})\big), \tag{IV.35}$$

where $\phi_{Gauss}$ refers to the PDF of the multivariate normal of the GP discretized on the training points $\boldsymbol{x}_{train}$:

$$\phi_{Gauss}(\boldsymbol{\rho}, \mu, \Sigma) = \frac{\exp\Big(-\frac{1}{2}(\boldsymbol{\rho}-\mu)^{\intercal}\Sigma(\boldsymbol{\rho}-\mu)\Big)}{\sqrt{(2\pi)^{N_{train}}|\Sigma|}}. \tag{IV.36}$$

However, using $\phi_{GP}$ as proposal distribution will penalize smooth functions, with high $\phi_{GP}$ value, in favor of rougher functions, with low $\phi_{GP}$. On the contrary, we aim here to retain the inherent smoothness of the GP surrogate model.

To this extent, we propose to replace the joint computation of $\phi_{GP}$ with an uncorrelated one $\phi'_{GP}$ by setting every off-diagonal values of $\Sigma_{GP}(\mathcal{X}_{train})$ from Equation (IV.36) to zero.

Furthermore, we propose to tune this proposal density $\phi'_{GP}$ to ensure that the constructed probabilistic surrogate model relapses to a heteroscedastic GP in the case where all KDE densities are Gaussian. This can be achieved by enforcing $\alpha = 1$ when the KDE densities are Gaussian.

If we assume all KDE to be Gaussian, each density corresponds to the Gaussian noise associated to this location, that is used to construct the proposal heteroscedastic GP. Thus, $\phi_{KDE}$ corresponds to the uncorrelated noise density $\phi_{noise}$ defined hereafter. With $\mu_{noise}$ the training data and $\Sigma_{noise}$ the diagonal matrix containing all input variances, $\phi_{noise}^{(i)}$ writes:

$$\phi_{noise}^{(i)} = \phi_{Gauss}\big(\mathbf{P}^{(i)}(\mathcal{X}_{train}), \mu_{noise}, \Sigma_{noise}\big). \tag{IV.37}$$

Setting $\phi_{proposal}$ to $\phi_{noise}$ gives the following formula for :

$$\alpha = \frac{\phi_{KDE}^{(j)}\phi_{noise}^{(i)}}{\phi_{KDE}^{(i)}\phi_{noise}^{(j)}}, \tag{IV.38}$$

which verifies $\phi_{KDE} = \phi_{noise} \implies \alpha = 1$, relapsing to the simple heteroscedastic Gaussian Process.

**Remark** *For numerical reasons, it is highly recommended to consider log probabilities. This choice usually avoids reaching impracticably high values during the computation.*

### IV–3.2.1 Illustration

This MH technique for drawing realizations $\mathbf{P}^{(i)}$ that agree with the training sets of samples at $\boldsymbol{x}_{train}$ is applied on the same data set as in Figure IV.6. Note that here the samples at $\boldsymbol{x}_{train}$ are not supposed to be jointly drawn and are considered independent. These samples $\mathbf{P}^{(i)}(\boldsymbol{x}_{train})$ are drawn in light blue at each training point, and realizations from the probabilistic SA surrogate model are represented with colored curves in Figure IV.8. Again, this example features bimodal behaviors at $\boldsymbol{x} = 0.15$ and $\boldsymbol{x} = 0.45$.

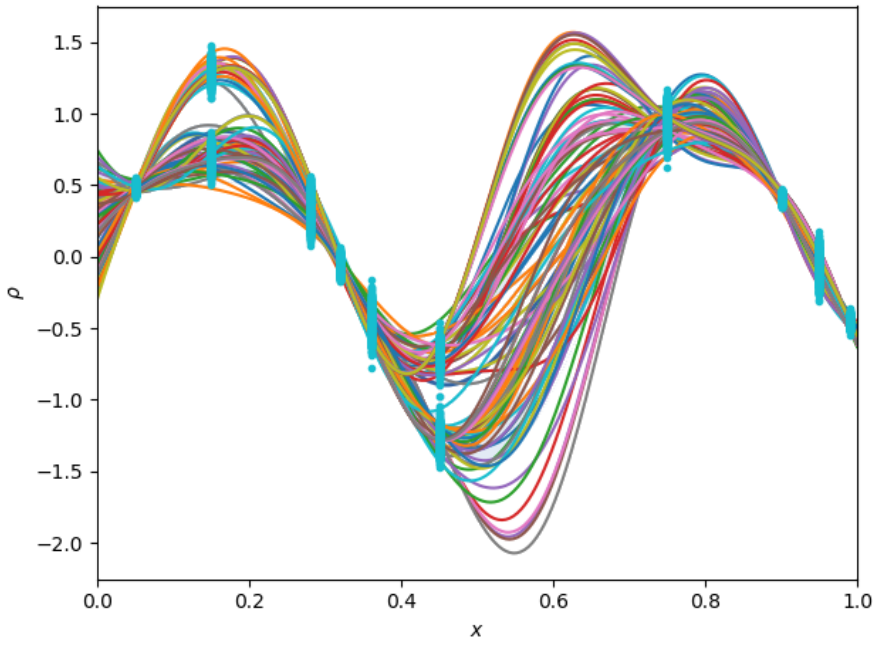



Figure IV.8 Realizations of the proposed surrogate model with training data as light blue points.

The proposed strategy efficiently captures these non-Gaussian behaviors while relapsing to its proposal GP distribution in areas where the training data is close to Gaussian. The proposed approach can be seen as a generalization of the heteroscedastic GP to non-Gaussian noises and can be readily used as Surrogate-Assisting model in a *Separated Spaces* sampling-based SAMATA strategy.

#### Conditional Mean Embeddings

**Remark** *For constructing this KDE-based surrogate model, another option is to use the Kernel Mean Embedding (KME) setting presented in Section I–2.2. More specifically, Conditional Mean Embedding (CME) gives a non-parametric kernel-based reconstruction of the conditional PDF $p_{\boldsymbol{x}}(\boldsymbol{\rho})$ based on some training densities $\phi_{\boldsymbol{x}_{train}}(\boldsymbol{\rho})$.*
*The main advantage of the KDE-based surrogate model introduced in the preceding section is that CME only ensures smoothness between distributions instead of between values of $\boldsymbol{\rho}$, which is our value of interest. Additionally, contrarily to GP models, CME does not yield predictive variance or any accuracy metric. For these reasons, CME is not exploited here, in favor of the KDE-based surrogate model presented above.*

**Discussion** *In this section, a strategy has been proposed for approximating the random field* $\mathbf{P}$ *based on disjoint realizations at some training locations. This approach relies on KDE approximations of the Probability Density Function (PDF) at these training locations and MH algorithm for rejecting functional realizations of* $\mathbf{P}$ *that do not comply with these KDE densities.*
*This strategy permits to approximate the random field of interest with good representativeness without relying on a coupled-space surrogate model, as in the preceding section. However, the computational burden associated with the use of the MH algorithm can be significant.*

# IV–4 Application to parametric uncertainties

We first assess the performance of the proposed algorithms in the classical context of parametric uncertainties. In this section, uncertain parameters $\xi$ are assumed to lie in a known space $\Xi$ and values $\boldsymbol{q}(\boldsymbol{x},\xi)$ can be computed at any location $(\boldsymbol{x},\xi)$ in the coupled space. This context notably allows for constructing coupled-space surrogate models and using GP-based measure approximations in separated-spaces approaches.

## IV–4.1 Cost assessment

Similarly to Section III–4.5, several variants of the proposed approaches are quantitatively compared on some analytical test-cases. First, we make direct use of the algorithm depicted in Alg. IV.2, and refer to this strategy as the Joint SAMATA approach. Then, the Surrogate-Assisting model is replaced with the KDE-based surrogate model introduced in Section IV–3. This second approach draws disjoint realizations of $\mathbf{P}$ and makes use of Equation (IV.4) for comparing different designs. In the following, it is referred to as the Disjoint SAMATA approach. These techniques are also compared to SABBa, using formulations from Section III–4. Convergence is again regarded in terms of expected modified Hausdorff distance, as introduced in Section III–4.3.

**Remark** *Contrarily to the Joint SAMATA strategy, the Disjoint one can be performed with Separated-Spaces surrogate models, which is referred to as the SS Disjoint SAMATA strategy. As with Bounding-Boxes, one may expect the Separated-Spaces approach to be less cost-efficient than the Coupled-Space one in general.*

In summary, the compared approaches are the following:

- Joint-SAMATA-CS: Sampling-based SAMATA approach with Coupled-Space surrogate model and joint realizations.
- Disjoint-SAMATA-CS: Sampling-based SAMATA approach with Coupled-Space surrogate model and disjoint realizations.
- Disjoint-SAMATA-SS: Sampling-based SAMATA approach with Separated-Spaces surrogate models and disjoint realizations.
- SABBa-CS: SABBa with Coupled-Space surrogate model for Bounding-Box estimation.

- SABBa-SS: SABBa with Separated-Spaces surrogate models for Bounding-Box estimation.

The NSGA-II optimization algorithm is exploited in the following test-cases. In general , the number of individuals per generation is set to 32. Parameters for both SAMATA and SABBa are set to: $N_{init} = 32$, $N_{new} = 1$ (sequential optimizer), $N_{first} = 5$ and $N_{ref} = 1$ (sequential refinement). Recall that $N_{first}$ is only used for performing first measure approximations in Separated-Spaces techniques.

All Separated-Spaces algorithms are initialized by running $N_{first}$ evaluations of $\boldsymbol{q}$ on all $N_{init}$. Then, the number of individuals per NSGA-II iteration and $N_{init}$ are set to 16 when Separated-Spaces techniques are used, to lower the cost of the first iteration.

As for Coupled-Space algorithms, an initial DoE of 30 points is first computed in the Coupled Space before launching the optimization procedure. This allows for well-distributed information within the Coupled Space to initialize our adaptive strategies.

Each technique is launched ten times to get the mean and standard deviation of the convergence curves.

### IV–4.1.1 Unconstrained Taguchi optimization

The first test-case is the unconstrained Taguchi optimization problem tackled in Section III–4.5.1. This problem is bi-objective and features two design variables $\boldsymbol{x}$ and one uncertain parameter $\xi$.

The behavior of the Separated-Spaces strategies is represented in Figure IV.9. The SAMATA strategy shows quicker convergence than SABBa, which corroborates our assumption that sampling-based approximations are more representative than Bounding-Boxes and allow for a more efficient optimization procedure. Note that because the Surrogate-Assisting strategy is extensively exploited in the context of Separated Spaces, the gap between the two convergence curves is not as important as one may have expected.

**Remark** *Here, 80 evaluations of $\boldsymbol{q}$ are performed ($N_{init} \times N_{first}$) instead of 50 in the previous chapter, which explains the difference between Fig. IV.9 and Fig. III.26 for the SABBa-SS strategy.*

The performance of the Coupled-Space strategies is then depicted in Figure IV.10. They all show much quicker convergence than the Separated-Spaces approaches from Figure IV.9. The Joint and Disjoint characteristics seem not to have a very significant impact overall, but one can remark that the Joint approach has a steeper convergence curve than the Disjoint one. This behavior can be explained by the POP computation, that yields more accurate values when considering Joint realizations. However, as long as the measures are only roughly approximated, the Joint POP computation emphasizes the model error while the Disjoint assumption smoothen the POP values. This explains why the Disjoint approach yields better results when the number of evaluations is low.

### IV–4.1.2 Quantile-constrained optimization

These approaches are also compared on the quantile-constrained optimization problem depicted in Section III–4.5.2. This mono-objective problem features a quantile 95% constraint and possesses three design parameters and four uncertain variables.

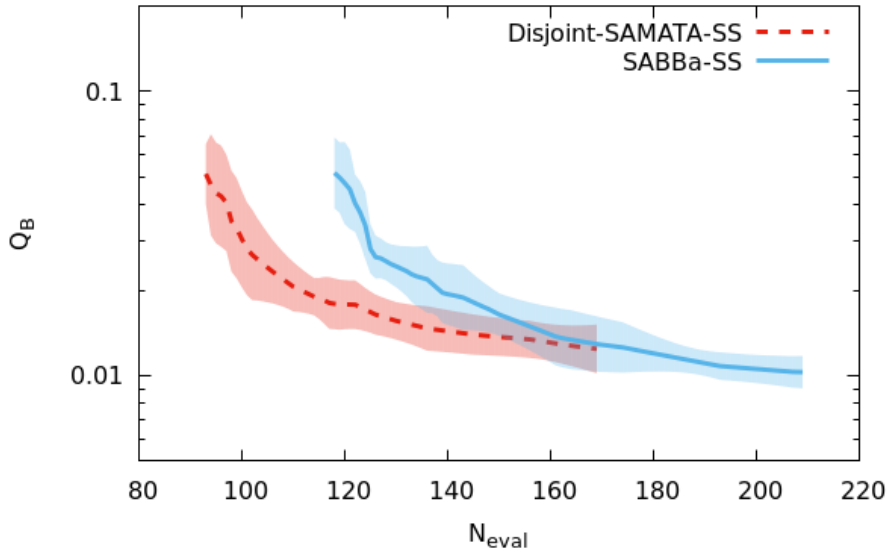



Figure IV.9 Comparison of SABBa and sampling-based SAMATA in the context of Separated-Spaces surrogate models.

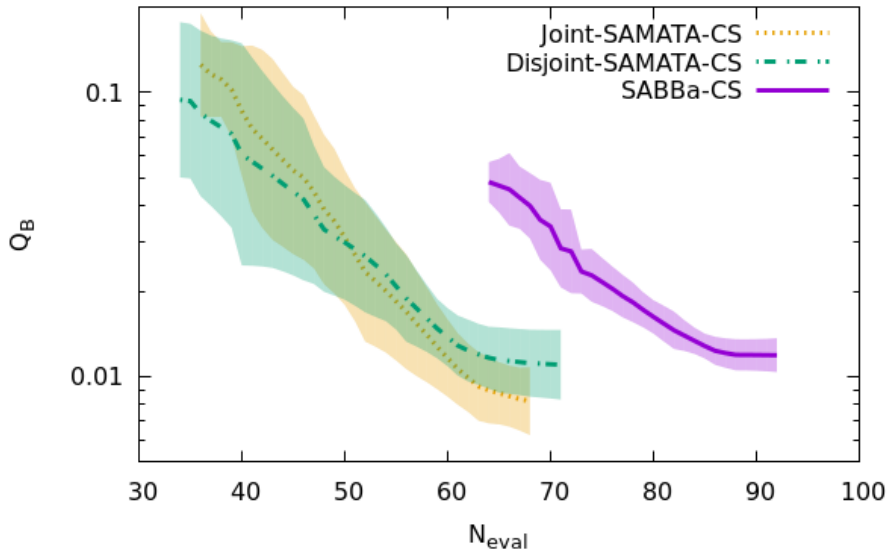



Figure IV.10 Comparison of SABBa and sampling-based SAMATA in the context of Coupled-Space surrogate model. The Joint and Disjoint SAMATA variants are both represented.

For their superior behavior in the context of low-dimensional parametric uncertainties, only the Coupled-Space approaches are compared here.

The convergence curves associated with the Joint SAMATA, Disjoint SAMATA and SABBa strategies are depicted in Figure IV.11. Similarly to Figure III.30, the convergence is slower than the previous test-case. However, also Figure III.30 only shows a small improvement for SABBa with respect to the A Priori MetaModel strategy, by reaching the 0.1 value around 150 evaluations, both SAMATA approaches show a significant improvement compared to SABBa. Using NSGA-II instead of NOMAD has allowed SABBa to reach the 0.1 indicator value around 120 evaluations. As for the

Joint and Disjoint SAMATA techniques, they can achieve this value nearly two times faster. In this case, Joint SAMATA seems to yield slightly better performance than Disjoint SAMATA consistently.

**Remark** *Recall that the plateau at the end of these convergence curves (for example at the end of Joint and Disjoint SAMATA) are mostly plotting artifacts that arise when one run out of the 10 is significantly longer.*

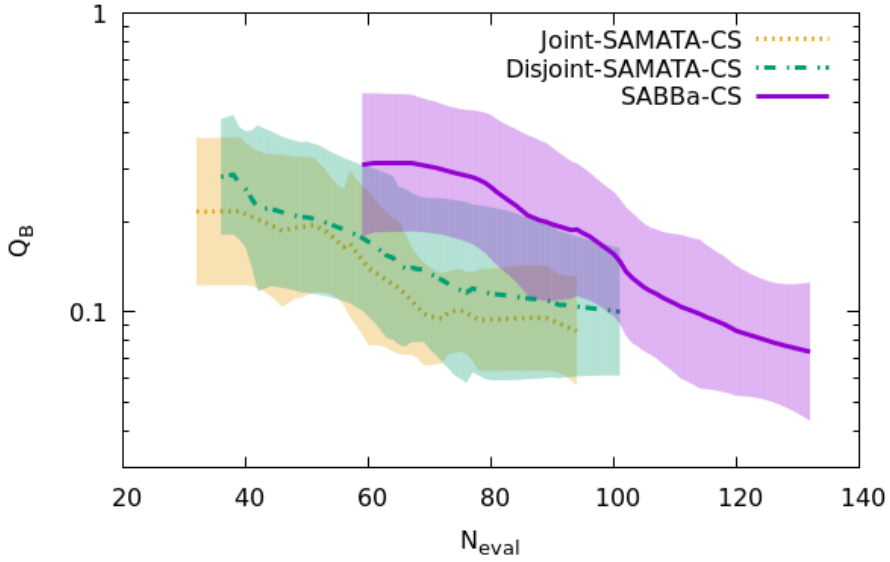



Figure IV.11 Comparison of SABBa and sampling-based SAMATA in the context of Coupled-Space surrogate model. The Joint and Disjoint SAMATA variants are both represented.

## IV–4.2 Benefit of sampling-based approximations

Several works in the literature perform Optimization Under Uncertainty with Gaussian measure approximations with the so-called projected processes [Da Veiga and Delbos, 2013, Baudoui, 2012, Janusevskis and Le Riche, 2013, Williams et al., 2000], which permits to use a Bayesian Optimization setting.

We investigate here the benefits of performing sampling-based measure approximations compared to a more classical Gaussian approach. Intuitively, sampling-based measure approximation permits to grasp better the distribution of non-Gaussian statistics such as variance or extreme quantiles. Besides, it allows to recover dependencies between these statistics, *i.e.* between quantiles and expectation. On the other side, Gaussian approximations are cheaper to estimate and give an analytical formulation for the measures distribution shape. The expectation measure is particularly adapted to this Gaussian approximation and only requires to compute the first two moments through kernel integration. For deeper insights, see [Baudoui, 2012].

We carry out the optimizations using the Disjoint SAMATA variant depicted in Algorithm IV.2. To focus the comparison on the distribution shapes, and to avoid adding noise, the SA strategy is not employed. This strategy is simply achieved in practice by

setting $\boldsymbol{s}_1$ to $\mathbf{0}$ throughout the optimization process. We then compare this sampling-based SAMATA strategy to a Gaussianized approach, where each computed set of samples is replaced with a set of *iid* samples coming from a decorrelated normal distribution. The first two moments are kept constant for resampling the measure realizations. Practically, at any design $\boldsymbol{x}$, the empirical mean $\boldsymbol{\mu}_{\mathbf{P}}(\boldsymbol{x})$ and empirical standard deviation $\boldsymbol{\sigma}_{\mathbf{P}}(\boldsymbol{x})$ are computed from the set of samples $\left\{\mathbf{P}^{(i)}(\boldsymbol{x})\right\}_i$, and new Gaussianized samples are drawn instead from $\mathcal{N}\big(\boldsymbol{\mu}_{\mathbf{P}}(\boldsymbol{x}), \boldsymbol{\sigma}_{\mathbf{P}}(\boldsymbol{x})\big)$.

The two variants are compared on the unconstrained Taguchi optimization problem. The full optimization process is performed 30 times for each approach, and the mean convergence curves are depicted in Figure IV.12, with its Bollinger band $\pm\sigma$ variability.

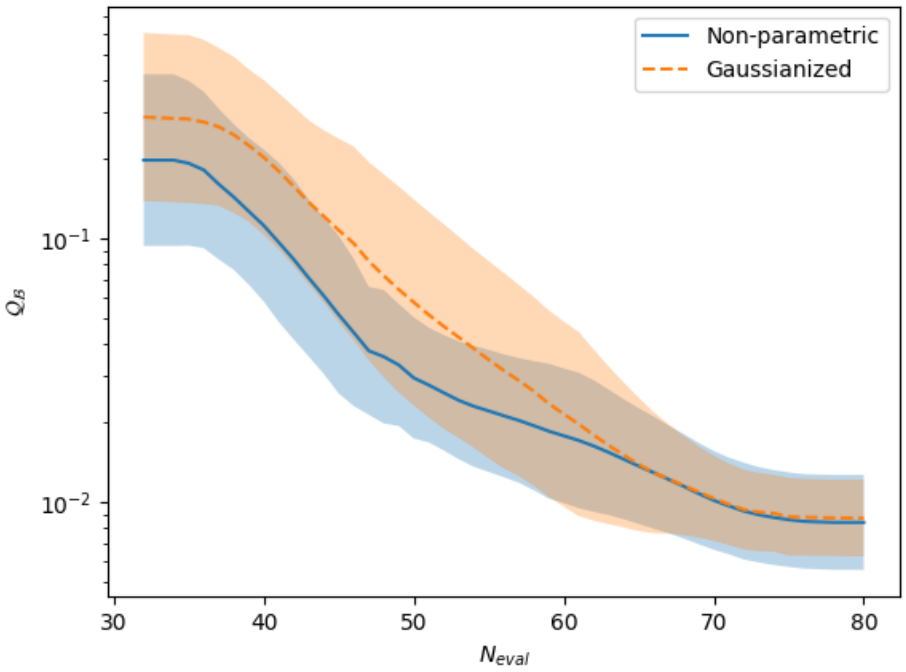



Figure IV.12 Convergence curves of the non-parametric SAMATA strategy against its Gaussianized equivalent. Variability represented with the transluscent $\pm\sigma$ band.

Although the final convergence is similar between the two approaches, the Gaussianized approach shows slightly inferior accuracy compared to the original non-parametric strategy. Intuitively, accurate distribution shapes and dependencies are lost when Gaussian density is forced, which deteriorate the accuracy of the Pareto optimal set. However, when enough evaluations are performed, and the $\mathbf{P}$ distributions on the Pareto front are very refined, the non-parametric efficiency becomes similar to the Gaussian one.

In this test-case, non-parametric measure representations seem to have a relevant effect on the convergence speed at low and medium accuracy, thus providing accurate Pareto front approximations more quickly than its Gaussian counterpart.

**Discussion** *The non-parametric SAMATA strategy presented in Sections IV–1, IV–2 and IV–3 has been applied to two Optimization Under Uncertainty problems tackled in Chapter III. Several key conclusions can be drawn from these analytical test-cases. Convergence speed seems to be improved through the use of non-parametric measure approximations compared to fixed distribution shape such as Gaussians. This trend is notably true at early to mid-stages of the optimization process when the measure still features a substantial variability.*
*Also, by being more representative and avoiding over-conservativeness, the non-parametric SAMATA approaches show much quicker convergence toward the Pareto optima than SABBa, that is based on Bounding-Boxes. This trend holds both in the setting of Separated-Spaces and Coupled-Space surrogate models.*
*Finally, the use of KDE-based SA model for computing Pareto Optimal Probabilities in a disjoint setting permits to smooth the optimal area at the beginning of the optimization process. Eventually, it gets outperformed by the joint approach, that shows a better convergence slope. Nevertheless, these techniques show comparable overall performance.*

## IV–5 Application to non-parametric uncertainties

We finally study the performance of the proposed strategy in the case of non-parametric uncertainties. In this context, $\boldsymbol{q}$ are not regarded in the coupled space $\mathcal{X}\times\Xi$ and values $\boldsymbol{q}(\boldsymbol{x},\xi)$ are not directly accessible. Instead, only realizations $\boldsymbol{q}(\boldsymbol{x},\xi^{(i)})$ can be drawn, with no information on $\xi^{(i)}$. To highlight this behavior, in the following of this section, the function at a given design $\boldsymbol{x}$ is denoted as $\boldsymbol{Q}(\boldsymbol{x})$ and only samples $\boldsymbol{Q}^{(i)}(\boldsymbol{x})$ can be drawn.

In this setting, the Joint SAMATA using Algorithm IV.2 is not applicable, because the coupled-space GP surrogate model requires parametric uncertainties. Hence, in this section, we make use of the KDE-based surrogate model to speed-up the optimization process on disjoint sampling-based measure approximations.

**Remark** *Because surrogate model construction is impractical when dimensionality is very large, [Sabater et al., 2020] exploits the non-parametric uncertainties setting and proposes an algorithm for Optimization Under Uncertainty which is insensible to the problem dimension.*

### IV–5.1 Adjustment of SAMATA

SAMATA can be applied in the presence of non-parametric uncertainties through the use of the KDE-based Surrogate-Assisting (SA) model and a technique for performing the sampling-based Measure Approximation (MA). Note that the refinement criterion defined in Section III–4.1.1 cannot be applied in this context. We thus propose to use a bootstrap strategy for sampling realizations of the robustness and reliability measures and to refine the current approximations through the computation of random additional samples.

The bootstrap sampling strategy is presented in Section IV–5.1.1, and is then incorporated in the global algorithm, which is given in Section IV–5.1.2. The results are finally pictured on an analytical test-case in Section IV–5.2

### IV–5.1.1 Bootstrap measure sampling

Function $\boldsymbol{q}$ at a design $\boldsymbol{x}$ is regarded as a random value $\boldsymbol{Q}(\boldsymbol{x})$, that can be sampled at a given cost. These samples are denoted by $\boldsymbol{Q}^{(i)}(\boldsymbol{x})$.

The classic bootstrap scheme permits to estimate the distribution of a statistic regardless of the sample's distribution. Although approximated, this approach extends the classical theory to the non-parametric world.

In practice, given a set of samples $\mathcal{Q}(\boldsymbol{x}) = \left\{\boldsymbol{Q}^{(i)}(\boldsymbol{x})\right\}_i^{N_{samples}}$ at location $\boldsymbol{x}$, the statistics of interest (robustness and reliability measures) can be empirically estimated as $\widetilde{\boldsymbol{\rho}}(\boldsymbol{x}) = H\big[\mathcal{Q}(\boldsymbol{x})\big]$, where $H$ can refer *e.g.* to an expectation, a variance or a quantile. The bootstrap method then resamples from the set $\mathcal{Q}(\boldsymbol{x})$ with replacement. Each element of a given bootstrap resampling $\mathcal{Q}_{bs}^{(k)}(\boldsymbol{x})$ is an element $\boldsymbol{Q}^{(I)}(\boldsymbol{x})$ of $\mathcal{Q}(\boldsymbol{x})$ with $I \sim \mathcal{U}\big([\![1, N_{samples}]\!]\big)$. The set of statistics $\left\{\mathbf{P}^{(k)}(\boldsymbol{x})\right\}_k = \left\{H\big[\mathcal{Q}_{bs}^{(k)}(\boldsymbol{x})\big]\right\}_k$ can then be considered as *iid* realizations of the robustness and reliability measures.

**Remark** *Although bootstrap may appear as a rather naive, it can be regarded as resampling $\boldsymbol{Q}$ from its empirical Cumulative Density Function (CDF), which is the best least-square approximation of the underlying CDF based on empirical samples.*

By adding a small deviation around each sampled value, the so-called smoothed bootstrap permits to smooth out the estimated distribution $\mathbf{P}$.

Numerically, this corresponds to drawing realizations from a Kernel Density Estimate (KDE) of $\boldsymbol{Q}$ during the bootstrap resampling step. Thus, the measure realizations $\mathbf{P}^{(k)}$ are here defined as statistics computed over a resampled set of values $H\big[\mathcal{Q}_{smooth}^{(k)}(\boldsymbol{x})\big]$, where each element of the resampled set $\mathcal{Q}_{smooth}^{(k)}(\boldsymbol{x})$ is a realization from the KDE density $\psi_{\boldsymbol{x}}^{KDE}$ constructed on samples $\mathcal{Q}(\boldsymbol{x})$.

For performing this smoothed bootstrap strategy, the Kernel Density Estimation is carried out similarly to Section IV–3.1, using Silverman's bandwidth.

**Remark** *Note that more intrusive methods could be used for the bootstrap computation or Surrogate-Assisted model computation. For example, measure dependent surrogate model could be used as the SA model, such as [Sabater et al., 2020] with quantile regression.*
*Here we chose to show that SAMATA adapts to non-parametric uncertainties and performs well, even with fundamental and non-intrusive techniques.*

### IV–5.1.2 Algorithm

Only a few adjustments are needed within the SAMATA algorithm for performing Optimization Under Uncertainty with non-parametric uncertainties. Namely, no coupled or separated spaces surrogate is constructed, and the smoothed bootstrap technique presented above is applied for drawing realizations of $\mathbf{P}$. Also, the refinement strategy is not criterion-based but instead simply generates a new realization of $\boldsymbol{q}$.

In practice, the SAMATA algorithm for non-parametric uncertainties is given in Alg. IV.3 and follows the general flowchart from Figure II.5:

**Algorithm IV.3** SAMATA algorithm for non-parametric uncertainties

Loop over values of $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$
**while** Optimization running **do**
  Get new designs $\mathcal{X}_{new}$
  Initialize $\mathcal{X}_c$ empty
  Compute $\mathbf{P}_{SA}$ for all $\boldsymbol{x} \in \mathcal{X}_{new}$ from (IV.26)
  Compute $\mathcal{X}_{SA}^0 = \left\{\boldsymbol{x} \in \mathcal{X}_{new} \mid \boldsymbol{\Delta}_{SA}(\boldsymbol{x}) \leq \boldsymbol{s}_1\right\}$ and $\mathcal{X}_c^0 = \mathcal{X}_{new} \setminus \mathcal{X}_{SA}^0$
  Deduce $\mathcal{X}_r^0$ from $\mathcal{X}_c^0$ using (II.26)
  $k = 0$
  **while** $\mathcal{X}_r^k \neq \emptyset$ **do**
    Compute $\forall \boldsymbol{x} \in \mathcal{X}_{new}, POP_{min}(\boldsymbol{x})$ w.r.t. all designs using (II.24) and (IV.2)
    Find $\boldsymbol{x}^*$ from $\mathcal{X}_r^k$ using (II.27)
    **if** $\boldsymbol{x}^* \notin \mathcal{X}_c$ **then**
      Add $\boldsymbol{x}^*$ in $\mathcal{X}_c$
      Compute $N_{first}$ realizations $\boldsymbol{Q}^{(i)}(\boldsymbol{x}^*)$
    **else**
      Compute $N_{ref}$ realizations $\boldsymbol{Q}^{(i)}(\boldsymbol{x}^*)$
    **end if**
    Draw bootstrap realizations $\mathbf{P}_c^{(i)}(\boldsymbol{x}^*)$ as presented in Section IV–5.1.1
    Update $\mathbf{P}_{SA}$ for all $\boldsymbol{x} \in \mathcal{X}_{new} \setminus \mathcal{X}_c$
    Compute $\mathcal{X}_c^{k+1}$, $\mathcal{X}_{SA}^{k+1}$ and $\mathcal{X}_r^{k+1}$
    $k = k + 1$
  **end while**
  Return $\mathbb{E}\left[\mathbf{P}(\boldsymbol{x})\right]$ to the optimizer for all $\boldsymbol{x} \in \mathcal{X}_{new}$
**end while**
Return all measure approximations $\mathbf{P}(\boldsymbol{x})$ and their associated POP

**Remark** *Similarly to all Separated-Spaces techniques, during the first iteration, the SA model cannot be computed, and thus, the $POP_{min}$ metric cannot be computed. We simply propose to add all these first designs to $\mathcal{X}_c$ and compute all the associated measures to initialize the approach.*

## IV–5.2 Analytical application

This bootstrap-based SAMATA technique is applied to two analytical test-cases in the following. First, the unconstrained Taguchi optimization presented at the beginning of Section IV–4 is solved by considering non-parametric uncertainties. Then SAMATA is applied to a simple mono-objective quantile optimization under non-parametric uncertainty that cannot be solved with fully Gaussian assumptions.

### IV–5.2.1 Taguchi optimization

The unconstrained Taguchi optimization from Section III–4.5.1 is solved here, considering the uncertain variable to be non-parametric. In practice, anytime $\boldsymbol{q}$ is computed at a design $\boldsymbol{x}$, a value $\boldsymbol{Q}^{(i)}(\boldsymbol{x}) = \boldsymbol{q}(\boldsymbol{x}, \xi^{(i)})$ is returned, with $\xi^{(i)}$ a realization of the random variable $\xi \sim \mathcal{U}([0,1])$.

This uncertainty-based optimization is solved using the bootstrap SAMATA algorithm given in Alg. IV.3 and the convergence results are gathered in Table IV.1. The five rows correspond to normalized thresholds of 50%, 40%, 30%, 20% and 10% re-

spectively, using notations from Equation (III.35). The final computational cost for reaching a 10% threshold is 56300 evaluations. These $Q_{\mathcal{B}}$ values can be compared with Figures from Section III–4.5.1, where $Q_{\mathcal{B}} \approx 10^0$ signifies very high disagreement with the true Pareto front while $Q_{\mathcal{B}} \approx 10^{-2}$ reflects very good convergence. The proposed approach already gives acceptable results for 3900 evaluations, with $Q_{\mathcal{B}} < 10^{-1}$, and eventually converges toward the exact Pareto optimal area.

| $N_{eval}$ | $Q_{\mathcal{B}}$ |
|---|---|
| 3900 | $8.115 \times 10^{-2}$ |
| 6400 | $4.184 \times 10^{-2}$ |
| 9850 | $3.160 \times 10^{-2}$ |
| 17800 | $2.649 \times 10^{-2}$ |
| 56300 | $1.288 \times 10^{-2}$ |

Table IV.1 Convergence results for the bootstrap SAMATA approach.

More classically, this uncertainty-based optimization problem could be solved using the bootstrap technique within a Double-Loop approach, introduced in the previous chapters. With a final measure accuracy of 10%, we can make a rough estimate of the number of function evaluations required for solving this test-case.

As an optimistic assumption, we only consider the first objective, which is an expectation. From normal theory, a Gaussian approximation for the measure distribution should be close to the truth. We consider that $\sim 200$ individuals are visited during the optimization process, and approximate the cost of one measure approximation. Normal theory gives us that $\sigma \approx \dfrac{\sigma_q}{\sqrt{N}}$, with $\sigma$ the standard deviation of the expectation operator. Target accuracy is reached when $3\sigma \leq 10\%$ of the range $l_\mu$ of the robustness measure. Here $l_\mu \approx 0.35$, hence we aim for $\sigma \leq 0.012$. Finally, because $\sigma_q \approx 0.7$, the number of evaluations can be estimated as $\left(\dfrac{0.7}{0.012}\right)^2 \approx 4000$. Multiplied by 200 optimization iterations, this represents nearly $10^6$ evaluations of $\boldsymbol{q}$. The Double-Loop approach reveals much less cost-efficient than the proposed SAMATA strategy.

**Remark** *Note that the rough approximation carried out for the Double Loop computational cost is verified in practice with the most refined individuals in SAMATA requiring up to 5000 evaluations for reaching the target* 10% *accuracy.*

### IV–5.2.2 Quantile optimization

A second test-case is proposed here in order to show the resilience of the proposed approach to heavily non-Gaussian distributions. To this extent, a stochastic Quantity of Interest $Q(x)$ is constructed with a constant variance over the design space. Practically:

$$\begin{aligned} & Q(x) = -10\sin(9x) - x + Z(x) \\ \text{with} \quad & \phi_Z(z \mid x) = \frac{1}{2}\Big(\phi_{Gauss}\big(z, -\mu(x), \sigma(x)\big) + \phi_{Gauss}\big(z, \mu(x), \sigma(x)\big)\Big). \end{aligned} \tag{IV.39}$$

The random additive bias $Z(x)$ thus follows a centered Gaussian mixture with two components. The statistics are initialized at $\mu(0) = 2$ and $\sigma(0) = 0.5$, which gives an overall variance of $0.5^2 + 2^2 = \sigma^2_{stoch}$. The standard deviation $\sigma(x)$ is then defined as

follows:

$$\sigma(x) = 0.5 + (\sigma_{stoch} - 0.5)t(x) \tag{IV.40}$$

$$\text{with} \quad t(x) = \begin{cases} 0 & \text{if } x \leq 0.3 \\ 2x - 0.6 & \text{if } 0.3 \leq x \leq 0.8 \\ 1 & \text{if } x \geq 0.8 \end{cases}.$$

The standard deviation associated to the random additive bias $Z(x)$ thus goes from $\sigma(0) = 0.5$ to $\sigma(1) = \sigma^2_{stoch}$. The function $\mu(x)$ is then specifically constructed so that the variance of $Q(x)$ remains constant for all $x$. Using the variance formula for Gaussian mixture, and fixing the standard deviation, to $\sigma_{stoch}$, we have $\sigma_{stoch} = \sqrt{\sigma^2(x) + \mu^2(x)}$. This gives the mean values:

$$\mu(x) = \sqrt{\sigma^2_{stoch} - \sigma^2(x)}. \tag{IV.41}$$

Figure IV.13 gives a visual representation of the stochastic Quantity of Interest. It depicts the Probability Density Function (PDF) of $Q(x)$ as a color map at each location $x \in [0, 1]$.

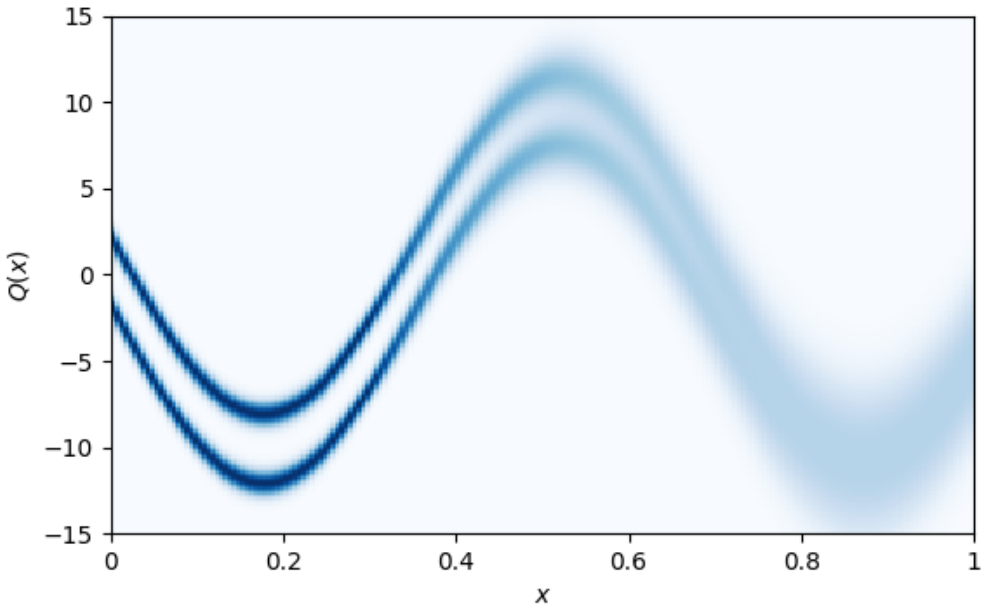



Figure IV.13 PDF of the random field $Q(x)$.

The uncertainty-based optimization problem reads:

$$\begin{aligned} \text{minimize:} \quad & \rho_f(x) = \mathrm{q}^{0.99}\big[Q(x)\big] \\ \text{by changing:} \quad & x \in [0, 1]. \end{aligned} \tag{IV.42}$$

Using the constant variance $\sigma_{stoch}$ over the design space, the random process $Q(x)$ can be readily approximated with a Gaussian Process, for which the quantile value $\rho_f(x)$ is analytical. The Gaussian 99% quantile measure approximation reads:

$$\mathrm{q}^{0.99}_{Gauss}(\boldsymbol{x}) = -10\sin(9x) - x + 2.326\sigma_{stoch}. \tag{IV.43}$$

This Gaussian quantile approximation is plotted as a green dotted curve in Figure IV.14 over the stochastic function $Q(x)$. The associated minimum is represented as a green dot at $x = 0.874$. However, the real quantile function is drawn as a red dashed curve with minimum at $x = 0.17577$.

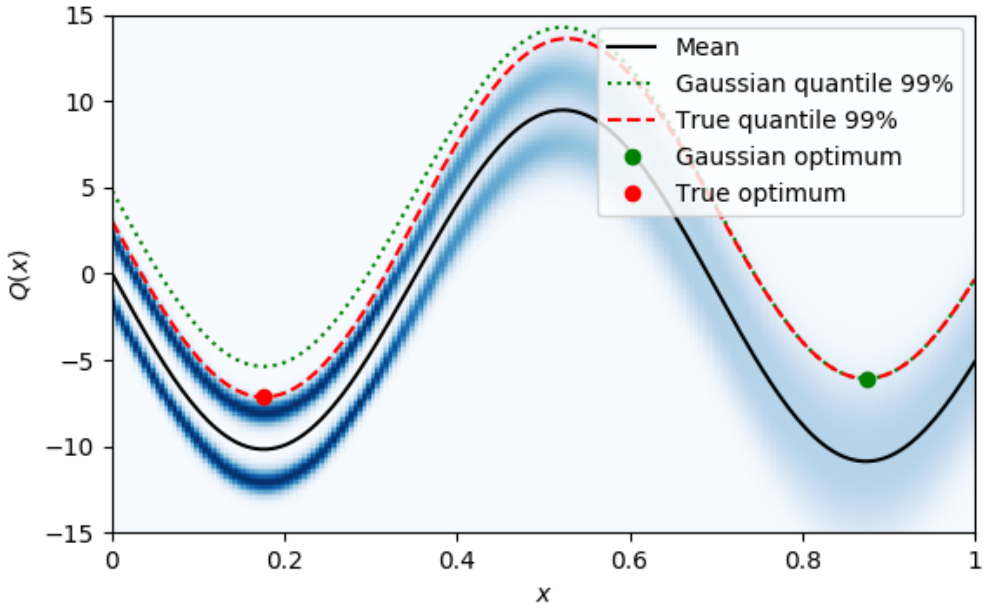



Figure IV.14 True and approximated quantiles of the stochastic process.

As for the previous example, Table IV.2 gives the evolution of the quantitative indicator with respect to the number of evaluations. SAMATA is applied with four iterative threshold values of 20%, 10%, 5% and 3%. One can see that the true optimum area around $x = 0.17577$ maximizes the POP at the end of the second iteration, at the cost of 680 evaluations.

The Pareto Optimal Probabilities associated with the visited designs at each iteration are plotted in Figure IV.15. The Pareto Optimal Probability (POP) of the true optimum takes over the Gaussian optimum during the second SAMATA iteration and following refinements further ensure that the optimum lies at $x = 0.17577$.

| $N_{eval}$ | $Q_{\mathcal{B}}$ |
|---|---|
| 340 | $6.999 \times 10^{-1}$ |
| 680 | $1.772 \times 10^{-2}$ |
| 1180 | $7.638 \times 10^{-3}$ |
| 1640 | $5.893 \times 10^{-3}$ |

Table IV.2 Convergence values of bootstrap SAMATA applied to problem (IV.42).

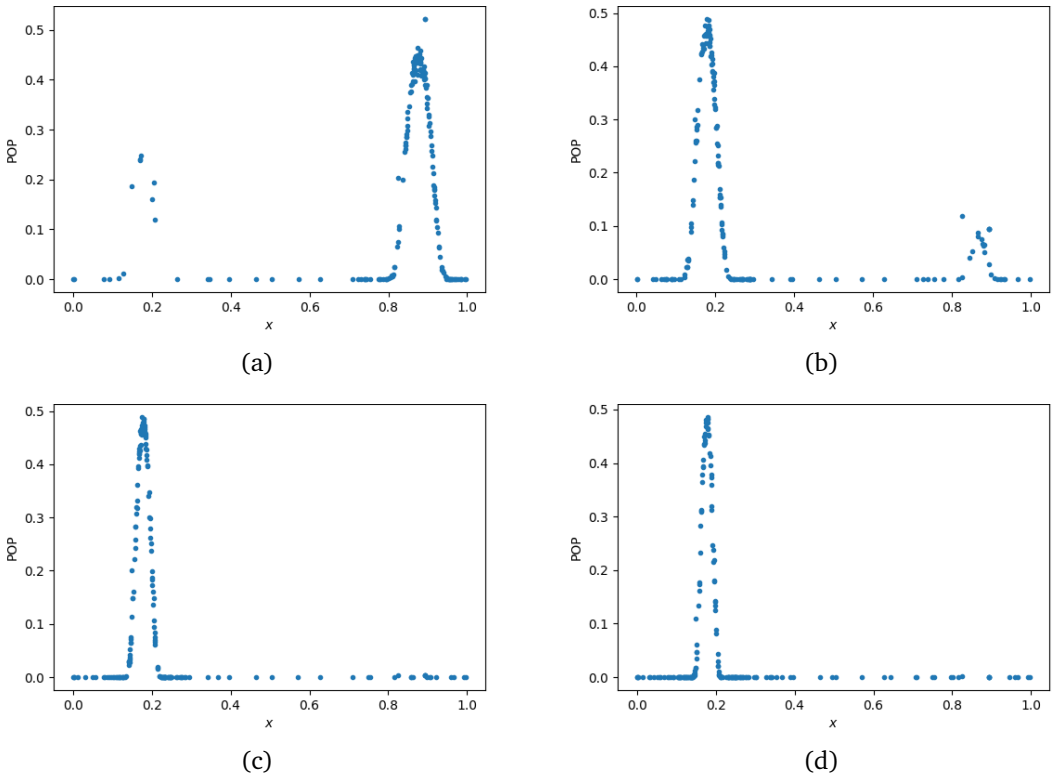



Figure IV.15 Pareto Optimal Probability of all visited designs associated with (a) 20%, (b) 10%, (c) 5% and (d) 3% accuracy.

**Discussion** *In this section, SAMATA has been applied to a special setting for Optimization Under Uncertainty. Contrarily to the previous test-cases, we consider non-parametric uncertainties, meaning that no surrogate model can be constructed in the uncertain space. In this context, coupled-space GP techniques are unfeasible, and the Surrogate-Assisting strategy is required to ensure an acceptable computational cost. Robustness and reliability measure realizations are drawn with a smoothed bootstrap strategy, and the KDE-based surrogate model from Section IV–3 is exploited as Surrogate-Assisting strategy. Indeed, we have illustrated in Section IV–4.2 the advantage of performing the sampling-based estimation of the robustness and reliability measures instead of making a Gaussian assumption.*
*This bootstrap strategy has been applied to two uncertainty-based optimization problems revealing excellent performance and featuring low computational cost. The absence of distribution assumption notably permits to tackle problems that could not be solved with classical GP-based approaches.*

**Conclusion of the chapter** *This chapter has proposed several tools for performing non-parametric sampling-based measure approximations. This setting and associated Pareto Optimal Probability computations have been introduced in Section IV–1. The computational impact of drawing joint or disjoint realizations has been highlighted and is quantitatively studied in the following sections.*
*Then, Sections IV–2 and IV–3 introduced two approaches for performing sampling-based Optimization Under Uncertainty. First, an adaptive strategy for drawing joint realizations of the robustness and reliability measure over numerous designs at the same time is presented. It relies on an approximation scheme for drawing large-scale Gaussian random field realizations, with controlled accuracy. Then a KDE-based surrogate model is introduced as an extension of the classical heteroscedastic Gaussian process, with the capability to take as input non-parametric density estimations.*
*These sampling-based algorithms are then quantitatively compared to SABBa from Chapter III, showing a significant improvement in the convergence speed, either with joint or disjoint approximations. The sampling-based context is also shown to sensibly improve the mean convergence speed compared to its Gaussian counterpart. Note, however, that the computational cost for computing realizations of the robustness and reliability measures is significantly higher than SABBa intervals. Thus, SABBa outperforms sampling-based SAMATA on test-cases where evaluations are inexpensive, and most of the computational burden comes from SAMATA itself.*
*Finally, SAMATA has been applied to uncertainty-based optimization problems with non-parametric uncertainties. Bootstrap-based realizations are drawn for creating the sampling-based measure approximations, and the KDE-based surrogate model is exploited for lowering the overall computational time. SAMATA yields much better convergence speed than a direct Double-Loop approach and exploits its non-parametric distribution shape to outperform Gaussian-based techniques.*

# Part C

# Applications

CHAPTER V

# Engineering applications

**Overview** *Several efficient formulations for performing Optimization Under Uncertainty have been proposed in Chapters III and IV. These techniques, respectively called SABBa and sampling-based SAMATA, both derive from the general SAMATA strategy proposed in Chapter II. In the previous chapter, sampling-based SAMATA has been shown more efficient than SABBa on small analytical test-cases.*
*In this chapter, we aim to apply these strategies to real-world engineering applications. To this extent, we propose to assess their performance on three test-cases of increasing complexity. SABBa and sampling-based SAMATA are compared on a first structural application where analytical expressions exist. Then, we apply SABBa to an aerospace optimization problem with medium dimensionality and smooth behavior of the simulation output $\boldsymbol{q}$. Finally, Joint SAMATA, which has proven very efficient in the preceding chapter, is exploited for solving a problem of shape Optimization Under Uncertainty in the context of Organic Ranking Cycle (ORC) turbines, with medium dimensionality and complex behavior of the function $\boldsymbol{q}$.*

**Outline** *In Section V–1, the first application is the famous two-bar truss structural problem. A simple uncertainty-based optimization problem is proposed based on [Baudoui, 2012] and [Jin et al., 2003], with four design variable and two uncertain parameters. Formally, the overall volume is minimized under two uncertainty-based constraints on the stress. The chosen statistics for both constraints is 95% quantile. This test-case is solved with both SABBa and Joint SAMATA to compare their behavior and verify their convergence.*
*Then, the Thermal Protection System (TPS) of a spacecraft is optimized in Section V–2 under material, chemical and geometrical uncertainties. These uncertain parameters have been shown in the literature to have a significant influence on the Quantities of Interest. Practically, we aim to minimize the mean mass of the TPS under a worst-case constraint on the bottom temperature. This problem features two design parameters, twelve uncertain dimensions, and a pretty smooth input-output relationship. In this context, i.e. low-dimensional design space and smooth functions, the Surrogate-Assisting strategy yields a significant cost reduction. Thus, SABBa can be employed for solving this test-case with a reasonable budget.*
*Finally, in Section V–3, we optimize the shape of an ORC turbine to minimize the drag under uncertain pressure and temperature operating conditions. This application is computationally intensive and features nine design parameters and three uncertainties. To cope with the high-dimensionality of the design space and the complex behavior of the simulation code, we exploit the Joint SAMATA strategy, that allows for very parsimonious mono- and bi-objective optimizations compared to state-of-the-art approaches.*

**Contribution** *The TPS optimization is an original contribution of this work. Uncertainty Propagation and Sensitivity Analysis have been presented in [Rivier et al., 2019]. However, to the authors' knowledge, no uncertainty-based optimization has ever been carried out in the literature considering uncertain pyrolysis parameters. The shape optimization problem for the ORC turbine has been solved in [Razaaly, 2019]. We show here that the use of joint sampling-based measure approximations provides a gain of at least one order of magnitude in terms of computational cost. A Taguchi formulation is also proposed to study the computational cost of a bi-objective formulation.*

## V–1 Two bar truss structure

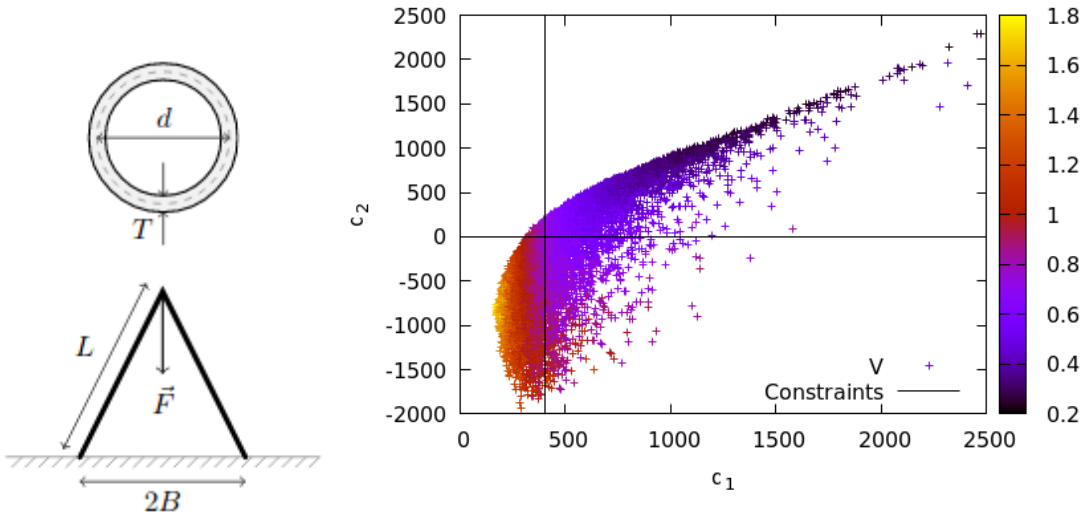
The two-bar truss optimization problem is notably illustrated in Refs. [Baudoui, 2012] and [Jin et al., 2003]. A simplified uncertainty-based optimization problem is derived and formulated as follows (with a schematic representation in Figure V.1(a)):

$$\begin{aligned}
\text{minimize:} \quad & \boldsymbol{\rho}_f(\boldsymbol{x}) = V(\boldsymbol{x}) \\
\text{satisfying:} \quad & \boldsymbol{\rho}_g(\boldsymbol{x}) = \begin{pmatrix} c_1(\boldsymbol{x}) \\ c_2(\boldsymbol{x}) \end{pmatrix} \leq \begin{pmatrix} s_{\max} \\ 0 \end{pmatrix} \\
\text{where:} \quad & c_1(\boldsymbol{x}) = \mathrm{q}^{0.95}[s(\boldsymbol{x},\xi)] \\
& c_2(\boldsymbol{x}) = \mathrm{q}^{0.95}[s(\boldsymbol{x},\xi) - s_{crit}(\boldsymbol{x},\xi)] \\
\text{with:} \quad & \xi \sim \mathcal{N}\left( \begin{pmatrix} 150000 \\ 210000 \end{pmatrix}, \begin{pmatrix} 30000^2 & 0 \\ 0 & 21000^2 \end{pmatrix} \right) \\
\text{by changing:} \quad & \boldsymbol{x} \in [20,80] \times [800,1200] \times [700,800] \times [2,3]
\end{aligned} \tag{V.1}$$

In the above, the design variables are: $(x_1)$ the diameter of the cross section $d$, $(x_2)$ the bar length $L$, $(x_3)$ the structure half-width $B$ and $(x_4)$ the thickness of the cross section $T$. The uncertain parameters refer to $(\xi_1)$ the external force $F$ and $(\xi_2)$ the elastic modulus $E$. The objective is to minimize the total volume $V(\boldsymbol{x}) = 2\pi x_1 x_2 x_4 \times 10^{-6}$ while verifying, with the 95% quantile constraints, that the probability of the stress $s(\boldsymbol{x},\xi) = \dfrac{x_2 \xi_1}{2\pi x_1 x_4 \sqrt{x_2^2 - x_3^2}}$ exceeding $s_{\max} = 400$ N.mm$^{-2}$ and $s_{crit}(\boldsymbol{x},\xi) = \dfrac{\pi^2 \xi_2 (x_1^2 + x_4^2)}{8 x_2^2}$ are both below 0.05.

The problem is represented in Fig. V.1(b), where the objective value $V(\boldsymbol{x})$ to be minimized is plotted in the constraints space with a color scale. The abscissa and ordinate directions refer to the constraints $c_1(\boldsymbol{x})$ and $c_2(\boldsymbol{x})$ ; the associated thresholds are drawn with red lines. The admissible set is in the lower left quadrangle ($c_1 < 400$ N.mm$^{-2}$ and $c_2 < 0$ N.mm$^{-2}$) and one may note that the objective is nearly constant on the limit $c_1 = 400$ N.mm$^{-2}$, with optimum value at $(c_1, c_2) = (400, 0)$.

This optimization problem is solved both with the SA-CS variant of SABBa presented in Section III–4.5 and the Joint SAMATA variant of the sampling-based SAMATA strategy introduced in Section IV–2. The thresholds $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ are sequentially reduced up to 0.1% of the total ranges, with four steps 3%, 1%, 0.3% and 0.1%. Optimization is again performed with the NSGA-II technique.



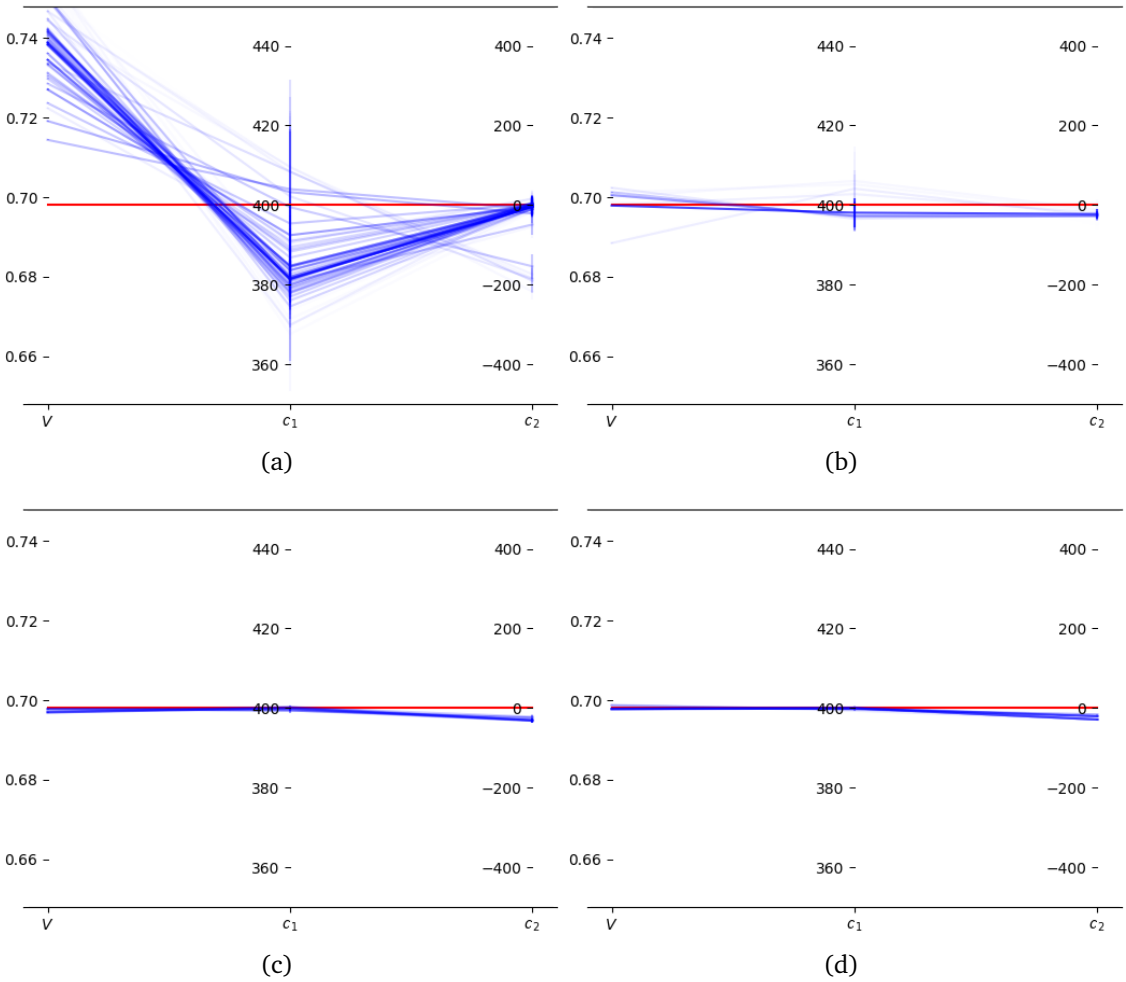

(a) (b)

Figure V.1 (a) Schematic representation of the two bars, with bar section above (from [Baudoui, 2012]). (b) Representation in the objective/constraints space, constraints in abscissa and ordinate, objective in color.

SABBa results are depicted in Figure V.2 with parallel coordinates plots. The four thresholds are sequentially reached with a total number of **42**, **44**, **46** and **52** function evaluations, and an optimal objective value of roughly 0.698 is found. Uniform approximation errors from SABBa are represented as vertical intervals for each design with non-zero POP. Note that the second constraint $c_2$ is slightly off its optimal value 0 because the objective value $V$ is nearly constant w.r.t. $c_2$, as can be observed in Figure V.1(b).

Results from the Joint SAMATA strategy are then given in Figure V.3. Vertical intervals are replaced here with sample sets. In this case, threshold values are reached with only **31**, **35**, **39** and **49** evaluations. Once again, $c_2$ is not as accurately optimized as $c_1$ because of its lesser impact on objective $V$.

**Discussion** *On this analytical engineering test-case, Joint SAMATA reveals again more parsimonious than SABBa. This difference is surely driven by the representativeness of the sets of samples compared to the over-conservative intervals from SABBa. However, regardless of the SAMATA variant, uncertainty-based optimization with 95% quantiles has been solved at a very low computational cost (∼ 30 to 50 evaluations).*
*Because of the superior parsimony of Joint SAMATA, we propose to solve a simulation-based OUU problem with medium complexity using SABBa in the next section (V–2), and to perform a higher-complexity uncertainty-based optimization with Joint SAMATA in Section V–3.*




(a) (b) (c) (d)

Figure V.2 Objective and constraints values with (a) **42**, (b) **44**, (c) **46** and (d) **52** evaluations. Transparency driven by POP value and optimal value in red.

## V–2 Optimization of a Thermal Protection System for atmospheric reentry

We tackle here the design of a Thermal Protection System (TPS) for a reentry vehicle. This test-case deals with twelve dimensions and aims at minimizing the mean mass density under worst-case temperature constraint. Because this test-case is only of medium complexity and dimensionality, we choose to solve this uncertainty-based optimization problem with SABBa.

We study the reentry of Stardust, the first mission using a low-density carbon-phenolic ablator in 2006. Stardust was the fastest human-made object reentering the earth atmosphere, at a velocity of 12.7 $km/s$. A generic heat-shield for atmospheric reentry is depicted in Figure V.4.

Here, surface total pressure and heat flux are computed with hypersonic computational fluid dynamics (CFD) simulations. Following the state-of-the-art design approach, we assume that the problem is locally mono-dimensional. The analysis is performed using the properties of the Theoretical Ablative Composite for Open Testing (TACOT), which composition and properties are comparable to the Phenolic-Impregnated Carbon Ablator (PICA), pictured in Figure V.5. Nominal TACOT prop-

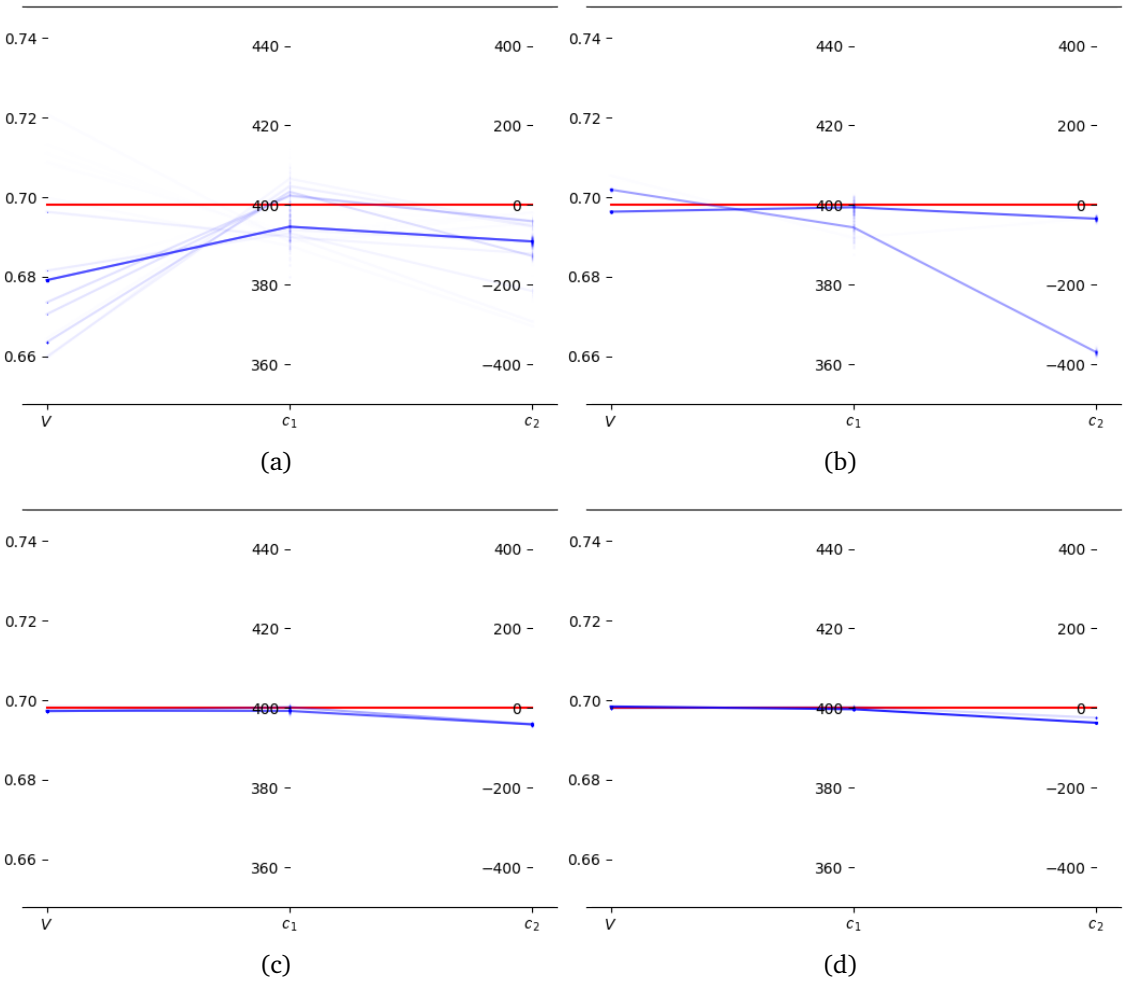



Figure V.3 Objective and constraints values with (a) **31**, (b) **35**, (c) **39** and (d) **49** evaluations. Transparency driven by POP value and optimal value in red.

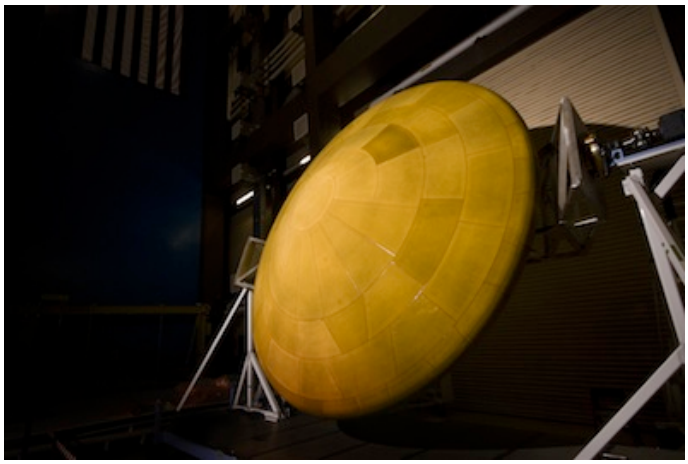


Figure V.4 Thermal Protection System (TPS) of the Mars Science Laboratory (MSL) for atmospheric reentry.

erties are available in the open literature. Volume-wise, TACOT is made of 10% of carbon fibers, 10% of phenolic resin, and is 80% porous. The thickness of the ablative material is two inches, and adiabatic conditions are used at the wall.

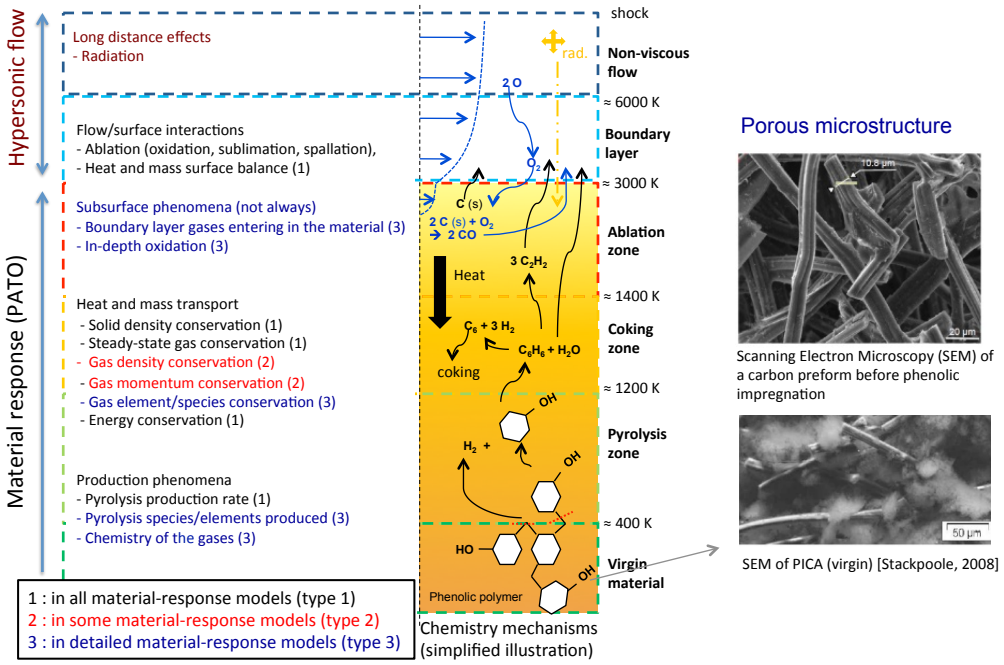



Figure V.5 PICA material, pyrolysis phenomenon. Picture from [Lachaud and Mansour, 2014].

The physical model used here is a generic heat and mass transfer model for porous media [Lachaud and Mansour, 2014]. The model is implemented in the Porous material Analysis Toolbox (PATO), distributed Open Source. First-order implicit finite-volume schemes in time and space were used for the simulations. The 1D problem is solved on 300 finite-volume cells with a logarithmic refinement of parameter 0.2 towards the surface. In this study, we use an equilibrium chemistry model and study the material response at the stagnation point, which reaches the highest temperature during the reentry. Figure V.6 illustrates the temporal evolution of temperature inside the material (dashed lines) and at the heated surface (plain red line).

Formally, the optimization problems reads as follows:

$$\begin{aligned}
\text{minimize:} \quad & \boldsymbol{\rho}_f(\boldsymbol{x}) = \mu(\boldsymbol{x}) \\
\text{satisfying:} \quad & \boldsymbol{\rho}_g(\boldsymbol{x}) = M(\boldsymbol{x}) \leq 473.15 \\
\text{where:} \quad & \mu(\boldsymbol{x}) = \mathbb{E}_\xi[\sigma(\boldsymbol{x}, \xi)] \\
& M(\boldsymbol{x}) = \max_\xi[T_b(\boldsymbol{x}, \xi)] \\
\text{with:} \quad & \xi \sim \mathcal{U}(\Xi) \\
\text{by changing:} \quad & \boldsymbol{x} \in [0.01, 0.1] \times [3.5, 7]
\end{aligned} \tag{V.2}$$

with $x_1$ the resin volume fraction (originally 10%) an $x_2$ the overall width of the TPS (originally 7.21 cm). We observe in Figure V.6 that with these initial parameters, the bottom temperature stays merely constant during the whole reentry. Hence, the search space is centered on lower $x_1$ and $x_2$ values, which tends to increase the bottom temperature.

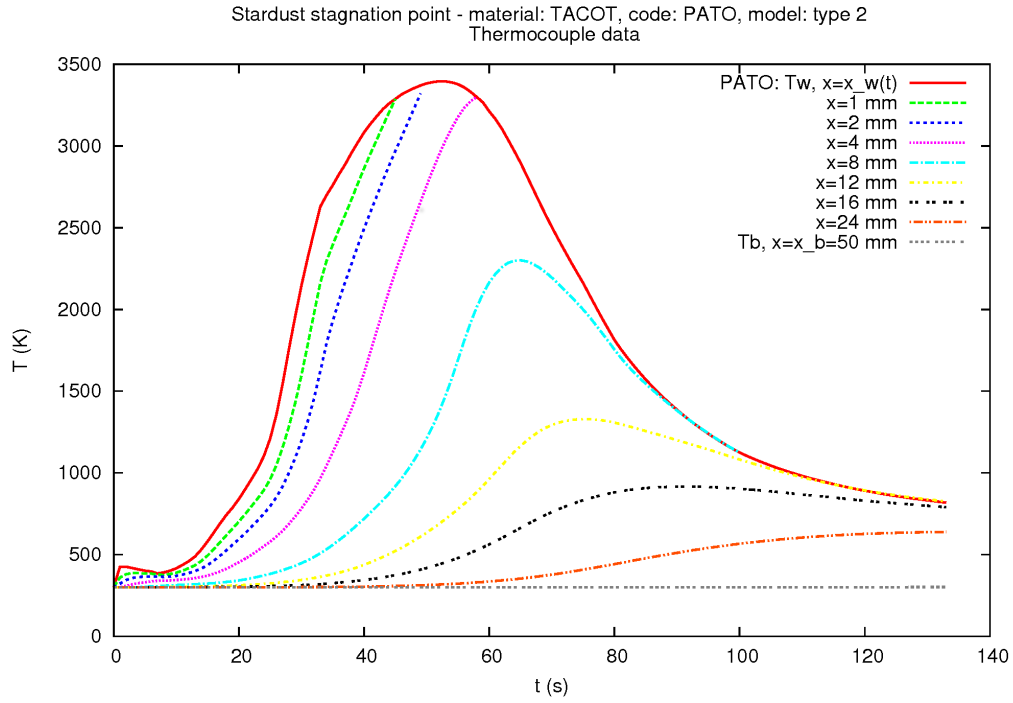


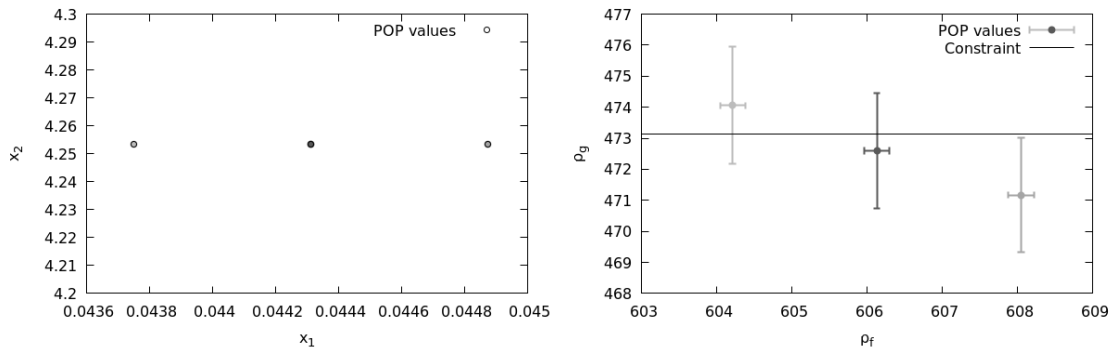

Figure V.6 Surface and in-depth temperatures obtained with nominal parameters.

In this test-case, $\Xi$ is of dimension 12, and all uncertainties are assumed uniform. The choice of the uncertain parameters relies on a sensitivity analysis performed on this test-case in a previous paper [Rivier et al., 2019]. Retained uncertain parameters are: 1) Density and volume fraction of the fibrous preform (±5% uncertainty each); 2) Density and volume fraction of the phenolic resin (±5% uncertainty each); 3) Thermal properties of the charred material: heat capacity, conductivity and emissivity (±5% uncertainty each); 4) Oxygen fraction in the pyrolysis gases (±10% uncertainty); 5) Pyrolysis reaction activation energy (±10% uncertainty); 6) Overall width of the system (0.1 cm uncertainty). Both design parameters are also affected by an uncertainty (±0.005 for $x_1$ and ±0.1 for $x_2$).

Each function evaluation requires the 1D simulation of heat transfer and shield ablation, taking approximately 10 minutes to compute on a 2.90 GHz processor. An acceptable global cost should remain within a day, or equivalently below 100 to 150 evaluations.

SABBa is ran with parameters $N_{init} = 5$, $N_{new} = 1$ (sequential optimizer), $N_{first} = 4$ and $N_{ref} = 1$ (sequential refinement). As for the normalized thresolds $\bar{\boldsymbol{s}}_1$ and $\bar{\boldsymbol{s}}_2$, they are both sequentially taken as 50%, 40%, 30%, 20%, 10%, 5%, 3%, 2% and 1% in all dimensions.

Three optima are found using SABBa with coupled-space surrogate model (SA-CS) for a computational cost of only **40** function evaluations. These Pareto optimal designs are plotted in Figure V.7. The design $\boldsymbol{x}^* \approx (4.43 \times 10^{-2}, 4.25)$ has the highest Pareto Optimal Probability (POP), of approximately 65%. It corresponds to the middle box in Fig. V.7(b), where $\boldsymbol{\rho}_f$ must be minimised and the admissible set is below the constraint threshold $\boldsymbol{\rho}_g = 473.15$ K.

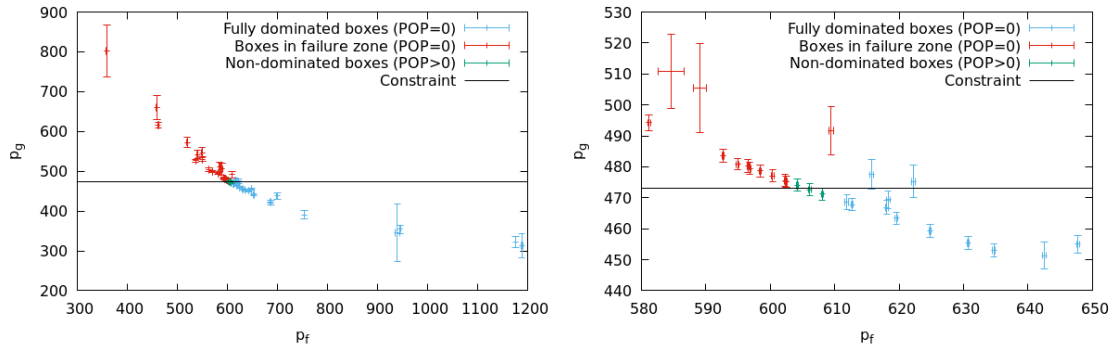



(a) (b)

Figure V.7 Outputs of SABBa in (a) the input space, (b) the objective/constraint space.

In Figure V.8, we have reported the performance of all the designs evaluated during the optimization. They are plotted in blue when dominated, red when entirely in the failure zone and green when non-dominated.

Among the three optimal designs depicted in Figure V.7, only the right one lies entirely in the admissible set, and thus belongs to $\mathcal{A}_{\mathcal{B}}$, as defined in Eq. (III.5) and (III.6). Contrarily to the two other optimal designs, it can dominate other boxes. In Figure V.8, it can be observed that this design dominates all the blue boxes, that are not entirely in the failure zone but are strictly worse in the objective dimension. Contrarily, the red boxes belong to $\mathcal{F}_{\mathcal{B}}$ and lie entirely in the failure zone. Hence, they are all considered dominated. This behavior explains why only three designs are kept, as shown in Figure V.7. It can be observed in Figure V.8 that the Bounding-Box approach permits to exploit the tunable accuracy since extreme boxes (on the very left or very right) are estimated with very wide uncertainty.




(a) (b)

Figure V.8 All boxes, depicted in blue when dominated, in red when entirely in failure zone and in green when non-dominated. (a) Full view, (b) zoomed-in view.

Figure V.9 provides a visualisation of the final SA model. In particular, we plot the SA mean surface density $\boldsymbol{\rho}_{SA_f}(\boldsymbol{x})$, wherever the constraint $\boldsymbol{\rho}_{SA_g}(\boldsymbol{x}) \leq 473.15$ k is satisfied. One can see that (i) the optima plotted in Figure V.7 are coherent with this SA model, and (ii) the objective value $\boldsymbol{\rho}_{SA_f}(\boldsymbol{x})$ seems merely constant on the constraint

limit. The final accuracy is imposed at 1%, which may not be enough to discriminate designs on the constraint limit. The design $\boldsymbol{x}^* \approx (4.43 \times 10^{-2}, 4.25)$ is returned by SABBa but the whole border from $(0.01, 5.1)$ to $(0.07, 4)$ is of high interest.

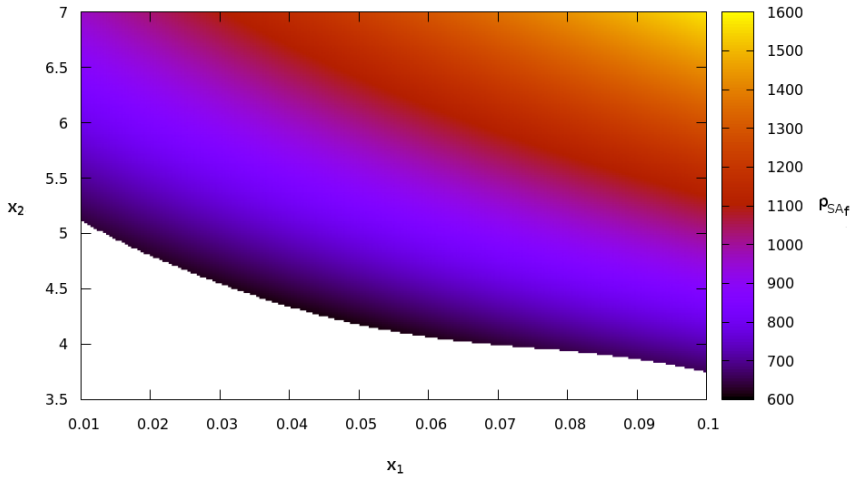



Figure V.9 Final SA model, representing $\boldsymbol{\rho}_{SA_f}(\boldsymbol{x})$ where the constraint $\boldsymbol{\rho}_{SA_g}(\boldsymbol{x}) \leq 473.15$ K is satisfied.

**Discussion** *For the cost of only 40 evaluations, SABBa has provided an optimal design with 1% final accuracy. This application gives insights for designing Thermal Protection Systems and shows that SABBa can be very cost-efficient when a simple behavior of the robustness and reliability measure is detected. Here, the objective is found to be linear, and the constraint limit shows low complexity, which explains that the SA strategy was able to return accurate optima quickly.*

## V–3 ORC turbine blade optimization

Robust shape optimization is here applied to the design of a typical converging-diverging turbine nozzle for ORC applications. This configuration has been extensively studied in the context of deterministic optimization [Pini et al., 2015a, Vitale et al., 2017, Persico et al., 2019] and multi-point optimization [Pini et al., 2014a]. Some recent works investigated the design of the blade mentioned above under epistemic uncertainties [Razaaly et al., 2019a] due to turbulence modelling, and investigated the robustness of the blade design under aleatoric uncertainties due to variability in the operating conditions, uncertainty in the thermodynamic model parameters, and geometric tolerances [Razaaly et al., 2019b]. First, we minimize the mean performance of the ORC turbine, under the constraint that the mean mass-flow rate lies within a prescribed range around the nominal value. Secondly, we explore the interest in reducing the variability of the turbine performance by formulating a Taguchi-like bi-objective problem.

Because of the higher complexity, both formulations of this test-case are solved using Joint SAMATA, that has shown very efficient in the previous sections. We then compare the computational cost associated with solving mono- and multi-objective problems and assess Joint SAMATA on a real-world constrained multi-objective engineering application.

Section V–3.1 first presents the physical problem of interest. Then, Section V–3.2 presents the uncertain parameters and both Optimization Under Uncertainty (OUU) formulations. The numerical ingredients for the simulation are gathered in Section V–3.3 and results for the mono-objective and bi-objective formulations are finally given in Sections V–3.4 and V–3.5, respectively.

## V–3.1 Physical application

The Biere represents a reference two-dimensional benchmark geometry to test the design of devices operating with the siloxane fluid MDM (Octamethyltrisiloxane, $C_8H_{24}O_2Si_3$) which properties are reported in Table V.1. This blade profile is meant to obtain a convergent-divergent cascade passage which serves to accelerate the fluid up to supersonic speed. Across the cascade, the fluid is expanded from superheated conditions. As the flow past the cascade is highly supersonic (M $\approx$ 2 at the blade trailing edge), compressibility effects play a crucial role. Indeed, because of the high Mach number achieved at the nozzle exit, a typical fish-tail shock pattern is generated downstream the trailing edge. The presence of strong shocks past stator vanes may result in significant losses, and thus, the design of the trailing edge region is critical for the turbine efficiency. Moreover, shock-waves propagate through the vane and usually interact with the boundary layer developing over the suction side of the neighboring blade, thus further compromising the efficiency of the cascade. The Mach field is depicted in Figure V.10 with the baseline blade profile.

| | |
|---|---|
| Critical pressure | 14.152 bar |
| Critical temperature | 564.1 K |
| Critical density | 256.82 $kg.m^{-3}$ |
| $\gamma$ | 1.0165 |
| Acentric factor $\omega$ | 0.529 |
| Gas constant | 35.152 J/kg/K |
| $\mu$ | $1.1517 \times 10^{-5}$ Pa.s |
| $k$ | 0.03799 W/(m.K) |

Table V.1 Gas properties of the siloxane MDM

## V–3.2 Problem setting

The objective function $\Delta P$ is defined as the standard deviation of the azimuthal distribution of static pressure half an axial chord downstream of the blade Trailing Edge (TE). Indeed, minimizing $\Delta P$ within the optimization is convenient for such highly supersonic cascade by yielding a significant reduction of the shock strength, and hence of the shock loss. A constraint is imposed on the mass-flow rate per unit span $\dot{m}$, normalized with respect to the nominal value.

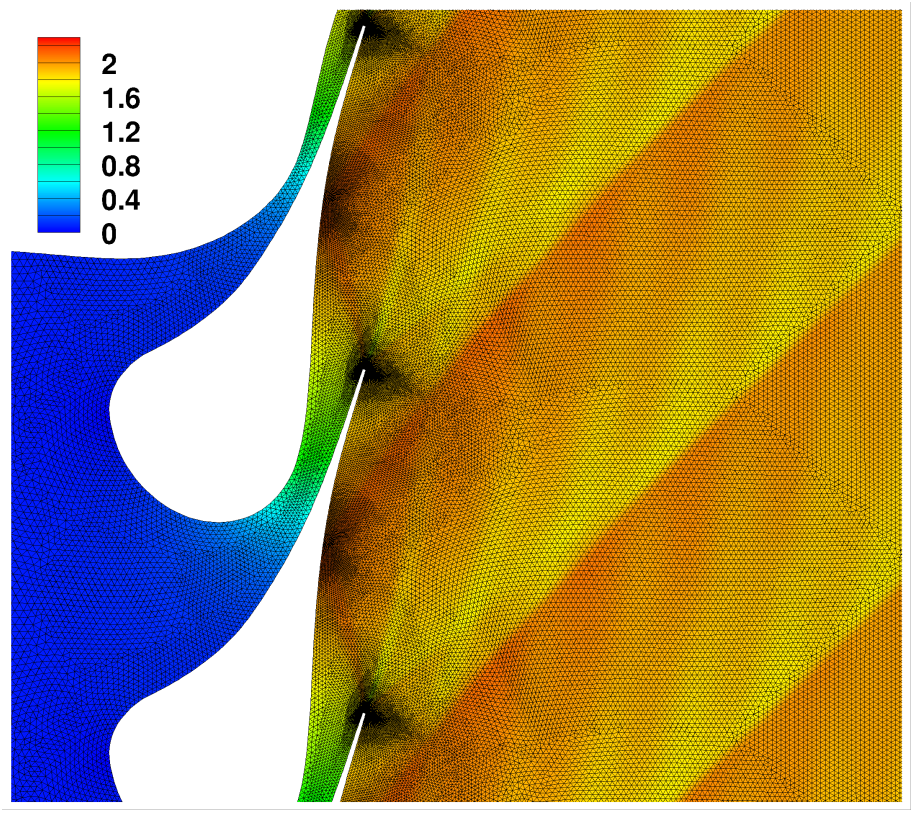



Figure V.10 Mach contours at nominal conditions for the baseline profile and computational grid of 36k cells.

Uncertainties on the operating conditions are considered both at the inlet and the outlet of the turbine. These variabilities simply result in a change of boundary conditions for each cascade. Following [Pini et al., 2014b], we model the operational variability as independent and uniform uncertainties, gathered in the random vector $\xi = [P_{in}^t, T_{in}^t, P_{out}^s]$. The range of these uncertainties is also reported in Table V.2. Note that the uncertainties on the parameters of the thermodynamical model are neglected since previous studies [Congedo et al., 2013a] provided evidence of their limited impact with respect to the uncertainties on operating conditions for turbine cascades.

| $\xi$ | $P_{in}^t$ (bar) | $T_{in}^t$ (K) | $P_{out}^s$ (bar) |
|---|---|---|---|
| Nominal ($\xi_0$) | 8 | 545.15 | 1.072 |
| Mean ($\xi_\mu$) | 8 | 545.15 | 1.5 |
| Random | $\mathcal{U}[7.95, 8.05]$ | $\mathcal{U}[541.15, 549.15]$ | $\mathcal{U}[1, 2]$ |

Table V.2 Operating Conditions: Nominal, Mean and Random.

Two problems are treated here. The first problem is mono-objective and reads as follows:

$$\begin{aligned}
\text{minimize:} \quad & \boldsymbol{\rho}_f(\boldsymbol{x}) = \mu_1(\boldsymbol{x}) \\
\text{satisfying:} \quad & \boldsymbol{\rho}_g(\boldsymbol{x}) = \mu_2(\boldsymbol{x}) \in [0.98\dot{m}_0, 1.02\dot{m}_0] \\
\text{where:} \quad & \mu_1(\boldsymbol{x}) = \mathbb{E}_\xi[\Delta P(\boldsymbol{x}, \xi)] \\
& \mu_2(\boldsymbol{x}) = \mathbb{E}_\xi[\dot{m}(\boldsymbol{x}, \xi)] \\
& \dot{m}_0 = 344.843 \\
\text{with:} \quad & \xi \sim \mathcal{U}(\Xi) \\
\text{by changing:} \quad & \boldsymbol{x} \in \mathcal{X}
\end{aligned} \tag{V.3}$$

where $\Xi = [7.95, 8.05] \times [541.15, 549.15] \times [1, 2]$. The design space $\mathcal{X}$ allows to moshape of the blade as much as possible while ensuring convergence of the numerical simulation.

The second problem of interest is a multi-objective Taguchi formulation:

$$\begin{aligned}
\text{minimize:} \quad & \boldsymbol{\rho}_f(\boldsymbol{x}) = \begin{pmatrix} \mu_1(\boldsymbol{x}) \\ \sigma^2(\boldsymbol{x}) \end{pmatrix} \\
\text{satisfying:} \quad & \boldsymbol{\rho}_g(\boldsymbol{x}) = \mu_2(\boldsymbol{x}) \in [0.98\dot{m}_0, 1.02\dot{m}_0] \\
\text{where:} \quad & \mu_1(\boldsymbol{x}) = \mathbb{E}_\xi[\Delta P(\boldsymbol{x}, \xi)] \\
& \sigma^2(\boldsymbol{x}) = \mathbb{V}_\xi[\Delta P(\boldsymbol{x}, \xi)] \\
& \mu_2(\boldsymbol{x}) = \mathbb{E}_\xi[\dot{m}(\boldsymbol{x}, \xi)] \\
& \dot{m}_0 = 344.843 \\
\text{with:} \quad & \xi \sim \mathcal{U}(\Xi) \\
\text{by changing:} \quad & \boldsymbol{x} \in \mathcal{X}
\end{aligned} \tag{V.4}$$

The optimal blade profile will be compared with two deterministic optima, computed with at the Nominal and Mean operating conditions, as presented in Table V.2.

## V–3.3 Numerical ingredients

**Parametrization** The pressure and suction sides are parametrized using a unique B-spline curve of degree 3 ([Farin, 2002]), defined over a total number of 30 Control Points (CP). The design vector $\boldsymbol{x}$ parametrizing the 2D cascade is constituted by a subset of 9 CP allowed to be displaced in the direction normal to the baseline geometry, as illustrated in Figure V.11.

**SU2 Solver** Since the study aims at the aerodynamic optimization of the blade profile, the flow model focuses on the two-dimensional flow at the midspan section of the cascade. The numerical domain is periodic with a pitch spacing of 45-mm. The flow is simulated up to a distance of 0.5 and 2 chord-lengths ahead and past the blade, respectively. Total Pressure $P_{in}^t$, total Temperature $T_{in}^t$, and axial flow direction are assigned at the inlet, while static pressure $P_{out}^s$ is fixed at the outlet. To estimate the aerodynamic performances of the supersonic turbine, the Non-Ideal Compressible-Fluid Dynamics solver included in the SU2 [Palacios et al., 2013, Economon et al., 2016, Pini et al., 2016, Vitale et al., 2015] suite is employed, embedding, in particular, the Peng-Robinson-Stryjek-Vera Equation of State to describe the fluid thermodynamic behavior.

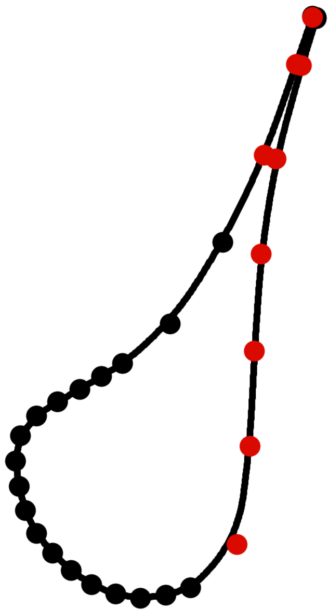


Figure V.11 B-splines parametrization. Fixed CP in black, moving CP in red.

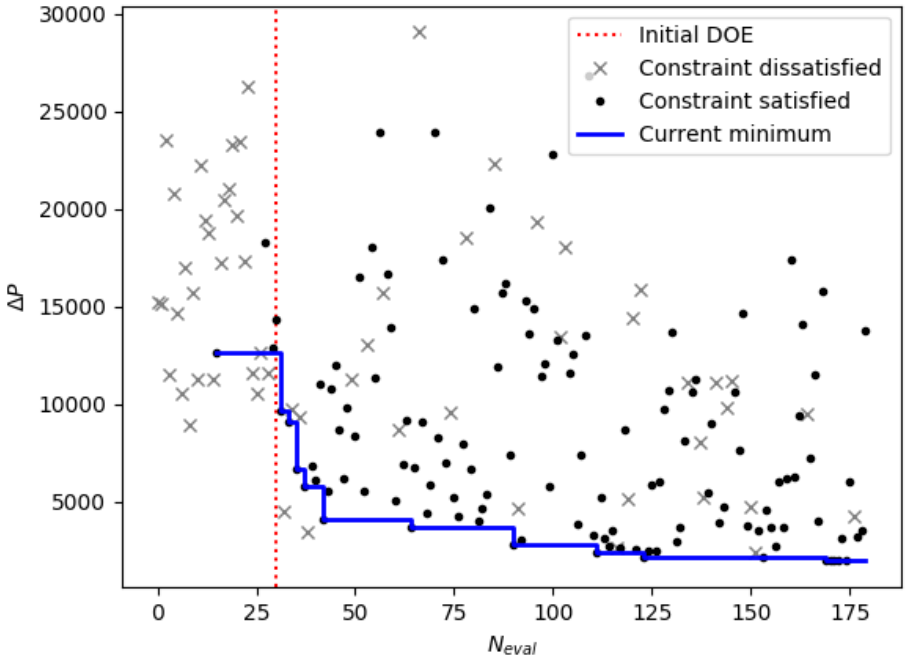
Inviscid fluxes are discretized using a MUSCL approach based on an approximate Riemann solver of Roe upwind type [Roe, 1981, Vinokur and Montagné, 1990] along with the slope limiter proposed by van Albada, while central differences are used for the viscous terms. Non-Reflecting Boundary Conditions [Giles, 1990] are exploited in SU2 to avoid pressure oscillations due to the reflection of spurious pressure waves at domain boundaries. Simulations are performed using a restart file corresponding to the baseline simulation.

**Mesh** The unstructured grids are generated using an in-house tool based on an advancing-front/Delaunay algorithm, requiring special treatment of the blade trailing edge region, since the lack of viscosity prevents the inviscid flow to detach from solid boundaries and this may lead to non-physical solutions and convergence issues. A truncated trailing edge address those issues by introducing two sharp corners that enforce the separation of the flow. Detailed convergence analyses (not shown) have been performed, leading to a computational mesh of the flow domain with 36,000 triangular elements (Figure V.10), which represents a trade-off between accuracy and computational cost. During the optimization process, several blade profiles are generated. A dedicated mesh deformation tool based on Radial Basis Functions (RBF), initially presented in [De Boer et al., 2007] and successfully applied in [Pini et al., 2015b, Razaaly et al., 2019b] allows high flexibility and robustness while maintaining the grid connectivity.

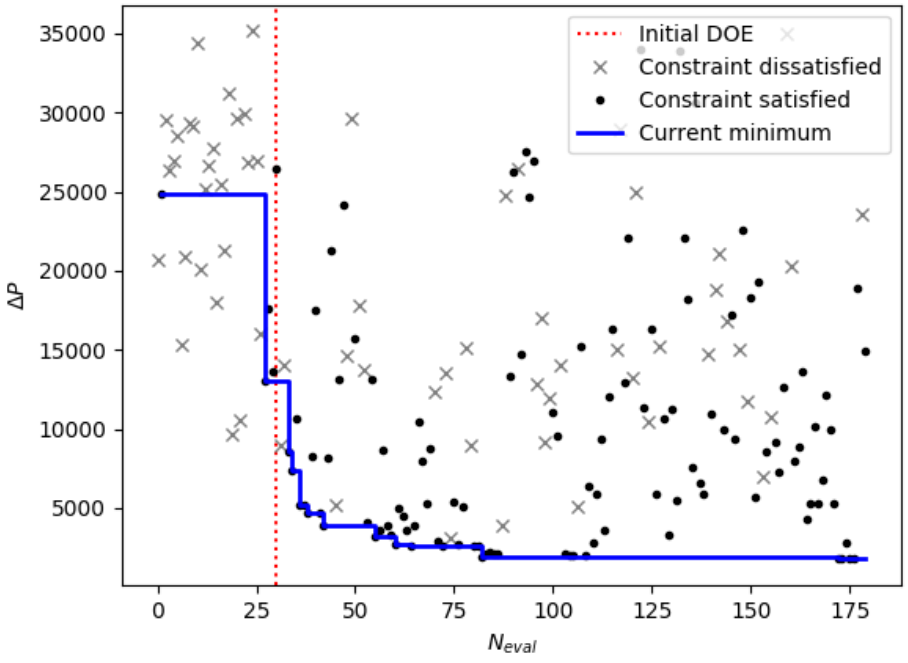
## V–3.4 Mono-objective results

In the case of problem (V.3), deterministic solutions are first computed in Section V–3.4.1 to give a reference for comparing robust optima, and to assess the impact of considering uncertainties within the optimization process. Notably, because nominal conditions are not centered with the range of the uncertain parameters, we conduct the deterministic optimization process both at nominal conditions and at mean con-

ditions. The robust optimization process is then carried out in Section V–3.4.2 with the Joint SAMATA strategy. Intermediate and final optima are depicted with respect to the number of simulation calls. The statistics associated with deterministic and robust optima are pictured in Section V–3.4.3, and reveal the impact of taking uncertainties into account during the optimization phase.




(a)




(b)

Figure V.12 Deterministic optimizations convergence curves with operating conditions at (a) Nominal values and (b) Mean values.

### V–3.4.1 Deterministic optimizations

Deterministic optimizations of the blade profile are performed at $\xi_0$ and $\xi_\mu$ using a state-of-the-art optimization strategy based on Bayesian Optimization, similarly to [Razaaly, 2019]. Gaussian Processes (GP) are constructed on $\Delta P(\boldsymbol{x}, \xi_0)$ and $\dot{m}(\boldsymbol{x}, \xi_0)$ (and similarly at $\xi_\mu$), and the Expected Improvement (EI) criterion determines the next design $\boldsymbol{x}$ to visit. Note that in this constrained setting, the EI criterion is multiplied by the Probability of Feasibility (PF) [Gardner et al., 2014], to comply with the constraint. The Matern 5/2 kernel is chosen for building these GPs.

The convergence curves associated with these optimizations are drawn in Figure V.12. Black dots are related to satisfying constraints while grey crosses have a value $\dot{m}$ outside of the 2% range. An initial Design of Experiment (DoE) of 30 evaluations is performed and represented with the dotted red line. The current minimum objective value is represented with the blue curve. Note that in both cases, the minimum $\Delta P$ value is roughly reached between 75 and 100 simulation calls, and further refinements only provide minimal improvements.

We denote by $\boldsymbol{x}_0^*$ the optimal blade profile at the nominal conditions $\xi_0$ and $\boldsymbol{x}_\mu^*$ at mean conditions $\xi_\mu$. These two blades are plotted in Figure V.13, with $\boldsymbol{x}_0^*$ in blue and $\boldsymbol{x}_\mu^*$ in orange.

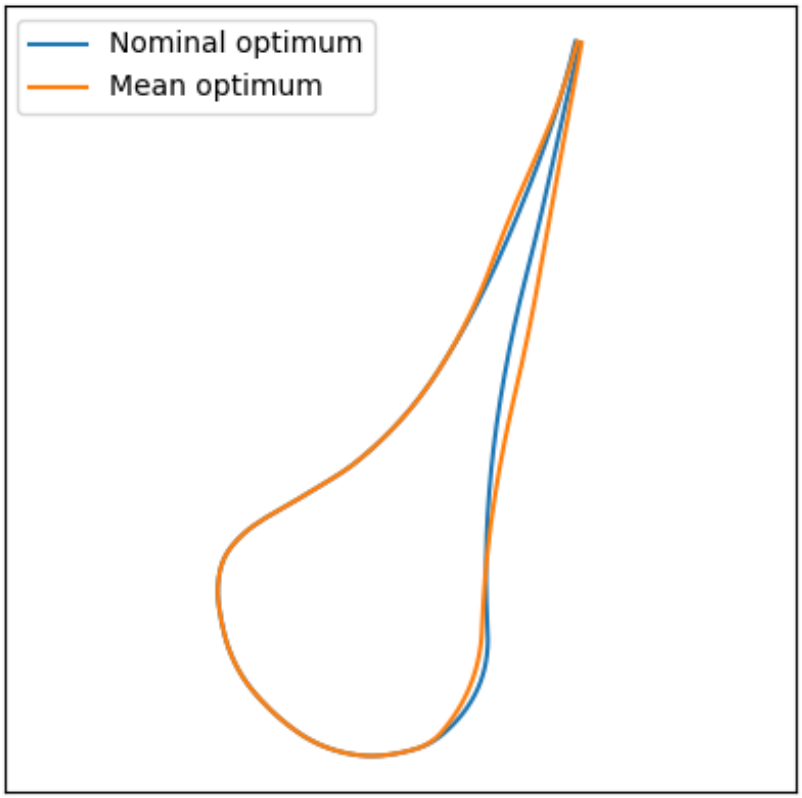



Figure V.13 Optimal blades in the Nominal (blue) and Mean (orange) conditions.

### V–3.4.2 Robust optimization

Problem (V.3) is then solved using the sampling-based SAMATA strategy with coupled-space surrogate model and large-scale random field realizations, as presented in Algorithm IV.2. The NSGA-II optimization procedure is again exploited, and constraints are handled through a penalization. The initial DoE consists of 30 evaluations drawn with a maximin Latin Hypercube Sampling (LHS) in the coupled space. Each NSGA-II population contains 32 individuals, and evaluations for measure refinement are all performed sequentially. The normalized threshold values iteratively take values of 10%, 5%, 3%, 2% and 1%. The design with the highest POP at the end of each iteration is represented in Figure V.14 w.r.t. the number of function evaluations. The violin plot is constructed using a set of 1000 samples of $\boldsymbol{\rho}$. As expected, lower threshold values induce more evaluations but yield more accurate measure approximations. Note that only the objective measure $\boldsymbol{\rho}_f$ is plotted in Figure V.14.

Similarly to Fig. V.12, samples associated with satisfied constraint are represented with a black dot in Figure V.14. Note that all samples are represented as such, meaning that they all satisfy the prescribed constraints. Convergence values are gathered in Table V.3, with a final optimum centered on $\boldsymbol{\rho}_f = 8087$, with an standard deviation of the approximation error of 44.26. This optimum satisfies the 1% accuracy imposed by the normalized thresholds and has required 145 simulation calls.

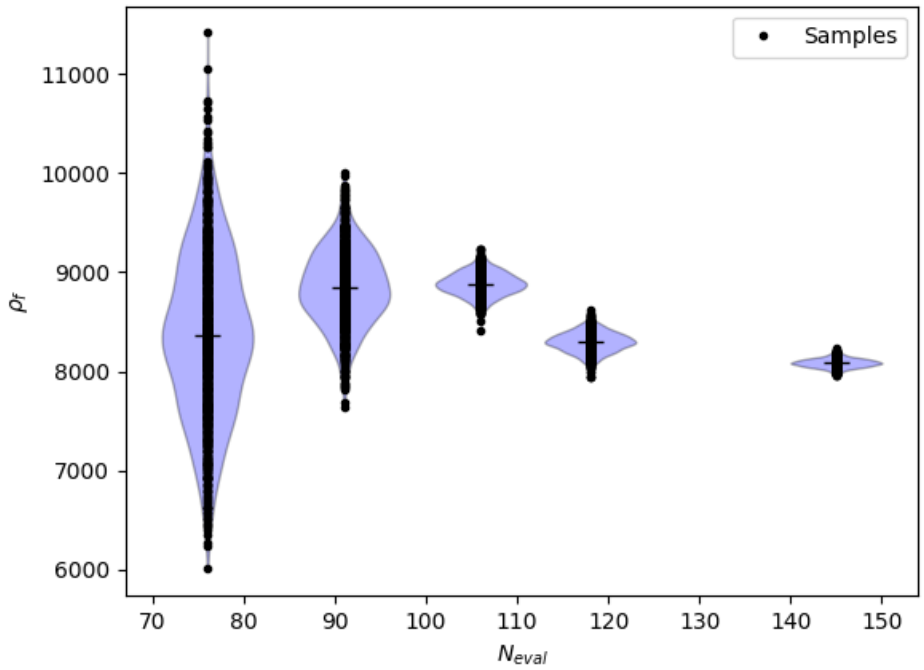



Figure V.14 Violin plot of the sampled robustness measure associated with highest POP against the number of function evaluations. Each violin correspond to a specific threshold value (10%, 5%, 3%, 2% and 1%).

| $N_{eval}$ | $\rho_f$ | | $\rho_g$ | |
|---|---|---|---|---|
| | Mean | Std. dev. | Mean | Std. dev. |
| 76 | $8.328 \times 10^3$ | $6.355 \times 10^2$ | $3.464 \times 10^2$ | $1.065 \times 10^{-1}$ |
| 91 | $8.854 \times 10^3$ | $3.986 \times 10^2$ | $3.514 \times 10^2$ | $3.149 \times 10^{-2}$ |
| 106 | $8.869 \times 10^3$ | $1.124 \times 10^2$ | $3.387 \times 10^2$ | $1.534 \times 10^{-2}$ |
| 118 | $8.288 \times 10^3$ | $1.068 \times 10^2$ | $3.500 \times 10^2$ | $1.144 \times 10^{-2}$ |
| 145 | $8.087 \times 10^3$ | $4.426 \times 10^1$ | $3.505 \times 10^2$ | $1.812 \times 10^{-2}$ |

Table V.3 Optimal outputs associated to the different threshold values.

**Remark** *Note that the number of simulation calls required for performing this Optimization Under Uncertainty is not significantly higher than the cost of the deterministic optimizations carried out in Section V–3.4.1 with Bayesian Optimization algorithms. It corroborates the very high efficiency of the proposed SAMATA strategy.*
*Notably, such robust optimization has been carried out in [Razaaly, 2019], with a more classical Double Loop approach, where a full Uncertainty Propagation is carried out at each optimization iteration, at the cost of several thousand evaluations. Here, SAMATA permits to reduce by around one order of magnitude the number of required function evaluations.*

The design associated with the highest Pareto Optimal Probability (POP) at the last iteration is denoted $\boldsymbol{x}^*_{OUU}$ and is plotted in Figure V.15 against the deterministic optimal blade profiles. As one could expect, the optimal shape is closer to the Mean optima than to the Nominal one. However, the Robust blade shape is slightly flatter, as can be seen in Figure V.15. The Nominal blade is represented in blue, the Mean one in orange, and the Robust one in green.

### V–3.4.3 Post-processing analysis

Each optimization performed in the preceding sections yields a blade profile that shows more or less sensitivity to the uncertain parameters. Notably, the pressure and velocity fields are very dissimilar when computed at the Nominal operating conditions $\xi_0$ or the Mean operating conditions $\xi_\mu$. This behavior comes from the fact the output pressure in $\xi_0$ and $\xi_\mu$ are significantly different.

| Blade profile | $\Delta P(\boldsymbol{x}, \xi_0)$ | $\Delta P(\boldsymbol{x}, \xi_\mu)$ | $\mathbb{E}_\xi[\Delta P(\boldsymbol{x}, \xi)]$ |
|---|---|---|---|
| Nominal optimum $\boldsymbol{x} = \boldsymbol{x}^*_0$ | $1.958 \times 10^3$ | $1.526 \times 10^4$ | $1.624 \times 10^4$ |
| Mean optimum $\boldsymbol{x} = \boldsymbol{x}^*_\mu$ | $1.079 \times 10^4$ | $1.773 \times 10^3$ | $8.472 \times 10^3$ |
| Robust optimum $\boldsymbol{x} = \boldsymbol{x}^*_{OUU}$ | $1.210 \times 10^4$ | $3.082 \times 10^3$ | $8.084 \times 10^3$ |

Table V.4 Nominal values and statistics for different blade profiles.

The statistics associated with each blade profile from Figure V.15 are gathered in Table V.4 and the full distributions of $\Delta P$ induced by uncertain parameters $\xi$ are plotted in Figure V.16. As expected, the Nominal optimum yields very high variability in Fig. V.16(a) but very low nominal output of $1.958 \times 10^3$ bar, represented with a dashed black line. The Mean and Robust optima are again quite similar, Figs V.16(b)

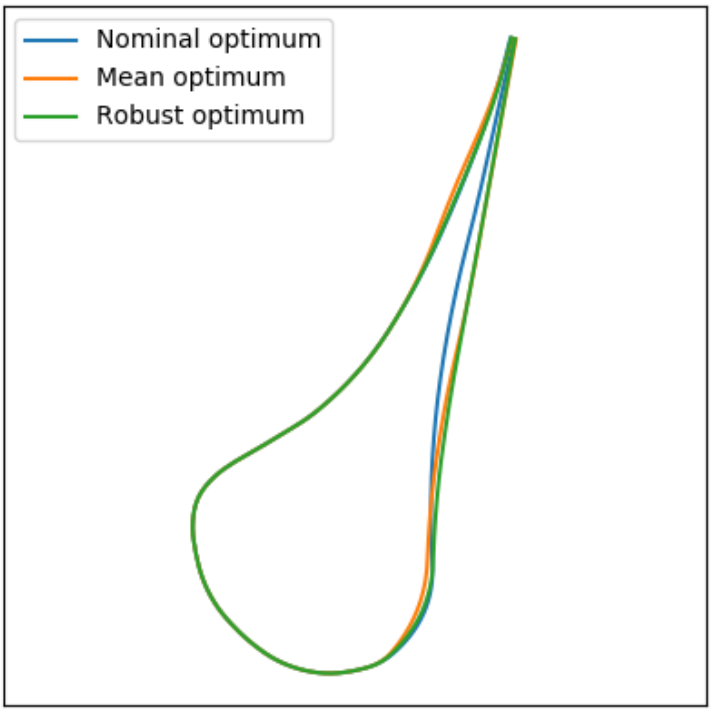



Figure V.15 Nominal (blue), Mean (orange) and Robust (green) optimal blades.

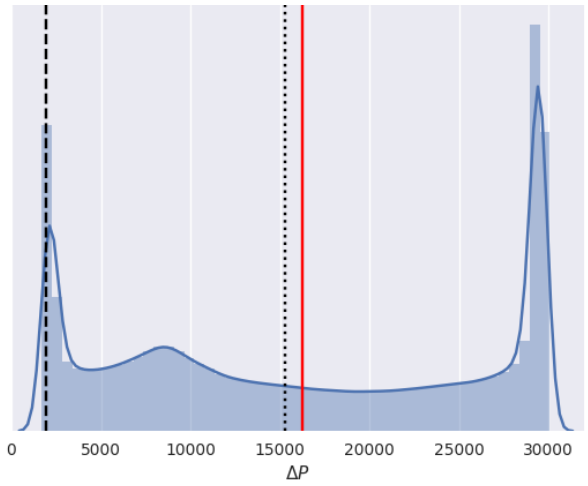
and V.16(c). However, while the Mean optimum gives very low $\Delta P = 1.773 \times 10^3$ bar at $\xi_\mu$, represented with a dotted black line, the Robust optimal blade shows a lower value for the expectation $8.084 \times 10^3 < 8.472 \times 10^3$, drawn as a plain red line.

These distributions are computed from an extensive Monte Carlo sampling drawn on a Gaussian Process surrogate model in $\Xi$. This GP is built on 50 function evaluations $\{\Delta P(\boldsymbol{x}, \xi_i)\}_i$.

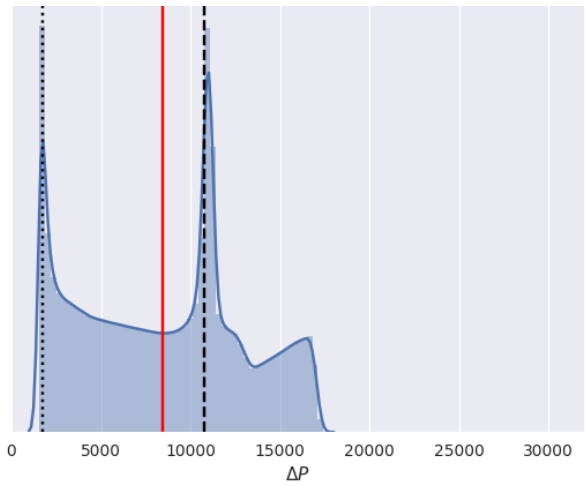
Figure V.17 then depicts the Mach field computed at $\xi_0$ and $\xi_\mu$ for each optimal blade drawn in Figure V.15. As expected, velocities are higher at $\xi_0$, where the output pressure is low. In addition, as the Mean (orange) and Robust (green) optimal blades in Figure V.15 are quite similar, the Mach fields V.17(c) and V.17(d) are comparable with V.17(e) and V.17(f). Finally, as expected, shock strengh is minimized by $\boldsymbol{x}_0^*$ at Nominal operating conditions $\xi_0$, Figure V.17(a), and by $\boldsymbol{x}_\mu^*$ at Mean conditions $\xi_\mu$, Figure V.17(d).

Similarly, the mean Mach field and associated standard deviation are drawn in Figure V.18. This figure permits to grasp the overall variability and anticipate the performance of each design over the uncertain space $\Xi$. The Nominal optimal design, Figures V.18(a) and V.18(b) reveals high variations of the Mach field and significant shocks in the mean-field. On the contrary, Figures V.18(c) to V.18(f) show lower variability and smooth mean Mach field for the Mean and Robust optimal blades. Notably, the secondary shocks are nearly eliminated in the Robust mean-field, Fig. V.18(e).

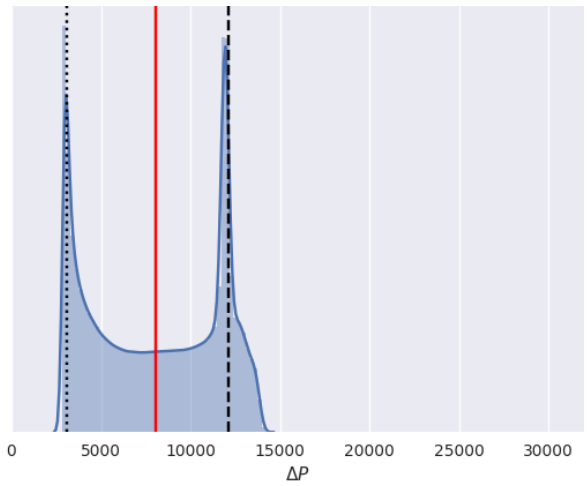
All these mean fields and standard deviations are empirically computed based on 50 function evaluations equally distributed within $\Xi$.




(a)




(b)




(c)

Figure V.16 Distribution of $\Delta P$ at (a) $\boldsymbol{x}_0^*$, (b) $\boldsymbol{x}_\mu^*$ and (c) $\boldsymbol{x}_{OUU}^*$. $\Delta P(\boldsymbol{x}, \boldsymbol{\xi}_0)$ represented as a dashed black line, $\Delta P(\boldsymbol{x}, \boldsymbol{\xi}_\mu)$ as a dotted black line and $\mathbb{E}_{\boldsymbol{\xi}}[\Delta P(\boldsymbol{x}, \boldsymbol{\xi})]$ as a plain red line.

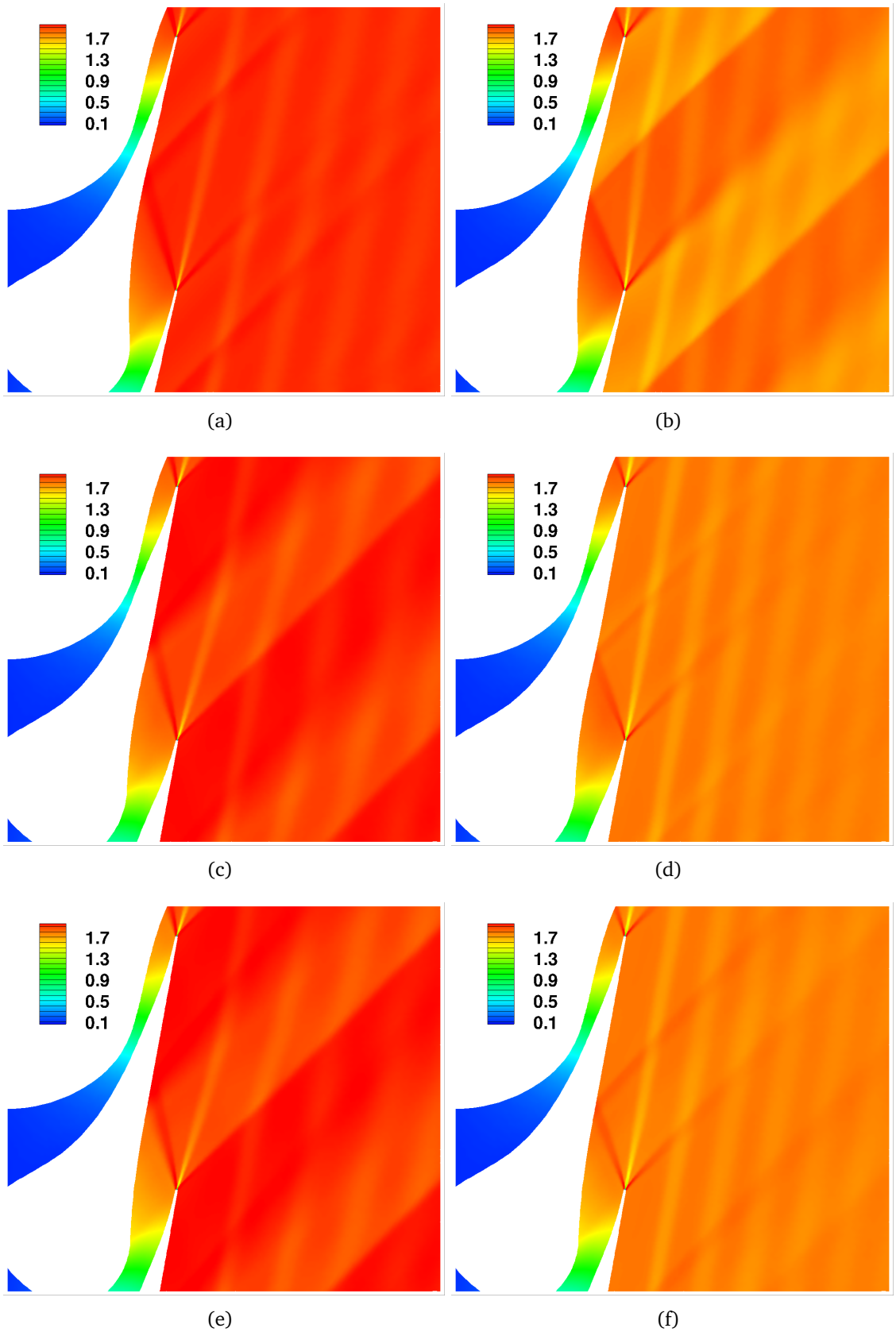



Figure V.17 Mach field computed at (a) $(\boldsymbol{x}_0^*, \xi_0)$, (b) $(\boldsymbol{x}_0^*, \xi_\mu)$, (c) $(\boldsymbol{x}_\mu^*, \xi_0)$, (d) $(\boldsymbol{x}_\mu^*, \xi_\mu)$, (e) $(\boldsymbol{x}_{OUU}^*, \xi_0)$ and (f) $(\boldsymbol{x}_{OUU}^*, \xi_\mu)$, .

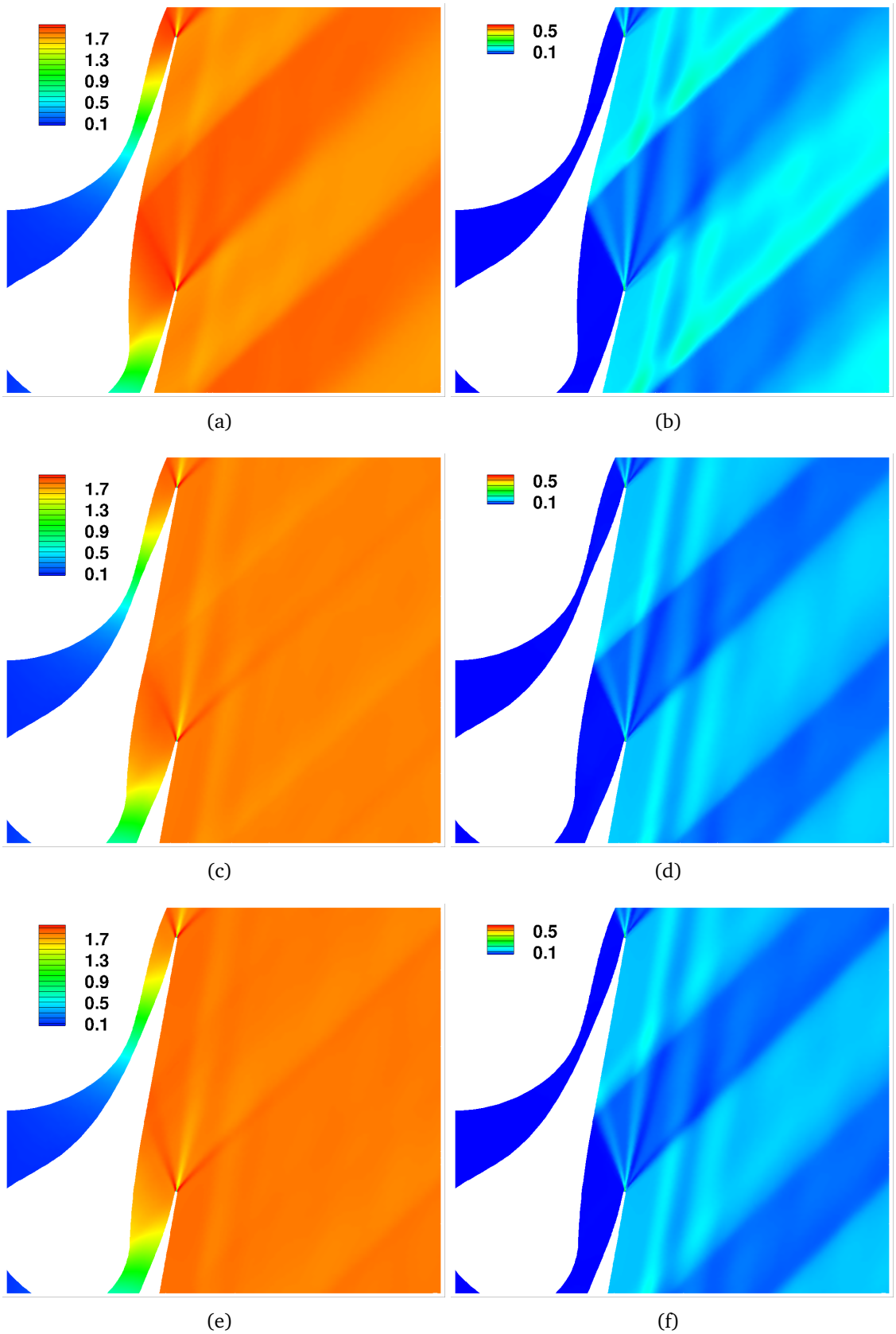



Figure V.18 Mean and standard deviation of the Mach field over $\Xi$ at (a,b) $\boldsymbol{x}_0^*$, (c,d) $\boldsymbol{x}_\mu^*$ and (e,f) $\boldsymbol{x}_{OUU}^*$.

## V–3.5 Bi-objective results

Finally, we demonstrate the capability of SAMATA to tackle real-world multi-objective problems on a Taguchi bi-objective formulation, in Equation (V.4). This formulation aims to find the Pareto front of trade-offs between good expected performance and low variability. The Joint SAMATA algorithm is again employed, with the NSGA-II optimization method and normalized thresholds of 10%, 5%, 2% and 1%.

The Pareto fronts associated with these threshold values are depicted in Figure V.19, and reveal great improvements in the Pareto front accuracy throughout the iterations. The total computational cost for attaining the chosen thresholds is of **72**, **122**, **276** and **419** simulation calls respectively. Note that each plus sign in Figure V.19 is a realization of the objectives $\mu_1$ and $\sigma^2$, and realization cloud associated with a given design has its transparency driven by its POP value (1: fully visible, 0: transparent). The mono-objective optimum found in the preceding section is represented as a red dot, and lies on the Pareto front, as expected.

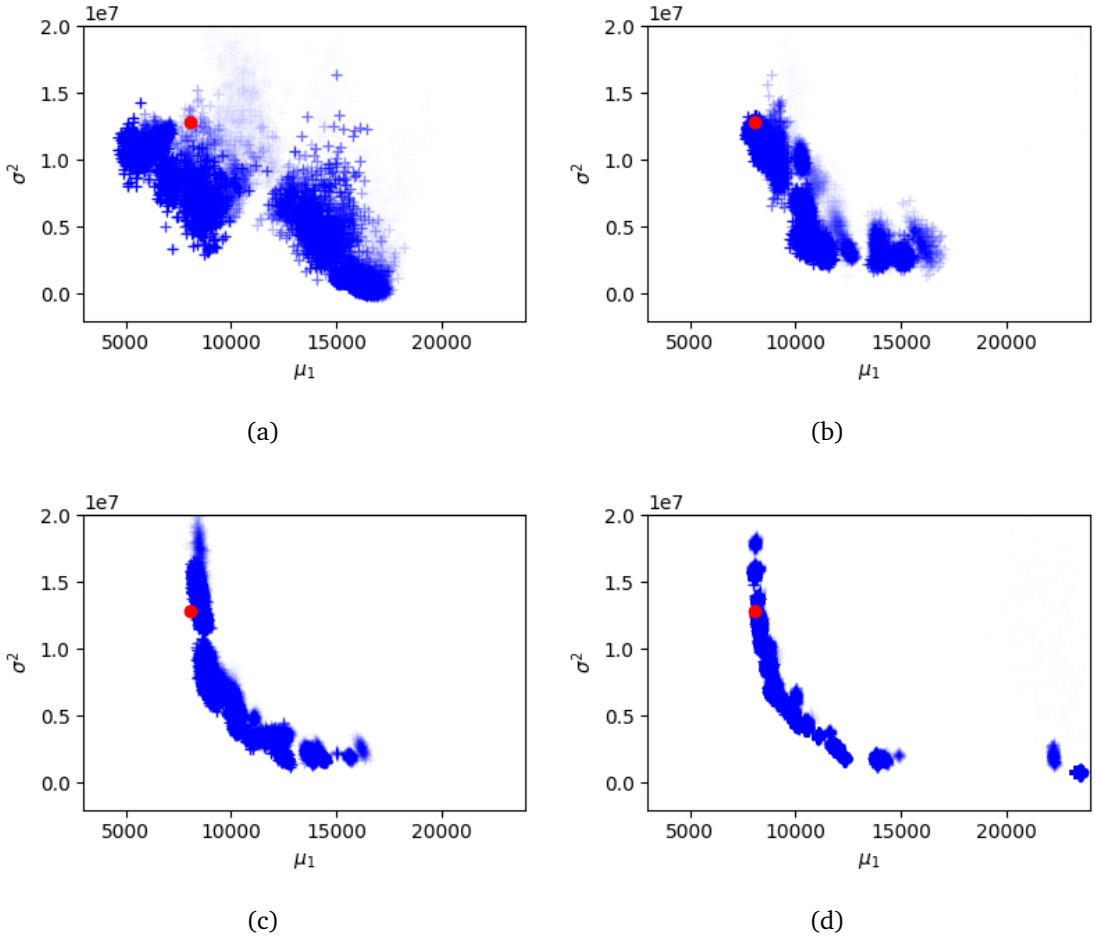



Figure V.19 Pareto front of the bi-objective OUU problem with normalized threshold values of (a) 10% (**72** evaluations), (b) 5% (**122** evaluations), (c) 2% (**276** evaluations) and (d) 1% (**419** evaluations). Transparency driven by POP value and mono-objective optimum from previous section as a red dot.

Joint SAMATA reveals very parsimonious, by returning a whole Pareto front with fine robustness measure computations for only **419** simulation calls, which is three times the cost of the mono-objective problem (**145** evaluations).

**Discussion** *In this section, a real-world engineering application has been tackled by the sampling-based SAMATA strategy. This robust optimization problem had already been solved in [Razaaly, 2019] with a more standard approach. The reference optimal robustness measure is* $7.9 \times 10^3$ *bar, which is well approximated by the optimum found here with* $8.084 \times 10^3$ *bar. Notably, SAMATA returns an optimal value as well as the associated impreciseness, which explains the small bias. However, the robust optimization process has been solved here at the cost of 100 to 150 function evaluations, which is one order of magnitude lower than the cost associated with the robust optimization in [Razaaly, 2019]. Furthermore, the bi-objective formulation only requires roughly 400 evaluations, which is still lower than the literature mono-objective study. These applications reveal the very high efficiency of the Joint SAMATA strategy for performing Optimization Under Uncertainty on a real-world engineering application.*

**Conclusion of the chapter** *Three engineering-based Optimization Under Uncertainty problems have been solved in this chapter. The first has been tackled with both SABBa, presented in Chapter III and Joint SAMATA, from Chapter IV, and permits to compare their behavior and parsimony on a real-world application. Because of the better behavior of Joint SAMATA, the second test-case of medium complexity is solved with SABBa, and the third and most difficult one is carried out with Joint SAMATA. The first test-case is the classical two-bar truss structure, with quantile-based reliability constraints. The problem has been solved at a very reasonable cost (*$< 50$ *evaluations), and one constraint has been shown to have a low impact on the optimization output.*

*In the second test-case, SABBa is applied with success to the design of a Thermal Protection System under material, chemical and geometrical uncertainties. Even though the coupled space features twelve dimensions, the Surrogate-Assisting strategy has quickly detected a linear behavior of the robustness measure and quasi-linear one for the reliability constraint. This behavior has permitted to get an optimal geometry for the TPS under a worst-case reliability constraint at the cost of fewer than 50 evaluations.*

*Finally, the third test-case deals with the geometric optimization of the blade profile in an Organic Rankine Cycle turbine. It has been tackled with the non-parametric SAMATA strategy with the coupled-space model for the cost of 100 to 150 evaluations, which is one order of magnitude less than the state of the art approach. A second formulation has been proposed for assessing the performance of Joint SAMATA on a bi-objective optimization problem. The Pareto front is accurately recovered and notably comply with the mono-objective optimum, at the cost of 100 to 400 evaluations, depending on the accuracy thresholds.*

*In summary, the proposed strategies have been successfully applied to several engineering-based problems, with both mono- and multi-objective formulations, and yield very low computational cost. Notably, the Joint SAMATA strategy is very parsimonious in terms of the number of evaluations when dealing with parametric uncertainties.*

# Part D

# Perspectives

CHAPTER VI

# Conclusions



**Overview** *This final chapter aims to illustrate the main achievements of this thesis and the limitations of the proposed techniques.*
*Some perspectives are then drawn for further improving the flexibility and parsimony.*

**Outline** *Section VI–1 first reviews the main contributions of this work..*
*Then, Section VI–2 highlights difficulties that can arise when applying the proposed strategies and proposes some techniques for coping with these issues.*
*Finally, Section VI–3 gives some perspectives, on improvements that could be incorporated within the proposed strategies, as well as on ideas that build on concepts introduced in this work.*

## VI–1 Main achievements

The major contributions of this thesis are summarized in this section.

## VI–1.1 Tunable accuracy formulation

In the context of Optimization under Uncertainty (OUU) problems, we have proposed an adaptive refinement strategy called Measure Approximation with Tunable Accuracy (MATA). Specifically, the accuracy for approximating the objectives and constraints at a given design is driven by the probability for this design of being non-dominated. This choice permits to reduce the effort for evaluating designs which are unlikely to be optimal.

This technique heavily relies on our capability to compare current approximations and select the most promising ones. To this extent, we have introduced the concept of probabilistic dominance for constrained multi-objective optimization under uncertainty as well as the so-called Pareto-Optimal Probability (POP), that associates a quantitative score to each design.

Finally, the MATA strategy has been coupled with a Surrogate-Assisting (SA) technique, that permits to increase the overall parsimony further. The approximated evaluations and their associated errors can be used to construct a predictive representation of the objectives and constraints over the whole design space to accelerate the optimization process.

Overall, the approximation of different uncertainty-based metrics with tunable accuracy and the use of a Surrogate-Assisting strategy are the main ingredients of the proposed algorithm, called SAMATA. This approach is flexible in terms of metrics formulations and reveals very parsimonious. This algorithm is notably applicable to generic optimization methods.

## VI–1.2 Bounding-Box approximations

This thesis then explores the influence of the error distribution on the algorithm performance. A first simplifying and conservative assumption is to consider the approximation error to follow independent uniform distributions. This assumption permits to derive specific Pareto dominance rules, that have been exploited in the literature within the so-called Bounding-Box framework. We have proposed a convergence analysis of such adaptive refinement technique within the optimization process. The recursive characteristic has notably been highlighted, and the coupling with the proposed SA strategy has been tackled. In this context, SAMATA is called Surrogate-Assisted Bounding-Box approach (SABBa), and its convergence has been demonstrated under some assumptions. The asymptotic computational cost has finally been pictured on generic noisy optimization problems against more conventional techniques.

SABBa has then been put in practice on OUU problems through the use of Gaussian Process (GP)-based formulations for computing lower and upper bounds on the robustness and reliability measures. The proposed method relies on a $\pm 3\sigma$ paradigm that is very conservative and usually overestimates the true approximation error. Specific POP formulations for Bounding-Boxes have been provided, and a quantitative quality indicator has been proposed for assessing the convergence rate in test-cases where optimal designs are known.

This strategy has then been put in practice on some analytical test-cases with several inner techniques for constructing the GP surrogate model. Notably, constructing these models in the coupled design/uncertainty space revealed particularly efficient when the surrogate yields accurate predictions. Overall, SABBa shows slightly faster convergence and much higher reliability than a classical A Priori Metamodel strategy.

Its performance has also been assessed on a structural mechanics test-case and an aerospace engineering application. SABBa has shown good parsimony in both cases and was notably able to accurately find the optimal area within 50 simulation calls.

### VI–1.3 Sampling-based approximations

We finally proposed to compute non-parametric approximations of the error distributions. To this extent, a sampling-based strategy is introduced, that approximates the current error distribution with an empirical Probability Density Function (PDF), namely by a set of realizations. In a context where these realizations can be drawn from the current robustness and reliability measure estimations, *e.g.* with GP or bootstrapping techniques, this setting permits to construct non-parametric measure approximations with complex shapes that may exhibit dependencies between some objectives and constraints.

We have proposed two settings for applying this approach, either with joint or disjoint realizations. Intuitively, joint realizations correspond to discretizing a single realization of the robustness and reliability measures random field at different design locations while disjoint realizations treat all designs separately and then require a random field reconstruction on scattered data.

We proposed a first algorithm relying on an approximation scheme with controlled accuracy for drawing large-scale Gaussian random field realizations in the coupled space between design and uncertain parameters. These realizations of the function of interest in the coupled space can then be translated into joint realizations of the robustness and reliability measures. It notably permits a sharper computation of the POP and detects possible correlations between the different objectives and constraints. Joint realizations drawn on multiple designs allow generating Surrogate-Assisting models of the objectives and constraints. The main limitation comes from the number of sampled locations in the coupled space, which increases with both the number of visited designs and the number of Monte Carlo samples required for computing the empirical estimate of the robustness and reliability measures. A specific technique has thus been proposed for limiting the computational burden while controlling the associated accuracy.

Since the construction of a Gaussian random field can be unfeasible in the context of high dimensionality or non-parametric inputs, we proposed a second algorithm, *i.e.* a KDE-based Surrogate-Assisting model, as an extension of the classical heteroscedastic Gaussian process with the capability to take as input non-Gaussian disjoint objective and constraint realizations. Notably, this reconstruction aims to avoid any a priori assumption on the distribution but to properly relapse to the classical heteroscedastic Gaussian Process in the Gaussian case.

The performance of both these techniques has been assessed considering parametric uncertainties, where surrogate models can be constructed in the uncertain dimensions and thus in the coupled space. In this setting, coupled-space strategies reveal very parsimonious, and the Joint and Disjoint characteristic are quantitatively compared, showing slightly faster convergence of Joint techniques. These sampling-based strategies reveal consistently more parsimonious than SABBa throughout these comparisons. The Disjoint SAMATA strategy has also been applied in the context of non-parametric uncertainties through the use of bootstrapping techniques and is shown to outperform a classical Double-Loop approach by several orders of magnitude.

The Joint SAMATA technique has then been applied to the most computationally intensive engineering application of this work, which is the shape optimization of an ORC turbine blade. In addition to the mono-objective problem, we illustrate a bi-objective Taguchi formulation for showing the applicability of the proposed method to multi-objective test-cases. Joint SAMATA has demonstrated a gain of one order of magnitude compared to the literature for returning a robust blade profile.

### VI–1.4 Properties of the proposed framework

Overall the proposed framework features the following properties:

- **Generality**: The proposed algorithm can tackle some classical robustness and reliability measures, and can be easily extended to treat new ones.
- **Constrained multi-objective context**: Multiple objectives and constraints are inherently tackled through the probabilistic Pareto dominance rules and the proposed Pareto Optimal Probability (POP) indicator.
- **High parsimony**: SABBa and sampling-based SAMATA revealed very parsimonious, both on analytical test-cases, where they outperformed conventional approaches, and on engineering applications, with gains of roughly one order of magnitude compared to the literature.
- **Computational impact**: Much effort was dedicated to lowering the computational cost of the proposed algorithms. SABBa is notably extremely cheap to run and thus particularly adapted for solving analytical test-cases. Sampling-based SAMATA, on the other hand, can be quite computationally expensive, notably when using the Metropolis-Hasting strategy for non-parametric uncertainties.
- **Coupling strategy**: SAMATA can be coupled to any optimization technique (NOMAD and NSGA-II in this work) and can make use of any UP strategy that also provides bounds or realizations (GP surrogate modeling and bootstrapping in this work).

## VI–2 Limitations

However, several limitations can be pointed out. First, because the proposed method relies on coupling optimization procedures with UQ techniques, one must have appropriate strategies for the problem at hand to use the SAMATA setting. As a consequence, the performance of the proposed framework will be highly dependent on the chosen optimization and UQ procedures.

In this manuscript, we have applied the proposed strategies using mainly GP surrogate modeling. Such techniques can face significant challenges when dealing with high dimensions or non-stationary functions. The former will require additional dimensionality reduction techniques, such as auto-encoders, latent variables or manifold methods. The latter reflects the difficulty of selecting a covariance kernel and optimizing its hyperparameters. Although universal kernels allow for great generality, it has been shown that kernel selection plays a crucial role in the final accuracy of the model. Similarly, Maximum Likelihood hyperparameters are usually relevant in

a low-dimensional setting. However, they can fail to give a representative response surface in a more complex context. Fully Bayesian treatment or adapted likelihood maximization techniques should then be employed for allowing GP surrogate modeling.

Finally, the relevance of the proposed Pareto Optimal Probability (POP) indicator is inherently linked with the applicability of the Pareto dominance setting. Notably, in a many-objective context, or when the admissible set is very small, this metric will become impractical. In the many-objective case, any design eventually belongs to the Pareto front. It would require reformulation or scalarization for the optimization to be solvable within the proposed framework. When the admissible set is very small, because of specific choices in our Pareto dominance rules, all POP will be equal to zero. In this context, one should rely on the optimization procedure to eventually reach the admissible set.

In light of these remarks, we recommend using the proposed techniques on OUU formulations with one to three objectives and a limited number of constraints. Also, the total number of dimensions (design and uncertainties) should not be much higher than 15 or 20, and complex behaviors (discontinuity, non-stationarity, etc.) should be avoided. To exploit SAMATA outside of these conditions, one must incorporate better surrogate models or dimensionality reduction techniques, as discussed above.

# VI–3 Perspectives

## VI–3.1 Improvements within the SAMATA framework

As an answer to the limitations raised above, several future developments can be considered. First, in terms of overall parsimony, cost improvement could be achieved by improving the Uncertainty Quantification (UQ) techniques associated with specific statistical measures. For example, extreme quantile estimation and refinement should be performed with appropriate techniques such as [Echard et al., 2011] or [Razaaly and Congedo, 2018]. The use of state-of-the-art UQ and optimization techniques is promoted by the proposed framework and should allow tackling uncertainty-based optimization problems with an evergrowing efficiency.

Similarly, in this work, only sequential refinement was performed in practice. As raised in the methodologic part, kriging believer can be exploited to return a set of new locations and to perform multi-point refinement. Note, however, that this approach is greedy and nearly always suboptimal. Often, numerical simulation cannot be intrinsically parallelized. In this setting, to take advantage of the high parallelization capacities of nowadays computational clusters, multi-point refinement can be very attractive. To this extent, parallel refinements in the uncertain space should be compared to parallel UQ procedures over different designs. To the author's opinion, such dynamic allocation of computational resources is critical for minimizing the overall cost.

Besides, one could take advantage of specific characteristics of the Quantities of Interest to further lower the overall computational burden. Notably, some numerical solvers possess an adjoint solver that returns the QoI gradients alongside their values. Taking this information into account could be achieved, for example, with Gradient-Enhanced Kriging (GEK) surrogate models. Similarly, multiple fidelities of the Quantities of Interest (QoI) are often available, for example by modeling in different dimen-

sions (0-D to 3-D) or with formulations of different fidelities (Euler, Reynold-Averaged Numerical Simulation (RANS), Large Eddy Simulation (LES) or Direct Numerical Simulation (DNS)). This multi-fidelity and multilevel capacity can be exploited on-the-fly for obtaining accurate predictions based on a few expensive simulations and many cheap ones. Such an example of multi-fidelity surrogate modeling technique that could be easily employed in the proposed setting is Co-Kriging [Myers, 1982].

## VI–3.2 Ideas

Finally, we argue that some of the proposed metrics, such as the Pareto Optimal Probability (POP) could be employed in a Bayesian Optimization setting. Namely, similarly to the classical Probability of Improvement (PI) acquisition function, the POP yields the probability for a given design to be optimal. However, instead of comparing with a current optimum, the POP indicator compares all individuals within a given population. This formulation has the significant downside that POP does not have an analytical formulation, which means that maximizing this acquisition function is much more involved than maximizing classical criteria such as PI and Expected Improvement (EI). However, the POP metric, unlike any existing acquisition function, can naturally deal with non-Gaussian approximation errors in a multi-objective setting. Similarly to PI, the POP metric suffers from a lack of exploratory behavior, and we propose to multiply the POP value with the standard deviation associated with each individual. In doing so, the so-called POPstd metric is able, as an acquisition function, to give as efficient results as EI on a simple 1-D Gaussian test-case. Mean convergence curves over 50 optimization runs are plotted in Figure VI.1. The lack of exploration of PI and POP makes them worse than random search on average, whereas EI, PIstd and POPstd yield similar efficient results.

To the author's opinion, efficient Optimization Under Uncertainty could be achieved with this strategy, which naturally accounts for multiple objectives and constraints in a non-Gaussian noisy context. Another strategy that should be pursued is to derive Predictive Entropy Search (PES) [Shah and Ghahramani, 2015, Hernández-Lobato et al., 2016a, Garrido-Merchán and Hernández-Lobato, 2019, Hernández-Lobato et al., 2015, Hernández-Lobato et al., 2014] techniques to compute mutual information between the current Pareto-optimal locations and function evaluations in the coupled design/uncertainty space.

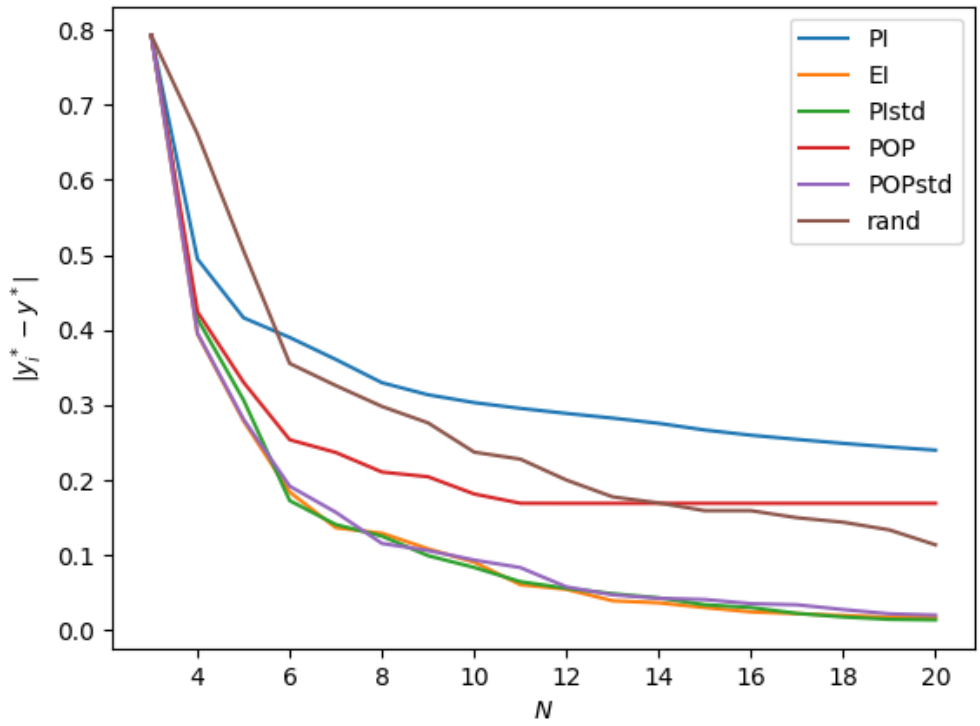



Figure VI.1 Convergence curves for Bayesian Optimization procedure with different acquisition functions.

# Bibliography

[Abdolshah et al., 2018] Abdolshah, M., Shilton, A., Rana, S., Gupta, S., and Venkatesh, S. (2018). Expected hypervolume improvement with constraints. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 3238–3243. IEEE.

[Adje et al., 2013] Adje, A., Bouissou, O., Goubault-Larrecq, J., Goubault, E., and Putot, S. (2013). Static analysis of programs with imprecise probabilistic inputs. In *Working Conference on Verified Software: Theories, Tools, and Experiments*, pages 22–47. Springer.

[Amir et al., 2012] Amir, O., Sigmund, O., Lazarov, B. S., and Schevenels, M. (2012). Efficient reanalysis techniques for robust topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 245:217–231.

[Ammar, 2014] Ammar, K. (2014). *Conception multi-physique et multi-objectif des cœurs de RNR-Na hétérogènes: développement d'une méthode d'optimisation sous incertitudes*. PhD thesis.

[Anjyo and Lewis, 2011] Anjyo, K. and Lewis, J. (2011). Rbf interpolation and gaussian process regression through an rkhs formulation. *Journal of Math-for-Industry*, 3(6):63–71.

[Aronszajn, 1950] Aronszajn, N. (1950). Theory of reproducing kernels. *Transactions of the American mathematical society*, 68(3):337–404.

[Asadpoure et al., 2011] Asadpoure, A., Tootkaboni, M., and Guest, J. K. (2011). Robust topology optimization of structures with uncertainties in stiffness–application to truss structures. *Computers & Structures*, 89(11-12):1131–1141.

[Au and Beck, 2001] Au, S.-K. and Beck, J. L. (2001). Estimation of small failure probabilities in high dimensions by subset simulation. *Probabilistic engineering mechanics*, 16(4):263–277.

[Balesdent et al., 2016] Balesdent, M., Brevault, L., Price, N., Defoort, S., Le Riche, R., Kim, N., Haftka, R., and Bérend, N. (2016). Space vehicle design taking into account multidisciplinary couplings and mixed epistemic/aleatory uncertainties. *Space Engineering: Modeling and Optimization with Case Studies*.

[Barrico and Antunes, 2006] Barrico, C. and Antunes, C. H. (2006). Robustness analysis in multi-objective optimization using a degree of robustness concept. In *2006 IEEE International Conference on Evolutionary Computation*, pages 1887–1892. IEEE.

[Barth, 2016] Barth, T. J. (2016). On the calculation of uncertainty statistics with error bounds for cfd calculations containing random parameters and fields.

[Basseur and Zitzler, 2006] Basseur, M. and Zitzler, E. (2006). Handling uncertainty in indicator-based multiobjective optimization. *International Journal of Computational Intelligence Research*, 2(3):255–272.

[Basudhar et al., 2008] Basudhar, A., Missoum, S., and Sanchez, A. H. (2008). Limit state function identification using support vector machines for discontinuous responses and disjoint failure domains. *Probabilistic Engineering Mechanics*, 23(1):1–11.

[Baudoui, 2012] Baudoui, V. (2012). *Optimisation robuste multiobjectifs par modèles de substitution*. PhD thesis, Toulouse, ISAE.

[Baudoui et al., 2012] Baudoui, V., Klotz, P., Hiriart-Urruty, J.-B., Jan, S., and Morel, F. (2012). Local uncertainty processing (loup) method for multidisciplinary robust design optimization. *Structural and Multidisciplinary Optimization*, 46(5):711–726.

[Bect et al., 2019] Bect, J., Bachoc, F., Ginsbourger, D., et al. (2019). A supermartingale approach to gaussian process based sequential design of experiments. *Bernoulli*, 25(4A):2883–2919.

[Bect et al., 2012] Bect, J., Ginsbourger, D., Li, L., Picheny, V., and Vazquez, E. (2012). Sequential design of computer experiments for the estimation of a probability of failure. *Statistics and Computing*, 22(3):773–793.

[Ben-Tal et al., 2009] Ben-Tal, A., El Ghaoui, L., and Nemirovski, A. (2009). *Robust optimization*, volume 28. Princeton University Press.

[Benassi, 2013] Benassi, R. (2013). *Nouvel algorithme d'optimisation bayésien utilisant une approche Monte-Carlo séquentielle.* PhD thesis.

[Benassi et al., 2011] Benassi, R., Bect, J., and Vazquez, E. (2011). Robust gaussian process-based global optimization using a fully bayesian expected improvement criterion. In *International Conference on Learning and Intelligent Optimization*, pages 176–190. Springer.

[Berlinet and Thomas-Agnan, 2011] Berlinet, A. and Thomas-Agnan, C. (2011). *Reproducing kernel Hilbert spaces in probability and statistics*. Springer Science & Business Media.

[Bertsimas et al., 2011] Bertsimas, D., Brown, D. B., and Caramanis, C. (2011). Theory and applications of robust optimization. *SIAM review*, 53(3):464–501.

[Beyer and Sendhoff, 2007] Beyer, H.-G. and Sendhoff, B. (2007). Robust optimization–a comprehensive survey. *Computer methods in applied mechanics and engineering*, 196(33-34):3190–3218.

[Bourinet, 2018] Bourinet, J.-M. (2018). *Reliability analysis and optimal design under uncertainty-Focus on adaptive surrogate-based approaches*. PhD thesis, Université Clermont Auvergne.

[Boyle and Frean, 2005] Boyle, P. and Frean, M. (2005). Multiple output gaussian process regression.

[Brevault, 2015] Brevault, L. (2015). *Contributions to multidisciplinary design optimization under uncertainty, application to launch vehicle design.* PhD thesis.

[Brochu et al., 2010] Brochu, E., Cora, V. M., and De Freitas, N. (2010). A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599*.

[Buche et al., 2005] Buche, D., Schraudolph, N. N., and Koumoutsakos, P. (2005). Accelerating evolutionary algorithms with gaussian process fitness function models. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 35(2):183–194.

[Bufi, 2016] Bufi, E. (2016). *Robust optimization of ORC turbine expanders*. PhD thesis, Ph. D. thesis, Ecole Nationale Supérieure d'Arts et Métiers, Paris, France.

[Bufi and Cinnella, 2017] Bufi, E. A. and Cinnella, P. (2017). Robust optimization of supersonic orc nozzle guide vanes. In *Journal of Physics: Conference Series*, volume 821, page 012014. IOP Publishing.

[Buyukada, 2017] Buyukada, M. (2017). Probabilistic uncertainty analysis based on monte carlo simulations of co-combustion of hazelnut hull and coal blends: Data-driven modeling and response surface optimization. *Bioresource technology*, 225:106–112.

[Chan, 2013] Chan, A. B. (2013). Multivariate generalized gaussian process models. *arXiv preprint arXiv:1311.0360*.

[Chastaing et al., 2015] Chastaing, G., Gamboa, F., and Prieur, C. (2015). Generalized sobol sensitivity indices for dependent variables: numerical methods. *Journal of Statistical Computation and Simulation*, 85(7):1306–1333.

[Chen et al., 2016] Chen, N., Yu, D., Xia, B., and Ma, Z. (2016). Topology optimization of structures with interval random parameters. *Computer Methods in Applied Mechanics and Engineering*, 307:300–315.

[Chen et al., 1996] Chen, W., Allen, J. K., Tsui, K.-L., and Mistree, F. (1996). A procedure for robust design: minimizing variations caused by noise factors and control factors.

[Chen et al., 2013] Chen, X., Park, E.-J., and Xiu, D. (2013). A flexible numerical approach for quantification of epistemic uncertainty. *Journal of Computational Physics*, 240:211–224.

[Cheng et al., 2006] Cheng, G., Xu, L., and Jiang, L. (2006). A sequential approximate programming strategy for reliability-based structural optimization. *Computers & structures*, 84(21):1353–1367.

[Cheng et al., 2017] Cheng, Q., Wang, S., Yan, C., and Xiao, F. (2017). Probabilistic approach for uncertainty-based optimal design of chiller plants in buildings. *Applied energy*, 185:1613–1624.

[Chevalier et al., 2014a] Chevalier, C., Bect, J., Ginsbourger, D., Vazquez, E., Picheny, V., and Richet, Y. (2014a). Fast parallel kriging-based stepwise uncertainty reduction with application to the identification of an excursion set. *Technometrics*, 56(4):455–465.

[Chevalier et al., 2014b] Chevalier, C., Picheny, V., and Ginsbourger, D. (2014b). Kriginv: An efficient and user-friendly implementation of batch-sequential inversion strategies based on kriging. *Computational Statistics & Data Analysis*, 71:1021–1034.

[Ching and Hsu, 2008] Ching, J. and Hsu, W.-C. (2008). Approximate optimization of systems with high-dimensional uncertainties and multiple reliability constraints. *Computer Methods in Applied Mechanics and Engineering*, 198(1):52–71.

[Chu, 2016] Chu, L. (2016). *Reliability and optimization, application to safety of aircraft structures*. PhD thesis.

[Coelho, 2014] Coelho, R. F. (2014). Probabilistic dominance in multiobjective reliability-based optimization: Theory and implementation. *IEEE Transactions on evolutionary computation*, 19(2):214–224.

[Coelho and Bouillard, 2011] Coelho, R. F. and Bouillard, P. (2011). Multi-objective reliability-based optimization with stochastic metamodels. *Evolutionary computation*, 19(4):525–560.

[Congedo et al., 2013a] Congedo, P., Geraci, G., Abgrall, R., Pediroda, V., and Parussini, L. (2013a). Tsi metamodels-based multi-objective robust optimization. *Engineering Computations (Swansea, Wales)*, 30(8):1032–1053.

[Congedo et al., 2013b] Congedo, P. M., Witteveen, J., and Iaccarino, G. (2013b). A simplex-based numerical framework for simple and efficient robust design optimization. *Computational Optimization and Applications*, 56(1):231–251.

[Contal, 2016] Contal, E. (2016). *Statistical learning approaches for global optimization*. PhD thesis.

[Contal et al., 2013] Contal, E., Buffoni, D., Robicquet, A., and Vayatis, N. (2013). Parallel gaussian process optimization with upper confidence bound and pure exploration. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 225–240. Springer.

[Cortesi, 2018] Cortesi, A. F. (2018). *Predictive numerical simulations for rebuilding freestream conditions in atmospheric entry flows*. PhD thesis, Université de Bordeaux.

[Da Veiga, 2015] Da Veiga, S. (2015). Global sensitivity analysis with dependence measures. *Journal of Statistical Computation and Simulation*, 85(7):1283–1305.

[Da Veiga and Delbos, 2013] Da Veiga, S. and Delbos, F. (2013). Robust optimization for expensive simulators with surrogate models: Application to well placement for oil recovery. *Safety, Reliability, Risk and Life-Cycle Performance of Structures and Infrastructures*, 11:3321–3328.

[Da Veiga and Gamboa, 2013] Da Veiga, S. and Gamboa, F. (2013). Efficient estimation of sensitivity indices. *Journal of Nonparametric Statistics*, 25(3):573–595.

[Da Veiga et al., 2009] Da Veiga, S., Wahl, F., and Gamboa, F. (2009). Local polynomial estimation for sensitivity analysis on models with correlated inputs. *Technometrics*, 51(4):452–463.

[Dai et al., 2012] Dai, H., Zhang, H., Wang, W., and Xue, G. (2012). Structural reliability assessment by local approximation of limit state functions using adaptive markov chain simulation and support vector regression. *Computer-Aided Civil and Infrastructure Engineering*, 27(9):676–686.

[De Boer et al., 2007] De Boer, A., Van der Schoot, M., and Bijl, H. (2007). Mesh deformation based on radial basis function interpolation. *Computers & structures*, 85(11-14):784–795.

[Deb and Gupta, 2006] Deb, K. and Gupta, H. (2006). Introducing robustness in multi-objective optimization. *Evolutionary computation*, 14(4):463–494.

[Dellino et al., 2009] Dellino, G., Kleijnen, J. P., and Meloni, C. (2009). Robust simulation-optimization using metamodels. In *Proceedings of the 2009 Winter Simulation Conference (WSC)*, pages 540–550. IEEE.

[Dellino et al., 2012] Dellino, G., Kleijnen, J. P., and Meloni, C. (2012). Robust optimization in simulation: Taguchi and krige combined. *INFORMS Journal on Computing*, 24(3):471–484.

[Du et al., 2004] Du, X., Sudjianto, A., and Chen, W. (2004). An integrated framework for optimization under uncertainty using inverse reliability strategy. *J. Mech. Des.*, 126(4):562–570.

[Dubourg, 2011] Dubourg, V. (2011). *Méta-modèles adaptatifs pour l'analyse de fiabilité et l'optimisation sous contrainte fiabiliste*. PhD thesis, Clermont-Ferrand 2.

[Dubourg et al., 2011] Dubourg, V., Bourinet, J.-M., and Sudret, B. (2011). Reliability-based design optimization of shells with uncertain geometry using adaptive kriging metamodels. *arXiv preprint arXiv:1104.3479*.

[Dubreuil et al., 2018] Dubreuil, S., Bartoli, N., Gogu, C., Lefebvre, T., and Colomer, J. M. (2018). Extreme value oriented random field discretization based on an hybrid polynomial chaos expansion—kriging approach. *Computer Methods in Applied Mechanics and Engineering*, 332:540–571.

[Dubuisson and Jain, 1994] Dubuisson, M. P. and Jain, A. K. (1994). A modified hausdorff distance for object matching. In *Proceedings of 12th International Conference on Pattern Recognition*, volume 1, pages 566–568 vol.1.

[Duvenaud, 2014] Duvenaud, D. (2014). *Automatic model construction with Gaussian processes*. PhD thesis, University of Cambridge.

[Duvenaud et al., 2011] Duvenaud, D. K., Nickisch, H., and Rasmussen, C. E. (2011). Additive gaussian processes. In *Advances in neural information processing systems*, pages 226–234.

[Duvigneau, 2007] Duvigneau, R. (2007). Aerodynamic shape optimization with uncertain operating conditions using metamodels.

[Duvigneau et al., 2013] Duvigneau, R., Martinelli, M., and Chandrashekarappa, P. (2013). Uncertainty quantification for robust design. *Multidisciplinary Design Optimization in Computational Mechanics*, pages 405–424.

[Echard et al., 2011] Echard, B., Gayton, N., and Lemaire, M. (2011). Ak-mcs: an active learning reliability method combining kriging and monte carlo simulation. *Structural Safety*, 33(2):145–154.

[Economon et al., 2016] Economon, T. D., Mudigere, D., Bansal, G., Heinecke, A., Palacios, F., Park, J., Smelyanskiy, M., Alonso, J. J., and Dubey, P. (2016). Performance optimizations for scalable implicit {RANS} calculations with {SU2}. *Computers & Fluids*, 129:146 – 158.

[El-Beltagy and Keane, 1998] El-Beltagy, M. and Keane, A. (1998). Optimisation for multilevel problems: A comparison of various algorithms. In *Adaptive computing in design and manufacture*, pages 111–120. Springer.

[Eldred et al., 2002] Eldred, M., Giunta, A., Wojtkiewicz, S., and Trucano, T. (2002). Formulations for surrogate-based optimization under uncertainty. In *9th AIAA/ISSMO symposium on multidisciplinary analysis and optimization*, page 5585.

[Elishakoff et al., 1994] Elishakoff, I., Haftka, R., and Fang, J. (1994). Structural design under bounded uncertainty—optimization with anti-optimization. *Computers & structures*, 53(6):1401–1405.

[Emmerich et al., 2002] Emmerich, M., Giotis, A., Özdemir, M., Bäck, T., and Giannakoglou, K. (2002). Metamodel—assisted evolution strategies. In *International Conference on parallel problem solving from nature*, pages 361–370. Springer.

[Emmerich and Klinkenberg, 2008] Emmerich, M. and Klinkenberg, J.-w. (2008). The computation of the expected improvement in dominated hypervolume of pareto front approximations. *Rapport technique, Leiden University*, 34:7–3.

[Emmerich et al., 2006] Emmerich, M. T., Giannakoglou, K. C., and Naujoks, B. (2006). Single-and multiobjective evolutionary optimization assisted by gaussian random field metamodels. *IEEE Transactions on Evolutionary Computation*, 10(4):421–439.

[Eskandari et al., 2007] Eskandari, H., Geiger, C. D., and Bird, R. (2007). Handling uncertainty in evolutionary multiobjective optimization: Spga. In *2007 IEEE Congress on Evolutionary Computation*, pages 4130–4137. IEEE.

[Farin, 2002] Farin, G. (2002). *Curves and Surfaces for CAGD: A Practical Guide*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 5 edition.

[Feliot, 2017] Feliot, P. (2017). *Une approche Bayésienne pour l'optimisation multi-objectif sous contraintes*. PhD thesis.

[Feliot et al., 2017] Feliot, P., Bect, J., and Vazquez, E. (2017). A bayesian approach to constrained single-and multi-objective optimization. *Journal of Global Optimization*, 67(1-2):97–133.

[Ferson et al., 2010] Ferson, S., Van Den Brink, P., Estes, T., Gallagher, K., O'Connor, R., and Verdonck, F. (2010). Bounding uncertainty analyses. *Application of uncertainty analysis to ecological risks of pesticides*.

[Fieldsend and Everson, 2005] Fieldsend, J. E. and Everson, R. M. (2005). Multi-objective optimisation in the presence of uncertainty. In *2005 IEEE Congress on Evolutionary Computation*, volume 1, pages 243–250. IEEE.

[Fusi and Congedo, 2016] Fusi, F. and Congedo, P. M. (2016). An adaptive strategy on the error of the objective functions for uncertainty-based derivative-free optimization. *Journal of Computational Physics*, 309:241–266.

[Gabrel et al., 2014] Gabrel, V., Murat, C., and Thiele, A. (2014). Recent advances in robust optimization: An overview. *European journal of operational research*, 235(3):471–483.

[Gano et al., 2006] Gano, S. E., Renaud, J. E., Martin, J. D., and Simpson, T. W. (2006). Update strategies for kriging models used in variable fidelity optimization. *Structural and Multidisciplinary Optimization*, 32(4):287–298.

[Gardner et al., 2014] Gardner, J. R., Kusner, M. J., Xu, Z. E., Weinberger, K. Q., and Cunningham, J. P. (2014). Bayesian optimization with inequality constraints. In *ICML*, pages 937–945.

[Garrido-Merchán and Hernández-Lobato, 2019] Garrido-Merchán, E. C. and Hernández-Lobato, D. (2019). Predictive entropy search for multi-objective bayesian optimization with constraints. *Neurocomputing*.

[Giannakoglou, 2002] Giannakoglou, K. (2002). Design of optimal aerodynamic shapes using stochastic optimization methods and computational intelligence. *Progress in Aerospace Sciences*, 38(1):43–76.

[Giles, 1990] Giles, M. B. (1990). Nonreflecting boundary conditions for euler equation calculations. *AIAA journal*, 28(12):2050–2058.

[Giotis et al., 2001] Giotis, A., Emmerich, M., Naujoks, B., Giannakoglou, K., and Bäck, T. (2001). Low-cost stochastic optimization for engineering applications. *Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems (EUROGEN-2001)*, pages 361–366.

[Goh and Tan, 2009] Goh, C.-K. and Tan, K. C. (2009). Evolutionary multi-objective optimization in uncertain environments. *Issues and Algorithms, Studies in Computational Intelligence*, 186:5–18.

[Gong et al., 2010] Gong, D.-w., Qin, N.-n., and Sun, X.-y. (2010). Evolutionary algorithms for multi-objective optimization problems with interval parameters. In *2010 IEEE Fifth International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA)*, pages 411–420. IEEE.

[Gorissen et al., 2015] Gorissen, B. L., Yanıkoğlu, İ., and den Hertog, D. (2015). A practical guide to robust optimization. *Omega*, 53:124–137.

[GPy, 2012] GPy (since 2012). GPy: A gaussian process framework in python. http://github.com/SheffieldML/GPy.

[Gretton, 2013] Gretton, A. (2013). Introduction to rkhs, and some simple kernel algorithms. *Adv. Top. Mach. Learn. Lecture Conducted from University College London*, 16.

[Grünewälder et al., 2012] Grünewälder, S., Lever, G., Baldassarre, L., Patterson, S., Gretton, A., and Pontil, M. (2012). Conditional mean embeddings as regressors-supplementary. *arXiv preprint arXiv:1205.4656*.

[Guerra, 2016] Guerra, J. (2016). *Optimisation multi-objectif sous incertitudes de phénomènes de thermique transitoire*. PhD thesis.

[Guest and Igusa, 2008] Guest, J. K. and Igusa, T. (2008). Structural optimization under uncertain loads and nodal locations. *Computer Methods in Applied Mechanics and Engineering*, 198(1):116–124.

[Gutjahr and Pflug, 1996] Gutjahr, W. J. and Pflug, G. C. (1996). Simulated annealing for noisy cost functions. *Journal of global optimization*, 8(1):1–13.

[Haftka et al., 2016] Haftka, R. T., Villanueva, D., and Chaudhuri, A. (2016). Parallel surrogate-assisted global optimization with expensive functions–a survey. *Structural and Multidisciplinary Optimization*, 54(1):3–13.

[Hao et al., 2017] Hao, P., Wang, Y., Liu, C., Wang, B., and Wu, H. (2017). A novel non-probabilistic reliability-based design optimization algorithm using enhanced chaos control method. *Computer Methods in Applied Mechanics and Engineering*, 318:572–593.

[Hennig and Schuler, 2012] Hennig, P. and Schuler, C. J. (2012). Entropy search for information-efficient global optimization. *Journal of Machine Learning Research*, 13(Jun):1809–1837.

[Herdin et al., 2005] Herdin, M., Czink, N., Ozcelik, H., and Bonek, E. (2005). Correlation matrix distance, a meaningful measure for evaluation of non-stationary mimo channels. In *2005 IEEE 61st Vehicular Technology Conference*, volume 1, pages 136–140. IEEE.

[Hernández-Lobato et al., 2016a] Hernández-Lobato, D., Hernandez-Lobato, J., Shah, A., and Adams, R. (2016a). Predictive entropy search for multi-objective bayesian optimization. In *International Conference on Machine Learning*, pages 1492–1501.

[Hernández-Lobato et al., 2016b] Hernández-Lobato, J. M., Gelbart, M. A., Adams, R. P., Hoffman, M. W., and Ghahramani, Z. (2016b). A general framework for constrained bayesian optimization using information-based search. *The Journal of Machine Learning Research*, 17(1):5549–5601.

[Hernández-Lobato et al., 2015] Hernández-Lobato, J. M., Gelbart, M. A., Hoffman, M. W., Adams, R. P., and Ghahramani, Z. (2015). Predictive entropy search for bayesian optimization with unknown constraints.

[Hernández-Lobato et al., 2014] Hernández-Lobato, J. M., Hoffman, M. W., and Ghahramani, Z. (2014). Predictive entropy search for efficient global optimization of black-box functions. In *Advances in neural information processing systems*, pages 918–926.

[Ho and Parpas, 2016] Ho, C. P. and Parpas, P. (2016). Multilevel optimization methods: Convergence and problem structure.

[Hong et al., 2017] Hong, X., Huang, B., Ding, Y., Guo, F., Chen, L., and Ren, L. (2017). Multi-model multivariate gaussian process modelling with correlated noises. *Journal of Process Control*, 58:11–22.

[Hu and Qiu, 2010] Hu, J. and Qiu, Z. (2010). Non-probabilistic convex models and interval analysis method for dynamic response of a beam with bounded uncertainty. *Applied Mathematical Modelling*, 34(3):725–734.

[Hu et al., 2016] Hu, X., Chen, X., Parks, G. T., and Yao, W. (2016). Review of improved monte carlo methods in uncertainty-based design optimization for aerospace vehicles. *Progress in Aerospace Sciences*, 86:20–27.

[Huang et al., 2006] Huang, D., Allen, T. T., Notz, W. I., and Zeng, N. (2006). Global optimization of stochastic black-box systems via sequential kriging meta-models. *Journal of global optimization*, 34(3):441–466.

[Hughes, 2001] Hughes, E. J. (2001). Evolutionary multi-objective ranking with uncertainty and noise. In *International Conference on Evolutionary Multi-Criterion Optimization*, pages 329–343. Springer.

[Ide and Schöbel, 2016] Ide, J. and Schöbel, A. (2016). Robustness for uncertain multi-objective optimization: a survey and analysis of different concepts. *OR spectrum*, 38(1):235–271.

[Ishibuchi et al., 2008] Ishibuchi, H., Tsukamoto, N., and Nojima, Y. (2008). Evolutionary many-objective optimization: A short review. In *2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)*, pages 2419–2426. IEEE.

[Jala, 2013] Jala, M. (2013). *Plans d'expériences adaptatifs pour le calcul de quantiles et application à la dosimétrie numérique*. PhD thesis, PhD thesis, Télécom ParisTech, 2013. 64.

[Janon et al., 2014] Janon, A., Nodet, M., and Prieur, C. (2014). Uncertainties assessment in global sensitivity indices estimation from metamodels. *International Journal for Uncertainty Quantification*, 4(1).

[Janusevskis and Le Riche, 2013] Janusevskis, J. and Le Riche, R. (2013). Simultaneous kriging-based estimation and optimization of mean response. *Journal of Global Optimization*, 55(2):313–336.

[Jensen et al., 2009] Jensen, H., Valdebenito, M., Schuëller, G., and Kusanovic, D. (2009). Reliability-based optimization of stochastic systems using line search. *Computer methods in applied mechanics and engineering*, 198(49-52):3915–3924.

[Jiang et al., 2014] Jiang, X., Gao, J., Hong, X., and Cai, Z. (2014). Gaussian processes autoencoder for dimensionality reduction. In *Pacific-asia conference on knowledge discovery and data mining*, pages 62–73. Springer.

[Jin et al., 2003] Jin, R., Du, X., and Chen, W. (2003). The use of metamodeling techniques for optimization under uncertainty. *Structural and Multidisciplinary Optimization*, 25(2):99–116.

[Jin, 2011] Jin, Y. (2011). Surrogate-assisted evolutionary computation: Recent advances and future challenges. *Swarm and Evolutionary Computation*, 1(2):61–70.

[Jin et al., 2005] Jin, Y., Branke, J., et al. (2005). Evolutionary optimization in uncertain environments-a survey. *IEEE Transactions on evolutionary computation*, 9(3):303–317.

[Jin et al., 2002] Jin, Y., Olhofer, M., and Sendhoff, B. (2002). A framework for evolutionary optimization with approximate fitness functions. *IEEE Transactions on evolutionary computation*, 6(5):481–494.

[Jin and Sendhoff, 2003] Jin, Y. and Sendhoff, B. (2003). Trade-off between performance and robustness: an evolutionary multiobjective approach. In *international conference on Evolutionary Multi-Criterion Optimization*, pages 237–251. Springer.

[Jones et al., 1998] Jones, D. R., Schonlau, M., and Welch, W. J. (1998). Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, 13(4):455–492.

[Jung and Cho, 2004] Jung, H.-S. and Cho, S. (2004). Reliability-based topology optimization of geometrically nonlinear structures with loading and material uncertainties. *Finite elements in analysis and design*, 41(3):311–331.

[Kanagawa and Fukumizu, 2014] Kanagawa, M. and Fukumizu, K. (2014). Recovering distributions from gaussian rkhs embeddings. In *Artificial Intelligence and Statistics*, pages 457–465.

[Kang and Luo, 2009] Kang, Z. and Luo, Y. (2009). Non-probabilistic reliability-based topology optimization of geometrically nonlinear structures using convex models. *Computer Methods in Applied Mechanics and Engineering*, 198(41-44):3228–3238.

[Kang et al., 2011] Kang, Z., Luo, Y., and Li, A. (2011). On non-probabilistic reliability-based design optimization of structures with uncertain-but-bounded parameters. *Structural Safety*, 33(3):196–205.

[Kaymaz, 2005] Kaymaz, I. (2005). Application of kriging method to structural reliability problems. *Structural Safety*, 27(2):133–151.

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

[Keshavarzzadeh et al., 2017] Keshavarzzadeh, V., Fernandez, F., and Tortorelli, D. A. (2017). Topology optimization under uncertainty via non-intrusive polynomial chaos expansion. *Computer Methods in Applied Mechanics and Engineering*, 318:120–147.

[Keshavarzzadeh et al., 2016] Keshavarzzadeh, V., Meidani, H., and Tortorelli, D. A. (2016). Gradient based design optimization under uncertainty via stochastic expansion methods. *Computer Methods in Applied Mechanics and Engineering*, 306:47–76.

[Khosravi et al., 2018] Khosravi, F., Borst, M., and Teich, J. (2018). Probabilistic dominance in robust multi-objective optimization. In *2018 IEEE Congress on Evolutionary Computation (CEC)*, pages 1–6. IEEE.

[Khosravi et al., 2015] Khosravi, F., Müller, M., Glaß, M., and Teich, J. (2015). Uncertainty-aware reliability analysis and optimization. In *2015 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pages 97–102. IEEE.

[Khosravi et al., 2019] Khosravi, F., Raß, A., and Teich, J. (2019). Efficient computation of probabilistic dominance in robust multi-objective optimization. *arXiv preprint arXiv:1910.08413*.

[Klamroth et al., 2013] Klamroth, K., Köbis, E., Schöbel, A., and Tammer, C. (2013). A unified approach for different concepts of robustness and stochastic programming via non-linear scalarizing functionals. *Optimization*, 62(5):649–671.

[Knio and Le Maitre, 2006] Knio, O. and Le Maitre, O. (2006). Uncertainty propagation in cfd using polynomial chaos decomposition. *Fluid dynamics research*, 38(9):616.

[Knowles, 2006] Knowles, J. (2006). Parego: a hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation*, 10(1):50–66.

[Kuhn et al., 2016] Kuhn, K., Raith, A., Schmidt, M., and Schöbel, A. (2016). Bi-objective robust optimisation. *European Journal of Operational Research*, 252(2):418–431.

[Kuroiwa and Lee, 2012] Kuroiwa, D. and Lee, G. M. (2012). On robust multiobjective optimization. *Vietnam J. Math*, 40(2-3):305–317.

[Lachaud and Mansour, 2014] Lachaud, J. and Mansour, N. N. (2014). Porous material analysis toolbox based on openfoam and applications. *Journal of Thermophysics and Heat Transfer*, 28(2):191–202. doi: 10.2514/1.T4262.

[Lagaros et al., 2008] Lagaros, N. D., Garavelas, A. T., and Papadrakakis, M. (2008). Innovative seismic design optimization with reliability constraints. *Computer Methods in Applied Mechanics and Engineering*, 198(1):28–41.

[Lantuéjoul and Desassis, 2012] Lantuéjoul, C. and Desassis, N. (2012). Simulation of a gaussian random vector: a propagative version of the gibbs sampler. In *The 9th international geostatistics congress*, pages 174–181.

[Lawrence, 2004] Lawrence, N. D. (2004). Gaussian process latent variable models for visualisation of high dimensional data. In *Advances in neural information processing systems*, pages 329–336.

[Le Digabel, 2011] Le Digabel, S. (2011). Nomad: Nonlinear optimization with the mads algorithm. 37:44.

[Le Maître and Knio, 2010] Le Maître, O. and Knio, O. M. (2010). *Spectral methods for uncertainty quantification: with applications to computational fluid dynamics*. Springer Science & Business Media.

[Le Riche et al., 2009] Le Riche, R., Picheny, V., Meyer, A., Kim, N.-H., and Ginsbourger, D. (2009). Gears design with shape uncertainties using controlled monte carlo simulations and kriging. In *50th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference 17th AIAA/ASME/AHS Adaptive Structures Conference 11th AIAA No*, page 2257.

[Lee et al., 2002] Lee, J.-O., Yang, Y.-S., and Ruy, W.-S. (2002). A comparative study on reliability-index and target-performance-based probabilistic structural design optimization. *Computers & structures*, 80(3-4):257–269.

[Lee and Park, 2006] Lee, K.-H. and Park, G.-J. (2006). A global robust optimization using kriging based approximation model. *JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing*, 49(3):779–788.

[Li et al., 2006] Li, H.-s., Lü, Z.-z., and Yue, Z.-f. (2006). Support vector machine for structural reliability analysis. *Applied Mathematics and Mechanics*, 27(10):1295–1303.

[Li et al., 2005] Li, M., Azarm, S., and Aute, V. (2005). A multi-objective genetic algorithm for robust design optimization. In *Proceedings of the 7th annual conference on Genetic and evolutionary computation*, pages 771–778. ACM.

[Li et al., 2015] Li, M., Silva, R., Guimarães, F., and Lowther, D. (2015). A new robust dominance criterion for multiobjective optimization. *IEEE Transactions on Magnetics*, 51(3):1–4.

[Li et al., 2017] Li, Y., Wang, X., Wang, L., Fan, W., and Qiu, Z. (2017). Non-probabilistic stability reliability measure for active vibration control system with interval parameters. *Journal of Sound and Vibration*, 387:1–15.

[Liang et al., 2008] Liang, J., Mourelatos, Z. P., and Tu, J. (2008). A single-loop method for reliability-based design optimisation. *International Journal of Product Development*, 5(1-2):76–92.

[Limbourg, 2005] Limbourg, P. (2005). Multi-objective optimization of problems with epistemic uncertainty. In *International Conference on Evolutionary Multi-Criterion Optimization*, pages 413–427. Springer.

[Limbourg and Aponte, 2005] Limbourg, P. and Aponte, D. E. S. (2005). An optimization algorithm for imprecise multi-objective problem functions. In *2005 IEEE Congress on Evolutionary Computation*, volume 1, pages 459–466. IEEE.

[Lin et al., 2012] Lin, G., Engel, D. W., and Eslinger, P. W. (2012). Survey and evaluate uncertainty quantification methodologies. Technical report, Pacific Northwest National Lab.(PNNL), Richland, WA (United States).

[Lombardi, 1998] Lombardi, M. (1998). Optimization of uncertain structures using non-probabilistic models. *Computers & structures*, 67(1-3):99–103.

[Lucor et al., 2007] Lucor, D., Enaux, C., Jourdren, H., and Sagaut, P. (2007). Stochastic design optimization: Application to reacting flows. *Computer methods in applied mechanics and engineering*, 196(49-52):5047–5062.

[Luo et al., 2011] Luo, Y., Li, A., and Kang, Z. (2011). Reliability-based design optimization of adhesive bonded steel–concrete composite beams with probabilistic and non-probabilistic uncertainties. *Engineering Structures*, 33(7):2110–2119.

[March and Willcox, 2012] March, A. and Willcox, K. (2012). Provably convergent multifidelity optimization algorithm not requiring high-fidelity derivatives. *AIAA journal*, 50(5):1079–1089.

[Marelli and Sudret, 2014] Marelli, S. and Sudret, B. (2014). Uqlab: A framework for uncertainty quantification in matlab. In *Vulnerability, uncertainty, and risk: quantification, mitigation, and management*, pages 2554–2563.

[Marrel et al., 2012] Marrel, A., Iooss, B., Da Veiga, S., and Ribatet, M. (2012). Global sensitivity analysis of stochastic computer models with joint metamodels. *Statistics and Computing*, 22(3):833–847.

[Marrel et al., 2015] Marrel, A., Perot, N., and Mottet, C. (2015). Development of a surrogate model and sensitivity analysis for spatio-temporal numerical simulators. *Stochastic environmental research and risk assessment*, 29(3):959–974.

[Martinelli and Duvigneau, 2010] Martinelli, M. and Duvigneau, R. (2010). On the use of second-order derivatives and metamodel-based monte-carlo for uncertainty estimation in aerodynamics. *Computers & fluids*, 39(6):953–964.

[Martinez-Frutos et al., 2016] Martinez-Frutos, J., Herrero-Perez, D., Kessler, M., and Periago, F. (2016). Robust shape optimization of continuous structures via the level set method. *Computer Methods in Applied Mechanics and Engineering*, 305:271–291.

[Marzat et al., 2013] Marzat, J., Walter, E., and Piet-Lahanier, H. (2013). Worst-case global optimization of black-box functions through kriging and relaxation. *Journal of Global Optimization*, 55(4):707–727.

[Maute and Frangopol, 2003] Maute, K. and Frangopol, D. M. (2003). Reliability-based design of mems mechanisms by topology optimization. *Computers & Structures*, 81(8-11):813–824.

[McWilliam, 2001] McWilliam, S. (2001). Anti-optimisation of uncertain structures using interval analysis. *Computers & Structures*, 79(4):421–430.

[Medina and Taflanidis, 2014] Medina, J. C. and Taflanidis, A. A. (2014). Adaptive importance sampling for optimization under uncertainty problems. *Computer Methods in Applied Mechanics and Engineering*, 279:133–162.

[Meneghini et al., 2016] Meneghini, I. R., Guimaraes, F. G., and Gaspar-Cunha, A. (2016). Competitive coevolutionary algorithm for robust multi-objective optimization: The worst case minimization. In *2016 IEEE Congress on Evolutionary Computation (CEC)*, pages 586–593. IEEE.

[Meng et al., 2016] Meng, Z., Zhou, H., Li, G., and Yang, D. (2016). A decoupled approach for non-probabilistic reliability-based design optimization. *Computers & Structures*, 175:65–73.

[Mlakar et al., 2014] Mlakar, M., Tušar, T., and Filipič, B. (2014). Comparing solutions under uncertainty in multiobjective optimization. *Mathematical Problems in Engineering*, 2014.

[Moens and Vandepitte, 2005] Moens, D. and Vandepitte, D. (2005). A survey of non-probabilistic uncertainty treatment in finite element analysis. *Computer methods in applied mechanics and engineering*, 194(12-16):1527–1555.

[Morio, 2012] Morio, J. (2012). Extreme quantile estimation with nonparametric adaptive importance sampling. *Simulation Modelling Practice and Theory*, 27:76–89.

[Most, 2007] Most, T. (2007). An adaptive response surface approach for structural reliability analyses based on support vector machines. In *Proceedings of the eleventh international conference on civil, structural and environmental engineering computing, BHV Topping*.

[Moustapha and Sudret, 2019] Moustapha, M. and Sudret, B. (2019). Surrogate-assisted reliability-based design optimization: a survey and a unified modular framework. *Structural and Multidisciplinary Optimization*, pages 1–20.

[Moustapha et al., 2016] Moustapha, M., Sudret, B., Bourinet, J.-M., and Guillaume, B. (2016). Quantile-based optimization under uncertainties using adaptive kriging surrogate models. *Structural and multidisciplinary optimization*, 54(6):1403–1421.

[Muandet et al., 2017] Muandet, K., Fukumizu, K., Sriperumbudur, B., Schölkopf, B., et al. (2017). Kernel mean embedding of distributions: A review and beyond. *Foundations and Trends® in Machine Learning*, 10(1-2):1–141.

[Myers, 1982] Myers, D. (1982). 1982, myers,d.e., matrix formulation of cokriging mathematical geology 14, 249-257. *Mathematical Geology*, 14:249–257.

[Nakayama et al., 2009] Nakayama, H., Yun, Y., and Yoon, M. (2009). *Sequential approximate multiobjective optimization using computational intelligence*. Springer Science & Business Media.

[Palacios et al., 2013] Palacios, F., Colonno, M. F., Aranake, A. C., Campos, A., Copeland, S. R., Economon, T. D., Lonkar, A. K., Lukaczyk, T. W., Taylor, T. W. R., and Alonso, J. J. (2013). Stanford University Unstructured (SU2): An open-source integrated computational environment for multi-physics simulation and design. In *51st AIAA Aerospace Sciences Meeting and Exhibit*.

[Panunzio et al., 2018] Panunzio, A. M., Cottereau, R., and Puel, G. (2018). Large scale random fields generation using localized karhunen–loève expansion. *Advanced Modeling and Simulation in Engineering Sciences*, 5(1):20.

[Papadrakakis and Lagaros, 2002] Papadrakakis, M. and Lagaros, N. D. (2002). Reliability-based structural optimization using neural networks and monte carlo simulation. *Computer methods in applied mechanics and engineering*, 191(32):3491–3507.

[Pastel, 2012] Pastel, R. (2012). *Estimation de probabilités d'évènements rares et de quantiles extrêmes: applications dans le domaine aérospatial*. PhD thesis, Rennes 1.

[Perrin, 2016] Perrin, G. (2016). Active learning surrogate models for the conception of systems with multiple failure modes. *Reliability Engineering System Safety*, 149:130 – 136.

[Perrin and Defaux, 2019] Perrin, G. and Defaux, G. (2019). Efficient evaluation of reliability-oriented sensitivity indices. *Journal of Scientific Computing*, 79(3):1433–1455.

[Persico et al., 2019] Persico, G., Rodriguez-Fernandez, P., and Romei, A. (2019). High-fidelity shape-optimization of non-conventional turbomachinery by surrogate evolutionary strategies. *Journal of Turbomachinery*, 141(8):081010.

[Picheny, 2009] Picheny, V. (2009). *Improving accuracy and compensating for uncertainty in surrogate modeling*. PhD thesis, University of Florida Gainesville.

[Picheny, 2015] Picheny, V. (2015). Multiobjective optimization using gaussian process emulators via stepwise uncertainty reduction. *Statistics and Computing*, 25(6):1265–1280.

[Picheny et al., 2010a] Picheny, V., Ginsbourger, D., and Richet, Y. (2010a). Noisy expected improvement and on-line computation time allocation for the optimization of simulators with tunable fidelity.

[Picheny et al., 2010b] Picheny, V., Kim, N. H., and Haftka, R. T. (2010b). Application of bootstrap method in conservative estimation of reliability with limited samples. *Structural and Multidisciplinary Optimization*, 41(2):205–217.

[Picheny et al., 2013] Picheny, V., Wagner, T., and Ginsbourger, D. (2013). A benchmark of kriging-based infill criteria for noisy optimization. *Structural and Multidisciplinary Optimization*, 48(3):607–626.

[Pini et al., 2014a] Pini, M., Persico, G., and Dossena, V. (2014a). Robust adjoint-based shape optimization of supersonic turbomachinery cascades. In *ASME Turbo Expo 2014: Turbine Technical Conference and Exposition*, pages V02BT39A043–V02BT39A043. American Society of Mechanical Engineers.

[Pini et al., 2014b] Pini, M., Persico, G., and Dossena, V. (2014b). Robust adjoint-based shape optimization of supersonic turbomachinery cascades. In *ASME Turbo Expo 2014: Turbine Technical Conference and Exposition*, pages V02BT39A043–V02BT39A043. American Society of Mechanical Engineers.

[Pini et al., 2015a] Pini, M., Persico, G., Pasquale, D., and Rebay, S. (2015a). Adjoint method for shape optimization in real-gas flow applications. *ASME Journal of Engineering for Gas Turbines and Power*, 137(3).

[Pini et al., 2015b] Pini, M., Persico, G., Pasquale, D., and Rebay, S. (2015b). Adjoint method for shape optimization in real-gas flow applications. *ASME Journal of Engineering for Gas Turbines and Power*, 137(3).

[Pini et al., 2016] Pini, M., Vitale, S., , Colonna, P., Gori, G., Guardone, A., Economon, T., Alonso, J., and Palacios, F. (2016). Su2: the open-source software for non-ideal compressible flows. In *NICFD 2016: 1st International Seminar on Non-Ideal Compressible-Fluid Dynamics for Propusion & Power*, Varenna, Italy.

[Pisaroni et al., 2019] Pisaroni, M., Nobile, F., and Leyland, P. (2019). Continuation multilevel monte carlo evolutionary algorithm for robust aerodynamic shape design. *Journal of Aircraft*, 56(2):771–786.

[Pourahmadi, 2011] Pourahmadi, M. (2011). Covariance estimation: The glm and regularization perspectives. *Statistical Science*, pages 369–387.

[Powell et al., 2014] Powell, C. E. et al. (2014). Generating realisations of stationary gaussian random fields by circulant embedding. *matrix*, 2(2):1.

[Qiu and Elishakoff, 1998] Qiu, Z. and Elishakoff, I. (1998). Antioptimization of structures with large uncertain-but-non-random parameters via interval analysis. *Computer methods in applied mechanics and engineering*, 152(3-4):361–372.

[Qiu and Wang, 2003] Qiu, Z. and Wang, X. (2003). Comparison of dynamic response of structures with uncertain-but-bounded parameters using non-probabilistic interval analysis method and probabilistic approach. *International Journal of Solids and structures*, 40(20):5423–5439.

[Qiu and Wang, 2005a] Qiu, Z. and Wang, X. (2005a). Parameter perturbation method for dynamic responses of structures with uncertain-but-bounded parameters based on interval analysis. *International journal of solids and structures*, 42(18-19):4958–4970.

[Qiu and Wang, 2005b] Qiu, Z. and Wang, X. (2005b). Two non-probabilistic set-theoretical models for dynamic response and buckling failure measures of bars with unknown-but-bounded initial imperfections. *International Journal of Solids and Structures*, 42(3-4):1039–1054.

[Razaaly, 2019] Razaaly, N. (2019). *Rare event estimation and robust optimization methods with applications to ORC turbine cascade*. PhD thesis, Université Paris-Saclay.

[Razaaly and Congedo, 2018] Razaaly, N. and Congedo, P. M. (2018). Novel algorithm using active metamodel learning and importance sampling: application to multiple failure regions of low probability. *Journal of Computational Physics*, 368:92–114.

[Razaaly et al., 2019a] Razaaly, N., Gori, G., Iaccarino, G., and Congedo, P. (2019a). Optimization of an orc supersonic nozzle under epistemic uncertainties due to turbulence models. In *GPPS 2019*.

[Razaaly et al., 2017] Razaaly, N., Persico, G., and Congedo, P. M. (2017). Uncertainty Quantification of Inviscid Flows Through a Supersonic ORC Turbine Cascade. *Energy Procedia*.

[Razaaly et al., 2019b] Razaaly, N., Persico, G., and Congedo, P. M. (2019b). Impact of geometric, operational, and model uncertainties on the non-ideal flow through a supersonic orc turbine cascade. *Energy*, 169:213–227.

[Ribaud et al., 2020] Ribaud, M., Blanchet-Scalliet, C., Helbert, C., and Gillot, F. (2020). Robust optimization: a kriging-based multi-objective optimization approach. *Reliability Engineering & System Safety*, page 106913.

[Rivier et al., 2019] Rivier, M., Lachaud, J., and Congedo, P. M. (2019). Ablative thermal protection system under uncertainties including pyrolysis gas composition. *Aerospace Science and Technology*, 84:1059–1069.

[Rocquigny, 2009] Rocquigny, E. d. (2009). Quantifying uncertainty in an industrial approach: an emerging consensus in an old epistemological debate. *SAPI EN. S. Surveys and Perspectives Integrating Environment and Society*, (2.1).

[Roe, 1981] Roe, P. L. (1981). Approximate riemann solvers, parameter vectors, and difference schemes. *J. Comput. Phys.*, 43(2):357–372.

[Sabater et al., 2020] Sabater, C., Le Maître, O., Congedo, P., and Görtz, S. (2020). A bayesian approach for quantile optimization problems with high-dimensional uncertainty sources.

[Saltelli et al., 2008] Saltelli, A., Ratto, M., Andres, T., Campolongo, F., Cariboni, J., Gatelli, D., Saisana, M., and Tarantola, S. (2008). *Global sensitivity analysis: the primer*. John Wiley & Sons.

[Sanson, 2019] Sanson, F. (2019). *Estimation du risque humain lié à la retombée d'objets spatiaux sur Terre*. PhD thesis, Université Paris-Saclay.

[Santner et al., 2003] Santner, T. J., Williams, B. J., Notz, W., and Williams, B. J. (2003). *The design and analysis of computer experiments*, volume 1. Springer.

[Schöbi, 2019] Schöbi, R. (2019). Surrogate models for uncertainty quantification in the context of imprecise probability modelling. *IBK Bericht*, 505.

[Schöbi et al., 2016] Schöbi, R., Sudret, B., and Marelli, S. (2016). Rare event estimation using polynomial-chaos kriging. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering*, 3(2):D4016002.

[Schölkopf et al., 2001] Schölkopf, B., Herbrich, R., and Smola, A. J. (2001). A generalized representer theorem. In *International conference on computational learning theory*, pages 416–426. Springer.

[Schuëller and Jensen, 2008] Schuëller, G. I. and Jensen, H. A. (2008). Computational methods in optimization considering uncertainties–an overview. *Computer Methods in Applied Mechanics and Engineering*, 198(1):2–13.

[Seeger, 2000] Seeger, M. (2000). Relationships between gaussian processes, support vector machines and smoothing splines. *Machine Learning*.

[Sejdinovic and Gretton, 2012] Sejdinovic, D. and Gretton, A. (2012). What is an rkhs?

[Selçuklu et al., 2020] Selçuklu, S. B., Coit, D. W., and Felder, F. A. (2020). Pareto uncertainty index for evaluating and comparing solutions for stochastic multiple objective problems. *European Journal of Operational Research*.

[Seshadri et al., 2016] Seshadri, P., Constantine, P., Iaccarino, G., and Parks, G. (2016). A density-matching approach for optimization under uncertainty. *Computer Methods in Applied Mechanics and Engineering*, 305:562–578.

[Shah and Ghahramani, 2015] Shah, A. and Ghahramani, Z. (2015). Parallel predictive entropy search for batch global optimization of expensive objective functions. In *Advances in Neural Information Processing Systems*, pages 3330–3338.

[Shah et al., 2014] Shah, A., Wilson, A., and Ghahramani, Z. (2014). Student-t processes as alternatives to gaussian processes. In *Artificial intelligence and statistics*, pages 877–885.

[Shahriari et al., 2015] Shahriari, B., Swersky, K., Wang, Z., Adams, R. P., and De Freitas, N. (2015). Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175.

[Silverman, 1986] Silverman, B. W. (1986). *Density estimation for statistics and data analysis*, volume 26. CRC press.

[Soares et al., 2009] Soares, G. L., Guimarães, F. G., Maia, C. A., Vasconcelos, J. A., and Jaulin, L. (2009). Interval robust multi-objective evolutionary algorithm. In *2009 IEEE Congress on Evolutionary Computation*, pages 1637–1643. IEEE.

[Soize, 2005] Soize, C. (2005). A comprehensive overview of a non-parametric probabilistic approach of model uncertainties for predictive models in structural dynamics. *Journal of sound and vibration*, 288(3):623–652.

[Song et al., 2013] Song, L., Fukumizu, K., and Gretton, A. (2013). Kernel embeddings of conditional distributions: A unified kernel framework for nonparametric inference in graphical models. *IEEE Signal Processing Magazine*, 30(4):98–111.

[Sudret, 2008] Sudret, B. (2008). Global sensitivity analysis using polynomial chaos expansions. *Reliability engineering & system safety*, 93(7):964–979.

[Sudret et al., 2017] Sudret, B., Marelli, S., and Wiart, J. (2017). Surrogate models for uncertainty quantification: An overview. In *2017 11th European Conference on Antennas and Propagation (EUCAP)*, pages 793–797. IEEE.

[Syberfeldt et al., 2010] Syberfeldt, A., Ng, A., John, R. I., and Moore, P. (2010). Evolutionary optimisation of noisy multi-objective problems using confidence-based dynamic resampling. *European Journal of Operational Research*, 204(3):533–544.

[Taflanidis and Beck, 2008] Taflanidis, A. A. and Beck, J. L. (2008). An efficient framework for optimal robust stochastic system design using stochastic simulation. *Computer Methods in Applied Mechanics and Engineering*, 198(1):88–101.

[Tan and Goh, 2008] Tan, K. C. and Goh, C. K. (2008). Handling uncertainties in evolutionary multi-objective optimization. In *IEEE World Congress on Computational Intelligence*, pages 262–292. Springer.

[Tang and Périaux, 2012] Tang, Z. and Périaux, J. (2012). Uncertainty based robust optimization method for drag minimization problems in aerodynamics. *Computer Methods in Applied Mechanics and Engineering*, 217:12–24.

[Teich, 2001] Teich, J. (2001). Pareto-front exploration with uncertain objectives. In *International Conference on Evolutionary Multi-Criterion Optimization*, pages 314–328. Springer.

[Thore et al., 2017] Thore, C.-J., Holmberg, E., and Klarbring, A. (2017). A general framework for robust topology optimization under load-uncertainty including stress constraints. *Computer Methods in Applied Mechanics and Engineering*, 319:1–18.

[Tipireddy et al., 2019] Tipireddy, R., Barajas-Solano, D. A., and Tartakovsky, A. M. (2019). Conditional karhunen-lo\eve expansion for uncertainty quantification and active learning in partial differential equation models. *arXiv preprint arXiv:1904.08069*.

[Tootkaboni et al., 2012] Tootkaboni, M., Asadpoure, A., and Guest, J. K. (2012). Topology optimization of continuum structures under uncertainty–a polynomial chaos approach. *Computer Methods in Applied Mechanics and Engineering*, 201:263–275.

[Torczon and Trosset, 1998] Torczon, V. and Trosset, M. (1998). Using approximations to accelerate engineering design optimization. In *7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, page 4800.

[Toscano-Palmerin and Frazier, 2018] Toscano-Palmerin, S. and Frazier, P. I. (2018). Bayesian optimization with expensive integrands. *arXiv preprint arXiv:1803.08661*.

[Trappler et al., 2020] Trappler, V., Arnaud, E., Vidard, A., and Debreu, L. (2020). Robust calibration of numerical models based on relative regret.

[ur Rehman et al., 2014] ur Rehman, S., Langelaar, M., and van Keulen, F. (2014). Efficient kriging-based robust optimization of unconstrained problems. *Journal of Computational Science*, 5(6):872–881.

[Valdebenito and Schuëller, 2010a] Valdebenito, M. and Schuëller, G. (2010a). Reliability-based optimization considering design variables of discrete size. *Engineering Structures*, 32(9):2919–2930.

[Valdebenito and Schuëller, 2011] Valdebenito, M. and Schuëller, G. (2011). Efficient strategies for reliability-based optimization involving non-linear, dynamical structures. *Computers & Structures*, 89(19-20):1797–1811.

[Valdebenito and Schuëller, 2010b] Valdebenito, M. A. and Schuëller, G. I. (2010b). A survey on approaches for reliability-based optimization. *Structural and Multidisciplinary Optimization*, 42(5):645–663.

[Villemonteix, 2008] Villemonteix, J. (2008). *Optimisation de fonctions coûteuses Modèles gaussiens pour une utilisation efficace du budget d'évaluations: théorie et pratique industrielle*. PhD thesis.

[Villemonteix et al., 2009a] Villemonteix, J., Vazquez, E., Sidorkiewicz, M., and Walter, E. (2009a). Global optimization of expensive-to-evaluate functions: an empirical comparison of two sampling criteria. *Journal of Global Optimization*, 43(2-3):373–389.

[Villemonteix et al., 2009b] Villemonteix, J., Vazquez, E., and Walter, E. (2009b). An informational approach to the global optimization of expensive-to-evaluate functions. *Journal of Global Optimization*, 44(4):509.

[Vinokur and Montagné, 1990] Vinokur, M. and Montagné, J. L. (1990). Generalized flux-vector splitting and roe average for an equilibrium real gas. *J. Comput. Phys.*, 89:276.

[Vitale et al., 2017] Vitale, S., Albring, T. A., Pini, M., Gauger, N. R., and Colonna, P. (2017). Fully turbulent discrete adjoint solver for non-ideal compressible flow applications. *Journal of the Global Power and Propulsion Society*, 1:Z1FVOI.

[Vitale et al., 2015] Vitale, S., Gori, G., Pini, M., Guardone, A., Economon, T. D., Palacios, F., Alonso, J. J., and Colonna, P. (2015). Extension of the SU2 open source CFD code to the simulation of turbulent flows of fluids modelled with complex thermophysical laws. Number AIAA Paper 2015-2760.

[Wagner et al., 2010] Wagner, T., Emmerich, M., Deutz, A., and Ponweiser, W. (2010). On expected-improvement criteria for model-based multi-objective optimization. In *International Conference on Parallel Problem Solving from Nature*, pages 718–727. Springer.

[Wang et al., 2017] Wang, C., Qiu, Z., Xu, M., and Li, Y. (2017). Novel reliability-based optimization method for thermal structure with hybrid random, interval and fuzzy parameters. *Applied Mathematical Modelling*, 47:573–586.

[Wang and Peng, 2014] Wang, X. and Peng, Z. (2014). Method of moments for estimating uncertainty distributions. *Journal of Uncertainty Analysis and Applications*, 2(1):5.

[Wang and Jegelka, 2017] Wang, Z. and Jegelka, S. (2017). Max-value entropy search for efficient bayesian optimization. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3627–3635. JMLR. org.

[Williams et al., 2000] Williams, B. J., Santner, T. J., and Notz, W. I. (2000). Sequential design of computer experiments to minimize integrated response functions. *Statistica Sinica*, pages 1133–1152.

[Williams and Rasmussen, 2006] Williams, C. K. and Rasmussen, C. E. (2006). *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA.

[Williams and Seeger, 2001] Williams, C. K. and Seeger, M. (2001). Using the nyström method to speed up kernel machines. In *Advances in neural information processing systems*, pages 682–688.

[Wilson et al., 2018] Wilson, J., Hutter, F., and Deisenroth, M. (2018). Maximizing acquisition functions for bayesian optimization. In *Advances in Neural Information Processing Systems*, pages 9884–9895.

[Wu et al., 2016] Wu, J., Gao, J., Luo, Z., and Brown, T. (2016). Robust topology optimization for structures under interval uncertainty. *Advances in Engineering Software*, 99:36–48.

[Wu et al., 2017] Wu, J., Luo, Z., Li, H., and Zhang, N. (2017). Level-set topology optimization for mechanical metamaterials under hybrid uncertainties. *Computer Methods in Applied Mechanics and Engineering*, 319:414–441.

[Wu et al., 2014] Wu, J., Luo, Z., Zhang, Y., and Zhang, N. (2014). An interval uncertain optimization method for vehicle suspensions using chebyshev metamodels. *Applied Mathematical Modelling*, 38(15-16):3706–3723.

[Xie et al., 2016] Xie, Q., Wang, J., Lu, S., and Hensen, J. L. (2016). An optimization method for the distance between exits of buildings considering uncertainties based on arbitrary polynomial chaos expansion. *Reliability Engineering & System Safety*, 154:188–196.

[Xu et al., 2016] Xu, J., Zhang, W., and Sun, R. (2016). Efficient reliability assessment of structural dynamic systems with unequal weighted quasi-monte carlo simulation. *Computers & Structures*, 175:37–51.

[Yang, 2010] Yang, X.-S. (2010). *Engineering optimization: an introduction with metaheuristic applications*. John Wiley & Sons.

[Youn and Choi, 2004] Youn, B. D. and Choi, K. K. (2004). A new response surface methodology for reliability-based design optimization. *Computers & structures*, 82(2-3):241–256.

[Youn et al., 2005] Youn, B. D., Choi, K. K., and Du, L. (2005). Enriched performance measure approach for reliability-based design optimization. *AIAA journal*, 43(4):874–884.

[Youn et al., 2003] Youn, B. D., Choi, K. K., and Park, Y. H. (2003). Hybrid analysis method for reliability-based design optimization. *Journal of Mechanical Design*, 125(2):221–232.

[Zahir and Gao, 2013] Zahir, M. K. and Gao, Z. (2013). Variable-fidelity optimization with design space reduction. *Chinese Journal of Aeronautics*, 26(4):841–849.

[Zhang et al., 2017] Zhang, J., Taflanidis, A., and Medina, J. (2017). Sequential approximate optimization for design under uncertainty problems utilizing kriging metamodeling in augmented input space. *Computer Methods in Applied Mechanics and Engineering*, 315:369–395.

[Zhao et al., 2016] Zhao, Q., Chen, X., Ma, Z., and Lin, Y. (2016). A comparison of deterministic, reliability-based topology optimization under uncertainties. *Acta Mechanica Solida Sinica*, 29(1):31–45.

[Zhu et al., 2017] Zhu, H., Tian, H., and Cai, G. (2017). Hybrid uncertainty-based design optimization and its application to hybrid rocket motors for manned lunar landing. *Chinese Journal of Aeronautics*, 30(2):719–725.

[Žilinskas, 2010] Žilinskas, A. (2010). On similarities between two models of global optimization: statistical models and radial basis functions. *Journal of Global Optimization*, 48(1):173–182.

[Zitzler et al., 2000] Zitzler, E., Kalyanmoy, D., and Thiele, L. (2000). Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary computation*, 8(2):173–195.

ECOLE
DOCTORALE
DE MATHEMATIQUES
HADAMARD

**Titre :** Méthodes à faible coût pour l'optimisation sous incertitude multi-objectif sous contrainte

**Mots clés :** Incertitudes, Optimisation, Robustesse et fiabilité, Statistiques, Modèle de substitution

**Résumé :** L'optimisation sous incertitude est un axe de recherche fondamental chez de nombreuses entreprises, de par l'accroissement de la puissance de calcul et la recherche continuelle d'efficience, de fiabilité et d'optimalité des coûts. Parmi les difficultés associées, on peut citer, entre autres, la bonne formulation d'une métrique d'optimisation pour le problème d'intérêt et la recherche d'un compromis idéal entre parcimonie et précision lorsque des simulations complexes et coûteuses sont impliquées. Cet travail vise à traiter les problèmes d'optimisation multi-objectif sous contrainte, où les objectifs et contraintes tiennent compte des paramètres incertains sous la forme, par exemple, de moments statistiques ou de quantiles.
Cette thèse repose sur deux idées principales. Premièrement, la précision d'approximation des objectifs et contraintes à chaque point devrait être guidée par la probabilité de ce point d'être non-dominé. Cela permet de réduire l'effort alloué aux points dont l'optimalité est peu probable. Pour cela, nous introduisons le concept de dominance probabiliste pour l'optimisation multi-objectif sous contrainte, basé sur le concept de probabilité d'être Pareto-optimal (POP). Deuxièmement, dans le but d'accélérer le processus d'optimisation, ces approximations et leur erreur associée peuvent être exploitée afin de construire un modèle prédictif des objectifs et contraintes. Ces deux techniques, approximations des objectifs et contraintes avec une précision adaptative et assistance par métamodèle, sont au coeur de l'algorithme proposé, appelé SAMATA. Ce dernier est flexible en termes de métriques utilisées et se révèle très parcimonieux. Cet algorithme est d'ailleurs applicable avec des méthodes génériques d'optimisation.
Nous explorons ensuite l'influence de la distribution de l'erreur d'approximation. Une première supposition simplificatrice et conservative consiste à considérer cette distribution uniforme. La formulation proposée devient une approche par boîtes, ou l'erreur est réduite à un intervalle (en mono-dimensionel) ou un produit d'intervalles (en multi-dimensionel) autour de la valeur estimée, ce qui permet naturellement le calcul d'un front de Pareto imprécis. Cette approche est couplée à une assistance par métamodèle, construit directement sur les objectifs et contraintes. Sous quelques hypothèses, nous étudions la convergence du front de Pareto approximé vers le front réel.
Par la suite, nous proposons une approximation non-paramétrique par échantillonnage de la distribution de l'erreur. Un premier algorithme repose sur une approximation avec contrôle de précision permettant de tirer des réalisations de champ Gaussien sur un très grand nombre de points dans l'espace couplé entre variables de contrôle et incertaines. Cela permet notamment des comparaisons probabilistes plus précises et la capture de corrélations entre les différents objectifs et contraintes. Des réalisations jointes sont tirées à différents points afin de générer un métamodèle pour assister l'optimisation. La construction de ce champ Gaussien dans l'espace couplé peut se révéler infaisable lorsque la dimensionalité est trop élevée ou que les incertitudes sont non-paramétriques. Nous proposons donc aussi la construction d'un métamodèle, construit sur des réalisations disjointes des objectifs et contraintes, comme extension des processus Gaussiens hétéroscedastiques classiques.
Les variantes proposées sont testées sur plusieurs cas analytiques d'optimisation sous incertitudes, et sont comparées à une approche par métamodèle a priori à l'aide d'un indicateur probabiliste de distance de Hausdorff au front de Pareto exact. La méthode est finalement mise en pratique sur plusieurs applications pour l'ingénierie : la charpente à deux barres, un système de protection thermique pour la rentrée atmosphérique et les pâles d'une turbine à cycle de Rankine organique.

**Title :** Low-cost methods for constrained multi-objective optimization under uncertainty

**Keywords :** Uncertainty, Optimization, Robustness and reliability, Statistics, Surrogate model

**Abstract :** Optimization Under Uncertainty is a fundamental axis of research in many companies nowadays, due to both the evergrowing computational power available and the need for efficiency, reliability and cost optimality. Among others, some challenges are the formulation of a suitable metric for the optimization problem of interest and the search for an ideal trade-off between computational cost and accuracy in the case of problems involving complex and expensive numerical solvers. The targeted class of problem here is constrained multi-objective optimization where fitness functions are uncertainty-driven metrics, such as statistical moments or quantiles.
This thesis relies on two main ideas. First, the accuracy for approximating the objectives and constraints at a given design should be driven by the probability for this design of being non-dominated. This choice permits to reduce the effort for evaluating designs which are unlikely to be optimal. To this extent, we introduce the concept of probabilistic dominance for constrained multi-objective optimization under uncertainty through the computation of the so-called Pareto-Optimal Probability (POP). Secondly, these approximated evaluations and their associated errors can be used to construct a predictive representation of the objectives and constraints over the whole design space to accelerate the optimization process. Overall, the approximation of different uncertainty-based metrics with tunable accuracy and the use of a Surrogate-Assisting strategy are the main ingredients of the proposed algorithm, called SAMATA. This approach is flexible in terms of metrics formulations and reveals very parsimonious. Note that this algorithm is applicable with generic optimization methods.
This thesis then explores the influence of the error distribution on the algorithm performance. We first make a simplifying and conservative assumption by considering a Uniform distribution of the error. In this case, the proposed formulation yields a Bounding-Box approach, where the estimation error can be regarded with the abstraction of an interval (in one-dimensional problems) or a product of intervals (in multi-dimensional problems) around the estimated value, naturally allowing for the computation of an approximated Pareto front. This approach is then supplemented by a Surrogate-Assisting strategy that directly estimates the objective and constraint values. Under some hypotheses, we study the convergence properties of the method in terms of the distance between the approximated Pareto front and the true continuous one.
Secondly, we propose to compute non-parametric approximations of the error distributions with a sampling-based technique. We propose a first algorithm relying on an approximation scheme with controlled accuracy for drawing large-scale Gaussian random field realizations in the coupled space between design and uncertain parameters. It notably permits a sharper computation of the POP and detects possible correlations between the different objectives and constraints. Joint realizations can be drawn on multiple designs in order to generate Surrogate-Assisting models of the objectives and constraints. Since the construction of a Gaussian random field can be hard in the context of high dimensionality or non-parametric inputs, we also propose a KDE-based Surrogate-Assisting model as an extension of the classical heteroscedastic Gaussian process, with the capability to take as input disjoint objective and constraint realizations.
We assess the proposed variants on several analytical uncertainty-based optimization test-cases with respect to an a priori metamodel approach by computing a probabilistic modified Hausdorff distance to the exact Pareto optimal set. The method is then employed on several engineering applications: the two-bar truss, a thermal protection system for atmospheric reentry and the blades of an Organic Rankine Cycle turbine.

**Institut Polytechnique de Paris**
91120 Palaiseau, France