Table des matières

Chapitre 1

Introduction

1.1 Contexte et formalisme

Ce travail de thèse s'intéresse au problème de l'*optimisation globale* d'une fonction *coûteuse*, dans un cadre bayésien. Il s'agit de déterminer le maximum d'une fonction sur un domaine généralement compact et continu. Nous disons qu'une fonction est coûteuse lorsque l'évaluation de celle-ci en un point de son domaine de définition nécessite l'utilisation de ressources importantes. Dans l'industrie, il s'agit généralement de ressources informatiques, lorsque la fonction à optimiser correspond à une grandeur d'intérêt calculée au moyen de simulations numériques. Certaines simulations numériques peuvent durer des heures ou des jours sur des moyens de calcul performants, ce qui implique le plus souvent un coût financier conséquent également.

Les méthodes d'optimisation globale forment un domaine assez vaste. Une vue d'ensemble du domaine peut être appréhendée à partir d'ouvrages de références tels que Törn et Zilinskas (1989) ; Pintér (1996) ; Zhigljavsky et Zilinskas (2007) ; Conn et al. (2009) ; Tenne et Goh (2010). Parmi les méthodes classiques d'optimisation globale, citons par exemple l'existence d'algorithmes du type séparation et évaluation (*branch and bound* en anglais) ou bien l'algorithme DIRECT introduit par Jones et al. (1993). D'autres approches comme le recuit simulé ou les algorithmes génétiques font quant à elles intervenir des processus d'exploration par simulation de variables aléatoires, et donnent lieu

à une littérature spécifique (Kirkpatrick et al., 1983 ; Glover et Laguna, 1997 ; Storn et Price, 1997 ; Beyer et Schwefel, 2002). Un algorithme d'optimisation globale nécessite généralement un nombre d'évaluations élevé afin d'obtenir une optimisation de bonne qualité, comme mis en évidence par de nombreux résultats numériques (voir, par exemple, les travaux de Egea Larrosa, 2008). Pourtant, dans le contexte de l'optimisation d'une fonction coûteuse, il apparaît évident que l'optimisation ne peut être effectuée qu'à l'aide d'un nombre limité d'évaluations. Nous parlerons de budget d'évaluation pour désigner le nombre maximal d'évaluations qu'il est possible de conduire.

D'un point de vue formel, le problème considéré peut s'exprimer de la façon suivante. Nous considérons une *fonction objectif* f définie sur un domaine compact $\mathbb{X} \subset \mathbb{R}^d$ et à valeurs dans $\mathbb{R}$, dont nous cherchons à déterminer le maximum [1] (voir figure 1.1)

$$M = \max_{x \in \mathbb{X}} f(x), \tag{1.1}$$

ainsi que le(s) maximiseur(s)

$$x^\star \in \operatorname*{argmax}_{x \in \mathbb{X}} f(x). \tag{1.2}$$

L'optimisation de la fonction f se fait de façon séquentielle. Nous supposons disposer d'un budget de N évaluations et notre objectif est de choisir les points d'évaluation $X_1, X_2, \ldots, X_N \in \mathbb{X}$ afin d'obtenir la valeur la plus faible possible de $M - M_N$, avec $M_N = \max(f(X_1), f(X_2), \ldots, f(X_N))$. Lorsque $n < N$ résultats d'évaluation sont connus, nous nous intéressons au choix du point X_{n+1}, choix effectué à partir de l'ensemble de l'information disponible $\mathcal{F}_n = (X_1, f(X_1), X_2, f(X_2), \ldots, X_n, f(X_n))$. Nous considérons ici que le gradient de f ne fait pas partie de l'information disponible. Le choix successif des points d'échantillonnage se fait à l'aide d'une fonction J, définie sur $\mathbb{X}$ dépendant de l'information disponible, et que nous nommons *critère d'échantillonnage*. La nouvelle évaluation est alors faite en un point qui maximise ce critère J,

$$X_{n+1} \in \operatorname*{argmax}_{x \in \mathbb{X}} J\left(x; \mathcal{F}_n\right). \tag{1.3}$$

1. il serait rigoureusement équivalent de chercher à minimiser la fonction, il suffirait pour cela de considérer la maximisation de la fonction $-f$

Par la suite, nous utilisons la notation plus compacte $J_n(.) = J(.; \mathcal{F}_n)$. Concrètement, le cœur de cette approche est de remplacer le problème d'optimisation initial coûteux par une succession de problèmes d'optimisation, globaux eux aussi, mais au coût moindre. Le succès de ces algorithmes repose donc en partie sur un coût de maximisation de J suffisamment faible en comparaison du coût d'évaluation de f.

Il est à noter également que nous nous restreignons par la suite aux seuls cas d'optimisation de fonctions non-bruitées. Autrement dit, les résultats d'évaluation de f sont supposés déterministes.

1.2 Construction bayésienne d'un critère d'échantillonnage

Nous considérons ici une approche *bayésienne* qui consiste à affecter à f un *a priori* sous la forme d'un processus aléatoire ξ. L'idée est ensuite de conditionner ξ par les résultats d'évaluations bien choisies, afin de guider l'échantillonnage des futurs points (voir, par exemple les articles de Mockus et al., 1978 ; Jones et al., 1998 ; Gutmann, 2001). Plusieurs articles (Betrò, 1991 ; Mockus, 1994 ; Jones, 2001 ; Brochu et al., 2010) offrent un panorama de l'approche bayésienne pour l'optimisation globale. Historiquement, le mouvement brownien a été le premier modèle utilisé dans le domaine de l'optimisation bayésienne (Kushner, 1964), puis diverses heuristiques se sont développées, en particulier pour généraliser l'approche à des fonctions à plusieurs variables (Mockus, 1989 ; Perttunen, 1991). Le modèle est par la suite choisi gaussien par Jones et al. (1998) et, dès lors, dans la plupart des publications concernant cette approche, que ce soit pour des variantes (Williams et al., 2000 ; Huang et al., 2006), ou pour un objectif autre (études comparatives, nouveaux algorithmes...) dans les articles de Jones (2001) ; Sasena (2002) ; Villemonteix (2008).

Le choix d'un processus gaussien comme *a priori* sur f permet de mener très simplement les opérations de conditionnement en utilisant le principe du krigeage. Le krigeage est une technique initialement utilisée en géostatistique,

fondée sur les travaux de l'ingénieur minier sud-africain Krige (Krige, 1951), et développée par le mathématicien français Matheron dans le courant des années 60 (Matheron, 1963). Pour plus d'information sur la théorie, se référer à des ouvrages de référence (Cressie, 1993 ; Stein, 1999 ; Vazquez, 2005 ; Chilès et Delfiner, 2012).

Le krigeage permet de calculer la moyenne $\widehat{\xi}_n$ du processus ξ conditionné par des évaluations $\xi(x_1) = f(x_1)$, $\xi(x_2) = f(x_2)$, $\ldots$, $\xi(x_n) = f(x_n)$. Lorsque $\widehat{\xi}_n$ a été calculé, une approche naïve pour construire un critère d'échantillonnage en vue d'optimiser f, serait de choisir comme futur point d'évaluation le maximiseur de $\widehat{\xi}_n$. Les points d'évaluation auront tendance à s'accumuler au voisinage du maximiseur courant de $\widehat{\xi}_n$, tandis que des zones entières resteront inexplorées, ce qui n'est pas satisfaisant pour un algorithme d'optimisation global. La figure 1.2 illustre ce phénomène d'accumulation et met en évidence que l'optimisation ne peut être efficace qu'à la condition de faire un *compromis* entre une recherche au voisinage des évaluations les plus prometteuses et une recherche dans les zones les plus inexplorées (dans lesquelles la présence d'un maximum global ne peut être exclue). Ce compromis entre recherche locale et globale est également parfois appelé, en particulier dans la littérature relative aux « bandits » (Auer et al., 2002) compromis exploitation/exploration. Pour le satisfaire, il ne faut pas seulement considérer la moyenne conditionnée $\widehat{\xi}_n$ de ξ mais également une estimation de l'erreur de prédiction par le calcul de la variance s_n^2 en chaque point de $\mathbb{X}$. La figure 1.3 illustre un exemple de prédiction par krigeage où le prédicteur ainsi que les intervalles de confiance à plus ou moins $1.96 s_n$ sont mis en évidence.

Il existe de nombreux critères dans la littérature qui prennent en compte l'erreur de prédiction. Parmi les plus célèbres, nous pouvons évoquer la probabilité d'amélioration, introduite par Kushner (1964), ainsi que des variantes diverses (extension au cas multidimensionnel, ajout de paramètres type seuil) décrites par Perttunen (1991) ; Zilinskas (1992) ; Jones (2001), la borne supérieure de confiance (UCB) (Cox et John, 1997 ; Srinivas et al., 2010) ou encore l'entropie conditionnelle des minimiseurs globaux (ECM) (Villemonteix, 2008). Tous ces critères sont, d'une façon ou d'une autre, construits afin d'atteindre

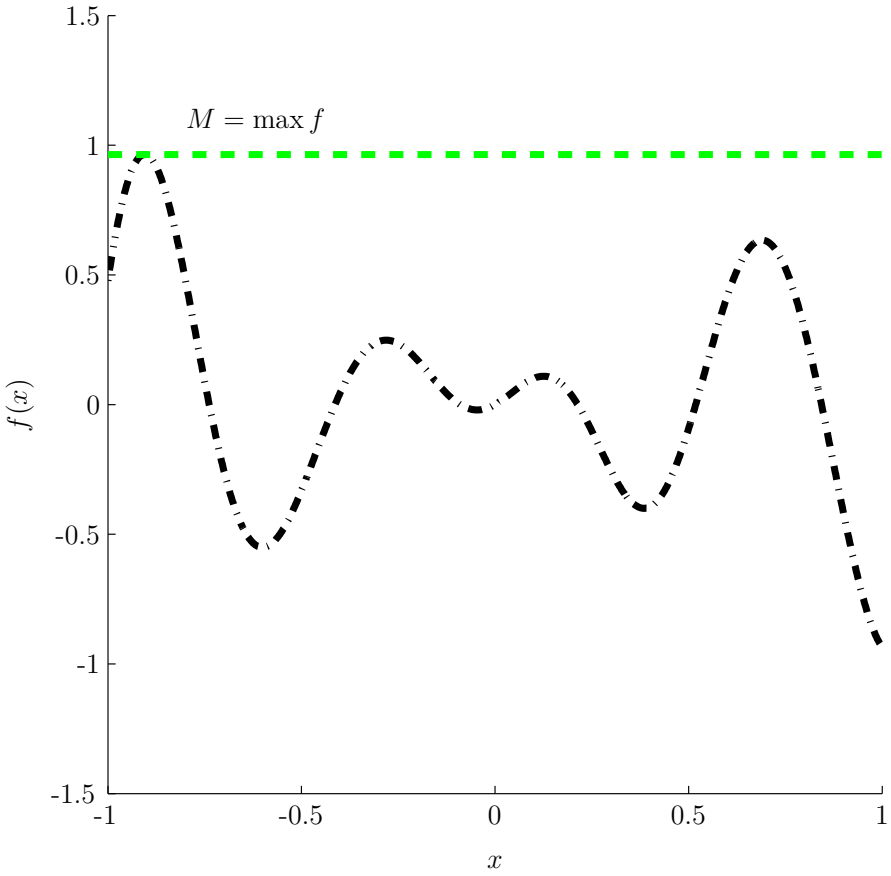
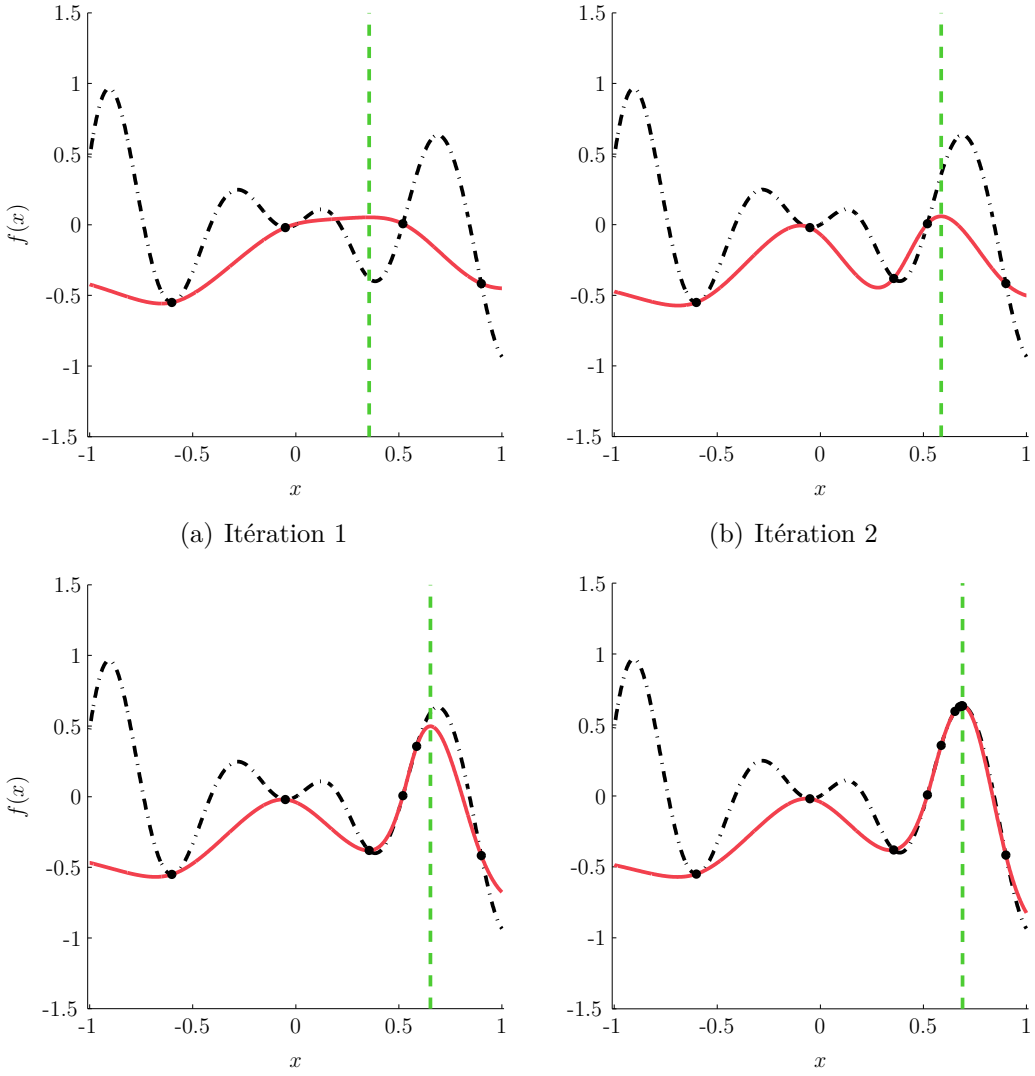


Figure 1.1 – Exemple de fonction objectif (trait mixte). Le niveau correspondant au maximum de la fonction est représenté par un trait pointillé.



(a) Itération 1

(b) Itération 2

(c) Itération 3

(d) Itération 8.

Figure 1.2 – Illustration d'une optimisation naïve par maximisation du prédicteur, à l'itération 1. La fonction objectif est en traits mixtes et le prédicteur issu de la théorie du krigeage en trait plein. Les points du domaine déjà observés sont représentés en noir. Le trait vertical vert représente le maximum du prédicteur. À l'itération 8, fig. 1.2(d), se remarque une accumulation des points d'échantillonnage au voisinage d'un maximiseur local et une exploration faible du domaine de recherche.

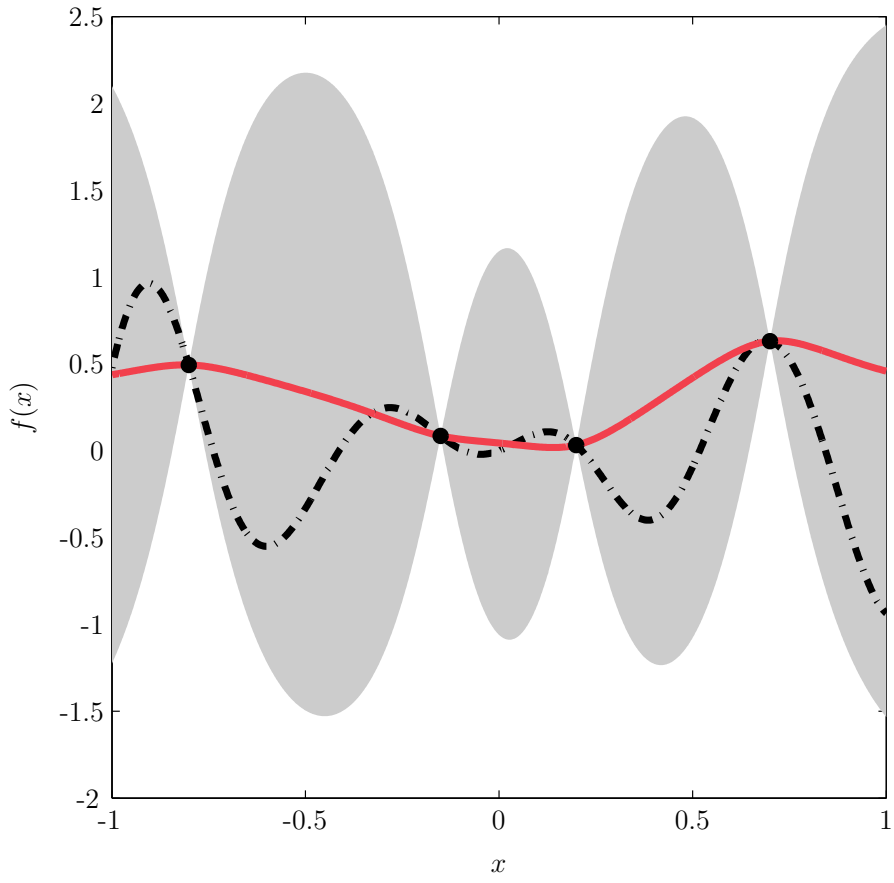


Figure 1.3 – Illustration d'une prédiction par krigeage. La fonction objectif est en traits mixtes et la prédiction par krigeage en trait plein. Les intervalles de confiance calculés à partir de l'écart type de l'erreur de prédiction correspondent aux zones grises. Les points du domaine déjà observés sont représentés en noir.

un compromis entre *exploitation* et *exploration*. L'espérance de l'amélioration, aussi connue sous le nom critère EI pour *Expected Improvement*, est un critère d'échantillonnage introduit par Mockus et al. (1978) et popularisé à partir de l'article de Jones et al. (1998). Il présente, à notre sens, certains avantages sur les autres critères cités ci-dessus (ce qui est discuté en détails au chapitre 2), et c'est la raison pour laquelle il occupe une place privilégiée dans la suite du manuscrit. Par la suite, la notation ρ_n désigne le critère EI construit à partir de $\widehat{\xi}_n$ et s_n. Le fonctionnement d'un algorithme d'optimisation utilisant l'EI comme critère d'échantillonnage est présenté à la figure 1.4.

La valeur du critère EI dépend de l'*a priori* gaussien ξ qui est déterminé de façon unique par sa moyenne et sa fonction de covariance. En pratique, les fonctions de moyenne et de covariance ne sont pas supposées connues et sont estimées à partir des données. Les fonctions de moyenne et de covariance sont généralement choisies dans une classe paramétrée. Le cas de la moyenne se traite simplement en considérant des formes paramétriques linéaires et un *a priori* uniforme sur les paramètres. Nous utilisons ensuite le principe du krigeage universel qui permet de calculer une estimation, conditionnellement aux données, de la moyenne inconnue. En ce qui concerne la fonction de covariance, plusieurs approches sont possibles. Par la suite le vecteur des paramètres de la covariance sera noté $\theta \in \mathbb{R}^s$, avec $s \in \mathbb{N}$. L'approche utilisée par l'algorithme EGO (*efficient global optimization*), introduit par Jones et al. (1998) et dont le critère d'échantillonnage est l'EI, consiste à substituer au θ inconnu son estimateur du maximum de vraisemblance $\widehat{\theta}_n$. Cette estimation des paramètres à l'aide du maximum de vraisemblance (méthode MV) se retrouve dans de nombreuses publications (Schonlau et Welch, 1996 ; Schonlau, 1997 ; Schonlau et al., 1997 ; Jones et al., 1998 ; Sasena, 2002). Une démarche semblable mais préférant le maximum de la loi *a posteriori* à celui de la vraisemblance est également envisageable, comme évoqué dans Lizotte et al. (2012), par exemple. Cette valeur $\widehat{\theta}_n$ peut ainsi être directement injectée dans l'expression du critère d'échantillonnage, ce qui est alors qualifié de méthode par substitution (ou *plug-in* en anglais). La figure 1.5 décrit le fonctionnement de l'algorithme EGO, tandis que la figure 1.6 donne un exemple concret de son déroulement.

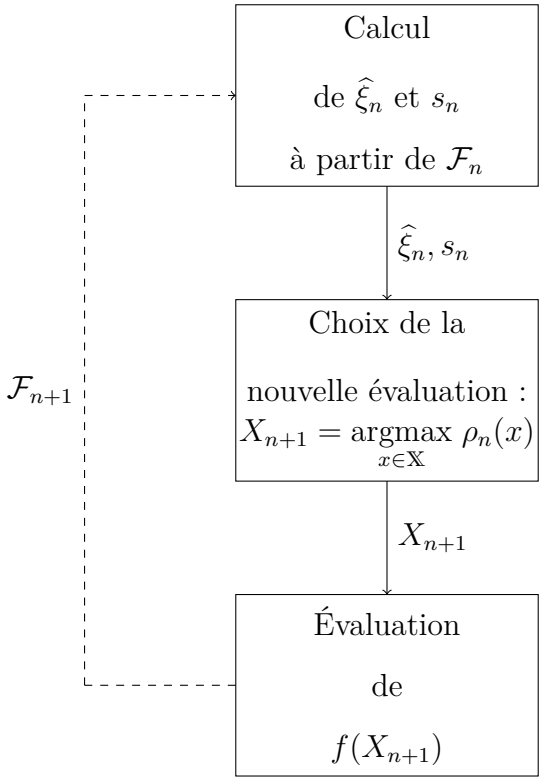


Figure 1.4 – Schéma bloc du fonctionnement d'un algorithme bayésien d'optimisation.

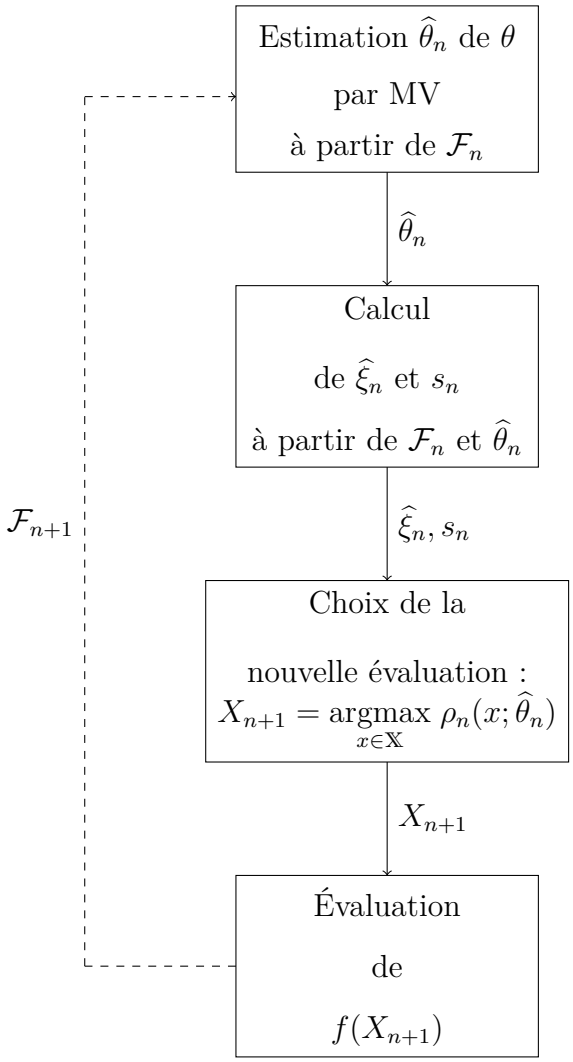
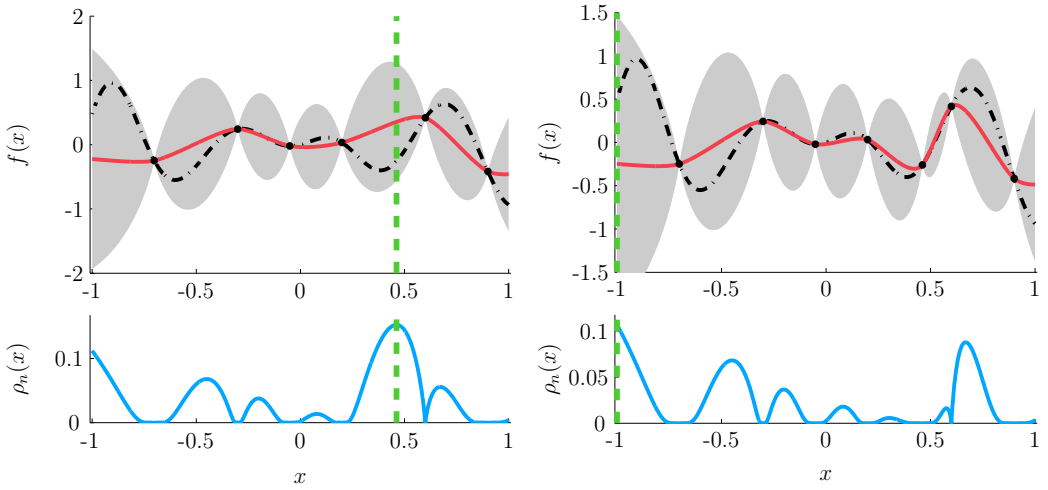
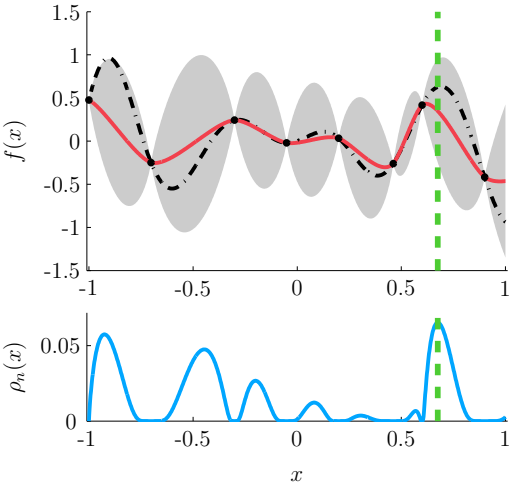


Figure 1.5 – Schéma bloc du fonctionnement de l'algorithme EGO.

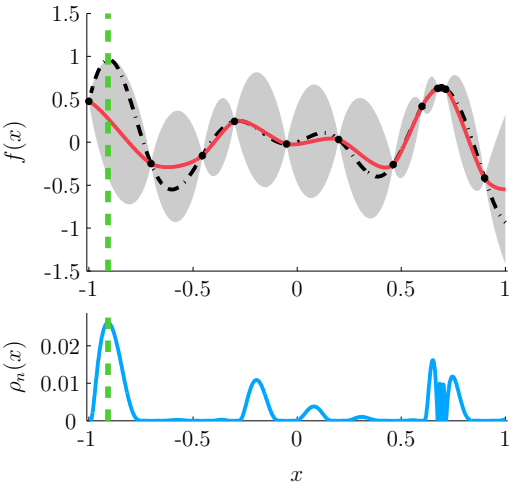


(a) Itération 1

(b) Itération 2



(c) Itération 3



(d) Itération 7

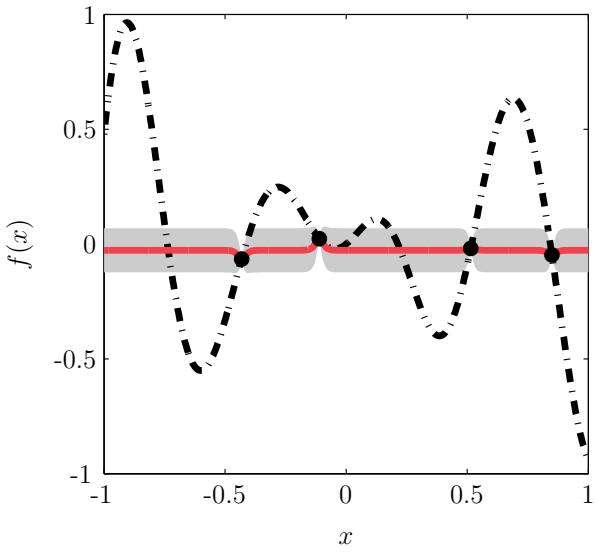
Figure 1.6 – Illustration d'une optimisation avec l'algorithme EGO, à l'itération 1. En haut : la fonction objectif (trait mixte), le prédicteur (trait plein), les intervalles de confiance à 95% contruit à partir de l'écart type (zone grise), les points d'échantillonnage (points noirs) et la position de la prochaine évaluation (ligne verticale pointillée, correspondant au maximum du critère). En bas : le critère EI. La fig. 1.6(d) nous montre que le maximum global sera finalement bien approché à la prochaine itération.

Il est néanmoins bien connu (Jones, 2001 ; Forrester et Jones, 2008) que lorsque les évaluations disponibles apportent une quantité d'information insuffisante afin d'estimer θ, la valeur de l'erreur de prédiction peut être largement sous-estimée (voir la figure 1.7). Ce genre de situations est généralement associé au terme *fonctions trompeuses*, dont l'expression anglaise équivalente est *deceptive functions*, qui décrit les fonctions dont la prédiction $\widehat{\xi}_n$, à partir des évaluations précédentes, apparaît particulièrement « plate ». En réalité, ce phénomène est susceptible de se produire pour n'importe quelle fonction, en fonction du plan d'expériences initial choisi. La conséquence, puisque le critère est calculé à partir du résultat de la prédiction, est un choix sous-optimal des futurs points d'échantillonnage. Des phénomènes d'accumulation de points au voisinage du maximum courant sont alors généralement observés (voir figure 1.8).

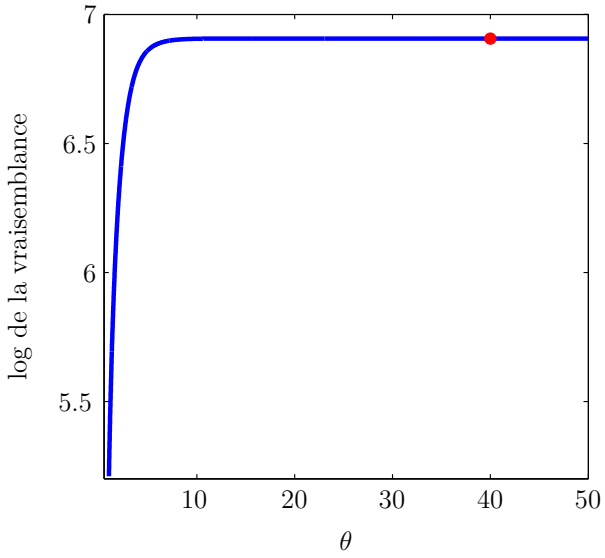
Le phénomène observé vient du fait que la valeur choisie pour θ ne rend pas suffisamment compte des variations de la fonction et que l'incertitude de l'estimateur est négligé. En particulier, lorsque peu d'évaluations de la fonction sont disponibles, l'incertitude liée à l'estimation de $\widehat{\theta}_n$ peut être grande.

1.3 Approche complètement bayésienne

Une solution pour prendre en compte l'incertitude liée aux paramètres est de considérer une approche *complètement bayésienne*, comme évoqué par Locatelli et Schoen (1995) ; Locatelli (1997) ; Williams et al. (2000) et, plus récemment, par Osborne et al. (2008, 2009) ; Osborne (2010) ; Gramacy et Polson (2011). Il faut entendre ici, par complètement bayésien, le principe de ne plus chercher à substituer une estimation de θ directement dans le critère d'échantillonnage mais d'intégrer ce critère par rapport à la loi *a posteriori* du paramètre. Que ce soit dans un contexte complètement bayésien ou éventuellement par substitution, l'introduction de la loi *a posteriori* implique le choix, par l'utilisateur, d'un *a priori*. Dans le cas général, un *a priori* permet d'affecter une pondération aux différentes valeurs possibles de θ selon la plausibilité supposée de chacune d'elles. L'avantage d'une approche complètement bayé-



(a) Fonction trompeuse



(b) Fonction de vraisemblance

Figure 1.7 – Exemple de fonction trompeuse (ligne mixte sur la première figure), et fonction de vraisemblance associée (seconde figure). Sur la première figure, les points d'évaluation (en noirs) sont choisis de sorte que les valeurs de la fonction en ces points soient proches de zéro. Après l'estimation des paramètres de covariance par maximum de vraisemblance (point rouge sur la seconde figure), la prédiction est particulièrement plate (ligne pleine) et les intervalles de confiance calculés à partir de l'écart type de l'erreur de prédiction (zones grises) sont largement sous-estimés.

sienne par rapport à une approche par substitution, est la prise en compte de la méconnaissance de la valeur de θ. L'expression du critère EI complètement bayésien, noté ρ_n, peut ainsi s'écrire

$$\rho_n(x) = \int \tilde{\rho}_n(x;\theta)\, \pi_n(\theta)\, \mathrm{d}\theta, , \tag{1.4}$$

avec $\tilde{\rho}_n(.;\theta)$ le critère EI pour ξ conditionné par θ, et π_n la densité *a posteriori* associée à celui-ci. Un algorithme EI complètement bayésien consiste donc à choisir, étant donné $\mathcal{F}_n$, un point X_{n+1} vérifiant

$$X_{n+1} \in \operatorname*{argmax}_{x\in\mathbb{X}} \rho_n(x). \tag{1.5}$$

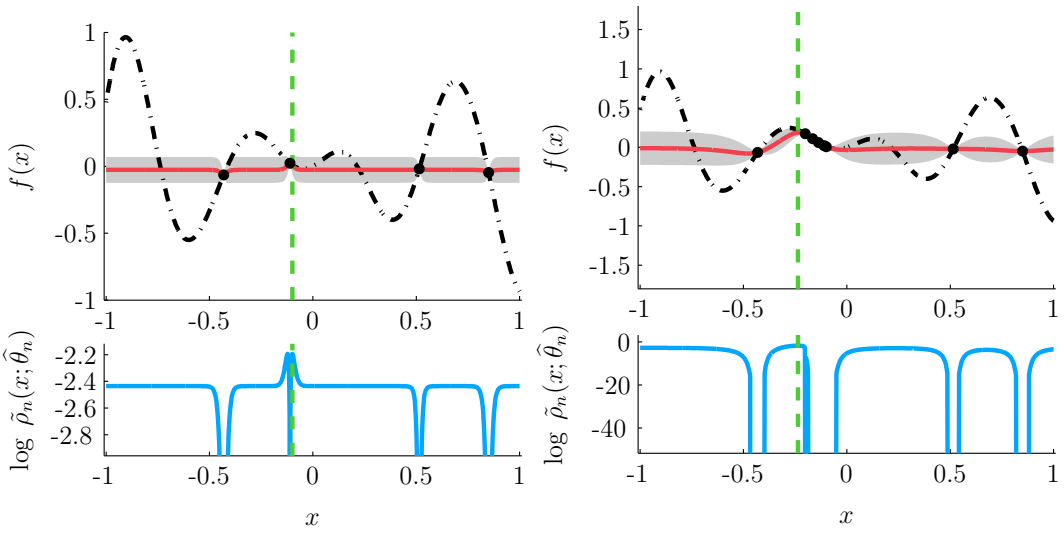
Cependant, un problème concernant la mise en œuvre se pose. Comment calculer une estimation de l'intégrale (1.4) ? De quelle façon procéder pour résoudre le problème d'optimisation (1.5) ?

L'utilisation d'une approche complètement bayésienne présente donc une difficulté supplémentaire comparativement à une approche par substitution, mais semble plus robuste, ce qui est un critère important dans le cadre de l'optimisation de fonctions coûteuses. Apporter une réponse aux deux questions de mise en œuvre formulées ci-dessus constitue la contribution principale de ce travail de thèse.

1.4 Plan du manuscrit

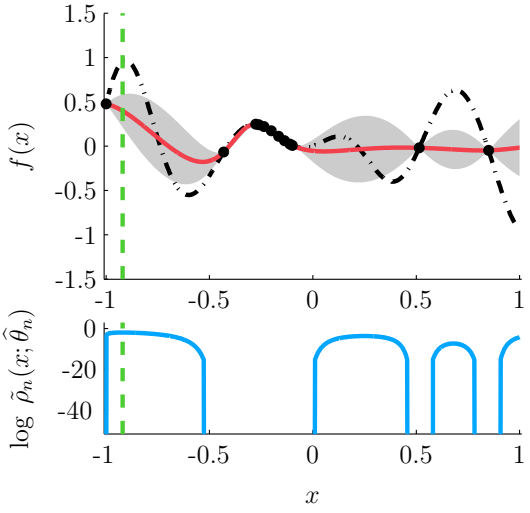
Le chapitre 2 est un chapitre introductif dressant un rapide état de l'art des critères d'échantillonnage de la littérature. Dans ce chapitre, une place importante est accordée au critère EI. En particulier, le lien y est fait entre un critère EI à horizon de plusieurs pas et la notion de programmation dynamique. Une illustration, soutenue par des résultats numériques, d'un critère EI avec un horizon de deux pas y est présentée.

Le chapitre 3 met en évidence les limites de l'approche par substitution, et la façon dont une approche complètement bayésienne permet de les dépasser. Quelques illustrations numériques mettent en évidence l'apport d'un critère EI complètement bayésien.



(a) Itération 1

(b) Itération 5



(c) Itération 9

Figure 1.8 – Algorithme EGO à l'itération 1. En haut : la fonction objectif (trait mixte), le prédicteur (trait plein), les intervalles de confiance à 95% construit à partir de l'écart type (zone grise), les points d'échantillonnage (points noirs) et la position de la prochaine évaluation (ligne verticale pointillée). En bas : le critère EI.

Le chapitre 4 présente la contribution principale de ce travail, à savoir un nouvel algorithme d'optimisation complètement bayésien. Celui-ci apporte une solution conjointe au problème relatif à l'intégrale (1.4), ainsi qu'à la maximisation du critère EI (1.5).

Des benchmarks permettant de comparer les performances de notre nouvel algorithme à des algorithmes classiques du domaine, ainsi que des applications sont présentés au chapitre 5. Les applications concernent la maximisation du rendement d'un convertisseur de puissance, ainsi qu'un problème d'identification.

1.5 Publications

Communications avec actes

C. Tugui, R. Benassi, S. Apostol et P. Benabes. *Efficient optimization methodology for CT functions based on modified Bayesian kriging approach.* ICECS 2012, Séville, Espagne, 2012

R. Benassi, J. Bect et E. Vazquez. *Optimisation bayésienne par méthodes SMC.* Journées de statistiques 2012, Bruxelles, Belgique, 2012

R. Benassi, J. Bect et E. Vazquez. *Bayesian optimization sequential Monte Carlo* Learning and Intelligent (LION 6), Paris, France, 2012

R. Benassi, J. Bect et E. Vazquez. *Robust Gaussian process-based global optimization using a fully Bayesian expected improvement criterion* Learning and Intelligent (LION 5), Rome, Italie, 2011

Communications sans actes

R. Benassi, J. Bect et E. Vazquez. *Optimisation bayésienne par méthodes SMC* GDR Mascot-num, Bruyères-le-Châtel, France, 2012

R. Benassi, J. Bect et E. Vazquez. *Optimisation de fonctions coûteuses à l'aide d'une approche bayésienne* 8ème journée Optimeo, Orsay, France, 2011

R. Benassi, J. Bect et E. Vazquez. *Correction de critère EI pour la prise en compte d'incertitude sur les paramètres d'une covariance* Atelier optimisation GDR, Mascot-num, Paris, France, 2011

R. Benassi, J. Bect et E. Vazquez. *Étude d'un nouveau critère d'optimisation bayésienne* GDR Mascot-num, Avignon, France, 2010

J. Bect, E. Vazquez, J. Villemonteix et R. Benassi. *Optimization of expensive-to-evaluate functions* Forum E3S, Supélec, 2010

Chapitre 2

Optimisation d'une fonction modélisée par un processus gaussien

2.1 Critères d'échantillonnage

Nous considérons un modèle de processus aléatoire gaussien ξ sur f, avec des fonctions moyenne et covariance connues, et nous supposons que n résultats d'évaluation ont déjà été obtenus. Nous notons $M_n = \max(\xi(x_1), \ \ldots \ , \xi(x_n))$ le maximum courant et $M = \sup_{x \in \mathbb{X}} \xi(x)$. Pour rappel, les notations $\widehat{\xi}_n$ et s_n correspondent respectivement au prédicteur par krigeage et à l'écart-type de l'erreur de prédiction de ξ à partir de l'information disponible $\mathcal{F}_n$. Les critères d'échantillonnage considérés dans ce chapitre sont calculés à partir de $\mathcal{F}_n$, résumée par la connaissance de $\widehat{\xi}_n$ et s_n. De manière générale, les notations E_n, $\mathbb{P}_n$ et H_n représentent respectivement l'espérance, la probabilité et l'entropie conditionnellement à $\mathcal{F}_n$.

2.1.1 Probabilité d'amélioration

Historiquement, le premier critère utilisé dans un cadre de l'optimisation bayésienne fut introduit par Kushner (1964), et correspond à la *probabilité*

d'amélioration. L'amélioration dont il est question est celle relative au maximum courant entre la n-ième et la n+1-ième évaluation. Le point du domaine $\mathbb{X}$ en lequel la probabilité d'améliorer est maximale, constitue ainsi la position de la nouvelle évaluation. Initialement proposé pour l'optimisation de trajectoires browniennes dans des espaces à une dimension, ce critère a ensuite été étendu à des cas plus généraux (voir, par exemple, Perttunen, 1991 ; Mockus, 1994). L'*a priori* choisi étant gaussien, la probabilité d'amélioration en un point $x \in \mathbb{X}$ s'écrit

$$\mathbb{P}_n(\xi(x) \geq M_n) = \Phi\left(\frac{\widehat{\xi}_n(x) - M_n}{s_n(x)}\right) \tag{2.1}$$

$$= \frac{1}{2}\left(1 + \operatorname{erf}\left(\frac{\widehat{\xi}_n(x) - M_n}{\sqrt{2}s_n(x)}\right)\right), \tag{2.2}$$

avec Φ la fonction de répartition de la loi normale centrée réduite et erf la fonction d'erreur de Gauss. Deux types de zones du domaine sont le plus souvent associées aux plus fortes probabilités d'amélioration, à savoir celles proches du maximum courant (aspect exploitation) et celles où la fonction est mal connue (aspect exploration). Le problème de ce critère est qu'il favorise une amélioration extrêmement faible mais probable à une amélioration qui serait bien plus importante mais légèrement moins probable. Le voisinage du maximum courant a ainsi tendance à être échantillonné de façon dense avant que l'algorithme ne reprenne un comportement exploratoire. Une variante de ce critère, décrite par Jones (2001) et conçue dans le but de résoudre ce problème, est de considérer la probabilité d'amélioration par rapport à un seuil $T = M_n + \epsilon_n$ avec $\epsilon_n \geq 0$ et tendant vers zéro lorsque n augmente. Dans cette situation, l'expression de la probabilité d'amélioration est

$$\mathbb{P}_n(\xi(x) \geq T) = \Phi\left(\frac{\widehat{\xi}_n(x) - T}{s_n(x)}\right). \tag{2.3}$$

Si ϵ_n est petit, une recherche locale aux alentours du maximum courant aura tendance à être privilégiée. Au contraire, une valeur de seuil grande sera à l'origine d'une exploration plus globale. Jones (2001) propose donc de considérer à chaque fois plusieurs valeurs de seuil T, par exemple une basse, une haute et une intermédiaire, et d'échantillonner ainsi trois nouveaux points à

chaque nouvelle itération de l'algorithme d'optimisation. Le choix est alors motivé par une perspective de recherche à la fois locale et globale. L'évaluation de la fonction en plusieurs points à la fois peut se faire par des méthodes de parallélisation des simulations informatiques. Toujours dans le même article, la valeur empirique suivante est proposée pour ϵ_n

$$\alpha_n(M_n - m_n),$$

avec m_n le minimum des résultats d'évaluation déjà disponibles et α_n un paramètre positif ou nul. Jouer sur α_n permet de régler le compromis exploitation/exploration (voir les resultats numériques de Jones (2001)). Le choix de ce paramètre n'en reste pas moins arbitraire.

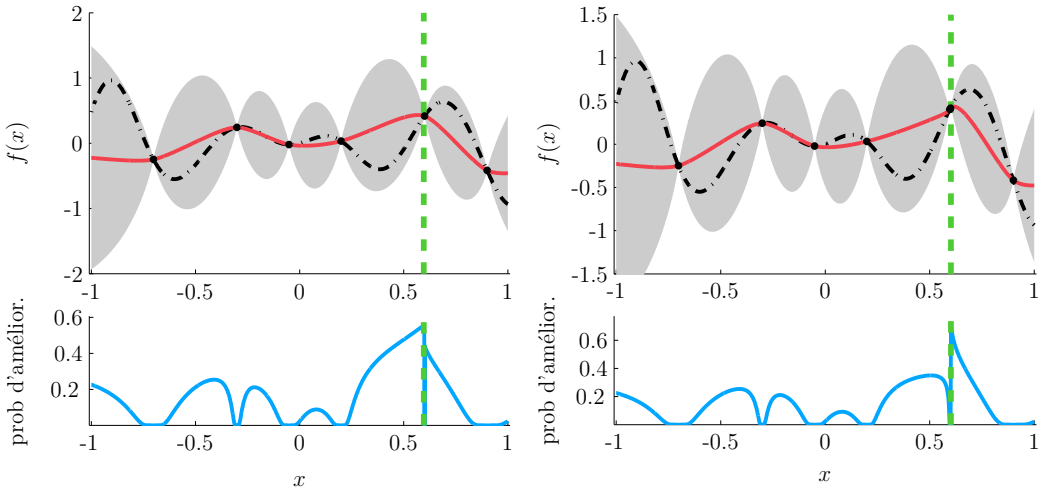
Depuis (Zilinskas, 1992), l'algorithme utilisant la probabilité d'amélioration comme critère d'échantillonnage est le plus souvent présenté sous le nom *P-algorithm*. Des résultats de convergence sont disponibles dans (Calvin et Zilinskas, 2001). Une illustration de l'algorithme, pour une valeur seuil $T = M_n$, est présentée à la figure 2.1. Celle-ci met en évidence le comportement relativement peu exploratoire de l'algorithme.

2.1.2 Espérance de l'amélioration (critère EI)

Une alternative pour contourner les limites présentées par la probabilité d'amélioration est de considérer comme critère d'échantillonnage l'espérance de l'amélioration, *Expected Improvement* en anglais, généralement noté EI. Ce critère est initialement introduit par Mockus et al. (1978), puis popularisé dans un article de Jones et al. (1998) qui l'utilise au sein de l'algorithme d'optimisation EGO. La valeur du critère, noté ρ_n, en un point x du domaine correspond à l'espérance de l'amélioration offerte par une évaluation en x (par rapport au maximum courant M_n), sachant l'information disponible $\mathcal{F}_n$,

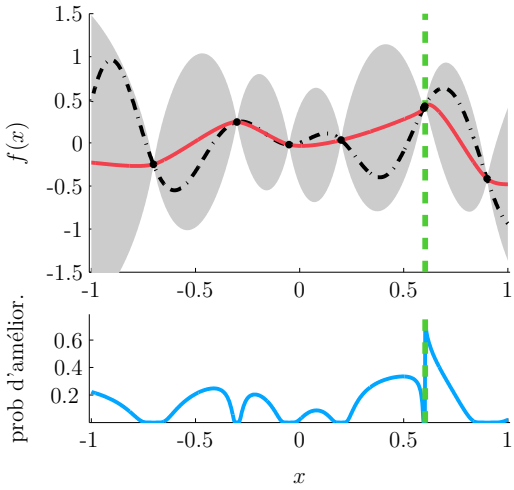
$$\rho_n(x) := \mathsf{E}_n((\xi(x) - M_n)_+). \tag{2.4}$$

L'avantage de l'EI est qu'il ne favorise pas nécéssairement une amélioration probable mais faible au détriment d'une amélioration moins probable mais

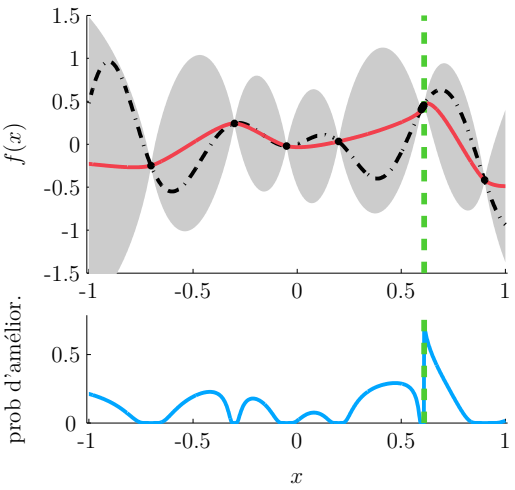


(a) Itération 1

(b) Itération 2



(c) Itération 3



(d) Itération 7. L'exploration est inexistante.

Figure 2.1 – Illustration d'une optimisation à l'aide de la probabilité d'amélioration, à partir d'un plan d'expérience initial de taille $n_0 = 6$, à l'itération 1. En haut : la fonction objectif (trait mixte), le prédicteur (trait plein), les intervalles de confiance à 95% construits à partir de l'écart type (zone grise), les points d'échantillonnage (points noirs) et la position de la prochaine évaluation (ligne verticale pointillée, correspondant au maximum du critère). En bas : le critère.

plus importante. De plus, pour un processus aléatoire ξ gaussien, il existe une expression analytique peu coûteuse à calculer de ρ_n :

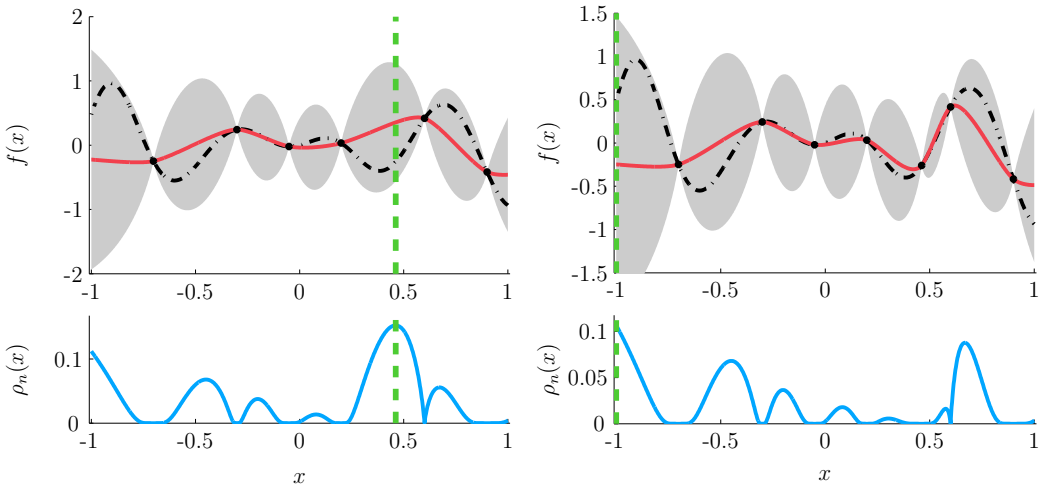
$$\rho_n(x) = \begin{cases} s_n(x)\,\Phi'\left(\frac{\widehat{\xi}_n(x)-M_n}{s_n(x)}\right) + (\widehat{\xi}_n(x) - M_n)\,\Phi\left(\frac{\widehat{\xi}_n(x)-M_n}{s_n(x)}\right) & \text{si } s_n(x) > 0, \\ \left(\widehat{\xi}_n(x) - M_n\right)_+ & \text{si } s_n(x) = 0, \end{cases} \tag{2.5}$$

avec Φ la fonction de répartition de la loi normale centrée réduite. La figure 2.2 illustre le fait que, du moins dans cette configuration spécifique, le compromis exploration/exploitation est bien plus satisfaisant qu'avec la probabilité d'amélioration. Les avantages associés à l'EI sont principalement cet équilibre entre exploration locale et globale, l'absence de paramètres à définir de façon heuristique, ainsi que l'existence d'une expression analytique. Toutes ces raisons font que ce critère est très utilisé dans le contexte de l'optimisation globale de fonctions coûteuses. Ce critère fera l'objet des chapitres suivants. Des résultats sur la convergence de l'EI sont donnés dans (Vazquez et Bect, 2010 ; Bull, 2011).

2.1.3 Entropie conditionnelle des maximiseurs globaux (critère ECM)

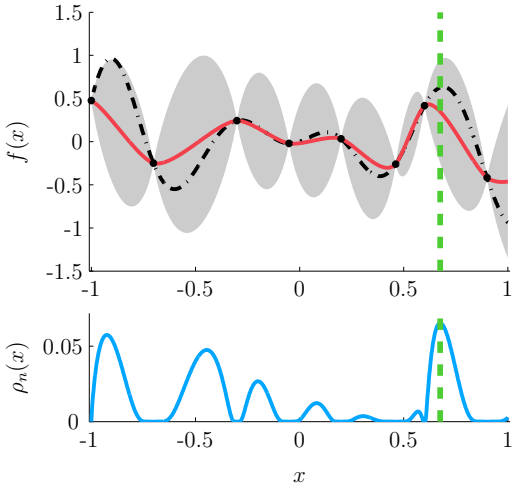
Une autre approche possible consiste à minimiser un critère d'échantillonnage caractérisant l'*entropie conditionnelle des maximiseurs globaux* (critère ECM) introduit dans (Villemonteix et al., 2009). Ce qui distingue ce critère est qu'il est orienté vers une caractérisation des maximiseurs plutôt que sur celle des maxima. Concrètement, l'entropie sert ici à quantifier l'information gagnée sur la position des maximiseurs à partir d'une nouvelle évaluation de f. En reprenant le formalisme de Villemonteix et al. (2009), nous considérons une approximation discrète $\mathbb{G}$ du domaine $\mathbb{X}$, ainsi que $\mathcal{M}_{\mathbb{G}}$ l'ensemble des maximiseurs globaux de ξ sur $\mathbb{G}$. La *distribution conditionnelle des maximiseurs globaux*

$$\mathbb{P}_n^{\tilde{X}^\star}(x) = \mathbb{P}_n(\tilde{X}_n^\star = x),\ x \in \mathbb{G}, \tag{2.6}$$

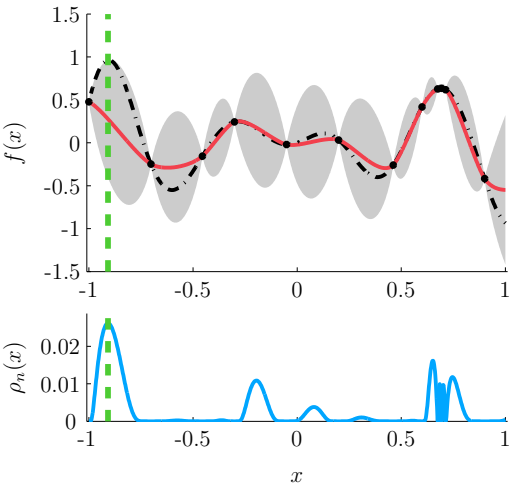


(a) Itération 1

(b) Itération 2



(c) Itération 3



(d) Itération 7. Le maximum global sera atteint à la prochaine itération.

Figure 2.2 – Illustration d'une optimisation à l'aide du critère EI, à partir d'un plan d'expérience initial de taille $n_0 = 6$.

est la loi du vecteur $\tilde{X}_n^\star$ uniformément distribué sur $\mathcal{M}_\mathbb{G}$. Cette distribution conditionnelle des maximiseurs globaux caractérise l'information obtenue sur les maximiseurs de la fonction objectif f étant donné $\mathcal{F}_n$. L'incertitude sur $\tilde{X}_n^\star$, conditionnellement à $\mathcal{F}_n$, peut être mesurée par l'entropie de Shannon

$$H_n(\tilde{X}_n^\star) = -\sum_{x\in\mathbb{G}} \mathbb{P}_n^{\tilde{X}^\star}(x) \log_2 \mathbb{P}_n^{\tilde{X}^\star}(x). \tag{2.7}$$

Cette grandeur, qui permet de caractériser l'information disponible sur les minimiseurs globaux, diminue lorsque le nombre d'évaluations augmente. L'objectif, pour augmenter ce degré de connaissance, est de minimiser cette valeur d'entropie. Étant donné $\mathcal{F}_n$, nous choisissons le point de $\mathbb{G}$ qui minimise l'incertitude attendue sur les maximiseurs globaux, autrement dit qui maximise le gain d'information attendu sur les maximiseurs :

$$X_{n+1} = \operatorname*{argmin}_{x\in\mathbb{G}} \mathsf{E}_n(H_n(\tilde{X}_{n+1}^\star \mid X_{n+1} = x)). \tag{2.8}$$

Il s'agit donc ici de mesurer l'intérêt d'effectuer la prochaine évaluation au point x, et ce, grâce au calcul de l'entropie conditionnelle moyennée sur tous les résultats d'évaluation possible de ξ en x. Le calcul de ce critère est plus complexe que celui de l'EI, et nous ne l'utiliserons pas dans le cadre de cette thèse. Pour plus de détails, se rapporter aux travaux de Villemonteix (2008).

2.1.4 Borne supérieure de confiance (UCB)

Le critère de la borne supérieure de confiance, noté UCB pour *Upper confidence bound*, dérive directement de l'algorithme d'optimisation SDO (pour *Sequential Design for Optimization*) introduit par Cox et John (1997). Il est fréquemment utilisé dans le cadre de la théorie des bandits (voir, par exemple, Auer et al., 2002 ; Garivier et Cappé, 2011), et également sous un formalisme bayésien comme pour l'algorithme Bayes-UCB (voir Kaufmann et al., 2012), ou encore le cas des « bandits gaussiens » où il est généralement appelé critère GP-UCB (voir Srinivas et al., 2010). C'est ce critère GP-UCB que nous considérons dans la suite. Son fonctionnement consiste à choisir comme nouveau point d'évaluation

$$X_{n+1} \in \operatorname*{argmax}_{x\in\mathbb{X}} \mathrm{UCB}(x) := \widehat{\xi}_n(x) + \kappa_n\, s_n(x), \tag{2.9}$$

où $\kappa_n \geq 0$ est un paramètre à définir. Une faible valeur de κ_n favorise le terme associé au prédicteur, et donc ainsi une recherche locale. Inversement, une forte valeur donne une importance au terme s_n, privilégiant ainsi les zones où f est mal connue, et donc l'exploration. Une illustration de l'algorithme SDO, lorsque κ_n est constant égal à 1.5, est donnée grâce à la figure 2.3. Dans cette configuration, et pour cette valeur de κ_n, l'optimisation est efficace. Le maximum global est bien approché dès la sixième itération. Cependant, pour κ_n constant égal à 1, l'algorithme se bloque dans un maximum local sans explorer suffisamment (figure 2.4). L'algorithme est ainsi sensible au choix de la valeur κ_n par l'utilisateur, ce qui représente une limite. Néanmoins, Srinivas et al. (2010) introduisent une approche liée à la théorie des bandits en s'intéressant à la notion de *regret cumulé*, et proposent un réglage automatique de κ_n.

2.2 Prise en compte du budget d'évaluations

2.2.1 EI et programmation dynamique

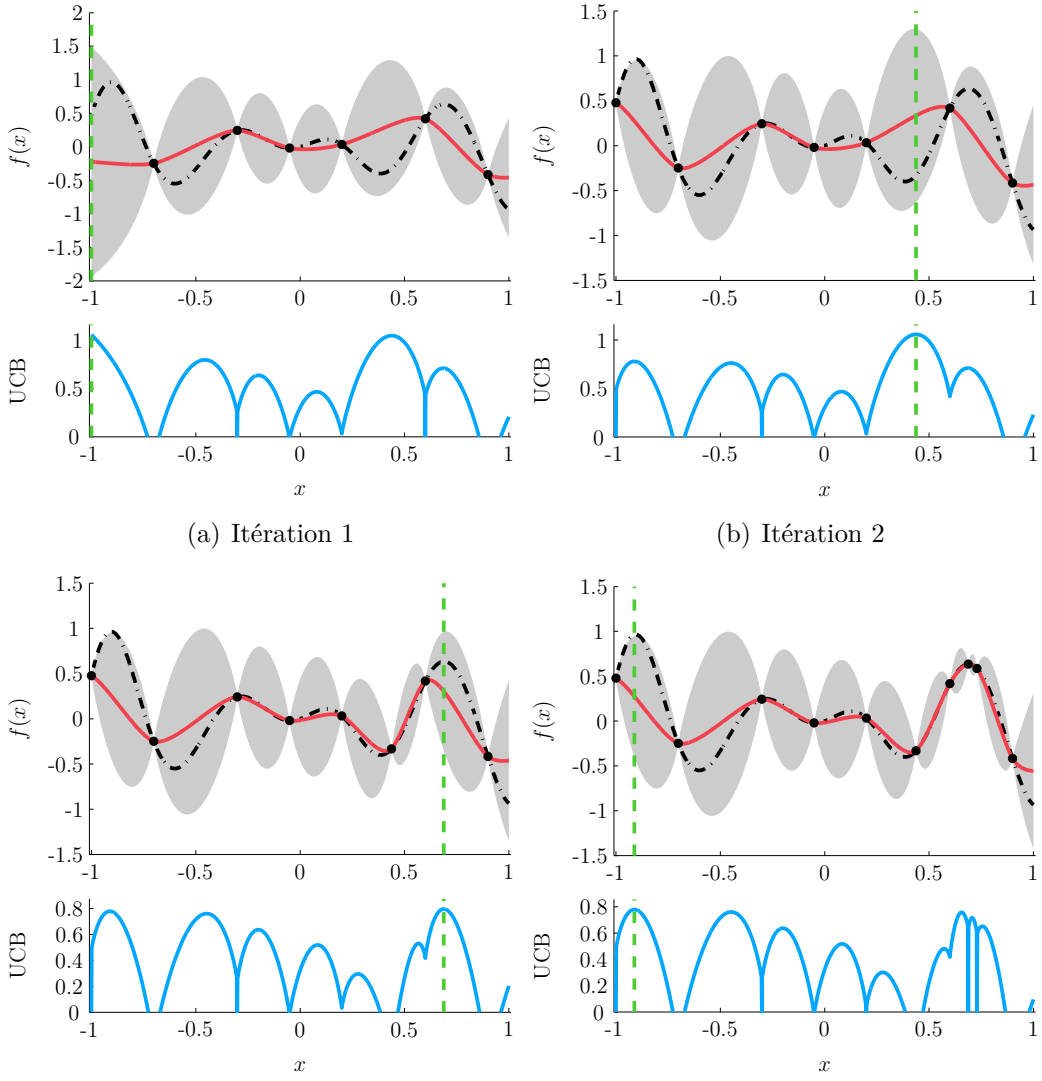
Le critère EI décrit au paragraphe 2.1.2 consiste à choisir comme nouveau point celui qui maximise l'espérance de l'amélioration à l'instant courant sans prendre en compte le fait que d'autres évaluations pourront être faites par la suite. Néanmoins, il est possible, au moins en théorie, de considérer des critères plus généraux où le budget d'évaluations est pris en compte.

Considérons l'erreur d'approximation du maximum $\varepsilon(\underline{X}_N, \xi) = M - M_N$ où $\underline{X}_N$ correspond à $(X_1, X_2, \ \ldots \ , X_N)$. Une bonne stratégie de choix de points d'évaluation est une stratégie qui atteint le risque bayésien

$$r_B = \inf_{\underline{X}_N} \mathsf{E}(\varepsilon(\underline{X}_N, \xi)),$$

où l'infimum est choisi parmi l'ensemble de toutes les stratégies séquentielles.

Il est bien connu (Mockus et al., 1978 ; Mockus, 1989 ; Betrò, 1991 ; Locatelli et Schoen, 1995 ; Auger et Teytaud, 2008 ; Ginsbourger et Le Riche, 2010 ; Grünewälder et al., 2010) qu'une stratégie d'optimisation bayésienne optimale, c'est à dire une stratégie $\underline{X}_N^\star$ telle que $\mathsf{E}(\varepsilon(\underline{X}_N^\star, \xi)) = r_B$, peut être obtenue de façon formelle par *programmation dynamique*. Notons E_n, $n \in \mathbb{N}^\star$,



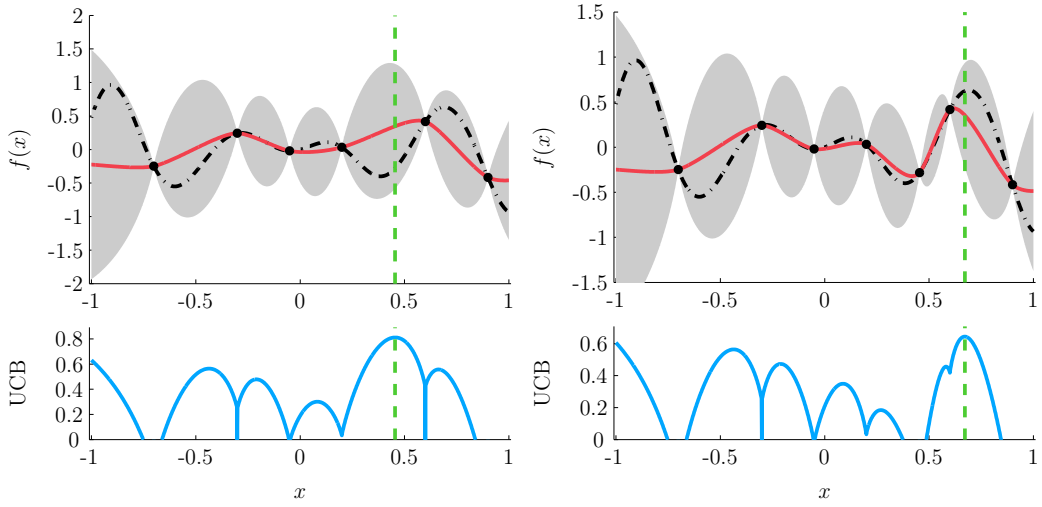
(a) Itération 1

(b) Itération 2

(c) Itération 3

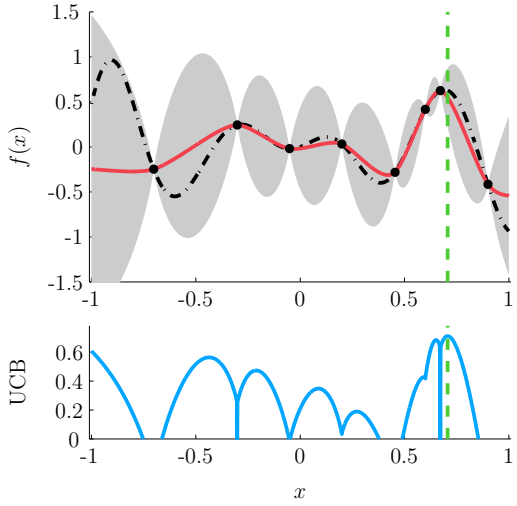
(d) Itération 8

Figure 2.3 – Illustration d'une optimisation à l'aide du critère UCB avec $\kappa_n = 1.5$ et $n_0 = 6$.

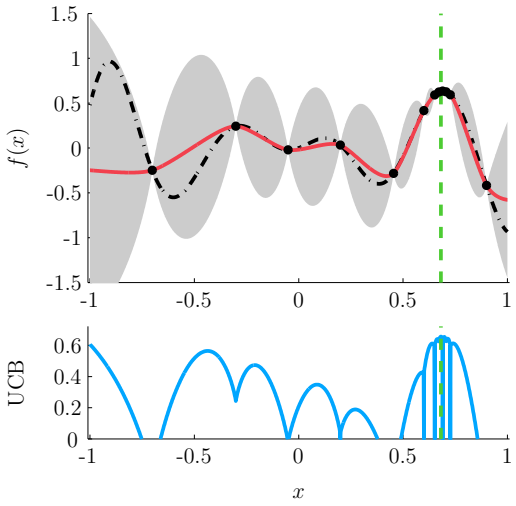


(a) Itération 1

(b) Itération 2



(c) Itération 3



(d) Itération 7

Figure 2.4 – Illustration d'une optimisation à l'aide du critère UCB avec $\kappa_n = 1$ et $n_0 = 6$.

l'espérance conditionnée par $\mathcal{F}_n$. Soit $R_N = \mathsf{E}_N(\varepsilon(\underline{X}_N, \xi))$ le risque terminal associé à la stratégie $\underline{X}_N$, et définissons de façon récursive

$$R_n = \min_{x \in \mathbb{X}} \mathsf{E}_n(R_{n+1} \mid X_{n+1} = x),\ n = N-1, \ldots, 0. \tag{2.10}$$

Nous avons $R_0 = r_B$, et la stratégie $\underline{X}_N^\star$ définie par

$$X_{n+1}^\star = \operatorname*{argmin}_{x \in \mathbb{X}} \mathsf{E}_n(R_{n+1} \mid X_{n+1} = x),\ n = 1, \ldots, N-1, \tag{2.11}$$

est optimale. Malheureusement, la résolution numérique de (2.10)-(2.11) avec un horizon de N supérieure à quelques unités n'est pas envisageable. En effet, supposons que nous discrétisions le problème en considérant un espace $\mathbb{X}$ discret de cardinal p et q valeurs possibles pour $\xi(x)$. Si, pour simplifier, nous considérons que le coût d'une résolution à un pas est toujours de c, alors pour un horizon de N la complexité serait de l'ordre de $(pq)^{N-1}c$. En effet, si déterminer la valeur de R_N a une complexité de c alors, de l'équation (2.10), nous déduisons que le calcul de R_{N-1} nécessiterait pour chacun des p points du domaine q calculs de R_N, ce qui représente une complexité de $(pq)c$. Un raisonnement par récurrence justifie alors la complexité annoncée. Les valeurs de p et q devant être particulièrement grandes pour que le problème discrétisé constitue une approximation raisonnable du problème continu, il apparaît alors naturel que cette approche ne peut être utilisée dès lors que l'horizon dépasse quelques entiers.

Une façon naturelle de résoudre ce problème est de considérer une stratégie sous-optimale à un pas, ce qui nous conduit à choisir chaque nouveau point d'évaluation selon la règle

$$\begin{aligned}
X_{n+1} &= \operatorname*{argmin}_{x \in \mathbb{X}} \mathsf{E}_n(M - M_{n+1} \mid X_{n+1} = x) \\
&= \operatorname*{argmax}_{x \in \mathbb{X}} \mathsf{E}_n(M_{n+1} \mid X_{n+1} = x) \\
&= \operatorname*{argmax}_{x \in \mathbb{X}} \mathsf{E}_n(\max(\xi(X_{n+1}), M_n) \mid X_{n+1} = x) \\
&= \operatorname*{argmax}_{x \in \mathbb{X}} \rho_n(x) := \mathsf{E}_n((\xi(X_{n+1}) - M_n)_+ \mid X_{n+1} = x).
\end{aligned} \tag{2.12}$$

Le critère d'échantillonnage obtenu n'est en fait rien d'autre que le critère EI introduit au paragraphe 2.1.2.

2.2.2 Stratégie optimale à deux pas

La stratégie à un pas, définie ci-dessus, est bien évidemment sous-optimale, mais sa complexité numérique est faible. Néanmoins, nous avons vu que la complexité liée à la stratégie optimale est rédhibitoire. Afin d'évaluer qualitativement le compromis entre performance et complexité, nous décidons de nous intéresser à une stratégie ayant un horizon à *deux* pas. Une telle stratégie consiste à considérer qu'à chaque pas le budget d'évaluations est égal à deux (voir par exemple les travaux de Ginsbourger et Le Riche (2010) ; Osborne et al. (2009)).

Ainsi nous choisissons

$$X_{n+1} = \operatorname*{argmin}_x \mathsf{E}_n(\tilde{R}_{n+1} \mid X_{n+1} = x), \tag{2.13}$$

avec

$$\begin{aligned} \tilde{R}_{n+1} &:= \min_{x\in\mathbb{X}} \mathsf{E}_{n+1}(M - M_{n+2} \mid X_{n+2} = x) \\ &= -\min_{x\in\mathbb{X}} \mathsf{E}_{n+1}(M_{n+2} - M_n \mid X_{n+2} = x) + M - M_n, \end{aligned} \tag{2.14}$$

la dernière ligne étant justifiée par le fait que M et M_n, sachant $\mathcal{F}_{n+1}$, sont connus. L'équation (2.13) peut alors se réécrire

$$X_{n+1} = \operatorname*{argmax}_x \rho_{2,n}(x),$$

avec

$$\rho_{2,n}(x) := \mathsf{E}_n\left(\min_{x'\in\mathbb{X}} \mathsf{E}_{n+1}(M_{n+2} - M_n \mid X_{n+2} = x') \mid X_{n+1} = x\right).$$

Le terme $M - M_n$ n'apparaît pas car, étant déterministe sachant $\mathcal{F}_n$, il n'a pas d'influence sur le maximiseur.

En faisant apparaître le terme M_{n+1} au sein de l'espérance, nous obtenons l'expression suivante d'un critère à deux pas

$$\begin{aligned}
\rho_{2,n}(x) &= \mathsf{E}_n\left(M_{n+1} - M_n + \min_{x'} \mathsf{E}_{n+1}(M_{n+2} - M_{n+1} \mid X_{n+2} = x') \mid X_{n+1} = x\right) \\
&= \mathsf{E}_n(M_{n+1} - M_n \mid X_{n+1} = x) \\
&\quad + \mathsf{E}_n\left(\min_{x'} \mathsf{E}_{n+1}(M_{n+2} - M_{n+1} \mid X_{n+2} = x') \mid X_{n+1} = x\right) \\
&= \rho_n(x) + \mathsf{E}_n\left(\max_{x'} \rho_{n+1}(x') \mid X_{n+1} = x\right).
\end{aligned} \tag{2.15}$$

Le premier terme n'est autre qu'un simple calcul d'EI à un pas, ce qui est facilement calculable d'après l'équation (2.5). Le second terme nécessite, d'un point de vue technique, un calcul d'EI à un pas, une détermination de maximum et le calcul d'une espérance. Une approximation du maximum peut se faire à l'aide d'une discrétisation $\mathbb{G} = \{x_1, x_2, \ldots, x_p\}$ du domaine $\mathbb{X}$, tandis que le calcul de l'espérance, pour chacun de ces points, peut se faire à partir d'une approximation d'intégrale par une somme finie (méthodes Monte Carlo par exemple) de $q \in \mathbb{N}^\star$ termes à valeurs dans $\mathbb{R}$. Ces q termes dépendent du point $x \in \mathbb{X}$ considéré et sont échantillonnées selon $\mathcal{N}\left(\widehat{\xi}_n(x),\, s_n^2(x)\right)$. En reprenant l'expression (2.15), si nous notons

$$x_i^\star = \operatorname*{argmax}_{x \in \mathbb{G}} \mathsf{E}_n\left((\xi(x) - M_{n+1})_+ \mid \xi(x_{n+1}) = Z_i\right),$$

nous pouvons ainsi obtenir pour $\rho_{2,n}(x)$ l'approximation suivante

$$\rho_n(x) + \frac{1}{q}\sum_{i=1}^{q} \mathsf{E}_n\left((\xi(x_i^\star) - M_{n+1})_+ \mid \xi(x_{n+1}) = Z_i\right), \tag{2.16}$$

où les valeurs Z_i sont échantillonnées selon $\mathcal{N}\left(\widehat{\xi}_n(x),\, s_n^2(x)\right)$.

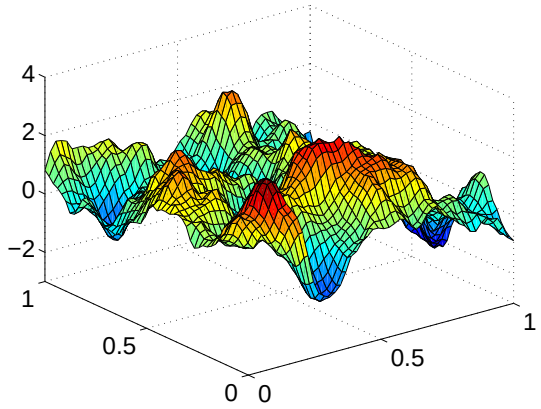
Comparaisons de plusieurs stratégies d'optimisation

Nous considérons plusieurs stratégies d'optimisation, afin d'estimer le gain que peut apporter une stratégie avec horizon à deux pas en comparaison d'une stratégie à un pas :

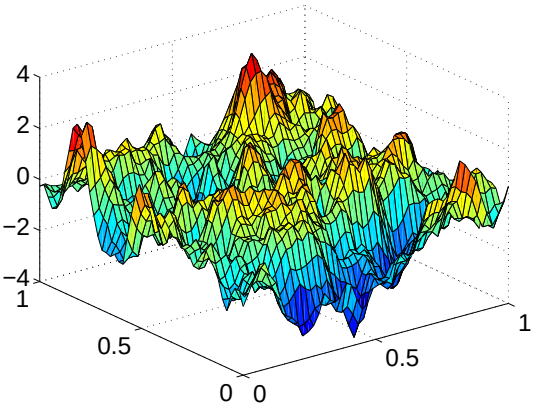
- stratégie 1111 : à chaque étape, l'horizon considéré est égal à un. Chaque nouvelle évaluation X_{n+1} est ainsi choisie par maximisation du critère ρ_n.
- stratégie 2222 : à chaque étape, l'horizon considéré est égal à deux. Chaque nouvelle évaluation X_{n+1} est ainsi choisie par maximisation du critère $\rho_{2,n}$.
- stratégie 2221 : à chaque étape, l'horizon considéré est égal à deux sauf pour la dernière où il est considéré égal à un. Autrement dit, X_N est choisi par maximisation de ρ_{N-1}, tandis que toutes les précédentes évaluations X_{n+1} sont choisies par maximisation du critère $\rho_{2,n}$. Il est à noter que cette stratégie utilise la stratégie optimale pour le choix des deux dernières évaluations.
- stratégie 2121 : alternance à chaque étape entre un horizon égal à deux et à un horizon égal à un. L'évaluation X_{n+1} est choisie par maximisation alternative de $\rho_{2,n}$ et ρ_n. Cela revient à considérer plusieurs fois de suite la statégie optimale pour un budget égal à deux.

Nous avons effectué des comparaisons entre ces différentes stratégies en les testant sur 2000 trajectoires, générées aléatoirement, de processus gaussiens définis sur le domaine $[0,1]^2$ de moyenne nulle et de covariances de Matérn (voir la figure 2.5 pour des exemples de trajectoires). Les valeurs de paramètres choisis pour cette fonction de covariance sont de 1.5 pour la régularité et des valeurs de 0.2 ou 0.1, en fonction de la série de tests considérés, pour la portée. À chaque itération, les différents critères sont maximisés sur un ensemble de 500 points candidats répartis selon un LHS (*Latin Hypercube Sampling*), pour plus de détails voir (McKay et al., 1979). Pour le calcul du critère à deux pas, nous utilisons l'approximation présentée au (2.16). Pour chaque point candidat x, nous considérons $q = 40$ points d'échantillonnage Z_i valant $\widehat{\xi}_n(x) + s_n\ \Phi^{-1}(z_i)$, où Φ^{-1} correspond à la fonction de répartition inverse de la loi $\mathcal{N}(0,\,1)$ et les z_i choisis de façon régulière dans l'intervalle $[0.01; 0.99]$. Le plan d'expérience initial est constitué d'un LHS de quatre points.

Les performances de ces différents algorithmes sont évaluées selon plusieurs critères. La figure 2.6 représente, en moyenne sur l'ensemble des trajectoires et pour les sept dernières itérations d'un budget de $N = 30$, l'erreur d'estimation



(a) Paramètre de portée égal à 0.2

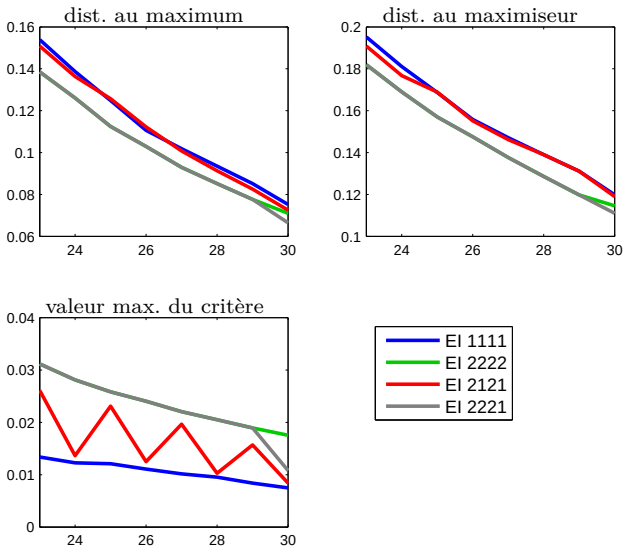


(b) Paramètre de portée égal à 0.1

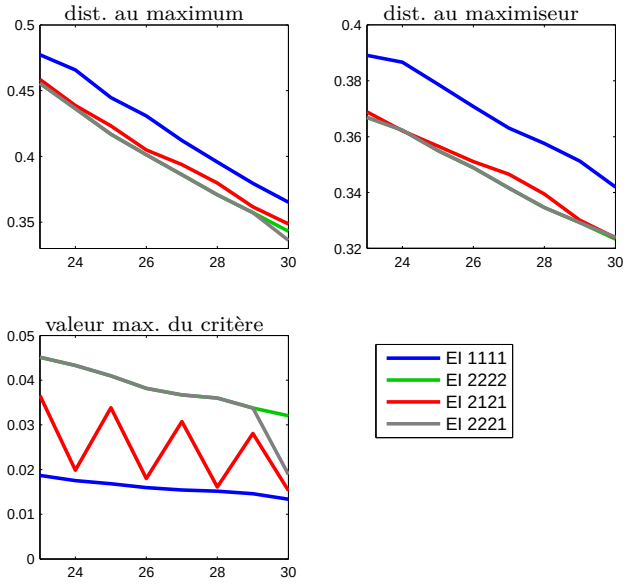
Figure 2.5 – Exemple des trajectoires gaussiennes pour deux valeurs de portées différentes (covariance de Matérn de régularité 1.5)

au maximum (à savoir $M - M_n$), l'erreur sur la position du maximiseur $x^\star$ par rapport à $x_n^\star = X_i$ où l'indice i est choisi tel que $M_n = f(X_i)$. Nous représentons également la valeur maximale du critère utilisé pour choisir la nouvelle évaluation, c'est à dire ρ_n ou $\rho_{2,n}$ selon la stratégie considérée. Une comparaison entre les figures 2.6(a) et 2.6(b) nous indique que la stratégie 2121 semble plus sûre que la stratégie 1111 lorsque les paramètres de portée ont tendance à être faibles. Autrement dit, lorsque le problème est difficile (faible valeur de portée), il semble plus robuste d'alterner des approches à deux et un pas, plutôt que de s'en remettre systématiquement à une stratégie ayant seulement un horizon d'un pas. La figure 2.7 représente quant à elle, estimée sur l'ensemble des trajectoires, la distribution de l'erreur d'estimation au maximum pour une portée de 0.1, et ce, pour différentes itérations. Une analyse de l'ensemble de ces résultats montre que toutes ces stratégies ont sensiblement les mêmes performances. Toutefois, la stratégie 2221 semble légèrement meilleure, et la stratégie classique fondée sur le critère EI à un pas (notée 1111) est généralement moins bonne. Nous pouvons remarquer l'allure « dentée » de la valeur maximale du critère pour la stratégie 2121. Elle oscille alternativement entre les valeurs associées aux stratégies 2222 (critère EI à deux pas) et 1111 (critère EI à un pas), ce qui peut s'expliquer par l'utilisation alternative du critère à deux ou un pas.

À partir des expériences effectuées, il apparaît que l'utilisation d'une stratégie fondée sur l'EI à deux pas n'est pas significativement avantageuse en moyenne sur les trajectoires d'un processus gaussien de covariance connue. Considérer, dans ce cadre, des stratégies d'optimisation avec un horizon supérieur à un pas ne nous semble donc pas particulièrement concluant mis en regard de la complexité inhérente à ce type d'algorithmes. C'est la raison pour laquelle nos travaux de thèse n'ont pas exploré plus en avant cette approche, et se concentrent essentiellement dans la suite sur l'utilisation d'un critère EI avec horizon d'un pas.



(a) Paramètre de portée égal à 0.2



(b) Paramètre de portée égale à 0.1

Figure 2.6 – Résultats des expériences numériques décrites aux pages 42 et 44 sur les sept dernières itérations pour $N = 30$ (covariance de Matérn de régularité 1.5). Les grandeurs représentées sur ces différents graphiques sont explicitées dans le texte.

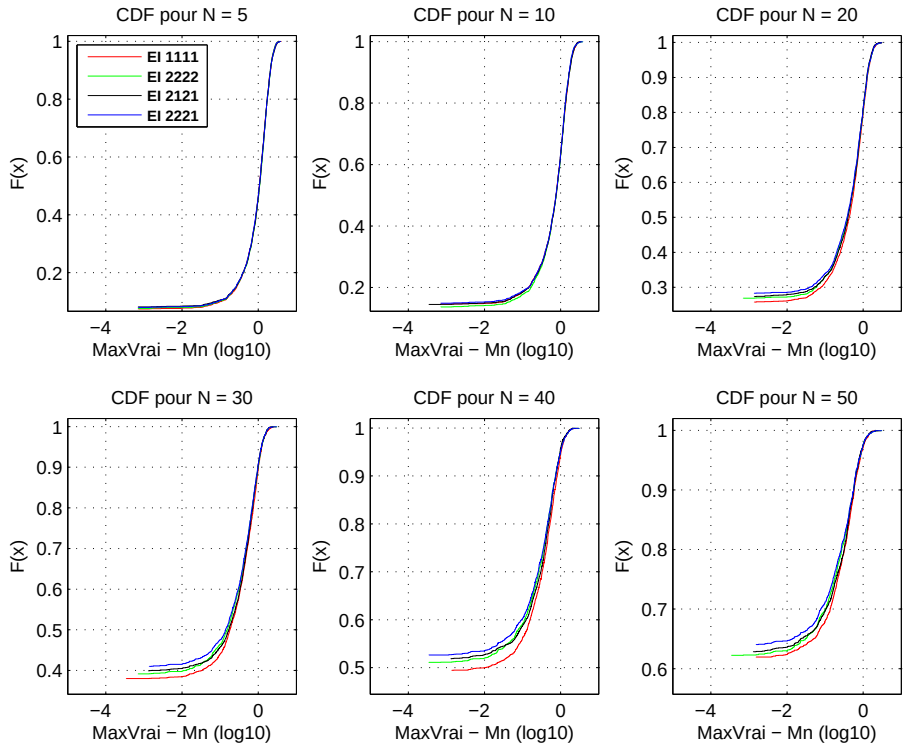


Figure 2.7 – Expériences numériques décrites aux pages 42 et 44 : fonction de répartition de l'erreur $\max f - M_N$ (covariance de Matérn de régularité 1.5 et de portée 0.1)

2.3 Résumé du chapitre

Pour choisir un critère d'échantillonnage, plusieurs considérations entrent en jeu. La première est de s'assurer que l'algorithme effectue un compromis exploration/exploitation acceptable, et ce, sans avoir nécessairement à régler de façon arbitraire des valeurs de paramètres. La seconde, est de calculer ce critère très rapidement par rapport au coût d'évaluation de la fonction objectif. Ces deux contraintes incitent à considérer le critère EI avec une attention particulière. Par la suite, lorsque le critère EI est évoqué, il s'agit du critère avec un horizon d'un pas. En effet, les résultats concernant l'extension du critère à un plus grand horizon ne nous ont pas paru très intéressants, principalement à cause de la complexité algorithmique nettement accrue pour ce type d'approche. La suite se concentre donc essentiellement sur l'approche à un pas. Implicitement, il a été supposé au cours de ce chapitre que la moyenne et la fonction de covariance du processus ξ sont connues. En pratique, une telle hypothèse n'est pas raisonnable et l'objectif du chapitre suivant est de s'en affranchir.

Chapitre 3

Approche complètement bayésienne

Au chapitre précédent, nous avons motivé notre intérêt pour le critère EI, et présenté les raisons justifiant son emploi dans l'essentiel des développements qui suivent. Ces avantages sont, en particulier, un compromis exploitation/exploration satisfaisant, et l'existence d'une formule analytique simple dans le cas où l'*a priori* ξ est gaussien. Dans la pratique il est nécessaire de choisir la covariance de ξ dans une classe paramétrée. Le problème de l'estimation des paramètres de la covariance et la prise en compte de cette estimation dans l'algorithme d'optimisation doit donc être traitée. Une réponse, déjà brievement présentée au paragraphe 1.2 dans l'introduction, est apportée par l'algorithme EGO introduit par Jones et al. (1998). À chaque nouvelle itération, l'estimation $\widehat{\theta}_n$ de θ par maximum de vraisemblance est substituée au sein de l'expression analytique de l'EI.

Une telle approche présente cependant des problèmes de robustesse dans certains cas particuliers comme celui des *fonctions trompeuses*, évoquées dans l'introduction. Ces problèmes sont illustrées à la section 3.1. Une explication de ce défaut de robustesse est la non prise en compte de l'incertitude entourant l'estimation des paramètres. Une façon de résoudre le problème consiste à affecter un *a priori* à l'ensemble des paramètres du processus ξ, approche que nous qualifions de *complètement bayésienne*, présentée aux sections 3.2 et 3.3.

Le gain apporté par une approche complètement bayésienne, sur un exemple de fonction trompeuse ainsi que sur quelques fonctions tests classiques, est discuté à partir des résultats numériques présentés à la section 3.4.

3.1 Algorithme EGO et fonctions trompeuses

3.1.1 Algorithme EGO

Dans le cadre d'une approche par substitution, une estimation de σ et θ (paramètres de la fonction de covariance paramétrée) est calculée à partir d'un certain critère à définir (maximum de vraisemblance, pénalisé ou non, maximum *a posteriori* ...).

L'algorithme EGO, particulièrement répandu dans la littérature (Schonlau, 1997 ; Schonlau et al., 1997 ; Schonlau et Welch, 1996 ; Jones et al., 1998), estime les paramètres inconnus de la covariance par la méthode du *maximum de vraisemblance*. Ces valeurs estimées sont utilisées pour calculer le critère EI à partir de la formule (2.5).

L'estimation par maximum de vraisemblance est ainsi rappelée ci-dessous. Pour ξ suivant une loi normale de moyenne m et de fonction de covariance $k_\theta(x,y) = \sigma^2 r_\theta(x,y)$, la vraisemblance des résultats d'évaluation peut s'écrire

$$\ell_n(\underline{\xi}_n; m, \sigma^2, \theta) = \frac{1}{(2\pi\sigma^2)^{n/2}|R_n(\theta)|^{1/2}}\, e^{-\frac{1}{2\sigma^2}\left(\underline{\xi}_n - m\mathbb{1}_n\right)^{\mathsf{T}} R_n(\theta)^{-1}\left(\underline{\xi}_n - m\mathbb{1}_n\right)}, \quad (3.1)$$

avec $\underline{\xi}_n = (\xi(X_1), \xi(X_2), \ldots, \xi(X_n))^T$, $\mathbb{1}_n$ la matrice $n \times 1$ dont tous les éléments sont égaux à 1, et $R_n(\theta)$ la matrice de corrélation de $\underline{\xi}_n$, paramétrée par θ. Au maximum de vraisemblance, les dérivées partielles de ℓ_n par rapport à m et σ^2 sont nulles ce qui entraîne l'expression suivante pour la moyenne et la variance estimées :

$$\widehat{m}_n(\theta) = \frac{\mathbb{1}_n^{\mathsf{T}} R_n(\theta)^{-1} \underline{\xi}_n}{\mathbb{1}_n^{\mathsf{T}} R_n(\theta)^{-1} \mathbb{1}_n}, \quad (3.2)$$

$$\widehat{\sigma}_n^2(\theta) = \frac{1}{n}\left(\underline{\xi}_n - \widehat{m}_n\mathbb{1}_n\right)^{\mathsf{T}} R_n(\theta)^{-1}\left(\underline{\xi}_n - \widehat{m}_n\mathbb{1}_n\right). \quad (3.3)$$

La démonstration est disponible en annexes à la section D.1. L'estimation

$\widehat{\theta}_n$ peut ainsi être obtenue par une maximisation du profil de vraisemblance $\tilde{\ell}_n(\underset{\sim}{\xi}_n;\theta) = \ell_n(\underset{\sim}{\xi}_n, \widehat{m}_n(\theta), \widehat{\sigma}_n^2(\theta), \theta)$ selon θ.

3.1.2 Un exemple de fonction trompeuse

Nous considérons ici le cas des fonctions trompeuses, dont une illustration est donnée par la figure 1.7. Lorsque les résultats d'évaluation disponibles n'apportent pas suffisamment d'information sur la fonction objectif f pour estimer les paramètres avec une précision acceptable, la variance de l'erreur de prédiction est le plus souvent largement sous-estimée comme montré sur la figure 3.1. Cette situation est généralement à l'origine de comportements particulièrement décevants, non seulement d'EGO, mais aussi de la plupart des algorithmes utilisant une approche par substitution, de part leur tendance à privilégier une recherche locale autour des maxima courants (exploitation), et ce dès le début de la procédure d'optimisation, au détriment d'une recherche globale (exploration). Une illustration de ce problème est donnée à la figure 3.2. Lorsque l'information disponible (ou un choix judicieux d'*a priori*, dans le cas d'une estimation par MAP) permet de calculer une bonne approximation de θ, ce problème peut être évité.

Plus généralement, l'utilisation d'un modèle stationnaire pour ξ pourrait être considérée comme un élément favorisant l'émergence de certaines fonctions trompeuses, en particulier si une zone relativement étendue du domaine est désertée à l'initialisation. Néanmoins, choisir les points d'évaluation initiaux de façon à couvrir le domaine (répartition selon, par exemple, un LHS *space filling*) permet de diminuer le risque lié à ce type particulier de fonctions trompeuses. Nous faisons donc le choix, comme l'essentiel de la littérature traitant de l'optimisation bayésienne, de ne considérer que des modèles stationnaires.

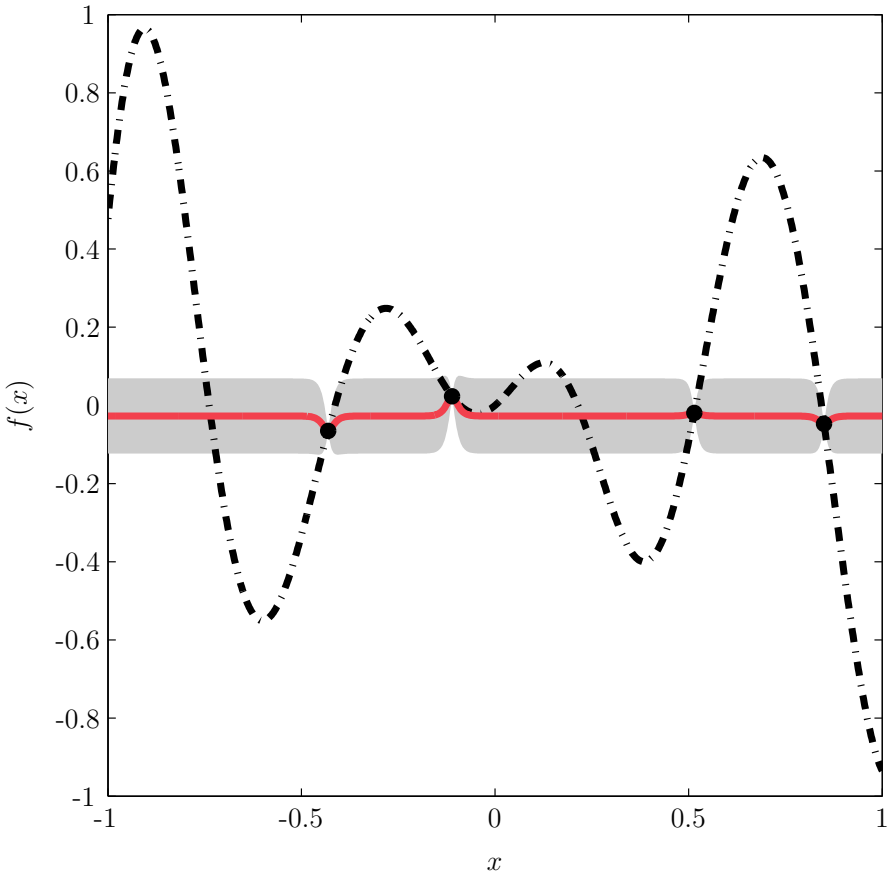


Figure 3.1 – Exemple de fonction trompeuse (ligne mixte). Les points d'évaluation (en noir) sont choisis de sorte que les valeurs de la fonction en ces points soient proches de zéro. Après l'estimation des paramètres de covariance par maximum de vraisemblance, la prédiction est particulièrement plate (ligne pleine) et les intervalles de confiance calculés à partir de l'écart type de l'erreur de prédiction (zones grises) sont largement sous-estimés.

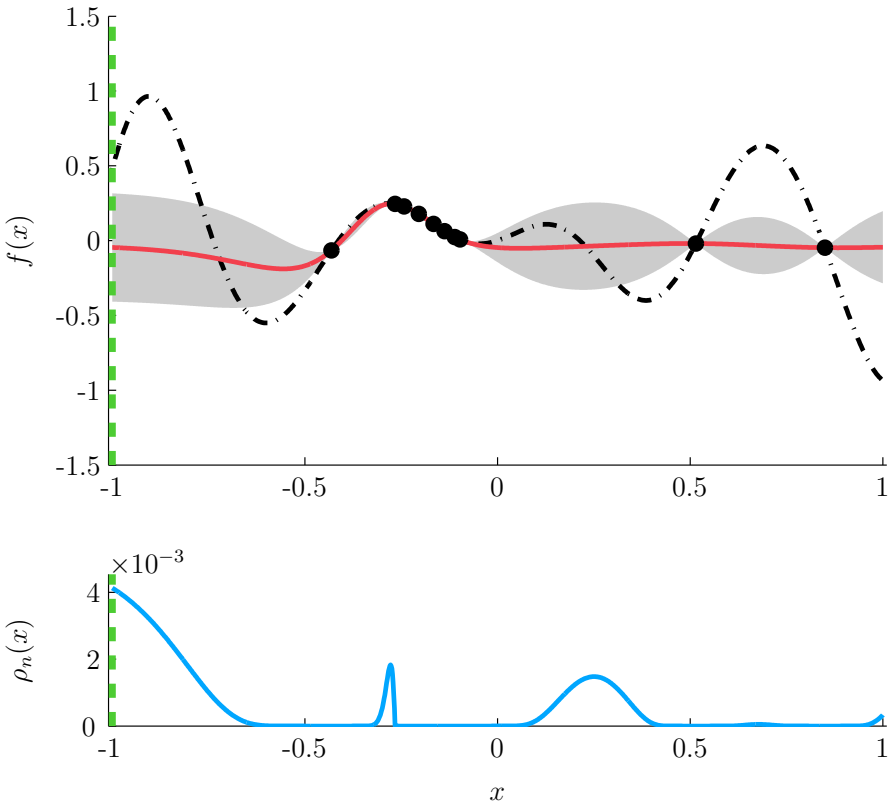


Figure 3.2 – En haut : Illustration du phénomène de « cluster » caractéristique d'une mauvaise estimation des paramètres par maximum de vraisemblance. En bas : valeur d'un critère EI « substitution »

3.2 Approche bayésienne pour l'optimisation par EI

3.2.1 État de l'art

Bien que les approches complètement bayésiennes appliquées aux processus gaussiens aient émergées dans la littérature il y a plus de vingt ans (voir O'Hagan (1978) ; Handcock et Stein (1993), ainsi que les références associées), étonnamment peu a été écrit de ce point de vue dans le contexte de l'optimisation bayésienne. Une des premières tentatives dans cette voie est décrite dans (Locatelli et Schoen, 1995 ; Locatelli, 1997), où le paramètre de variance d'un mouvement brownien est supposé distribué selon une loi inverse-gamma et ensuite intégré analytiquement. Plus récemment, l'approche complètement bayésienne a été développée dans un cadre plus général au sein des articles Williams et al. (2000) ; Osborne et al. (2008, 2009) ; Osborne (2010) ; Gramacy et Polson (2011).

Les mélanges discrets de distributions gaussiennes et le critère EI correspondant ont également été introduits par Ginsbourger et al. (2008) afin de permettre l'utilisation de plusieurs classes paramétrées de fonctions de covariance, et d'apporter ainsi un gain de robustesse en fonction du choix d'une classe en particulier. L'approche en question n'est pas bayésienne puisque les poids intervenant dans le mélange ne correspondent pas à des lois *a posteriori*.

3.2.2 Principe

Nous entendons par approche *complètement bayésienne*, une approche où *tous* les paramètres inconnus du modèle sont supposés distribués selon une densité *a priori*. Concernant la moyenne inconnue m, nous faisons le choix de la supposer constante, dans la suite du manuscrit, et distribuée selon un *a priori* uniforme (impropre) sur $\mathbb{R}$. Ce choix d'*a priori* permet d'intégrer analytiquement le critère EI par rapport à ce paramètre (voir les détails dans la section A.1). Soit π_0 la densité *a priori* du vecteur des paramètres de covariance $\theta' = (\sigma^2, \theta)$, et soient π_n la densité *a posteriori* sachant n résultats

d'évaluation. La distribution *a posteriori* de $\xi(x)$ est alors un mélange de distributions gaussiennes $\mathcal{N}\left(\widehat{\xi}_n(x;\theta'), s_n^2(x;\theta')\right)$ pondérées par $\pi_n(\theta')$. L'espérance d'amélioration pour ce modèle peut donc être écrite sous la forme

$$\begin{aligned}\rho_n(x) = \mathsf{E}_n\left((\xi(x) - M_n)_+\right) &= \mathsf{E}_n\Big(\mathsf{E}_n\left((\xi(x) - M_n)_+ \,\middle|\, \theta'\right)\Big) \\ &= \int \tilde{\rho}_n(x;\theta')\,\pi_n(\theta')\,\mathrm{d}\theta', \end{aligned} \tag{3.4}$$

avec $\tilde{\rho}_n(.;\theta')$ le critère EI conditionné par θ', autrement dit le critère EI classique pour lequel nous disposons de la formule analytique (2.5). D'après le théorème de Bayes, les densités *a posteriori* π_n s'écrivent (à une constante multiplicative près) en fonction de la vraisemblance $l_n(\xi,\theta')$ et de la densité *a priori* π_0

$$\pi_n(\theta') \propto l_n(\underline{\xi}_n;\theta')\;\pi_0(\theta'). \tag{3.5}$$

Remarquons que l'expression (2.5) de l'EI (avec paramètres fixés, donc par substitution) peut être vue comme une approximation du critère complètement bayésien (3.4)

$$\int \tilde{\rho}_n(x;\theta')\,\pi_n\left(\mathrm{d}\theta'\right) \;\approx\; \tilde{\rho}_n(x;\widehat{\theta}'_n)\,,$$

et que cette approximation est de bonne qualité lorsque la distribution *a posteriori* est concentrée autour du paramètre estimé $\widehat{\theta}'_n$, c'est-à-dire lorsque

$$\pi_n(\theta)\,\mathrm{d}\theta \approx \delta_{\widehat{\theta}'_n}(\mathrm{d}\theta).$$

3.3 Problème de l'intégration

Nous avons vu qu'une approche complètement bayésienne nécessite d'évaluer l'intégrale multidimensionnelle (3.4). Ce calcul d'intégrale constitue la principale difficulté de l'approche, c'est pourquoi nous essayons, autant que faire se peut, d'intégrer analytiquement selon les différentes composantes de θ'. Supposons que la fonction de covariance de ξ est de la forme $k(x,y) = \sigma^2 r(x,y)$ avec r connu. Williams et al. (2000) fait le choix d'un *a priori* de Jeffreys pour σ^2. Nous faisons le choix plus général[1], à partir des travaux de (Gaudard

1. Il suffit de considérer la loi $\mathrm{IG}\,(0,0)$, pour se ramener à l'*a priori* de Jeffreys.

et al., 1999 ; Pilz et Spöck, 2008) sur les *a priori* conjugués[2], de supposer que σ^2 suit une loi inverse-gamma, avec un paramètre de forme a_0 et pour paramètre d'échelle b_0, notée $\mathrm{IG}(a_0, b_0)$, et dont la densité g_{a_0,b_0} en $z > 0$ vaut

$$g_{a_0,b_0}(z) = \frac{b_0^{a_0}}{\Gamma(a_0)}(1/z)^{a_0+1}\exp(-b_0/z). \tag{3.6}$$

Proposition 1. *Si σ^2 suit une loi $\mathrm{IG}(a_0, b_0)$, alors la loi conditionnelle de σ^2 sachant $\mathcal{F}_n$ est $\mathrm{IG}(a_n, b_n)$, avec*

$$\begin{aligned} a_n &= a_0 + \frac{n-1}{2}, \\ b_n &= b_0 + \frac{1}{2}\left(\underline{\xi}_n - \widehat{m}_n \mathbb{1}_n\right)^\mathsf{T} R_n^{-1}\left(\underline{\xi}_n - \widehat{m}_n \mathbb{1}_n\right), \end{aligned}$$

avec $\widehat{m}_n = \frac{\mathbb{1}_n^\mathsf{T} R_n^{-1}\underline{\xi}_n}{\mathbb{1}_n^\mathsf{T} R_n^{-1}\mathbb{1}_n}$.

Démonstration. Voir annexes : section D.2. □

À l'aide de ce résultat, et du fait que $\xi(x) \mid \sigma^2, \mathcal{F}_n$ suit une loi normale, il apparaît que la loi prédictive de $\xi(x)$ est une loi de Student comme indiqué dans la proposition suivante.

Proposition 2. *Sous les hypothèses de cette section, et en reprenant l'ensemble des notations, soit t_η la loi de Student avec $\eta > 0$ degrés de liberté. Pour tout $x \in \mathbb{X}$,*

$$\frac{\xi(x) - \widehat{\xi}_n(x)}{\gamma_n(x)} \mid \mathcal{F}_n \sim t_{\eta_n},$$

où

$$\begin{cases} \eta_n = 2a_n, \\ \gamma_n^2(x) = b_n/a_n\, \kappa_n^2(x), \\ \kappa_n^2(x) = 1 - \underline{r}_n(x)^\mathsf{T} R_n^{-1}\underline{r}_n(x) + \frac{(1-\underline{r}_n(x)^\mathsf{T} R_n^{-1}\mathbb{1}_n)^2}{\mathbb{1}_n^\mathsf{T} R_n^{-1}\mathbb{1}_n}, \\ \underline{r}_n(x) \text{ est le vecteur de corrélation entre } \xi(x) \text{ et } \underline{\xi}_n. \end{cases}$$

2. Un *a priori* est dit conjugué s'il permet d'obtenir une loi *a posteriori* de la même forme que la sienne (en l'occurrence, ici, une loi inverse gamma).

Démonstration. Voir annexes : section D.3. □

En d'autres termes, la loi prédictive en x est une loi de Student à η_n degrés de liberté, avec un paramètre de position $\widehat{\xi}_n(x)$ et un paramètre d'échelle $\gamma_n(x)$. L'utilisation conjointe de la proposition 2 et du lemme suivant mène à une expression explicite du critère EI.

Lemme 1. *Soit $T \sim t_\eta$ avec $\eta > 0$:*

$$\mathsf{E}\Big((T+u)_+\Big) \;=\; \begin{cases} +\infty & \text{si } \eta \leq 1, \\ \frac{\eta+u^2}{\eta-1}\, F'_\eta(u) \;+\; u\, F_\eta(u) & \text{sinon,} \end{cases}$$

où F_η est la fonction de répartition de t_η.

Démonstration. Voir annexes : section D.4. □

Théorème 1. *Sous les hypothèses de cette section, pour tout $x \in \mathbb{X}$,*

$$\mathsf{E}_n\Big((\xi(x) - M_n)_+\Big) \;=\; \gamma_n(x) \left(\frac{\eta_n + u^2}{\eta_n - 1}\, F'_{\eta_n}(u) \;+\; u\, F_{\eta_n}(u)\right), \tag{3.7}$$

avec $u = \Big(\widehat{\xi}_n(x) - M_n\Big)/\gamma_n(x)$ et F_η est la fonction de répartition de t_η.

Démonstration. Il suffit de remarquer que

$$\xi(x) - M_n = \gamma_n(x) \left(\frac{\xi(x) - \widehat{\xi}_n(x)}{\gamma_n(x)} + \frac{\widehat{\xi}_n(x) - M_n}{\gamma_n(x)}\right),$$

et d'utiliser le lemme 1 avec

$$T = \frac{\xi(x) - \widehat{\xi}_n(x)}{\gamma_n(x)},$$

ce qui est valable d'après la proposition 2, et

$$u = \frac{\widehat{\xi}_n(x) - M_n}{\gamma_n(x)}.$$

□

Ce théorème généralise un résultat de Williams et al. (2000) qui concernait le cas d'un *a priori* de Jeffreys sur σ^2 (autrement dit, une loi $\mathrm{IG}(0,0)$). L'expression du critère EI donnée par le théorème 1 fait intervenir une loi de Student, et nous l'appellons donc critère *Student-EI*.

Il a été supposé, dans ce qui précède, que le seul paramètre inconnu intervenant dans la fonction de covariance est la variance σ^2. Plus généralement, supposons $k(x,y) = \sigma^2 r(x,y;\theta)$ où θ est indépendant de (m,σ^2) et suit une loi *a priori* π_0. Le théorème 1 nous donne en fait la valeur du critère EI $\tilde{\rho}_n(x;\theta) = \mathsf{E}_n\left((\xi(x) - M_n)_+ \mid \theta\right)$, lorsque le paramètre θ est connu. Le critère complètement bayésien peut alors s'écrire

$$\mathsf{E}_n\left((\xi(x) - M_n)_+\right) \;=\; \mathsf{E}_n\Big(\tilde{\rho}_n\left(x;\theta\right)\Big) \;=\; \int \tilde{\rho}_n(x;\theta)\,\pi_n(\mathrm{d}\theta), \tag{3.8}$$

avec π_n la densité *a posteriori* de θ après n évaluations. L'évaluation numérique du critère (3.8) nécessite le calcul d'une intégrale multidimensionnelle et les techniques pouvant être utilisée pour effectuer ce calcul seront exposées au chapitre 4. Une technique élémentaire, mais peu satisfaisante en pratique, consiste à supposer que la mesure de référence de π_0 est discrète (support discret).

3.4 Comparaison approche par substitution, approche complètement bayésienne

Prendre en compte l'incertitude associée à θ permet en principe d'éviter le problème des fonctions trompeuses. L'objectif de cette section est de montrer le gain de robustesse apporté.

3.4.1 Optimisation d'une fonction trompeuse

Expériences

Soit la fonction objectif $f : \mathbb{X} = [-1,1] \to \mathbb{R}$ définie par

$$f(x) \;=\; x\left(\sin(10x+1) + 0.1\sin(15x)\right), \quad \forall x \in \mathbb{X}\,.$$

À l'initialisation, nous choisissons un plan d'experiences de quatre points d'abscisses -0.43, -0.11, 0.515 et 0.85, comme illustré dans la figure 3.1. L'objectif est de comparer les points d'évaluation choisis à l'aide d'une approche par substitution (EGO, ici, en l'occurrence) à ceux choisis à l'aide d'une approche complètement bayésienne telle qu'introduite dans la section 3.3.

Pour les deux approches, nous faisons le choix d'une covariance de Matérn (voir la section A.2.2) avec une régularité connue $\nu = 2$. Pour le critère complètement bayésien, nous choisissons une loi inverse gamma $IG(0.2, 12)$ pour σ^2, ce qui nous permet d'intégrer analytiquement σ^2. Le domaine $\mathbb{X}$ étant de dimension un, il n'y a qu'un seul paramètre de portée β qui intervient. Pour simplifier la mise en œuvre de l'approche proposée, nous supposons que la loi de β est uniforme sur un support fini. Plus précisémment, nous définissons une valeur $\beta_{\min}$ et une valeur $\beta_{\max}$, telles que $\beta_{\min} < \beta_{\max}$, ainsi que, pour tous $i = 0, \ldots, I-1$, des valeurs $\beta_i = \beta_{\min} \left(\frac{\beta_{\max}}{\beta_{\min}}\right)^{i/(I-1)}$. Nous prenons $\beta_{\min} = 2 \times 10^{-3}$, $\beta_{\max} = 2$, et $I = 100$. Nous associons aux β_is des probabilités *a priori* égales à $1/I$.

L'optimisation du critère d'échantillonnage sur le domaine $\mathbb{X}$ est réalisée de façon identique quel que soit le critère utilisé (par substitution ou complètement bayésien). Concrètement, un ensemble de $q = 600$ points candidats est généré uniformément sur $\mathbb{X}$, et n'est plus modifié par la suite. Les maximisations successives du critère considéré sont effectuées en l'évaluant en chaque point de cet ensemble fini (les mêmes points étant utilisés pour les deux critères).

Résultats

Les figures 3.3 à 3.5 nous montrent que l'optimisation n'est pas menée de façon efficace lorsque l'écart-type de l'erreur de prédiction est largement sous estimée, ce qui est le cas pour une utilisation de l'algorithme EGO. Si l'incertitude sur les paramètres est prise en compte, l'écart-type de l'erreur devient plus satisfaisant, comme expliqué plus haut. Les figures 3.4(b) et 3.5(a) nous montrent que le maximum est approximativement atteint après seulement quatre itérations lorsque l'approche est complètement bayésienne, tandis que

l'algorithme EGO requiert neuf itérations avant d'évaluer un point proche du maximiseur. En effet, EGO reste « bloqué » au voisinage de l'optimum local au cours de nombreuses itérations, tandis que le domaine $\mathbb{X}$ reste largement non exploré. Ce comportement est particulièrement problématique dans un contexte d'optimisation de fonctions coûteuses.

3.4.2 Résultats sur des fonction tests

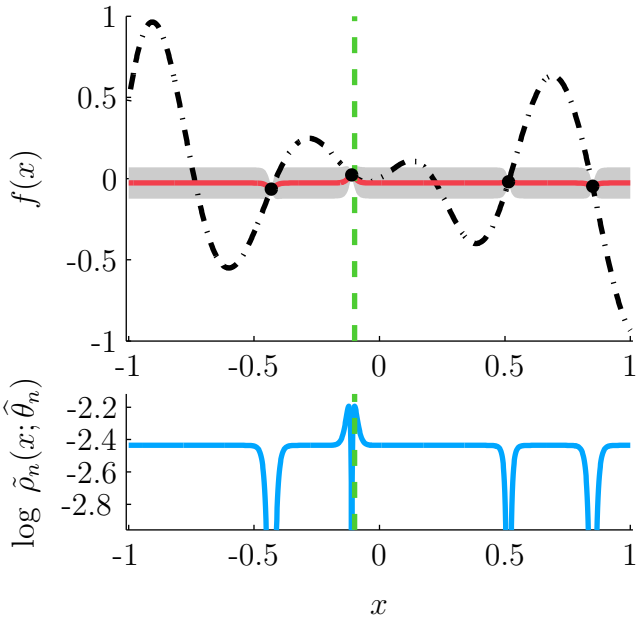
Afin de tester l'approche complètement bayésienne, que nous notons désormais FB-EI (pour *fully bayesian* EI), dans un cadre plus général que l'exemple de fonction trompeuse du 3.4.1, nous considérons des tests effectués à partir de fonctions régulièrement utilisées dans des contextes d'optimisation. En plus de FB-EI, nous considérons à nouveau un algorithme EGO, algorithme bayésien également. Nous nous intéressons principalement à la moyenne ainsi qu'à la médiane de l'erreur d'estimation du maximum. Deux autres stratégies sont utilisées pour comparaison, à savoir l'algorithme déterministe DIRECT (dont une mise en œuvre est donnée par Finkel, 2003), et une stratégie de référence consistant à choisir les points d'évaluation de façon aléatoire uniforme sur le domaine.

Ces tests permettent d'obtenir plusieurs informations :

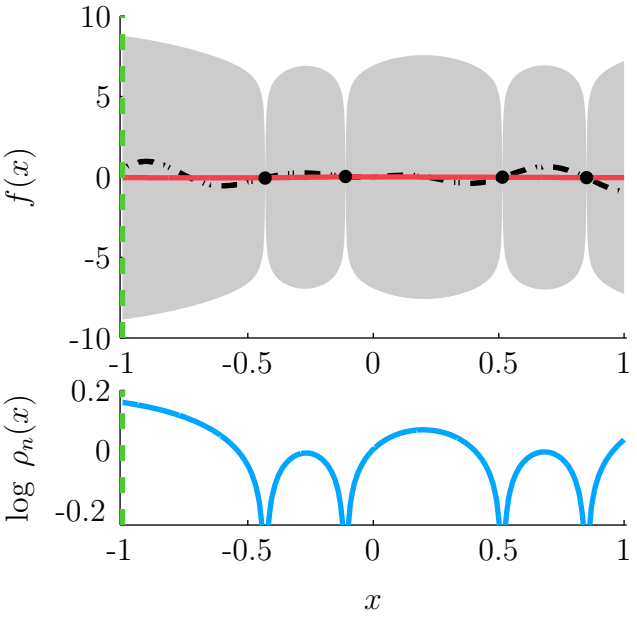
- Comparaison des stratégies bayésienne avec un algorithme non bayésien de référence (DIRECT).
- Comparaison de toutes ces stratégies d'optimisation par rapport à un échantillonnage uniforme.
- Comparaison d'une approche complètement bayésienne avec une approche par substitution (EGO).
- Influence, pour les algorithmes bayésien, de la taille du plan d'expérience initial.

3.4.3 Paramètres de simulation

L'analyse statistique des résultats est faite à partir de 200 simulations répétées pour les deux algorithmes bayésiens (FB-EI et EGO), et de 1000 pour la

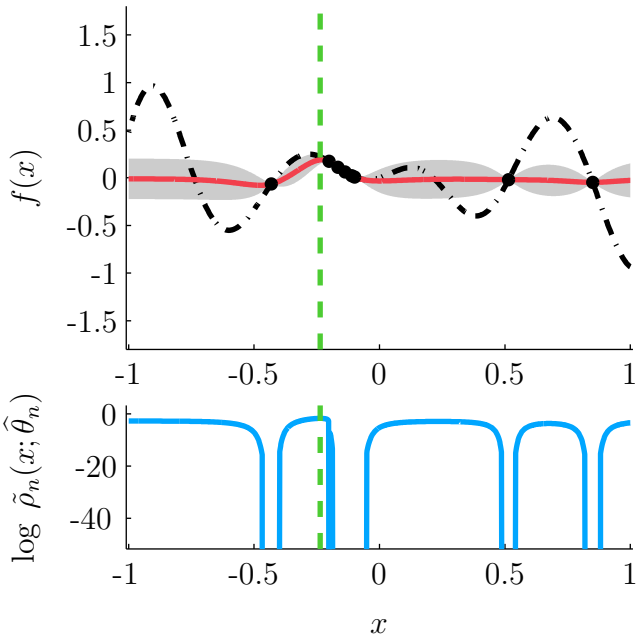


(a) Paramètres estimés par MV

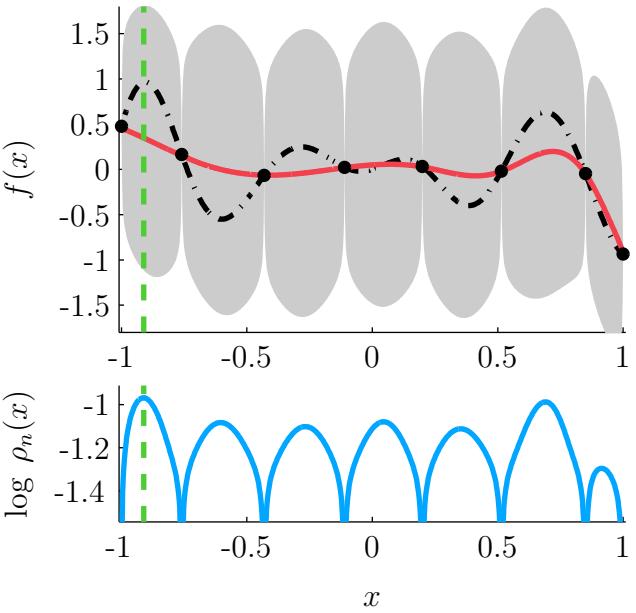


(b) Approche bayésienne pour les paramètres

Figure 3.3 – Une comparaison entre a) EGO et b) l'approche complètement bayésienne à l'itération 1. En haut : la fonction objectif (traits mixtes), le prédicteur (trait plein), les intervalles de confiance à 95% (zones grises), les points d'échantillonnage (en noir) et la position de la nouvelle itération (ligne verticale avec traits pointillés). En bas : critère EI.

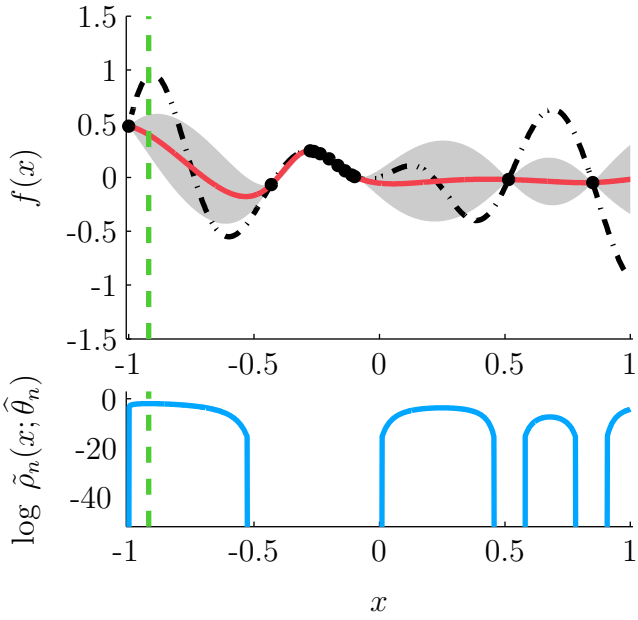


(a) Paramètres estimés par MV

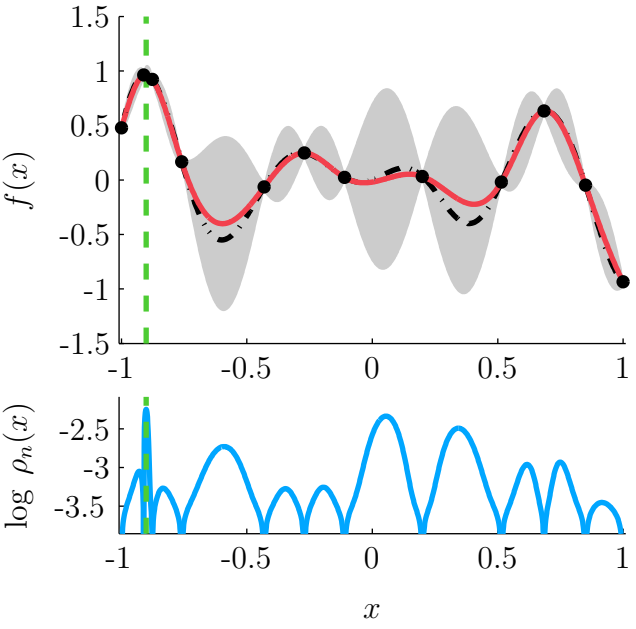


(b) Approche bayésienne pour les paramètres

Figure 3.4 – Itération 5 (voir Figure 3.3 pour plus de précisions)



(a) Paramètres estimés par MV



(b) Approche bayésienne pour les paramètres

Figure 3.5 – Itération 9 (voir Figure 3.3 pour plus de précisions)

référence par échantillonnage uniforme. Le caractère déterministe de DIRECT ne nécessite pas de simulations répétées. Pour chacune des stratégies bayésiennes, deux tailles n_0 de plan d'expériences sont considérées, ce qui permet de définir quatre cas. Les deux tailles n_0 considérées sont prises égales à deux fois, et dix fois la dimension d de $\mathbb{X}$ (variable selon les fonctions considérées). Ces points d'initialisation sont répartis selon un LHS choisi de façon à recouvrir le domaine (LHS *space-filling*, construit en s'efforçant de maximiser la distance minimal entre deux points du plan d'expérience), généré aléatoirement à chacune des 200 simulations. L'optimisation du critère EI (complètement bayésien ou non) se fait sur un ensemble de 10^4 points candidats générés selon une loi uniforme sur le domaine $\mathbb{X} = \prod_{i=1}^{d}[u_1^i; u_2^i]$. La régularité ν est choisie constante égale à $5/2$.

Pour l'algorithme complètement bayésien, nous choisissons sont un *a priori* de Jeffreys pour la variance σ^2, et un *a priori* log-normal sur l'inverse des portées de la forme $\ln \mathcal{N}(\mu_0^i, 0.5^2)$, pour la i-ième composante du vecteur paramètre θ, où $\mu_0^i = -\ln(\sqrt{d}(u_2^i - u_1^i)/3)$. Cet *a priori*, choisi en fonction de la dimension et de la taille du domaine, est à notre sens raisonnable dans la mesure où il favorise des valeurs de portées correspondant à une situation où la stratégie d'optimisation est adaptée. En effet, un *a priori* favorisant des portées trop petites, favoriserait ainsi des situations où l'optimisation serait peu efficace car trop difficile. Comme vu précédemment, cette approche complètement bayésienne nécessite de calculer une estimation de l'intégrale dans l'expression (3.8), ce qui est fait à l'aide d'une approche par méthodes SMC, détaillée au chapitre suivant à la section 4.1.

Concernant EGO, l'estimation des paramètres par maximum de vraisemblance est effectuée, à chaque nouvelle itération, grâce à la fonction matlab *fmincon*, méthode d'optimisation locale à l'aide du gradient, avec comme point de départ la valeur de paramètres estimée utilisée lors de l'itération précédente. À l'initialisation, le point de départ utilisé pour (σ^2, θ) vaut $(1, \mu_0)$, avec $\mu_0 = [\mu_0^1, \mu_0^2, \ \ldots \ , \mu_0^d]$.

3.4.4 Résultats

Les fonctions tests considérées sont la fonction associée à l'hyper-sphère (centrée sur zéro) en dimension 4 (figure 3.6), la fonction de Branin (figure 3.7), et le logarithme de la fonction Hartman 3 (figure 3.8). Les expressions de ces fonctions tests sont explicitées en annexes : section C. Pour l'hyper-sphère, le maximiseur se situe au milieu du domaine de définition, qui est également toujours le premier point d'évaluation de l'algorithme DIRECT, ce qui implique de trouver le maximum global dès la première évaluation. C'est la raison pour laquelle la courbe associée à cet algorithme n'apparaît pas sur la figure 3.6.

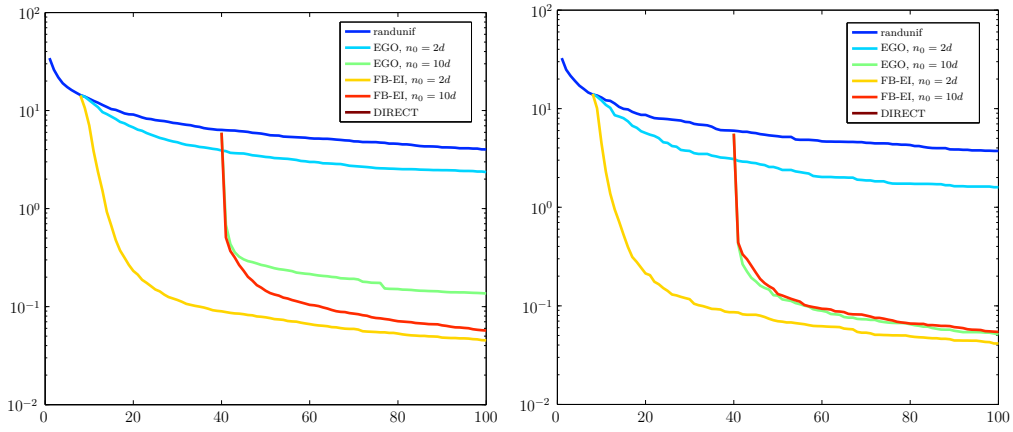
Intéressons nous aux algorithmes non bayésiens. Comme attendu, la stratégie de référence consistant à échantillonner uniformément est, dans tous les cas testés, la moins efficace. L'algorithme DIRECT, mis à part le cas particulier de l'hyper-sphère, est le meilleur sur les première itérations, mais s'avère moins bon que FB-EI et EGO pour la médiane. Les résultats en moyenne sont également moins bons que pour les algorithmes bayésiens, mis à part le cas d'EGO avec $n_0 = 2d$.

Les algorithmes bayésiens, pour la médiane, donnent des résultats similaires à l'approche de la centaine d'évaluations. Néanmoins, l'étude des résultats en moyenne permet d'affiner l'analyse. En effet, lorsque $n_0 = 2d$, les résultats sont favorables en moyenne à l'approche FB-EI au détriment d'EGO. Pour une telle taille de plan d'expérience initial, les performances d'EGO en moyenne sont mauvaises en comparaison des autres stratégies bayésiennes, ce qui n'apparaît pas lors de l'étude des médianes. Pour $n_0 = 10d$, le même phénomène s'observe sur les figures 3.6 et 3.8 bien qu'il soit moins marqué, les performances étant même légèrement à l'avantage d'EGO sur la figure 3.7. Ces résultats, en particulier pour une petite valeur de n_0, peuvent s'expliquer par un défaut de robustesse inhérent à l'algorithme EGO. En effet, sur la plupart des situations, l'optimisation se fait aussi bien avec EGO qu'avec une approche complètement bayésienne, tel que montré par les médianes. Cependant, les pires situations sont très défavorables à EGO et impactent sur les moyennes de façon significative. Commencer avec un n_0 petit augmente les risques de se trouver dans une situation où l'optimisation est difficiles pour EGO, ce qui fait echo aux

considérations relatives aux fonctions trompeuses de la section 3.4.1.

Afin de mener plus loin l'analyse, nous nous intéressons, pour l'exemple de la fonction de Branin, à l'ensemble des 200 simulations effectuées. Pour $n_0 = 2d$, la figure 3.9(a) nous permet de comparer l'erreur au maximum obtenue avec EGO et celle obtenu avec FB-EI, pour chaque simulation. Le défaut de robustesse d'EGO, pour une telle valeur de n_0, est alors particulièrement mis en évidence puisque les courbes d'erreurs les plus défavorables (les plus hautes sur la figure) lui sont toutes associées. Ceci offre un éclairage pour comprendre la différence observée pour cet algorithme entre les résultats en moyenne et la médiane. Les figures 3.9(b) et 3.9(c) nous montrent les différentes estimations des deux paramètres de portée, où plutôt de leur inverse. Pour EGO, la valeur estimée correspond au maximum de vraisemblance, tandis que pour FB-EI il s'agit de la moyenne selon la loi *a posteriori* (plus de détails sur le fonctionnement de l'algorithme au chapitre suivant). À nouveau, si une valeur moyenne sur l'ensemble des trajectoires semblent émerger, il apparaît lors de certaines simulations que les valeurs estimées par EGO sont particulièrement heurtées et éloignées de la valeur moyenne. Au contraire, les valeurs estimées par FB-EI constituent un faisceau compact, sans qu'aucune trajectoire ne présente une dérive particulière. La figure 3.10 représente exactement les mêmes grandeurs mais dans le cas où n_0 est égal à $10d$. Concernant les trajectoires, EGO ne semble pas moins robuste cette fois-ci. Néanmoins, pour l'estimation des paramètres, il apparaît sur certaines simulations que la valeur de l'estimation par maximum de vraisemblance a des variations heurtées.

Le bilan de ces résultats est qu'une stratégie complètement bayésienne semble plus robuste qu'une stratégie par substitution, en particulier en ce qui concerne la taille du plan d'expérience initial n_0. Lorsque cette taille est de dix fois la dimension, les deux stratégies ont des performances proches. Néanmoins, choisir un n_0 petit, tel que deux fois la dimension, représente l'avantage de commencer une optimisation effective plus rapidement, mais ne peut être considéré raisonnable que pour une approche complètement bayésienne telle que FB-EI. De manière plus générale, à l'exception des résultats en moyenne pour un EGO avec $n_0 = 2d$, les stratégies bayésiennes s'avèrent plus perfor-



(a) Moyenne (b) Médiane

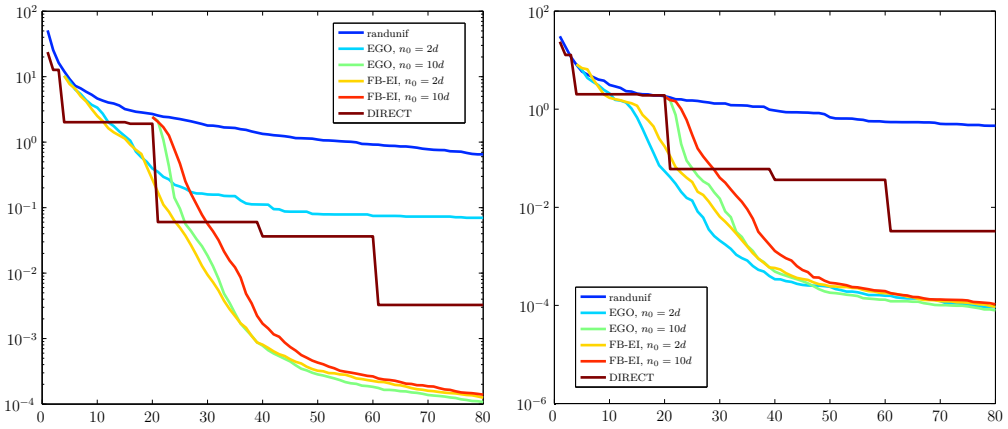
Figure 3.6 – Comparaison de l'erreur d'estimation du maximum (moyenne et médiane à partir de 200 simulations) en fonction du nombre d'évaluations pour différents algorithmes. La fonction considérée est celle associée à l'hyper-sphère en dimension 4.

mantes que l'algorithme DIRECT.

3.5 Résumé du chapitre

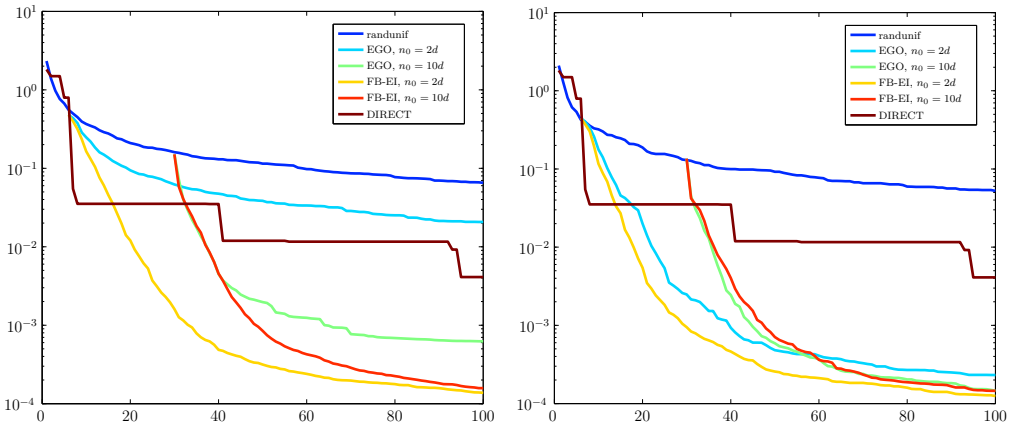
L'existence d'une formule analytique de l'EI (2.5) lorsque le processus aléatoire ξ est choisi gaussien est un atout majeur. Cependant, considérer un modèle gaussien pour ξ amène naturellement à la question du choix du modèle, en particulier concernant la covariance, généralement choisie parmi une classe paramétrée par un vecteur noté θ. Essentiellement deux approches s'ouvrent à nous, une première dite par « substitution » et une seconde « complètement bayésienne ».

L'approche par substitution est la plus simple d'utilisation et la plus largement répandue, en particulier au travers de l'algorithme EGO. Elle consiste à utiliser l'information déjà disponible grâce aux évaluations précédentes afin de calculer une estimation de θ, correspondant par exemple au maximiseur de la vraisemblance, puis de les substituer directement au sein de la formule analytique de l'EI. Ces approches donnent généralement de bons résultats, mais il a été mis en évidence au cours de ce chapitre que cette façon de procéder



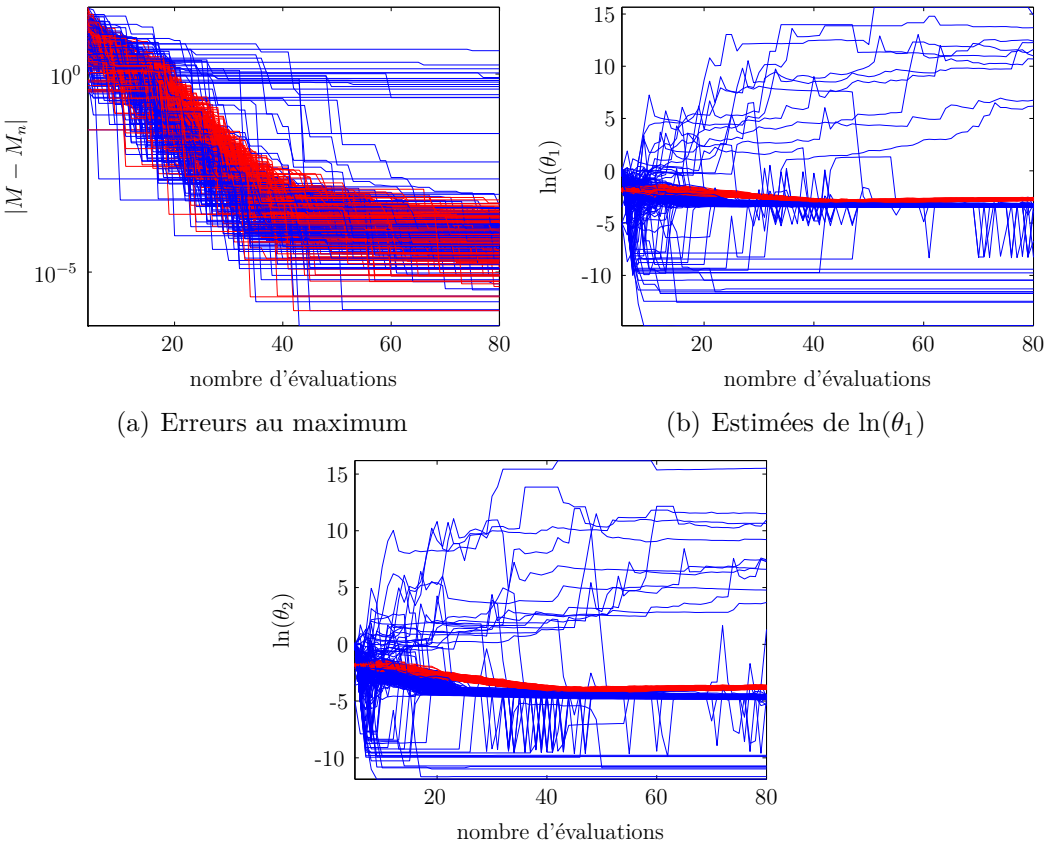
(a) Moyenne (b) Médiane

Figure 3.7 – Comparaison de l'erreur d'estimation du maximum (moyenne et médiane à partir de 200 simulations) en fonction du nombre d'évaluations pour différents algorithmes. La fonction considérée est Branin.



(a) Moyenne (b) Médiane

Figure 3.8 – Comparaison de l'erreur d'estimation du maximum (moyenne et médiane à partir de 200 simulations) pour différents algorithmes. La fonction considérée est Hartman 3 (en log).

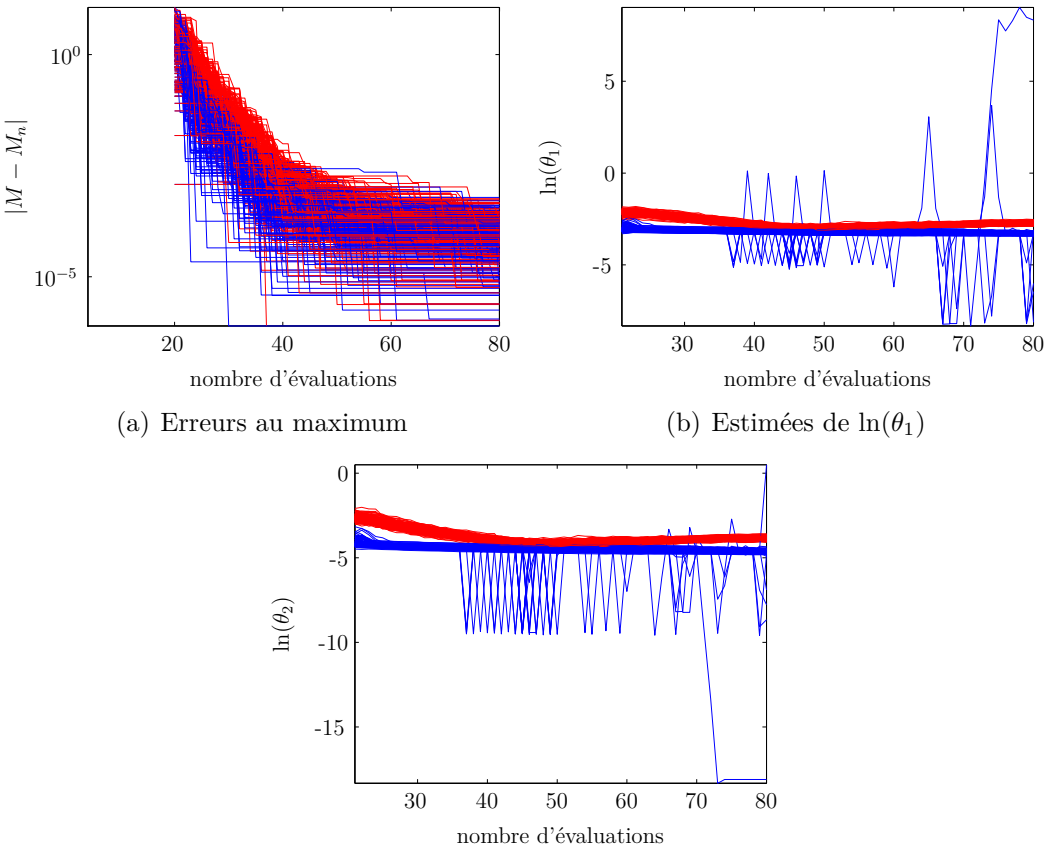


(a) Erreurs au maximum

(b) Estimées de $\ln(\theta_1)$

(c) Estimées de $\ln(\theta_2)$

Figure 3.9 – Comportement comparé, pour $n_0 = 2d$, des algorithmes EGO (en bleu) et FB-EI (en rouge) lors de l'optimisation de la fonction de Branin. Pour chacun des deux algorithmes, les résultats sur l'ensemble des 200 simulations sont représentés. Les grandeurs θ_1 et θ_2 représentent l'inverse des paramètres de portées selon respectivement la première et la seconde variable de la fonction de Branin.



(a) Erreurs au maximum

(b) Estimées de $\ln(\theta_1)$

(c) Estimées de $\ln(\theta_2)$

Figure 3.10 – Comportement comparé, pour $n_0 = 10d$, des algorithmes EGO (en bleu) et FB-EI (en rouge) lors de l'optimisation de la fonction de Branin. Pour chacun des deux algorithmes, les résultats sur l'ensemble des 200 simulations sont représentés. Les grandeurs θ_1 et θ_2 représentent l'inverse des paramètres de portées selon respectivement la première et la seconde variable de la fonction de Branin.

peut s'avérer particulièrement décevante dans certaines situations. L'exemple le plus emblématique est probablement celui des *fonctions trompeuses*, présenté au 3.1.2. En effet, si l'information disponible au préalable est particulièrement trompeuse c'est à dire, dans ce cas, non représentative des variations réelles de la fonction objectif, l'estimation de θ peut alors s'avérer particulièrement mauvaise. Une approche par substitution ne permet pas de rendre compte de cette réalité, le vecteur θ estimé étant directement « injecté » au sein de l'expression du critère. L'incertitude associée à l'estimation de θ n'est ainsi nullement prise en compte dans ce type d'algorithme.

L'approche complètement bayésienne, quant à elle, n'élude pas la question de l'incertitude inhérente au choix des paramètres. Le caractère complètement bayésien se traduit par l'utilisation, en plus de l'*a priori* ξ sur la fonction f, d'un nouvel *a priori* sur θ. Si se donner un *a priori* sur θ est également possible dans le cadre d'une approche par substitution (utilisation du maximum *a posteriori* par exemple), elle revêt ici un rôle différent. En effet, il n'est plus question de calculer une valeur estimée mais bien d'intégrer le critère EI sur l'ensemble des valeurs possibles de θ à partir de la mesure associée à la loi *a posteriori* (dépendant, bien évidemment, de l'*a priori* choisi). Procéder de la sorte permet la prise en compte de *toutes* les valeurs possibles de θ, chacune pondérée en accord avec les valeurs de la loi *a posteriori*. Il s'agit d'une façon de prendre en compte l'incertitude, ce qui offre potentiellement la possibilité de construire des algorithmes à la robustesse accrue. Les tests présentés dans ce chapitre montrent une réelle amélioration apportée par une approche complètement bayésienne en comparaison avec un algorithme par substitution tel que EGO. Ceci est particulièrement visible lorsque la taille du plan d'expérience initial n_0 est égale à $2d$.

L'utilisation d'une approche complètement bayésienne nécessite néanmoins de calculer efficacement une estimation de l'intégrale de l'EI (3.8) sur l'ensemble des valeurs possibles de θ. La façon dont nous avons surmonté cette difficulté est développée au chapitre suivant. De plus, nous y présenterons un nouvel algorithme ayant une spécificité supplémentaire. Celle-ci permet une optimisation efficace du critère EI lors du choix d'une nouvelle évaluation.

Chapitre 4

Construction d'un algorithme complètement bayésien utilisant une approche Monte-Carlo séquentielle

Le chapitre 3 a mis en avant l'intérêt d'une approche complètement bayésienne par rapport à l'approche par substitution dans le cadre de l'optimisation d'une fonction à l'aide du critère EI. Cependant, pour utiliser une telle approche, il faut être capable de calculer efficacement le critère EI complètement bayésien (1.4), ce qui nécessite de construire une approximation numérique de l'intégrale sur l'espace Θ, par rapport à la loi *a posteriori* des paramètres. Par ailleurs, le problème de la maximisation du critère EI sur le domaine $\mathbb{X}$ doit également être abordé. Dans cette thèse, nous proposons une approche SMC (Monte Carlo séquentielle ou *sequential Monte Carlo* en anglais) permettant de résoudre conjointement les deux difficultés. La section 4.1 porte essentiellement sur la question de l'intégration. La section 4.2 concerne la maximisation du critère EI et décrit des méthodes déjà existantes. La section 4.3 introduit un nouvel algorithme qui constitue la contribution principale de la thèse. Des considérations sur la complexité de l'algorithme sont discutées à la section 4.4. Pour finir, la section 4.5 donne des illustrations numériques du fonctionnement

de notre algorithme.

4.1 Intégration en θ par méthode de Monte Carlo sequentielle

4.1.1 Principe

Nous rappelons dans cette section le principe d'une approche SMC (voir Robert et Casella, 2004 ; Del Moral et al., 2006 ; Cappé et al., 2007 ; Liu, 2008, ainsi que les références associées) afin d'estimer la valeur du critère EI complètement bayésien. Une alternative faisant appel à des méthodes de quadratures bayésiennes (O'Hagan, 1991) a été proposée dans Osborne et al. (2008, 2009) ; Osborne (2010). Dans tous les cas, le critère EI est approché par une expression de la forme $\sum_i w_i \rho_n(x; \theta'_i)$, ce qui revient à approcher π_n par $\sum_i w_i \delta_{\theta'_i}$.

L'algorithme présenté dans cette section repose sur des principes classiques, utilisés pour la première fois, pour le calcul d'un critère EI bayésien, par Gramacy et Polson (2011). Le principe général consiste à générer (séquentiellement) un ensemble $\mathfrak{T}_n = \{(\theta_{n,i}, w_{n,i}) \in \Theta \times \mathbb{R}, 1 \leq i \leq I\}$ distribué selon la densité *a posteriori* $\pi_n(\theta)$, pour $n = 1, 2, 3 \ldots$, et d'approcher le critère EI complètement bayésien, dont l'expression est donnée au (3.8), à l'aide de la somme finie

$$\sum_{i=1}^{I} w_{n,i}\ \tilde{\rho}_n(x; \theta_{n,i}) \to_I \rho_n(x) = \int \tilde{\rho}_n(x; \theta)\ \pi_n(\theta)\ \mathrm{d}\theta. \tag{4.1}$$

La façon de contruire ces ensembles $\mathfrak{T}_n$ est donc un point essentiel. Nous partons du principe que, pour $n \in \mathbb{N}$, les densités *a posteriori* successives π_n et π_{n+1} sont proches. L'idée est alors de profiter de cette caractéristique en construisant l'ensemble $\mathfrak{T}_{n+1}$ à partir de $\mathfrak{T}_n$. Une approche SMC, telle que celle proposée par Chopin (2002), est particulièrement indiquée dans ce contexte. Dans ce cadre, le terme *particule* désigne une paire $(\theta_{n,i}, w_{n,i})$ pour i fixé.

Lorsque le résultat d'une nouvelle évaluation est disponible, la première étape, dite de *pondération*, consiste à mettre à jour les poids afin que les particules soient distribuées selon la nouvelle densité *a posteriori* π_{n+1}. Comme

les zones de fortes probabilité pour la densité π_{n+1} ne sont pas exactement les mêmes que celles de π_n et, à plus forte raison, que celle de π_0, il peut arriver que certaines particules se voient attribuer un poids extrêmement faible. Il n'est ainsi pas satisfaisant de seulement repondérer les particules en gardant toujours les mêmes valeurs $\theta_{n,i}$. La seconde étape consiste donc à regénérer les particules. Cette étape peut elle-même être décomposée en deux phases : un rééchantillonnage pour éliminer les particules de poids les plus faibles (et donc les moins intéressantes), et une étape de « déplacement ». L'ensemble de ces étapes sont décrites en détails aux sections 4.1.2 et 4.1.3.

4.1.2 Étapes de la mise en œuvre proposée

L'ensemble du processus depuis l'initialisation jusqu'aux opérations permettant de passer de $\mathfrak{T}_n$ à $\mathfrak{T}_{n+1}$ est maintenant décrit plus en détails.

Initialisation À l'initialisation, nous considérons un plan d'expériences de n_0 évaluations, et nous construisons un échantillon pondéré

$$\mathfrak{T}_{n_0} = \{(\theta_{n_0,i}, w_{n_0,i}), 1 \leq i \leq I\}$$

réparti selon la densité *a posteriori* π_{n_0}. Échantillonner les $\theta_{n_0,i}$ directement à partir de π_{n_0} n'est pas possible dans le cas général. L'idée retenue est de générer des particules à partir de l'*a priori* π_0, généralement choisi suffisamment simple pour permettre un échantillonnage direct, puis d'utiliser pour chaque particule un algorithme de Metropolis-Hastings indépendant (IMH) de densité invariante π_{n_0}(voir, par exemple, Robert et Casella, 2004, pour plus de détails sur l'algorithme de Metropolis-Hastings en général). Il est nécessaire d'effectuer un certain nombre de transitions IMH. En effet, lorsque la taille du plan d'expérience initial n_0 est grande, les densités π_0 et π_{n_0} peuvent être très différentes l'une de l'autre. Nous choisissons $m_0 = 10$ pas IMH dans les expériences numériques que nous avons conduites. Les valeurs des poids $w_{n_0,i}$ sont toutes prises égales à $1/I$. L'utilisation précise que nous faisons de l'algorithme IMH est décrite à la section 4.1.3.

Pondération/échantillonnage/déplacement Pour $n \geq n_0$, le résultat de l'évaluation en un nouveau point d'échantillonnage X_{n+1}, fournit de l'information supplémentaire et permet de considérer la densité *a posteriori* mise à jour π_{n+1}. Nous voulons disposer de particules pondérées réparties selon cette nouvelle densité. Nous savons déjà, par construction, que les particules pondérées de l'ensemble $\mathfrak{T}_n = \{(\theta_{n,i}, w_{n,i}),\ 1 \leq i \leq I\}$ sont distribuées selon π_n. Nous utilisons à nouveau les $\theta_{n,i}$ précédents, mais en leur associant un nouveau poids $w'_{n+1,i}$, afin d'en obtenir une répartition selon π_{n+1}. Un choix possible (voir Liu et Chen, 1998 ; Gilks et Berzuini, 2001 ; Chopin, 2002, pour une justification) pour ces nouveaux poids est

$$w'_{n+1,i} \propto \frac{\pi_{n+1}(\theta_{n,i})}{\pi_n(\theta_{n,i})}\, w_{n,i},\ i = 1, 2,\ \ldots\ I$$

calculés grâce à l'expression analytique des densités *a posteriori* donnée en annexe D.5. Cette façon de procéder est classique pour les méthodes de type SMC, et le facteur π_{n+1}/π_n est appelé *poids incrémental*. Comme voulu, les particules de la forme $\mathfrak{T}^0_n = \{\theta_{n,i}, w'_{n+1,i}, 1 \leq i \leq I\}$ sont réparties selon π_{n+1}.

Néanmoins, garder à chaque itération de l'algorithme les mêmes valeurs de θ, même si la pondération change, n'est pas raisonnable car les densités *a posteriori* π_n et π_{n+k} peuvent être relativement différentes l'une de l'autre pour un entier k grand. Afin de favoriser les particules de poids les plus élevés, nous utilisons une étape de rééchantillonnage. Pour la simplicité de mise en œuvre, nous avons fait le choix de l'échantillonnage « multinomial », dont des descriptions peuvent être trouvées aux sections 3.1 de Del Moral et al. (2006) et 2.3 de Douc et Moulines (2008). Concrètement, cela consiste à générer de façon indépendante et identiquement distribué I valeurs $\theta'_{n+1,i}$ à partir de la loi $\sum_{i=1}^{I} w'_{n+1,i}\, \delta_{\theta_{n,i}}$, et à considérer une pondération uniforme, c'est à dire des poids $w_{n+1,i} = 1/I$, pour ces nouvelles valeurs. Nous notons $\mathfrak{T}'_{n+1} = \{(\theta'_{n+1,i}, w_{n+1,i}),\ 1 \leq i \leq I\}$ l'ensemble pondéré associé. D'après le théorème 3 de Douc et Moulines (2008), les particules de $\mathfrak{T}'_{n+1}$ sont distribuées selon la densité π_{n+1}.

Une telle approche permet d'éliminer les particules ne représentant plus d'intérêt, mais elle réduit également le nombre de valeurs distinctes de θ, ce qui est appelé un phénomène de *dégénérescence*. Afin d'éviter cet écueil, nous

considérons une étape dite de *déplacement* consistant à utiliser un algorithme IMH, de densité invariante π_{n+1}. Dans nos expériences numériques, nous faisons trois pas d'IMH. L'emploi que nous faisons de cet algorithme IMH est très similaire à ce qui est fait pour l'initialisation, et il suffit donc de se reporter à nouveau à la section 4.1.3 pour plus de détails. Nous notons $\theta_{n+1,i}$ les nouvelles valeurs obtenues.

À la fin de cette procédure, nous disposons d'un ensemble

$$\mathfrak{T}_{n+1} = \{(\theta_{n+1,i}, w_{n+1,i}), 1 \leq i \leq I\}$$

vérifiant les propriétés de convergence de la relation (4.1), et permettant de calculer le critère EI dans sa version complètement bayésienne. Nous faisons remarquer à nouveau que, par construction, les valeurs $w_{n+1,i}$ sont toutes égales à $1/I$ (ce qui pourrait être différent si un échantillonnage autre que multinomial était considéré). Le fonctionnement de l'algorithme est résumé à l'aide du schéma bloc présenté dans la figure 4.1.

4.1.3 Algorithme de Metropolis Hastings indépendant

Nous voulons générer un nouvel ensemble de particules $\mathfrak{T}_{n+1}$, distribué selon la densité cible π_{n+1}, à partir de l'ensemble $\mathfrak{T}^1_{n+1} = \{(\theta^1_{n+1,i}, w_{n+1,i}), 1 \leq i \leq I\}$. La méthode employée est proche de ce que fait Chopin (2002).

Nous procédons de façon séquentielle. Pour $k \geq 1$, nous supposons disposer d'un ensemble

$$\mathfrak{T}^k_{n+1} = \{(\theta^k_{n+1,i}, w_{n+1,i}), 1 \leq i \leq I\}$$

réparti selon π_{n+1}. Nous utilisons les particules de $\mathfrak{T}^k_{n+1}$ afin de construire une densité instrumentale q. Pour satisfaire cette contrainte, nous choisissons $q = \phi_{\widehat{\mu}_k, \widehat{\Sigma}^2_k}$ où $\widehat{\mu}_k$ est la moyenne pondérée des particules de l'ensemble $\mathfrak{T}^k_{n+1}$, $\widehat{\Sigma}^2_k$ leur matrice de covariance, et ϕ_{μ,Σ^2} est la densité gaussienne de moyenne μ et de matrice de covariance Σ^2. Nous avons donc

$$\widehat{\mu}_k = \sum_{i=1}^{I} w_{n+1,i}\, \theta^k_{n+1,i},$$

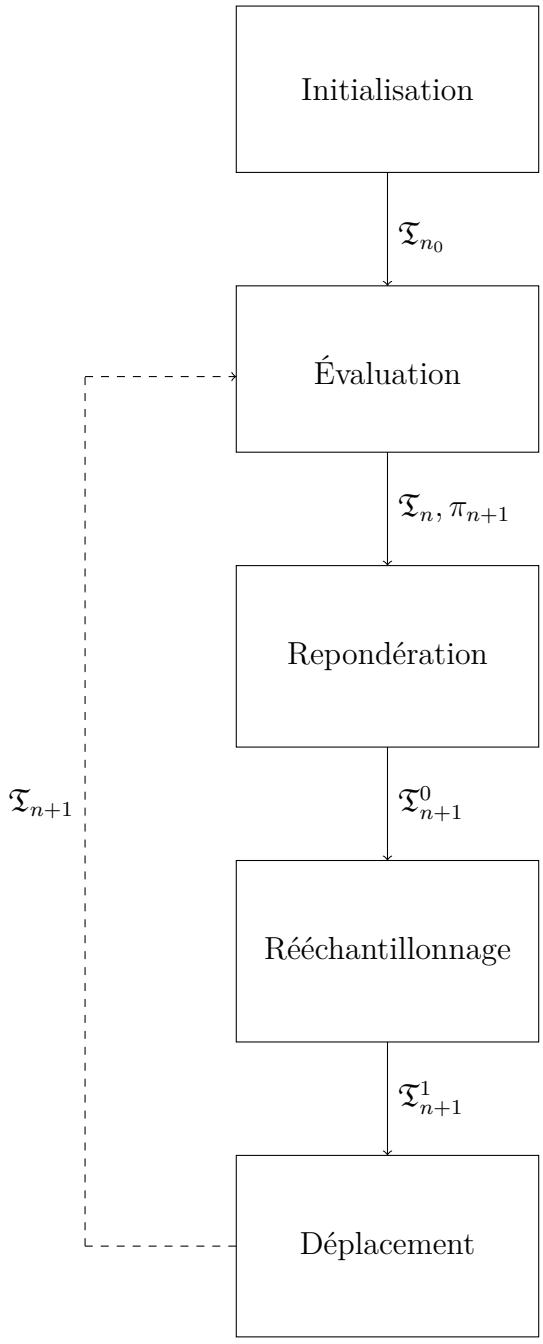


Figure 4.1 – Schéma bloc du fonctionnement de notre algorithme complètement bayésien avec utilisation de méthodes SMC sur les valeurs de θ.

et

$$\widehat{\Sigma}_k^2 = \sum_{i=1}^{I} w_{n+1,i} \, (\theta_{n+1,i}^k - \widehat{\mu}_k)(\theta_{n+1,i}^k - \widehat{\mu}_k)^T.$$

À partir de cette densité q, un nouvel ensemble $\{\tilde{\theta}_{n+1,i}^k,\ 1 \leq i \leq I\}$ de I valeurs de θ est construit. Pour ce faire, nous utilisons pour chaque particule de $\mathfrak{T}_{n+1}^k$ un algorithme de Metroplis Hastings indépendant (IMH) de densité invariante π_{n+1} et de densité instrumentale q. Chacune des I nouvelles valeurs de θ peut être *acceptée* ou *refusée* :

$$\theta_{n+1,i}^{k+1} = \begin{cases} \tilde{\theta}_{n+1,i}^k & \text{avec une probabilité } r_i, \\ \theta_{n+1,i}^k & \text{avec une probabilité } 1 - r_i, \end{cases}$$

où

$$r_i = \min\left(1, \frac{\pi_{n+1}(\tilde{\theta}_{n+1,i}^k)\, q(\theta_{n+1,i}^k)}{\pi_{n+1}(\theta_{n+1,i}^k)\, q(\tilde{\theta}_{n+1,i}^k)}\right). \tag{4.2}$$

En gardant la pondération associée à $\mathfrak{T}_{n+1}^k$ (uniforme dans le cas présent, mais cela serait vrai dans une situation plus générale), l'ensemble

$$\mathfrak{T}_{n+1}^{k+1} = \{(\theta_{n+1,i}^{k+1}, w_{n+1,i})), 1 \leq i \leq I\}$$

est bien réparti selon π_{n+1} d'après le théorème 1 de Douc et Moulines (2008). Nous utilisons m pas de l'algorithme de Metropolis-Hastings, et prenons $\mathfrak{T}_{n+1} = \mathfrak{T}_{n+1}^m$. En effet, plusieurs pas de Metropolis-Hastings permettent d'augmenter la diversité. À l'initialisation, nous prenons $m = 10$, puis $m = 3$ lors de chacune des itérations suivantes.

4.2 Stratégies de maximisation du critère EI

Cette section a pour objectif d'évoquer brièvement des stratégies de la littérature possibles pour la maximisation du critère EI.

Bates et Pronzato (2001) proposent d'optimiser l'EI en décomposant le domaine de définition à l'aide d'une triangulation de Delaunay, dont les sommets

correspondent aux points d'évaluation. Une telle construction permet généralement d'obtenir un caractère unimodal de l'EI dans chacune des cellules. Sans entrer dans les détails, le nouveau point d'évaluation est choisi à l'issue d'une recherche locale dans chaque cellule.

Jones et al. (1998) proposent d'utiliser un algorithme de type *branch-and-bound*. Pour cela, il est nécessaire de calculer une borne supérieure de l'EI sur tout pavé rectangulaire du domaine $\mathbb{X}$. La façon de procéder décrite dans l'article consiste à calculer des bornes supérieures de l'EI à partir des ses dérivées partielles relativement à $\widehat{\xi}_n(x)$ et $s_n(x)$, dérivées dont les expressions sont particulièrement simples. Néanmoins, la méthode décrite dans l'article est tributaire du choix d'une fonction de covariance exponentielle séparable pour le processus aléatoire ξ, ce qui en restreint la généralité.

Ginsbourger et Roustant (2011) délèguent la maximisation du critère à un algorithme génétique, faisant usage du gradient de l'EI, disponible dans le *package rgenoud* (Walter et Jasjeet, 2011) du logiciel R .

En ce qui concerne Gramacy et Polson (2011), l'approche utilisée est la suivante. Lorsque n évaluations sont disponibles, un ensemble de $I \in \mathbb{N}$ points candidats $\mathfrak{X}_n = x_{n,i}, 1 \leq i \leq I$ est échantillonné par LHS. Un point supplémentaire x'_{n+1} est ajouté à cet ensemble tel que $x'_{n+1} = \operatorname{argmax}_x \widehat{\xi}_{n+1}(x \mid \theta_{i^\star})$ avec $i^\star = \operatorname{argmax}_i \pi_n(\theta_i)$, autrement dit le maximiseur du prédicteur associé à la particule en θ maximisant la loi *a posteriori*. La maximisation du prédicteur s'effectue à partir d'un algorithme d'optimisation, initialisé à l'optimum du prédicteur sur l'ensemble $\mathfrak{X}_n$. Il suffit alors de maximiser le critère EI sur l'ensemble $x'_{n+1} \cup \mathfrak{X}_n$, où $\mathfrak{X}_n$ permet une optimisation globale tandis que x'_{n+1} en permet une plutôt locale.

Bardenet et Kégl (2010) proposent quant à eux, afin d'échantillonner itérativement dans des régions où le critère est élevé, d'utiliser un algorithme de type *cross-entropy maximization* (CEM) avec une loi instrumentale correspondant à un mélange de gaussiennes, et ce, afin de prendre en compte le caractère multi-modal du critère EI. Des résultats expérimentaux montrent que cette approche offre une recherche plus efficace, en dimension deux, qu'une recherche exhaustive sur grille et, en dimension dix, qu'une recherche sur un

LHS. Cependant cette approche demande de régler certains paramètres, comme le nombre de composantes du mélange de gaussiennes. Une autre difficulté est liée à la procédure d'initialisation dont la complexité algorithmique augmente fortement avec la dimension (*curse of dimensionality*).

4.3 SMC en (θ, x) : description de l'algorithme

Cette section présente un nouvel algorithme apportant une réponse conjointe à la question du calcul de l'EI et d'une maximisation efficace du critère. Le choix que nous faisons est de considérer une approche SMC non plus sur les seules valeurs du paramètre θ mais plutôt sur des valeurs de couples, associant ainsi à chaque valeur de θ un point candidat $x \in \mathbb{X}$.

4.3.1 Construction d'une densité sur le domaine $\mathbb{X}$

L'étape de maximisation du critère EI complétement bayésien sur le domaine $\mathbb{X}$ est particulièrement importante, comme évoqué dans l'introduction et dans la section 4.2. La réponse que nous apportons est de maximiser ρ_n sur un ensemble de points candidats $\mathfrak{X}_n = \{x_{n,j}\} \subset \mathbb{X}$, que nous construisons afin de satisfaire un compromis faisant intervenir deux propriétés. La première est de contenir un point suffisamment proche du maximum réel de ρ_n pour que la maximisation soit effective. La seconde est de garder un nombre de points candidats suffisamment faible pour que les estimations du critère, calculées à partir de l'approche SMC en θ vue à la section 4.1, ne soient pas trop nombreuses, et offrir ainsi une complexité algorithmique acceptable[1]. Afin de limiter le nombre de points candidats, tout en préservant une bonne estimation du maximum, nous faisons en sorte qu'ils soient répartis selon une densité « pertinente » p_n. Nous entendons ici par « pertinente », une densité qui favorise les zones du domaine où les valeurs du critère ont tendance à être élevées. Au chapitre 2, nous avons passé en revue un ensemble de critères apportant

1. l'utilisation de méthodes SMC en θ permet d'obtenir une estimation efficace du critère, mais calculer cette estimation sur un nombre particulièrement grand de points candidats reste rédhibitoire, comme dit précédemment.

une caractérisation des zones intéressantes du domaine. Ils étaient définis pour un processus ξ dont le paramètre θ était implicitement supposé connu. C'est à partir de ces critères, que nous noterons de façon générique $g_n : \mathbb{X} \times \Theta \to \mathbb{R}^+$, que nous allons chercher à construire p_n.

Obtenir une densité à partir d'une fonction g_n positive bornée peut se faire en principe en normalisant cette dernière. Deux options peuvent être considérées. La première, probablement la plus naturelle, consiste à choisir une densité de la forme

$$p_n(x) \propto \int_\theta g_n(x \mid \theta)\pi_n(\theta)\, \mathrm{d}\theta. \tag{4.3}$$

Néanmoins, cette construction présente un désavantage. En effet, bien que p_n s'avère être la marginale en x de la densité

$$\begin{array}{rccc} \pi'_n : & \Theta \times \mathbb{X} & \to & [0\,;+\infty[\\ & \gamma = (\theta, x) & \mapsto & \frac{g_n(x|\theta)\pi_n(\theta)}{\int_{\mathbb{X}}\int_\Theta g_n(x|\theta')\pi_n(\theta')\,\mathrm{d}\theta'\,\mathrm{d}x}, \end{array}$$

la marginale en θ de π'_n n'est pas la densité *a posteriori* π_n, ce qui ne permet pas de considérer une approche SMC conjointe sur les espaces Θ et $\mathbb{X}$. Pour cette raison, nous choisissons une autre construction et considérons

$$p_n(x) = \int_\Theta \tilde{g}_n(x \mid \theta)\ \pi_n(\theta)\ \mathrm{d}\theta, \tag{4.4}$$

où $\tilde{g}_n(x \mid \theta) = g_n(x \mid \theta)/c_n(\theta)$, et où $c_n(\theta)$ est un terme de normalisation valant $\int_{\mathbb{X}} g_n(x \mid \theta)\, \mathrm{d}x$. Dans ce cas, p_n est la marginale en x de la densité

$$\begin{array}{rccc} \pi'_n : & \Theta \times \mathbb{X} & \to & [0\,;+\infty[\\ & \gamma = (\theta, x) & \mapsto & \tilde{g}_n(x \mid \theta)\ \pi_n(\theta), \end{array}$$

dont la marginale en θ est bien π_n. Remarquons que l'intégrale (4.4) n'est jamais calculée explicitement en pratique. C'est la façon dont les points candidats sont échantillonnés qui permet l'estimation de p_n (voir la section 4.3.2).

Parmi les critères présentés au chapitre 2, plusieurs choix s'offrent à nous pour la fonction g_n. Une possibilité immédiate est d'utiliser à nouveau le critère

EI. Une autre est de considérer la probabilité d'amélioration. Nulle évidence ne s'impose pour un tel choix, mais des considérations relatives au compromis local/global peuvent néanmoins constituer une motivation. Les résultats du chapitre 2, par exemple, indiquent que prendre la probabilité d'amélioration pour g_n favorise plus nettement les zones au voisinage des évaluations les plus hautes que ne le ferait l'EI, plus enclin, quant à lui, à l'exploration. Il est possible de construire des variantes en faisant intervenir un critère ρ_n^α, avec $\alpha > 0$, au lieu de l'EI classique (ceci s'adapte également à la probabilité d'amélioration). Pour $\alpha > 1$, le caractère global est favorisé, inversement pour $\alpha < 1$. La figure 4.2 montre, pour un même plan d'expérience initial de quatre points pour une fonction test de dimension un, la valeur de différents critères possibles g_n sur le domaine. Les critères considérés sont la probabilité d'amélioration et trois variantes d'EI. Une rapide étude comparative permet de caractériser le caractère local ou global de chacun d'eux.

Dans la suite du manuscrit, pour l'ensemble des simulations effectuées, nous choisissons pour p_n la probabilité d'amélioration normalisée. L'idée est d'obtenir une forte densité de points candidats au voisinage du maximum courant de l'EI, ce qui correspond à une zone du domaine où le vrai maximum est susceptible de se trouver. De plus, la variabilité entre p_n et p_{n+1} sur cette zone est généralement plus faible pour la probabilité d'amélioration que pour l'EI (ce qui peut s'avérer utile dans le cadre d'une approche SMC, où chaque nouvelle densité est estimée à partir de la précédente).

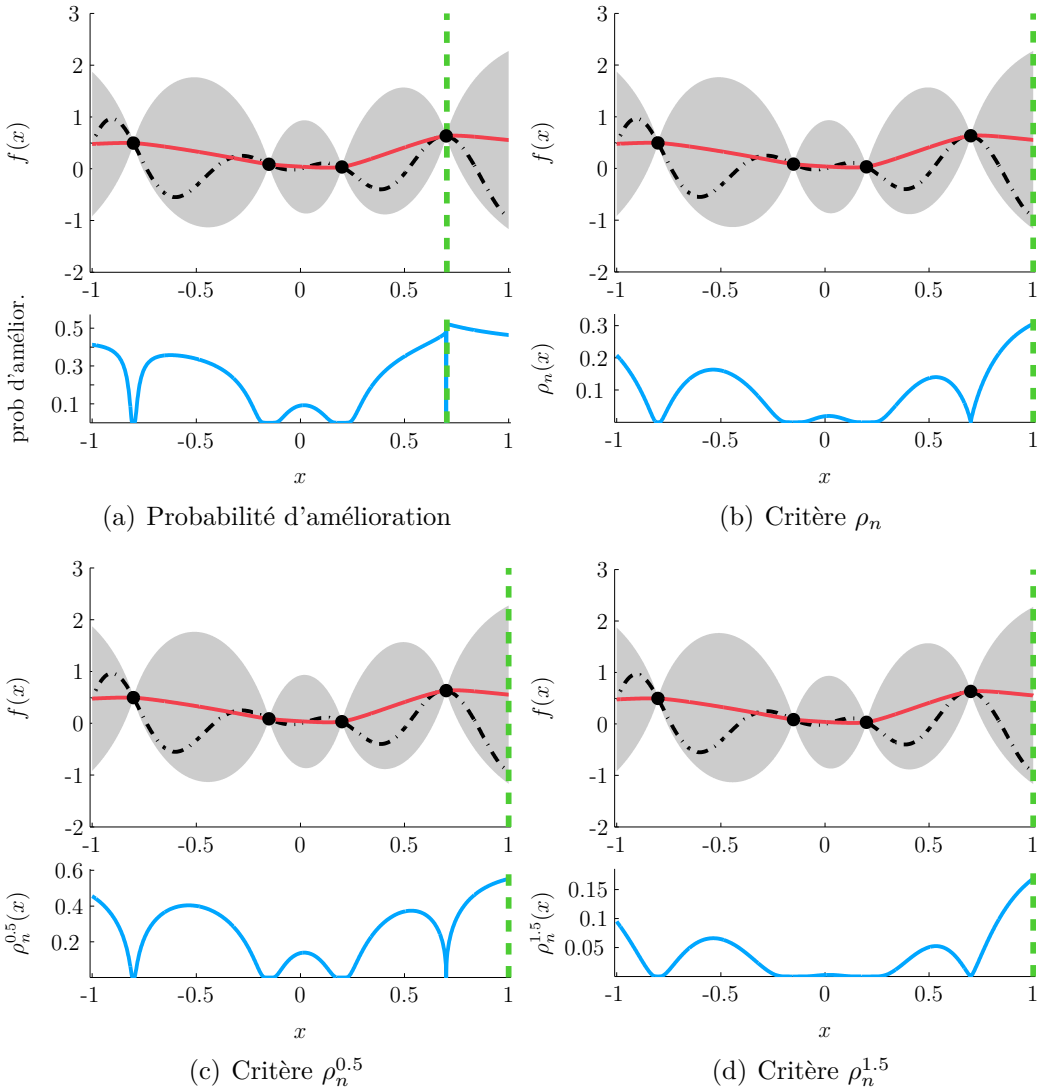
4.3.2 Principe de l'algorithme SMC(θ, x)

L'objectif est d'obtenir un ensemble de particules pondérées

$$\mathfrak{G}_n = \{(\gamma_{n,i}, w_{n,i}), 1 \leq i \leq I\}, \quad \gamma_{n,i} = (\theta_{n,i}, x_{n,i}), \tag{4.5}$$

distribué selon la densité π'_n définie à la section 4.3.1.

Une contrainte importante que nous prenons en compte dans la construction de cet algorithme est d'obtenir une complexité algorithmique acceptable. Chaque itération nécessite I nouvelles évaluations de vraisemblance, comme indiqué par la relation (4.2). Ceci nous incite à considérer un petit nombre



(a) Probabilité d'amélioration

(b) Critère ρ_n

(c) Critère $\rho_n^{0.5}$

(d) Critère $\rho_n^{1.5}$

Figure 4.2 – Comparaisons entre différents critères à partir d'une situation identique en dimension un (même fonction test, et même plan d'expérience initiale de taille $n_0 = 4$). Les valeurs atteintes par ces critères sur le domaine caractérise leur caractère exploratoire ou local. Nous remarquons que la probabilité d'amélioration est maximum au voisinage de l'optimum courant, tandis que l'EI et ses variantes atteignent leurs valeurs les plus hautes sur les bords. Il est à noter également que, pour un critère du type ρ_n^α, le caractère exploratoire est de plus en plus prononcé lorsque α augmente.

de valeurs $\theta_{n,i}$ et donc de particules $\gamma \in \mathbb{G} = \Theta \times \mathbb{X}$. Cependant, considérer peu de points candidats $x_{n,i}$ est préjudiciable car cela dégrade l'estimation du maximum de l'EI.

Afin de surmonter cet écueil, nous proposons une structure différente où, pour une même valeur de θ, plusieurs points candidats x sont considérés. Plus précisémment, nous utilisons l'ensemble de particules suivant

$$\mathfrak{G}_n = \{(\gamma_{n,i,j}, w_{n,i,j}), 1 \leq i \leq I, 1 \leq j \leq J_i\}, \quad \gamma_{n,i,j} = (\theta_{n,i}, x_{n,i,j}), \tag{4.6}$$

avec $\sum_{i=1}^{I} \sum_{j=1}^{J_i} w_{n,i,j} = 1$, $J_i \geq 1$, et $\sum_{i=1}^{I} J_i = N_\gamma$ le nombre total de particules. Le nombre I de valeurs distinctes de θ est ainsi plus faible que N_γ. Dans la suite, toutes les valeurs J_i sont prises égales à une même valeur J, impliquant un nombre de particule N_γ égal à IJ. Il est à noter que l'ensemble pondéré issu de la marginalisation selon la première variable θ a au plus I valeurs distinctes, et peut ainsi s'écrire

$$\mathfrak{T}_n = \{(\theta_{n,i}, w_{n,i}), 1 \leq i \leq I\}, \quad \sum_{i=1}^{I} w_{n,i} = 1. \tag{4.7}$$

Nous décrivons ci-dessous le fonctionnement de l'algorithme, illustré également sous la forme du schéma bloc présenté à la figure 4.3. Le déroulement des étapes reprend en partie celui de l'algorithme présenté à la section 4.1. Les aspects spécifiques sont donnés dans les sections 4.3.1 à 4.3.6.

Algorithme 1. *Algorithme SMC*(θ, x)

À l'initialisation, un échantillon pondéré $\mathfrak{T}_{n_0} = \{(\theta_{n_0,i}, w_{n_0,i}),\ 1 \leq i \leq I\}$ *est généré à partir de la densité* π_{n_0} *(voir la section 4.1.2). Nous choisissons une densité (uniforme, pour nos expériences numériques)* q_{n_0} *sur* $\mathbb{X}$*. Ensuite, pour* $n \geq n_0$*, étant donnés* $\mathfrak{T}_n$ *et* q_n*, l'algorithme procède en quatre étapes,*

Étape 1 : démarginalisation — *Utiliser* $\mathfrak{T}_n$ *et* q_n *pour construire un ensemble* $\mathfrak{G}_n$*, avec* $x_{n,i,j} \overset{iid}{\sim} q_n$*,* $w_{n,i,j} \propto w_{n,i} \frac{g_n(x_{n,i,j}|\theta_{n,i})}{q_n(x_{n,i,j}) c_{n,i}}$*, où* $c_{n,i} = \frac{1}{N_\gamma} \sum_{i'=1}^{I} \sum_{j'=1}^{J} \frac{g_n(x_{n,i',j'}|\theta_{n,i})}{q_n(x_{n,i',j'})}$*. L'ensemble* $\mathfrak{G}_n$ *se construit alors en considérant les valeurs* $\gamma_{n,i,j} = (\theta_{n,i}, x_{n,i,j})$ *associées aux poids* $w_{n,i,j}$*.*

Étape 2 : calcul et maximisation du critère EI — *Évaluer* ξ *en* $X_{n+1} = \operatorname{argmax}_{i,j} \sum_{i'=1}^{I} w_{n,i'}\, \rho_n(x_{n,i,j}; \theta_{n,i'})$*.*

Étape 3 : pondération/échantillonnage/déplacement — *Construire $\mathfrak{T}_{n+1}$ à partir de $\mathfrak{T}_n$: repondérer les $\theta_{n,i}$s avec $w_{n+1,i} \propto \frac{\pi_{n+1}(\theta_{n,i})}{\pi_n(\theta_{n,i})} w_{n,i}$, rééchantillonner (par exemple à l'aide d'un échantillonnage multinomial), et « déplacer » les $\theta_{n,i}$ pour obtenir les $\theta_{n+1,i}$ à l'aide d'un noyau de Metropolis-Hastings indépendant (voir la section 4.1.3 pour plus de détails).*

Étape 4 : construire q_{n+1} — *Construire une estimation q_{n+1} de la seconde densité marginale de π'_n à partir de $\mathfrak{X}_n = \{(x_{n,i,j}, w_{n,i,j}), 1 \leq i \leq I, 1 \leq j \leq J\}$. Tout estimateur de densité (qu'il soit paramétrique ou non) peut être utilisé, à condition qu'il permette un échantillonnage facile. Nous avons, au cours de nos travaux, utilisé un estimateur à base d'arbres (voir description à la section 4.3.6).*

4.3.3 Étape 1 : Démarginalisation

Nous examinons ici plus en détails l'étape de démarginalisation de l'algorithme. Elle consiste à construire un ensemble

$$\mathfrak{G}_n = \{(\gamma_{n,i,j}, w_{n,i,j}), 1 \leq i \leq I, 1 \leq j \leq J\}$$

de particules pondérées réparties selon la densité *a posteriori*

$$\pi'_n(\gamma) = \pi_n(\theta)\ \tilde{g}_n(x \mid \theta),$$

ce qui permet d'obtenir une répartition des θ selon π_n, et des x selon p_n.

Nous allons donner, dans ce qui suit, une justification empirique au calcul des poids $w_{n,i,j}$. Considérons une fonction mesurable $h : \Theta \times \mathbb{X} \to \mathbb{R}$, telle que

$$\int_{\Theta \times \mathbb{X}} h(\theta, x)\ \pi'_n(\theta, x)\ \mathrm{d}\theta\ \mathrm{d}x < +\infty,$$

et intéressons nous à la quantité

$$\mathcal{I}(h) \ = \ \sum_{i=1}^{I} \sum_{j=1}^{J} w_{n,i} \frac{\tilde{g}_n(x_{n,i,j} \mid \theta_{n,i})}{q_n(x_{n,i,j})}\ h(\theta_{n,i}, x_{n,i,j}), \tag{4.8}$$

avec les $\theta_{n,i}$ issus de $\mathfrak{T}_n$ et les $x_{n,i,j}$ échantillonnés de façon indépendante et identique selon q_n. Lorsque J est grand, la quantité $\mathcal{I}(h)$ semble être une

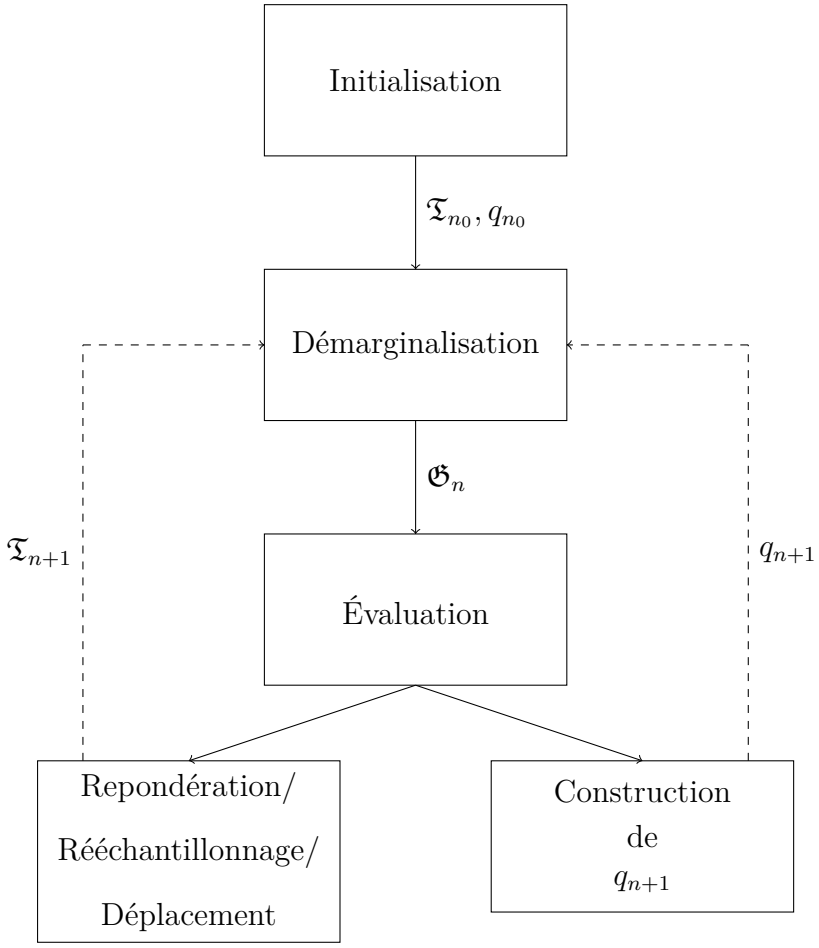


Figure 4.3 – Schéma bloc du fonctionnement de l'algorithme avec approche SMC sur les couples $(\theta_{n,i}, x_{n,i,j})$.

approximation raisonnable de

$$\tilde{\mathcal{I}}(h) \;=\; \sum_{i=1}^{I} w_{n,i} \int_{\mathbb{X}} h(\theta_{n,i}, x)\; \tilde{g}_n(x \mid \theta_{n,i})\; \mathrm{d}x. \tag{4.9}$$

(En utilisant un argument de type loi des grands nombres, et le fait que les particules $x_{n,i,j}$ sont indépendantes et identiquement distribuées selon q_n conditionnellement au passé.) Lorsque I est grand, $\tilde{\mathcal{I}}(h)$ est une approximation de

$$\int_{\Theta\times\mathbb{X}} h(\theta, x) \underbrace{\tilde{g}_n(x \mid \theta)\; \pi_n(\theta)}_{=\pi'_n(\theta,x)} \mathrm{d}\theta\; \mathrm{d}x. \tag{4.10}$$

(Il est toujours question de la loi des grands nombres mais, cette fois, pour les particules $(\theta_{n,i}, w_{n,i})$ de l'ensemble $\mathfrak{T}_n$.) Nous avons donc l'approximation, pour I et J grands,

$$\sum_{i=1}^{I}\sum_{j=1}^{J} w_{n,i} \frac{\tilde{g}_n(x_{n,i,j} \mid \theta_{n,i})}{q_n(x_{n,i,j})}\; h(\theta_{n,i}, x_{n,i,j}) \approx \int_{\Theta\times\mathbb{X}} h(\theta, x)\; \pi'_n(\theta, x)\mathrm{d}\theta\; \mathrm{d}x. \tag{4.11}$$

En pratique, nous substituons dans l'expression de $w_{n,i,j}$, à la grandeur $c_n(\theta) = \int_x g_n(x|\theta_{n,i})\; \mathrm{d}x$ qui apparaît dans l'expression de $\tilde{g}_n(x_{n,i,j} \mid \theta_{n,i})$, une approximation $c_{n,i}$ valant

$$\frac{1}{N_\gamma} \sum_{i'=1}^{I}\sum_{j'=1}^{J} \frac{g_n(x_{n,i',j'}|\theta_{n,i})}{q_n(x_{n,i',j'})}. \tag{4.12}$$

Nous justifions cette approximation en invoquant à nouveau un argument de loi des grands nombres. Des considérations sur la complexité algorithmique, induites par le calcul des valeurs $c_{n,i}$, sont discutées à la section 4.4.

Le développement présenté dans cette section n'a nullement l'ambition d'être une démonstration rigoureuse de résultats de convergence. Une démonstration de la validité asymptotique des approximations présentées ici pourrait utiliser des théorèmes de lois limites tels que proposés par Douc et Moulines (2008), mais nous n'avons pas effectué ce travail dans le cadre de cette thèse.

4.3.4 Étape 2 : Calcul et maximisation du critère EI

Le critère est calculé sur l'ensemble des points candidats $x_{n,i,j}$, générés à partir de l'étape 1, choisis afin de favoriser les zones où les valeurs du critère

EI sont élevées. L'objectif est d'estimer efficacement le maximum de l'EI. Le critère en $x_{n,i,j}$ est approché à l'aide de la somme finie

$$\sum_{k=1}^{I} w_{n,k}\, \tilde{\rho}_n(x_{n,i,j};\theta_{n,k}) \to_I \rho_n(x_{n,i,j}) = \int \tilde{\rho}_n(x_{n,i,j};\theta)\, \pi_n(\theta)\, \mathrm{d}\theta. \tag{4.13}$$

La complexité algorithmique de cette étape est discutée à la section 4.4, et une alternative de calcul, moins précise mais faisant intervenir moins de termes dans la somme, est proposée.

4.3.5 Étape 3 : Pondération, réchantillonnage et déplacement

Il suffit de se référer à la section 4.1.2 et plus généralement à l'ensemble de la section 4.1. En effet, cette étape y est rigoureusement identique, la partie SMC sur le domaine $\mathbb{X}$ n'intervenant pas à cette étape de l'algorithme.

4.3.6 Étape 4 : Choix de la densité instrumentale q_n

La construction de l'ensemble des points candidats permet d'obtenir des points répartis selon p_{n+1}, définie à la section 4.3.1. La densité instrumentale q_{n+1} considérée se doit donc d'être proche de p_{n+1} tout en permettant un échantillonnage aisé. La densité p_n étant supposée proche de p_{n+1}, construire un estimateur q_{n+1} de p_n répond au problème. Construire une telle densité q_{n+1} est rendu possible grâce à l'ensemble pondéré $\mathfrak{X}_n$ dont les éléments sont répartis selon p_n.

Nous reprenons l'idée de Klemelä (2009), consistant à estimer des densités à partir d'arbres (ou d'histogrammes dyadiques) dont la construction repose sur des partitions dépendant des données disponibles, $\mathfrak{X}_n$ en l'occurence pour nous. Concrètement, nous considérons un arbre T respectant les contraintes suivantes. Sa racine t_0 correspond à l'ensemble du domaine de définition $\mathbb{X}$ de la fonction. Ce domaine peut se réécrire sous la forme d'un rectangle du type $R_0 = \Pi_{m=1}^{d}[c_m^0, d_m^0]$. Soit t un nœud de T, une sous-partie du domaine $\mathbb{X}$ lui est associée sous la forme du rectangle $R_t = \Pi_{m=1}^{d}[c_m^t, d_m^t]$. Si t n'est pas une feuille, alors une direction (dimension) $l \in \{1, 2, \ldots, d\}$ particulière lui est

associée. Cette direction permet de construire deux fils à t, notés $t_{1,l}$ et $t_{2,l}$, caractérisant deux nouveaux rectangles de volumes égaux construits à partir de la bissectrice de R_t selon la direction l. Dans ce cas, il s'agit respectivement des ensembles

$$R_{t_{1,l}} = \{x \in R_t, x_l < s\}, \qquad R_{t_{2,l}} = \{x \in R_t, x_l \geq s\},$$

avec le point de séparation $s = (d_l - c_l)/2$ et x_l la $l - ième$ composante de x. L'ensemble des rectangles associés aux feuilles d'un tel arbre constitue une partition du domaine $\mathbb{X}$ initial. Nous verrons par la suite que nous associons en réalité à chaque feuille de T une valeur de probabilité, et que la partition pondérée de $\mathbb{X}$ ainsi obtenue induit une densité q.

L'objectif de cette section est de construire un arbre T, vérifiant les contraintes du paragraphe précédent, et induisant un estimateur q_{n+1} de la densité p_n. Nous donnons une description de l'algorithme que nous utilisons pour faire cela, puis nous en justifions les points essentiels. Nous notons T l'arbre construit à l'issue du déroulement de l'algorithme et $L(T)$ l'ensemble de ses feuilles. Nous introduisons également une valeur seuil $\mathfrak{p}_{\min}$, dont l'utilisation est justifiée par la suite.

Algorithme 2. *Construction de la densité instrumentale q_{n+1}.*

À l'initialisation, nous considérons le nœud t_0 associé au domaine $R_{t_0} = \mathbb{X}$ et à une probabilité $\mathfrak{p}_0 = 1$. Ensuite, pour un nœud t associé à une valeur $\mathfrak{p}_t$, nous procédons de la façon suivante,

Étape 1 : — *Si la condition $\mathfrak{p}_t \geq \mathfrak{p}_{\min}$ n'est pas vérifiée, alors $t \in L(T)$. Sinon, il faut aller à l'étape suivante.*

Étape 2 : — *Pour $l \in \{1, 2, \ldots, d\}$, nous notons*

$$U_{t,l} = \frac{\sum_{x_{n,i,j} \in R_{t_{1,l}}} w_{n,i,j}}{\sum_{x_{n,i,j} \in R_t} w_{n,i,j}}. \tag{4.14}$$

Nous décidons de construire $t_{1,l^\star}$ et $t_{2,l^\star}$ les deux fils de t selon la direction

$$l^\star = \underset{l}{\operatorname{argmin}}\{\min\{U_{t,l}\ ;\ 1 - U_{t,l}\}\}. \tag{4.15}$$

Nous retournons à l'étape 1 avec, successivement, le nœud $t_{1,l^\star}$ *et la probabilité*

$$\mathfrak{p}_{t_{1,l^\star}} = \mathfrak{p}_t U_{t,l^\star}, \tag{4.16}$$

puis le nœud $t_{2,l^\star}$ *et la probabilité*

$$\mathfrak{p}_{t_{2,l^\star}} = \mathfrak{p}_t - \mathfrak{p}_{t_{1,l^\star}} = \mathfrak{p}_t(1 - U_{t,l^\star}). \tag{4.17}$$

En pratique, nous utilisons une définition un peu différente pour $\mathfrak{p}_{t_{1,l^\star}}$*. Ceci est expliqué dans la suite.*

Étape finale : — *Nous avons construit un arbre* T *vérifiant*

$$\bigcup_{t \in L(T)} R_t = \mathbb{X},$$

$$R_t \bigcap R_{t'} = \varnothing, \ \forall t, t' \in L(T)$$

et

$$\sum_{t \in L(T)} \mathfrak{p}_t = 1.$$

La densité q_{n+1} *est alors prise égale à*

$$\sum_{t \in L(T)} \mathfrak{p}_t \, \mathcal{U}_{R_t},$$

avec $\mathcal{U}_R$ *la densité uniforme sur le domaine* R*.*

Nous justifions dans les paragraphes suivants les étapes de l'algorithme.

Étape 1 Un critère d'arrêt à la subdivision du domaine est choisi. Il y a plusieurs possibilités, comme limiter la profondeur maximale de l'arbre ou le nombre de divisions selon une dimension considérée. Nous décidons ici que, si la probabilité $\mathfrak{p}_t$ associée au nœud t est en dessous d'une valeur seuil $\mathfrak{p}_{\min}$, alors il n'est pas nécessaire de subdiviser à nouveau la région associée à R_t. Expérimentalement, la valeur de $\mathfrak{p}_{\min}$ est prise égale à 10^{-3}.

Étape 2 Le développement qui suit justifie les équations (4.14) et (4.15), c'est-à-dire le choix respectivement des valeurs $U_{t,l}$ et de la direction de subdivision $l^\star$. Nous procédons, pour chaque nœud t de T, de façon gloutonne en cherchant l'estimateur q_{n+1} de p_n qui minimise la divergence de Kullback-Leibler sur le domaine R_t

$$\begin{aligned} D(p_n, q_{n+1}) &= \int_{\mathbb{X}} p_n(x)\ \ln(p_n(x)/q_{n+1}(x))\mathrm{d}x & (4.18)\\ &= -\int_{R_t} p_n(x)\ \ln(q_{n+1}(x))\mathrm{d}x + C, & (4.19)\end{aligned}$$

où C est une constante indépendante de la restriction de q_{n+1} sur R_t. Nous cherchons, sur cet ensemble R_t, un estimateur q_{n+1} de la forme

$$(U_{t,l}\mathbb{1}_{R_{t_{1,l}}} + (1 - U_{t,l})\mathbb{1}_{R_{t_{2,l}}})\mathfrak{p}_t. \tag{4.20}$$

Nous disposons d'un ensemble $\mathfrak{X}_n$ dont les particules sont supposées réparties selon la densité p_n. Dans l'expression de la divergence de Kullback-Leibler, nous remplaçons donc p_n par l'approximation

$$\sum_{i,j} w_{n,i,j}\ \delta_{x_{n,i,j}}, \tag{4.21}$$

ce qui nous donne

$$\begin{aligned}(l^\star, U_{t,l^\star}) &= \operatorname{argmax} \sum_{x_{n,i,j}\in R_t} w_{n,i,j} \ln(q_{n+1}(x_{n,i,j}))\\ &= \operatorname{argmax} \sum_{x_{n,i,j}\in R_{t_{1,l}}} w_{n,i,j} \ln(U_{t,l}) + \sum_{x_{n,i,j}\in R_{t_{2,l}}} w_{n,i,j} \ln(1 - U_{t,l}).\end{aligned}$$

Un simple calcul de dérivée pour cette expression nous permet d'établir que

$$U_{t,l} = \frac{\sum_{x_{n,i,j}\in R_{t_{1,l}}} w_{n,i,j}}{\sum_{x_{n,i,j}\in R_t} w_{n,i,j}},$$

ce qui explique l'équation (4.14). Nous en déduisons que

$$\begin{aligned} l^\star &= \operatorname*{argmax}_l\ U_{t,l}\ \ln(U_{t,l}) + (1 - U_{t,l})\ \ln(1 - U_{t,l})\\ &= \operatorname*{argmin}_l\{\min\{U_l\ ;\ 1 - U_l\}\}. & (4.22)\end{aligned}$$

L'équation (4.22) se justifie par le fait que nous reconnaissons un problème de minimisation d'entropie comme dans le cas d'algorithmes de type CART (voir, par exemple, Izenman, 2008). Nous retrouvons exactement l'équation (4.15).

Désormais, intéressons nous à la seconde partie de l'étape 2, concernant les valeurs $\mathfrak{p}_{t_{1,l^\star}}$ et $\mathfrak{p}_{t_{2,l^\star}}$. La définition de $U_{l^\star}$, et le fait que $\mathfrak{p}_t$ est supposé représenter la répartition empirique des éléments de $\mathfrak{X}_n$ sur R_t, impliquent de par les équations (4.16) et (4.23), que les valeurs $\mathfrak{p}_{t_{1,l^\star}}$ et $\mathfrak{p}_{t_{2,l^\star}} = \mathfrak{p}_t - \mathfrak{p}_{t_{1,l^\star}}$ représentent, elles aussi, la répartition empirique des éléments de $\mathfrak{X}_n$ sur leurs domaines respectifs. En pratique, nous utilisons une définition différente de l'équation (4.16). La valeur $\mathfrak{p}_{t_{1,l^\star}}$ que nous choisissons est la suivante

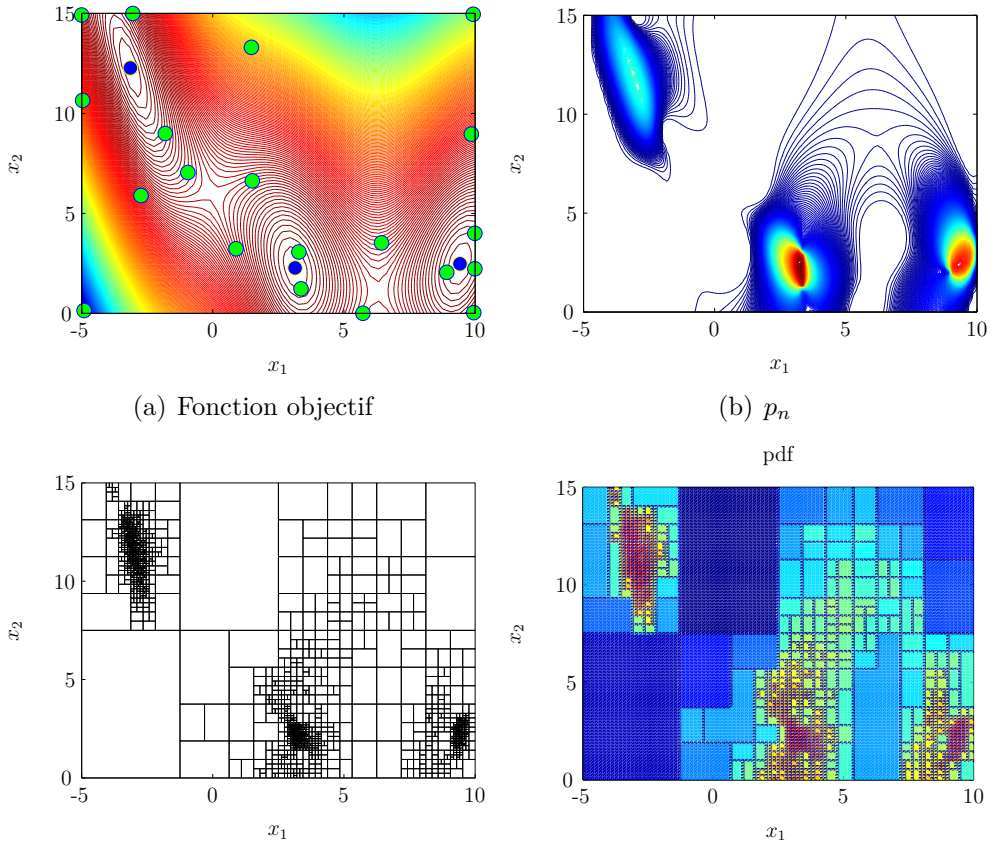
$$\mathfrak{p}_{t_{1,l^\star}} = \max\{\mathfrak{p}_{\min}/2 \; ; \; \min\{\mathfrak{p}_t - \mathfrak{p}_{\min}/2 \; ; \; \mathfrak{p}_t \, U_{l^\star}\}\}. \tag{4.23}$$

Cette valeur ne diffère de celle calculée par (4.16) que si la valeur de $U_{l^\star}$ est très proche de 1 ou de 0. Ceci revient à une situation où l'un des deux fils, nouvellement créés, de t est quasiment « déserté » (éventuellement complètement « déserté ») par les particules de $\mathfrak{X}_n$. Cette adaptation permet d'éviter la création des zones du domaine $\mathbb{X}$ qui ne pourraient être échantillonnées par q_{n+1}.

Étape finale Les propriétés énoncées à cette étape découlent des précédentes, en particulier de la construction des valeurs $\mathfrak{p}_t$ et de la façon dont les domaines sont subdivisés.

Illustrations Afin d'illustrer l'algorithme développé dans cette section, la figure 4.4 présente une partition de l'espace construite à partir de 10000 points répartis selon la densité p_n (ici, la probabilité d'amélioration normalisée) de cet estimateur, toujours pour la fonction de Branin avec un plan d'expérience de vingt points distribués. Cette partition pondérée définit de façon unique la densité instrumentale q_{n+1} utilisée.

Une seconde illustration, dans une situation d'optimisation faisant à nouveau intervenir 10000 points candidats répartis selon p_n, est également donnée. L'objectif des figures 4.5 et 4.6 est de montrer l'évolution de la partition en fonction des points d'évaluations. Concernant ces figures, l'objectif n'est pas



(a) Fonction objectif

(b) p_n

(c) Partition associée q_{n+1}

(d) Partition pondérée associée à q_{n+1}

Figure 4.4 – Illustration de l'utilisation d'un estimateur à base d'arbre, à partir de 10000 points, afin de construire une densité instrumentale q_{n+1} (avec $n = 20$) estimant la densité p_n (probabilité d'amélioration normalisée). La fonction objectif est la fonction de Branin. Les points verts correspondent aux évaluations, les points bleus aux maximiseurs.

de s'intéresser aux paramètres de la stratégie d'optimisation (c'est lors des applications suivantes qu'ils seront discutés) mais à la génération des points candidats via la densité instrumentale q_{n+1} et la partition du domaine $\mathbb{X}$. Nous remarquons que très rapidement le découpage de l'espace se fait particulièrement fin au voisinage des trois maxima de la fonction de Branin (cela s'observe déjà pour 10 itérations, et est flagrant pour 30). Cette constation indique que la précision disponible, concernant le prochain point d'évaluation choisi, est plus grande dans les zones les plus intéressantes.

4.4 Complexité algorithmique

D'un point de vue algorithmique, si nous omettons l'étape de construction des densités instrumentales q_n, les deux étapes les plus coûteuses correspondent au calcul des valeurs $g_n(x_{n,i,j} \mid \theta_{n,i})$ (étape de démarginalisation) et au calcul du critère EI pour l'ensemble des points candidats $x_{n,i,j}$. Dans les deux cas, la majeure partie du temps est utilisée afin de calculer le prédicteur, en particulier lorsque la fonction de covariance de Matérn intervient.

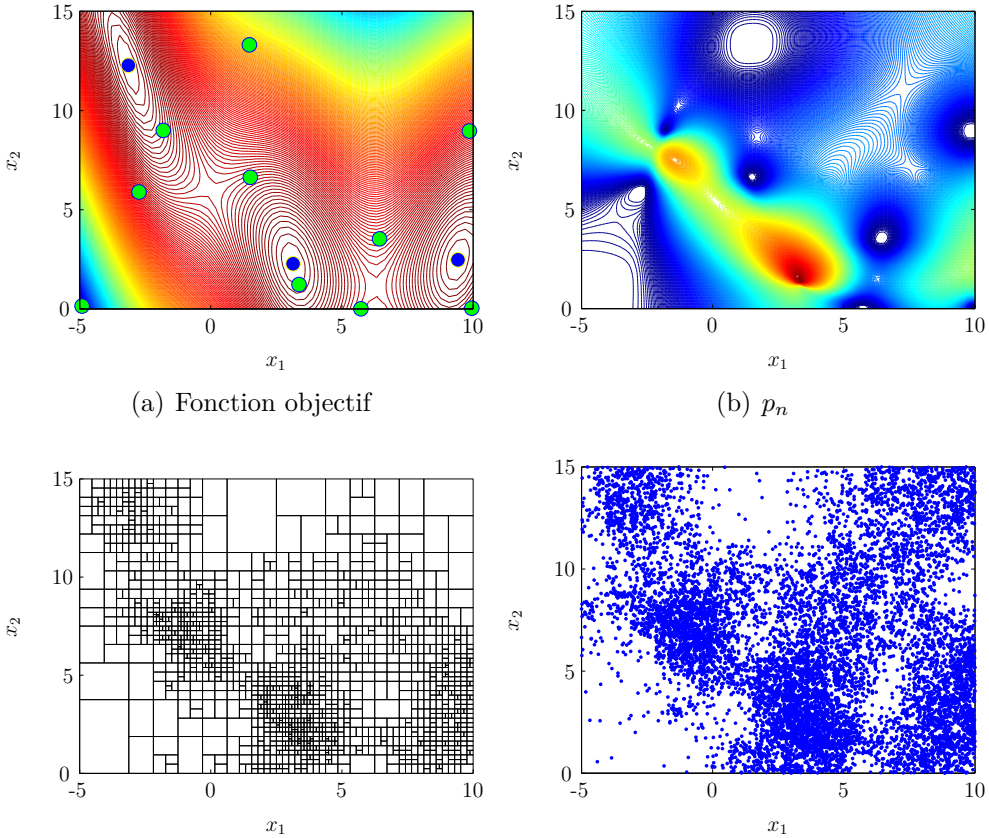
Au prix d'une perte de précision potentielle quelques adaptations de l'algorithme permettent d'économiser significativement du temps de calcul. Concernant l'étape de démarginalisation, le calcul, pour $i \in I$ fixé, de la constante $c_{n,i}$, à savoir une estimation de la valeur

$$\int_{x \in \mathbb{X}} g_n(x \mid \theta_{n,i}) \, \mathrm{d}x,$$

nécessite N_γ appels à la fonction $g_n(. \mid \theta_{n,i})$ (autrement dit, en chacun des points candidats $x_{n,i,j}$). Pour rappel

$$c_{n,i} = \frac{1}{N_\gamma} \sum_{i'=1}^{I} \sum_{j'=1}^{J} \frac{g_n(x_{n,i',j'} | \theta_{n,i})}{q_n(x_{n,i',j'})}. \tag{4.24}$$

Remarquons que nous calculons les valeurs $g(x_{n,i,j} \mid \theta_{n,i})$ pour tous $j \in J$, afin d'obtenir les valeurs des poids $w'_{n,i,j}$. Calculer $c_{n,i}$ à partir des J valeurs $g(x_{n,i,j} \mid \theta_{n,i})$, dont nous n'avons de toute façon pas la possibilité de faire l'économie, représente une alternative raisonnable. En effet, pour i fixé, les $x_{n,i,j}$ sont identiquement et indépendamment générés par q_n, ce qui implique



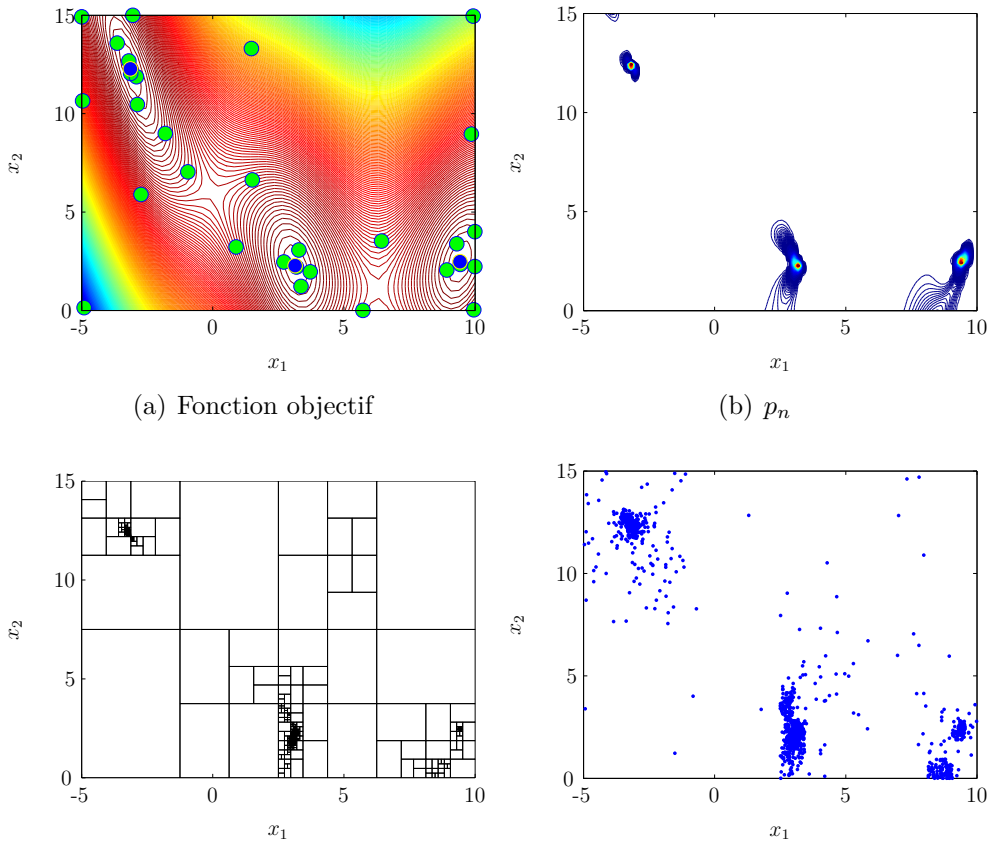
(a) Fonction objectif

(b) p_n

(c) Partition associée q_{n+1}

(d) Points générés à partir de q_{n+1}

Figure 4.5 – Utilisation d'un estimateur à base d'arbre, à partir de $N_\gamma = 10000$ points, et $n = 10$ évaluations, afin de construire N_γ nouveaux points candidats répartis selon la densité p_n (probabilité d'amélioration normalisée). La fonction objectif est la fonction de Branin. Les points verts correspondent aux évaluations, les points bleus aux maximiseurs.



(a) Fonction objectif

(b) p_n

(c) Partition associée q_{n+1}

(d) Points générés à partir de q_{n+1}

Figure 4.6 – Utilisation d'un estimateur à base d'arbre, à partir de $N_\gamma = 10000$ points, et $n = 30$ évaluations, afin de construire N_γ nouveaux points candidats répartis selon la densité p_n (probabilité d'amélioration normalisée). La fonction objectif est la fonction de Branin. Les points verts correspondent aux évaluations, les points bleus aux maximiseurs.

$$c_{n,i} = \frac{1}{J}\sum_{j'=1}^{J}\frac{g_n(x_{n,i,j'}|\theta_{n,i})}{q_n(x_{n,i,j'})} \tag{4.25}$$

$$\rightarrow_J \int_{x\in\mathbb{X}} g_n(x \mid \theta_{n,i})\,\mathrm{d}x. \tag{4.26}$$

Nous remplaçons une somme à IJ termes par une à seulement J termes. En pratique, nous avons utilisé cette adaptation pour toutes les simulations numériques présentées dans le manuscrit.

Nous proposons également une approche moins coûteuse pour le calcul du critère EI sur les points candidats. L'expression telle que présentée à la section 4.3 implique le calcul de IN_γ valeurs d'EI afin de déterminer le nouveau point d'échantillonnage X_{n+1}. Afin de diminuer la complexité, nous proposons une heuristique tendant à diminuer le nombre de calculs d'EI effectifs. En effet, les valeurs d'EI sont calculées dans un ordre bien particulier, ce qui offre de l'information dont nous faisons usage pour discriminer les points candidats les moins « prometteurs » des autres, et les éliminer au fur et à mesure des calculs d'EI suivants. En pratique, nous considérons une permutation s de l'ensemble des entiers allant de 1 à I, telle que les $\theta_{n,s(i)}$ soient rangés par ordre des poids $w_{n,s(i)}$ décroissant. Pour un point candidat $x_{n,i,j}$, le critère EI vaut $\sum_{i'=1}^{I} w_{n,i'}\,\rho_n(x_{n,i,j};\theta_{n,i'})$, et c'est donc le terme associé à $\rho_n(x_{n,i,j};\theta_{n,s(1)})$ qui est dominant dans la somme (puisque, par construction, le plus fortement pondéré). Dans un premier temps, nous calculons pour l'ensemble des points candidats ces seules valeurs $\rho_n(.;\theta_{n,s(1)})$. Nous faisons alors le choix de considérer que, parmi ces N_γ valeurs calculées, les $\lfloor N_\gamma/2 \rfloor$ les plus faibles caractérisent des points candidats n'ayant qu'une très faible probabilité d'être des maximiseurs de l'EI. En d'autres termes, déterminer de nouvelles valeurs d'EI en ces points est superflu. Pour les $\lceil N_\gamma/2 \rceil$ points candidats restants, nous calculons alors les valeurs $\rho_n(.;\theta_{n,s(2)})$, et nous éliminons à nouveau la moitié en lesquelles les valeurs $w_{n,s(1)}\rho_n(.;\theta_{n,s(1)}) + w_{n,s(2)}\rho_n(.;\theta_{n,s(2)})$ sont les plus faibles. Sans entrer plus avant dans les détails, la même stratégie est itérée pour tous $1 \leq i \leq I$, menant ainsi à force économie de calculs d'EI, tout en faisant émerger un « maximiseur » X_{n+1}, que nous pouvons raisonnablement espérer

proche du vrai. Il est à noter que ces approximations, se faisant potentiellement au prix d'une perte de performance, ne sont donc pas utilisées au chapitre 5, où des comparaisons sont effectuées avec d'autres algorithmes. Afin de disposer d'un résultat quantitatif sur le temps susceptible d'être gagné grâce à cette adaptation, nous avons effectué dans les deux cas, c'est-à-dire avec et sans, une optimisation de la fonction de Branin avec un budget d'évaluations de 100 (dont 4 d'entre elles constituent le plan d'expérience initial). Le résultat est que cette adaptation permet une optimisation complète en 6.2 min au lieu de 23.8 min, soit un gain d'environ 74%, ce qui est significatif. Les performances, quant à elles, sont dégradées dans un rapport de 2.3. Les paramètres et les *a priori* utilisés pour cette optimisation sont exactement les mêmes que ceux de la section 4.5.

4.5 Illustration et comparaisons

Nous illustrons ici le fonctionnement de l'algorithme à nouveau sur la fonction de Branin. Les *a priori* utilisés sont les mêmes que ceux de la section 3.4.3. Pour l'ensemble de cette section, nous prenons $I = 100$ particules pour les θ à chacune de laquelle nous associons $J = 100$ valeurs de points candidats x. La densité p_n choisie correspond à la probabilité d'amélioration normalisée. Nous avons déjà observé, à la figure 4.6, qu'après 30 évaluations la densité des points candidats est particulièrement importante au voisinage des maxima, ce qui permet une maximisation efficace du critère EI (peu de points, mais répartis sur les zones particulièrement prometteuses). La figure 4.7, quant à elle, met en évidence la répartition des particules selon la densité *a posteriori* π_n pour le même nombre d'évaluations. Ceci montre que l'approche SMC utilisée ici est efficace aussi bien sur le domaine $\mathbb{X}$ que sur l'espace Θ.

Tests. Au chapitre précédent, à la section 3.4.2, des comparaisons entre les performances de plusieurs algorithmes ont été effectuées sur des fonctions tests. Nous reprenons ici, pour la fonction de Branin, les résultats obtenus à la section 3.4.2 avec notre algorithme complètement bayésien pour deux tailles de plan d'expérience intial ($n_0 = 2d$ et $n_0 = 10d$). Pour rappel, le critère EI était

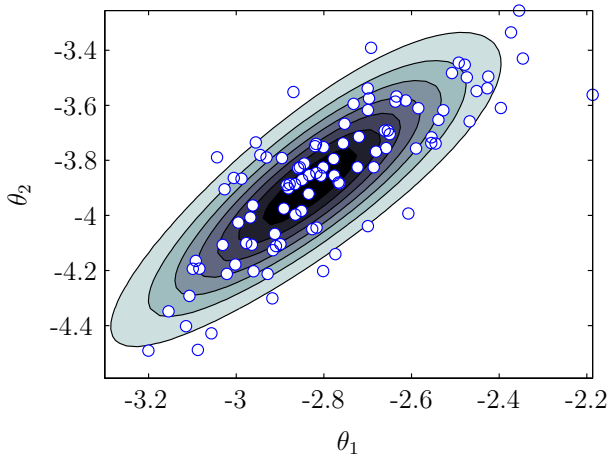
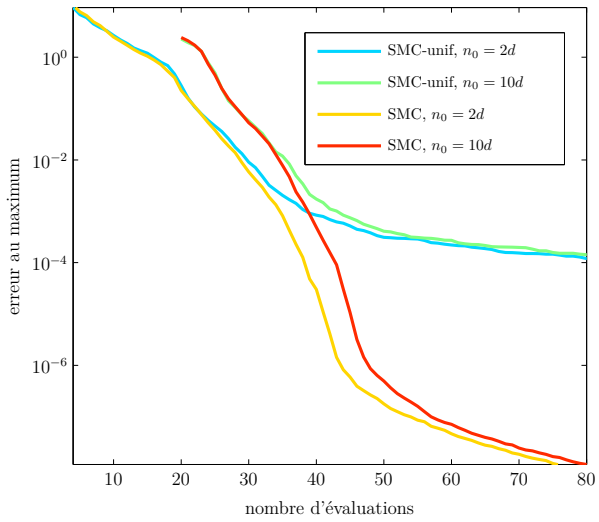


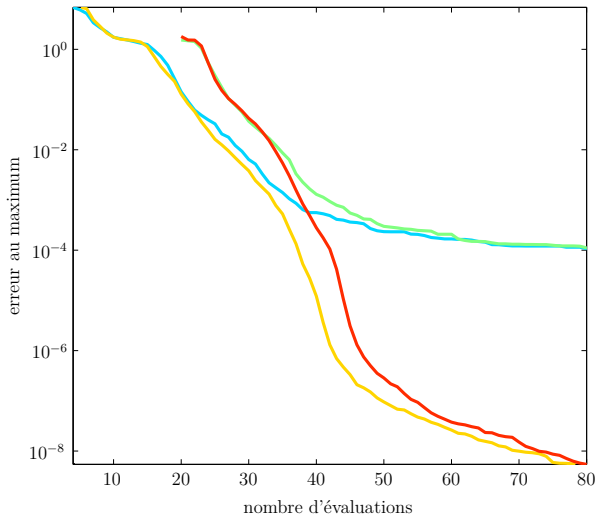
Figure 4.7 – Densité *a posteriori* $\pi_n(\theta)$ et répartition des $I = 100$ particules en θ (pour $n = 30$ évaluations).

maximisé à partir d'un ensemble de points candidats échantillonnés de façon uniforme sur le domaine $\mathbb{X}$. Nous pouvons désormais comparer ces résultats avec les performances de l'algorithme, introduit à la section 4.3, reposant sur une approche SMC sur l'espace $\Theta \times \mathbb{X}$ (et non plus seulement sur Θ). Le mode opératoire, ainsi que le choix des paramètres et des *a priori*, sont les mêmes qu'à la section 3.4.3. L'objectif est donc d'illustrer le gain apporté par l'utilisation de méthodes SMC pour générer, à chaque étape, les points candidats.

Résultats. Les résultats obtenus, visibles sur la figure 4.8, sont probants. En effet, pour les deux tailles de plan d'expérience intial n_0 considérées, les performances obtenues grâce à un échantillonnage des points candidats x selon p_n sont bien meilleures que dans le cas d'un échantillonnage uniforme (dans un rapport proche de 10^4), aussi bien en ce qui concerne la moyenne de l'erreur que la médiane. Considérer un algorithme complètement bayésien construit à partir de méthodes SMC sur $\Theta \times \mathbb{X}$ permet ainsi non seulement d'être plus robuste concernant la taille n_0 du plan d'expérience (ce qui avait été mis en évidence au chapitre 3) mais également d'améliorer significativement la convergence vers l'optimum.



(a) Moyenne



(b) Médiane

Figure 4.8 – Comparaison de l'erreur d'estimation du maximum (moyenne et médiane à partir de 200 simulations) pour une approche complètement bayésienne avec échantillonnage uniforme sur $\mathbb{X}$ (noté SMC-unif) et avec échantillonnage selon la probabilité d'amélioration normalisée p_n (noté SMC). Deux tailles n_0 de plan d'expérience initial sont considérées, à savoir $2d$ et $10d$. La fonction objectif utilisée est Branin.

4.6 Résumé du chapitre

La principale contribution de ce travail de thèse a été présentée dans ce chapitre. Il s'agit ainsi d'un algorithme, utilisant une méthode SMC, efficace à la fois pour calculer une estimation du critère EI complètement bayésien, ainsi que pour maximiser ce critère. En cela, il répond aux deux difficultés présentées dans l'introduction, qui avaient servi de motivation à ce travail. Concernant celle relative à une maximisation efficace du critère d'optimisation, l'idée est de l'obtenir en ne calculant le critère qu'en un nombre de points candidats du domaine $\mathbb{X}$ (relativement) faible. Il est ainsi nécessaire de bien choisir la façon dont les points candidats sont répartis dans le domaine, les zones où le critère est particulièrement élevé devant logiquement être celles où leur concentration est la plus importante. D'un point de vue expérimental, les résultats montrent qu'un véritable gain résulte de cette approche en comparaison à un échantillonnage uniforme du domaine. En effet, la forte concentration des points au voisinage des maxima globaux, au fur et à mesure que le nombre d'évaluations augmente, permet de maximiser le critère de façon plus précise et ainsi de s'approcher de plus en plus près des maxima recherchés. À notre connaissance, l'algorithme présenté ici l'est pour la première fois. Si l'utilisation de méthodes SMC pour calculer un critère complètement bayésien avait déjà émergé dans la littérature au cours des dernières années, considérer une approche SMC conjointe afin de maximiser efficacement ce critère relève manifestement de l'inédit. La suite du manuscrit est consacrée à la mise à l'épreuve de l'algorithme dans des contextes de *benchmarking* et d'applications industrielles.

Chapitre 5

Applications

Les chapitres précédents nous ont permis d'introduire le contexte de l'optimisation globale à l'aide de méthodes bayésiennes, ainsi que de motiver la construction d'un nouvel algorithme d'optimisation complètement bayésien. Dans ce qui suit, nous testons ses performances et son comportement sur différents exemples d'application. À la section 5.1 nous donnons explicitement la configuration ainsi que les paramètres utilisés pour notre algorithme dans ce chapitre, ainsi que quelques précisions sur la façon dont les résultats sont interprétés. Des exemples d'applications, comme un problème d'identification de système ou l'optimisation du rendement d'un convertisseur de puissance sont étudiés respectivement aux sections 5.2 et 5.3. Pour finir, une comparaison avec d'autres algorithmes, non nécessairement bayésiens, est effectuée sur des fonctions tests classiques à la section 5.4.

5.1 Configuration et choix des paramètres

Tout au long de ce chapitre, les paramètres choisis concernant les différents algorithmes sont essentiellement les mêmes, et sont ainsi explicités dans cette section. Concernant notre algorithme complètement bayésien, ses paramètres (moyenne, fonction de covariance) ainsi que les *a priori* choisis sont les mêmes qu'à la section 3.4.3. Pour rappel, il s'agit d'un *a priori* de Jeffreys sur la variance σ^2 et le paramètre θ correspond de façon générique à un vecteur du

type $(\ln \alpha_1, \ln \alpha_2, \ldots, \ln \alpha_d)$ où, pour $1 \leq k \leq d$, α_k est l'inverse de la portée associée à la k-ième dimension. Pour $\mathbb{X} = \prod_{i=1}^{d} [u_1^i; u_2^i]$, l'*a priori* sur les α_k est choisi log-normal de la forme $\ln \mathcal{N}(\mu_0^k, 0.5^2)$, où $\mu_0^k = -\ln(\sqrt{d}(u_2^k - u_1^k)/3)$. Nous considérons $I = 100$ valeurs différentes de θ et, nous associons à chacune d'elle, $J = 100$ valeurs de points candidats x. Nous disposons donc d'un nombre $IJ = 10^4$ de particules du type (θ, x). Nous choisissons la probabilité d'amélioration, normalisée, comme loi cible pour les x, que nous notons p_n pour n évaluations. Cette configuration reste identique tout le le long de ce chapitre. En ce qui concerne la taille n_0 du plan d'expérience initial, elle est prise égale à deux ou dix fois la dimension ($n_0 = 2d$ ou $n_0 = 10d$). La valeur n_0 choisie est toujours précisée dans ce qui suit. Il est à noter que pour l'ensemble des figures de ce chapitre, ainsi que pour les analyses associées, notre algorithme complètemement bayésien est noté SMC-EI afin de le distinguer d'EGO.

L'algorithme EGO utilisé dans les sections 5.2 et 5.3 l'est dans les mêmes conditions, et avec la même mise en œuvre qu'à la section 3.4.3. L'estimation des paramètres par maximum de vraisemblance restreinte est effectuée, à chaque nouvelle itération, grâce à la fonction matlab *fmincon*, avec comme point de départ la valeur de paramètres estimée utilisée lors de l'itération précédente. À l'initialisation, c'est la valeur 1 qui est utilisée pour la variance σ^2 et, pour les portées, la valeur θ_0 introduite plus haut. En ce qui concerne la section 5.4, une mise en œuvre d'EGO différente est considérée puisque nous utilisons le *package* R DiceOptim (Ginsbourger et Roustant, 2011). Comme pour notre algorithme complètement bayésien, nous considérons deux tailles n_0 différentes ($2d$ ou $10d$). La valeur n_0 retenue pour une simulation spécifique est toujours précisée explicitement.

Comme expliqué ci-dessus, le paramètre θ correspond à un vecteur du type $(\ln \alpha_1, \ln \alpha_2, \ldots, \ln \alpha_d)$. L'espace Θ est ainsi égal à $]-\infty; +\infty[^d$, et une particule θ_i peut donc s'écrire sous la forme $(\ln(\alpha_{i,1}), \ln(\alpha_{i,2}), \ldots, \ln(\alpha_{i,d}))$ avec, pour $1 \leq k \leq d$, $\alpha_{i,k} \in]0; +\infty[$. Par la suite, nous notons $F_{n,k}$ la fonction de répartition de la marginale de π_n selon la k-ième variable. La marginale de π_n selon la k-ième variable permet de caractériser l'influence de cette variable sur la fonction objectif. Si de petites valeurs de α_k (donc de grandes valeurs de portées) sont

favorisées, alors peu de variations de f sont mises en évidence selon cette dimension par les n résultats d'évaluation disponibles, et inversement pour des grandes valeurs de α_k. En pratique, pour des raisons de simplicité, nous calculons les fonctions de répartition empiriques de ces marginales,

$$F_{n,k} : \quad]-\infty;+\infty[\quad \to \qquad [0;1]$$

$$\beta = \ln(\alpha) \quad \mapsto \quad \frac{\sum_{\substack{i=1 \\ \alpha_{i,k} \leq \alpha}}^{I} w_{n,i}}{\sum_{i=1}^{I} w_{n,i}}.$$

Dans la suite nous faisons parfois intervenir ces fonctions de répartition $F_{n,k}$ afin d'étudier l'influence des différentes variables sur les variations de la fonction objectif.

5.2 Exemple 1 : problème d'identification de système dynamique

Nous considérons l'exemple du problème d'identification de système présenté par Villemonteix et al. (2007), et dont nous reprenons ici l'essentiel du formalisme. Il s'agit de considérer un modèle à deux états $q(x,t) = (q_1(x,t), q_2(x,t))^T$, avec $x = (x_1, x_2, x_3) \in [0,1]^3$, caractérisant la quantité de matière dans deux compartiments, et dont l'évolution est décrite par le système dynamique

$$\begin{cases} \frac{\partial q_1}{\partial t} = & -(x_1 + x_3)\ q_1 + x_2\ q_2 \\ \frac{\partial q_2}{\partial t} = & x_1\ q_1 - x_2\ q_2. \end{cases} \tag{5.1}$$

L'état initial $t = 0$ pour q est $(1,0)^T$, ce qui correspond à l'ajout d'une quantité de matière de 1 dans le premier compartiment. Nous notons $y(t_i) = q_2(x^\star, t_i)$ la quantité de matière mesurée, sans bruit d'observation, dans le compartiment 2 aux instants $t_i, i = 1, \ldots, 15$ pour $x^\star = (0.6, 0.15, 0.35)^T$. La fonction objectif que nous considérons alors est définie sur $[0,1]^3$ par

$$f(x) = \sum_{i=1}^{15} (q_2(x,t_i) - y(t_i))^2. \tag{5.2}$$

Cette fonction représente une erreur d'adéquation que nous chercherons donc à minimiser.

Comme expliqué dans l'article de Villemonteix et al. (2007), cet exemple n'a que l'apparence de la trivialité. En effet, la zone de l'espace où f est faible est étendue. De plus, la symétrie concernant les variables x_2 et x_3 (Walter et Pronzato, 1997) entraîne un défaut d'unicité quant à l'identification des paramètres du modèle. Deux minimiseurs globaux $x_\star^{(1)}$ et $x_\star^{(2)}$ apparaissent donc, valant respectivement $(0.6, 0.15, 0.35)$ et $(0.6, 0.35, 0.15)$. L'ensemble de ces caractéristiques sont illustrées par la figure 5.1 représentant la fonction f sur le plan $x_1 = 0.6$, plan contenant les deux maximiseurs globaux. L'évolution des grandeurs q_1 et q_2 en fonction du temps est représentée quant à elle sur la figure 5.2.

Nous nous intéressons aux fonctions de répartition empiriques $F_{n,1}$, $F_{n,2}$ et $F_{n,3}$ des marginales de π_n afin de déterminer de façon qualitative l'influence des trois variables du problème (voir explications à la section 5.1). La figure 5.3 permet de les visualiser à plusieurs stades de la procédure d'optimisation. Nous pouvons remarquer que la première variable semble la moins influente du problème. En effet, $F_{n,1}$ connaît sa plus forte variation pour de faibles valeurs de α_1.

Concernant la répartition des particules en θ, nous considérons essentiellement leur projeté orthogonal sur le sous-espace, de dimension deux, relatif aux variables 2 et 3. Nous avons en effet remarqué précédemment que ce sont les deux variables les plus influentes du problème. L'objectif de la figure 5.4 est une fois de plus de comparer la répartition de ces particules avec la densité *a posteriori* π_n, où plutôt la coupe de cette loi pour une valeur spécifique de x_1. Pour des raisons pratiques, nous choisissons comme valeur de coupe la moyenne, pour cette variable, de l'ensemble des particules en θ disponibles. La répartition est bonne pour 16 et 56 évaluations. Pour 126 évaluations, la répartition est sensiblement moins bonne, certaines particules étant relativement loin de la zone où π_n a ses valeurs les plus élevées. Une étude des marginales de π_n en deux dimensions pourrait donner éventuellement plus d'information sur la validité de la répartition des particules pour ce nombre d'évaluations en

particulier. Nous avons pas eu le temps de mener cette étude des marginales en deux dimensions.

Des comparaisons concernant la moyenne et la médiane de l'erreur au maximum sont effectuées, et visibles sur la figure 5.5. Les algorithmes utilisés sont notre algorithme complètement bayésien, noté SMC-EI, ainsi que EGO, configurés tels que décrit à la section 5.1. Pour ces deux algorithmes bayésiens, les résultats sont calculés à partir de 100 expériences. À l'instar des simulations présentées à la section 3.4.2, nous considérons également deux autres algorithmes, à savoir DIRECT (déterministe) et une stratégie d'échantillonnage uniforme (dont les résultats sont calculés à partir de 1000 expériences). Les résultats sont cohérents avec ce qui a été présenté à cette section 3.4.2. Les performances de DIRECT sont les meilleures lors des toutes premières itérations, mais l'optimisation devient rapidement beaucoup moins efficace lors des itérations suivantes. Le problème de robustesse de EGO est à nouveau mis en évidence. En effet, considérer un plan d'expérience initial de taille $n_0 = 10d$ donne de bien meilleurs résultats qu'avec $n_0 = 2d$. Dans le cas de cette petite taille n_0, l'écart entre la moyenne et la médiane met en évidence ce défaut de robustesse. Ce problème ne se rencontre pas avec SMC-EI, les performances étant bonnes pour les deux valeurs de n_0. À la fin du budget d'évaluation les meilleurs algorithmes sont EGO avec $n_0 = 10d$ et les deux mises en œuvre de SMC-EI ($n_0 = 2d$ et $n_0 = 10d$). Les performances pour ces trois stratégies sont comparables jusqu'à l'évaluation 120 environ, où EGO connaît un décrochement sensible. Ce phénomène s'explique sans doute par une meilleure optimisation locale de la part de SMC-EI, liée à la technique de maximisation du critère EI présentée à la section 4.3. Pour conclure, l'algorithme que nous recommandons pour ce problème d'optimisation est SMC-EI avec $n_0 = 2d$. En effet, bien que ses performances soient comparables à celles, pour $n_0 = 10d$, de SMC-EI et EGO, il permet cependant de commencer le processus d'optimisation à partir de l'itération 7 au lieu de l'itération 30, ce qui peut être utile dans le cas où le budget d'évaluations disponible serait petit.

Les constatations sur le lien entre la robustesse comparée des algorithmes bayésiens (SMC-EI et EGO) en fonction de n_0 sont corroborées par les fi-

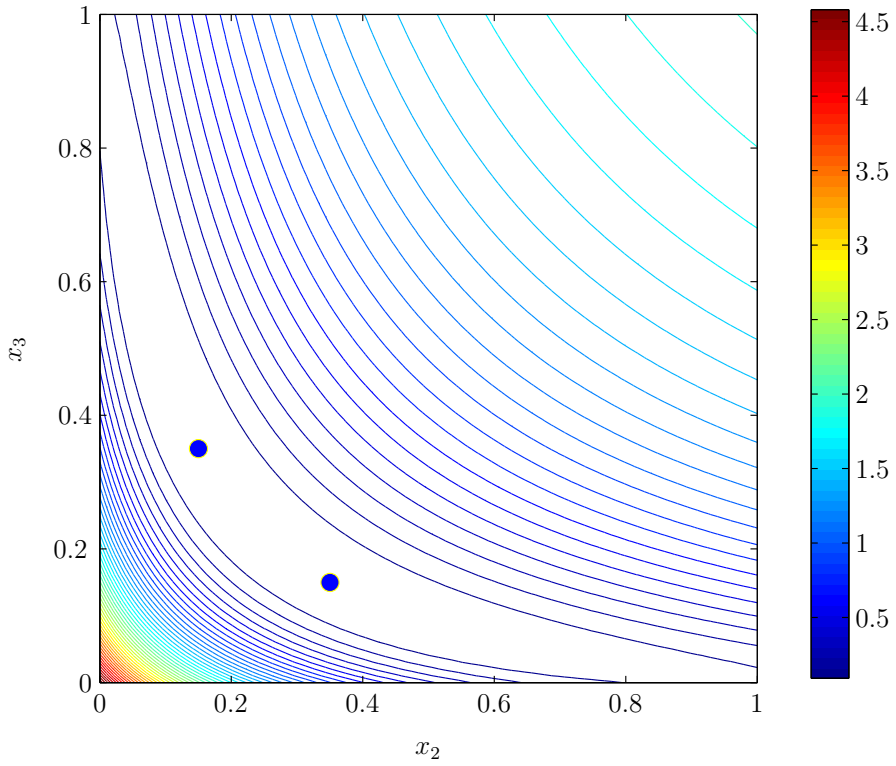


Figure 5.1 – Projection de l'erreur d'adéquation f sur le plan $x_1 = 0.6$. Les deux minima globaux sont représentés par les points bleus. Il est à remarquer que la zone la plus basse, à laquelle ces deux points appartiennent, est particulièrement large.

gures 5.6 et 5.7 représentant l'erreur au maximum. Pour $n_0 = 2d$, nous remarquons que certaines trajectoires d'EGO sont particulièrement mauvaises en comparaison de celles de SMC-EI. Pour $n_0 = 10d$, il n'y a pas de différence notable entre les deux algorithmes.

5.3 Exemple 2 : Optimisation du rendement d'un convertisseur de puissance

Dans cette section, nous appliquons l'algorithme introduit au chapitre précédent afin d'optimiser le rendement d'un convertisseur de puissance. Ce cas test nous a été fourni par Pierre Lefranc, alors enseignant-chercheur à Supélec. Une description détaillée de ce convertisseur est donnée ci-dessous. La

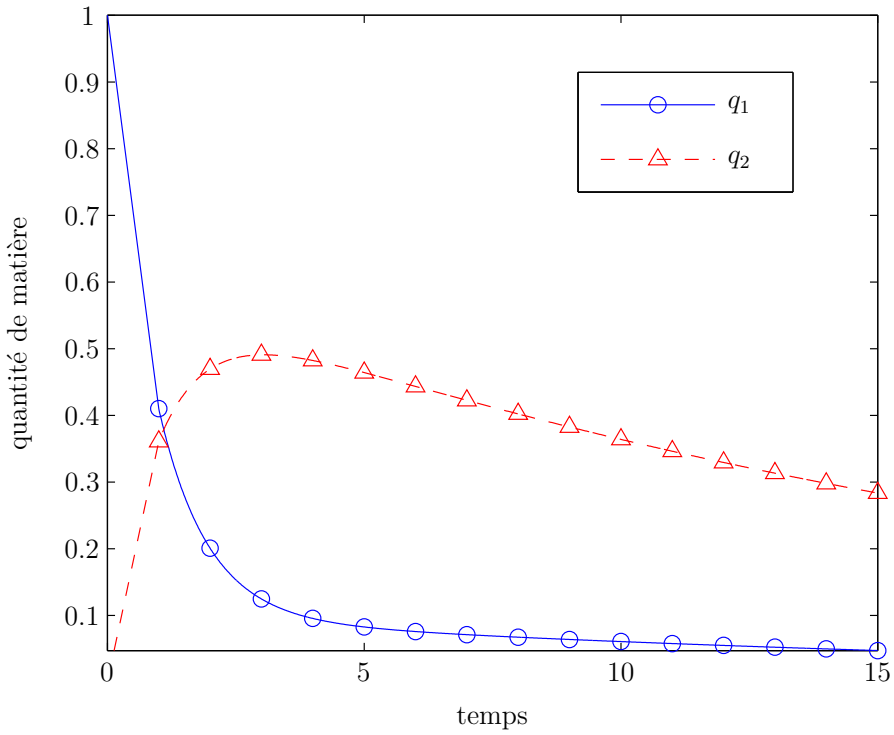
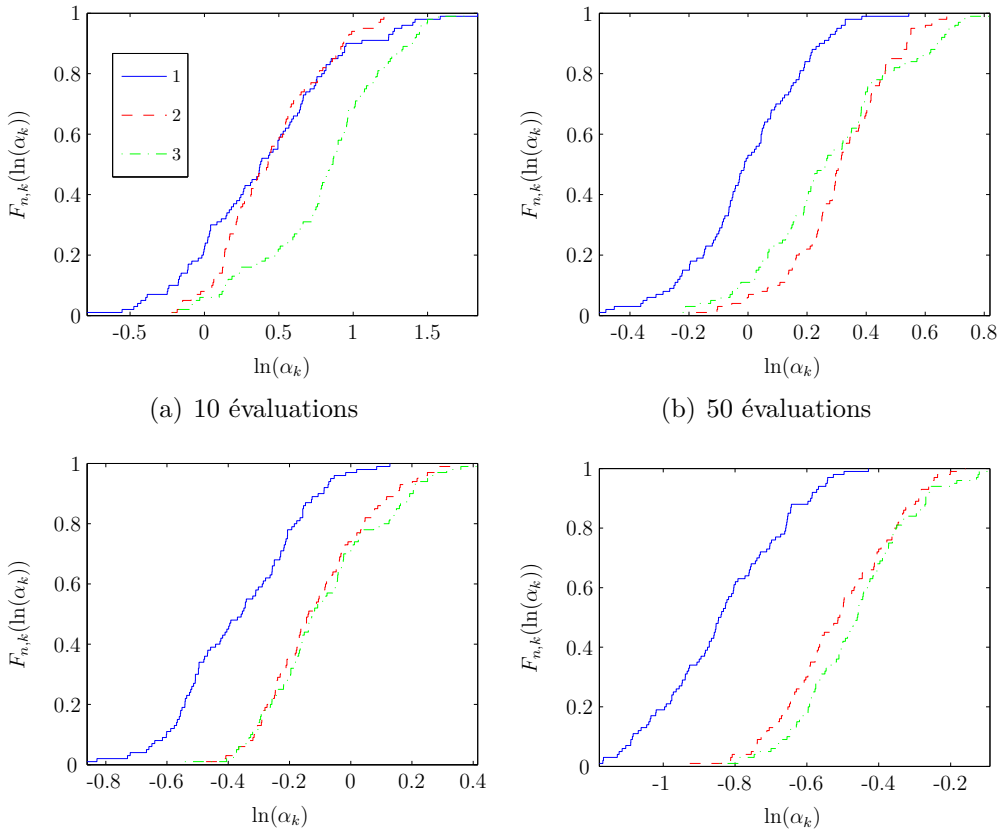


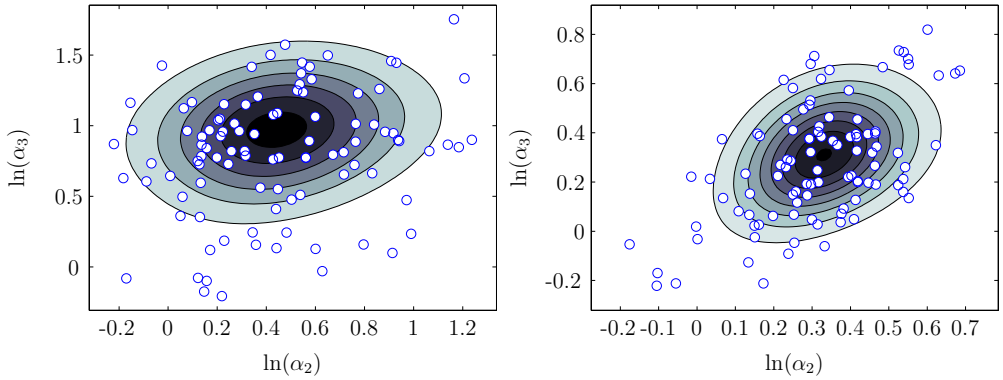
Figure 5.2 – Représentation des grandeurs q_1 et q_2 de l'instant $t = 0$ à $t = 15$. Nous avons représenté les instants d'observations sur les courbes par des ronds et des triangles.



(a) 10 évaluations (b) 50 évaluations

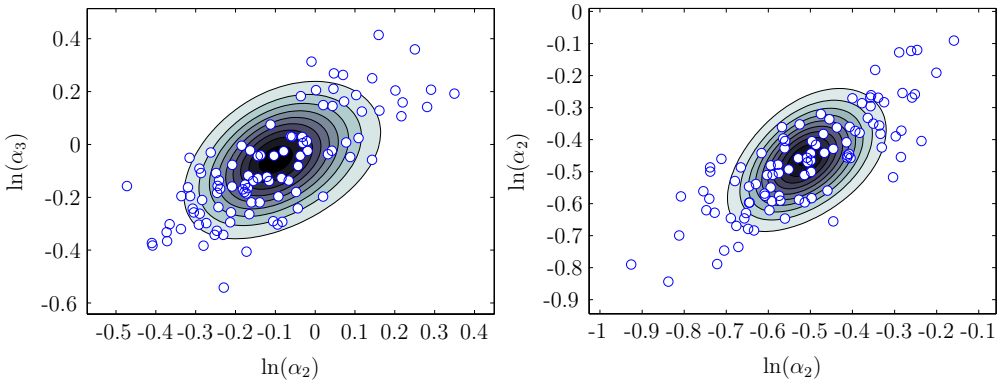
(c) 80 évaluations (d) 120 évaluations

Figure 5.3 – Fonctions de répartition empiriques des marginales de π_n



(a) 16 évaluations

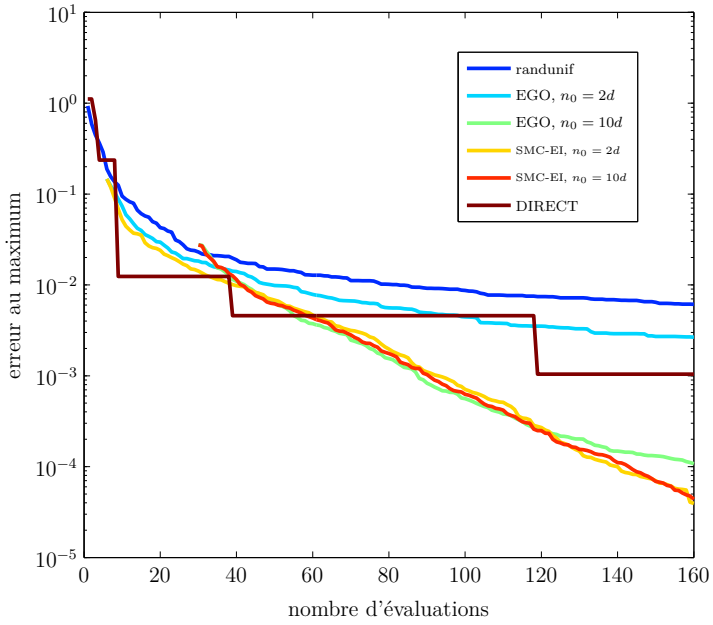
(b) 56 évaluations



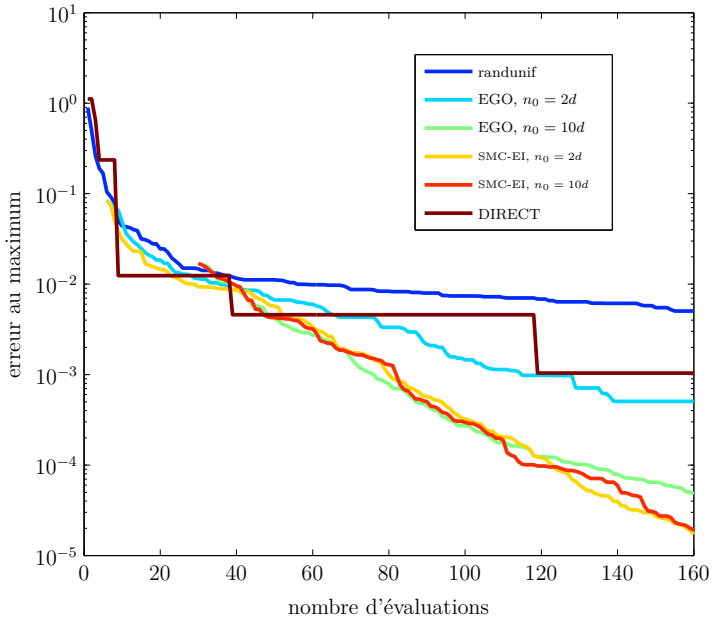
(c) 86 évaluations

(d) 126 évaluations

Figure 5.4 – Répartitions des différentes valeurs $(\alpha_2; \alpha_3)$ des particules $\theta = (\alpha_1, \alpha_2, \alpha_3)$ après projection sur l'espace associé, et représentation de coupes de la loi *a posteriori* π_n (plan de coupes correspondant aux moyennes des valeurs de α_1).

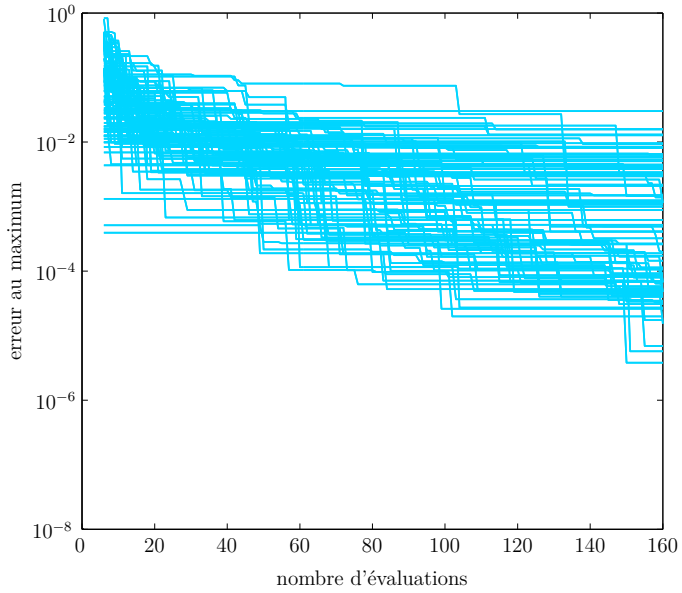


(a) Moyenne

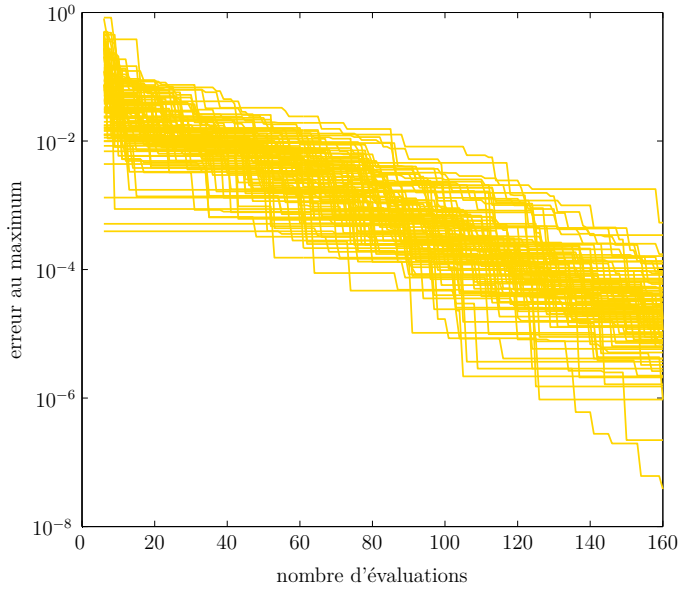


(b) Médiane

Figure 5.5 – Comparaisons entre différents algorithmes (bayésiens ou non) de l'erreur au maximum pour le problème d'identification (les moyennes et médianes sont calculées à partir de 100 processus d'optimisation pour les algorithmes bayésiens et de 1000 pour la référence uniforme).

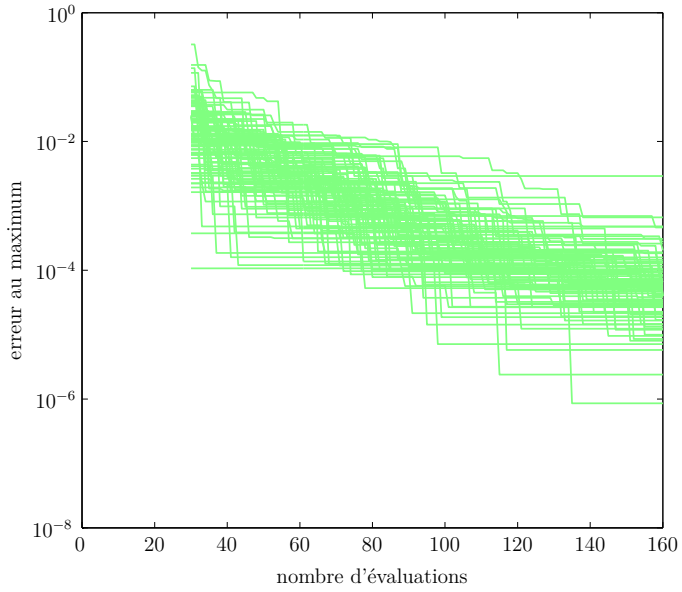


(a) EGO

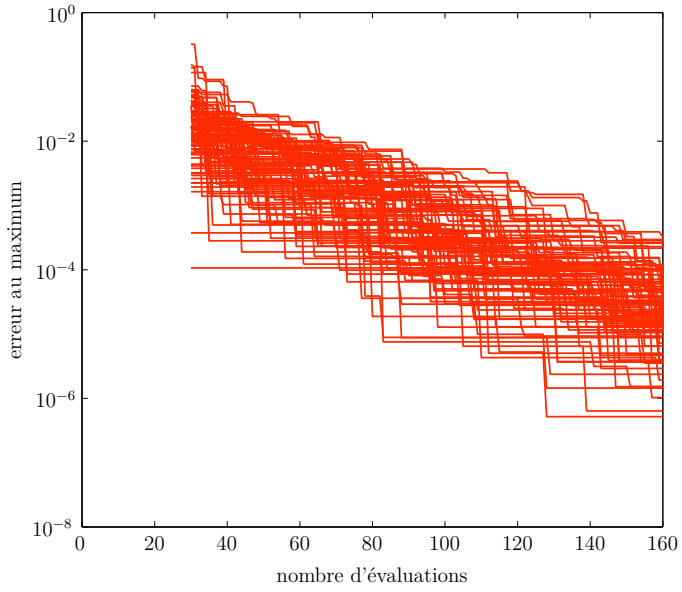


(b) SMC-EI

Figure 5.6 – Comportement comparé, sur l'ensemble des 100 trajectoires, de l'erreur au maximum en fonction du nombre d'évaluations pour les stratégies EGO (en bleu) et SMC-EI (en orange). Les comparaisons sont faites pour une taille de plan d'expérience initial n_0 valant 6 ($2d$). Le code couleur est le même que pour la figure 5.5.



(a) EGO



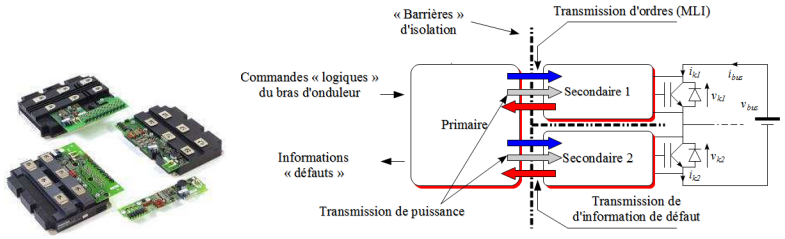
(b) SMC-EI

Figure 5.7 – Comportement comparé, sur l'ensemble des 100 trajectoires, de l'erreur au maximum en fonction du nombre d'évaluations pour les stratégies EGO (en vert) et SMC-EI (en rouge). Les comparaisons sont faites pour une taille de plan d'expérience initial n_0 valant 30 ($10d$). Le code couleur est le même que pour la figure 5.5.

construction de cet exemple d'application relève essentiellement du Département Énergie et de Pierre Lefranc en particulier, et non d'un travail effectué par nos soins au cours de la thèse, et n'est présentée ici qu'à but indicatif. Par la suite, nous indiquons et analysons les résultats d'optimisation obtenus.

5.3.1 Description du convertisseur de puissance étudié

Les composants de puissance de type IGBT (*InsulatedGateBipolar Transistor*) sont communément utilisés (Lefranc, 2005) dans la conception de convertisseurs de forte puissance (supérieure à quelques centaines de kilowatt), pour des domaines d'application tels que la traction ferroviaire, les énergies renouvelables (éoliennes), le transport d'énergie électrique via l'utilisation de liaisons HVDC (*High Voltage Direct Current*) où les composants sont utilisés. Cependant, l'interrupteur de puissance IGBT (puces silicium et *packaging* associé) doit être piloté par un système de commande rapprochée appelé *driver*. Ce dernier permet d'apporter la puissance nécessaire à la commutation du composant (en général de quelques centaines de milliwatt à quelques watt) et les ordres de commande tout en garantissant un certain niveau de fiabilité et de protection (sur-intensité, court-circuit). Sur la figure 5.8 sont montrés des exemples d'interrupteurs de puissance de type IGBT avec leur système de commande rapprochée (communément appelés drivers d'IGBT) et le synoptique d'un driver d'IGBT pour une configuration classique d'un bras d'onduleur à deux niveaux.



(a) Modules IGBT et drivers associés

(b) Synoptique d'un driver d'IGBT

Figure 5.8 – Exemples de modules IGBT et drivers, synoptique d'un driver d'IGBT pour bras d'ondulateur.

Dans cette étude, un intérêt particulier est porté à la fonction de transmission de puissance basée sur une structure de transformateur magnétique à grand entrefer afin de garantir de bonnes propriétés de tenue en tension statique et dynamique. Sur la figure 5.9 est donnée la description géométrique du transformateur magnétique. Il est constitué d'un enroulement primaire et d'un secondaire. Chacun d'eux possède un matériau magnétique en forme de 'U' séparés par un isolant d'une épaisseur e.

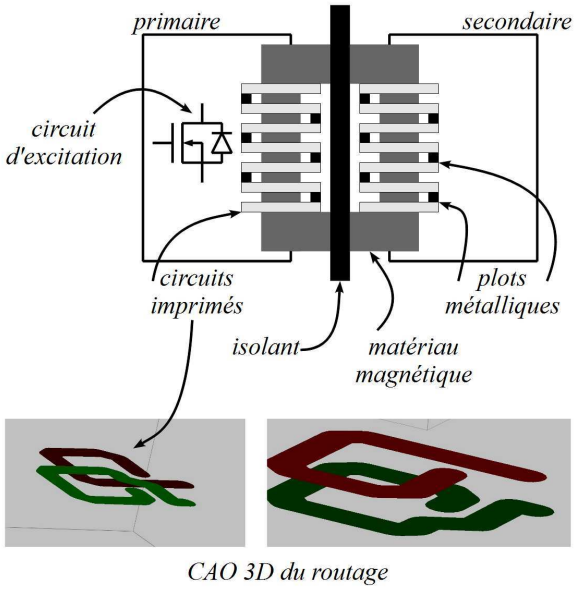


Figure 5.9 – Description géométrique du transformateur magnétique pour transmission de puissance. Mise en évidence des deux enroulements primaire et secondaire, de l'isolation galvanique et de la géométrie 3D des enroulements en cuivre.

Une première optimisation bi-objectif a permis d'optimiser la géométrie 3D du transformateur (noyaux magnétiques et bobinages) en considérant la résistance équivalente du bobinage secondaire en fonction du coefficient de couplage magnétique entre primaire et secondaire, le tout paramétré avec l'épaisseur de l'isolation entre le primaire et le secondaire. Plus de détails sont données par Lefranc et al. (2012).

La topologie du convertisseur de puissance est constituée d'un MOSFET et d'un condensateur de résonance au primaire, d'une diode et d'un condensateur

de découplage au secondaire. Les variables d'optimisation pour le convertisseur de puissance, ainsi que leur domaine de variation, sont :

- la fréquence de découpage F : $[0.5; 2.0]$ MHz,
- le condensateur de résonance C : $[5.0; 30.0]$ nF,
- la largeur de la section carrée de la ferrite x_1 : $[2.0; 5.0]$ mm,
- la largeur des pistes en cuivre x_2 : $[0.5; 2.0]$ mm,
- la distance entre l'isolant (barrière) et le bord de PCB e_1 : $[0.5; 2.0]$ mm,
- la distance entre la résistance PCB et la piste e_2 : $[0.2; 2.0]$ mm,
- la distance entre le bord PCB intérieur et la piste e_3 : $[0.2; 2.0]$ mm.

En pratique, nous effectuons une normalisation du domaine de variation des différentes variables afin qu'elles évoluent toutes entre 0 et 1, rendant ainsi plus simple l'analyse des résultats. Sur la figure 5.10 est décrite la topologie du convertisseur de puissance et est proposée une allure des formes d'ondes associées au convertisseur avec les schémas électriques équivalents correspondant aux différentes phases de fonctionnement.

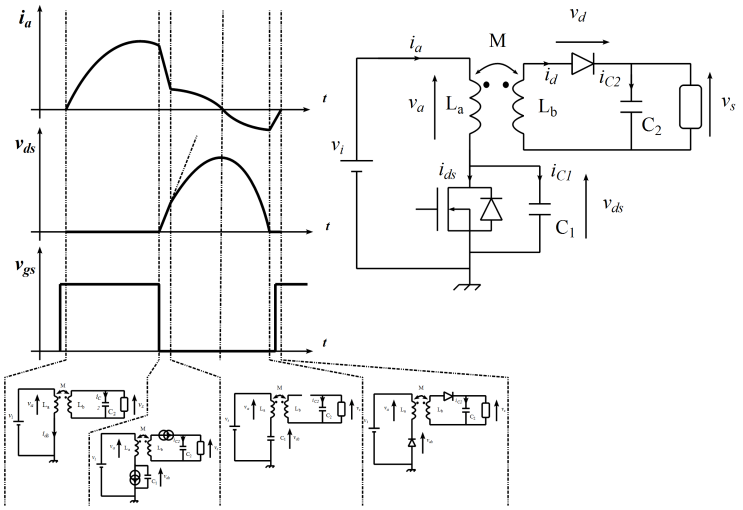


Figure 5.10 – Convertisseur de puissance à commutation douce pour transmission de puissance avec transformateur magnétique à grand entrefer. Principe de fonctionnement et chronogrammes associés.

L'objectif premier pour ce type de convertisseur est d'assurer un rendement le plus élevé possible. Le rendement est donc considéré dans la suite comme la

fonction objectif à maximiser. Dans le détail, le calcul de la fonction objective est organisé comme suit :

- Détermination des grandeurs F, C, x_1, x_2, e_1, e_2 et e_3 par l'algorithme d'optimisation.
- Calcul des éléments magnétiques et résistifs du transformateur de puissance à la fréquence de commutation f (logiciel éléments-finis FEMM Meeker, 2010).
- Simulation transitoire du convertisseur de puissance avec LTSpice (Brocard, 2011).
- Post-traitement pour le calcul du rendement.
- Renvoie la valeur numérique du rendement à l'algorithme d'optimisation.

Sur la figure 5.11, un synoptique est proposé afin d'illustrer au mieux la démarche de calcul du rendement avec les logiciels FEMM et LTSpice. Dans un

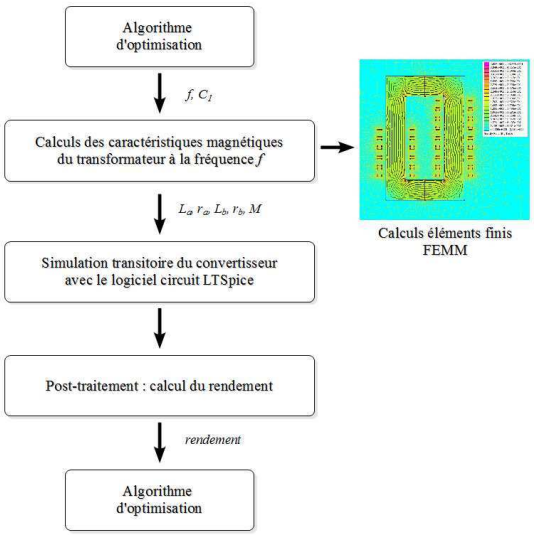


Figure 5.11 – Synoptique du calcul du rendement au sein de l'algorithme d'optimisation. Mise en œuvre de simulations éléments finis (FEMM) et de type circuit (LTSpice).

premier temps, nous restreignons le problème d'optimisation du rendement à un problème à seulement deux variables actives (F et C), puis nous considérons l'optimisation avec l'ensemble des sept variables actives. Ceci nous permet

de tester le comportement et les performances de notre algorithme pour des domaines $\mathbb{X}$ de dimensions différentes. Considérer le cas de la dimension deux permet également de faciliter certaines représentations graphiques.

5.3.2 Optimisation en dimension 2

Nous fixons les valeurs de x_1, x_2, e_1, e_2 et e_3 respectivement à 4.1mm, 1.97mm, 1mm, 1mm et 0.5mm (valeurs que nous savons raisonnables). Il s'agit donc ici d'un problème à deux dimensions, dont les variables sont la valeur de la fréquence de découpage F et la capacité du condensateur de résonance C, le cas général étant quant à lui traité à la section suivante. Cette fonction rendement en deux dimensions est visible sur la figure 5.12. La zone correspondant aux valeurs de rendement les plus élevées, dont fait partie le maximum, est assez large et les variations y sont assez faibles, rendant ainsi non trivial le problème d'optimisation.

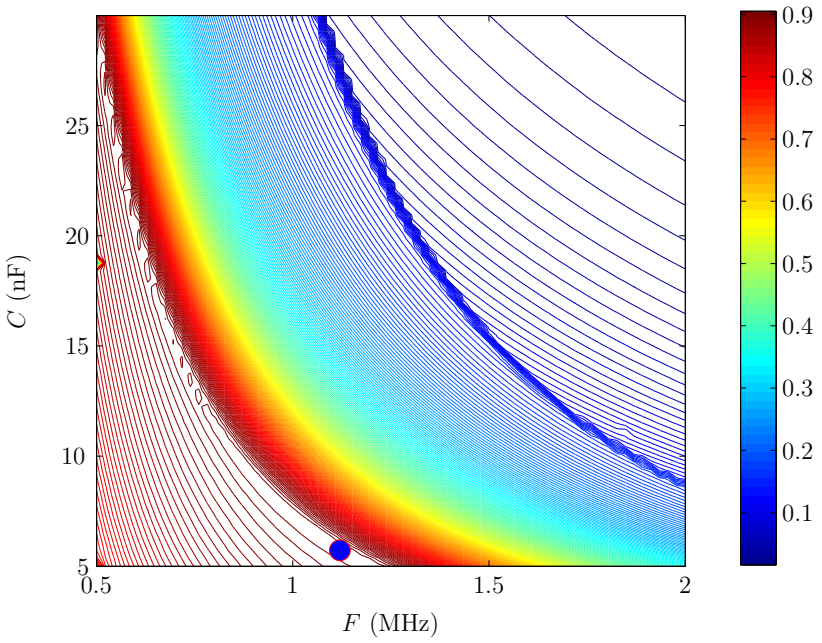
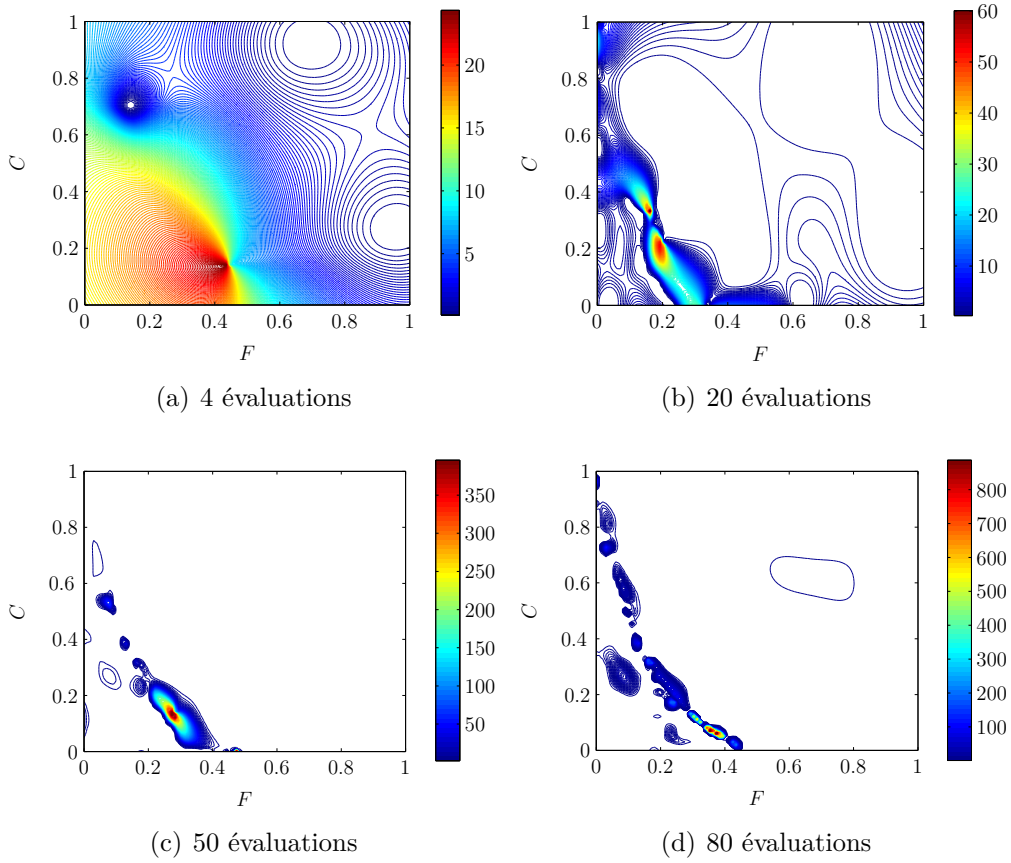


Figure 5.12 – Représentation du rendement du convertisseur en fonction des valeurs de fréquence F et de capacité C choisies. Le point correspondant au rendement maximal est représenté en bleu. La zone auquelle il appartient (correspondant à la teinte de rouge la plus foncée) est assez large.



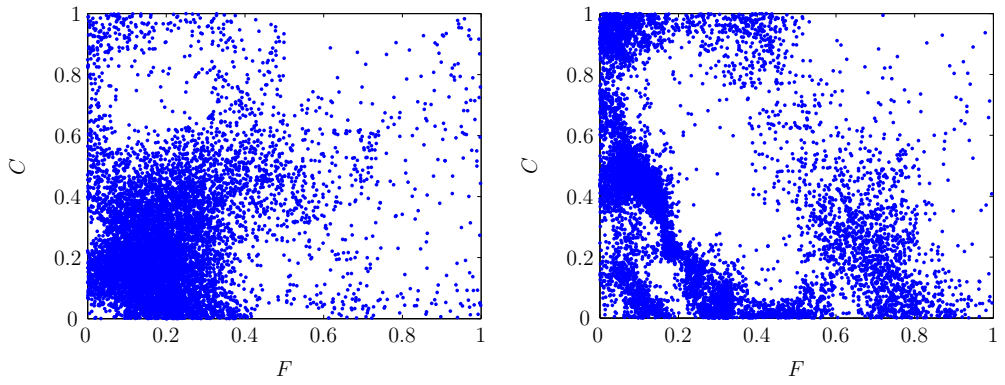
(a) 4 évaluations

(b) 20 évaluations

(c) 50 évaluations

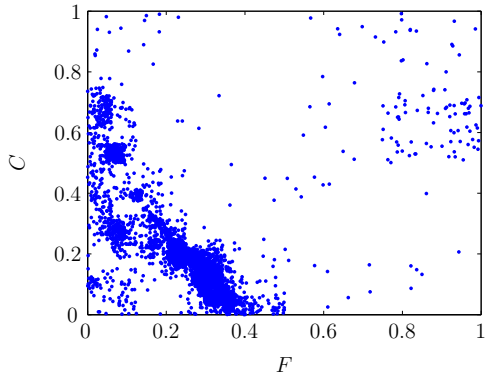
(d) 80 évaluations

Figure 5.13 – Valeurs du critère (probabilité d'amélioration normalisée) pour la répartition des points candidats x, pour différents nombres d'évaluations.

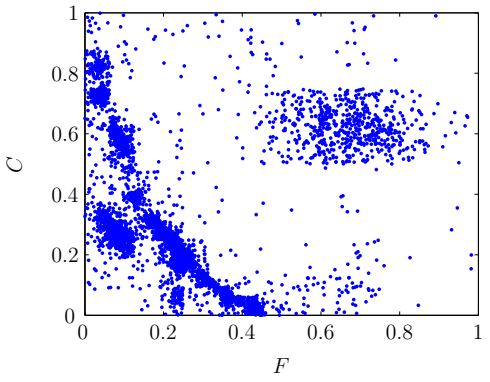


(a) 4 évaluations

(b) 20 évaluations

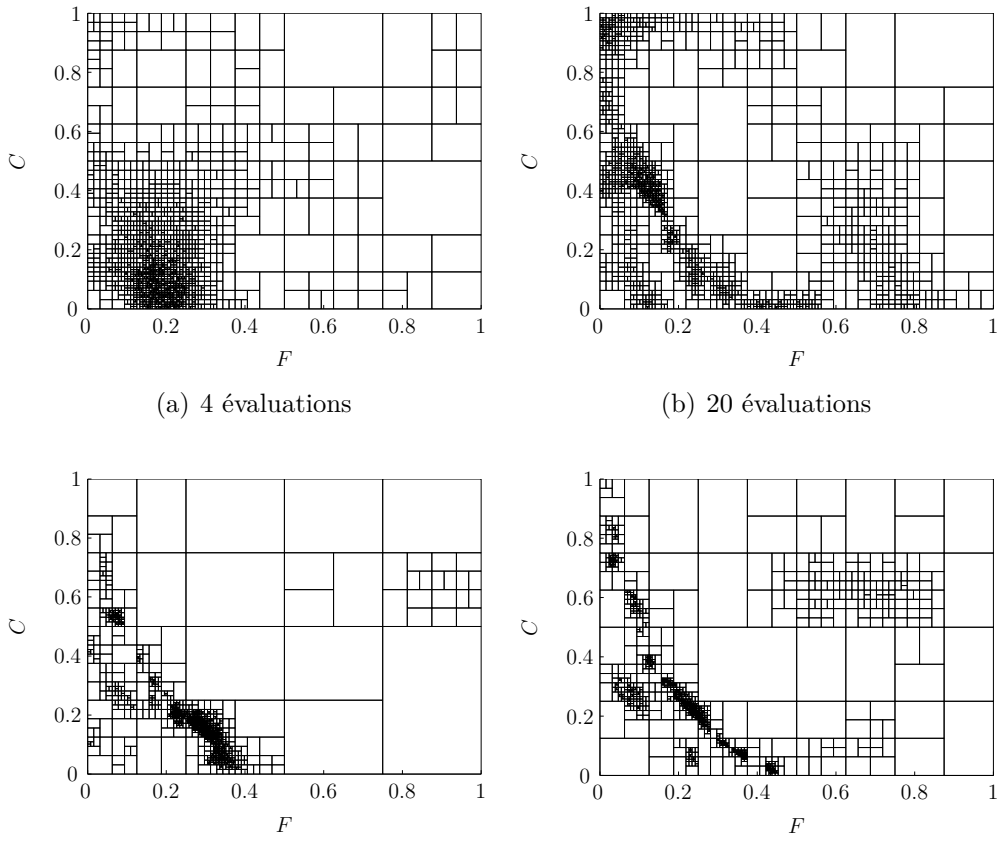


(c) 50 évaluations



(d) 80 évaluations

Figure 5.14 – Répartition des points candidats x, pour différents nombres d'évaluations.



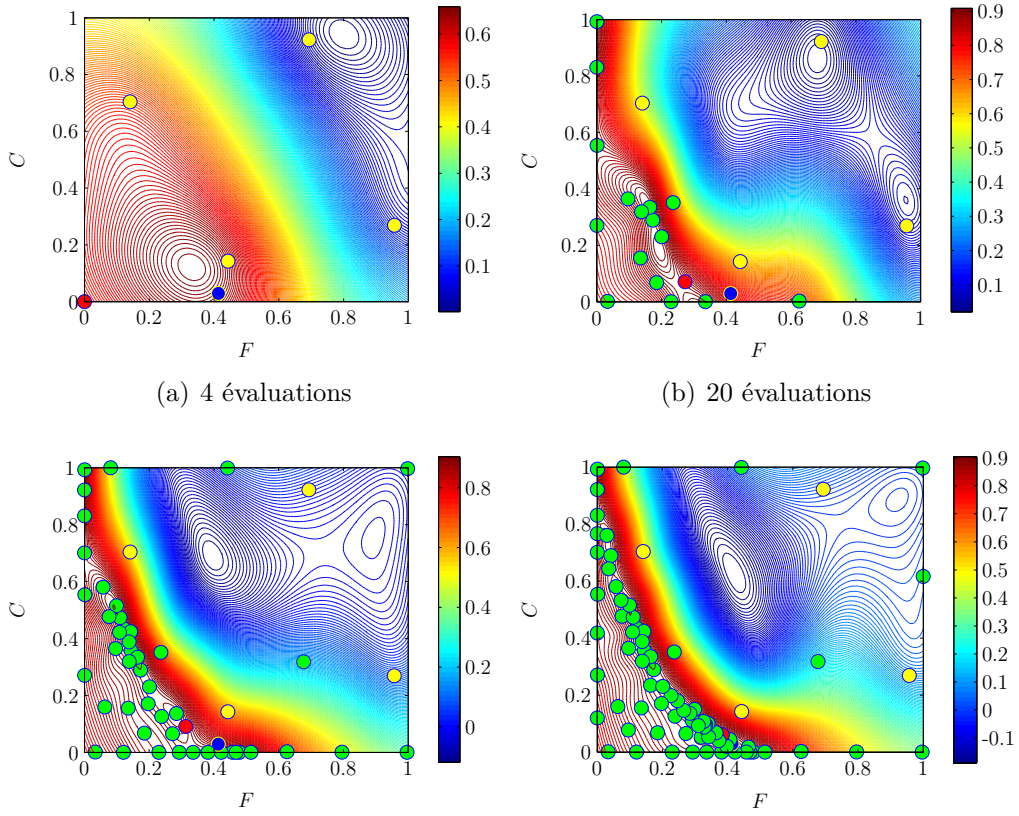
(a) 4 évaluations

(b) 20 évaluations

(c) 50 évaluations

(d) 80 évaluations

Figure 5.15 – Partitions séquentielles du domaine de définition pour différents nombres d'évaluations. Construction d'un arbre utilisé afin d'estimer la probabilité d'amélioration normalisée (représentée à la figure 5.13).



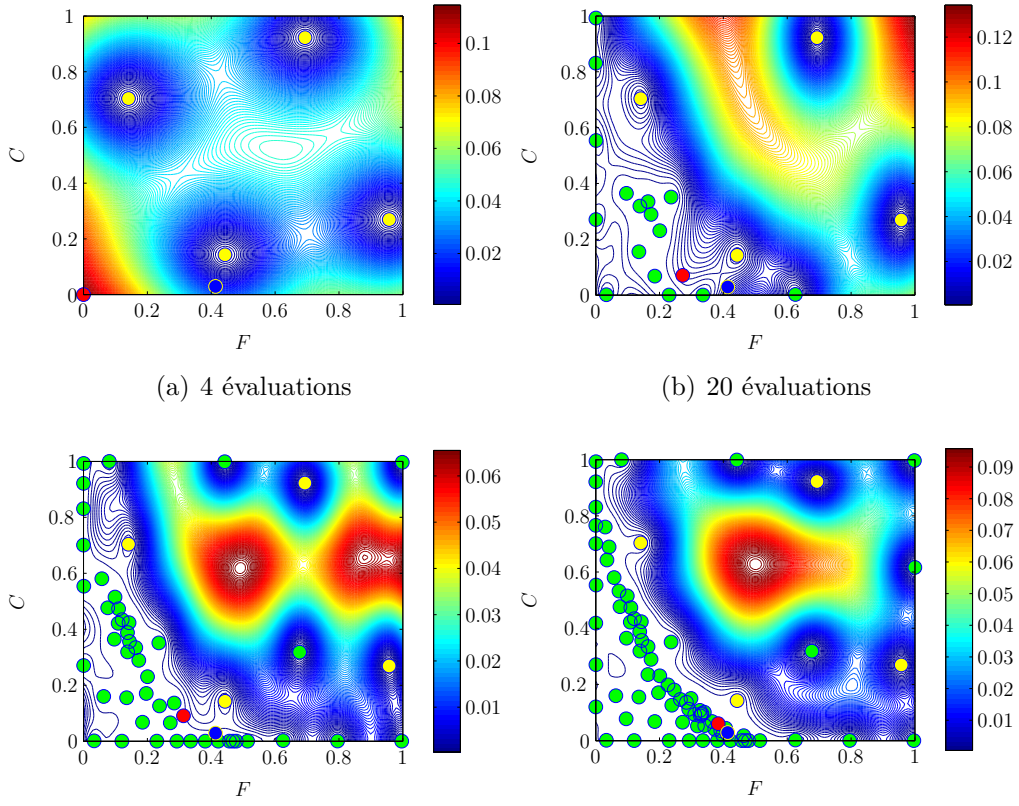
(a) 4 évaluations

(b) 20 évaluations

(c) 50 évaluations

(d) 80 évaluations

Figure 5.16 – Valeurs du prédicteur de la fonction rendement sur le domaine, pour différents nombres d'évaluations. Les points jaunes correspondent au plan d'expériences initial, les points verts aux évaluations suivantes. Le point rouge indique la position de la prochaine évaluation tandis que le point bleu correspond au maximum global du rendement.



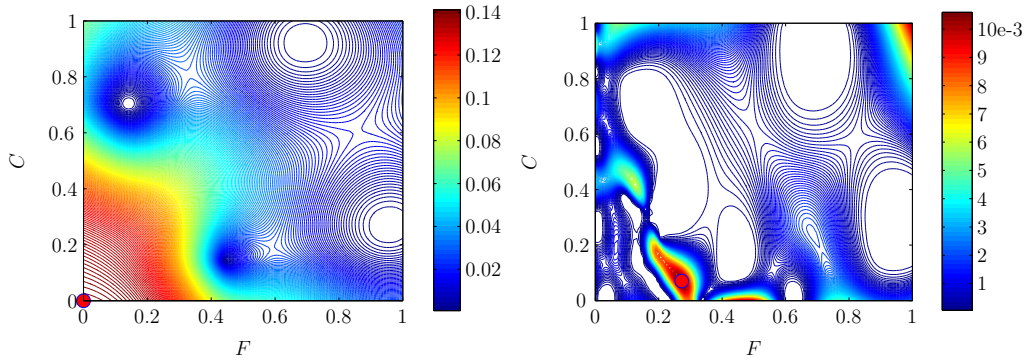
(a) 4 évaluations

(b) 20 évaluations

(c) 50 évaluations

(d) 80 évaluations

Figure 5.17 – Valeurs de la variance de prédiction sur le domaine, pour différents nombres d'évaluations. Les points jaunes correspondent au plan d'expériences initial, les points verts aux évaluations suivantes. Le point rouge indique la position de la prochaine évaluation tandis que le point bleu correspond au maximum global du rendement.



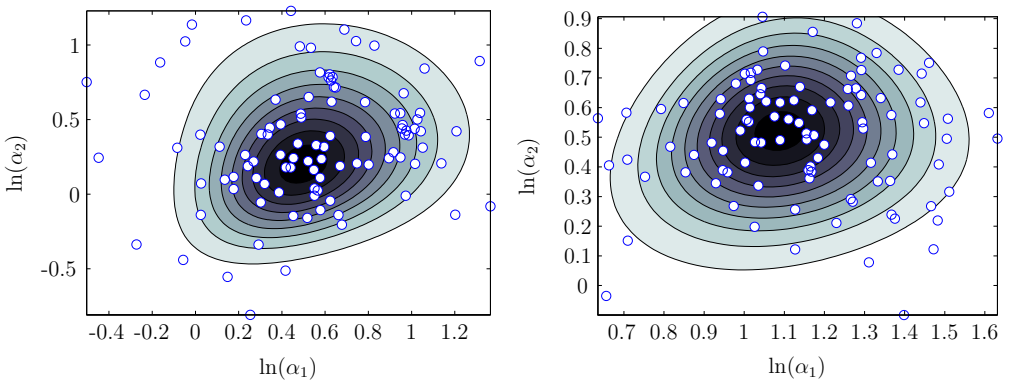
(a) 4 évaluations

(b) 20 évaluations

(c) 50 évaluations

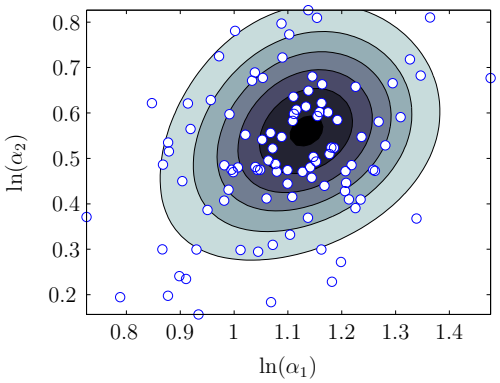
(d) 80 évaluations

Figure 5.18 – Valeurs du critère EI (complètement bayésien) sur le domaine, pour différents nombres d'évaluations. Le point rouge représente le vrai maximum de l'EI.

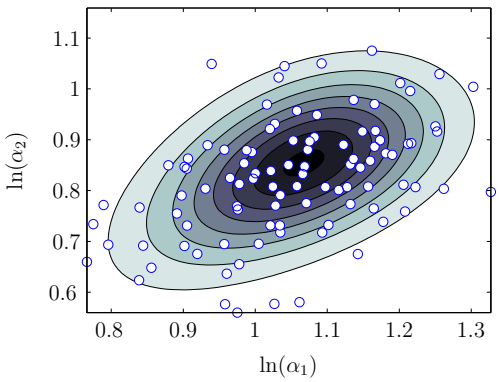


(a) 4 évaluations

(b) 20 évaluations



(c) 50 évaluations



(d) 80 évaluations

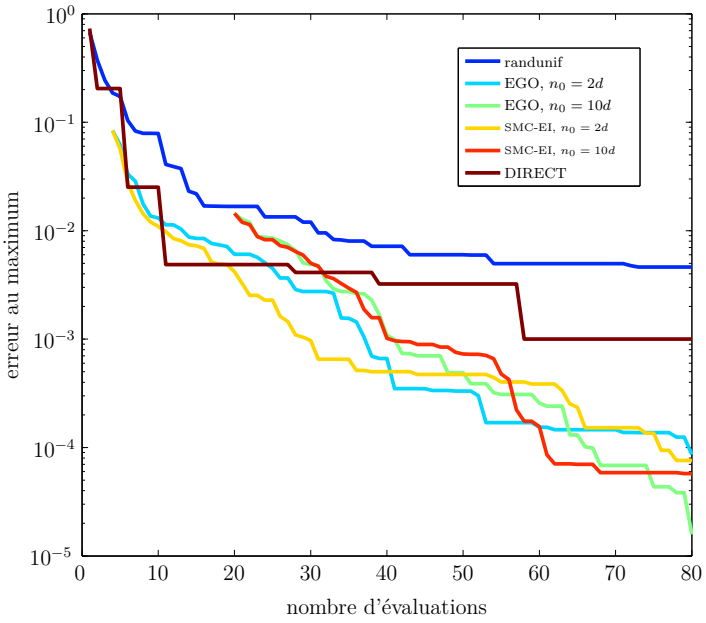
Figure 5.19 – Densité *a posteriori* π_n et répartition des particules en $\theta = (\alpha_1, \alpha_2)$ et comparaison avec la densité *a posteriori* théorique.

Afin d'illustrer le comportement de notre algorithme complètement bayésien, nous observons le comportement des I valeurs de θ ainsi que des IJ valeurs de points candidats en fonction du nombre d'évaluations de la fonction objectif (le rendement du convertisseur de puissance). Nous considérons un plan d'expérience initial de taille 4 (deux fois la dimension) évaluations réparties selon un LHS. Les *a priori* ainsi que les valeurs de paramètres choisis sont ceux décrits à la section 5.1. Nous étudions le comportement de l'algorithme, aussi bien sur le domaine $\mathbb{X}$ que sur l'espace Θ, pour différents nombres d'évaluations (4, 20, 50 et 80).

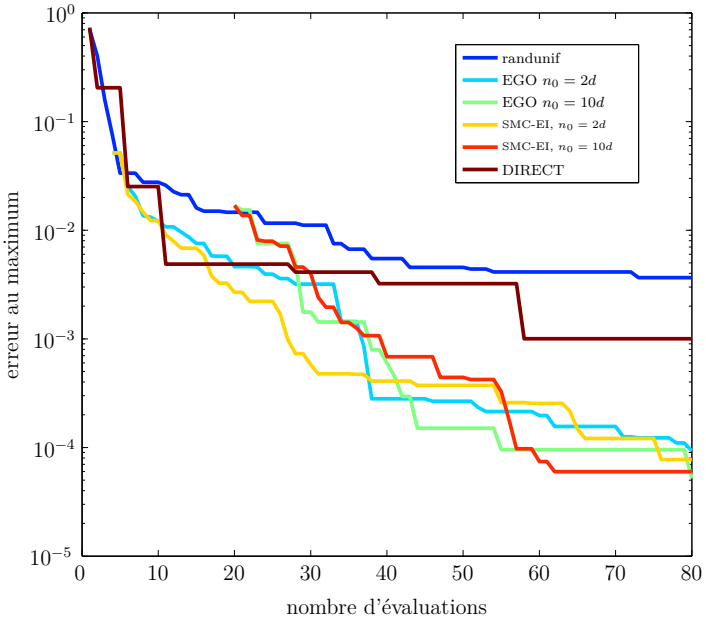
La figure 5.13 représente l'évolution de la probabilité d'amélioration normalisée, et doit être mise en regard avec les figures 5.14 et 5.15 représentant respectivement la position des points candidats et la partition du domaine utilisée pour les générer. En effet, la comparaison de ces trois figures nous montre que les points candidats sont échantillonnés dans des zones ou le critère est très élevé. Après quelques dizaines d'évaluations, deux zones de répartition privilégiées se distinguent. Mettre en regard la fonction objectif (figure 5.12) avec ces deux zones permet de les caractériser. L'une d'entre elles correspond à la partie du domaine de forme oblongue incurvée caractérisant les rendements les plus élevés, dont fait partie le maximum, tandis que l'autre correspond à une zone où le rendement est particulièrement faible, et où très peu d'évaluations ont été faites. Cette répartition des x semblent raisonnable. En effet, la densité est importante dans la zone contenant le maximum, mais également dans les parties du domaine où peu d'évaluations ont été effectuées. Il y a donc un compromis local/global effectif dans la répartition des points candidats.

La figure 5.16, montre les valeurs des prédicteurs. Nous observons que la zone où le prédicteur est le plus élevé correspondant à celle où le rendement est le plus élevé. Nous avons remarqué ci-dessus que cette partie du domaine est celle avec la plus forte densité de points candidats x.

La figure 5.17 correspond aux valeurs de la variance de prédiction. Nous observons que la variance est faible dans les zones où de nombreuses évaluations de f ont été faites. Elle est néanmoins importante dans les zones les moins évaluées. En effet, le peu d'information disponible entraîne une forte valeur de



(a) moyenne



(b) mediane

Figure 5.20 – Comparaisons entre différents algorithmes (bayésiens ou non) de l'erreur au maximum pour la maximisation du convertisseur en dimension deux (les moyennes et médianes, pour les algorithmes stochastiques, sont calculées à partir de 10 processus d'optimisation).

la variance de prédiction en ces points.

Le critère EI (complètement bayésien) est représenté sur la figure 5.18. Lorsque nous disposons de 50 ou 80 résultats d'évaluations, les zones où l'EI est le plus haut sont très réduites et très proches du maximum théorique. Un retour à la figure 5.14 nous montre que ce sont également les zones où la densité des points candidats est la plus importante. Le maximum de l'EI, pour ces itérations de l'algorithme, est donc estimé avec beaucoup de précision, ce qui était une des motivations essentielles de la construction de notre algorithme. Sur ces figures, l'EI semble favoriser une optimisation locale à une optimisation globale. Néanmoins, la variance de prédiction étant plus élevée dans la partie du domaine la plus loin des points d'évaluations, il est attendu que l'EI finisse par explorer à nouveau dans cette zone (la forte concentration de points candidats dans celle-ci devrait permettre, le cas échéant, à nouveau une optimisation efficace de l'EI).

La figure 5.19 montre les valeurs des particules en θ, superposées à la densité *a posteriori* π_n. La répartition des particules correspond à la densité *a posteriori*, ce qui traduit le bon fonctionnement de l'algorithme concernant les particules en θ.

Nous effectuons également des comparaisons de notre algorithme avec EGO, pour différentes tailles du plan d'expérience initial ($n_0 = 2d$ ou $n_0 = 10d$). La grandeur considérée pour ces comparaisons est, en fonction du nombre d'évaluations, l'erreur entre le maximum courant et la plus grande valeur de rendement obtenue expérimentalement sur l'ensemble des simulations ayant été effectuées jusqu'alors (la valeur du maximum théorique n'étant pas connue). Nous étendons ces comparaisons à l'algorithme DIRECT, ainsi qu'à une stratégie de référence consistant à échantillonner à chaque itération selon une loi uniforme sur le domaine. Pour les algorithmes stochastisques, c'est-à-dire tous sauf DIRECT qui est déterministe, nous nous intéressons aux moyenne et médiane de ces erreurs à partir de 10 simulations répétées entièrement avec, pour chacune d'elles, un budget de 80 évaluations. Nous considérons seulement 10 simulations pour des raisons pratiques, la fonction objectif étant particulièrement coûteuse. La configuration et le choix de paramètres des deux algo-

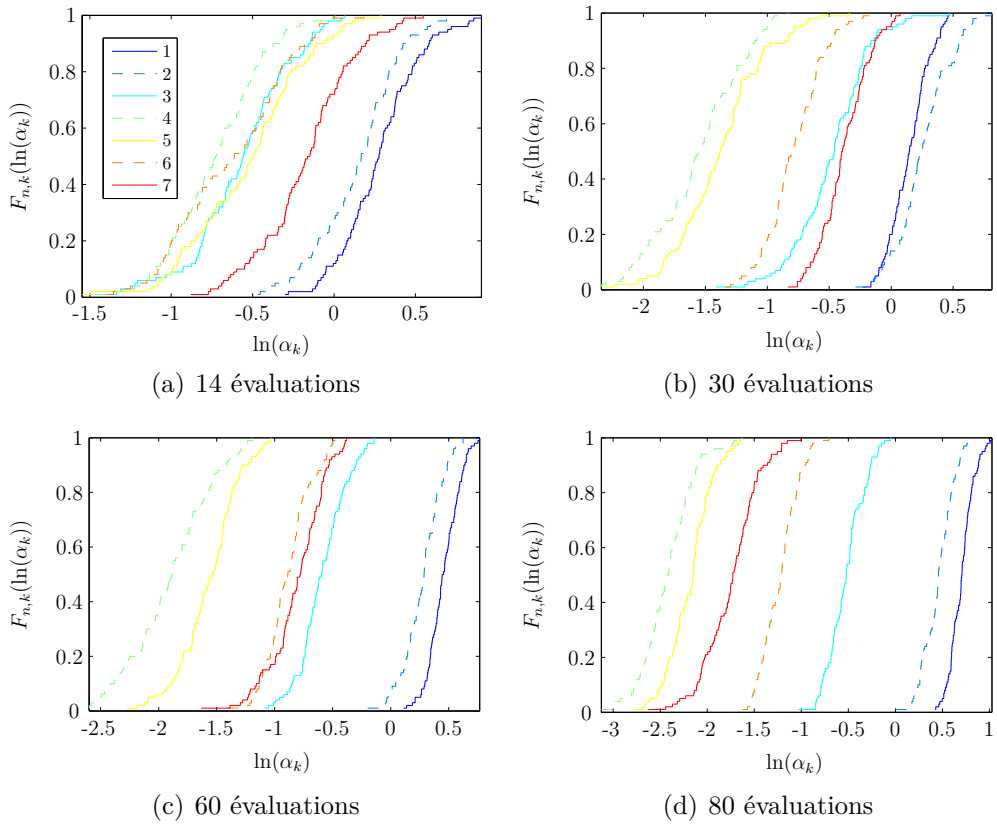
rithmes bayésiens sont décrits à la section 5.1. Les résultats obtenus, illustrés par la figure 5.20, nous montrent que les algorithmes bayésiens (SMC-EI et EGO) font mieux que DIRECT (et que la référence). Cependant, aussi bien pour la moyenne que la médiane, il est difficile de les discriminer, quel que soit la taille du plan d'expérience initial ($n_0 = 2d$ ou $n_0 = 10d$).

5.3.3 Optimisation en dimension 7

Les analyses ainsi que les figures considérées dans cette section sont analogues à celles de la section précédente (optimisation du convertisseur en dimension deux). S'y référer, si besoin, pour plus de détails.

La figure 5.21 présente l'évolution des fonctions de répartition empiriques $F_{n,i}$ marginales de π_n, définies à la section 5.1. Le budget est de 80 évaluations dont 14 (soit $2d$), répartis selon un LHS, constituent le plan d'expérience initiale. Après 80 évaluations, ce sont F et C, les deux variables considérées pour le cas à seulement deux dimensions, qui s'avèrent les plus importantes. Viennent ensuite, dans cet ordre, les variables x_1, e_2, e_3, e_1 et x_2. Trois « groupes » de variables semblent se distinguer, F et C semblent très influentes, e_1 et x_2 semblent très peu l'être, et les restantes présentent une influence relativement faible.

Nous faisons des comparaisons des performances (erreur au maximum) en suivant exactement le même mode opératoire qu'à la section 5.3.2. Les résultats, présentés à la figure 5.22, sont sensiblement les mêmes concernant moyenne et médiane. Pour les deux algorithmes stochastiques (SMC-EI et EGO), considérer un plan d'expérience initial de taille $n_0 = 70$ (c'est-à-dire dix fois la dimension) ne permet pas d'optimiser efficacement lorsque le budget total est de 80 évaluations. Ceci n'est pas particulièrement surprenant mais met en évidence qu'optimiser, avec $n_0 = 10d$, nécessite un temps certain (celui nécessaire à évaluer les $10d$ premières évaluations) avant d'optimiser réellement, ce qui peut poser problème lorsque d est grand. En effet, pour $n_0 = 14$ (deux fois la dimension), l'optimisation est effective aussi bien pour EGO que pour notre stratégie SMC-EI, avec un léger avantage pour cette dernière lors des ultimes évaluations. Il est à noter que les performances de DIRECT sont



(a) 14 évaluations (b) 30 évaluations

(c) 60 évaluations (d) 80 évaluations

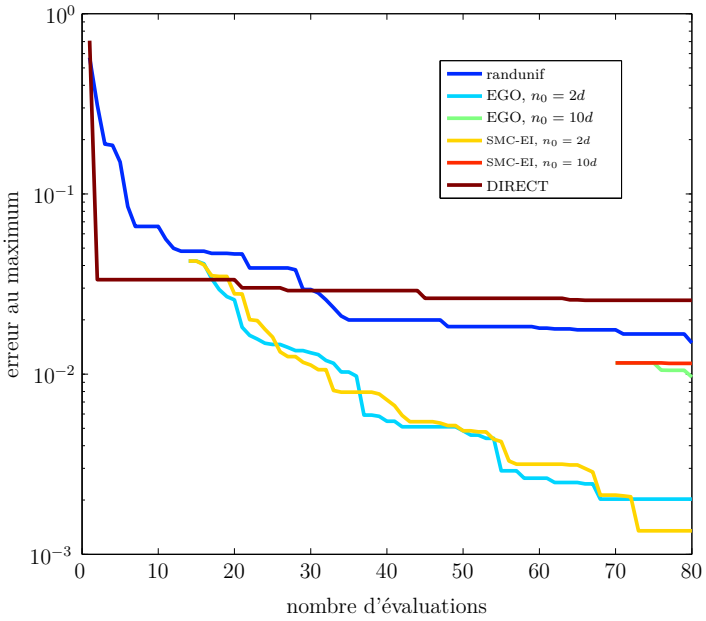
Figure 5.21 – Fonctions de répartition empiriques des marginales de π_n.

les plus mauvaises, puisqu'également moins bonnes que celle de la moyenne de la référence (échantillonnage selon une loi uniforme).

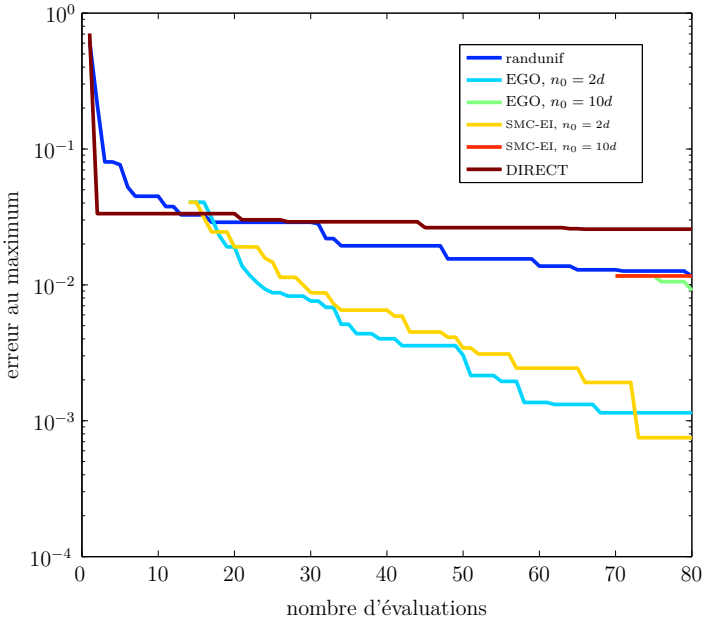
5.4 Étude de performances sur des fonctions de test classiques

5.4.1 Configuration des algorithmes considérés

Afin de mettre en perspective les performances de notre algorithme avec d'autres, nous faisons plusieurs jeux d'expériences avec différentes fonctions tests. L'idée est de situer ainsi les performances par rapport à d'autres algorithmes, d'ordre bayésien tel qu'EGO dont la mise en œuvre considérée ici



(a) moyenne



(b) mediane

Figure 5.22 – Comparaisons entre différents algorithmes (bayésiens ou non) de l'erreur au maximum pour la maximisation du convertisseur en dimension sept (les moyennes et médianes, pour les algorithmes stochastiques, sont calculées à partir de 10 processus d'optimisation).

est issue du *package DiceOptim* (Ginsbourger et Roustant, 2011), mais également de nature plus générale. Nous utilisons en effet également des algorithmes déterministes comme DIRECT et MCS (*Multilevel Coordinate Search*). L'algorithme DIRECT correspond à la description faite par Finkel (2003), dont nous reprenons la mise en œuvre. L'algorithme MCS est une variante de DIRECT, et nous le reprenons tel que décrit par Huyer et Neumaier (1999), avec les paramètres conseillés par défaut, et l'option de recherche locale activée. Un échantillonnage uniforme sur le domaine $\mathbb{X}$, noté rand, est utilisé comme référence.

Notre algorithme complètement bayésien, noté SMC-EI, est utilisé avec les paramètres et les *a priori* définis à la section 5.1. Le plan d'expérience initial est de taille $n_0 = 2d$, et correspond à un LHS. L'algorithme EGO considéré ici est mis en œuvre à partir du *package*, codé en R (langage de programmation et environnement statistique), du nom de *DiceOptim*. Le calcul du critère EI se fait avec une approche type *substitution*, à chaque itération le modèle de krigeage ainsi que les valeurs des paramètres de covariance sont réestimés en fonction des évaluations du plan d'expérience initial ainsi que des évaluations issues des précédentes itérations. Nous utilisons les paramètres par défaut et, parmi les fonctions de covariance disponibles, nous choisissons une covariance de Matérn avec une régularité de 5/2. Il est à noter que contrairement à celle que nous utilisons pour notre algorithme, la fonction de covariance de Matérn disponible dans le *package DiceKriging* est séparable. Son expression est donnée explicitement par Roustant et al. (2012), et est construite à partir de l'expression de (Rasmussen et Williams, 2006). La maximisation du critère EI est effectuée à l'aide d'un algorithme génétique, faisant usage du gradient de l'EI, disponible dans le *package rgenoud* (Walter et Jasjeet, 2011) du logiciel R. Deux tailles de plan d'expériences initial, $n_0 = 2d$ et $n_0 = 10d$, sont considérées pour EGO. Ces plans d'expérience initiaux sont répartis selon un LHS.

5.4.2 Fonctions tests et nature des tests

L'objectif est de comparer les performances de notre algorithme complètement bayésien avec des algorithmes de référence de la littérature, sur des

exemples de fonctions tests classiques dans le domaine de l'optimisation, issues de (Dixon et Szego, 1978), dont les expressions sont également données en annexes : section C. Un récapitulatif des fonctions utilisées, ainsi que de leurs principales propriétés, est fait sous la forme du tableau 5.1. Des représentations graphiques sont données aux figures 5.23 à 5.30. Les fonctions tests de dimension 2 sont directement représentées sur les figures. En dimensions supérieures, seules des coupes le sont. Les coupes sont effectuées au milieu du domaine de définition.

Fonctions test	Dim.	Nb. optima locaux (globaux)	Val. max.
Branin	2	3 (3)	-0.397887
Goldstein & Price	2	- (1)	-3
Camel back	2	6 (2)	0
Shubert	2	- (18)	186.7309
Hartman 3	3	4 (1)	3.86278
Hartman 6	6	6 (1)	3.32237
Shekel 5	4	5 (1)	10.1532
Shekel 7	4	7 (1)	10.4029
Shekel 10	4	10 (1)	10.5364

Table 5.1 – Récapitulatif des fonctions tests utilisées. Les fonctions sont considérées ici dans un cadre de maximisation, dans la littérature elles le sont parfois dans un cadre de minimisation (le passage de l'un à l'autre se fait par multiplication par un facteur -1). Lorsque le nombre d'optima locaux n'est pas précisé, c'est qu'ils sont très nombreux. Dans ce cas, nous renvoyons le lecteur directement aux figures associées pour prendre conscience de la difficulté du problème.

Afin de comparer les performances des différents algorithmes nous nous intéressons à l'erreur entre le maximum de la fonction (connu) et le maximum courant pour n résultats d'évaluations. Plus particulièrement, nous fixons plusieurs seuils de précision à atteindre, et nous regardons le nombre d'évaluations nécessaires pour l'obtenir. En pratique un budget de 100 évaluations est considéré et, si la précision voulue n'est pas obtenue après épuisement du budget, c'est alors cette valeur 100 qui est considérée. Pour les algorithmes stochas-

tiques, le nombre d'évaluation nécessaires, pour un seuil de précision donné, est calculé en moyenne sur 100 processus d'optimisation. Lorsque la précision voulue n'est pas atteinte à la fin d'un des processus d'optimisation, nous considérons par défaut dans le calcul de la moyenne que la précision est atteinte pour 100 évaluations. Notons également que pour SMC-EI et EGO, les plans d'expériences initiaux sont regénérés au début de chacun des 100 processus d'optimisation.

Les différentes fonctions tests considérées ont parfois des plages de variations très différentes les unes des autres. Afin que les seuils de précision considérés caractérisent l'efficacité de l'optimisation indépendamment de la fonction test considérée, nous effectuons une « normalisation » de l'erreur. Concrètement, pour une précision ϵ donnée, nous considérons que celle-ci est atteinte à la plus petite valeur n possible vérifiant

$$\frac{|M - M_n|}{v_2 - v_1} \leq \epsilon,$$

où v_1 et v_2 correspondent respectivement au quantile à 5 et à 95% d'un échantillon de 10^6 résultats d'évaluation de la fonction considérée. Ces évaluations ont été obtenues à partir de points échantillonnés sur le domaine de définition $\mathbb{X}$ à partir d'une loi uniforme.

5.4.3 Résultats

Les tables 5.2 à 5.9 représentent, pour chaque fonction test considérée, les résultats obtenus. Nous représentons en gras, pour chaque niveau de précision, les performances du meilleur algorithme ainsi que les performances des algorithmes se situant à moins de 10% du meilleur. Pour les algorithmes stochastiques, nous donnons entre parenthèses le nombre de fois où la précision seuil est atteinte avant épuisement du budget d'évaluation. Deux algorithmes se distinguent, le nôtre et MCS. Dans un premier temps, nous analysons les résultats pour les algorithmes bayésiens et pour les algorithmes déterministes séparément.

Concernant les algorithmes bayésiens SMC-EI s'avère, dans la grande majorité des cas, plus performant qu'EGO. Nous avons en effet mis en évidence au

cours du manuscrit que notre approche SMC conjointe permet à la fois une plus grande robustesse (c'est-à-dire une diminution du risque que certains processus d'optimisation aient des performances mauvaises en comparaison des autres) ainsi qu'une recherche locale efficace. Ces deux aspects sont particulièrement visibles pour l'ensemble des fonctions tests considérées, mise à part pour Hartman 6 (en log) où les performances de SMC-EI ne sont pas meilleures que celles d'EGO mais comparables. Ceci nous permet de conclure que SMC-EI apporte une amélioration significative dans le domaine des algorithmes d'optimisation globale bayésiens. Nous pouvons également noter que, sur les exemples considérés, disposer d'un plan d'expérience initial de taille n_0 plus grande ne permet pas à EGO d'améliorer significativement ses performances.

Nous comparons également les performances des deux algorithmes déterministes DIRECT et MCS. Si les performances de DIRECT sont légèrement meilleures pour la fonction Hartman 3 ainsi que les fonctions Shekel, il apparaît néamoins sur toutes les autres que MCS est nettement plus performant. En effet, MCS réussit à effectuer une recherche locale efficace dans les zones les plus intéressantes, ce qui permet d'atteindre plus rapidement les niveaux de précisions que DIRECT, et d'en atteindre plus.

Il convient désormais de comparer notre algorithme SMC-EI avec MCS. Nous sommes nettement supérieur sur la fonction Hartman 3 et, mais moins nettement, pour Goldstein & Price, Shekel 7 et Shekel 10. Pour les quatre autres fonctions, MCS est le plus performant et SMC-EI est le deuxième meilleur (sauf pour Hartman 6, où la distinction entre SMC-EI et EGO est plus difficile). Concrètement, sur l'ensemble de ces exemples les deux stratégies les plus efficaces sont donc MCS et SMC-EI.

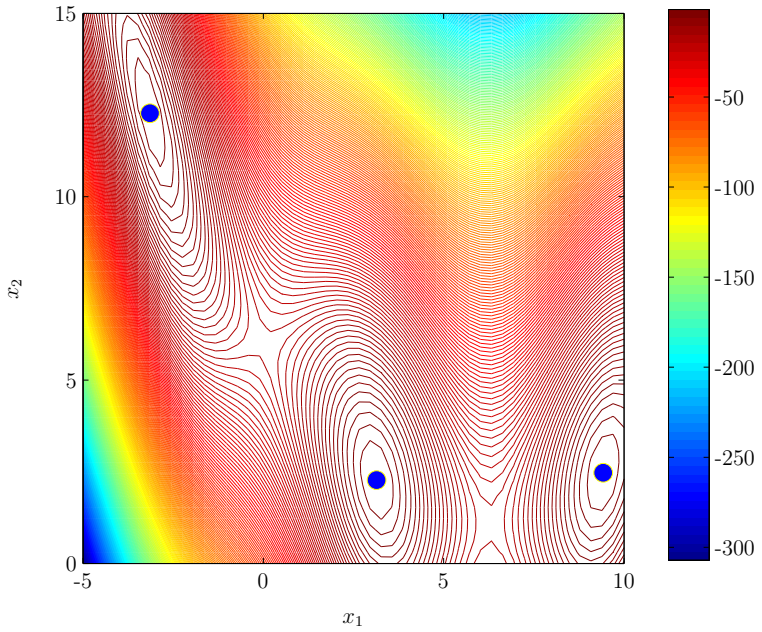


Figure 5.23 – Fonction Branin

Niv. de précision	SMC-EI	EGO ($n_0 = 2d$)	EGO ($n_0 = 10d$)	DIRECT	MCS	rand
1e−1	7.3 (100)	6.9 (100)	21.0 (100)	**2**	4	3.7 (100)
1e−2	15.1 (100)	11.3 (100)	21.5 (100)	21	**7**	23.8 (96)
1e−3	23.0 (100)	**15.6** (100)	21.9 (100)	21	23	79.7 (35)
1e−4	29.0 (100)	**21.4** (100)	26.5 (100)	61	26	97.5 (6)
1e−5	35.9 (100)	36.4 (95)	36.3 (99)	95	**26**	≥ 100
1e−6	40.7 (100)	66.5 (65)	66.4 (65)	≥ 100	**26**	
1e−7	43.7 (100)	89.8 (29)	86.6 (30)		**36**	
1e−8	45.8 (100)	95.8 (12)	94.9 (14)		**36**	
1e−9	53.2 (100)	98.7 (5)	≥ 100		**36**	
1e−10	69.9 (97)	≥ 100			**36**	
1e−11	92.8 (47)				**46**	
1e−12	98.8 (7)				**46**	

Table 5.2 – Résultats sur la fonction de Branin. Représentation des nombres d'évaluations nécessaires pour atteindre les différents seuils de précision (voir description détaillée aux pages 134 et 135). Les performances du meilleur algorithme, ainsi que celles des algorithmes à moins de 10% du meilleur, sont représentées en gras. Pour les algorithmes stochastiques, nous donnons entre parenthèses le nombre de fois où la précision seuil est atteinte, sur les 100 processus d'optimisation, avant épuisement du budget de 100 évaluations.

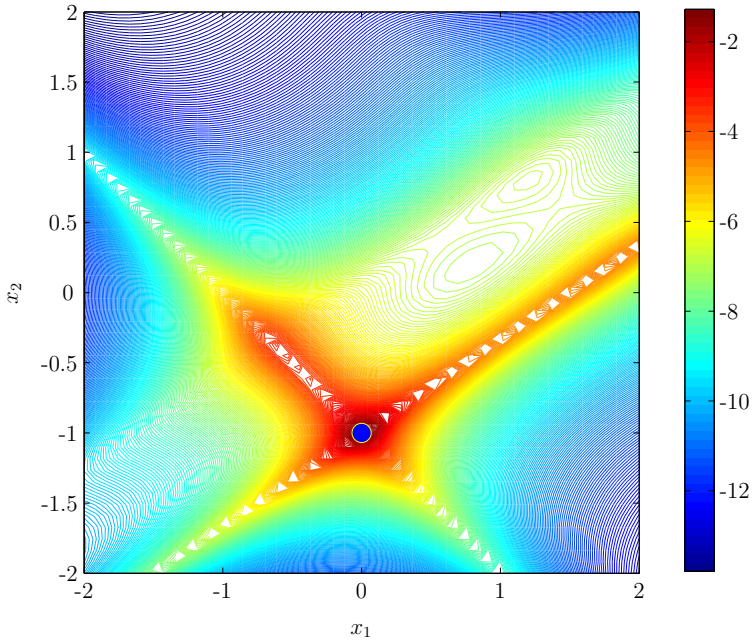


Figure 5.24 – Fonction de Goldstein & Price (en log)

Niv. de précision	SMC-EI	EGO ($n_0 = 2d$)	EGO ($n_0 = 10d$)	DIRECT	MCS	rand
1e−1	**22.7** (100)	27.7 (100)	34.8 (100)	36	45	92.0 (18)
1e−2	**27.8** (100)	56.3 (86)	63.4 (83)	55	55	≥ 100
1e−3	**36.8** (100)	87.7 (29)	90.8 (26)	72	59	
1e−4	**55.5** (97)	99.2 (2)	99.1 (2)	95	59	
1e−5	77.7 (72)	≥ 100	≥ 100	≥ 100	**61**	
1e−6	88.1 (52)				**61**	
1e−7	**93.5** (36)				≥ 100	
1e−8	**96.9** (23)					
1e−9	**98.1** (17)					
1e−10	**99.4** (10)					
1e−11	**99.8** (4)					
1e−12	≥ 100					

Table 5.3 – Résultats sur la fonction de Goldstein & Price (en log)

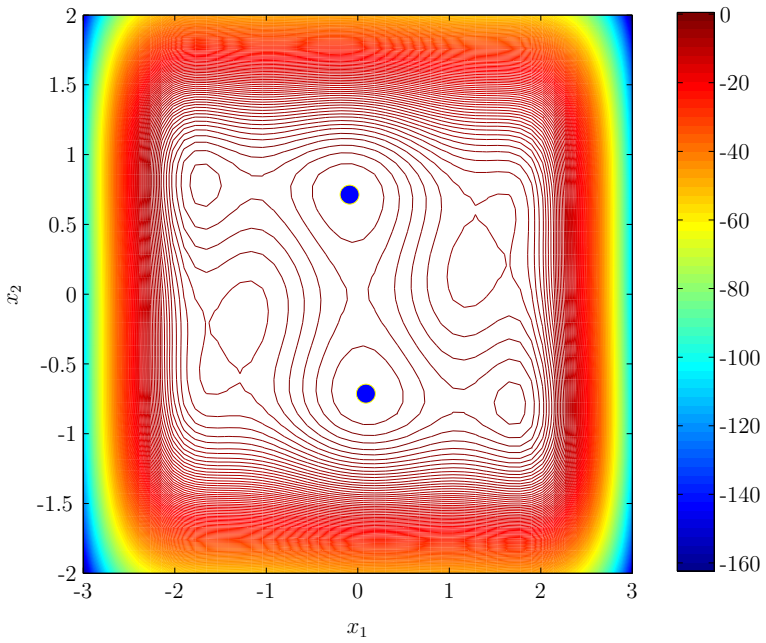


Figure 5.25 – Fonction Six-hump camel back

Niv. de précision	SMC-EI	EGO ($n_0 = 2d$)	EGO ($n_0 = 10d$)	DIRECT	MCS	rand
1e−1	6.6 (100)	6.03 (100)	21.1 (100)	**1**	**1**	2.1 (100)
1e−2	13.5 (100)	14.19 (100)	24.4 (100)	12	**7**	23.7 (100)
1e−3	26.9 (100)	24.55 (100)	28.7 (100)	76	**11**	81.0 (37)
1e−4	39.3 (100)	35.31 (100)	37.0 (100)	≥ 100	**28**	98.9 (3)
1e−5	49.4 (100)	58.7 (95)	52.7 (94)		**38**	99.9 (1)
1e−6	61.7 (100)	88.5 (38)	86.3 (47)		**38**	≥ 100
1e−7	67.2 (100)	96.2 (13)	98.3 (7)		**38**	
1e−8	72.1 (100)	99.4 (2)	99.1 (3)		**42**	
1e−9	82.7 (96)	≥ 100	≥ 100		**47**	
1e−10	94.2 (44)				**47**	
1e−11	99.3 (7)				**47**	
1e−12	≥ 100				**47**	

Table 5.4 – Résultats sur la fonction Six-hump camel back

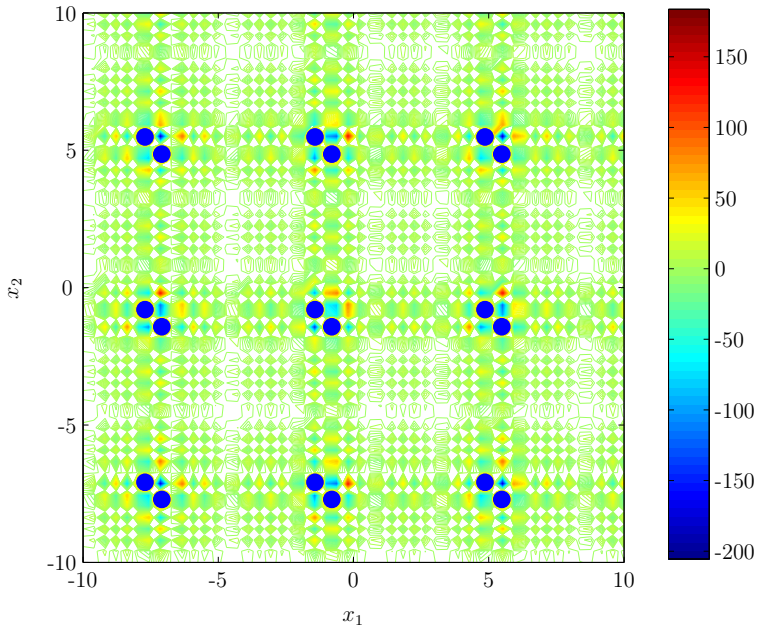
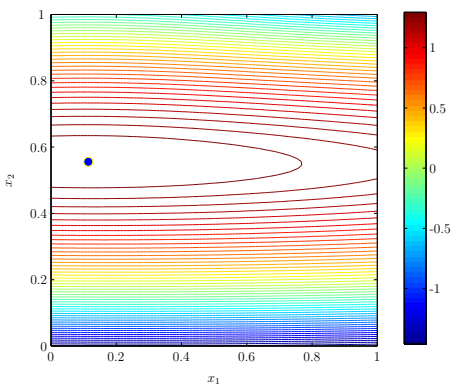


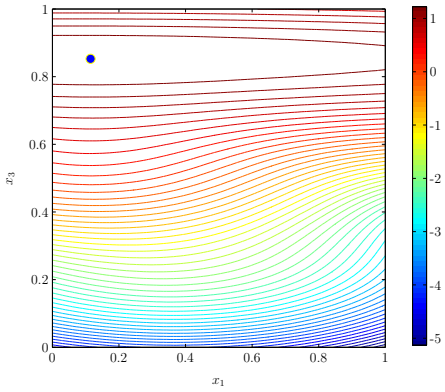
Figure 5.26 – Fonction de Shubert

Niv. de précision	SMC-EI	EGO ($n_0 = 2d$)	EGO ($n_0 = 10d$)	DIRECT	MCS	rand
1e−1	77.7 (43)	93.4 (20)	93.8 (17)	≥ 100	**54**	99.0 (3)
1e−2	82.5 (40)	98.0 (9)	97.5 (7)		**54**	100 (0)
1e−3	89.5 (34)	98.9 (5)	99.4 (2)		**64**	≥ 100
1e−4	94.2 (20)	≥ 100	≥ 100		**64**	
1e−5	97.2 (9)				**68**	
1e−6	99.5 (2)				**74**	
1e−7	≥ 100				**74**	
1e−8					**83**	
1e−9					**83**	
1e−10					**88**	
1e−11					**93**	
1e−12					**93**	

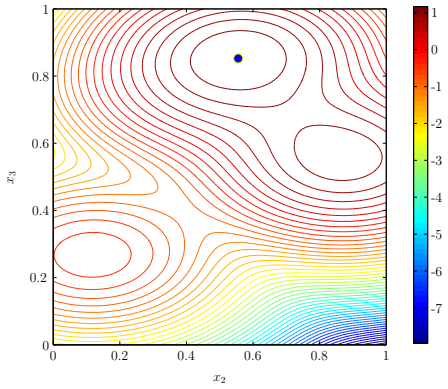
Table 5.5 – Résultats sur la fonction de Shubert



(a) Hart 3 (en log)-Coupe 1-2

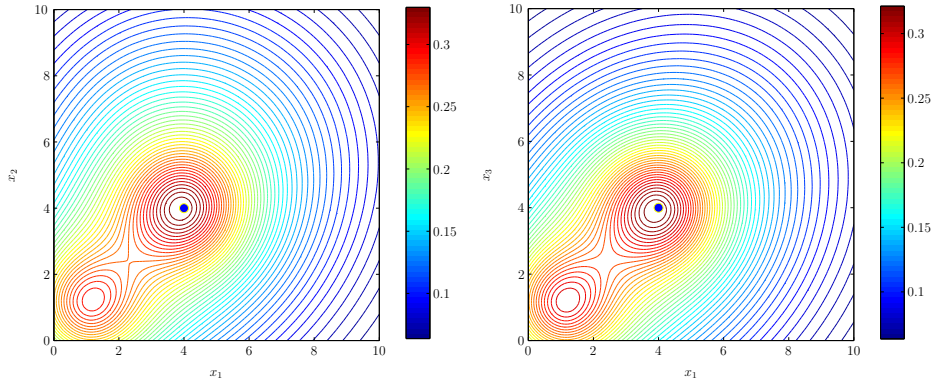


(b) Hart 3 (en log)-Coupe 1-3



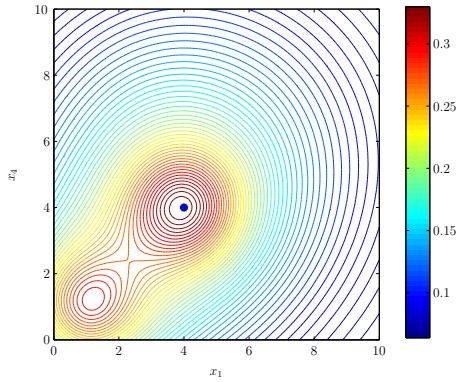
(c) Hart 3 (en log)-Coupe 2-3

Figure 5.27 – Fonction Hartman 3

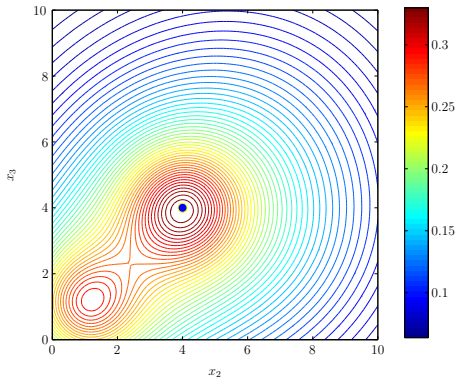


(a) Shekel 5-Coupe 1-2

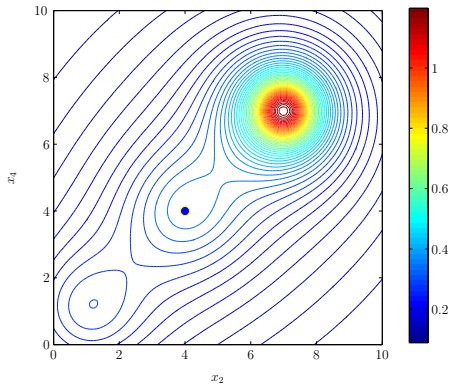
(b) Shekel 5-Coupe 1-3



(c) Shekel 5-Coupe 1-4



(d) Shekel 5-Coupe 2-3



(e) Shekel 5-Coupe 2-4

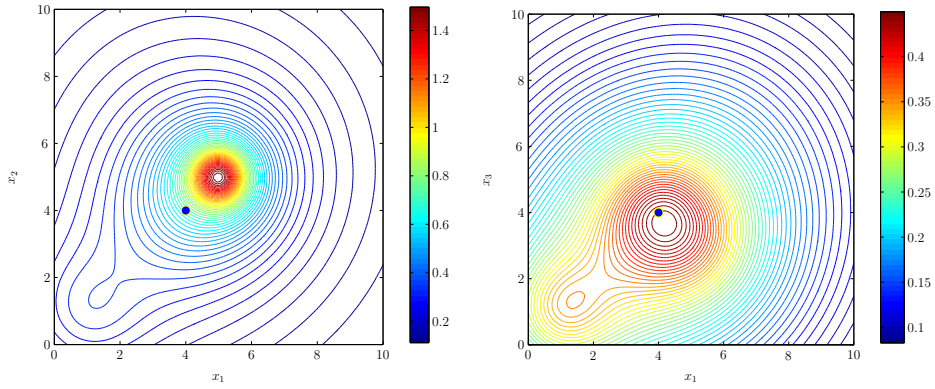
Figure 5.28 – Fonction Shekel 5

Niv. de précision	SMC-EI	EGO ($n_0 = 2d$)	EGO ($n_0 = 10d$)	DIRECT	MCS	rand
1e−1	10.5 (100)	9.14 (100)	31.1 (100)	7	**5**	6.3 (100)
1e−2	18.9 (100)	13.2 (100)	32.0 (100)	**8**	≥ 100	64.6 (58)
1e−3	**25.4** (100)	**23.8** (100)	40.6 (100)	95		98.6 (3)
1e−4	**33.6** (100)	48.1 (93)	58.3 (92)	≥ 100		≥ 100
1e−5	**40.0** (100)	89.7 (29)	93.7 (20)	100		
1e−6	**46.6** (100)	99.4 (3)	≥ 100			
1e−7	**51.7** (100)	≥ 100				
1e−8	**57.1** (100)					
1e−9	**73.1** (91)					
1e−10	**97.2** (16)					
1e−11	**99.7** (2)					
1e−12	≥ 100					

Table 5.6 – Hartman 3 (log).

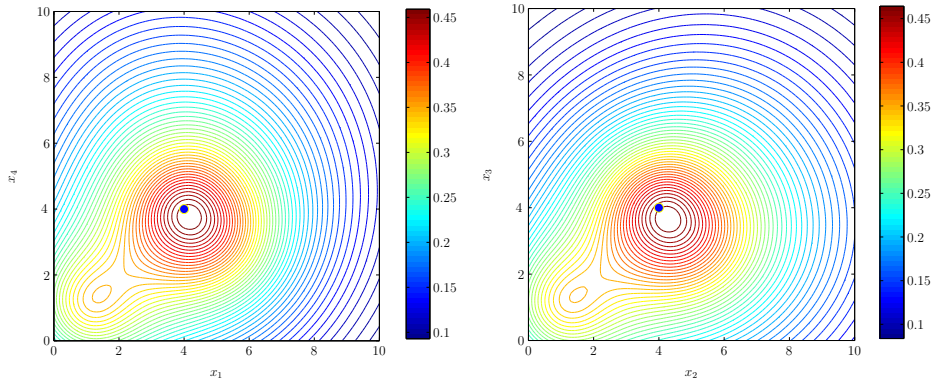
Niv. de précision	SMC-EI	EGO ($n_0 = 2d$)	EGO ($n_0 = 10d$)	DIRECT	MCS	rand
1e2	9 (100)	9 (100)	41 (100)	**1**	**1**	**1** (100)
1e1	**9** (100)	99.9 (1)	≥ 100	75	39	100 (0)
1e0	92.9 (40)	≥ 100		≥ 100	**81**	≥ 100
1e−1	96.4 (26)				**81**	
1e−2	98.6 (16)				**83**	
1e−3	99.5 (7)				**83**	
1e−4	≥ 100				**98**	
1e−5					**98**	
1e−6					**98**	
1e−7					**98**	
1e−8					**98**	
1e−9					**98**	
1e−10					**98**	
1e−11					**98**	
1e−12					**98**	

Table 5.7 – Shekel 5.



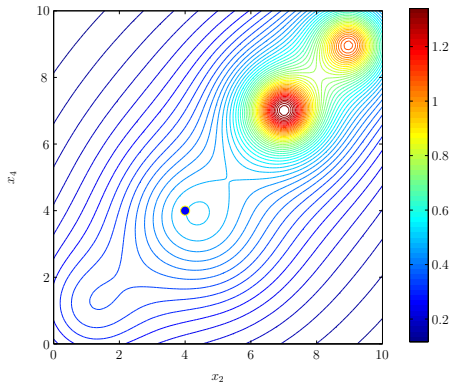
(a) Shekel 7-Coupe 1-2

(b) Shekel 7-Coupe 1-3



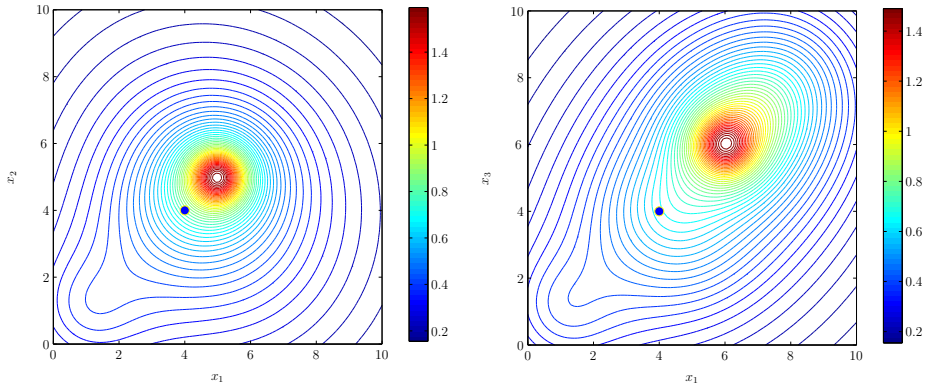
(c) Shekel 7-Coupe 1-4

(d) Shekel 7-Coupe 2-3



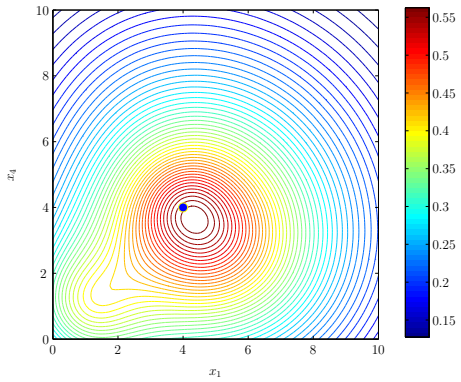
(e) Shekel 7-Coupe 2-4

Figure 5.29 – Fonction Shekel 7

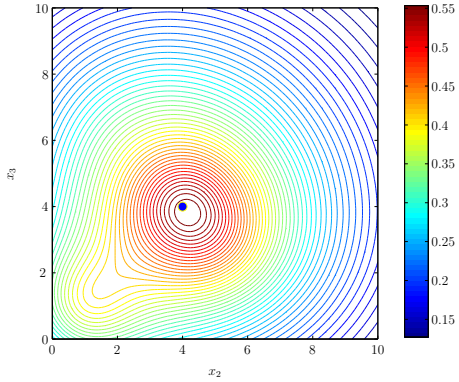


(a) Shekel 10-Coupe 1-2

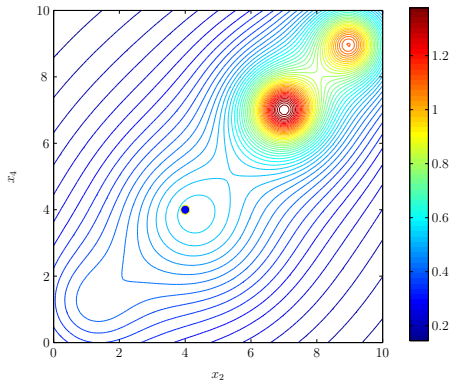
(b) Shekel 10-Coupe 1-3



(c) Shekel 10-Coupe 1-4



(d) Shekel 10-Coupe 2-3



(e) Shekel 10-Coupe 2-4

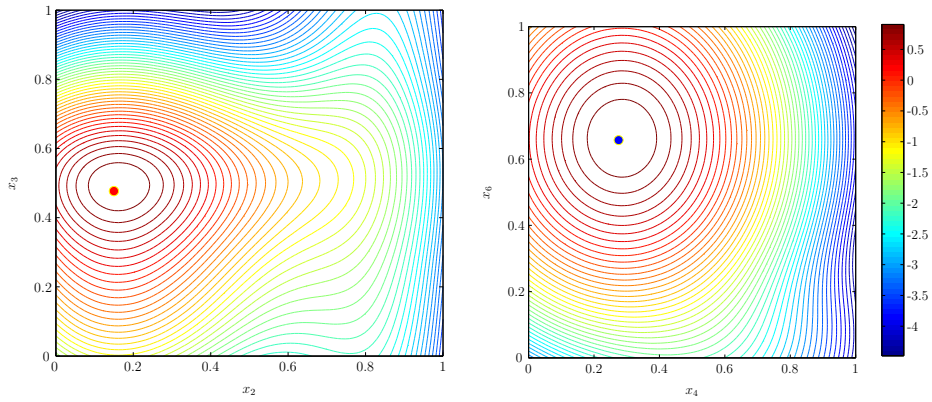
Figure 5.30 – Fonction Shekel 10

Niv. de précision	SMC-EI	EGO ($n_0 = 2d$)	EGO ($n_0 = 10d$)	DIRECT	MCS	rand
1e2	9 (100)	9 (100)	41 (100)	**1**	**1**	**1** (100)
1e1	**9** (100)	99.6 (1)	≥ 100	52	78	100 (0)
1e0	**90.0** (52)	≥ 100		97	≥ 100	≥ 100
1e−1	**95.1** (32)			≥ 100		
1e−2	**98.6** (17)					
1e−3	**99.3** (8)					
1e−4	≥ 100					
1e−5						
1e−6						
1e−7						
1e−8						
1e−9						
1e−10						
1e−11						
1e−12						

Table 5.8 – Shekel 7.

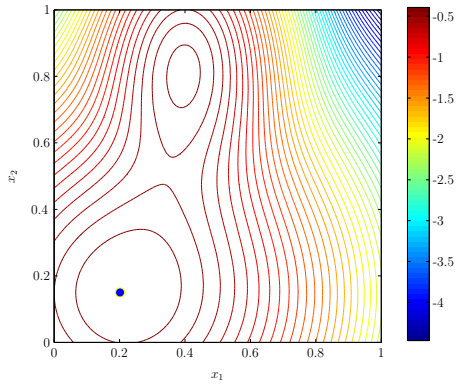
Niv. de précision	SMC-EI	EGO ($n_0 = 2d$)	EGO ($n_0 = 10d$)	DIRECT	MCS	rand
1e2	9 (100)	9 (100)	41 (100)	**1**	**1**	**1** (100)
1e1	**9** (100)	99.1 (3)	≥ 100	52	79	100 (0)
1e0	**88.5** (58)	≥ 100		97	≥ 100	≥ 100
1e−1	**93.9** (37)			≥ 100		
1e−2	**97.6** (21)					
1e−3	**99.6** (5)					
1e−4	≥ 100					
1e−5						
1e−6						
1e−7						
1e−8						
1e−9						
1e−10						
1e−11						
1e−12						

Table 5.9 – Shekel 10.

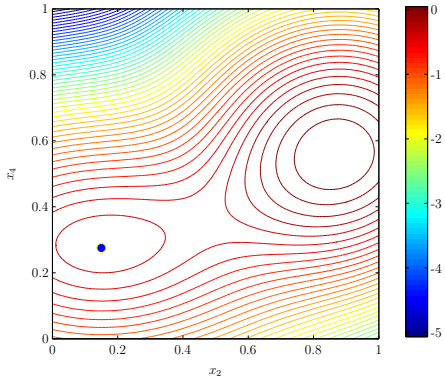


(a) Hart 6 (en log) -Coupe 2-3

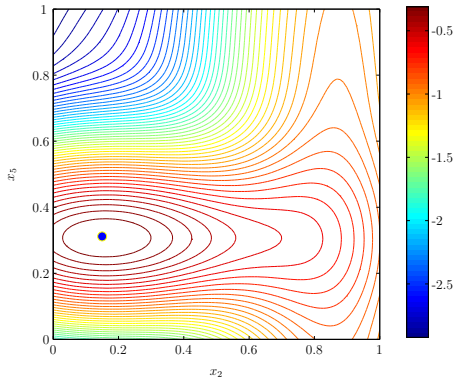
(b) Hart 6-Coupe 4-6



(c) Hart 6 (en log) -Coupe 1-2



(d) Hart 6 (en log)-Coupe 2-4



(e) Hart 6 (en log)-Coupe 2-5

Figure 5.31 – Fonction Hartman 6

Niv. de précision	SMC-EI	EGO ($n_0 = 2d$)	EGO ($n_0 = 10d$)	DIRECT	MCS	rand
1e−1	27.0 (100)	20.7 (100)	61.3 (100)	52	**16**	58.4 (73)
1e−2	**43.4** (100)	**39.9** (100)	75.9 (98)	≥ 100	68	≥ 100
1e−3	90.8 (33)	77.3 (58)	98.6 (17)		**68**	
1e−4	99.9 (3)	98.5 (3)	≥ 100		**77**	
1e−5	≥ 100	≥ 100			**77**	
1e−6					**94**	
1e−7					**94**	
1e−8					**94**	
1e−9					≥ 100	
1e−10						
1e−11						
1e−12						

Table 5.10 – Hartman 6 (log).

Chapitre 6

Conclusions et perspectives

6.1 Résumé et contributions

Les travaux de thèse présentés dans ce manuscrit mènent à la présentation d'un nouvel algorithme, et à l'étude de ses performances. Cet algorithme s'inscrit dans le cadre de l'optimisation bayésienne, ce qui implique l'utilisation d'un *a priori* ξ sur la fonction objectif. Concernant le critère d'échantillonnage, nous faisons le choix du critère EI, dont l'utilisation est majoritaire dans la littérature récente, de par ses qualités discutées au chapitre 2. Notre approche s'inscrit de plus dans un cadre *complètement bayésien*. Par *complètement bayésien*, nous entendons le fait qu'un *a priori* est également affecté au vecteur paramètre $\theta \in \Theta$ de la covariance de ξ [1], choisie au sein d'une famille de covariance paramétrée. L'avantage de cette approche est de prendre en compte l'incertitude inhérente à ces paramètres de covariance plutôt que d'en calculer une estimation (approche par substitution), ce qui apporte un gain de robustesse mis en évidence au chapitre 3. Néanmoins, cet avantage s'accompagne également d'une complexité algorithmique pour le calcul du critère EI. En premier lieu, considérer un *a priori* sur les paramètres plutôt qu'une estimée de ceux-ci revient à ajouter un calcul d'intégrale, au critère EI, sans résolution analytique. En second lieu, l'optimisation séquentielle de f nécessite des maximisations successives du critère sur le domaine $\mathbb{X}$, ce qui peut s'avérer coûteux.

1. Ce qui ajoute une seconde « couche » de bayésien, en plus de l'*a priori* ξ

Notre algorithme apporte au chapitre 4 une réponse conjointe à ces deux points à l'aide d'une approche SMC sur l'espace $\Theta \times \mathbb{X}$. En effet, nous procédons de la sorte pour construire simultanément, à l'étape n, un ensemble de particules sur l'espace des paramètres Θ réparti selon la densité *a posteriori* π_n, et un ensemble de particules de $\mathbb{X}$ réparti selon une densité d'intérêt p_n. Le premier ensemble est utilisé afin de calculer une estimation, sous la forme d'une somme finie, de l'intégrale apparaissant dans l'expression du critère EI. Le second ensemble correspond aux points candidats en lesquels le critère EI est évalué. Un choix pertinent pour la densité p_n permet une optimisation efficace du critère en ne l'évaluant qu'en un nombre de points relativement petit, mais situés dans les zones où le critère est le plus élevé. Le choix que nous faisons pour p_n est de considérer la probabilité d'amélioration normalisée. Ce nouvel algorithme constitue une contribution au domaine. Nous en avons présenté les performances sur des exemples d'applications industrielles et des fonctions tests au chapitre 5. Nous rappelons les conclusions relatives à ces performances à la section suivante.

6.2 Performances

Le caractère complètement bayésien offre plus de robustesse qu'une approche par substitution. Un des avantages de cette robustesse accrue est qu'elle permet d'utiliser un plan d'expériences avec un nombre d'évaluations initiales relativement faible en comparaison de ce qui est généralement d'usage (plan d'expériences initial de taille $2d$ au lieu de $10d$, dans les simulations effectuées). L'utilisation d'une approche SMC pour la maximisation du critère EI, consistant à « déplacer », à chaque itération, l'ensemble des points candidats du domaine $\mathbb{X}$ selon une loi d'intérêt (par exemple, une probabilité d'amélioration normalisée) offre une alternative plus efficace que des ensembles de points candidats sélectionnés sur des grilles ou des LHS fixés. L'approche SMC sur les valeurs de paramètres permet quant à elle de garder une complexité

algorithmique acceptable[2]. Les résultats concernant deux différents types de problème (identification de système dynamique et maximisation du rendement d'un convertisseur de puissance), ainsi que plusieurs *benchmarks* sur des fonctions tests classiques montrent que les performances obtenues sont largement comparables, voir meilleures qu'avec les autres algorithmes de nature bayésienne mais également que d'autres algorithmes d'optimisation de nature plus générale. Ceci semble confirmer l'intérêt dans un contexte général d'optimisation, et non plus seulement bayésien, de notre algorithme ainsi que, de façon plus large, de l'approche bayésienne.

6.3 Perspectives

Il reste de nombreux aspects pouvant faire l'objet d'une étude plus poussée. En premier lieu, le choix de l'*a priori* associé aux paramètres de covariance est particulièrement important. Nous l'avons choisi en fonction de la taille du domaine $\mathbb{X}$, d'une façon nous semblant raisonnable. Une étude plus poussée à ce sujet est effectivement envisageable. Il est également à noter que l'utilisation d'un prédicteur issu de la théorie du krigeage augmente de manière importante le temps d'exécution des algorithmes lorsque la dimension augmente. Des méthodes, comme une analyse de sensibilité, pourraient être envisagées pour donner une complexité raisonnable à ce type d'approche en grandes dimensions. Il serait également envisageable de mettre en place des *benchmarks* plus complets, ainsi que d'étudier, d'un point de vue théorique, les hypothèses nécessaires à la convergence de notre algorithme SMC+EI. Plusieurs autres points, reliés directement à l'algorithme en lui-même pourraient être améliorés. Par exemple, une étude comparative entre plusieurs lois d'intérêt, pour la répartition des points candidats du domaine $\mathbb{X}$, pourrait être faite. Une réflexion sur les lois instrumentales q_n les plus appropriées, et les estimateurs à utiliser serait également de l'ordre du possible.

2. En réalité la complexité algorithmique ne peut être jugée acceptable qu'à l'aune d'une comparaison avec la complexité de la fonction objectif f, l'objectif *in fine* étant bien évidemment un gain de temps.

Annexe A

Processus gaussien

La loi d'un processus gaussien ξ est déterminée uniquement par sa moyenne $m(x) = \mathsf{E}(\xi(x))$, $x \in \mathbb{X}$, et sa fonction de covariance $k(x,y) = \mathsf{E}((\xi(x) - m(x))(\xi(y) - m(y)))$, $(x,y) \in \mathbb{X}^2$. Nous faisons le choix d'une moyenne constante sur $\mathbb{X}$, et nous utilisons la notation $\xi \sim \mathrm{GP}(m,k)$ pour indiquer que le processus ξ est gaussien de moyenne $m(x) = m \in \mathbb{R}$ et de fonction de covariance k.

A.1 Calcul de la prédiction par krigeage

Soit k une fonction de covariance stationnaire pouvant s'écrire sous la forme $k(x,y) = \sigma^2 r(x-y)$, $(x,y) \in \mathbb{X}^2$, avec $\sigma^2 > 0$ et $r(0) = 1$ (r est ainsi une fonction de corrélation). Sous hypothèse que $\xi \mid m \sim \mathrm{GP}(m, k)$ et $m \sim \mathcal{U}(\mathbb{R})$, avec $\mathcal{U}(\mathbb{R})$ est la répartition uniforme (impropre) sur $\mathbb{R}$, alors pour tout $x \in \mathbb{X}$,

$$\xi(x) \mid \mathcal{F}_n \sim \mathcal{N}\left(\widehat{\xi}_n(x),\, s_n^2(x)\right),$$

où

$$\widehat{\xi}_n(x) = \widehat{m}_n + \underline{r}_n(x)^{\mathsf{T}} R_n^{-1}(\underline{\xi}_n - \widehat{m}_n \mathbb{1}_n), \tag{A.1}$$

avec

$$\begin{cases} \underline{\xi}_n = (\xi(X_1), \ldots, \xi(X_n))^\mathsf{T}\,, \text{le vecteur des résultats d'évaluation} \\ \mathbb{1}_n = (1, \ldots, 1)^\mathsf{T} \in \mathbb{R}^n, \\ R_n \text{ la matrice de corrélation de } \underline{\xi}_n, \\ \underline{r}_n(x) \text{ le vecteur de corrélation entre } \xi(x) \text{ et } \underline{\xi}_n, \\ \widehat{m}_n = \frac{\mathbb{1}_n^\mathsf{T} R_n^{-1} \underline{\xi}_n}{\mathbb{1}_n^\mathsf{T} R_n^{-1} \mathbb{1}_n}, \text{ l'estimation par moindres carrés pondérés de } m, \end{cases}$$

et

$$s_n^2(x) = \sigma^2 \kappa_n^2(x)\,, \tag{A.2}$$

avec

$$\kappa_n^2(x) = 1 - \underline{r}_n(x)^\mathsf{T} R_n^{-1} \underline{r}_n(x) + \frac{(1 - \underline{r}_n(x)^\mathsf{T} R_n^{-1} \mathbb{1}_n)^2}{\mathbb{1}_n^\mathsf{T} R_n^{-1} \mathbb{1}_n}\,. \tag{A.3}$$

Tout ceci, ainsi que l'équation (2.5) montrent que lorsqu'un ensemble donné de points d'évaluation est à disposition et que le processus est gaussien, le critère EI peut être calculé avec un temps de calcul relativement faible (le calcul de (A.1) en q différents points de $\mathbb{X}$ demande $O(qn^2 + n^3)$ opérations).

A.2 Fonctions de covariance classiques

Dans la littérature traitant des processus gaussiens utilisés pour la modélisation d'expériences, il y a différentes familles de fonctions de covariance paramétrées. Nous allons nous intéresser aux *covariances exponentielles généralisées* , et aux *covariances de Matérn*. L'utilisation des covariances de Matérn permet d'ajuster la dérivabilité à l'origine de ξ à l'aide d'un seul paramètre, la régularité, ce qui n'est pas le cas de l'autre famille évoquée.

A.2.1 Covariance exponentielle généralisée

Soit $r_\theta : \mathbb{X}^2 \to \mathbb{R}^+$, tel que, $\forall h \geq 0$,

$$r_\theta(x, y) = \sigma^2 \exp\left(- \sum_{i=1}^{d} (||x_{[i]} - y_{[i]}||/\beta_i)^{l_i} \right), \tag{A.4}$$

où le scalaire positif σ^2 correspond au paramètre de variance (nous avons $k_\theta(x,x) = \sigma^2$), $x_{[i]}$, $y_{[i]}$ représentent respectivement la $i^{\text{ème}}$ coordonnée de x et y, les scalaires positifs β_i représente un paramètre d'échelle, ou portée, de la covariance. Le scalaire l_i permet de régler la décroissance de la fonction de covariance dans la direction i.

A.2.2 Covariance de Matérn

Soit $\upsilon_\nu : \mathbb{R}^+ \to \mathbb{R}^+$ tel que, $\forall h \geq 0$,

$$\upsilon_\nu(h) = \frac{1}{2^{\nu-1}\Gamma(\nu)} \left(2\nu^{1/2}h\right)^\nu \mathcal{K}_\nu \left(2\nu^{1/2}h\right) , \tag{A.5}$$

où Γ est la fonction Gamma et $\mathcal{K}_\nu$ est la fonction de Bessel de seconde espèce modifiée d'ordre ν. Le paramètre $\nu > 0$ contrôle la régularité à l'origine de υ_ν.

La forme anisotrope de la covariance de Matérn sur $\mathbb{R}^d$ peut s'écrire comme $k_\theta(x,y) = \sigma^2 r_\theta(x,y)$, avec

$$r_\theta(x,y) = \upsilon_\nu \left(\sqrt{\sum_{i=1}^{d} \frac{(x_{[i]} - y_{[i]})^2}{\beta_i^2}} \right) , \quad x, y \in \mathbb{R}^d , \tag{A.6}$$

où le scalaire positif σ^2 correspond au paramètre de variance (nous avons $k_\theta(x,x) = \sigma^2$), $x_{[i]}$, $y_{[i]}$ représentent respectivement la $i^{\text{ème}}$ coordonnée de x et y, les scalaires positifs β_i représente un paramètre d'échelle, ou portée, de la covariance. Ces paramètres β_i correspondent à la taille caractéristique de la corrélation, ce qui nous donne finalement $\theta = (\nu, \beta_1, \ldots, \beta_d) \in \mathbb{R}_+^{d+1}$ le vecteur des paramètres de la covariance de Matérn. Il peut être observé que la forme isotrope de la covariance de Matérn est obtenue en prenant $\beta_1 = \ldots = \beta_d = \beta$. Le vecteur des paramètres est ensuite simplement $\theta = (\nu, \beta) \in \mathbb{R}_+^2$.

Annexe B

Lois de probabilité utiles

B.1 Loi inverse-gamma

Il s'agit de la loi de l'inverse d'une variable aléatoire distribuée selon une loi gamma. Deux paramètres réels strictement positifs a et b sont considérés. La densité de probabilité associée est alors définie, pour $z > 0$, par

$$g_{a,b}(z) = \frac{b^a}{\Gamma(a)}(1/z)^{a+1}\exp(-b/z). \tag{B.1}$$

Les paramètres a et b sont respectivement nommés paramètre de forme (*shape*) et paramètre d'échelle (*scale*). Nous notons cette loi $\mathrm{IG}(a, b)$.

B.2 Loi de Student multivariée

La loi de Student à ν degrés de liberté est considérée ici. Les paramètres, pour une dimension $n \in \mathbb{N}^*$, sont un vecteur de position $m \in \mathbb{R}^n$, et une matrice de dispersion $K \in \mathcal{S}_n^+(\mathbb{R})$. La densité de probabilité vaut ainsi, pour un vecteur $z \in \mathbb{R}^n$,

$$f_{\nu,m,K}(z) = \left(\frac{1}{\pi\nu}\right)^{n/2} \frac{\Gamma\left(\frac{\nu+n}{2}\right)}{|K|^{1/2}\Gamma(\frac{\nu}{2})} \left(1 + \frac{(z-m)^T K^{-1}(z-m)}{\nu}\right)^{-\frac{\nu+n}{2}}. \tag{B.2}$$

Cette loi est notée $t_{\nu,m,K}$ et, en prenant $n = 1$, $m = 0$ et $K = 1$, elle correspond à la loi de Student t_ν classique à une dimension pour $z \in \mathbb{R}$,

$$f_\nu(z) = \frac{1}{\sqrt{\pi\nu}} \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\Gamma\left(\frac{\nu}{2}\right)} \left(1 + \frac{z^2}{\nu}\right)^{-\frac{\nu+1}{2}}. \tag{B.3}$$

B.3 Loi log-normale

La loi log-normale dépend de deux paramètres $\mu \in \mathbb{R}$ et $\sigma^2 > 0$, et nous la notons $\ln\mathcal{N}(\mu, \sigma^2)$. La densité de probabilité associée, pour $z > 0$, vaut

$$h_{\mu,\sigma^2}(z) = \frac{1}{z\sigma\sqrt{2\pi}} \exp\left(-\frac{(\ln(z) - \mu)^2}{2\sigma^2}\right). \tag{B.4}$$

Si la variable aléatoire Z suit la loi $\ln\mathcal{N}(\mu, \sigma^2)$ alors $Y = \ln(Z)$ suit la loi normale $\mathcal{N}(\mu, \sigma^2)$, ce qui justifie le nom, ainsi que la notation adoptée ici, de la loi log-normale.

Annexe C

Expressions des fonctions tests

Les expressions des fonctions test utilisées aux chapitres 3 et 5 sont données ci-dessous, la plupart étant issues de (Dixon et Szego, 1978). Cette thèse s'inscrit essentiellement dans un cadre de maximisation, le signe de ces fonctions tests est donc choisi en conséquence. Les valeurs des maxima et maximiseurs données sont généralement tronquées après plusieurs chiffres significatifs.

C.1 Branin

L'expression de cette fonction de dimension 2 est

$$\begin{aligned} f: \quad [-5,10]\times[0,15] \quad &\to \quad \mathbb{R} \\ x=(x_1,x_2) \quad &\mapsto \quad -(x_2-\tfrac{5.1}{4\pi^2}\,x_1^2+\tfrac{5}{\pi}\,x_1-6)^2-10(1-\tfrac{1}{8\pi})\cos(x_1)-10. \end{aligned}$$

Il y a trois maximiseurs, contenus dans l'ensemble $\{(-\pi, 12.275), (\pi, 2.275), (9.42478, 2.475)\}$, et la valeur du maximum est de -0.387887.

C.2 Goldstein & Price

L'expression de cette fonction de dimension 2 est

$$\begin{array}{rccl} f: & [-2,2]^2 & \to & \mathbb{R} \\ & x=(x_1,x_2) & \mapsto & -(1+(x_1+x_2+1)^2(19-14x_1+3x_1^2-14x_2+6x_1x_2+3x_2^2)) \\ & & & (30+(2x_1-3x_2)^2(18-32x_1+12x_1^2+48x_2-36x_1x_2+27x_2^2)). \end{array}$$

Le maximum vaut -3 et est atteint au point $(0,-1)$. En pratique, c'est le logarithme (décimal) de cette fonction qui est considéré dans le manuscrit.

C.3 Camel back

L'expression de cette fonction de dimension 2 est

$$\begin{array}{rccl} f: & [-3,3]\times[-2,2] & \to & \mathbb{R} \\ & x=(x_1,x_2) & \mapsto & -(4-2.1x_1^2+x_1^4/3)x_1^2-x_1x_2-(-4+4x_2^2)x_2^2. \end{array}$$

Il y a deux maximiseurs, à savoir $(-0.08984201, 0.71265640)$ et $(0.08984201, -0.71265640)$. La valeur du maximum est de 1.031628453488552.

C.4 Shubert

L'expression de cette fonction de dimension 2 est

$$\begin{array}{rccl} f: & [-10,10]^2 & \to & \mathbb{R} \\ & x=(x_1,x_2) & \mapsto & -(\sum_{i=1}^5 i\cos((i+1)x_1+i))(\sum_{j=1}^5 j\cos((j+1)\,x_2+j)). \end{array}$$

Il y a 18 maximiseurs et la valeur du maximum est de 186.7309.

C.5 Hartman 3

L'expression de cette fonction de dimension 3 est

$$\begin{array}{rccl} f: & [0,1]^3 & \to & \mathbb{R} \\ & x=(x_1,x_2,x_3) & \mapsto & \sum_{i=0}^{3} c_i \exp(-\sum_{j=0}^{2} A_{i,j}(x_j-p_{i,j})^2)). \end{array}$$

avec

$$A = \begin{pmatrix} 3 & 10 & 30 \\ 0.1 & 10 & 35 \\ 3 & 10 & 30 \\ 0.1 & 10 & 35 \end{pmatrix}, \quad p = \begin{pmatrix} 0.3689 & 0.1170 & 0.2673 \\ 0.4699 & 0.4387 & 0.7470 \\ 0.1091 & 0.8732 & 0.5547 \\ 0.03815 & 0.5743 & 0.8828 \end{pmatrix}$$

et $c = (1, 1.2, 3, 3.2)^T$.

Le maximiseur est $(0.114614, 0.555649, 0.852547)$ et le maximum vaut 3.86278. En pratique, c'est le logarithme (décimal) de cette fonction qui est considéré dans le manuscrit.

C.6 Hartman 6

L'expression de cette fonction de dimension 6 est

$$\begin{array}{rccl} f: & [0,1]^6 & \to & \mathbb{R} \\ & x=(x_1,x_2,x_3) & \mapsto & \sum_{i=0}^{3} c_i \exp(-\sum_{j=0}^{5} A_{i,j}(x_j-p_{i,j})^2)), \end{array}$$

avec

$$A = \begin{pmatrix} 10 & 3 & 17 & 3.5 & 1.7 & 8 \\ 0.05 & 10 & 17 & 0.1 & 8 & 14 \\ 3 & 3.5 & 1.7 & 10 & 17 & 8 \\ 17 & 8 & 0.05 & 10 & 0.1 & 14 \end{pmatrix},$$

$$p = \begin{pmatrix} 0.1312 & 0.696 & 0.5569 & 0.0124 & 0.8283 & 0.5886 \\ 0.2329 & 0.4135 & 0.8307 & 0.3736 & 0.1004 & 0.9991 \\ 0.2348 & 0.1451 & 0.3522 & 0.2883 & 0.3047 & 0.6650 \\ 0.4047 & 0.8828 & 0.8732 & 0.5743 & 0.1091 & 0.0381 \end{pmatrix}$$

et $c = (1, 1.2, 3, 3.2)^T$.

Le maximiseur est $(0.20169, 0.150011, 0.476874, 0.275332, 0.311652, 0.6573)$ et le maximum vaut 3.32237. En pratique, c'est le logarithme (décimal) de cette fonction qui est considéré dans le manuscrit.

C.7 Shekel

Il s'agit d'une fonction de dimension 4, dépendant également d'un paramètre m que nous considérons dans l'ensemble $\{5, 7, 10\}$.

$$\begin{array}{rccl} f_m : & [0,10]^4 & \to & \mathbb{R} \\ & x & \mapsto & \sum_{i=1}^{m} \frac{1}{(x-A_i)^T(x-A_i)+c_i}, \end{array}$$

avec A_i la i-ième colonne de la matrice

$$A = \begin{pmatrix} 4 & 1 & 8 & 6 & 3 & 2 & 5 & 8 & 6 & 7 \\ 4 & 1 & 8 & 6 & 7 & 9 & 5 & 1 & 2 & 3.6 \\ 4 & 1 & 8 & 6 & 3 & 2 & 3 & 8 & 6 & 7 \\ 4 & 1 & 8 & 6 & 7 & 9 & 3 & 1 & 2 & 3.6 \end{pmatrix}$$

et $c = (0.1, 0.2, 0.2, 0.4, 0.4, 0.6, 0.3, 0.7, 0.5, 0.5)^T$.

Le maximiseur est $(4, 4, 4, 4)$, et le maximum, pour m égal à 5, 7 et 10, vaut respectivement 10.1532, 10.4029 et 10.5364.

C.8 Hyper-sphère

Nous considérons ici la fonction associée à l'hyper-sphère centrée en zéro. Elle est définie pour toute dimension d, et son expression est

$$\begin{array}{rccl} f : & [-5.12, 5.12]^d & \to & \mathbb{R} \\ & x = (x_1, x_2, \ldots, x_d) & \mapsto & -\sum_{i=1}^{d} x_i^2. \end{array}$$

Le maximum est situé au milieu du domaine de définition et vaut 0. Dans le manuscrit, nous l'utilisons pour une dimension d égale à 4.

Annexe D

Preuves

Nous donnons ici les démonstrations des résultats énoncés dans le manuscrit. Nous décidons de noter $p(.)$ la densité lorsqu'aucune évaluation n'est connue, et $p_n(.)$ la densité *a posteriori* $p(.|\mathcal{F}_n)$.

D.1 Maximisation de la vraisemblance

Démonstration. Nous voulons montrer que les valeurs $\widehat{m}_n(\theta)$ et $\widehat{\sigma}_n^2(\theta)$ correspondent au maximum de vraisemblance, autrement dit qu'elles annulent les dérivées partielles de ℓ_n par rapport, respectivement, à m et σ^2. Pour plus de concision, nous écrivons de façon légèrement abusive dans la suite $\ell_n(m)$ ou $\ell_n(\sigma^2)$ (en fonction de la dépendance considérée) au lieu de $\ell_n(\underline{\xi}_n; m, \sigma^2, \theta)$, et nous omettons généralement de faire aparaître explicitement la dépendance en θ pour les valeurs $\widehat{m}_n(\theta)$ et $\widehat{\sigma}_n^2(\theta)$.

Dans un premier temps, nous nous intéressons à la dérivée partielle de ℓ_n par rapport à m. Ceci nous permet de ne considérer ℓ_n qu'à une constante (indépendante de m) près,

$$\ell_n(m) \propto e^{-\frac{1}{2\sigma^2}Q_n(m)}, \tag{D.1}$$

où

$$\begin{aligned} Q_n(m) &= \left(\underline{\xi}_n - m\mathbb{1}_n\right)^{\mathsf{T}} R_n(\theta)^{-1} \left(\underline{\xi}_n - m\mathbb{1}_n\right) \\ &= m^2 \mathbb{1}_n^{\mathsf{T}} R_n(\theta)^{-1} \mathbb{1}_n - 2m\mathbb{1}_n^{\mathsf{T}} R_n(\theta)^{-1} \underline{\xi}_n + \underline{\xi}_n^{\mathsf{T}} R_n(\theta)^{-1} \underline{\xi}_n. \end{aligned} \tag{D.2}$$

Ceci nous donne

$$\frac{\partial \ell_n}{\partial m}(m) \propto Q_n'(m)\, e^{-\frac{1}{2\sigma^2}Q_n(m)}, \tag{D.3}$$

avec

$$Q_n'(m) = 2m\mathbb{1}_n^\mathsf{T} R_n(\theta)^{-1}\mathbb{1}_n - 2\,\mathbb{1}_n^\mathsf{T} R_n(\theta)^{-1}\underline{\xi}_n. \tag{D.4}$$

La valeur $\widehat{m}_n$ annule la dérivée partielle (voir l'équation (D.3)) si et seulement si

$$\widehat{m}_n(\theta) = \frac{\mathbb{1}_n^\mathsf{T} R_n(\theta)^{-1}\underline{\xi}_n}{\mathbb{1}_n^\mathsf{T} R_n(\theta)^{-1}\mathbb{1}_n}, \tag{D.5}$$

ce qui correspond bien à l'équation (3.2).

Il nous reste à déterminer la valeur de σ^2 qui annule la dérivée partielle de ℓ_n par rapport à σ^2. En ne considérant la vraisemblance qu'à une constante (indépendante de σ^2) près,

$$\ell_n(\sigma^2) \propto \frac{1}{(\sigma^2)^{n/2}} e^{-\frac{1}{2\sigma^2}Q_n(\widehat{m}_n)}, \tag{D.6}$$

ce qui nous donne

$$\begin{aligned}
\frac{\partial \ell_n}{\partial \sigma^2}(\sigma^2) &\propto -\frac{n}{2(\sigma^2)^{n/2+1}} e^{-\frac{1}{2\sigma^2}Q_n(\widehat{m}_n)} + \frac{Q_n(\widehat{m}_n)}{2(\sigma^2)^{n/2+2}} e^{-\frac{1}{2\sigma^2}Q_n(\widehat{m}_n)} && \text{(D.7)}\\
&\propto \frac{1}{2(\sigma^2)^{n/2+1}} e^{-\frac{1}{2\sigma^2}Q_n(\widehat{m}_n)} \left(-n + \frac{Q_n(\widehat{m}_n)}{\widehat{\sigma}_n^2}\right). && \text{(D.8)}
\end{aligned}$$

La valeur $\widehat{\sigma}_n^2$ annule donc la dérivée partielle si et seulement si

$$\widehat{\sigma}_n^2 = \frac{1}{n} Q_n(\widehat{m}_n). \tag{D.9}$$

En remplaçant $Q_n(\widehat{m}_n)$ par son expression, nous retrouvons l'équation (3.3),

$$\widehat{\sigma}_n^2(\theta) = \frac{1}{n}\left(\underline{\xi}_n - \widehat{m}_n\mathbb{1}_n\right)^\mathsf{T} R_n(\theta)^{-1}\left(\underline{\xi}_n - \widehat{m}_n\mathbb{1}_n\right). \tag{D.10}$$

□

D.2 Proposition 1

Démonstration. La densité conditionnelle $\sigma^2 \mid \mathcal{F}_n$ peut s'écrire sous la forme

$$p_n(\sigma^2) = \int_{\mathbb{R}} p_n(\sigma^2 \mid m)\, p_n(m)\, dm, \tag{D.11}$$

où $p_n(m)$ désigne la densité *a posteriori* de m telle que

$$p_n(m) \propto p(\underline{\xi}_n | m)\, p(m). \tag{D.12}$$

Le théorème de Bayes nous donne également,

$$p_n(\sigma^2 \mid m) = \frac{p(\underline{\xi}_n \mid \sigma^2, m)\, p(\sigma^2)}{p(\underline{\xi}_n \mid m)}. \tag{D.13}$$

Nous en déduisons de ces deux relations que

$$p_n(\sigma^2) \;\propto\; p(\sigma^2) \int_{\mathbb{R}} p(\underline{\xi}_n \mid \sigma^2, m)\, p(m)\, dm. \tag{D.14}$$

Le processus $\underline{\xi}_n \mid \sigma^2, m$ suit, par construction, une loi $\mathcal{N}(m, \sigma^2 R_n)$. Ceci permet d'écrire

$$p(\underline{\xi}_n \mid \sigma^2, m) \propto \frac{1}{(\sigma^2)^{n/2+a+1}}\, e^{-\frac{1}{2\sigma^2} Q_n(m)}, \tag{D.15}$$

avec

$$Q_n(m) = \left(\underline{\xi}_n - m\mathbb{1}\right)^T R_n^{-1} \left(\underline{\xi}_n - m\mathbb{1}\right). \tag{D.16}$$

Nous remarquons que $Q_n(m)$ peut se réécrire de la façon suivante,

$$Q_n(m) = \mathbb{1}^T R_n^{-1} \mathbb{1}\, (m - \widehat{m}_n)^2 + Q_n(\widehat{m}_n),$$

où $\widehat{m}_n$ est le même qu'à l'équation (D.5). Puisque la densité *a priori* de m est uniforme, nous reconnaissons donc une densité gaussienne en m sous l'intégrale de l'équation (D.14), ce qui permet de calculer cette intégrale. La loi associée à $p(\sigma^2)$ est, par hypothèse, une inverse gamma $\mathrm{IG}\,(a_0, b_0)$ ce qui permet d'écrire la densité conditionnelle $p_n(\sigma^2)$, à une constante indépendante de σ^2 près

$$\frac{1}{(\sigma^2)^{a_0+(n+1)/2}} \exp\left(-\frac{b_0 + Q_n(\widehat{m}_n)/2}{\sigma^2}\right).$$

Pour

$$\begin{aligned} a_n &= a_0 + \frac{n-1}{2}, \\ b_n &= b_0 + Q_n(\widehat{m}_n)/2, \end{aligned}$$

la densité recherchée est proportionnelle à

$$\frac{1}{(\sigma^2)^{a_n+1}} \exp\left(-\frac{b_n}{\sigma^2}\right).$$

En remplaçant $Q_n(\widehat{m}_n)$ par son expression, nous reconnaissons bien la densité associée à la loi IG (a_n, b_n). □

D.3 Proposition 2

Afin de démontrer la proposition 2 nous donnons, et démontrons, au préalable deux lemmes que nous utiliserons au cours de la démonstration de la proposition.

D.3.1 Lemmes préalables et remarque

Lemme 2. *Pour $n \in \mathbb{N}^*$, si $Z \sim t_{\nu,m,K}$ et $T = AZ + c$, avec $A \in \mathcal{M}_n(\mathbb{R})$ inversible et $c \in \mathbb{R}^n$, alors $T \sim t_{\nu,Am+c,AKA^T}$.*

Démonstration. Notons p la densité en T. Le changement de variable est affine ce qui nous donne

$$\begin{aligned} p(t) &= f_{\nu,m,K}(A^{-1}(t-c))/|A| \\ &= \left(\frac{1}{\pi\nu}\right)^{n/2} \frac{\Gamma\left(\frac{\nu+n}{2}\right)}{|K|^{1/2}|A|\,\Gamma(\frac{\nu}{2})} \left(1 + \frac{(A^{-1}(t-c)-m)^T K^{-1}(A^{-1}(t-b)-m)}{\nu}\right)^{-\frac{\nu+n}{2}} \\ &= \left(\frac{1}{\pi\nu}\right)^{n/2} \frac{\Gamma\left(\frac{\nu+n}{2}\right)}{|A^T K A|^{1/2}\Gamma(\frac{\nu}{2})} \left(1 + \frac{(t-(Am+c))^T (AKAT)^{-1}(t-(Am+c))}{\nu}\right)^{-\frac{\nu+n}{2}} \\ &= f_{\nu,Am+c,AKA^T}(t). \end{aligned}$$

□

Remarque 1. *Si $Z \sim t_{\nu,m,K}$ alors le vecteur*

$$T = K^{-1/2}(Z - m)$$

suit une loi de Student (multivariée) à ν degrés de liberté. La densité de probabilité de T est ainsi

$$p(t) = \left(\frac{1}{\pi\nu}\right)^{n/2} \frac{\Gamma\left(\frac{\nu+n}{2}\right)}{\Gamma\left(\frac{\nu}{2}\right)} \frac{1}{\left(1 + \frac{||t||^2}{\nu}\right)^{\frac{\nu+n}{2}}}. \tag{D.17}$$

Dans le cas $n = 1$, nous retrouvons la loi de Student usuelle.

Lemme 3. *Soient $n \in \mathbb{N}^*$, $m \in \mathbb{R}^n$, R une matrice définie positive, et (a, b) un couple de réels positifs. Si $Z \mid \sigma^2 \sim \mathcal{N}(m, \sigma^2 R)$ avec $\sigma^2 \sim \mathrm{IG}(a, b)$, alors Z suit une loi de Student multivariée à $\nu = 2a$ degré de liberté, de paramètre m et $K = \frac{b}{a}R$.*

Démonstration. Pour prouver ce résultat, nous pouvons remarquer que

$$p(z) = \int_0^{+\infty} p(z \mid \sigma^2)\, p(\sigma^2) \mathrm{d}\sigma^2. \tag{D.18}$$

Or d'après les hypothèses,

$$p(z \mid \sigma^2) = \frac{1}{(2\pi\sigma^2)^{n/2}\sqrt{\det R}}\, e^{-\frac{(z-m)^T R^{-1}(z-m)}{2\sigma^2}}, \tag{D.19}$$

et

$$p(\sigma^2) = \frac{b^a}{\Gamma(a)} \left(1/\sigma^2\right)^{a+1} e^{-b/\sigma^2}, \tag{D.20}$$

ce qui nous donne ainsi, avec $b'(z) = b\left(1 + \frac{1}{2b}(z-m)^T R^{-1}(z-m)\right)$

$$p(z) \propto \int_0^{\infty} \frac{1}{(\sigma^2)^{n/2+a+1}} e^{-b'(z)/\sigma^2} \mathrm{d}\sigma^2. \tag{D.21}$$

En effectuant le changement de variable $t = b'(z)/\sigma^2$ on obtient une nouvelle expression

$$p(z) \propto \left(1 + \frac{1}{2b}(z-m)^T R^{-1}(z-m)\right)^{-\frac{2a+n}{2}} \underbrace{\int_0^{+\infty} t^{n/2+a-1} e^{-t} \mathrm{d}t}_{=\Gamma(a+n/2)}, \tag{D.22}$$

ceci nous permet de reconnaitre la densité voulue. □

D.3.2 Démonstration de la proposition

Démonstration. Nous savons déjà, d'après la théorie du krigeage (voir par exemple la section A.1), que $\xi(x) \mid \mathcal{F}_n \sim \mathcal{N}\left(\widehat{\xi}_n(x), \sigma^2\kappa_n^2(x)\right)$. La proposition 1 nous indique quant à elle que $\sigma^2 \mid \mathcal{F}_n \sim \mathrm{IG}(a_n, b_n)$. Le lemme 3 implique alors que $\xi(x) \mid \mathcal{F}_n$ suit une loi de Student multivariée $t_{\eta_n, \widehat{\xi}_n(x), \gamma_n^2(x)}$ (les expressions explicites des paramètres η_n et $\gamma_n^2(x)$ sont données dans la proposition). Le lemme 2 et plus directement la remarque 1, donne directement

$$\frac{\xi(x) - \widehat{\xi}_n(x)}{\gamma_n(x)} \mid \mathcal{F}_n \sim t_{\eta_n},$$

ce qui est bien le résultat voulu. □

D.4 Lemme 1

Démonstration. Notons f_η la densité associée à la loi de student t_η,

$$\begin{aligned}\mathbb{E}\left((T+u)_+\right) &= \int_{-\infty}^{+\infty} (t+u)_+ \, f_\eta(t) \, \mathrm{d}t \\ &= \int_{-u}^{+\infty} (t+u) \, f_\eta(t) \, \mathrm{d}t.\end{aligned}$$

Nous savons que

$$(t+u) \, f_\eta(t) \sim_t 1/t^\eta,$$

ce qui implique pour $\eta \leq 1$, que l'intégrale considérée est divergente, et que sa valeur vaut $+\infty$.

Intéressons nous maintenant au cas où $1 < \eta \leq +\infty$. Nous avons

$$\begin{aligned}\mathbb{E}\left((T+u)_+\right) &= \int_{-u}^{+\infty} (t+u) \, f_\eta(t) \, \mathrm{d}t \\ &= \int_{-u}^{+\infty} t \, f_\eta(t) \, \mathrm{d}t + \int_{-u}^{+\infty} u \, f_\eta(t) . diff t \\ &= \underbrace{\int_{-u}^{+\infty} t \, f_\eta(t) \, \mathrm{d}t}_{A} + \underbrace{u \int_{-u}^{+\infty} f_\eta(t) \, \mathrm{d}t}_{B}.\end{aligned}$$

Il suffit de calculer les deux termes A et B, puis de les additionner. Le second nous donne

$$B = u\,[F_\eta]_{-u}^{+\infty} = u\,(1 - F_\eta(-u)) = u\,F_\eta(u). \tag{D.23}$$

Puisque $1 < \eta < +\infty$, A peut se calculer en remarquant que la loi f_η de t_η s'écrit sous la forme $\lambda(1+t^2/\eta)^{-\frac{\eta+1}{2}}$ où λ est une constante indépendante de t.

$$\begin{aligned}
A &= \lambda \int_{-u}^{+\infty} t\left(1+\frac{t^2}{\eta}\right)^{-\frac{\eta+1}{2}} \mathrm{d}t \\
&= \left[\frac{\eta}{2\left(1-\frac{\eta+1}{2}\right)}\,\lambda\left(1+\frac{t^2}{\eta}\right)^{1-\frac{\eta+1}{2}}\right]_{-u}^{+\infty} \\
&= \left[\frac{\eta}{1-\eta}\left(1+\frac{t^2}{\eta}\right) f_\eta(t)\right]_{-u}^{+\infty} \\
&= \left[\frac{\eta+t^2}{1-\eta}\, f_\eta(t)\right]_{-u}^{+\infty} \\
&= \frac{\eta+u^2}{\eta-1} f_\eta(u).
\end{aligned}$$

Il suffit alors, pour obtenir le résultat voulu, de sommer A et B.

□

D.5 Calcul de la loi *a posteriori* π_n

Dans un cadre complètement bayésien, le calcul de la loi *a posteriori* $\pi_n(\theta)$ revêt une importance toute particulière pour le calcul du critère EI (voir (3.4)). Une expression analytique de π_n est connue à une constante près, mais échantillonner selon cette loi n'est pas trivial. Faire l'usage de méthodes SMC, afin d'obtenir un ensemble de points (ou un *jeu de particules* pour reprendre la terminologie inhérente à ce type de méthodes), apparaît alors comme un expédient efficace pour l'échanillonnage. Néanmoins, il est bien évidemment nécessaire de calculer la valeur de π_n ponctuellement afin d'obtenir cet ensemble de points. L'expression de π_n dépend naturellement du choix des *a priori* sur les paramètre de ξ. Dans l'ensemble du manuscrit, un *a priori* inverse gamme IG (a, b)

pour la variance σ^2, et un *a priori* impropre uniforme sur la moyenne m sont choisis. Écrivons la densité *a posteriori* $\pi_n(\theta)$, sous la forme de l'intégrale de la densité marginale en la moyenne m, et utilisons le théorème de Bayes sur cette loi

$$\begin{aligned}
\pi_n(\theta) &= \int_m p\left(\theta, m \mid \underline{\xi}_n\right) \mathrm{d}m \\
&\propto \int_m p\left(\underline{\xi}_n \mid \theta, m\right) p(\theta, m)\ \mathrm{d}m \\
&\propto \int_m p\left(\underline{\xi}_n \mid \theta, m\right) \pi_0(\theta)\ \mathrm{d}m \\
&\propto \pi_0(\theta) \int_m p\left(\underline{\xi}_n \mid \theta, m\right)\ \mathrm{d}m.
\end{aligned}$$

Déterminer la valeur de $\pi_n(\theta)$ revient à calculer l'intégrale $\int_m p\left(\underline{\xi}_n \mid \theta, m\right) \mathrm{d}m$, $\pi_0(\theta)$ correspondant à l'*a priori* choisi, et donc connu, de l'utilisateur. Nous avons alors, en considérant cette fois-ci l'intégrale d'une marginale en σ^2

$$p\left(\underline{\xi}_n \mid \theta, m\right) = \int_{\sigma^2} p\left(\underline{\xi}_n \mid \theta, m, \sigma^2\right) p(\sigma^2)\ \mathrm{d}\sigma^2,$$

ce qui nous permet d'écrire

$$\begin{aligned}
\int_m p\left(\underline{\xi}_n \mid \theta, m\right) \mathrm{d}m &= \int_m \int_{\sigma^2} p\left(\underline{\xi}_n \mid \theta, m, \sigma^2\right) p(\sigma^2)\ \mathrm{d}\sigma^2\ \mathrm{d}m \\
&= \int_{\sigma^2} p(\sigma^2) \int_m p\left(\underline{\xi}_n \mid \theta, m, \sigma^2\right)\ \mathrm{d}m\ \mathrm{d}\sigma^2.
\end{aligned}$$

Le calcul peut se mener en deux étapes. En premier lieu, le calcul de

$$\int_m p\left(\underline{\xi}_n \mid \theta, m, \sigma^2\right) \mathrm{d}m$$

et, en second lieu, celui de l'intégrale en σ^2. Détaillons cette première étape,

$$\int_m p\left(\underline{\xi}_n \mid \theta, m, \sigma^2\right) \mathrm{d}m = \frac{1}{((2\pi\ \sigma^2)^{n/2} |R_n(\theta)|^{1/2})} \int_m \exp\left(-\frac{Q_n(m,\theta)}{2\sigma^2}\right) \mathrm{d}m, \tag{D.24}$$

où $Q_n(m,\theta)$ a la même expression qu'aux (D.2) et (D.16), avec seulement la dépendance en θ écrite explicitement. Nous avons donc à nouveau

$$Q_n(m,\theta) = \mathbb{1}^T R_n^{-1}(\theta) \mathbb{1}\ (m - \widehat{m}_n)^2 + Q_n(\widehat{m}_n, \theta).$$

Après un simple changement de variable (la variable après changement est toujours notée m, il s'agit simplement d'une translation), une densité gaussienne apparaît au sein de l'intégrale en m de l'équation (D.24). Ceci nous donne, pour l'intégrale

$$\int_m p\left(\underline{\xi}_n \mid \theta, m, \sigma^2\right) \mathrm{d}m,$$

l'expression suivante

$$\frac{1}{((2\pi\ \sigma^2)^{n/2}|R_n(\theta)|^{1/2})} \exp\left(-\frac{Q_n(\widehat{m}_n, \theta)}{2\sigma^2}\right) \frac{\sigma\sqrt{\pi}}{\sqrt{\mathbb{1}^T R_n^{-1}(\theta)\mathbb{1}}}.$$

La première étape étant finie, il suffit de calculer l'intégrale en σ^2 afin de déterminer la valeur de $\int_m p\left(\underline{\xi}_n \mid \theta, m\right)\ \mathrm{d}m$. Il est important de remarquer que les facteurs ne dépendant ni de σ^2 ni de θ sont inutiles aux étapes de calculs, il n'est alors pas nécessaire de les écrire explicitement. À une constante près, la valeur de cette intégrale est donc

$$\frac{1}{|R_n^{-1}(\theta)|^{1/2}\sqrt{\mathbb{1}^T R_n^{-1}(\theta)\mathbb{1}}} \int_{\sigma^2} \frac{p(\sigma^2)}{(\sigma^2)^{(n-1)/2}} \exp\left(-\frac{Q_m(\widehat{m}_n, \theta)}{2\sigma^2}\right) \mathrm{d}\sigma^2,$$

ce qui donne, en remplaçant $p(\sigma^2)$ par son expression,

$$\frac{1}{|R_n^{-1}(\theta)|^{1/2}\sqrt{\mathbb{1}^T R_n^{-1}(\theta)\mathbb{1}}} \int_{\sigma^2} \frac{1}{(\sigma^2)^{a_n}} \exp\left(-\frac{b_n(\theta)}{\sigma^2}\right) \mathrm{d}\sigma^2,$$

avec $b_n(\theta) = b_0 + Q_n(\widehat{m}_n, \theta)/2$, et $a_n = a_0 + (n-1)/2$. Une façon de calculer l'intégrale est de remarquer que l'intégrande correspond à une loi IG $(a_n, b_n(\theta))$, d'où

$$\int_m p(\underline{\xi}_n \mid \theta, m) \propto \frac{b_n(\theta)^{-a_n}}{|R_n^{-1}(\theta)|^{1/2}\sqrt{\mathbb{1}^T R_n^{-1}(\theta)\mathbb{1}}}.$$

Il est désormais possible de donner la valeur, à une constante près, de la loi *a posteriori*

$$\pi_n(\theta) \propto \pi_0(\theta) \frac{b_n(\theta)^{-a_n}}{|R_n^{-1}(\theta)|^{1/2}\sqrt{\mathbb{1}^T R_n^{-1}(\theta)\mathbb{1}}}.$$

Bibliographie

P. Auer, N. Cesa-Bianchi et P. Fischer. « Finite time analysis of the multiarmed bandit problem ». *Machine Learning*, 47(2-3) : 235–256, 2002.

A. Auger et O. Teytaud. « Continuous lunches are free plus the design of optimal optimization algorithms ». *Algorithmica*, 57(1) : 121–146, 2008.

R. Bardenet et B. Kégl. « Surrogating the surrogate : accelerating Gaussian-process-based global optimization with a mixture cross-entropy algorithm ». In *Proceedings of the 27th International Conference on Machine Learning (ICML 2010)*, 2010.

R. A. Bates et L. Pronzato. « Emulator-based global optimisation using lattices and delaunay tesselation ». In *International symposium on sensitivity analysis of model output*, pp. 215–218, 2001.

B. Betrò. « Bayesian methods in global optimization ». *Journal of Global Optimization*, 1 : 1–14, 1991.

H.-G. Beyer et H.-P. Schwefel. « Evolution strategies — a comprehensive introduction ». *Natural Computing*, 1(1) : 3–52, 2002.

G. Brocard. *Le simulateur LTspice IV. Manuel, méthodes et applications*. Collection « Collection Technique et ingénierie ». Dunod, 2011.

E. Brochu, V. M. Cora et N. de Freitas. « A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning ». arXiv :1012.2599, 2010.

A. D. Bull. « Convergence rates of efficient global optimization algorithms ». *The Journal of Machine Learning Research*, 12 : 2879–2904, 2011.

J. M. Calvin et A. Zilinskas. « On convergence of a P-algorithm based on a statistical model of continously differentiable functions ». *Journal of Global Optimization*, 19 : 229–245, 2001.

O. Cappé, S. J. Godsill et É. Moulines. « An overview of existing methods and recent advances in sequential Monte Carlo ». *Proceedings of the IEEE*, 95 (5) : 899–924, 2007.

J.-P. Chilès et P. Delfiner. *Geostatistics : modeling spatial uncertainty*, volume 713. Wiley, 2012.

N. Chopin. « A sequential particle filter method for static models ». *Biometrika*, 89(3) : 539–552, 2002.

A. R. Conn, K. Scheinberg et L. N. Vicente. *Introduction to derivative-free optimization*. SIAM, 2009.

D. D. Cox et S. John. « SDO : A statistical method for global optimization ». In *M. N. Alexandrov and M.Y. Hussaini, editors, Multidisciplinary Design Optimization : State of the Art*, pp. 315–329. SIAM, 1997.

N. Cressie. *Statistic for spatial data*. John Wiley, New York, 1993.

P. Del Moral, A. Doucet et A. Jasra. « Sequential Monte Carlo samplers ». *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, 68(3) : 411–436, 2006.

L. C. W. Dixon et G. P. Szego. « The global optimization problem : an introduction ». *Towards Global Optimization*, 2 : 1–15, 1978.

R. Douc et É. Moulines. « Limits theorems for weighted samples with applications to sequential Monte Carlo methods ». *The Annals of Statistics*, 36 (5) : 2344–2376, 2008.

J. A. Egea Larrosa. *New heuristics for global optimization of complex bioprocesses*. Mémoire de thèse, Universidade de Vigo, 2008.

D. E. Finkel. « Direct optimization algorithm user guide ». *Center for Research in Scientific Computation, North Carolina State University*, 2, 2003.

A. I. J. Forrester et D. R. Jones. « Global optimization of deceptive functions with sparse sampling ». In *12th AIAA/ISSMO multidisciplinary analysis and optimization conference*, 10-12 September 2008.

A. Garivier et O. Cappé. « The KL-UCB algorithm for bounded stochastic bandits and beyond ». arXiv :1102.2490, 2011.

M. Gaudard, M. Karson, E. Linder et D. Sinha. « Bayesian spatial prediction ». *Environmental and Ecological Statistics*, 6(2) : 147–171, 1999.

W. R. Gilks et C. Berzuini. « Following a moving target–Monte Carlo inference for dynamic Bayesian models ». *Journal of the Royal Statistical Society*, 63 (1) : 127–146, 2001.

D. Ginsbourger, C. Helbert et L. Carraro. « Discrete mixtures of kernels for kriging-based optimization ». *Quality and Reliability Engineering International*, 24 : 681–691, 2008.

D. Ginsbourger et R. Le Riche. « Towards Gaussian process-based optimization with finite time horizon ». In A. Giovagnoli, A. C. Atkinson, B. Torsney et C. May (éditeurs), *mODa 9 — Advances in Model-Oriented Design and Analysis*, pp. 89–96. 2010.

D. Ginsbourger et O. Roustant. « DiceOptim : Kriging-based optimization for computer experiments ». *R package version 1.2*, 2011.

F. Glover et M. Laguna. *Tabu Search*. Kluwer Academic Publishers, Norwell, MA., 1997.

R. Gramacy et N. Polson. « Particle learning of Gaussian process models for sequential design and optimization ». *Journal of Computational and Graphical Statistics*, 20(1) : 102–118, 2011.

S. Grünewälder, J.-Y. Audibert, M. Opper et J. Shawe-Taylor. « Regret bounds for Gaussian process bandit problems ». In *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (AISTATS 2010)*, volume 9 dans *JMLR W&CP*, pp. 273–280, 2010.

H.-M. Gutmann. « A radial basis function method for global optimization ». *Journal of Global Optimization*, 19 : 201–227, 2001.

M. S. Handcock et M. L. Stein. « A Bayesian analysis of kriging ». *Technometrics*, 35(4) : 403–410, 1993.

D. Huang, T.T. Allen, W.I. Notz et N. Zeng. « Global optimization of stochastic black-box systems via sequential kriging meta-models ». *Journal of Global Optimization*, 34(3) : 441–466, 2006.

W. Huyer et A. Neumaier. « Global optimization by multilevel coordinate search ». *Journal of Global Optimization*, 14(4) : 331–355, 1999.

A. J. Izenman. *Modern multivariate statistical techniques : regression, classification, and manifold learning*. Springer, 2008.

D. R. Jones. « A taxonomy of global optimization methods based on response surfaces ». *Journal of Global Optimization*, 21 : 345–383, 2001.

D. R. Jones, C. D. Perttunen et B. E. Stuckman. « Lipschitz optimization without the Lipschitz constant ». *Journal of Global Optimization*, 79 : 157–181, 1993.

D. R. Jones, M. Schonlau et W. J. Welch. « Efficient global optimization of expensive black-box functions ». *J. Global Optim.*, 13(4) : 455–492, 1998.

É. Kaufmann, O. Cappé et A. Garivier. « On Bayesian upper-confidence bounds for bandit problems ». In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2012.

S. Kirkpatrick, C. D. Gelatt et M. P. Vecchi. « Optimization by simulated annealing ». *Science*, 220 : 671–680, 1983.

J. Klemelä. « Multivariate histograms with data-dependent partitions ». *Statistica Sinica*, 19 : 159–176, 2009.

D. G. Krige. *A statistical approach to some mine valuations and allied problems at the witwatersrand*. Mémoire de thèse, Master's thesis, University of Witwatersrand, 1951.

H. J. Kushner. « A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise ». *J. Basic Engineering*, 86 : 97–106, 1964.

P. Lefranc. *Étude, conception et réalisation de circuits de commande d'IGBT à forte puissance*. Mémoire de thèse, Institut National Des Sciences Appliquées De Lyon, 2005.

P. Lefranc, X. Jannot et P. Dessante. « Virtual prototyping and pre-sizing methodology for buck dc-dc converters using genetic algorithms ». *Power Electronics, IET*, 5(1) : 41–52, 2012.

J. Liu et R. Chen. « Sequential Monte Carlo methods for dynamic systems ». *Journal of American Statistical Association*, 93 : 1032–1044, 1998.

J. S. Liu. *Monte Carlo strategies in scientific computing*. Springer, 2008.

D. J. Lizotte, R. Greiner et D. Schuurmans. « An experimental methodology for response surface optimization ». *Journal of Global Optimization*, 53 : 699–736, 2012.

M. Locatelli. « Bayesian algorithms for one-dimensional global optimization ». *Journal of Global Optimization*, 10(1) : 57–76, 1997.

M. Locatelli et F. Schoen. « An adaptive stochastic global optimization algorithm for one-dimensional functions ». *Annals of Operations research*, 58 (4) : 261–278, 1995.

G. Matheron. « Principles of geostatistics ». *Economic Geology*, 58 : 1246–1266, 1963.

M. D. McKay, R. J. Beckman et W. J. Conover. « Comparison of three methods for selecting values of input variables in the analysis of output from computer code ». *Technometrics*, 21(2) : 239–245, 1979.

D. Meeker. « Finite element method magnetics ». *Users Manual. Version 4.2*, 2010.

J. Mockus. *Bayesian approach to Global Optimization : Theory and Applications*. Kluwer Acad. Publ., Dordrecht-Boston-London, 1989.

J. Mockus. « Application of Bayesian approach to numerical methods of global and stochastic optimization ». *J. Global Optim.*, 4 : 347–365, 1994.

J. Mockus, V. Tiesis et A. Zilinskas. « The application of Bayesian methods for seeking the extremum. ». In L. Dixon et G. Szego (éditeurs), *Towards Global Optimization*, volume 2, pp. 117–129. Elsevier, 1978.

A. O'Hagan. « Curve fitting and optimal design for prediction ». *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, 40(1) : 1–42, 1978.

A. O'Hagan. « Bayes-Hermite quadrature ». *Journal of statistical planning and inference*, 29(3) : 245–260, 1991.

M. A. Osborne. *Bayesian Gaussian Processes for Sequential Prediction Optimisation and Quadrature*. Mémoire de thèse, University of Oxford, 2010.

M. A. Osborne, R. Garnett et S. J. Roberts. « Gaussian processes for global optimization ». In *3rd International Conference on Learning and Intelligent Optimization (LION3), online proceedings*, Trento, Italy, 2009.

M. A. Osborne, S. J. Roberts, A. Rogers, S. D. Ramchurn et N. R. Jennings. « Towards real-time information processing of sensor network data using computationally efficient multi-output Gaussian processes ». In *Proceedings of the 7th International Conference on Information Processing in Sensor Networks*, pp. 109–120. IEE Computer Society, 2008.

C. Perttunen. « A computational geometric approach to feasible region division in constrained global optimization ». In *Proceedings of the 1991 IEEE Conference on Systems, Man, and Cybernetics*, volume 1, pp. 585–590, 1991.

J. Pilz et G. Spöck. « Why do we need and how should we implement Bayesian kriging methods ». *Stochastic Environmental Research and Risk Assessment*, 22(5) : 621–632, 2008.

J. D. Pintér. *Global optimization. Continuous and Lipschitz optimization : algorithms, implementations and applications*. Springer, 1996.

C.E. Rasmussen et C.K.I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

C.P. Robert et G. Casella. *Monte Carlo statistical methods*. Springer, 2004.

O. Roustant, D. Ginsbourger et Y. Deville. « DiceKriging, DiceOptim : Two R packages for the analysis of computer experiments by kriging-based metamodeling and optimization ». *Journal of Statistical Software*, 51(1) : 1–55, 2012.

M. J. Sasena. *Flexibility and efficiency enhancement for constrained global design optimization with Kriging approximations*. Mémoire de thèse, University of Michigan, Michigan, USA, 2002.

M. Schonlau. *Computer experiments and global optimization*. Mémoire de thèse, University of Waterloo, Waterloo, Ontario, Canada, 1997.

M. Schonlau et W. J. Welch. « Global optimization with nonparametric function fitting ». In *Proceedings of the ASA, Section on Physical and Engineering Sciences*, pp. 183–186. Amer. Statist. Assoc., 1996.

M. Schonlau, W. J. Welch et D. R. Jones. « A data analytic approach to Bayesian global optimization ». In *Proceedings of the ASA, Section on Physical and Engineering Sciences*, pp. 186–191. Amer. Statist. Assoc., 1997.

N. Srinivas, A. Krause, S. Kakade et M. Seeger. « Gaussian process optimization in the bandit setting : no regret and experimental design ». In *Proceedings of the 27th International Conference on Machine Learning (ICML 2010)*, 2010.

M. L. Stein. *Interpolation of spatial data : some theory for kriging*. Springer Verlag, 1999.

R. Storn et K. Price. « Differential evolution — a simple and efficient heuristic for global optimization over continuous spaces ». *Journal of Global Optimization*, 11 : 341–359, 1997.

Y. Tenne et C. K. Goh. *Computational intelligence in optimization : applications and implementations*. Springer, 2010.

A. Törn et A. Zilinskas. *Global Optimization*. Springer, Berlin, 1989.

E. Vazquez. *Modélisation comportementale de systèmes non-linéaires multivariables par méthodes à noyaux et applications*. Mémoire de thèse, Université Paris XI, Orsay, France, 2005.

E. Vazquez et J. Bect. « Convergence properties of the expected improvement algorithm with fixed mean and covariance functions ». *Journal of Statistical Planning and inference*, 140(11) : 3088–3095, 2010.

J. Villemonteix. *Optimisation de fonctions coûteuses*. Mémoire de thèse, Université Paris-Sud XI, Faculté des Sciences d'Orsay, 2008.

J. Villemonteix, E. Vazquez, M. Sidorkiewicz et E. Walter. « Global optimization of expensive-to-evaluate functions : an empirical comparison of two sampling criteria ». *Journal of Global Optimization*, 43(2-3) : 373–389, 2009.

J. Villemonteix, E. Vazquez et E. Walter. « Identification of expensive-to-simulate parametric models using kriging and stepwise uncertainty reduction ». In *Conference on Decision and Control*, Nouvelle Orléans, États-Unis, 2007.

E. Walter et L. Pronzato. *Identification of parametric models from experimental data. Communications and Control Engineering Series.* Springer, London, 1997.

R. M. Jr. Walter et S. Jasjeet. « Genetic optimization using derivatives : the rgenoud package for R ». *Journal of Statistical Software*, 42(11) : 1–26, 2011.

B. Williams, T. Santner et W. Notz. « Sequential design of computer experiments to minimize integrated response functions ». *Statistica Sinica*, 10(4) : 1133–1152, 2000.

A. Zhigljavsky et A. Zilinskas. *Stochastic global optimization.* Springer, 2007.

A. Zilinskas. « A review of statistical models for global optimization ». *Journal of Global Optimization*, 2 : 145–153, 1992.