Stochastic approximation methods for PDE constrained optimal control problems with uncertain parameters

Thèse N° 7233

Présentée le 22 mars 2019

à la Faculté des sciences de base
Calcul scientifique et quantification de l'incertitude - Chaire CADMOS
Programme doctoral en mathématiques

pour l'obtention du grade de Docteur ès Sciences

par

Matthieu Claude MARTIN

Acceptée sur proposition du jury

Prof. F. Eisenbrand, président du jury
Prof. F. Nobile, directeur de thèse
Prof. E. Ullmann, rapporteuse
Prof. S. Vandewalle, rapporteur
Prof. M. Picasso, rapporteur

2019

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

In memory of Clément . . .

Abstract

We consider the numerical approximation of a risk-averse optimal control problems constrained by an elliptic Partial Differential Equation (PDE) with random coefficients. Specifically, the control function is a deterministic distributed forcing term that minimizes the expected mean squared distance between the state (i.e. solution to the PDE) and a target function, subject to a regularization for well posedness.

For the computation of the approximated optimal control, we combine different approximation steps, namely: a Finite Element discretization of the underlying PDEs; a quadrature formula to approximate the expectation in the objective functional; and gradient type iterations to compute the approximated optimal control.

We start by considering a Monte Carlo quadrature formula, based on random points, and compare the complexity of a full gradient method, in which the finite element discretization and the Monte Carlo sample are chosen initially and kept fixed over the gradient iterations, with a *Stochastic Gradient* (SG) method in which the expectation in the computation of the steepest descent direction is approximated by an independent Monte Carlo estimator, with small sample size, at each iteration, and the finite element discretization is possibly refined along the iterations.

We then extend the SG method, by replacing the single evaluation of the gradient on a single mesh, by a multilevel Monte Carlo (MLMC) estimator of the gradient, that exploits a hierarchy of finite element discretizations. We propose, in particular, strategies to increase the numbers of discretization levels and Monte Carlo samples per level along the iterations, to achieve an optimal complexity.

As a last approach, we consider a tensorized Gaussian quadrature formula, in the case where the randomness in the PDE can be parametrized by a small number of random variables, and propose to use a generalized version of the Stochastic Average Gradient method (SAGA) to compute the approximated optimal control. SAGA is a type of SG algorithm with a fixed-length memory term, which computes at each iteration the gradient of the loss functional in only one quadrature point, randomly chosen from a possibly non-uniform distribution.

For all the methods developed in this work, we present a full theoretical error and complexity analysis. Specifically, we show that the SG strategy, when combined with a Monte Carlo approximation, results in an improved computational complexity with respect to a full gradient approach, and, incorporating a MLMC estimator, improves further the complexity. On the other hand, SAGA is well adapted when a quadrature formula with spectral convergence properties is considered, and the resulting algorithm has a similar asymptotic complexity as the full gradient method, although with possibly better pre-asymptotic behavior. All theoretical error estimates and complexity results are confirmed by some numerical experiments.

Résumé

Le sujet de ce travail de thèse est l'approximation numérique des problèmes de type contrôle optimal, avec des contraintes formées par des équations différentielles aux dérivées partielles (EDP) avec coefficients incertains. Plus précisément, le terme de contrôle est une fonction déterministe, distribuée dans tout le domaine spatial ; il agit comme un terme de force dans l'EDP, de façon à minimiser une fonctionnelle, égale à la moyenne de la distance au carré, entre la fonction d'état, solution de l'EDP, et une fonction cible. Un terme de régularisation est ajouté à la fonctionnelle, afin de garantir un problème bien posé.

Concernant le calcul approché de ce contrôle optimal, nous combinons différents niveaux d'approximation, i.e. une discrétisation de type Élément Finis (EF) pour approcher l'EDP, une formule de quadrature, pour approcher l'espérance qui apparaît dans ladite fonctionnelle, ainsi qu'une récurrence de type descente de gradient, afin de produire une séquence de fonctions, approchant ce contrôle optimal.

Après avoir introduit la formule de quadrature de type Monte Carlo (MC), qui repose sur des échantillons tirés de manière aléatoire, nous comparons la complexité de la méthode dite de gradient complet (GC), ou la discrétisation EF, et les réalisations MC tirées au hasard, sont choisies une fois pour toute au début de l'expérience et inchangées ensuite, avec la méthode du gradient stochastique (GS), ou l'on approxime, à chaque itération, l'espérance de la fonctionnelle par un estimateurs MC, lui-même échantillonné itération après itération (reposant sur un petit nombre fixe de réalisations), et ou la discrétisation de type EF peut devenir de plus en plus précise au fil des itérations. Nous généralisons ensuite cette idée de GS en utilisant un estimateur Monte Carlo à plusieurs niveaux (MLMC), qui, au lieu d'exploiter seulement un seul niveau de maillage du domaine spatial à chaque itération, comme dans le cas du GS évoqué précédemment, approxime l'espérance par une somme d'estimateur MC classiques, chacun d'eux reposant sur un maillage de discretisation de taille différent. Nous proposons en particulier l'augmentation du nombre d'estimateur MC classiques, qui apparaissent dans ladite somme, tout comme l'augmentation du nombre de réalisations intervenants dans ces estimateurs MC classiques, au cours des itérations, de sorte à obtenir une complexité algorithmique optimale.

Enfin, supposant que l'aléa de l'EDP peut être décrit en fonction d'un petit nombre de variables aléatoires, nous considérons une formule tensorielle de quadrature Gaussienne, et proposons de suivre l'algorithme SAGA afin d'approcher, in fine, le contrôle optimal. L'algorithme SAGA est un algorithme du type GS, suggérant (cependant) de stocker dans

une mémoire de taille fixée, certains gradients, calculés aux itérations précédentes, et de suivre ensuite la moyenne de ces gradients, en guise de direction de descente, afin d'obtenir l'itéré suivant. En particulier, chaque gradient calculé correspond à un seul nœud de quadrature, ce nœud étant tiré au hasard suivant une loi, pouvant être non-uniforme.
Pour l'ensemble des méthodes présentées dans ce travail de thèse, nous proposons une analyse complète d'erreur, ainsi que des complexités associées. Nous montrons en particulier que la stratégie GS, combinée à une approximation de type MC, améliore la complexité algorithmique, par rapport à celle du gradient complet ; de plus, cette complexité est encore améliorée en remplaçant l'estimateur MC par une approximation du type MLMC. D'un autre côté, l'algorithme SAGA est privilégié lorsque nous utilisons une formule de quadrature dotées de propriétés de convergence spectrale, produisant un algorithme de même complexité algorithmique que celle du gradient complet, bien que le comportement de convergence pré-asymptotique semble bien meilleur. Toutes les estimations d'erreur, et les complexités théoriques, sont accompagnées de vérifications numériques.

Contents

I Introduction

Many engineering problems involve uncertain systems, either because of intrinsic variability in the system, or imprecise manufacturing processes, or a lack of knowledge of the system parameters. Often, the system behavior is modeled by means of Partial Differential Equations (PDEs), with random input data, to account for the above mentioned variability. A highly relevant question in engineering applications is that of optimizing the performance of a system or controlling it to achieve certain target conditions. The topic of PDE-constrained optimization has been widely studied in the literature [HPUU09, BS12, De 15, Haz10, LBE$^+$14] in a deterministic setting and only more recently optimization under uncertainty in PDE based models has been addressed, e.g. in [CQ14, AAUH17, TKXP12, RW12, KS13, BOS16, APSG17, VBV18, BvW11, Kou12, KS16, KHRvBW13, GLL11]. As an example, think for instance at the problem of designing the shape of a wing of an aircraft to minimize a certain quantity of interest (gas consumption, risk of crash, etc.). Hence, one usually introduces a functional, mapping the shape properties onto the real line, which drives the optimization process to reach the desirable properties for the final shape. The actual production of the wing will involve imprecise manufacturing processes, i.e. the exact dimensions and shape of the wing will slightly differ from the nominal ones, and the design of an *optimal* shape should take this into account, and should be efficient and reliable in almost every in-flight condition, e.g. air speed, pressure, etc. As the manufacturing error, or the distribution of the flying conditions are unknown at the moment of the design, it is reasonable to consider shape optimization problems, constrained by models with uncertain coefficients, which could be described as random variables. Beside shape optimization, many other types of optimization or control of systems governed by PDEs can be found in applications. We refer to this type of problems as PDE-constrained optimal control problems (OCPs) under uncertainty, which is the main topic of this work.

Different approaches for PDE-constrained OCP under uncertainties can be found in the literature, which shall roughly be grouped into 2 classes. In the first one, one assumes that the randomness is observable and thus designs an optimal control with respect to (w.r.t.) the observed realization. The resulting control is random, and the works [CQ14, AAUH17, TKXP12, RW12, KS13, BOS16] are built on this approach. The dependence of the output Quantity of Interest (QoI) to be optimized, on the random parameters is typically approximated either by polynomial chaos expansions or Monte Carlo (MC) techniques. The former approach is considered e.g. in [KS13], where the authors prove analytic dependence of the control on the random parameters and study its best N-term polynomial chaos approximation for a linear parabolic PDE-constrained OCP. The work [CQ14], combines a stochastic collocation with a Finite Element (FE) based reduced basis method to alleviate the computational effort. In the works [RW12, TKXP12, BOS16] the authors address the problem of computing at once the stochastic control for all realizations thus leading to a fully coupled (in space and stochastic variables) optimality system discretized by either Galerkin or collocation approaches. They propose different methods, such as sequential quadratic programming, or block diagonal preconditioning to solve the coupled system efficiently. Monte Carlo and multilevel Monte Carlo (MLMC) approaches are considered in [AAUH17] instead, where the case of random coefficients with limited spatial regularity is addressed.

In the second class, the control is chosen *deterministic* [APSG17, VBV18, BvW11, Kou12, KS16, KHRvBW13, GLL11]. This situation happens when randomness in the system is not observable at the time of designing the control, so that the latter should be *robust* in the sense that it minimizes the *risk* of obtaining a solution which leads to high values of the objective function. This situation always leads to a fully coupled optimality system in the random parameters. The idea of minimizing a risk to obtain a solution with favorable properties goes back to the origins of robust optimization [SDR09]. Here, *risk* refers to a suitable statistical measure of the objective function to be minimized, such as its expectation, expectation plus variance, a quantile, or a conditional expectation above a quantile (so called Conditional Value at Risk (CVaR) [RU02]).

We can find a vast literature about numerical approximation of PDE-constrained OCPs in the deterministic setting (see e.g. [BS12, HPUU09]) , as well as on the numerical approximation of uncontrolled PDEs with random coefficients (see e.g. [GWZ14, BNT10, LPS14]and references therein). On the other hand, the analysis of PDE-constrained OCPs under uncertainty is much more recent and still incomplete. Numerical methods for such OCPs under uncertainty typically depend on the choice of the risk measure. We review some relevant literature in the next Chapter.

This thesis focuses on numerical approximation of OCPs, constrained by elliptic PDEs with random coefficients, where the control acts as a volumetric forcing term, so that the state, solution of the PDE, should be as close as possible to a given target function. The functional to optimize is the L^2 squared distance between the state function and a target function, plus a regularization term, depending on the L^2 squared norm of the control, and the risk measure considered is just the expected value of the cost functional.

Specifically, the PDE writes:

$$\begin{cases} -\operatorname{div}(a(x,\omega)\nabla y(x,\omega) = g(x) + u(x) & \text{for } x \in D \text{ a.e. } \omega \in \Gamma \\ y(x,\omega) = 0 & \text{for } x \in \partial D \text{ a.e. } \omega \in \Gamma. \end{cases} \tag{I.1}$$

where $(\Gamma, \mathcal{F}, \mathbb{P})$ is a complete probability space, and $D \subset \mathbb{R}^d$ is the spacial domain, while the optimization problem is:

$$\text{find } u_\star \in \underset{u \in \mathcal{U}}{\arg\min}\, J(u), \quad J(u) = \mathbb{E}_\omega[f(y_\omega, u)], \tag{I.2}$$

where the functional to minimize is:

$$f(y_\omega, u) = \frac{1}{2}\|y_\omega - z_d\|^2_{\mathcal{X}} + \frac{\beta}{2}\|u\|^2_{\mathcal{U}}. \tag{I.3}$$

The state function y_ω solves problem (I.1), indexed by the stochastic parameter ω, and $\|\cdot\|_{\mathcal{X}}$ is a suitable norm to measure the distance of the state to the target z_d.

Different types of approximations are required, in order to transform this infinite dimensional problem, into a numerically tractable one: i) a discretization of the spatial domain D, in order to approximate the state/control functions in suitable finite dimensional spaces; ii) an approximation of the statistical moments appearing in the risk measure, e.g. in the particular setting considered here, the expected loss; iii) an optimization routine, to compute one (approximated) minimizer on the quantity of interest. We consider in particular: gradient type methods, where adjoint calculus is used to represent the gradient of the objective function; finite element approximations of the primal and dual problems; and collocation-type approximations for the expectation in the risk measure, either based on random points (Monte Carlo), or deterministic points associated to some Gaussian quadrature formula.

Although finite dimensional, the resulting discrete OCP will be of very high dimension, and each gradient evaluation will incur a very high computational cost as it requires multiple solutions of the primal and dual problems. It is therefore crucial to derive optimization algorithms that are robust w.r.t. the dimension of the state/control discretized spaces and efficient in the gradient evaluation. In this thesis, we have focused on stochastic-type gradient algorithms as they allow for fast updates of the (discretized) control, i.e. they dramatically reduce the cost of each iteration, compared to traditional gradient-type algorithm. The main focus of this work has been to analyze the convergence of such algorithms for the OCP (I.2)-(I.3), study their complexity, i.e. the amount of work required to achieve a prescribed tolerance, and compare them with more traditional gradient-type methods.

In the next Chapter, we recall some basic ingredients that will be used throughout the thesis. In particular, we introduce the framework for deterministic OCPs, recall the main well-posedness results and approximation concepts, and review the basic optimization techniques for unconstrained optimization. We then introduce the stochastic counterpart of OCPs, as well as the notion of risk measure, and review some machine learning techniques that are commonly used to solve stochastic optimization problems involving some expected loss. We postpone a detailed outline of the thesis to the end of the next Chapter.

II Ingredients

II.A. Deterministic PDE-constrained Optimal Control Problems

Optimal Control Problems (OCPs) constrained by deterministic Partial Differential Equations (PDEs) are a well studied topic in the literature, thanks to their relevance in industrial applications. The PDE models the physics of the system under study and involves a control term, acting on the boundary ∂D, or on part of the domain $D \in \mathbb{R}^d$ (possibly everywhere). The system is described by a state/primal function $y \in \mathcal{Y}$ that solves the underlying PDE, with the control term $u \in \mathcal{U}$, acting as a forcing term. The ultimate goal of an OCP is to find an optimal control $u_\star \in \mathcal{U}$ such that the state function $y(u) \in \mathcal{Y}$ is as close as possible to a target function z_d, where the subscript d stands for *desirable*. The notion of closeness evoked above, is expressed by minimizing a distance function between the two quantities $y(u)$ and z_d, using typically some problem specific norm. To ensure that the control function remains acceptable (e.g. with bounded energy), a regularization term is usually added to the *objective functional* to minimize.

One can use discretization techniques to solve the underlying PDE numerically, and the optimization routine usually involves an iterative procedure. The resulting finite dimensional optimization problem could be of very high dimension for distributed controls and the evaluation of the objective functional could be very costly, when the PDE is discretized on a fine mesh.

II.A.1. Notations and definitions for linear-quadratic OCP

In this subsection, to fix the ideas and notations, we set the framework for the OCPs studied in this thesis.

- The *control* function u belongs to an admissible set $\mathcal{U}_{ad}$, subset of a function space $\mathcal{U} \supset \mathcal{U}_{ad}$. When $\mathcal{U}_{ad} \subsetneq \mathcal{U}$, the problem is said to be *constrained.*

- For any control $u \in \mathcal{U}_{ad}$ the state $y(u)$ of the system is an element of a suitable function space $\mathcal{Y}$ that satisfies the linear state equation:

  $$Ay + Bu = g,$$

  where $A : \mathcal{Y} \to \mathcal{Z}$ is a differential linear operator, $B : \mathcal{U}_{ad} \to \mathcal{Z}$ is a linear operator, $g \in \mathcal{Z}$ is a source term, with $\mathcal{Y}$ and $\mathcal{Z}$ being Banach spaces, and $\mathcal{U}$ is a Hilbert space with inner product $(\cdot, \cdot)_{\mathcal{U}}$.

- The target function z_d is an element of a Hilbert space $\mathcal{X} \supset \mathcal{Y}$ and the distance between the state y and the target z_d is measured w.r.t. the norm $\|\cdot\|_{\mathcal{X}}$.

- A cost functional $f(y, u)$ is defined on the space $\mathcal{Y} \times \mathcal{U}$ by

  $$f(y, u) = \frac{1}{2}\|y - z_d\|^2_{\mathcal{X}} + \frac{\beta}{2}\|u\|^2_{\mathcal{U}},$$

  where the parameter β represents the *price of energy* (energy of u, to make $y(u)$ approach z_d).

Alternatively, one could consider the case where the state y is only partially observable (see e.g. [Qua09]) and the cost functional contains the term $\|\mathcal{H}y - z_d\|_{\mathcal{X}}$, with $\mathcal{H} : \mathcal{Y} \to \mathcal{X}$ a linear operator (observation operator).

II.A.2. Existence and uniqueness results for OCPs

For the rest of this work, we assume that we have access to full observation of the state function y, i.e. $\mathcal{H} = Id : \mathcal{Y} \to \mathcal{X}$. A linear-quadratic OCP can then be written generically in the following form:

$$\begin{cases} \min\limits_{y \in \mathcal{Y}, u \in \mathcal{U}_{ad}} f(y, u) = \frac{1}{2}\|y - z_d\|^2_{\mathcal{X}} + \frac{\beta}{2}\|u\|^2_{\mathcal{U}} \\ \text{s.t.} \quad Ay + Bu = g. \end{cases} \tag{II.1}$$

We define formally an optimal state-control pair, as:

Definition 1. *A state-control pair $(y_\star, u_\star) \in \mathcal{Y} \times \mathcal{U}_{ad}$ is said to be* optimal *for* (II.1) *if*

1. $Ay_\star + Bu_\star = g$,
2. $f(y_\star, u_\star) \le f(y, u) \quad \forall (y, u) \in \mathcal{Y} \times \mathcal{U}_{ad}$, s.t. $Ay + Bu = g$.

We now state standard assumptions (see e.g. [HPUU09]) that guarantee existence of solutions to the OCP (II.1).

Assumption 1.

1. *$\beta \geq 0$; $\mathcal{U}_{ad} \subset \mathcal{U}$ is convex, closed and, if $\beta = 0$, bounded;*
2. *There exists a pair $(y, u) \in \mathcal{Y} \times \mathcal{U}_{ad}$ such that $Ay + Bu = g$ (i.e. (y, u) is a feasible point);*
3. *$A \in \mathcal{L}(\mathcal{Y}, \mathcal{Z})$ has a bounded inverse.*

Theorem 1. *Let Assumption 1 hold. Then the OCP* (II.1) *has an optimal solution $(y_\star, u_\star)$. If $\beta > 0$, then the solution is unique.*

Proof. A proof can be found, for example, in [HPUU09, Theorem 1.43]. □

II.A.3. Formulation with Lagrange multipliers

The OCP we are interesting in this thesis is of the form (II.1), and is a constrained optimization problem under the constraints $Ay+Bu-g = 0$. We can write the Lagrangian function $L : \mathcal{Y} \times \mathcal{U} \times \mathcal{Z}^* \to \mathbb{R}$ as:

$$L(y, u, p) = f(y, u) + \langle p, Ay + Bu - g\rangle_{\mathcal{Z}^*,\mathcal{Z}} \tag{II.2}$$

where $\mathcal{Z}^*$ denotes the dual space of $\mathcal{Z}$ and $\langle \cdot, \cdot, \rangle_{\mathcal{Z}^*,\mathcal{Z}}$ the duality pairing. Using the identification $\mathcal{U}^* \equiv \mathcal{U}$, the optimality conditions reads:

$$\begin{cases} L_y(y, u, p) = f_y(y, u) + A^*p = 0, & \text{in } \mathcal{Y}^* \\ L_u(y, u, p) = f_u(y, u) + B^*p = 0, & \text{in } \mathcal{U} \\ L_p(y, u, p) = Ay + Bu - g = 0 & \text{in } \mathcal{Z}. \end{cases} \tag{II.3}$$

where the adjoint operators $A^* : \mathcal{Z}^* \mapsto \mathcal{Y}^*$ and $B^* : \mathcal{Z}^* \mapsto \mathcal{U}$ are defined as

$$\langle p, Ay\rangle_{\mathcal{Z}^*,\mathcal{Z}} = \langle A^*p, y\rangle_{\mathcal{Y}^*,\mathcal{Y}} \quad \forall y \in \mathcal{Y},\ p \in \mathcal{Z}^*,$$

$$\langle p, Bu\rangle_{\mathcal{Z}^*,\mathcal{Z}} = (B^*p, u)_{\mathcal{U}} \quad \forall u \in \mathcal{U},\ p \in \mathcal{Z}^*.$$

We recognize the adjoint problem on the first line of (II.3), the optimality condition on the second, and finally the primal problem on the third. From Assumption 1.1-1.3, we know that there exists an affine solution operator $u \mapsto y(u) = A^{-1}(g - Bu)$, thus we can introduce the reduced objective functional, defined as

$$J(u) = f(y(u), u) = \frac{1}{2}\|y(u) - z_d\|^2_{\mathcal{X}} + \frac{\beta}{2}\|u\|^2_{\mathcal{U}} \tag{II.4}$$

with the state solution $y(u) = A^{-1}(g - Bu)$. In order to use gradient based optimization algorithms, we need to derive an expression for the steepest descent direction. This is based on the adjoint theory. An expression of the Gateaux derivative of the reduced cost functional (II.4) w.r.t. u can be explicitly computed, and can be obtained from the partial derivative of the Lagrangian L w.r.t. u. We present hereafter, however, the direct derivation of the Gateaux derivative of J using adjoint calculus. We denote by $\mathrm{d}J(u,s)$ the **directional derivative** of J in the direction $s \in \mathcal{U}$ namely

$$\mathrm{d}J(u,s) = \lim_{\epsilon \to 0^+} \frac{J(u+\epsilon s) - J(u)}{\epsilon}.$$

Moreover, we recall that J is called **Gateaux differentiable** at every $u \in \mathcal{U}$, if J is directionally differentiable at u in any direction $s \in \mathcal{U}$ and the directional derivative $\mathcal{U} \ni s \mapsto \mathrm{d}J(u,s) \in \mathbb{R}$ is bounded and linear, i.e. there exists $J'(u) \in \mathcal{L}(\mathcal{U}, \mathbb{R})$ s.t. $J'(u)s = \mathrm{d}J(u,s)$, $\forall s \in \mathcal{U}$. Furthermore, we say that J is **Fréchet differentiable** at $u \in \mathcal{U}$ if J is Gateaux differentiable at u and if the following approximation condition holds:

$$|J(u+s) - J(u) - J'(u)s| = o\left(\|s\|_{\mathcal{U}}\right).$$

Thanks to the Riesz representation theorem, [RSN90],[Qua09, Theorem 2.1], every element of $\phi \in \mathcal{U}^*$ can be written uniquely in the form $\phi(\cdot) = (\Phi, \cdot)_{\mathcal{U}}$, with an element $\Phi \in \mathcal{U}$. In particular, with a little abuse of notation, we use the same symbol $J'(u)$ to denote the functional in $\mathcal{U}^*$ and its representation in $\mathcal{U}$. We use the adjoint approach and derive the adjoint equation, and an expression for $J'(u)$:

$$\begin{aligned}(J'(u), s)_{\mathcal{U}} = \langle J'(u), s\rangle_{\mathcal{U}^*,\mathcal{U}} &= \langle f_y(y(u),u), y'(u)s\rangle_{\mathcal{Y}^*,\mathcal{Y}} + (f_u(y(u),u), s)_{\mathcal{U}} \\ &= (y'(u)^* f_y(y(u),u), s)_{\mathcal{U}} + (f_u(y(u),u), s)_{\mathcal{U}}.\end{aligned}$$

So

$$J'(u) = y'(u)^* f_y(y(u),u) + f_u(y(u),u),$$

and, since $y'(u) = -A^{-1}B$, we can write

$$y'(u)^* f_y(y(u),u) = (-A^{-1}B)^* f_y(y(u),u) = -B^* A^{-*} f_y(y(u),u).$$

It follows that

$$y'(u)^* f_y(y(u),u) = B^* p(u)$$

with the adjoint variable $p = p(u) \in \mathcal{Z}^*$ satisfying the *adjoint equation*

$$A^* p = -f_y(y(u),u).$$

In particular, *in the linear-quadratic OCP setting* (II.1), we have the following expression for the gradient $J'(u)$, of the reduced functional $J(u)$:

$$J'(u) = \beta u + B^* p(u), \tag{II.5}$$

with the dual function p solving the adjoint problem, formulated in weak form:

$$\langle A^* p, v\rangle_{\mathcal{Y}^*,\mathcal{Y}} = \langle -f_y(y(u),u), v\rangle_{\mathcal{Y}^*,\mathcal{Y}} = -\left(y(u) - z_d, v\right)_{\mathcal{X}}, \quad \forall v \in \mathcal{Y}. \tag{II.6}$$

Notice that if, in particular, $\mathcal{U} = \mathcal{X}$, we end up with

$$A^* p = -(y(u) - z_d). \tag{II.7}$$

After recalling some usual function spaces, we particularize the above adjoint method, to two OCPs involving an elliptic PDE: i) a PDE with Dirichlet boundary condition and distributed control over the whole domain D; ii) a PDE with Neumann boundary condition, with control acting only on the boundary ∂D.

II.A.4. Function spaces

Let $L^p(D)$ for $1 \leq p < \infty$ denote the space of functions for which the p-th power of their absolute value is Lebesgue integrable, that is

$$L^p(D) = \{y : D \to \mathbb{R}, \;\; f \text{ measurable, and } \int_D |y|^p \mathrm{d}x < +\infty\},$$

and $L^\infty(D)$ the space of measurable functions that are bounded almost everywhere (a.e.) on D. Throughout this work, we will denote by $\|\cdot\| \equiv \|\cdot\|_{L^2(D)}$ the usual $L^2(D)$-norm induced by the inner product $\langle f, g\rangle = \int_D fg\mathrm{d}x$ for any $f, g \in L^2(D)$. Furthermore, we introduce the Sobolev spaces

$$H^1(D) = \{y \in L^2(D), \quad \partial_{x_i} y \in L^2(D), \quad i = 1, \ldots, n\}$$

and

$$H_0^1(D) = \{y \in H^1(D), \quad y|_{\partial D} = 0\}.$$

We use the equivalent H^1-norm on the space $H_0^1(D)$ defined by $\|y\|_{H^1(D)} = \|y\|_{H_0^1(D)} = \|\nabla y\|$ for any $y \in H_0^1(D)$. Moreover, we recall the Poincaré inequality for any function $y \in H_0^1(D)$:

$$\|y\| \leq C_p \|\nabla y\| = C_p \|y\|_{H^1(D)},$$

where C_p is the Poincaré constant, and denote by $H^{-1}(D) = \left(H_0^1(D)\right)^*$ the topological dual of $H_0^1(D)$. For $r \in \mathbb{N}$ we further recall the space $H^r(D)$ of $L^2(D)$ functions with all partial derivatives up to order r in $L^2(D)$ with norm $\|y\|_{H^r(D)}$ and semi-norm $|y|_{H^r(D)}$

given by

$$\|y\|^2_{H^r(D)} = \sum_{|\boldsymbol{\alpha}|\leq r} \left\| \frac{\partial^{|\boldsymbol{\alpha}|} y}{\partial x^{\boldsymbol{\alpha}}} \right\|^2_{L^2(D)} \quad \text{and} \quad |y|^2_{H^r(D)} = \sum_{|\boldsymbol{\alpha}|= r} \left\| \frac{\partial^{|\boldsymbol{\alpha}|} y}{\partial x^{\boldsymbol{\alpha}}} \right\|^2_{L^2(D)},$$

respectively, where $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_n)$ is a multi-index.

II.A.5. Dirichlet boundary conditions and interior control

We consider here a *distributed* control problem, with Dirichlet boundary conditions on the underlying PDE. The control u acts on the whole domain D, in order to influence the state function y, through the following elliptic PDE:

$$\begin{cases} -\operatorname{div}(a\nabla y) = g + u & \text{in } D \\ \qquad\quad y = 0 & \text{on } \partial D. \end{cases} \tag{II.8}$$

We state here standard assumptions to guarantee existence and uniqueness of solutions to such elliptic PDE.

Assumption 2. *The diffusion coefficient $a \in L^\infty(D)$ is bounded away from zero in D, i.e.*

$$a_{min} := \operatorname*{essinf}_{x\in D} a(x) > 0 \quad \text{and} \quad a_{max} := \operatorname*{esssup}_{x\in D} a(x) < +\infty$$

Now we are in the position to recall the well posedness of the elliptic PDE (II.8).

Lemma 1. *Let Assumption 2 hold. If $g + u \in H^{-1}(D)$, then problem* (II.8) *admits a unique solution $y \in H^1_0(D)$ s.t.*

$$\|y\|_{H^1_0(D)} \leq \frac{1}{a_{min}} \|g + u\|_{H^{-1}(D)}.$$

Furthermore, as we will occasionally need H^2-regularity in the following Sections, we also introduce a sufficient condition on the domain D and on the gradient of a.

Assumption 3. *The domain $D \subset \mathbb{R}^n$ is polygonal convex and the diffusion coefficient $a \in L^\infty(D)$ is such that $\nabla a \in L^\infty(D)$,*

Then, using standard regularity arguments for elliptic PDEs, one can prove the following result [Eva98].

Lemma 2. *Let Assumptions 2 and 3 hold. If $g + u \in L^2(D)$, then problem* (II.8) *has a unique solution $y \in H^2(D)$. Moreover there exists a constant C, independent of $g + u$, such that*

$$\|y\|_{H^2(D)} \leq C\|g + u\|_{L^2(D)}.$$

The global OCP writes:

$$\begin{cases} \min\limits_{y\in H_0^1(D),u\in L^2(D)} & f(y,u) = \dfrac{1}{2}\|y - z_d\|^2_{L^2(D)} + \dfrac{\beta}{2}\|u\|^2_{L^2(D)} \\ \text{s.t.} & -\operatorname{div}(a\nabla y) = g + u \quad \text{in } D, \\ \text{and} & \qquad\qquad y = 0 \qquad\quad \text{in } \partial D. \end{cases} \tag{II.9}$$

Following arguments in Section II.A.3, and setting $\mathcal{U} = \mathcal{U}_{ad} = L^2(D)$, $\mathcal{Y} = H_0^1(D)$, $\mathcal{Z} = H^{-1}(D)$, $\mathcal{X} = L^2(D)$ we can derive the optimality system:

- Primal problem:

$$\begin{cases} -\operatorname{div}(a\nabla y) = g + u & \text{in } D \\ \qquad\qquad y = 0 & \text{on } \partial D. \end{cases}$$

- Dual problem:

$$\begin{cases} -\operatorname{div}(a\nabla p) = -(y - z_d) & \text{in } D \\ \qquad\qquad p = 0 & \text{on } \partial D. \end{cases}$$

- Gradient:

$$J'(u) = \beta u + p.$$

- Optimality condition:

$$J'(u) = 0, \quad \text{in } D.$$

A steepest descent type algorithm can be constructed easily using the first 3 steps to compute the steepest descent direction.

II.A.6. Neumann boundary conditions and boundary control

Here we consider an elliptic PDE, with a fixed right hand side and a control u acting only on the boundary of the domain ∂D. Specifically, the PDE becomes:

$$\begin{cases} -\operatorname{div}(a\nabla y) + cy = g & \text{in } D \\ \qquad\qquad\quad \frac{\partial y}{\partial \nu} = u & \text{in } \partial D. \end{cases} \tag{II.10}$$

where ν denotes the unit outgoing normal vector to ∂D and $\frac{\partial y}{\partial \nu}$ the normal derivative of y to ∂D. The quantity to minimize is slightly modified (the energy of u is computed only on the border ∂D) as:

$$\min_{y\in H_0^1(D),u\in L^2(D)} f(y,u) = \frac{1}{2}\|y - z_d\|^2_{L^2(D)} + \frac{\beta}{2}\|u\|^2_{L^2(\partial D)} \tag{II.11}$$

In (II.10), the domain D is a subset of $\mathbb{R}^d$ and the function $c \in L^\infty(D)$ is positive, i.e. $c > 0$, a.e. in D. Finally, $g \in H^1(D)^*$ is the source term. The weak formulation of such Neumann problem write:

$$\int_D (a\nabla y \cdot \nabla v + cyv)\mathrm{d}x = \int_D gv\mathrm{d}x + \int_{\partial D} uv\mathrm{d}S \quad \forall v \in H^1(D). \tag{II.12}$$

Thus we can again construct the following linear operators $A \in \mathcal{L}(H^1(D), H^1(D)^*)$ s.t.:

$$\langle Ay, v\rangle_{H^1(D)^*,H^1(D)} = \int_D (a\nabla y \cdot \nabla v + cyv)\mathrm{d}x \quad \forall v \in H^1(D), \tag{II.13}$$

and $B \in \mathcal{L}(L^2(\partial D), H^1(D)^*)$

$$\langle Bu, v\rangle_{H^1(D)^*,H^1(D)} = \int_{\partial D} uv\mathrm{d}S \quad \forall v \in H^1(D), \tag{II.14}$$

Here the adjoint $B^* \in \mathcal{L}(H^1(D), L^2(\partial D))$ of B is such that $B^*v = v|_{\partial D}$. Actually, we have:

$$(B^*v, w)_{L^2(\partial D)} = \langle Bw, v\rangle_{H^1(D)^*,H^1(D)} = \int_{\partial D} wv\mathrm{d}S = (v, w)_{L^2(\partial D)}\,. \tag{II.15}$$

resulting in similar primal/dual equations, and optimality conditions, as in Section II.A.5, i.e. following arguments in Section II.A.3, and setting $\mathcal{U} = \mathcal{U}_{ad} = L^2(\partial D)$, $\mathcal{Y} = H^1(D)$, $\mathcal{Z} = H^1(D)^*$, $\mathcal{X} = L^2(D)$ we can derive the optimality system of the Neumann boundary OCP:

- Primal problem:

$$\begin{cases} -\operatorname{div}(a\nabla y) + cy = g & \text{in } D \\ \frac{\partial y}{\partial \nu} = u & \text{in } \partial D. \end{cases}$$

- Dual problem:

$$\begin{cases} -\operatorname{div}(a\nabla p) + cp = -(y - z_d) & \text{in } D \\ \frac{\partial p}{\partial \nu} = 0 & \text{in } \partial D. \end{cases}$$

- Gradient:

$$J'(u) = \beta u + p, \quad \text{on } \partial D.$$

- Optimality condition:

$$J'(u) = 0, \quad \text{on } \partial D.$$

Again, a steepest descent type algorithm can be constructed using the first 3 steps to compute the steepest descent direction.

II.A.7. Discrete concepts in PDE constrained optimization

We recall here some discrete concepts in PDE constrained OCPs. Specifically, in order to solve such problems numerically, one shall store the functions values for example in some vector variable or in a finite dimension/size memory. We present here a first discretize, then optimize approach, the other approach existing as well, see e.g. [Qua09]. Specifically, to approach numerically a PDE constrained OCP of the form (II.1), one substitutes all function spaces ($\mathcal{U}$, $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{Z}$) by sub-spaces of finite dimension (resp. $\mathcal{U}^h$, $\mathcal{X}^h$, $\mathcal{Y}^h$, $\mathcal{Z}^h$), and each operator (A, B) by a suitable approximate surrogate (resp. A^h, B^h), allowing its evaluation and resolution on a computer. Let us denote by h the discretization parameter (h can be thought of as the mesh size of the cells decomposing the domain D, for instance). We then re-write the OCP (II.1) as:

$$\begin{cases} \min\limits_{y^h \in \mathcal{Y}^h, u^h \in \mathcal{U}^h_{ad}} f^h(y^h, u^h) = \frac{1}{2}\|y^h - z^h_d\|^2_{\mathcal{X}^h} + \frac{\beta}{2}\|u^h\|^2_{\mathcal{U}^h} \\ \text{s.t.} \quad A^h y^h + B^h u^h = g^h. \end{cases} \tag{II.16}$$

where $f^h : \mathcal{Y}^h \times \mathcal{U}^h \to \mathbb{R}$ is a modified version of f, involving the associated norms of the considered surrogate sub-spaces $\mathcal{X}^h$, and $g^h \in \mathcal{Z}^h$, $z^h_d \in \mathcal{X}^h$ are suitable approximation of g, z_d. For the finite dimensional spaces, one may require $\mathcal{U}^h \subset \mathcal{U}$, $\mathcal{Y}^h \subset \mathcal{Y}$, and further Assumptions 4, in order to guarantee existence and uniqueness of the discretized deterministic OCP (II.16).

Assumption 4.

1. *$\beta \geq 0$, $\mathcal{U}^h_{ad} \subset \mathcal{U}^h$ is convex, closed and if $\beta = 0$ bounded*
2. *$\mathcal{Y}^h$ is convex, closed, such that* (II.16) *has a feasible point.*
3. *$A^h \in \mathcal{L}(\mathcal{Y}^h, \mathcal{Z}^h)$ has a bounded inverse.*
4. *$B^h \in \mathcal{L}(\mathcal{U}^h, \mathcal{Z}^h)$.*

Then, under Assumption 4, Theorem 1 applies as well, leading to well posedness of the OCP (II.16), and uniqueness of solutions if $\beta > 0$.

Thus, from this discretization method, we end up with only finite dimensional spaces, and we can treat the optimization routines from a finite dimensional point of view. This is the purpose of next Section.

II.B. Unconstrained optimization techniques

As evoked in the previous Section, we choose a discretize then optimize approach, implying that all spaces $\mathcal{U}$, $\mathcal{X}$, $\mathcal{Y}$ and $\mathcal{Z}$ are replaced with finite-dimensional sub-spaces $\mathcal{U}^h$, $\mathcal{X}^h$, $\mathcal{Y}^h$ and $\mathcal{Z}^h$. Hereafter, however, we removed the sup-scripts h, to lighten the notation and implicitly assume that the infinite dimensional optimal control problem has been discretized, and the goal is now to optimize the finite dimension problem. Here, we aim at minimizing a functional $J(u)$, with the variable $u \in \mathcal{U}$, where $\mathcal{U}$ is a generic finite dimensional space, for example $\mathcal{U} = \mathbb{R}^n$ to fix the ideas. The ultimate goal is to describe the set:

$$\arg\min_{u\in\mathcal{U}} J(u), \tag{II.17}$$

or derive an algorithmic routine to find one point $u_\star \in \arg\min_{u\in\mathcal{U}} J(u)$. A line search algorithm, is usually based on a recursive scheme of the form

$$u_{j+1} = u_j + \tau_j p_j \tag{II.18}$$

where the scalar τ_j is called the step-size, and the vector p_j is named the search direction. The following discussion presents different optimization algorithms, depending on how we choose the sequences $\{\tau_j\}_{j\in\mathbb{N}}$ and $\{p_j\}_{j\in\mathbb{N}}$.

II.B.1. Line search algorithm

Most of the line search algorithms require to compute a search direction p_j that is *descending*, i.e. having $p_j^T \nabla J(u_j) < 0$. Moreover, we often look for search directions p_j that solve a linear system of the type

$$p_j = -\Omega_j^{-1} \nabla J_j, \tag{II.19}$$

with the matrix Ω_j being non-singular and symmetric. For instance, in the steepest descent method, we have $\Omega_j = Id$, and in the Newton's method, the Ω_j is the exact Hessian $\nabla^2 J(u_j)$. The full Hessian is usually computationally very costly to compute and, for scalable algorithms, one tries to avoid such full computation. Quasi-Newton methods are often a good compromise, being more sophisticated then the steepest descent, but less costly then the Newton's method requiring the full Hessian $\Omega_j = \nabla^2 J(u_j)$. In Quasi-Newton methods, Ω_j is an approximation of the full Hessian, updated at every iteration of the recursion (II.18). We present in detail each of these methods in the next subsections, with their known convergence results.

II.B.2. Steepest descent

The steepest descent method is obtained by setting $p_j = -\nabla J(u_j)$, i.e. $\Omega_j = Id$ in equation (II.18). The recurrence scheme becomes

$$u_{j+1} = u_j - \tau_j \nabla J(u_j) \tag{II.20}$$

The choice for the sequence of $\{\tau_j\}_j$ exploits the fact that at each iteration, we aim at solving the minimization problem:

$$\tau_j = \arg\min_{\tau>0} J(u_j - \tau\nabla J(u_j)). \tag{II.21}$$

In practice, the optimal step-size is not easy to compute numerically, and optimization methods generally use sub-optimal sequences of $\{\tau_j\}_j$ guaranteeing sub-optimal convergence of $\{u_j\}_j$ to a minimizer $u_\star$. However, in the quadratic case, we can easily compute the exact optimal step-size sequence. Actually, a generic quadratic functional J writes:

$$J(u) = \frac{1}{2}u^T Q u - b^T u, \tag{II.22}$$

with a positive definite and symmetric matrix Q, and a vector $b \in \mathbb{R}^n$. Then one can easily derive the gradient expression $\nabla J(u) = Qu - b$ and the step-size optimization problem (II.21) writes:

$$\min_{\tau>0} J(u - \tau\nabla J(u)) = \frac{1}{2}(u - \tau\nabla J(u))^T Q(u - \tau\nabla J(u)) - b^T(u - \tau\nabla J(u)) \tag{II.23}$$

Differentiating the last expression w.r.t. τ, we compute the optimal step-size in the quadratic setting $\tau^* = \frac{\nabla J(u)^T \nabla J(u)}{\nabla J(u)^T Q \nabla J(u)}$, and so the optimal step-size, in this particular quadratic case, writes

$$\tau_j = \frac{\nabla J(u_j)^T \nabla J(u_j)}{\nabla J(u_j)^T Q \nabla J(u_j)}. \tag{II.24}$$

Theorem 2. *We denote by $0 < \lambda_1 \leq \cdots \leq \lambda_n$ the eigenvalues of the matrix Q, and we define the Q-norm of a point $x \in \mathbb{R}^n$, by $\|x\|_Q^2 = x^T Q x$. Then if we apply the steepest descent scheme, defined in equation* (II.20)*, in the particular quadratic setting of* (II.22)*, with optimal step size* (II.24)*, the convergence rate of the squared-error in Q-norm, $\|u_j - u_\star\|_Q^2$ is given by*

$$\|u_{j+1} - u_\star\|_Q^2 \leq \left(\frac{\lambda_n - \lambda_1}{\lambda_n + \lambda_1}\right)^2 \|u_j - u_\star\|_Q^2. \tag{II.25}$$

From the previous Theorem, we infer that the convergence is exponential w.r.t. the number of iterations

$$\|u_j - u_\star\|_Q \leq \rho^j \|u_0 - u_\star\|_Q, \tag{II.26}$$

with exponential rate $\rho = (\kappa(Q)-1)/(\kappa(Q)+1)$ depending only on the condition number $\kappa(Q) = \lambda_n/\lambda_1$ of the matrix Q. We present hereafter, an alternative convergence result on the error $\|u_j - u_\star\|$ measured in Euclidean norm instead of the Q-norm. It also shows an exponential convergence rate w.r.t. the number of iterations, although with a worse rate than in (II.26). This result and proof technique will be heavily used in the subsequent Chapters.

Theorem 3. *We denote by $0 < \lambda_1 \leq \cdots \leq \lambda_n$ the eigenvalues of the matrix Q. Then if we apply the steepest descent scheme, defined in equation* (II.20)*, in the particular quadratic setting of* (II.22)*, with step-size $\tau_j = \tau \in \left]0, \frac{2\lambda_1}{\lambda_n^2}\right[$, the convergence rate of the squared-error, $\|u_j - u_\star\|^2$ is given by*

$$\|u_{j+1} - u_\star\|^2 \leq \left(1 - 2\tau\lambda_1 + \tau^2\lambda_n^2\right)\|u_j - u_\star\|^2. \tag{II.27}$$

Moreover, the optimal step-size is $\tau = \tau^* = \frac{\lambda_1}{\lambda_n^2}$, for which the convergence reads

$$\|u_{j+1} - u_\star\|^2 \leq \left(1 - \frac{\lambda_1^2}{\lambda_n^2}\right)\|u_j - u_\star\|^2. \tag{II.28}$$

Proof. From (II.20) we have:

$$\begin{aligned}
\|u_{j+1} - u_\star\|^2 &= \|u_j - u_\star - \tau_j \nabla J(u_j)\|^2 \\
&= \|u_j - u_\star\|^2 - 2\tau_j\langle u_j - u_\star, \nabla J(u_j)\rangle + \|\tau_j \nabla J(u_j)\|^2 \\
&= \|u_j - u_\star\|^2 - 2\tau_j\langle u_j - u_\star, Qu_j - b\rangle + \tau_j^2\|Qu_j - b\|^2.
\end{aligned}$$

Now using the optimality condition, i.e.

$$\nabla J(u_\star) = Qu_\star - b = 0,$$

we obtain:

$$\begin{aligned}
\|u_{j+1} - u_\star\|^2 &= \|u_j - u_\star\|^2 - 2\tau_j\langle u_j - u_\star, Q(u_j - u_\star)\rangle + \tau_j^2\|Q(u_j - u_\star)\|^2 \\
&\leq \|u_j - u_\star\|^2 - 2\tau_j\lambda_1\|u_j - u_\star\|^2 + \tau_j^2\lambda_n^2\|u_j - u_\star\|^2 \\
&\leq \underbrace{\left(1 - 2\tau_j\lambda_1 + \tau_j^2\lambda_n^2\right)}_{=c(\tau_j)}\|u_j - u_\star\|^2.
\end{aligned}$$

Again minimizing the function $\tau \mapsto c(\tau)$ over the fixed step-size, we derive the optimal step size $\tau_j = \tau^* = \frac{\lambda_1}{\lambda_n^2}$ giving an optimal reduction factor of $c(\tau_j) := 1 - \frac{\lambda_1^2}{\lambda_n^2} < 1$, leading to the sub-optimal convergence rate:

$$\|u_{j+1} - u_\star\|^2 \leq \left(1 - \frac{\lambda_1^2}{\lambda_n^2}\right) \|u_j - u_\star\|^2. \tag{II.29}$$

□

Again, from Theorem 3, we infer that the convergence is exponential w.r.t. the number of iterations, i.e.

$$\|u_j - u_\star\| \leq \widetilde{\rho}^j \|u_0 - u_\star\|, \tag{II.30}$$

for any $\tau \in \left]0, \frac{2\lambda_1}{\lambda_n^2}\right[$ and in the optimal case, the exponential rate is $\widetilde{\rho} = \sqrt{1 - \kappa(Q)^{-2}}$. Notice that for every $0 < \lambda_1 < \lambda_n$ we have $\left(1 - \frac{\lambda_1^2}{\lambda_n^2}\right) > \frac{(\lambda_n - \lambda_1)^2}{(\lambda_n + \lambda_1)^2}$. The problem of (II.28) is that the optimal step size $\tau^* = \frac{\lambda_1}{\lambda_n^2}$ requires the knowledge of the eigenvalues of Q or good bounds $\lambda_n < L$ and $\lambda_1 > l/2$ on them. Actually, if one knows λ_1 and λ_n, then the choice $\tau_j = \frac{2}{\lambda_1 + \lambda_n}$ is equally good as (II.24) (but again not practical as λ_1 and λ_n are not known, in general). A special case of this result is when all the eigenvalues of the matrix Q are equal, implying that the convergence is achieved in only one iteration. Specifically, if the matrix Q is a multiple of the identity matrix, then all the steepest descent directions point exactly to the minimizer. The isovalues of such a quadratic function are circles, centered around the unique minimizer. When the condition number $\kappa(Q) = \lambda_n/\lambda_1$ increases, then the isovalues of the quadratic functional are transformed into ellipsoids that become more and more elongated, producing a zigzagging sequence of iterates (II.20) and a slow convergence rate. An advantage of the estimate in Theorem 3 w.r.t. that of Theorem 2 is that it generalizes immediately to the non-quadratic case as long as J is strongly convex and ∇J is globally Lipschitz continuous (see Section II.D.6).

II.B.3. Quasi-Newton methods

The main idea of Quasi-Newton methods is that changes in the gradient of J provide information on the second derivative of J, along the search direction [NW99]. Let us use the following shorthand notation: $\nabla J_j = \nabla J(u_j)$, $\nabla^2 J_j = \nabla^2 J(u_j)$ and $s_j = u_{j+1} - u_j$. Then we can write:

$$\nabla J_{j+1} = \nabla J_j + \nabla^2 J_{j+1}(s_j) + o(\|s_j\|) \tag{II.31}$$

Then, assuming that the last term $o(\|s_j\|)$ is *very small* for points u_j and u_{j+1} being in a neighborhood of the solution $u_\star$, we can approximate:

$$\nabla^2 J_{j+1}(s_j) \approx \nabla J_{j+1} - \nabla J_j =: y_j \tag{II.32}$$

Mimicking equation (II.32), we choose Ω_j to satisfy the *secant equation*:

$$\Omega_{j+1}s_j = y_j. \tag{II.33}$$

Usually, for a quasi-Newton method, we aim at having a symmetric matrix Ω_j, and the matrix difference $\Omega_{j+1} - \Omega_j$ being a low rank matrix. Two very popular update formulas for the matrix Ω_j are the following:

- The symmetric-rank-one (SR1) formula:

$$\Omega_{j+1} = \Omega_j - \frac{(y_j - \Omega_j s_j)(y_j - \Omega_j s_j)^T}{(y_j - \Omega_j s_j)^T s_j} \tag{II.34}$$

- The BFGS formula, named from its inventors initials, Broyden, Fletcher, Goldfarb and Shanno:

$$\Omega_{j+1} = \Omega_j - \frac{\Omega_j s_j s_j^T \Omega_j}{s_j^T \Omega_j s_j} + \frac{y_j y_j^T}{y_j^T s_j} \tag{II.35}$$

We just quote a convergence result from [NW99] for a general Quasi-Newton method:

Theorem 4. *We assume the function $J : \mathcal{U} \to \mathbb{R}$ is three times continuously differentiable. Using a fixed step-size of 1, i.e. considering the scheme $u_{j+1} = u_j + p_j$, with p_j solving the linear system (II.19), if we assume that the sequence $\{u_j\}_j$ converges to a point $u_\star$ being such that $\nabla J(u_\star) = 0$ and $\nabla^2 J(u_\star)$ is positive definite, then $\{u_j\}_j$ converges super-linearly, i.e.*

$$\lim_{j\to\infty} \frac{\|u_{j+1} - u_\star\|}{\|u_j - u_\star\|} = 0 \tag{II.36}$$

if and only if

$$\lim_{j\to\infty} \frac{\|(\Omega_j - \nabla^2 J(u_\star))p_j\|}{\|p_j\|} = 0. \tag{II.37}$$

Condition (II.37) just reflects the principle that the approximated Hessian Ω_j should approximate the exact Hessian $\nabla^2 J_j$ only following the line/direction p_j, in order to reach the super-linear convergence of the Quasi-Newton method.

II.B.4. Newton's method

For the Newton's method, we compute the full Hessian for the matrix $\Omega_j = -\nabla^2 J_j$ and choose the descent p_j such that

$$\nabla^2 J_j p_j = -\nabla J_j. \tag{II.38}$$

The main problem here is that the Hessian $\nabla^2 J_j$ is not always positive definite, in a way that we cannot guarantee that the direction p_j is *descending*. But assuming the following second order condition:

Assumption 5. *The Hessian $\nabla^2 J$ is continuous on an open neighborhood of the solution $u_\star$. Moreover $\nabla J(u_\star) = 0$ and $\nabla^2 J(u_\star)$ is positive definite,*

then we can state a local rate of convergence, i.e. requiring a starting point u_0 being in a neighborhood of $u_\star$, to guarantee that the matrix $\nabla^2 J_j$ remains positive definite for any $j \geq 0$.

Theorem 5. *Suppose the functional J is twice differentiable, that Assumption 5 is satisfied, and that the Hessian $\nabla^2 J(u)$ is Lipschitz continuous in a neighborhood $\Psi(u_\star)$ of $u_\star$, i.e. there exists $L > 0$ such that*

$$\|\nabla^2 J(u) - \nabla^2 J(v)\| \leq L\|u - v\| \quad \forall u, v \in \Psi(u_\star). \tag{II.39}$$

Then, considering the iteration $u_{j+1} = u_j + p_j$ where p_j solves equation (II.38) *and with $u_0 \in \Psi(u_\star)$ being close enough to $u_\star$, then the rate of convergence of u_j is* ***quadratic****, i.e. there exists* $0 < M < 1$ *such that:*

$$\lim_{j\to\infty} \frac{\|u_{j+1} - u_\star\|}{\|u_j - u_\star\|^2} < M. \tag{II.40}$$

Remark 1. *In the particular quadratic case of* (II.22)*, Newton's method converges in only one iteration, because $J(u) = \frac{1}{2}u^T Q u - b^T u$, $\nabla J(u) = Qu - b$, and $\nabla^2 J(u) = Q$. So equation* (II.38) *writes $Qp_j = -Qu_j + b$, with solution $p_j = -u_j + Q^{-1}b$, and finally the recursion $u_{j+1} = u_j + p_j$ reads for $j = 0$:*

$$u_1 = u_0 + Q^{-1}b - u_0 = Q^{-1}b = u_\star, \tag{II.41}$$

that solves exactly the quadratic problem (II.22).

Remark 2 (on Terminology)**.** *The convergence of the Newton method is quadratic in the sense that $\|u_{j+1} - u_\star\| \leq M\|u_{j+1} - u_\star\|^2$ asymptotically as $j \to \infty$. Observe, however, that with respect to the number of iterations, such convergence is "super-exponential", i.e. $\|u_{j+1} - u_\star\| \leq M^{2^j - 1}\|u_0 - u_\star\|^{2^j} = \frac{1}{M}\rho^{2^j}$ with $\rho = M\|u_0 - u_\star\| < 1$ if $\|u_0 - u_\star\|$ is small enough (smaller then $1/M$).*

II.B.5. Coordinate descent

This method is based on *a priori* choice for the descent direction p_j. Specifically, we set $p_1 = e_1, p_2 = e_2, \ldots, p_j = e_{[j \mod n]}, \ldots$, where $0 \leq [j \mod n] \leq n - 1$ denotes the rest in the Euclidean division of j by n, and $\{e_0, \ldots, e_{n-1}\}$ denotes the euclidean basis of $\mathbb{R}^n$. We aim at minimizing the functional along every basis direction, until convergence. But this convergence is not always guaranteed, as shown in [Pow73]. The coordinate descent method, with exact line searches (i.e. with an optimal step-size, alongside every direction p_j) can iterate infinitely without converging to any point where the gradient vanishes. Nevertheless, as stated in [NW99], the two major advantages of such a method are:

- we don't require any computation of the gradient ∇J_j
- the speed of convergence is usually *not too bad*/acceptable, if the variables are loosely coupled.

II.B.6. Conjugate gradient

This method was introduced by Hestenes ans Stiefel in 1950, and has the key feature of being faster than the steepest descent method, with the same no matrix storage requirement. Back in the quadratic problem setting (II.22), with a symmetric, positive definite matrix $Q \in \mathbb{R}^{n\times n}$, we introduce the notion of *conjugacy*, for a set of vectors:

Definition 2. *A set of vectors $\{p_0, p_1, \dots, p_l\}$ is said to be* conjugate *w.r.t. the symmetric positive definite matrix Q if*

$$p_i^T Q p_k = 0, \quad \forall i \neq k. \tag{II.42}$$

Now the question is how the directions p_j are chosen. As we shall limit the memory space required, the goal is to compute the next direction p_j, only from the previous one, p_{j-1}, in order to get an update formula with single storage. This is the fundamental property of the conjugate gradient method. For a formal definition, let us define the linear system associated to the quadratic minimization problem (II.22)

$$Qu = b \tag{II.43}$$

and its residual by $r(u) = Qu - b = \nabla J(u)$. This quantity quantifies how close the state u is from the optimal solution. We denote by $r_j = Qu_j - b$, the residual at iteration j and define the search directions for $j \geq 1$ as

$$p_j = -r_j + \beta_j p_{j-1}, \tag{II.44}$$

with a scalar β_j guaranteeing the conjugacy property, i.e. $i \neq k \Rightarrow p_i^T Q p_k = 0$. We find that

$$\beta_j = \frac{r_j^T Q p_{j-1}}{p_{j-1}^T Q p_{j-1}}. \tag{II.45}$$

The first direction p_0 is equal to the steepest descent computed at the initial point u_0:

$$p_0 = \nabla J(u_0). \tag{II.46}$$

One main advantage of such a method, is to guarantee the convergence to the solution of system (II.43), **in at most n steps**.

Nevertheless, when the matrix Q is of large size, or when round-off errors affects the numerical computation of the conjugate directions, this exactness property does not hold anymore, or is not so appealing. The following Theorem recalls a convergence bound, still valid in high dimension.

Theorem 6. *Following again the scheme* (II.18)*, we can bound the error in norm Q as:*

$$\|u_j - u_\star\|_Q \leq \frac{2c^j}{1+2c^{2j}}\|u_0 - u_\star\|_Q$$

with $c = \frac{\sqrt{\kappa(Q)}-1}{\sqrt{\kappa(Q)}+1}$, showing again exponential convergence in j.

Proof. A proof of this result can be found in [Hac16, Theorem 10.14] □

Notice that the rate of convergence $c = \frac{\sqrt{\kappa(Q)}-1}{\sqrt{\kappa(Q)}+1} \approx 1 - \frac{2}{\sqrt{\kappa(Q)}}$ as $\kappa(Q) \to \infty$, i.e. this conjugate gradient algorithm is faster, than the steepest descent algorithm (presented in Section II.B.2), for problems with a high condition number $\kappa(Q)$.

This conjugate gradient (CG) algorithm looks very promising, in order to solve the discrete form of the OCP (II.16), as it guarantees the fastest (exponential) rate of convergence among the previously enumerated optimization methods, without requiring the computation of the exact Hessian of the functional J. The coordinate descent method is not considered in this framework, as the dimensionality of the problem plays a crucial role in its convergence speed.

The (exact) Newton method is not really an option to solve the OCP (II.16). It converges in one iteration but requires to assemble the full Hessian matrix which implies computing explicitly $(A^h)^{-1}$, not feasible for large scale problems. One could alternatively solve the linear system involving the full Hessian by a matrix-free iterative method (Newton-Krylov) such as Conjugate Gradient, but this brings back to the CG method discussed above.

Quasi-Newton methods are an alternative to gradient of CG methods. Based on equation (II.19), they require computing the search direction p_j by inverting the matrix Ω_j. Even when the Sherman-Morrison-Woodbury formula is used to update the inverse $H_j = \Omega_j^{-1}$, the cost for computing p_j increases over the iterations since the rank of H_j increases, as stated in [BS12]. This makes the analysis of the complexity of Quasi-Newton methods more cumbersome than gradient or conjugate ones and is the reason why we have not considered these methods in this work.

Finally, when considering OCPs under uncertainty, (e.g. see the convergence rate in Section III.H), the main limitation in the rate of convergence comes from the discretization in probability and/or physical space, and the error introduced by the optimization algorithm is usually negligible, when the convergence is exponential in the number of iterations. For this reason we will mostly use the steepest descent strategy in the rest of this work, which guarantees a simple framework to derive theoretical error bounds and convergence rates for all sources of errors.

II.C. Risk averse optimization

II.C.1. Stochastic modeling

Starting from the deterministic OCP (II.1), we recall that the functional f we aim at minimizing is

$$f(y,u) = \frac{1}{2}\|y - z_d\|^2_{\mathcal{X}} + \frac{\beta}{2}\|u\|^2_{\mathcal{U}}.$$

However the deterministic framework may reach its limits, for example in real life problems, where the system itself might be random/uncertain or its physical parameters estimation may include uncertainty from measurement errors (see e.g. Section II.C.2 for details and examples). The deterministic PDE (II.8) with Dirichlet boundary conditions can be re-written more realistically, from a stochastic viewpoint, as

$$\begin{cases} -\operatorname{div}(a(x,\omega)\nabla y(x,\omega) = g(x) + u(x) & \text{for } x \in D \text{ a.e. } \omega \in \Gamma \\ y(x,\omega) = 0 & \text{for } x \in \partial D \text{ a.e. } \omega \in \Gamma. \end{cases} \tag{II.47}$$

where $(\Gamma, \mathcal{F}, \mathbb{P})$ is a complete probability space. Here, $\omega \in \Gamma$ represents the uncertainty in the system, modeled in terms of a random variable with probability measure $\mathbb{P}$ on Γ. Then, the associated deterministic functional f becomes stochastic as well, because of its dependence on ω through $y = y_\omega$

$$f(y_\omega, u) = \frac{1}{2}\|y_\omega - z_d\|^2_{\mathcal{X}} + \frac{\beta}{2}\|u\|^2_{\mathcal{U}}. \tag{II.48}$$

At this point, we face an optimization problem with a stochastic functional. Different approaches have been considered in the literature:

- if we are able to observe the realization ω of the random variable, and are able to compute the associated optimal control $u = u_\omega$ afterwards, for each realization (e.g. design multiple kind of turbines, depending on the daily running conditions, and adapt them in function of the conditions), then we can derive a stochastic control u_ω;

- on the other hand, if the randomness is not observable or, in other terms, the optimal control has to be computed before observing any realization of the system output, then we tackle a robust-optimization problem and we need to design one control u, being robust in some statistical sense w.r.t. the distribution of the functional f. Think for instance at the design of a wing of an aircraft, being efficient and reliable in almost every in-flight condition. For this purpose, we incorporate a risk measure σ into a Quantity of Interest J, such that $J = \sigma \circ f$, in order to choose the control u, being more or less risk averse to the stochasticity in the system, based on the choice of the risk measure σ.

In the following, we will only focus on the second approach, i.e. the robust optimization problems.

II.C.2. Main ideas in risk averse optimization

Uncertainty arises particularly in the field of physical modeling, because of a lack of knowledge of the parameters of the system, or because the system itself has some unpredictable behavior that can be described only in a probabilistic/statistical sense. For instance, the process of manufacturing an iron element, in fact produces a slightly different object than the designed one, because of manufacturing uncertainty. One can model this lack of precision, for example assigning a probability distribution on the shape/dimension/-topology of the actual produced element. Another context where uncertainty is very relevant is, for instance the design optimization of a turbine/airfoil/wing, under uncertain operating conditions. One shall model the uncertain environment with a statistical approach from recorded data and history matching, and quantify the resulting uncertainty in aerodynamic quantities such as lift or drag coefficient by performing several forward simulations and estimating the output probability distribution.

In an optimization problem, a quantity of interest to be optimized is introduced, and when facing a robust optimization approach, this quantity should only involve some statistics of the system output when we do not have access to the aleatory conditions beforehand. Thus, we need to introduce a map called a *risk measure* (that quantifies the level of aversion/unwillingness/hostility to the risk), and compose it with the stochastic objective functional of the uncertain system. Here, *risk* refers to a measure of the variation of the objective functional w.r.t. the random data of the model. For example, consider a stochastic objective functional

$$\begin{aligned} \widetilde{f} : \mathcal{U} \times \Gamma &\longrightarrow \mathbb{R} \\ (u, \omega) &\longmapsto \widetilde{f}(u, \omega), \end{aligned}$$

where $u \in \mathcal{U}$ is the control function (that generalize the shape/design previously evoked). In the context of PDE-constrained optimization under uncertainty, and objective functional (II.48), $\widetilde{f}$ reads $\widetilde{f}(u,\omega) = f(y_\omega(u), u)$ where $y_\omega(u)$ solves the elliptic random PDE (II.47). Here, $\mathcal{U}$ denotes the control space and $(\Gamma, \mathcal{F}, \mathbb{P})$ a complete probability space. Then one faces the trade-off between minimizing the mean $\mathbb{E}[\widetilde{f}(u,\cdot)]$ describing the expected outcome, and the risk (variability measure) $\mathbb{D}[\widetilde{f}(u,\cdot)]$ which quantifies the *deviation* (uncertainty) of the outcome. The formal definition of $\mathbb{D}$ is omitted here and detailed later on. The final quantity of interest may be a linear combination of the two quantities $\mathbb{E}[\widetilde{f}(u,\cdot)]$ and $\mathbb{D}[\widetilde{f}(u,\cdot)]$, which defines a risk measure of the form

$$\sigma(\widetilde{f}(u,\cdot)) := \mathbb{E}[\widetilde{f}(u,\cdot)] + c\mathbb{D}[\widetilde{f}(u,\cdot)],$$

where the parameter c reflects the *price of the risk*, or *risk aversion*. A high value of c penalizes more the deviation term $\mathbb{D}[\widetilde{f}(u,\cdot)]$ leading to a smaller variability of the outcome. The robust minimization problem then reads:

$$\min_{u \in \mathcal{U}} J(u), \quad J(u) = \mathbb{E}[\widetilde{f}(u,\cdot)] + c\mathbb{D}[\widetilde{f}(u,\cdot)]. \tag{II.49}$$

When the coefficient c varies, we generate an ensemble of minimization problems, from the most conservative one (i.e. $c \to \infty$ corresponding to the worst-case oriented risk-measure), up to the most *risky* one (when we don't pay attention to any variability of the quantity $\omega \mapsto \widetilde{f}(u_\star, \omega)$ around the optimal control) i.e. when $c = 0$. The variability measure $\mathbb{D}$ quantifies, to some extent, the width and thickness of the tail distribution of the stochastic objective functional.

A particularly convenient class of risk measures are the so-called *coherent risk measures* [RS06, RS07, SDR09], since they exhibit desirable properties, such as monotonicity or convexity. These two properties are required for example to transfer the convexity property from $\widetilde{f}$ to J. We define formally, in the following Section, the *coherent risk measures* and present a non-exhaustive list of risk measures, discussing advantages and limitations of each of them.

II.C.3. Coherent Risk Measure

On a complete probability space $(\Gamma, \mathcal{F}, \mathbb{P})$, we consider a random variable $Z = Z(\omega)$, with $\omega \in \Gamma$. The term *risk measure* refers to a map from the set of random variables (r.v.) set $\{Z : \Gamma \to \mathbb{R}, \mathcal{F} - \textit{measurable}\}$, onto the real line $\mathbb{R}$, assuming, some integrability property of the random variable Z. For this purpose, let us denote $\mathcal{Z} = L^p(\Gamma, \mathcal{F}, \mathbb{P})$ with $1 \leq p < \infty$ the space of random variables Z for which the p-th order moment with respect to the reference probability measure $\mathbb{P}$ is finite, that is $\mathcal{Z} = \{Z : \Gamma \to \mathbb{R},\ \ \mathcal{F} - \text{measurable},\ \text{s.t.}\ \int_\Gamma |Z(\omega)|^p \mathrm{d}\mathbb{P}(\omega) < +\infty\}$. We thus implicitly assume that the mapping $\sigma : \mathcal{Z} \to \mathbb{R}$ is defined on equivalence classes of functions that differ on sets of

$\mathbb{P}$-zero measure, i.e. for any $Z_1, Z_2 \in \mathcal{Z}$, $\sigma(Z_1) = \sigma(Z_2)$ if $\mathbb{P}\{\omega \in \Gamma : Z_1(\omega) \neq Z_2(\omega)\} = 0$. For technical convenience, we assume throughout this work, that

- $\sigma : \mathcal{Z} \to \mathbb{R}$ is *proper*, i.e. $\sigma(Z) > -\infty$ for all $Z \in \mathcal{Z}$
- the domain of σ is non-empty, i.e. $\mathrm{dom}(\sigma) = \{Z \in \mathcal{Z} : \sigma(Z) < +\infty\} \neq \emptyset$.

We state now the four properties defining a *coherent* risk measure.

Definition 3. *(coherent risk measure). A proper, non-empty domain map $\sigma : \mathcal{Z} \to \mathbb{R}$ is called coherent risk measure, in the sense of Artzner, Delbaen, Eber, and Heath [ADEH99] if it satisfies the following properties:*

- ***Convexity**: For all $Z_1, Z_2 \in \mathcal{Z}$ and $\lambda \in [0, 1]$,*

$$\sigma(\lambda Z_1 + (1-\lambda)Z_2) \leq \lambda\sigma(Z_1) + (1-\lambda)\sigma(Z_2); \tag{II.50}$$

- ***Monotonicity**: For all $Z_1, Z_2 \in \mathcal{Z}$:*

$$Z_1 \leq Z_2 \quad \mathbb{P}-\text{a.e. in } \Gamma \quad \Rightarrow \sigma(Z_1) \leq \sigma(Z_2); \tag{II.51}$$

- ***Translation Equivariance**: For all $Z \in \mathcal{Z}$, and $c \in \mathbb{R}$,*

$$\sigma(Z + c) = \sigma(Z) + c; \tag{II.52}$$

- ***Positive Homogeneity**: For all $Z \in \mathcal{Z}$, and $c > 0$,*

$$\sigma(cZ) = c\sigma(Z). \tag{II.53}$$

II.C.4. Example of risk measures

The mean-variance risk measure

We present here the most common risk measure found in literature, the mean-variance risk, incorporating a parameter $c \geq 0$ to scale the importance of the variance, i.e. the risk aversion, as discussed before. For a r.v. $Z \in \mathcal{Z} = L^2(\Gamma, \mathcal{F}, \mathbb{P})$, let us introduce the *mean-variance risk measure* as:

$$\sigma(Z) = \mathbb{E}[Z] + c\mathbb{E}\left[(Z - \mathbb{E}[Z])^2\right]. \tag{II.54}$$

It is straightforward to show that the mean-variance risk measure is convex, has the translation equivariance property, but it fails to satisfy the positive homogeneous, and the monotonicity properties, if $c > 0$.

Remark 3. *The particular case $c = 0$ leads to the risk neutral (or mean-based risk) measure $\sigma(Z) = \mathbb{E}[Z]$, which is obviously coherent.*

The mean-deviation risk measure of order p

Here we generalize the mean-variance risk measure, by replacing the variance term, with a deviation term, that involves moments of order p, i.e. given a r.v. $Z \in \mathcal{Z} = L^p(\Gamma, \mathcal{F}, \mathbb{P})$, for $p \geq 1$, we define the mean-deviation risk measure of order p as:

$$\sigma(Z) = \mathbb{E}[Z] + c\mathbb{E}\left[|Z - \mathbb{E}[Z]|^p\right]^{\frac{1}{p}} . \tag{II.55}$$

This risk measure remains convex, has the translation equivariance and the positive homogeneous property (thanks to the power $1/p$ to make the deviation homogeneous to the mean), but still fails to be monotonic, in general, when $p > 1$. Completing the description of this risk measure, based on [SDR09], the mean-deviation risk measure of order $p = 1$ is coherent, if and only if the risk aversion parameter $c \in [0, 1/2]$, assuming that $\mathcal{F}$ contains events A with arbitrarily small measure $\mathbb{P}(A)$ (which is guarantee, in particular if $\mathbb{P}$ is non-atomic).

The main drawback of the mean-deviation risk measure of order p, as well as the mean-variance risk measure, is that they both count the excess/surplus over the mean, as much as any scarcity/shortcoming/insufficiency over it. This symmetric property may limit considerably the use of such risk measures, because in many engineering scenarios, a surplus and a lack should not be counted the same way: one might be critical whereas the other does not matter.

Other risk measures involve only semi-deviations, that count excess (or shortcomings) over a desirable value.

The mean-upper semi-deviation of order p risk measure

For a r.v. $Z \in \mathcal{Z} = L^p(\Gamma, \mathcal{F}, \mathbb{P})$, we define the mean-upper semi-deviation of order p by

$$\sigma(Z) = \mathbb{E}[Z] + c\mathbb{E}\left[(Z - \mathbb{E}[Z])_+^p\right]^{\frac{1}{p}} . \tag{II.56}$$

with $(Z)_+ = \max\{0, Z\}$. This risk measure is convex, has the translation equivariance and the positive homogeneous property. However, following [SDR09], it is monotonic if $c \in [0, 1]$. Moreover if $\mathcal{F}$ contains events A with arbitrarily small measure $\mathbb{P}(A)$, then the condition $c \in [0, 1]$ is also necessary for monotonicity.

The Conditional Value at Risk (CVaR) risk measure

Besides moment based risk measures, a risk measure can also be based on quantiles. For a given parameter $\alpha \in (0,1)$ and a r.v. $Z \in \mathcal{Z} = L^1(\Gamma, \mathcal{F}, \mathbb{P})$, we recall the Value at Risk (VaR) threshold (α-quantile), defined as:

$$\mathrm{VaR}_\alpha[Z] = \inf\{t \in \mathbb{R} : \mathbb{P}(Z \leq t) \geq 1 - \alpha\}. \tag{II.57}$$

This risk measure is mainly used in finance. This community often uses also the so-called Conditional Value at Risk (CVaR), defined as:

$$\mathrm{CVaR}_\alpha[Z] = \inf_{t \in \mathbb{R}}\{t + \alpha^{-1}\mathbb{E}[(Z-t)_+]\}, \tag{II.58}$$

We can think at the CVaR_α as the mean of the r.v. Z over the $\mathrm{VaR}_\alpha[Z]$ threshold, i.e.

$$\mathrm{CVaR}_\alpha[Z] = \mathbb{E}[Z | Z \geq \mathrm{VaR}_\alpha[Z]]. \tag{II.59}$$

This risk measure is coherent, what makes it very convenient for engineering optimization problems. Moreover it combines every desirable property for a risk measure, such as it only counts excesses over the $\mathrm{VaR}_\alpha[Z]$, based on the *positive part* function $(\cdot)_+$. The next Section presents the OCPs, incorporating a risk measure σ in the quantity of interest.

II.C.5. Robust OCPs, incorporating a risk measure

Consider an OCP constrained by a PDE with random parameters. Let us call $y(u)$ the random solution of the PDE for a given control function $u \in \mathcal{U}$, and assume that such random solution $y(u) : \Gamma \to \mathcal{Y}$ has bounded q-th order moments for any $u \in \mathcal{U}$, i.e. $y(u) \in \Theta := L^q(\Gamma, \mathcal{F}, \mathbb{P}; \mathcal{Y})$. Let, moreover $\widehat{f}$ denote the random objective functional (function of $y(u)$) that we want to minimize in a robust sense, defined as

$$\begin{aligned} \widehat{f} : \Theta &\longrightarrow \mathcal{Z} := L^r(\Gamma, \mathcal{F}, \mathbb{P}), \quad \text{for some } r \geq 1. \\ y &\longmapsto \widehat{f}(y) \end{aligned}$$

Given a risk measure $\sigma : L^p(\Gamma, \mathcal{F}, \mathbb{P}) \to \overline{\mathbb{R}} = \mathbb{R} \cup \{\infty\}$ with $p \leq r$, we define the risk averse PDE-constrained OCP, incorporating the risk measure, as

$$\min_{u \in \mathcal{U}} \widehat{J}(u) = \sigma[\widehat{f}\,(y(u))] + \varsigma(u) \tag{II.60}$$

where $\varsigma : \mathcal{U} \to \mathbb{R}$ is a regularization term.

Numerical methods for robust OCPs, typically depend on the choice of the risk measure. For example, the work [APSG17] considers a risk measure that involves the mean and variance of the objective functional and uses second order Taylor expansions combined with randomized estimators to reduce the computational effort. The work [VBV18] considers a risk measure that involves only the mean of the objective functional (hereafter named mean-based risk), with an additional penalty on the variance of the state, and proposes a gradient type method, in which the expectation of the gradient is computed by a multilevel Monte Carlo method. In [BvW11], the authors also consider a mean-based risk problem and propose a reduced basis method on the space of controls to dramatically reduce the computational effort. In the work [Kou12], the author presents a more general type of OCP, using the general notion of a risk measure, and derives the corresponding optimality system of PDEs to be solved. For its numerical solution, a trust-region Newton conjugate gradient algorithm is proposed in [KHRvBW13], combined with an adaptive sparse grid collocation for the discretization of the PDE in the stochastic space. The work [KS16] considers derivative-based optimization methods for the robust CVaR risk measure, which are built upon introducing smooth approximations to the CVaR. Finally, in the work [GLL11], the authors consider a boundary OCP where the deterministic control appears as a Neumann boundary condition. Using again a mean-based risk, they derive an optimality system of equations and provide a complete error analysis of the finite element approximation, as well as of the random parameter space approximation.

II.C.6. Existence result for robust OCPs under uncertainty

We present here a result on existence of an optimal control, for OCPs of the form (II.60) constrained by stochastic equation that induces, for any admissible control function $u \in \mathcal{U}_{ad}$, a random map $y(u) : \Gamma \to \mathcal{Y}$. For consistency of notation, we write $y \in \Theta$ when considered as a random field, and $\underline{y} \in \mathcal{Y}$ when we refer to a realization of the latter map, namely $y(\omega) = \underline{y}$, for some $\omega \in \Gamma$. We now recall the Nemytskii (superposition) operator: for $\Theta \ni y : \Gamma \to \mathcal{Y}$, we write

$$\left[\widehat{f}(y)\right](\omega) = \overline{f}(y(\omega), \omega),$$

where the so-called *parametrized random objective functional* $\overline{f}$, maps an element $(\underline{y}, \omega) \in \mathcal{Y} \times \Gamma$ into the real line $\overline{\mathbb{R}} := \mathbb{R} \cup \{\infty\}$.

$$\begin{aligned} \overline{f} : \mathcal{Y} \times \Gamma &\longrightarrow \qquad \mathbb{R} \\ (\underline{y}, \omega) &\longmapsto \overline{f}(\underline{y}, \omega). \end{aligned}$$

Based on [KS18], we introduce some properties of the solution map y:

Assumption 6.

1. *There exists $q \geq 1$ such that for all $u \in \mathcal{U}_{ad}$, $y(u) : \Gamma \to \mathcal{Y}$ is strongly $\mathcal{F}$-measurable, and $y(u) \in \Theta = L^q(\Gamma, \mathcal{F}, \mathbb{P}; \mathcal{Y})$.*

2. *There exist a non negative increasing function $\rho : [0, \infty) \to [0, \infty)$ and a non*

negative random variable $C \in L^q(\Gamma, \mathcal{F}, \mathbb{P})$ *satisfying*

$$\|y(u)\|_{\mathcal{Y}} \leq C\rho\left(\|u\|_{\mathcal{U}}\right) \quad \mathbb{P} - a.e.$$

for all $u \in \mathcal{U}_{ad}$.

3. *If* $u_n \rightharpoonup u$ *in* $\mathcal{U}_{ad}$, *then* $y(u_n) \rightharpoonup y(u)$ *in* $\mathcal{Y}$ $\mathbb{P}$-*a.e.*

We now state a differentiability assumption on the map y.

Assumption 7. *There exists an open set* $V \subseteq \mathcal{U}$ *with* $\mathcal{U}_{ad} \subseteq V$ *such that the solution map* $u \mapsto y(u) : V \to \Theta$ *is continuously Fréchet differentiable.*

Now we state some assumptions about the objective functional $\overline{f} : \mathcal{Y} \times \Gamma \to \mathbb{R}$.

Assumption 8.

1. *Carathéodory.* $\overline{f}$ *is a Carathéodory function; i.e.,* $\overline{f}(\cdot, \omega)$ *is continuous for* $\mathbb{P}$-*a.e.* $\omega \in \Gamma$ *and* $\overline{f}(\underline{y}, \cdot)$ *is measurable for all* $\underline{y} \in \mathcal{Y}$.
2. *There exists* $p \geq 1$ *such that, for all* $y \in \Theta$, *the r.v.* $\left[\widehat{f}(y)\right](\omega) = \overline{f}(y(\omega), \omega)$ *has bounded moments up to order* p, *i.e.* $\widehat{f}(y) \in L^p(\Gamma, \mathcal{F}, \mathbb{P})$
3. *Growth condition. There exist* $a \in L^p(\Gamma, \mathcal{F}, \mathbb{P})$ *with* $a \geq 0$ $\mathbb{P}$-*a.e. and* $c > 0$ *such that for all* $\omega \in \Gamma$ *and all* $\underline{y} \in \mathcal{Y}$
$$|\overline{f}(\underline{y}, \omega)| \leq a(\omega) + c\|\underline{y}\|_{\mathcal{Y}}^{q/p}.$$
4. $\overline{f}(\cdot, \omega)$ *is convex for* $\mathbb{P}$-*a.e.* $\omega \in \Gamma$.

We are now in position to state the theorem, from [KS18, Proposition 3.12.], guaranteeing existence of an optimal control, for the OCP (II.60).

Theorem 7. *Let Assumptions 6, 7, and 8 hold. Let* $\sigma : L^p(\Gamma, \mathcal{F}, \mathbb{P}) \to \overline{\mathbb{R}} = \mathbb{R} \cup \{\infty\}$ *be a proper, lower semi-continuous, convex, and monotonic risk measure, and let* $\varsigma : \mathcal{U} \to \mathbb{R}$ *be proper, lower semi-continuous, and convex. Finally, suppose either* $\mathcal{U}_{ad}$ *is bounded or* $u \mapsto \sigma\left[\widehat{f}(y(u))\right] + \varsigma(u)$ *is coercive. Then* (II.60) *has a solution.*

We now particularize the general setting (II.60) to the OCP described in Section II.C.1, constrained by an elliptic random PDE. A risk averse formulation implies choosing a coherent risk measure and composing it with the random objective functional $f(y_\omega(u), u)$ of equation (II.48):

$$\min_{u \in \mathcal{U}} J(u) := \sigma\left[f\left(y_\omega(u), u\right)\right]. \tag{II.61}$$

The quantity of interest J in equation (II.61) can be re-written, in the form of (II.60), by setting $\widehat{f}(y(u))(\omega) = \frac{1}{2}\|y_\omega(u) - z_d\|^2_{\mathcal{X}}$, where $y_\omega(u)$ solves the random PDE constraint (II.47), and $\varsigma(u) = \frac{\beta}{2}\|u\|_{\mathcal{U}}$, so that

$$f(y_\omega(u), u) = \widehat{f}(y(u))(\omega) + \varsigma(u).$$

Since the risk measure σ has the translation equivariance property, the constant quantity $\varsigma(u)$ can be put equivalently inside or outside the coherent risk measure σ. Then following [KS18], as it can be shown that the assumptions of Theorem 7 hold true, in this particular setting, there exists an optimal control for the PDE-constrained OCPs (II.61). A simplified version of this result is proved, in a more straightforward manner, in Chapter III, in the case of the mean-based risk measure $\sigma = \mathbb{E}$.

We present in the next Section some common optimization techniques, widely used by the machine learning community, for mean-based stochastic optimization problems.

II.D. Machine Learning (ML) Optimization methods

In the Machine Learning (ML) community, one usually faces the problem of forecasting/predicting a quantity say z, from associated inputs θ, using historical data of (θ, z). Specifically, suppose we have access to historical data points

$$\Xi^{\text{train}} = \{(\theta_1, z_1), (\theta_2, z_2), \ldots, (\theta_n, z_n)\},$$

we would like to infer a predictive model $\widehat{y}$, based on Ξ^{train}, that is able to "explain" the data output z_i, from its associated input θ_i, for all $i \in \{1, \ldots, n\}$. Here Ξ^{train} is called the *training data set*.

II.D.1. ML problem setting

In order to propose a probabilistic framework (see e.g. [HTF01]), we denote by $\xi \in \Gamma \subset \mathbb{R}^q$ a random input vector, and $Z \in \mathcal{X}$, a $\mathcal{X}$-valued random output variable, with joint distribution $\Pr(\mathrm{d}\theta, \mathrm{d}z)$. We will call by $(\theta, z) \in \Gamma \times \mathcal{X}$ any realization of the r.v. (ξ, Z). The major assumption, in the *supervised learning* approach, is that there exists a *true* model $y \in \Delta$, with Δ a family of maps $y : \Gamma \to \mathcal{X}$ that link the random variables ξ and Z by the *exact* equation $y(\xi) = Z$. In particular, here, we omit any observation error on Z. The inputs $\theta_i \in \Gamma \subset \mathbb{R}^q$ in the training set represent the so-called *features*, while $z_i \in \mathcal{X}$ their associated *label*. For instance, when $\mathcal{X} = \{1, \ldots, C\}$, we can build a multi-class of the input variables θ_i in Ξ^{train} through its associated label/output $z_i \in \{1, \ldots, C\}$ by defining the C classes

$$\{\theta_i \in \Gamma : y(\theta_i) = k\}_{k=1,\ldots,C}. \tag{II.62}$$

Here we cluster the features, i.e. split the input data in the training set, into C classes, based on its label. In order to extend this classification, on the whole set Γ, i.e. in order to classify any new feature θ_{n+1}, we may generalize them by plugging the approximated predictive model $\widehat{y}$, in place of y in (II.62), i.e. the C classes become

$$\{\theta \in \Gamma : \widehat{y}(\theta) = k\}_{k=1,\dots,C}. \tag{II.63}$$

In the supervised learning approach, the goal is to *learn* the function/model $\widehat{y} : \Gamma \to \mathcal{X}$, given a collection of labeled input-output pairs

$$\Xi^{\text{train}} = \{(\theta_1, z_1), (\theta_2, z_2), \dots, (\theta_n, z_n)\}$$

which are assumed to be iid realizations of the r.v. (ξ, Z). The ultimate goal of such ML problem, is to construct a predictive (approximated) model $\widehat{y} \in \Delta$, based on the training set Ξ^{train}, that maps the input/features space Γ into the label set $\mathcal{X}$. This stage is called the *learning* stage, and is presented in the next Section.

II.D.2. Learning stage

One way to formalize the learning stage in a ML framework is by using function approximation (see e.g. [Mur12]). Assuming that $y \in \Delta$ is the true yet unknown model, the goal of *learning* is to approximate the exact function y, given a labeled training data set Ξ^{train}, and then classify any new feature θ_{n+1}, using the predicted label $\widehat{z}_{n+1} = \widehat{y}(\theta_{n+1})$. This theory requires introducing a *loss function* i.e. a mapping $l : \Delta \times (\Gamma \times \mathcal{X}) \to \mathbb{R}$ where Δ is a set of admissible models $\widehat{y} : \Gamma \to \mathcal{X}$. Specifically, the loss function l is such that $l(\widehat{y}, (\xi, Z))$ increases when $\widehat{y}$ is not an accurate predictor of the exact behavior of the r.v. (ξ, Z). In ML, it is customary to define the best model $\widehat{y}^* \in \Delta$ as the one that minimizes the *expected loss* w.r.t. the distribution $\Pr(\mathrm{d}\theta, \mathrm{d}z)$ of the r.v. (ξ, Z). Thus, the optimization problem writes

$$\widehat{y}^* \in \operatorname*{arg\,min}_{\widehat{y} \in \Delta} R(\widehat{y}). \tag{II.64}$$

where the risk $R(\widehat{y})$ is defined as

$$R(\widehat{y}) := \mathbb{E}_{(\xi,Z)}[l\left(\widehat{y}, (\xi, Z)\right)] = \int l\left(\widehat{y}, (\theta, z)\right) \Pr(\mathrm{d}\theta, \mathrm{d}z). \tag{II.65}$$

Unfortunately, the exact distribution of the r.v. (ξ, Z) is usually unknown, or if it is known, problem (II.64) is rarely tractable, from a numerical point of view. Thus, we usually replace problem (II.64), with its *empirical version*, based on the n iid training data points of the training set Ξ^{train}, i.e.

$$R(\widehat{y}) \approx \widehat{R}(\widehat{y}, \Xi^{\text{train}}) := \frac{1}{n}\sum_{i=1}^{n} l(\widehat{y}, (\theta_i, z_i)), \tag{II.66}$$

leading to the approximated optimization problem

$$\widehat{y}^* \in \arg\min_{\widehat{y}\in\Delta} \widehat{R}(\widehat{y}, \Xi^{\text{train}}). \tag{II.67}$$

In the case where the distribution of (ξ, Z) is known, one may still end up with the approximate optimization problem (II.67) if the expectation in (II.64) is replaced by a Monte Carlo estimator.

In the next Section, we present some desirable properties of the approximated model $\widehat{y}^*$, solution of the optimization problem (II.67), in terms of accuracy of $\widehat{y}^*$ on reproducing the training data set Ξ^{train} (bias), and in terms of variability of $\widehat{y}^*$ when the training set changes (variance).

II.D.3. Bias-Variance dilemma

The ML community usually looks for an approximated model $\widehat{y}^* \in \Delta$, that is:

- the most accurate for prediction on the training data set Ξ^{train} i.e. with small bias, guaranteeing that we extracted well the information from Ξ^{train};
- the most *robust* w.r.t. the training data selection i.e. with small variance, guaranteeing that the optimal model $\widehat{y}^*$ can be generalized to predict the new label of a new feature point θ_{n+1}.

Unfortunately, we cannot reduce the bias without increasing the variance, and *vice-versa*. The trade-off between how we should penalize high bias/variance is the big role of the ML engineer, at the training stage. Over-fitting occurs when the variance is too high, achieving a too small bias. Then the model over-fits the training data points, but is not robust to other training data sets, and, by extension, would not guarantee a good prediction of the model, for a new feature point θ_{n+1}.

A common method to limit over-fitting is to impoverish the set of admissible functions Δ, for example by restricting it to some subset, described by a finite dimensional parameter vector, say $u \in \mathcal{U}$, i.e. each function $\widehat{y}$ in the subset is parametrized by some parameter $u \in \mathcal{U}$ and the restricted set of admissible functions reads:

$$\Delta_{\mathcal{U}} = \{\widehat{y}(u;\cdot) \in \Delta \ \text{ s.t. } \theta \mapsto \widehat{y}(u;\theta) \in \mathcal{X}; \quad \forall u \in \mathcal{U}\} \subsetneq \Delta.$$

where the map $u \mapsto \widehat{y}(u;\cdot)$ is let vague at this stage. Then the ML optimization problem can be equivalently reformulated in terms of the parameter set $\mathcal{U}$, as:

$$\min_{u\in\mathcal{U}} \frac{1}{n}\sum_{i=1}^{n} l(\widehat{y}(u;\cdot),(\theta_i,z_i)). \tag{II.68}$$

Next Section present specific choices for the loss function l.

II.D.4. Examples of loss functions

Different examples of loss functions l are used in practice, based on the specific problem that one would like to solve. For instance, in a classification problem $\mathcal{X} = \{1,\ldots,C\}$, if we wish to *classify* new features θ_{n+1} based on the training data set Ξ^{train}, we usually use the error count function l, defined as:

$$l(\widehat{y}(u;\cdot),(\theta,z)) = 1_{\{\widehat{y}(u;\theta)\neq z\}}, \tag{II.69}$$

The optimization procedure on the training data Ξ^{train} tunes the parameters u of the function $\widehat{y}(u;\cdot)$, in order to make the number of errors on Ξ^{train}, i.e. the number of misclassified feature points θ_i whose predicted label $\widehat{z_i} = \widehat{y}(u;\theta_i)$ differs from the exact one, z_i, as small as possible.

From another perspective, if we want to design a relation model between the features θ and their associated labels $z \in \mathcal{X}$, we utilize instead the squared error loss function:

$$l(\widehat{y}(u;\cdot),(\theta,z)) = (\widehat{y}(u;\theta)-z)^2, \tag{II.70}$$

and call this framework *regression problem*. Based on a large training data set Ξ^{train}, the problem is then of minimizing a large sum of functions, of the form:

$$\widehat{J}(u) := \widehat{R}(\widehat{y}(u;\cdot),\Xi^{\text{train}}) + \varsigma(u) = \frac{1}{n}\sum_{i=1}^{n} l(\widehat{y}(u;\cdot),(\theta_i,z_i)) + \varsigma(u), \tag{II.71}$$

where, we added to the empirical risk $\widehat{R}$, in (II.71), a regularization term ς, in order to avoid over-fitting on the training data set. Usually $u \mapsto \varsigma(u)$ is an increasing function of a specific norm of u, depending on the desired properties of the minimizer u we would like to reach/guarantee. For instance, the L^1 norm is used to encourage sparsity in the parameter u; other norms could be related to smoothness properties of the function $\widehat{y}(u,\cdot)$. The following Section makes the link with the specific OCP framework studied in this thesis; we choose the **regression** loss function, and particularize the sets Γ, $\mathcal{X}$, $\mathcal{U}$ and $\Delta_{\mathcal{U}}$.

II.D.5. Link with the studied OCP

We consider again the OCP presented in Sections II.C.1 and II.C.6, and aim now at recasting it into a ML framework. For convenience, we rewrite here the formulation of the robust OCP. It is based on the elliptic random PDE with Dirichlet boundary conditions

$$\begin{cases} -\operatorname{div}(a(x,\omega)\nabla y(x,\omega) = g(x) + u(x) & \text{for } x \in D \text{ a.e. } \omega \in \Gamma \\ y(x,\omega) = 0 & \text{for } x \in \partial D \text{ a.e. } \omega \in \Gamma. \end{cases} \tag{II.72}$$

and its associated *deterministic* functional J, using the *mean-based* risk measure:

$$J(u) = \mathbb{E}_\omega[f(y_\omega(u), u)] = \frac{1}{2}\mathbb{E}_\omega\left[\|y_\omega(u) - z_d\|^2_{\mathcal{X}}\right] + \frac{\beta}{2}\|u\|^2_{\mathcal{U}}, \tag{II.73}$$

where the state function $y_\omega(u)$ solves the PDE (II.72) for every realization $\omega \in \Gamma$ and a given control $u \in \mathcal{U}$.

We assume moreover, that the spaces of controls $\mathcal{U}^h$ and state $\mathcal{Y}^h$ have been suitably discretized (see Section II.A.7) and omit in the following the superscript h. The OCP aims at finding the optimal control u for which the corresponding state $y_\omega(u)$ is as close as possible to the target function $z_d \in \mathcal{X}$. In a machine learning language, the input variable (feature) is $\omega \in \Gamma$ and the output variable (label) should ideally be z_d for any $\omega \in \Gamma$. We set therefore the data space $\Gamma \times \mathcal{X}$ and the data random variable $(\omega, z_d) \in \Gamma \times \mathcal{X}$. Observe that the input is random with probability measure $\mathbb{P}$ on Γ, whereas the output is deterministic, i.e. its probability measure is a Dirac mass in z_d.

Our predictive model, which links the feature space Γ to the output space $\mathcal{X}$, is then the map $\omega \mapsto y_\omega(u) = \widehat{y}(u;\omega)$, parameterized by the parameter $u \in \mathcal{U}$, and the set of admissible models Δ^{OCP} is the set of solutions of the PDE (II.72), where the parameter u acts as a forcing term, namely

$$\Delta^{OCP} := \{\widehat{y}(u;\cdot) : \omega \mapsto \widehat{y}(u;\omega) = A^{-1}(\omega)(g - Bu), \forall u \in \mathcal{U}\},$$

where $A(= A^h)$ and $B(= B^h)$ are the operators associated to the discretized version of the PDE (II.72), i.e. equation (II.16) (see Section II.A.7 for more detail).

Finally, the loss function is the regression-type one $l(\widehat{y}(u;\cdot),(\omega,z_d)) = \frac{1}{2}\|\widehat{y}(u;\omega) - z_d\|^2_{\mathcal{X}}$ and the ML formulation of the OCP reads

$$\min_{u\in\mathcal{U}} \mathbb{E}_\omega\left[l(\widehat{y}(u;\cdot),(\omega,z_d))\right] + \varsigma(u),$$

where $\varsigma(u) = \frac{\beta}{2}\|u\|^2_{\mathcal{U}}$ is a regularization term.

Its empirical version, when a training set $\Xi^{\text{train}} = \{(\omega_1, z_d), \ldots, (\omega_n, z_d)\}$ is generated e.g.

by Monte Carlo, reads:

$$\begin{aligned}\min_{u\in\mathcal{U}} \widehat{J}(u) = \widehat{R}(\widehat{y}(u;\cdot), \Xi^{\text{train}}) &= \frac{1}{n}\sum_{i=1}^{n} l(\widehat{y}(u;\cdot),(\omega_i, z_d)) + \varsigma(u)\\ &= \frac{1}{n}\sum_{i=1}^{n} \underbrace{\frac{1}{2}\|\widehat{y}(u;\omega_i) - z_d\|_{\mathcal{X}}^2 + \frac{\beta}{2}\|u\|_{\mathcal{U}}^2}_{:=g_i(u)}, \end{aligned} \tag{II.74}$$

where ω_i are iid copies of ω, and equation (II.74) is a Monte Carlo (MC) approximation of (II.73). In the following Section, we present common ML optimization methods to solve optimization problems involving an empirical risk and can be written in the form of optimizing a finite sample average, i.e.

$$\min_{u\in\mathcal{U}} J(u) = \frac{1}{n}\sum_{i=1}^{n} g_i(u). \tag{II.75}$$

with $g_i(u) = l(\widehat{y}(u;\cdot),(\omega_i, z_d)) + \varsigma(u)$. We assume, in particular, that the training set size is finite but large, $n \gg 1$ and each g_i is a strongly convex function.

II.D.6. The Full Gradient method

Different strategies have been proposed in the ML literature [SRB13] to solve the optimization problem (II.75), based on line search algorithms presented in subsection II.B.1. The Full Gradient (FG) strategy is based on computing at each iteration j, the full gradient of J, w.r.t. u. The *deterministic* FG algorithm is generated by:

$$\begin{aligned} u_{j+1} &= u_j - \tau_j \nabla J(u_j).\\ &= u_j - \frac{\tau_j}{n}\sum_{i=1}^{n} \nabla g_i(u_j). \end{aligned}$$

Theoretical convergence results for such a method can be derived using e.g. the argument in Theorem 3 under the following assumptions on the functions g_i's:

Assumption 9 (Lipschitz continuity)**.** *There exists a constant $Lip > 0$ such that:*

$$\|\nabla g_i(u) - \nabla g_i(v)\| \leq Lip\|u - v\| \quad \forall u, v \in \mathcal{U}, \quad \forall i \in \{1, \dots, n\}$$

Assumption 10 (Strong convexity)**.** *There exists a constant $l > 0$ such that:*

$$\frac{l}{2}\|u - v\|^2 \leq \langle u - v, \nabla g_i(u) - \nabla g_i(v)\rangle \quad \forall u, v \in \mathcal{U}, \quad \forall i \in \{1, \dots, n\}$$

where $\langle\cdot,\cdot\rangle$ stands for the canonical scalar product in $\mathbb{R}^p$, and $\|\cdot\| = \sqrt{\langle\cdot,\cdot\rangle}$ is its associated norm.

From now on, we will assume that the g_i's are Lipschitz continuous, and strong convex. Hereafter, for every optimization method, we will state the known convergence rate, in terms of $J(u_j) - J(u_\star)$, where $u_\star$ is the exact minimizer of problem (II.75). However, under the strong convexity assumption, everything can be transposed to $u_j - u_\star$, since for any $u \in \mathcal{U}$,

$$\frac{l}{2}\|u - u_\star\|^2 \leq J(u) - J(u_\star). \tag{II.76}$$

For the FG method, an exponential convergence rate is guaranteed, as recalled in the previous Steepest Descent Section II.B.2, i.e.

$$J(u_j) - J(u_\star) \lesssim \rho^j, \tag{II.77}$$

with a constant $\rho < 1$ that depends on l and Lip defined in Assumptions 9 and 10 as long as $\tau_j = \tau \in \left]0, l/Lip^2\right[$. The last exponential convergence is also known as *geometric*, or *linear*. This method is very costly, when n is large, because it requires to compute at each iteration the n gradients ∇g_i, $i \in \{1, \dots, n\}$ over the whole training set.

II.D.7. The Stochastic Gradient method

Another method introduced by Robbins and Monro in 1951, called Stochastic Gradient (SG) overpasses this limitation when n becomes large, by computing at iteration j the gradient only for one particular index i_j, sampled uniformly from the set $\{1, \dots, n\}$ and independently of the previous index. We end up with the formulation:

$$u_{j+1} = u_j - \tau_j \nabla g_{i_j}(u_j), \tag{II.78}$$

One can show that selecting a suitable sequence of decreasing step-sizes $\tau_j = \frac{\tau_0}{j}$, with $\tau_0 > \frac{1}{l}$ we obtain the algebraic rate of convergence (this convergence rate is denoted by sub-linear, by the ML community, as it is worse than the convergence rate of the gradient method, also denoted as linear):

$$\mathbb{E}[J(u_j)] - J(u_\star) \lesssim \frac{1}{j}, \tag{II.79}$$

i.e. an algebraic rate $O\left(j^{-1/2}\right)$ on $\|u_j - u_\star\|$. Here, the expectation is taken w.r.t. the selection of the i_j index. On the other hand, the convergence rate is independent of n and does not suffer from any limitation when n increases.

Note that one usually chooses $\tau_j = \tau_0/j$, with τ_0 being such that $\tau_0 > 1/l$, where l is the strong convexity parameter of Assumption 10. If the above condition does not hold, we can not guarantee any **algebraic convergence rate**, as shown in the following counter example, taken from [SDR09]. Assume we want to minimize $f(x) = \frac{1}{2}\kappa x^2$ with $0 < \kappa < 1$ and $X := [-1, 1] \subset \mathbb{R}$ (we are in the simple case where $l = \kappa/2$). Clearly $x^* = 0$. The recursive sequence using the SG algorithm, and $\tau_0 = 1$ writes:

$$x_{j+1} = x_j - \frac{1}{j} f'(x_j) = \left(1 - \frac{\kappa}{j}\right) x_j$$

As we have $\kappa < 1$, the factor term remains positive, i.e. $1 - \frac{\kappa}{j} > 0$. But the condition $1 = \tau_0 > 1/2l = 1/\kappa$ is NOT respected. We can write the recurrence:

$$x_{j+1} = \prod_{s=1}^{j} \left(1 - \frac{\kappa}{s}\right) x_1 = \exp\left(-\sum_{s=1}^{j} \ln\left(1 + \frac{\kappa}{s-\kappa}\right)\right) x_1 > \exp\left(-\sum_{s=1}^{j} \frac{\kappa}{s-\kappa}\right) x_1.$$

Moreover, we have

$$\sum_{s=1}^{j} \frac{\kappa}{s-\kappa} \leq \frac{\kappa}{1-\kappa} + \int_1^j \frac{\kappa}{t-\kappa} \mathrm{d}t < \frac{\kappa}{1-\kappa} + \kappa \ln j - \kappa \ln(1-\kappa).$$

Then it follows that

$$x_{j+1} > O(1) j^{-\kappa} \quad \text{and} \quad f(x_{j+1}) > O(1) j^{-2\kappa}.$$

So, although x_j may still converge to zero, we miss the general convergence rate of $O(1/\sqrt{j})$ for κ small enough.

II.D.8. The Stochastic Averaged Gradient method

The Stochastic Averaged Gradient (SAG) method is a variant of SG [SRB13], which stores part of the old computed gradients (a fixed amount of them, say n, i.e. the memory is not increasing during the procedure). The algorithm is the following:

$$u_{j+1} = u_j - \frac{\tau_j}{n} \sum_{i=1}^{n} \nabla g_i(\phi_i^{j+1}), \tag{II.80}$$

where at each iteration j an index $i_j \in \{1, \dots, n\}$ is selected *at random*, and we set

$$\phi_i^{j+1} = \begin{cases} u_j & \text{if } i = i_j, \\ \phi_i^j & \text{otherwise.} \end{cases} \tag{II.81}$$

Again, the index $i_j \sim \mathcal{U}(\{1, \dots, n\})$ are iid uniform random variables. If we assume that each g_i is strongly-convex with a common constant l defined as in (10), the authors of [SRB13] have proven that the convergence is *exponential* in j (using a fixed, chosen, step-size), like in the FG method rather than sub-linear as in SG, i.e.

$$\mathbb{E}[J(u_j)] - J(u_\star) \lesssim \left(1 - \min\{\frac{l}{16L}, \frac{1}{8n}\}\right)^j. \tag{II.82}$$

Remark 4. *Although n appears in the convergence rate, in the well-conditioned setting, i.e. where $n > \frac{2L}{l}$, if we perform n iterations of SAG (i.e., one effective pass through the data), the error is multiplied by $(1 - 1/8n)^n \leq \exp(-1/8)$, which is independent of n. Thus, in this setting each pass through the data reduces the excess objective by a constant multiplicative factor that is independent of the problem.*

Another version of the SAG method, known as SAGA, introduced in [DBLJ14], is defined as

$$(SAGA) \quad u_{j+1} = u_j - \tau_j \left(\nabla g_{i_j}(u_j) - \nabla g_{i_j}(\phi_{i_j}^{j+1}) + \frac{1}{n} \sum_{i=1}^{n} \nabla g_i(\phi_i^{j+1}) \right), \tag{II.83}$$

whereas the SAG method can be rewritten equivalently as:

$$(SAG) \quad u_{j+1} = u_j - \tau_j \left(\frac{\nabla g_{i_j}(u_j) - \nabla g_{i_j}(\phi_{i_j}^{j+1})}{n} + \frac{1}{n} \sum_{i=1}^{n} \nabla g_i(\phi_i^{j+1}) \right). \tag{II.84}$$

As stated in [DBLJ14], the major advantage of SAGA versus SAG is that it uses an *unbiased* update formula for the descent direction, producing a simpler and tighter theory, with better constants than SAG (see Chapter IV for more details).

II.D.9. SG with momentum

Several improved versions of SG can be found in the literature. For instance, a momentum term can be added to the iterative SG scheme:

$$u_{j+1} = u_j - \tau_j \nabla g_{i_j}(u_j) + \beta_j \left(u_j - u_{j-1} \right).$$

Usually, the momentum parameter β_j remains constant over the iteration, i.e. $\beta_j = \beta$, and the algorithm can be rewritten as:

$$u_{j+1} = u_j - \sum_{k=1}^{j} \tau_k \beta^{j-k} \nabla g_{i_k}(u_k).$$

Actually, this last momentum recursive scheme uses a geometric weighting of the previous gradients, while SAG/SAGA select and average the most recent evaluation of each previous gradient. The momentum scheme still requires a decreasing sequence of step-sizes to converge, and does not have a better convergence rate than SG, although it might improve practical performance.

II.D.10. SG with gradient averaging

Another closely related optimization scheme is to use an average of *all* previously used gradients:

$$u_{j+1} = u_j - \frac{\tau_j}{j} \sum_{k=1}^{j} \nabla g_{i_j}(u_j), \tag{II.85}$$

which is similar to the SAG algorithm, but all previously computed gradients are averaged. It does not require any increase in the required memory, to store previously drawn gradient, contrary to SAG since one can use a simple update formula for the averaged gradient which only requires a memory space of two gradient evaluations. This SG with gradient averaging method does not require a decreasing sequence of step-sizes, but it does not improve the sub-linear convergence of SG, either.

II.D.11. Iterate Averaging

Another variation of SG-type algorithms, is obtained by selecting a step-size sequence that decreases slower than $1/j$:

$$u_{j+1} = u_j - \widetilde{\tau}_j \nabla g_{i_j}(u_j) \quad \text{with} \quad \widetilde{\tau}_j = \frac{\widetilde{\tau}_0}{j^\kappa}, \tag{II.86}$$

with $1/2 < \kappa \leq 1$ and where the index i_j is sampled uniformly from the set $\{1, \dots, n\}$. Then the convergence rate is measured on the averaged computed control, i.e. on

$$\widetilde{u}_j = \frac{1}{j} \sum_{k=1}^{j} u_j. \tag{II.87}$$

This algorithm does not have any restriction on $\widetilde{\tau}_0$ anymore, contrary to the Robbins-Monro version of SG, which required $\tau_0 > 1/l$, instead. The idea of averaging iterates goes back to [Pol90] and [Rup88], and is often referred to as Polyak–Ruppert averaging (see also [PJ92]). This approach is thus more robust to the problem, as it does not require estimating the convexity constant l, and still converges with the same $1/j$ algebraic rate on $J(u_j) - J(u_\star)$, as SG.

A more detailed analysis of SG algorithms, with and without the use of the Polyak-Ruppert averaged trick, can be found in [BM11]. The authors generalize convergence rates results, based on the convexity (but not necessarily strong) and Lipschitz continuity assumptions, using the general step-size $\widetilde{\tau}_j = \widetilde{\tau}_0 j^{-\kappa}$ with $1/2 < \kappa \leq 1$.

II.D.12. Hybrid Methods

In 1997, Bertsekas [Ber97] proposed a modified SG method, for problems of the form (II.75), which gradually turns into a FG method over the iterations so that to improve the convergence rate. Using a pass through all the n data points in the sum (II.75), with a *cyclic* and *deterministic* manner, he recovered an *exponential* convergence rate. The method can be decomposed as:

$$\begin{aligned} \phi_0 &= u_j \\ \phi_i &= \phi_{i-1} - \tau_j \nabla g_i(\phi_{i-1}), \quad i = 1, \dots, n \\ u_{j+1} &= u_j - \tau_j \sum_{i=1}^{n} \nabla g_i(\phi_{i-1}). \end{aligned}$$

The condition on the sequence τ_j is however numerically unstable, i.e. the convergence of this method depends a lot on the step-size sequence τ_j in practice, as stated in [SRB13]. Another slight modification of SG to recover an exponential convergence has been proposed in [FS12], where the authors replace the *single* gradient evaluation ∇g_{i_j} with an averaged version, on batches of increasing size. They achieve exponential convergence, under suitable assumptions, and the exponential convergence rate is not independent of n, contrary to the SAG/SAGA methods (see e.g. Remark 4).

II.D.13. Incremental Aggregated Gradient

The Incremental Aggregated Gradient (IAG), introduced in [BHG07], uses the same update formula as SAG, although the index i_j are set beforehand in a deterministic way. Specifically, we go through the whole set of data points increasing the index one by one. For a constant step-size $\tau > 0$, n arbitrary initial points u_1, u_2, ..., $u_n \in \mathcal{U}$, an initial aggregated gradient, denoted by d_n defined as $d_n = \sum_{i=1}^{n} \nabla g_i(u_i)$, the iterative scheme reads, for $j \geq n$:

$$\begin{aligned} u_{j+1} &= u_j - \tau \frac{1}{n} d_j, \\ d_{j+1} &= d_j - \nabla g_{(j+1)_n}(u_{j+1-n}) + \nabla g_{(j+1)_n}(u_{j+1}), \end{aligned}$$

where $(j)_n \in \{1, \dots, n\}$ represents j modulo n, with the representative class $\{1, \dots, n\}$. The authors in [BHG07] show exponential convergence, in the strongly convex setting, but they do not provide any explicit rate. On the other hand, SAG, by using a random selection of the index i_j, (versus a cyclic selection in IAG) improves optimization performance, and is more robust, when choosing the (fixed) step-size τ, allowing for larger τ.

The OCP studied in this thesis uses the mean based risk measure and can be recast into a ML framework

$$\min_{u\in\mathcal{U}} \mathbb{E}_\omega \left[l(\widehat{y}(u;\cdot),(\omega,z_d)) + \varsigma(u) \right] \tag{II.88}$$

as shown in Section II.D.5, with the exact expectation. The SG method does not require that the exact expectation is approximated beforehand as a discrete sum and can be equally applied by sampling the random variable ω independently at each iteration, according to the probability measure $\mathbb{P}$ on Γ. The resulting algorithm is

$$u_{j+1} = u_j - \tau_j \nabla \left(l(\widehat{y}(u_j;\cdot),(\omega_j,z_d)) + \varsigma(u_j) \right), \quad \omega_j \overset{\text{iid}}{\sim} \mathbb{P}.$$

This is the approach taken in Chapter III and IV.

On the other hand, in Chapter V, we do first approximate the true expectation by a finite sum, using a suitable Gaussian quadrature formula. This recast the problem into the form (II.88) proper of the ML problems.

Hence, all the methods described in this section could be applied. Among those, we have chosen the SAG/SAGA method, for its simplicity and exponential convergence property.

II.E. Goals of the Thesis

In this thesis, we discuss numerical methods to solve an OCP constrained by elliptic PDEs, involving random coefficients. The elliptic PDE writes

$$\begin{cases} -\operatorname{div}(a(x,\omega)\nabla y(x,\omega) = g(x) + u(x) & \text{for } x \in D \text{ a.e. } \omega \in \Gamma \\ y(x,\omega) = 0 \quad \text{for } x \in \partial D \text{ a.e. } \omega \in \Gamma. \end{cases} \tag{II.89}$$

and contains a random coefficient $a(\cdot,\omega)$, where ω is an element of a complete probability space $(\Gamma,\mathcal{F},P)$, yielding that the state solution $y(\cdot,\omega)$ is stochastic as well. Notice that in (II.89), the deterministic control u is distributed over the whole domain D. The functional we aim at minimizing is the quadratic functional

$$f(y,u) = \frac{1}{2}\|y - z_d\|^2_{L^2(D)} + \frac{\beta}{2}\|u\|^2_{L^2(D)} \tag{II.90}$$

where the function z_d is the target function we would like the state to be as close as possible. Using the risk-neutral measure $\mathbb{E}$, the quantity of interest to minimize becomes:

$$\min_{u\in\mathcal{U}} J(u) = \mathbb{E}[f(y_\omega(u),u)] \tag{II.91}$$

with f defined in (II.90), and $y_\omega(u)$ solution of (II.89).

In order to make the problem tractable numerically, we will use three types of approximations:

- a Finite Element approximation, to discretize the PDE (II.89), using a mesh of characteristic size h of the physical domain D;
- a collocation method to approximate the expectation in (II.91) using randomized realizations (such as in a Monte Carlo estimator), or deterministic knots (such as in a Gaussian quadrature formula);
- a line-search type algorithm to approximate the optimal control.

The ultimate goal of the thesis, is to propose new algorithms and analyze their complexity ($W = W(tol)$), i.e. derive bounds on the computational cost W, required to reach a given tolerance *tol* on the approximated control. Since we will use randomized line search algorithms like SG and/or randomized quadrature formulas (as MC), the error between the approximate control and the exact one will always be measured in a root *mean* squared sense.

II.F. Outline of the thesis

The next 3 Chapters contain the core of the thesis and correspond to scientific papers, either submitted for publication (Chapter III, V), or in preparation (Chapter IV).

In Chapter III, we present a solid mathematical framework on the PDE-constrained OCP, with uncertain coefficients (II.91), providing existence and uniqueness results when using the mean-based risk measure and detailing and analyzing discretized versions where the PDE (II.89) is approximated by finite elements and the expectation in the cost functional is approximated by a Monte Carlo method. We first consider a strategy in which the finite element discretization and the Monte Carlo sample are fixed initially, according to the desired tolerance to achieve, and a Full Gradient (FG) algorithm is used to compute the approximate control. In particular, the same MC sample is used in all gradient iterations. For this algorithm, we show that the asymptotic computational complexity $W(tol)$ to reach a root mean squared error (RMSE) of order *tol* is given by $W(tol) \lesssim tol^{-2-\frac{d\gamma}{r+1}} |\log(tol)|$, where γ is a parameter representing the efficiency of the used PDE linear solver, i.e. ($\gamma = 3$ for a direct solver and $\gamma = 1$ up to logarithm factors for an optimal multigrid solver), d is the dimension of the physical domain $D \subset \mathbb{R}^d$, and r is the polynomial degree of the used finite element space.

We compare then this fixed MC approach, with a stochastic gradient method, in which the expectation in the computation of the approximated gradient, is obtained by independent Monte Carlo estimators, with small sample sizes (even a size $n = 1$). We follow, in particular, the Robbins-Monro strategy [RM51, Rup88, NJLS09] of reducing progressively the step-size to achieve convergence of the Stochastic Gradient iterations. We then show that this SG approach reaches a computational complexity of $W(tol) \lesssim tol^{-2-\frac{d\gamma}{r+1}}$, i.e. we improved the complexity by a log factor w.r.t. the FG approach. We could not lower this complexity further by using a variable mesh-SG algorithm, where we refine the discretization of the PDE along the iterations.

In Chapter IV, we extend the previous Robbins-Monro strategy, by replacing the small size MC gradient estimator, by a multilevel Monte Carlo (MLMC) estimator of the gradient, which exploits a hierarchy of computational grids of decreasing mesh size. The multilevel paradigm has been introduced by Heinrich [Hei00] for parametric integration. It has been extended to weak approximations of stochastic differential equations (SDEs) in [Gil08] and has shown its efficiency, as a tool in numerical computations. Recently, the application of MLMC methods to uncertainty quantification problems involving PDEs with random data has been investigated from the mathematical point of view in a number of works [BSZ11, BLS13, CST13, CGST11, MSŠ12, TSGU13, NT15]. In the most favorable cases, it has been shown that the cost $W(tol)$ of computing the expected value of some output quantity of a stochastic differential model with accuracy tol, scales as $W(tol) \lesssim tol^{-2}$, and does not see the cost of solving the problem on fine discretizations. In this work, we use MLMC within a SG algorithm. In particular, we present a full convergence and complexity analysis of the resulting MLSG algorithm in the case of the quadratic, strongly convex, OCP (II.91). By reducing progressively the bias and the variance of the MLMC estimator over the iterations at a proper rate, we are able to recover an optimal complexity $W(tol) \lesssim tol^{-2}$ in the computation of the optimal control, analogous to the one for the computation of a single expectation. This result considerably improves the ones obtained in Chapter III.

We also propose a randomized version of the MLSG algorithm, which uses the unbiased multilevel Monte Carlo algorithm proposed in [RG12, RG15] (see also [Gil15, Section 2.2]). In this randomized version, we replace the full MLMC sampler at each iteration j of the Stochastic Gradient algorithm by only one evaluation of the difference of the objective function on levels l_j and $l_j - 1$ where the level l_j is drawn randomly (and independently at each iteration) from a suitable probability mass function over all levels. We show that this version of the MLSG algorithm also achieves optimal complexity $W(tol) \lesssim tol^{-2}$. The main advantage of this randomized version, w.r.t. the one that uses a full MLMC estimator at each iteration, is that it requires fewer parameters to tune, which is preferable, from a numerical point of view.

In Chapter V, we address again the OCP (II.91), yet this time we consider a deterministic quadrature formula to approximate the expectation in the cost functional, in combination with the SAGA algorithm (II.83). Assuming that the randomness in the PDE (II.89) can be parameterized in terms of a small number M of independent random variables, the expectation appearing in the cost functional $J(u)$ can be written as a M-dimensional integral and suitably approximated by a quadrature formula as e.g. a tensorized Gaussian quadrature, leading to an approximate optimal control problem

$$\widehat{u}^* \in \arg\min_{u\in\mathcal{U}} \widehat{J}(u), \quad \widehat{J}(u) = \sum_{i=1}^{n} \zeta_i f(y_{\eta_i}(u), u) \tag{II.92}$$

where η_i are the quadrature knots and ζ_i the quadrature weights with $\sum_{i=1}^{n} \zeta_i = 1$. For a given control u, evaluating $\widehat{J}(u)$ entails the computation of the n solutions $\{y_{\eta_i}(u)\}_{i=1}^{n}$ of the underlying PDE. This approach is known in the literature as *stochastic collocation* method and has been analyzed e.g. in [BNT10]. It leads, in favorable cases, to an error in the functional that converges to zero (sub)-exponentially in n, although typically exposed to the curse of dimensionality, hence acceptable only for a small number of random variables. By replacing the tensorized quadrature by a suitable sparse one (see e.g. [BG04, NTT16]), dimension free convergence rates have been demonstrated in certain cases (see e.g. [SS13, HANTT16b, EST18, ZDS18] and references therein). However, in this chapter, we stick to the simpler setting of a tensorized Gaussian quadrature formula and a small number of random variables.

We then apply on the functional (II.92), the SAGA method, which computes at each iteration, the gradient of the approximated $\widehat{J}$, in only one quadrature point, randomly chosen from a possibly non-uniform distribution. The SAGA algorithm as stated in (II.83) is applicable to the case of uniform weights $\zeta_i = \frac{1}{n}$, $i = 1, \dots, n$ and uniformly drawn index i_j over $\{1, \dots, n\}$. A variant of the SAGA method that uses a non-uniform sampling of the indexes i_j has been proposed in [SBA$^+$]. In this Chapter we first extend the SAGA algorithm to the case of non-uniform weights, as they appear naturally in a Gaussian quadrature formula (II.92). In particular, we propose an importance sampling strategy, where the indexes i_j are drawn from a possibly non-uniform distribution, also different from the distribution induced by the weights $\{\zeta_i\}_{i=1}^{n}$. Following similar steps as in [DBLJ14, SBA$^+$], we present a full theoretical convergence analysis of the generalized SAGA method, for the OCP (II.92), and prove theoretically a complexity bound of $W(tol) \lesssim tol^{-\frac{d\gamma}{r+1}} |\log(tol)|^M$, where M is the stochastic dimension. Compared with the complexity of SG+Monte Carlo obtained in Chapter III, we see that the factor tol^{-2} proper of Monte Carlo is replaced by $|\log(tol)|^M$ which is the complexity of the Gaussian quadrature formula. This improvement is possible thanks to the fact that we use a SAGA method, whose convergence is exponential as recalled in Section II.D.8, instead of SG, whose convergence is sub-linear and would therefore retain the factor tol^{-2} even when using an accurate Gaussian quadrature. This complexity is reached at the price of requiring a larger memory size w.r.t. a standard SG algorithm, being multiplied by a factor $|\log(tol)|^{M+1}$. Finally, as shown in the numerical examples, the strength of the SAGA approach to solve an OCP, is in the pre-asymptotic regime, as acceptable solutions may be obtained already before a full sweep over all quadrature points (which is the cost of a single iteration, when we use the Full Gradient approach).

III Analysis of stochastic gradient methods for PDE-constrained OCPs with uncertain parameters

This Chapter is essentially the same as [MKN18] submitted for publication.

III.A. Introduction

Many problems in engineering and science, e.g. shape optimization in aerodynamics or heat transfer in thermal conduction problems, deal with optimization problems constrained by partial differential equations (PDEs) [HPUU09, BS12, De 15, Haz10, LBE$^+$14]. Often, these types of problems are affected by uncertainties, due to a lack of knowledge, intrinsic variability in the system, or an imprecise manufacturing process. For instance, to determine the optimal cooling of a super-computing center, one should take into account the fact that the heat source from the supercomputers could vary considerably over time and also the heat conduction properties of the machines might not be perfectly determined. As these material properties or boundary conditions are not precisely known, it is reasonable to consider optimal control problems (OCPs) constrained by PDEs with uncertain coefficients, which could be described as random variables or random fields. This OCP is sometimes also referred to as Optimization Under Uncertainty (OUU).

In this work we focus on the numerical approximation of the problem of controlling the solution of an elliptic PDE with random coefficients by a distributed unconstrained control. Specifically, the control acts as a volumetric forcing term, so that the solution is as close as possible to a given target function.

While there is a vast literature on the numerical approximation of PDE-constrained optimal control problems (see e.g. [BS12, HPUU09] and references therein) in the deterministic case, as well as on the numerical approximation of (uncontrolled) PDEs with random coefficients (see e.g. [GWZ14, BNT10, LPS14] and references therein), the analysis of corresponding PDE constrained control problem under uncertainty is much more recent and incomplete, although the topic has received increasing attention in the last few years.

The formulations of the PDE-constrained OCPs under uncertainty that can be found in the literature can be roughly grouped in two categories.

In the first category, the control is *random* [CQ14, AAUH17, TKXP12, RW12, KS13, BOS16]. This situation arises when the randomness in the PDE is observable hence an optimal control can be built for each realization of the random system. However, the corresponding optimality system might still be fully coupled in the random parameters if the objective function involves some statistics of the state variables. The dependence on the random parameters is typically approximated either by polynomial chaos expansions or Monte Carlo (MC) techniques.

The former approach is considered e.g. in [KS13], where the authors prove analytic dependence of the control on the random parameters and study its best N-term polynomial chaos approximation for a linear parabolic PDE-constrained OCP; the work [CQ14], combines a stochastic collocation with a Finite Element (FE) based reduced basis method to alleviate the computational effort; the works [RW12, TKXP12, BOS16] address the case of a fully coupled optimality system discretized by either Galerkin or collocation approaches and propose different methods, such as sequential quadratic programming, or block diagonal preconditioning to solve the coupled system efficiently. Monte Carlo and Multilevel Monte Carlo approaches are considered in [AAUH17] instead, where the case of random coefficients with limited spatial regularity is addressed.

In the second category, the control is *deterministic* [APSG17, VBV18, BvW11, Kou12, KS16, KHRvBW13, GLL11]. This situation arises when randomness in the system is not observable at the time of designing the control, so that the latter should be *robust* in the sense that it minimizes the *risk* of obtaining a solution which leads to high values of the objective function. This situation is also referred to as *risk-averse optimal control* and always leads to a fully coupled optimality system in the random parameters. The idea of minimizing a risk to obtain a solution with favorable properties goes back to the origins of robust optimization [SDR09]. Here, *risk* refers to a suitable statistical measure of the objective function to be minimized, such as its expectation, expectation plus variance, a quantile, or a conditional expectation above a quantile (so called Conditional Value at Risk (CVaR) [RU02]).

Numerical methods for OCPs of this category typically depend on the choice of the risk measure. For example, the work [APSG17] considers a risk measure that involves the mean and variance of the objective function and uses second order Taylor expansions combined with randomized estimators to reduce the computational effort. The work [VBV18] considers a risk measure that involves only the mean of the objective function (hereafter named mean-based risk), with an additional penalty on the variance of the state, and proposes a gradient type method, in which the expectation of the gradient is computed by a Multilevel Monte Carlo method. In [BvW11], the authors also consider a mean-based risk problem and propose a reduced basis method on the space of controls to dramatically reduce the computational effort. In the work [Kou12], the author presents a more general type of OCP, using the general notion of a risk measure, and derives the corresponding optimality system of PDEs to be solved. For its numerical solution, a trust-region Newton conjugate gradient algorithm is proposed in [KHRvBW13], combined with an adaptive sparse grid collocation for the discretization of the PDE in the stochastic space. The work [KS16] considers derivative-based optimization methods for the robust CVaR risk measure, which are building upon introducing smooth approximations to the CVaR. Finally, in the work [GLL11], the authors consider a boundary OCP where the deterministic control appears as a Neumann boundary condition.

In this work, we follow the second modeling category and consider the (robust) OCP of minimizing the mean-based risk of the objective function. We consider in particular gradient type methods where adjoint calculus is used to represent the gradient of the objective function, and FE approximations of the primal and dual problems, as well as a Monte Carlo approximation of the expectation in the risk measure are employed. The reason for looking at Monte Carlo approximations, instead of polynomial chaos ones, is to develop methods that can potentially handle many random parameters and possibly rough random coefficients.

Our main contribution is to provide a full error analysis including the finite element, the Monte Carlo and the gradient iterations errors, as well as a complexity analysis when all sources of errors are optimally balanced to achieve a given tolerance. The motivation for analyzing gradient type optimization methods is twofold. First, their rather simple structure allows for a complete complexity analysis, which is desirable in practice due to their wide-spread use. Second, our analysis reveals that the cost due to the FE and the Monte Carlo approximations dominate the overall computational complexity, in the sense that the gradient type method only increases the cost by a logarithmic term.

It is noteworthy that other error analysis have been proposed in [CQ14] in the case of a random control, with a discretization in space by Finite Elements and in probability by stochastic collocation, and in [GLL11] in the case of a mean-based risk for a deterministic boundary control problem, using a Finite Element discretization both in space and in probability.

The first gradient method that we consider is the standard gradient method (which we call fixed MC gradient), in which the Finite Element discretization and the Monte Carlo samples are chosen initially and kept fixed over the iterations of the gradient method. If N is the sample size of the Monte Carlo estimator, this method entails the solution of N primal and N dual problems at each iteration of the gradient method, which could be troublesome if a small tolerance is required, entailing a very large N and small Finite Element mesh size.

We then turn to stochastic versions of the gradient method in which the gradient is re-sampled independently at each iteration and the Finite Element mesh size can be refined along the iterations. This corresponds to taking, at each iteration, an independent Monte Carlo estimator with only one realization ($N = 1$) or a very small and fixed sample size ($N = \bar{N}$) independently of the required tolerance, with possibly a finer Finite Element mesh. We follow, in particular, the Robbins-Monroe strategy [RM51, Rup88, NJLS09] of reducing progressively the step-size to achieve convergence of the Stochastic Gradient iterations.

Stochastic Gradient (SG) techniques have been extensively applied to machine learning problems [KY97, FB15, DB15, DB16], but have not yet been used for risk-averse PDE-constrained optimization problems. Here, we show that our Stochastic Gradient method improves the complexity of the fixed MC gradient method by a logarithmic factor. Although the computational gain is not dramatic, we see potential in this approach as only one primal problem and one dual problem have to be solved at every iteration of the gradient method. Moreover, we believe that the whole construction is more amenable to an adaptive version, which, in combination with an appropriate error estimator, allows for a self-controlling algorithm. We leave this for future work.

The rest of the paper is organized as follows: in Section III.B we set the mean-based risk-averse optimal control problem and recall its well posedness and the optimality conditions; in Sections III.C, III.D, III.E we introduce, respectively, the finite element discretization, the Monte Carlo approximation, and the steepest descent (gradient) method, including their full error analysis. In particular, Theorem 12 in Section III.E gives an error bound for the fully discrete solution of the fixed MC gradient method, whereas Corollary 2 gives the corresponding computational complexity. In Section III.F we analyze the Stochastic Gradient method with fixed finite element discretization over the iterations (with error bound given in Theorem 13 and the corresponding complexity result in Corollary 3), whereas in Section III.G we analyze the Stochastic Gradient version in which the Finite Element mesh is refined over the iterations (Theorem 15 and Corollary 4). In Section III.H, we discuss a 2D test problem and confirm numerically the theoretical error bounds and complexities derived in the preceding Sections. Finally, in Section III.I we draw some conclusions.

III.B. Problem setting

We start introducing the primal problem that will be part of the OCP discussed in the following. Specifically, we consider the problem of finding the solution $y : D \times \Gamma \to \mathbb{R}$ of the elliptic random PDE

$$\begin{cases} -\operatorname{div}(a(x,\omega)\nabla y(x,\omega)) = \phi(x,\omega), & x \in D, \quad \omega \in \Gamma, \\ y(x,\omega) = 0, & x \in \partial D, \quad \omega \in \Gamma, \end{cases} \tag{III.1}$$

where $D \subset \mathbb{R}^d$ is open and bounded, denoting the physical domain, $(\Gamma, \mathcal{F}, P)$ is a complete probability space, and $\omega \in \Gamma$ is an elementary random event. The diffusion coefficient a is an almost surely (a.s.) continuous and positive random field on D, and ϕ is a stochastic source term (that could contain, for example, a deterministic control part).

Before addressing the optimal control problem related to the random PDE (III.1), we will first recall the well posedness results for (III.1). We begin by recalling some usual function spaces needed for the analysis that follows. Let $L^p(D)$ for $1 \leq p < \infty$ denote the space of functions for which the p-th power of their absolute value is Lebesgue integrable, that is

$$L^p(D) = \{y : D \to \mathbb{R}, \;\; f \text{ measurable, and } \int_D |y|^p \mathrm{d}x < +\infty\},$$

and $L^\infty(D)$ the space of measurable functions that are bounded almost everywhere (a.e.) on D. Throughout this work, we will denote by $\|\cdot\| \equiv \|\cdot\|_{L^2(D)}$ the usual $L^2(D)$-norm induced by the inner product $\langle f, g\rangle = \int_D fg\mathrm{d}x$ for any $f, g \in L^2(D)$. Furthermore, we introduce the Sobolev spaces

$$H^1(D) = \{y \in L^2(D), \quad \partial_{x_i} y \in L^2(D), \quad i = 1, \dots, n\}$$

and

$$H_0^1(D) = \{y \in H^1(D), \quad y|_{\partial D} = 0\}.$$

We use the equivalent H^1-norm on the space $H_0^1(D)$ defined by $\|y\|_{H^1(D)} = \|y\|_{H_0^1(D)} = \|\nabla y\|$ for any $y \in H_0^1(D)$. Moreover, we recall the Poincaré inequality for any function $y \in H_0^1(D)$

$$\|y\| \leq C_p \|\nabla y\| = C_p \|y\|_{H^1(D)},$$

where C_p is the Poincaré constant, and that $H^{-1}(D) = \left(H_0^1(D)\right)^*$ is the topological dual of $H_0^1(D)$. For $r \in \mathbb{N}$ we further recall the space $H^r(D)$ of $L^2(D)$ functions with all partial derivatives up to order r in $L^2(D)$ with norm $\|y\|_{H^r(D)}$ and semi-norm $|y|_{H^r(D)}$ given by

$$\|y\|_{H^r(D)}^2 = \sum_{|\boldsymbol{\alpha}| \leq r} \left\| \frac{\partial^{|\boldsymbol{\alpha}|} y}{\partial x^{\boldsymbol{\alpha}}} \right\|_{L^2(D)}^2 \quad \text{and} \quad |y|_{H^r(D)}^2 = \sum_{|\boldsymbol{\alpha}| = r} \left\| \frac{\partial^{|\boldsymbol{\alpha}|} y}{\partial x^{\boldsymbol{\alpha}}} \right\|_{L^2(D)}^2,$$

respectively, for the multi-index $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_n)$. Finally, we introduce the Bochner spaces $L^p(\Gamma, \mathcal{V})$, which are formal extensions of Lebesgue spaces $L^p(\Gamma)$, for functions with values in a separable Hilbert space $\mathcal{V}$ as

$$L^p(\Gamma, \mathcal{V}) = \{y : \Gamma \to \mathcal{V}, \ \ y \text{ measurable}, \int_\Gamma \|y(\omega)\|^p_{\mathcal{V}} \mathrm{d}P(\omega) < +\infty\},$$

equipped with the norm $\|y\|_{L^p(\Gamma,\mathcal{V})} = \left(\int_\Gamma \|y(\omega)\|^p_{\mathcal{V}} \mathrm{d}P(\omega)\right)^{\frac{1}{p}}$, see, e.g., [Eva98] for details.

As it is common for the well posedness of the elliptic PDE (III.1), we assume that the diffusion coefficient a in (III.1) is uniformly elliptic.

Assumption 11. *The diffusion coefficient $a \in L^\infty(D \times \Gamma)$ is bounded and bounded away from zero a.e. in $D \times \Gamma$, i.e.*

$$\exists \quad a_{\min}, a_{\max} \in \mathbb{R} \quad \text{such that} \quad 0 < a_{\min} \leq a(x, \omega) \leq a_{\max} \quad \text{a.e. in } D \times \Gamma.$$

Now we are in the position to recall the well posedness of the random PDE (III.1), which is a standard result, see e.g. [LPS14, BTZ04].

Lemma 3 (Well posedness of (III.1))**.** *Let Assumption 11 hold. If $\phi \in L^2(\Gamma, H^{-1}(D))$, then problem* (III.1) *admits a unique solution $y \in L^2(\Gamma, H^1_0(D))$ s.t.*

$$\|y(\cdot, \omega)\|_{H^1_0(D)} \leq \frac{1}{a_{min}} \|\phi(\cdot, \omega)\|_{H^{-1}(D)} \quad \text{for a.e. } \omega \in \Gamma$$

$$\text{and } \ \|y\|_{L^2(\Gamma, H^1_0(D))} \leq \frac{1}{a_{min}} \|\phi\|_{L^2(\Gamma, H^{-1}(D))}.$$

Finally, as we will occasionally need H^2-regularity in the following Sections, we also introduce a sufficient condition on the domain D and on the gradient of a.

Assumption 12. *The domain $D \subset \mathbb{R}^d$ is polygonal convex and the random field $a \in L^\infty(D \times \Gamma)$ is such that $\nabla a \in L^\infty(D \times \Gamma)$,*

Then, using standard regularity arguments for elliptic PDEs, one can prove the following result [Eva98].

Lemma 4. *Let Assumptions 11 and 12 hold. If $\phi \in L^2(\Gamma, L^2(D))$, then problem* (III.1) *has a unique solution $y \in L^2(\Gamma, H^2(D))$. Moreover there exists a constant C, independent of ϕ, such that*

$$\|y\|_{L^2(\Gamma, H^2(D))} \leq C \|\phi\|_{L^2(\Gamma, L^2(D))}.$$

We are now ready to introduce the optimal control problem linked with the PDE (III.1), which we will study in the rest of the paper.

III.B.1. Optimal Control Problem

We define the primal problem for the OCP as the elliptic PDE (III.1), by particularizing its right hand side to:

$$\begin{cases} -\operatorname{div}(a(x,\omega)\nabla y(x,\omega)) &= g(x) + u(x), \qquad x \in D, \quad \omega \in \Gamma, \\ y(x,\omega) &= 0, \qquad x \in \partial D, \quad \omega \in \Gamma, \end{cases} \tag{III.2}$$

with $g \in H^{-1}(D)$ and $u \in U$, where $U \subset L^2(D)$ denotes the set of all admissible (deterministic) control functions. We set the state space of the solution to (III.2) as $Y = H_0^1(D)$. To emphasize the dependence of the solution to the PDE on the control function and on a particular realization $a(\cdot,\omega)$ of the random field, we will use the notation $y_\omega(u)$. When the particular realization of a is not relevant, or when no confusion arises, we will also simply write $y(u)$ from times. In this work, we focus on the objective function

$$J(u) = \mathbb{E}[f(u,\omega)] \quad \text{with} \quad f(u,\omega) = \frac{1}{2}\|y_\omega(u) - z_d\|^2 + \frac{\alpha}{2}\|u\|^2, \tag{III.3}$$

where z_d is a given target function which we would like the state $y_\omega(u)$ to approach as close as possible, in a mean-square-error sense. The coefficient $\alpha \geq 0$ is a constant of the problem that models the price of energy, i.e. how expensive it is to add some energy in the control u in order to decrease the first distance term $\mathbb{E}\left[\|y_\omega(u) - z_d\|^2\right]$. The ultimate goal then is the OCP, of determining the optimal control $u_\star$, so that

$$u_\star \in \underset{u\in U}{\arg\min}\, J(u), \quad \text{s.t.} \quad y_\omega(u) \in Y \quad \text{solves} \quad \text{(III.2)} \quad \text{a.s.} \tag{III.4}$$

Remark 5. *The optimal control $u_\star$ in* (III.4) *is the one that provides the best fit $\|y_\omega(u_\star) - z_d\|$* on average *not requiring too much* control energy *(induced by the regularization term). In view of applications, one may consider a more general objective function $J(u) = \sigma\Big(\frac{1}{2}\|y_\omega(u) - z_d\|^2\Big) + \frac{\alpha}{2}\|u\|^2$, where $\sigma(\cdot)$ is a* more robust risk measure *such as the Conditional Value at Risk [KS16]. In this paper, however, we restrict to the simple* expectation *risk measure, namely $\sigma(\cdot) = \mathbb{E}[\cdot]$, for sake of simplicity.*

As we aim at minimizing the functional J, we will use the theory of optimization and calculus of variations. Specifically, we introduce the optimality condition for the OCP (III.4), in the sense that the optimal control $u_\star$ satisfies

$$\langle \nabla J(u_\star), v - u_\star\rangle \geq 0 \quad \forall v \in Y. \tag{III.5}$$

Here, by $\nabla J(u)$ we denote the $L^2(D)$-functional representation of the Gateaux derivative of J, namely

$$\int_D \nabla J(u)\delta u \,\mathrm{d}x = \lim_{\epsilon\to 0} \frac{J(u+\epsilon\delta u) - J(u)}{\epsilon} = DJ(u)(\delta u) \quad \forall\, \delta u \in L^2(D).$$

In order to study the well posedness of problem (III.4), we introduce a further assumption on α, U and g.

Assumption 13. *The regularization parameter α is strictly positive, i.e. $\alpha > 0$. Moreover, the space of admissible control functions is $U = L^2(D)$ and the deterministic source term is such that $g \in L^2(D)$.*

It follows from the results in [Kou12] that problem (III.4) is well posed. As a matter of fact, problem (III.4) is even well posed for more general settings than the one considered here. For completeness, we give a short proof for the particular setting considered in this work, as many of the following results will build on it. For this we first introduce the following solution operator corresponding to the elliptic PDE (III.1):

$$\begin{aligned} S : L^2(\Gamma, H^{-1}(D)) &\longrightarrow L^2(\Gamma, Y) \\ \phi &\longmapsto S\phi = y \ \text{ solution of } \ \text{(III.1)}. \end{aligned}$$

Notice that the operator S is continuous in view of Lemma 3. In the case of $\phi = g + u \in L^2(D)$ deterministic, we will sometimes use the notation $S_\omega(g+u) = y_\omega(u)$ to denote one ω-realization of y. As S is self-adjoint, we have $S^* = S$. Moreover, for any separable Hilbert space $\mathcal{V}$, we denote by $\mathbb{E}$ the usual expectation operator with respect to (w.r.t.) the probability measure P acting on the space $L^2(\Gamma, \mathcal{V})$, i.e. $\mathbb{E} : L^2(\Gamma, \mathcal{V}) \to \mathcal{V}$. Its adjoint operator is

$$\begin{aligned} \mathbb{E}^* : \mathcal{V}' &\longrightarrow L^2(\Gamma, \mathcal{V}') \\ v &\longmapsto v, \end{aligned}$$

which associates the *constant* stochastic (i.e. deterministic) function $v \in L^2(\Gamma, \mathcal{V}')$ to each deterministic function $v \in \mathcal{V}'$. Finally we define the two operators

$$\widetilde{S} = S\mathbb{E}^* : L^2(D) \to L^2(\Gamma, Y) \quad \text{and} \quad \widetilde{S}^* = \mathbb{E}S^* : L^2(\Gamma, Y) \to L^2(D).$$

Existence and uniqueness of the OCP (III.4) can then be stated as follows.

Theorem 8. *Suppose Assumptions 11 and 13 hold. Then the OCP* (III.4) *admits a unique control $u_\star \in U$. Moreover*

$$\nabla J(u) = \alpha u + \mathbb{E}[p_\omega(u)], \tag{III.6}$$

where $p_\omega(u) = p$ is the solution of the adjoint problem (a.s. in Γ)

$$\begin{cases} -\operatorname{div}(a(\cdot,\omega)\nabla p(\cdot,\omega)) &= \; y(\cdot,\omega) - z_d \qquad \text{in } D, \\ p(\cdot,\omega) &= \; 0 \qquad \text{on } \partial D. \end{cases} \tag{III.7}$$

Proof. Let us define the inner product $\ll \cdot,\cdot \gg$ on the Bochner space $L^2(\Gamma, U)$, $\ll u, v \gg = \mathbb{E}\left[< u, v >\right] = \int_\Gamma \int_D u(x,\omega)v(x,\omega)\mathrm{d}x\, \mathrm{d}P(\omega)$. Using the linearity of the introduced operators, we can write $J(u)$ as

$$\begin{aligned} J(u) &= \frac{1}{2}\mathbb{E}\left[< \widetilde{S}(g+u) - z_d, \widetilde{S}(g+u) - z_d >\right] + \frac{\alpha}{2} < u, u > \\ &= \frac{1}{2} \ll \widetilde{S}(g+u) - z_d, \widetilde{S}(g+u) - z_d \gg + \frac{\alpha}{2} < u, u > \\ &= \frac{1}{2} \ll \widetilde{S}u, \widetilde{S}u \gg + \ll \widetilde{S}g - z_d, \widetilde{S}u \gg + \frac{1}{2} \ll \widetilde{S}g - z_d, \widetilde{S}g - z_d \gg + \frac{\alpha}{2} < u, u > . \end{aligned}$$

Defining the bi-linear form $A : U \times U \to \mathbb{R}$, $A(u,v) = \ll \widetilde{S}u, \widetilde{S}v \gg + \alpha < u, v >$, the linear form $G : U \to \mathbb{R}$, $G(v) = \ll \widetilde{S}g - z_d, \widetilde{S}v \gg$, and the constant $k = \frac{1}{2} \ll \widetilde{S}g - z_d, \widetilde{S}g - z_d \gg \in \mathbb{R}$, we find

$$J(u) = \frac{1}{2}A(u,u) + G(u) + k.$$

Thanks to Assumptions 11 and 13, it is easy to see that A is coercive and continuous (cf. Lemma 3), that G is continuous, and that $k < +\infty$. Then, applying Thm. 7.1 of [Lio71], we conclude that there exists a unique solution $u_\star \in U$ to problem (III.4). Next, we compute the Gâteaux derivative of J at the point u in the direction δu:

$$\begin{aligned} DJ(u)(\delta u) &= \int_D \nabla J(u)\delta u \mathrm{d}x = A(u, \delta u) + G(\delta u) \\ &= \ll \widetilde{S}u, \widetilde{S}\delta u \gg + \alpha < u, \delta u > + \ll \widetilde{S}g - z_d, \widetilde{S}\delta u \gg \\ &= < \widetilde{S}^*(\widetilde{S}(g+u) - z_d), \delta u > + < \alpha u, \delta u > \\ &= < \mathbb{E}S^*(S(g+u) - z_d), \delta u > + < \alpha u, \delta u > \\ &= < \alpha u + \mathbb{E}\left[S^*(S(g+u) - z_d)\right], \delta u > . \end{aligned}$$

Defining $p_\omega(u)$ as $p_\omega(u) = S^*\left(S(g+u) - z_d\right) = S^*\left(y_\omega(u) - z_d\right)$, which is the solution of equation (III.7), we get $\nabla J(u) = \alpha u + \mathbb{E}\left[p_\omega(u)\right]$. □

Remark 6. *By computing the gradient of f w.r.t. u, we can easily get $\nabla f(u,\omega) = \alpha u + p_\omega(u)$. Consequently, the previous proof, also reveals that*

$$\nabla J(u) = \nabla \mathbb{E}[f(u,\omega)] = \mathbb{E}\left[\nabla f(u,\omega)\right].$$

In Theorem 8, $p_\omega(u) = p$ is the so-called adjoint function associated to the elliptic PDE (III.2) and satisfies the adjoint equation which depends on the solution $y = y_\omega(u)$. As p depends on u through y, we will also write $p(y_\omega(u))$ for $p_\omega(u)$ from times.

For notational convenience, we introduce the weak formulation of (III.2), which reads

$$\text{find } y_\omega \in Y \; s.t. \; b_\omega(y_\omega, v) = \langle g + u, v\rangle \quad \forall v \in Y \qquad \text{for a.e. } \omega \in \Gamma, \tag{III.8}$$

where $b_\omega(y,v) := \int_D a(\cdot,\omega)\nabla y \nabla v dx$. Similarly, the weak form of problem (III.7) reads:

$$b_\omega(v, p_\omega) = \langle v, y_\omega - z_d\rangle \quad \forall v \in Y \qquad \text{for a.e. } \omega \in \Gamma. \tag{III.9}$$

We can thus rewrite the OCP (III.4) equivalently as:

$$\begin{cases} \min_{u\in U} J(u) = \frac{1}{2}\mathbb{E}[\|y_\omega(u) - z_d\|^2] + \frac{\alpha}{2}\|u\|^2 \\ \text{s.t.} \quad y_\omega \in Y \quad \text{solving} \\ b_\omega(y_\omega, v) = \langle g + u, v\rangle \quad \forall v \in Y \qquad \text{for a.e. } \omega \in \Gamma. \end{cases} \tag{III.10}$$

As we want to compute numerically the problem solution, we introduce in the following Section a Finite Element approximation and different versions of error estimates.

III.C. Finite Element approximation in physical space

In this section we analyze the semi-discrete OCP obtained by approximating the underlying PDE by a Finite Element method. In particular, we provide a priori error bounds for the optimal control. Let us denote by $\{\tau_h\}_{h>0}$ a family of regular triangulations of D. Furthermore, let Y^h be the space of continuous piece-wise polynomial functions of degree r over τ_h that vanish on ∂D, i.e. $Y^h = \{y \in C^0(\overline{D}) : y|_K \in \mathbb{P}_r(K) \quad \forall K \in \tau_h, y|_{\partial D} = 0\} \subset Y = H^1_0(D)$. Finally, we set $U^h = Y^h$. We can then reformulate the OCP (III.10) as a finite dimensional OCP in the FE space:

$$\begin{cases} \min_{u^h\in U^h} J^h(u^h) := \frac{1}{2}\mathbb{E}[\|y^h_\omega(u^h) - z_d\|^2] + \frac{\alpha}{2}\|u^h\|^2 \\ \text{s.t. } y^h_\omega \in Y^h \text{ and} \\ b_\omega(y^h_\omega(u^h), v^h) = \langle u^h + g, v^h\rangle \quad \forall v^h \in Y^h \quad \text{for a.e. } \omega \in \Gamma. \end{cases} \tag{III.11}$$

Analogously to the (continuous) solution operator S of (III.1) introduced in Section III.B.1, here we introduce its discrete version associated to problem (III.11). That is, let $S^h_\omega : U \to Y^h$ be such that $y^h_\omega = S^h_\omega(g + u^h)$ solves $b_\omega(y^h_\omega, v^h) = \langle g + u^h, v^h\rangle \quad \forall v^h \in Y^h$. We also introduce the L^2-projection operator onto U^h, denoted by $g^h = \Pi_{U^h}(g)$, as

$$\forall q \in U, \;\; \langle \Pi_{U^h} q, v^h\rangle = \langle q, v^h\rangle \;\; \forall v^h \in U^h.$$

As mentioned before, we may suppress the index ω of S_ω when no ambiguity arises, we do so also for $S^h_\omega = S^h$. Moreover, we denote by $\left(S^h\right)^*$ the corresponding adjoint operator of S^h. From now on, and throughout the rest of this paper, we assume that Assumptions 11, 12 and 13 are verified. Then we can state the following FE approximation result.

Lemma 5. *The discrete OCP* (III.11) *is well posed and* ∇J^h *can be characterized as*

$$\nabla J^h(u_\star^h) = \Pi_{U^h}(\alpha u_\star^h + \mathbb{E}[p^h(u_\star^h)]) \tag{III.12}$$

and

$$p^h(u_\star^h) := \left(S^h\right)^* (S^h(u_\star^h + g) - z_d) \in L^2(\Gamma, Y^h).$$

Remark 7. *Notice that since we defined* $U^h = Y^h$*, it follows that* $\mathbb{E}[p^h(u_\star^h)] \in U^h$ *and* $\nabla J^h(u_\star^h) = \alpha u_\star^h + \mathbb{E}[p^h(u_\star^h)]$.

Following similar arguments as in Thm. 3.4 of [HPUU09] and using the optimality condition, and the weak form of the primal and dual problems, we can prove the following.

Theorem 9. *Let* $u_\star$ *be the optimal control solution of problem* (III.10) *and denote by* $u_\star^h$ *the solution of the approximated problem* (III.11)*. Then it holds that*

$$\frac{\alpha}{2}\|u_\star - u_\star^h\|^2 + \frac{1}{2}\mathbb{E}[\|y(u_\star) - y^h(u_\star^h)\|^2] \leq \frac{1}{2\alpha}\mathbb{E}[\|p(u_\star) - \widetilde{p}^h(u_\star)\|^2] + \frac{1}{2}\mathbb{E}[\|y(u_\star) - y^h(u_\star)\|^2], \tag{III.13}$$

where, $\widetilde{p}^h(u_\star) = \widetilde{p}^h_\omega(u_\star)$ *is such that*

$$b_\omega(v^h, \widetilde{p}^h_\omega) = \langle v^h, y_\omega - z_d\rangle \quad \forall v^h \in Y^h \text{ for a.e. } \omega \in \Gamma. \tag{III.14}$$

Proof. It follows from Theorem 8 and Lemma 5 that the FE version of the optimality condition (III.5) reads:

$$\langle \nabla J^h(u_\star^h), v^h - u_\star^h\rangle \geq 0 \quad \forall v^h \in U^h. \tag{III.15}$$

Choosing $v = u_\star^h \in Y^h \subset Y$ in (III.5), $v^h = \Pi_{U^h}(u_\star)$ in (III.15), and observing that

$$0 \leq \langle \nabla J^h(u_\star^h), \Pi_{U^h}(u_\star) - u_\star^h\rangle = \langle \nabla J^h(u_\star^h), \Pi_{U^h}(u_\star - u_\star^h)\rangle = \langle \nabla J^h(u_\star^h), u_\star - u_\star^h\rangle,$$

since $\nabla J^h(u_\star^h) \in U^h$, we obtain

$$\langle \alpha(u_\star - u_\star^h) + \mathbb{E}[p(u_\star)] - \mathbb{E}[p^h(u_\star^h)], u_\star^h - u_\star\rangle \geq 0.$$

Then introducing $\widetilde{p}^h(u_\star) = \left(S^h\right)^* (S(u_\star + g) - z_d)$, which belongs to $L^2(\Gamma, Y^h)$ since the two operators S and $\left(S^h\right)^*$ are bounded, we obtain

$$\alpha\|u_\star - u_\star^h\|^2 \leq \langle \mathbb{E}[p(u_\star)] - \mathbb{E}[\widetilde{p}^h(u_\star)] + \mathbb{E}[\widetilde{p}^h(u_\star)] - \mathbb{E}[p^h(u_\star^h)], u_\star^h - u_\star\rangle. \tag{III.16}$$

In the following, we will repeatedly use the primal and dual weak formulations (III.8),(III.9) and(III.14), for the continuous problem and its FE approximation, yielding

$$\begin{aligned}\langle \widetilde{p}^h_\omega(u_\star) - p^h_\omega(u^h_\star), u^h_\star - u_\star\rangle &= b_\omega(y^h_\omega(u^h_\star) - y^h_\omega(u_\star), \widetilde{p}^h_\omega(u_\star) - p^h_\omega(u^h_\star)) \\ &= \int_D \underbrace{(y^h_\omega(u^h_\star) - y^h_\omega(u_\star))}_{\pm y_\omega(u_\star)}(y_\omega(u_\star) - y^h_\omega(u^h_\star))\mathrm{d}x \\ &= -\|y_\omega(u_\star) - y^h_\omega(u^h_\star)\|^2 + \int_D (y_\omega(u_\star) - y^h_\omega(u_\star))(y_\omega(u_\star) - y^h_\omega(u^h_\star))\mathrm{d}x \\ &\leq -\|y_\omega(u_\star) - y^h_\omega(u^h_\star)\|^2 + \frac{1}{2}\|y_\omega(u_\star) - y^h_\omega(u_\star)\|^2 + \frac{1}{2}\|y_\omega(u_\star) - y^h_\omega(u^h_\star)\|^2 \\ &\leq -\frac{1}{2}\|y_\omega(u_\star) - y^h_\omega(u^h_\star)\|^2 + \frac{1}{2}\|y_\omega(u_\star) - y^h_\omega(u_\star)\|^2.\end{aligned}$$

Taking the mean over all realizations $\omega \in \Gamma$, using (III.16), and Fubini's theorem we have that

$$\begin{aligned}\alpha\|u_\star - u^h_\star\|^2 + &\frac{1}{2}\mathbb{E}[\|y(u_\star) - y^h(u^h_\star)\|^2] \\ &\leq \mathbb{E}[\langle p(u_\star) - \widetilde{p}^h(u_\star), u^h_\star - u_\star\rangle] + \frac{1}{2}\mathbb{E}[\|y(u_\star) - y^h(u_\star)\|^2] \\ &\leq \frac{1}{2\alpha}\|p(u_\star) - \widetilde{p}^h(u_\star)\|^2 + \frac{\alpha}{2}\|u^h_\star - u_\star\|^2 + \frac{1}{2}\mathbb{E}[\|y(u_\star) - y^h(u_\star)\|^2],\end{aligned}$$

which leads to the claim. □

The FE error $\|u_\star - u^h_\star\|$ is thus completely determined by the approximation properties of the discrete solution operators S^h and $\left(S^h\right)^*$. Using similar arguments as in [HPUU09, Thm. 3.5], we can also control the FE error of the state variable in H^1, i.e. of $\|y(u_\star) - y^h(u^h_\star)\|_{H^1_0}$.

Theorem 10. *With the same notations as in Theorem 9, there exists a constant $C > 0$ independent of h such that*

$$\begin{aligned}&\|u_\star - u^h_\star\|^2 + \mathbb{E}[\|y(u_\star) - y^h(u^h_\star)\|^2] + h^2\mathbb{E}[\|y(u_\star) - y^h(u^h_\star)\|^2_{H^1_0}] \\ &\quad\leq C\{\mathbb{E}[\|p(u_\star) - \widetilde{p}^h(u_\star)\|^2] + \mathbb{E}[\|y(u_\star) - y^h(u_\star)\|^2] + h^2\mathbb{E}[\|y(u_\star) - y^h(u_\star)\|^2_{H^1_0}]\}.\end{aligned} \tag{III.17}$$

Proof. From the uniform coercivity of the bi-linear form $b_\omega(\cdot,\cdot)$, c.f. Assumption 11, it immediately follows

$$\|y_\omega - y^h_\omega\|^2_{H^1_0} \leq \frac{1}{a_{min}}\left\{b_\omega\big(y_\omega - y^h_\omega, y_\omega - \widetilde{y}^h_\omega\big) + b_\omega\big(y_\omega - y^h_\omega, \widetilde{y}^h_\omega - y^h_\omega\big)\right\},$$

where we have used the notation $y_\omega = y_\omega(u_\star)$, $y^h_\omega = y^h_\omega(u^h_\star)$, and $\widetilde{y}^h_\omega = y^h_\omega(u_\star)$. Moreover

$$\begin{aligned}\frac{1}{a_{min}} b_\omega\big(y_\omega - y^h_\omega, y_\omega - \widetilde{y}^h_\omega\big) &\leq \frac{a_{max}}{a_{min}} \|y_\omega - y^h_\omega\|_{H^1_0} \|y_\omega - \widetilde{y}^h_\omega\|_{H^1_0} \\ &\leq \frac{1}{4}\|y_\omega - y^h_\omega\|^2_{H^1_0} + \frac{a^2_{max}}{a^2_{min}} \|y_\omega - \widetilde{y}^h_\omega\|^2_{H^1_0},\end{aligned}$$

as well as

$$\begin{aligned}\frac{1}{a_{min}} b_\omega\big(y_\omega - y^h_\omega, \widetilde{y}^h_\omega - y^h_\omega\big) &\leq \frac{1}{a_{min}} \langle u_\star - u^h_\star, \widetilde{y}^h_\omega - y^h_\omega\rangle \\ &\leq \frac{1}{a_{min}} \langle u_\star - u^h_\star, \widetilde{y}^h_\omega - y_\omega\rangle + \frac{1}{a_{min}} \langle u_\star - u^h_\star, y_\omega - y^h_\omega\rangle \\ &\leq \frac{C_p^2}{2a_{min}} \|u_\star - u^h_\star\|^2 + \frac{1}{2a_{min}} \|y_\omega - \widetilde{y}^h_\omega\|^2_{H^1_0} + \frac{C_p^2}{a^2_{min}} \|u_\star - u^h_\star\|^2 + \frac{1}{4}\|y_\omega - y^h_\omega\|^2_{H^1_0}.\end{aligned}$$

Finally, it follows that

$$\|y_\omega - y^h_\omega\|^2_{H^1_0} \leq C\{\|y_\omega - \widetilde{y}^h_\omega\|^2_{H^1_0} + \|u_\star - u^h_\star\|^2\}$$

and

$$h^2 \mathbb{E}[\|y_\omega - y^h_\omega\|^2_{H^1_0}] \leq h^2 C\{\mathbb{E}[\|y_\omega - \widetilde{y}^h_\omega\|^2_{H^1_0}] + \|u_\star - u^h_\star\|^2\},$$

which, combined with (III.13), completes the proof. □

We can now proceed and estimate the right hand side of (III.17), assuming the primal and dual solutions are sufficiently smooth.

Corollary 1. *Suppose that $y(u_\star), p(u_\star) \in L^2(\Gamma, H^{r+1}(D))$, then we have*

$$\begin{aligned}\|u_\star - u^h_\star\|^2 + \mathbb{E}[\|y(u_\star) - y^h(u^h_\star)\|^2] + h^2 \mathbb{E}[\|y(u_\star) - y^h(u^h_\star)\|^2_{H^1_0}] \\ \leq C h^{2r+2}\{\mathbb{E}[|y_\omega(u_\star)|^2_{H^{r+1}}] + \mathbb{E}[|p_\omega(u_\star)|^2_{H^{r+1}}]\}. \quad \text{(III.18)}\end{aligned}$$

Proof. Under the assumptions of the corollary, the operators $S_\omega, S^*_\omega : L^2(D) \to H^2(D) \cap H^1_0(D)$ are bounded. Using first the Aubin-Nitsche duality argument and then Céa's Lemma (see e.g. [Qua09]), for the first term on the right hand side of (III.17), we find

$$\begin{aligned}\mathbb{E}[\|p(u_\star) - \widetilde{p}^h(u_\star)\|^2] &\leq C\mathbb{E}[h^2 \|p(u_\star) - \widetilde{p}^h(u_\star)\|^2_{H^1_0}] \\ &\leq C\mathbb{E}[h^{2+2r} |p(u_\star)|^2_{H^{r+1}}].\end{aligned}$$

A similar argument holds for the second term on the right hand side of (III.17). Finally the third term on the right hand side of (III.17) can be bounded directly by

$$h^2\mathbb{E}[\|y(u_\star) - y^h(u_\star)\|^2_{H_0^1}] \leq Ch^2\mathbb{E}[Ch^{2r}|y|^2_{H^{r+1}}].$$

All these inequalities added together lead to the claim. □

III.D. Approximation in probability space

In this section we consider the semi-discrete (approximation in probability only) optimal control problem obtained by replacing the exact expectation $\mathbb{E}[\cdot]$ in (III.3) by a suitable quadrature formula $\widehat{E}[\cdot]$. The semi-discrete collocation problem then reads:

$$\begin{cases} \min_{u\in U} \widehat{J}(u) = \frac{1}{2}\widehat{E}[\|y_\omega(u) - z_d\|^2] + \frac{\alpha}{2}\|u\|^2 \\ \text{s.t.} \quad y_{\omega_i}(u) \in Y \quad \text{and} \\ b_{\omega_i}(y_{\omega_i}(u), v) = \langle g + u, v\rangle \quad \forall v \in Y \quad i = 1, \ldots, N. \end{cases} \tag{III.19}$$

This quadrature formula could either be based on deterministic quadrature points or randomly distributed points leading, in this case, to a Monte Carlo type approximation. In particular, if $X : \Gamma \to \mathbb{R}$, $\omega \mapsto X(\omega)$, is a random variable, let $\widehat{E}[X] = \sum_{i=1}^N \zeta_i X(\omega_i)$ be the quadrature operator, where ζ_i are the quadrature weights and ω_i the quadrature knots. In the case of a Monte Carlo approximation, we have $\zeta_i = \frac{1}{N}$ for every i, and ω_i being independent and identically distributed (iid) points in Γ, all distributed according to the measure P.

In the next sub-sections we will particularize results for the cases of a Monte Carlo type quadrature. Although for the sake of notation we present these results for the semi-discrete problem (i.e. continuous in space, discrete in probability), they extend straightforwardly to the fully discrete problem in probability and in space, using a control $\widehat{u}^h$ instead of $\widehat{u}$, a solution of (III.19). We study the deterministic Gaussian-type quadrature method in the appendix.

III.D.1. Monte Carlo method

Consider a Monte Carlo approximation of the expectation appearing in (III.10), namely the exact expectation $\mathbb{E}$ is replaced by $E_{MC}^{\overrightarrow{\omega}}[X(\omega)] := \frac{1}{N}\sum_{i=1}^N X(\omega_i)$, where N denotes the number of ω_i, $i = 1, \ldots, N$, of the random variable ω and denote by $\overrightarrow{\omega} = \{\omega_i\}_{i=1}^N$ the collection of these ω_i. We recall that the use of MC type approximations might be advantageous over a collocation/quadrature approach in cases where p is rough, which is, for example, the case when $a(\cdot, \cdot)$ is a rough random field w.r.t. the random parameter ω or has a short correlation length.

Remark 8. *We stress here that $\widehat{u}$ is a stochastic function because it depends on the N iid realizations $\overrightarrow{\omega} = \{\omega_i\}_{i=1}^N$ of the random variable ω.*

Theorem 11. *Let $\widehat{u}_\star$ be the optimal control of problem* (III.19) *with $\widehat{E} = E_{MC}^{\overrightarrow{\omega}}$ and $u_\star$ be the exact optimal control of the continuous problem* (III.10), *then we have*

$$\frac{\alpha}{2}\mathbb{E}[\|\widehat{u}_\star - u_\star\|^2] + \mathbb{E}[\|y(u_\star) - y(\widehat{u}_\star)\|^2] \leq \frac{1}{N}\frac{1}{2\alpha}\mathbb{E}[\|p(\widehat{u}_\star)\|^2].$$

Proof. Similarly to the proof of Theorem 9, the two optimality conditions read

$$\langle \nabla J(u_\star), v_1 - u_\star\rangle \geq 0 \quad \forall v_1 \in U \tag{III.20}$$

and

$$\langle \nabla J_{MC}(\widehat{u}_\star), v_2 - \widehat{u}_\star\rangle \geq 0 \quad \forall v_2 \in U \tag{III.21}$$

with

$$\nabla J_{MC}(\widehat{u}_\star) = \alpha\widehat{u}_\star + E_{MC}^{\overrightarrow{\omega}}[p(\widehat{u}_\star)] \quad p(\widehat{u}_\star) := S^*(S\widehat{u}_\star - z).$$

Choosing $v_1 = \widehat{u}_\star$ in (III.20) and $v_2 = u_\star$ in (III.21) we obtain:

$$\langle \alpha(u_\star - \widehat{u}_\star) + \mathbb{E}[p(u_\star)] - E_{MC}^{\overrightarrow{\omega}}[p(\widehat{u}_\star)], \widehat{u}_\star - u_\star\rangle \geq 0,$$

which implies

$$\alpha\|u_\star - \widehat{u}_\star\|^2 \leq \langle \mathbb{E}[p(u_\star)] - E_{MC}^{\overrightarrow{\omega}}[p(u_\star)] + E_{MC}^{\overrightarrow{\omega}}[p(u_\star)] - E_{MC}^{\overrightarrow{\omega}}[p(\widehat{u}_\star)], \widehat{u}_\star - u_\star\rangle. \tag{III.22}$$

We can split the right hand side of (III.22) into two parts:

$$\langle \mathbb{E}[p(u_\star)] - E_{MC}^{\overrightarrow{\omega}}[p(u_\star)], \widehat{u}_\star - u_\star\rangle \leq \frac{1}{2\alpha}\|\mathbb{E}[p(u_\star)] - E_{MC}^{\overrightarrow{\omega}}[p(u_\star)]\|^2 + \frac{\alpha}{2}\|\widehat{u}_\star - u_\star\|^2$$

Moreover, for every $i = 1, \cdots, N$

$$\begin{aligned}\langle \widehat{u}_\star - u_\star, p_{\omega_i}(u_\star) - p_{\omega_i}(\widehat{u}_\star)\rangle &= b_{\omega_i}(y_{\omega_i}(\widehat{u}_\star) - y_{\omega_i}(u_\star), p_{\omega_i}(u_\star) - p_{\omega_i}(\widehat{u}_\star)) \\ &= \langle y_{\omega_i}(u_\star) - y_{\omega_i}(\widehat{u}_\star), y_{\omega_i}(\widehat{u}_\star) - y_{\omega_i}(u_\star)\rangle \\ &= -\|y_{\omega_i}(u_\star) - y_{\omega_i}(\widehat{u}_\star)\|^2,\end{aligned}$$

leading to

$$\langle \widehat{u}_\star - u_\star, E_{MC}^{\overrightarrow{\omega}}[p(u_\star)] - E_{MC}^{\overrightarrow{\omega}}[p(\widehat{u}_\star)]\rangle \leq -E_{MC}^{\overrightarrow{\omega}}[\|y(u_\star) - y(\widehat{u}_\star)\|^2]$$

We finally take the expectation of (III.22), w.r.t. the random sample $\overrightarrow{\omega} = \{\omega_i\}_{i=1}^N$ and exploit the fact that the Monte Carlo estimator is unbiased, that is $\mathbb{E}[E_{MC}^{\overrightarrow{\omega}}[X(\omega)]] = \mathbb{E}[X]$ for a random variable $X : \Gamma \to \mathbb{R}$.

$$\begin{aligned}\mathbb{E}[\frac{\alpha}{2}\|\widehat{u}_\star - u_\star\|^2 + E_{MC}^{\overrightarrow{\omega}}[\|y(u_\star) - y(\widehat{u}_\star)\|^2] &= \frac{\alpha}{2}\mathbb{E}[\|\widehat{u}_\star - u_\star\|^2 + \mathbb{E}[\|y(u_\star) - y(\widehat{u}_\star)\|^2] \\ &\leq \frac{1}{2\alpha}\mathbb{E}[\|\mathbb{E}[p(\widehat{u}_\star)] - E_{MC}^{\overrightarrow{\omega}}[p(\widehat{u}_\star)]\|^2] \\ &\leq \frac{1}{2\alpha}\mathbb{E}[\|\frac{1}{N}\sum_{i=1}^{N} p_{\omega_i}(\widehat{u}_\star) - \mathbb{E}[p(\widehat{u}_\star)]\|^2] \\ &\leq \frac{1}{2\alpha}\mathbb{E}[\frac{1}{N^2}\sum_{i=1}^{N} \|p_{\omega_i}(\widehat{u}_\star) - \mathbb{E}[p(\widehat{u}_\star)]\|^2] \\ &\leq \frac{1}{2\alpha}\frac{1}{N}\mathbb{E}[\|p(\widehat{u}_\star) - \mathbb{E}[p(\widehat{u}_\star)]\|^2] \\ &\leq \frac{1}{2\alpha}\frac{1}{N}\mathbb{E}[\|p(\widehat{u}_\star)\|^2]\end{aligned}$$

what finishes the proof of the theorem. □

Theorem 11 shows that the semi-discrete optimal control $\widehat{u}_\star$ converges at the usual MC rate of $1/\sqrt{N}$ in the root mean squared sense, with the constant being proportional to $\sqrt{\mathbb{E}[\|p(\widehat{u}_\star)\|^2]}$.

III.E. Steepest descent method for fully discrete problem

Now we focus on a class of optimization methods to approximate the fully discrete minimization problem, using the Monte Carlo estimator to approximate the expectation in (III.11)

$$\begin{cases} \min_{u^h \in U^h} J_{MC}(u^h) = \frac{1}{2}E_{MC}^{\overrightarrow{\omega}}[\|y_\omega^h(u^h) - z_d\|^2] + \frac{\alpha}{2}\|u^h\|^2 \\ \text{s.t.} \quad y_\omega^h(u^h) \in Y^h \quad \text{and} \\ b_\omega(y_\omega^h(u^h), v^h) = \langle g + u^h, v^h\rangle \quad \forall v^h \in Y^h, \quad \text{for a.e. } \omega \in \Gamma. \end{cases} \tag{III.23}$$

Specifically, we consider a simple gradient method. The gradient method reads:

$$\widehat{u}_{j+1}^h = \widehat{u}_j^h - \tau E_{MC}^{\overrightarrow{\omega}}[\nabla f^h(\widehat{u}_j^h, \omega)], \tag{III.24}$$

where $f^h(u,\omega) = \frac{1}{2}\|y_\omega^h(u) - z_d\|^2 + \frac{\alpha}{2}\|u^h\|^2$. Here, the index j represents the j-th iteration in the optimization recursion (III.24), while the superscript h denotes that we discretize the control u as well as the underlying PDE using Finite Elements on a fixed mesh of characteristic size h.

We first analyze the convergence of the continuous version of (III.24), i.e. of

$$u_{j+1} = u_j - \tau \mathbb{E}[\nabla f(u_j, \omega)] \ . \tag{III.25}$$

For this we prove a Lipschitz and a strong convexity condition for the function $f(u, \omega)$ for a.e. $\omega \in \Gamma$; which is still valid when replacing $f(u, \omega)$ by its discrete version $f^h(u^h, \omega_i)$.

Lemma 6 (Lipschitz condition)**.** *For the elliptic problem* (III.4) *and* $f(u, \omega)$ *as in* (III.3) *it holds that:*

$$\|\nabla f(u_1, \omega) - \nabla f(u_2, \omega)\| \leq L \|u_1 - u_2\| \quad \forall u_1, u_2 \in U \text{ and a.e. } \omega \in \Gamma, \tag{III.26}$$

with $L = \alpha + \frac{C_p^4}{a_{min}^2}$*, where* C_p *is the Poincaré constant. For the Finite Element approximation as in* (III.11) *the same inequality holds with the same constant*

$$\|\nabla f^h(u_1^h, \omega) - \nabla f^h(u_2^h, \omega)\| \leq L \|u_1^h - u_2^h\| \quad \forall u_1^h, u_2^h \in U^h \text{ and a.e. } \omega \in \Gamma.$$

Proof. For a.e. $\omega \in \Gamma$, and every $u, u' \in U$ we have that

$$\nabla f(u', \omega) - \nabla f(u, \omega) = \alpha(u' - u) + p_\omega(u') - p_\omega(u), \tag{III.27}$$

and

$$\begin{aligned}
\|p_\omega(u') - p_\omega(u)\|^2 &\leq C_p^2 \|\nabla_x p_\omega(u') - \nabla_x p_\omega(u)\|^2 \\
&\leq \frac{C_p^2}{a_{min}} b_\omega\big(p_\omega(u') - p_\omega(u), p_\omega(u') - p_\omega(u)\big) \\
&\leq \frac{C_p^2}{a_{min}} \langle p_\omega(u') - p_\omega(u), y_\omega(u') - y_\omega(u) \rangle \\
&\leq \frac{C_p^2}{a_{min}} \|p_\omega(u') - p_\omega(u)\| \|y_\omega(u') - y_\omega(u)\|.
\end{aligned}$$

With same arguments we find that

$$\begin{aligned}
\|y_\omega(u') - y_\omega(u)\|^2 &\leq C_p^2 \|\nabla_x y_\omega(u') - \nabla_x y_\omega(u)\|^2 \\
&\leq \frac{C_p^2}{a_{min}} b_\omega\big(y_\omega(u') - y_\omega(u), y_\omega(u') - y_\omega(u)\big) \\
&\leq \frac{C_p^2}{a_{min}} \langle y_\omega(u') - y_\omega(u), u' - u \rangle \\
&\leq \frac{C_p^2}{a_{min}} \|y_\omega(u') - y_\omega(u)\| \|u' - u\|.
\end{aligned}$$

Combining (III.27) with the two last estimates, we find

$$\begin{aligned}\|\nabla f(u',\omega)-\nabla f(u,\omega)\| &\leq \alpha\|u'-u\|+\|p_\omega(u')-p_\omega(u)\|\\ &\leq \Big(\alpha+\frac{C_p^4}{a_{min}^2}\Big)\|u'-u\|.\end{aligned}$$

The proof in the FE setting follows verbatim the above one. □

Lemma 7 (Strong Convexity). *For the elliptic problem* (III.4) *and* $f(u,\omega)$ *as in* (III.3) *it holds that*

$$\frac{l}{2}\|u_1-u_2\|^2 \leq \langle\nabla f(u_1,\omega)-\nabla f(u_2,\omega),u_1-u_2\rangle \quad \forall u_1,u_2\in U \text{ and a.e. } \omega\in\Gamma, \quad \text{(III.28)}$$

with $l=2\alpha$. *The same estimate holds for the FE approximation as in* (III.11), *namely:*

$$\frac{l}{2}\|u_1^h-u_2^h\|^2 \leq \langle\nabla f^h(u_1^h,\omega)-\nabla f^h(u_2^h,\omega),u_1^h-u_2^h\rangle \quad \forall u_1^h,u_2^h\in U^h \text{ and a.e. } \omega\in\Gamma.$$

Proof. For every $\omega\in\Gamma$, and every $u,u'\in U$:

$$\begin{aligned}\langle u'-u,\nabla f(u',\omega)-\nabla f(u,\omega)\rangle &= \langle u'-u,\alpha(u'-u)+p_\omega(u')-p_\omega(u)\rangle\\ &= \alpha\|u'-u\|^2+\langle u'-u,p_\omega(u')-p_\omega(u)\rangle\\ &= \alpha\|u'-u\|^2+b_\omega\big(y_\omega(u')-y_\omega(u),p_\omega(u')-p_\omega(u)\big)\\ &= \alpha\|u'-u\|^2+\langle y_\omega(u')-y_\omega(u),y_\omega(u')-y_\omega(u)\rangle\\ &= \alpha\|u'-u\|^2+\|y_\omega(u')-y_\omega(u)\|^2\\ &\geq \alpha\|u'-u\|^2\end{aligned}$$

The same proof applies to the FE case. □

Based on the results of Lemmas 6 and 7, it is straightforward to show the convergence of the iterates. We state the result for the gradient method for the continuous problem (III.25) in the following Lemma and the result for the fully discretized problem(III.24) in Theorem 12.

Lemma 8. *Let* $u_\star$ *be the optimal solution of the control problem* (III.10) *and* $\{u_j\}_{j\in\mathbb{N}}$ *the iterations produced by* (III.25). *Then for any* $0<\tau<l/L^2$ *we have*

$$\|u_{j+1}-u_\star\|^2 \leq (1-\tau l+\tau^2L^2)\|u_j-u_\star\|^2 \leq (1-\tau l+\tau^2L^2)^{j+1}\|u_0-u_\star\|^2, \quad \text{(III.29)}$$

and $\|u_j-u_\star\|\to 0$ *as* $j\to\infty$.

Proof. Since $u_\star$ satisfies the optimality condition $\nabla J(u_\star)=0$ we have

$$u_{j+1}-u_\star = u_j-u_\star-\tau\mathbb{E}[\nabla f(u_j,\omega)-\nabla f(u_\star,\omega)].$$

Consequently,

$$\begin{aligned}\|u_{j+1} - u_\star\|^2 &= \|u_j - u_\star\|^2 + \tau^2 \|\mathbb{E}[\nabla f(u_j, \omega) - \nabla f(u_\star, \omega)]\|^2 \\ &\quad - 2\tau \langle u_j - u_\star, \mathbb{E}[\nabla f(u_j, \omega) - \nabla f(u_\star, \omega)]\rangle \\ &\le (1 - \tau l + \tau^2 L^2)\|u_j - u_\star\|^2.\end{aligned}$$

The condition $0 < \tau < l/L^2$ guarantees that $0 < 1 - \tau l + \tau^2 L^2 < 1$ and the claim follows. □

As mentioned before, we now provide an error bound for the approximate solution $\widehat{u}_j^h$ defined in (III.24), as a function of the discretization parameters j, h, and N.

Theorem 12. *Let $\widehat{u}_j^h$ be the solution produced by* (III.24) *at the j-th iteration and denote by $u_\star$ the solution of the optimal problem* (III.10). *Then under the assumptions of Corollary 1, there exist constants $C_1, C_2, C_3 > 0$ such that*

$$\mathbb{E}[\|\widehat{u}_j^h - u\|^2] \le C_1 e^{-\rho j} + \frac{C_2}{N} + C_3 h^{2r+2}\ , \tag{III.30}$$

with $\rho = -\log(1 - \tau l + \tau^2 L^2)$ for $0 < \tau < l/L^2$.

Proof. The global error can be decomposed as follows:

$$\mathbb{E}[\|\widehat{u}_j^h - u_\star\|^2] \le 3\underbrace{\mathbb{E}[\|\widehat{u}_j^h - \widehat{u}^{h,*}\|^2]}_{\text{gradient}} + 3\underbrace{\mathbb{E}[\|\widehat{u}^{h,*} - u_\star^h\|^2]}_{\text{MC}} + 3\underbrace{\mathbb{E}[\|u_\star^h - u_\star\|^2]}_{\text{FE error}}.$$

The first term $\mathbb{E}[\|\widehat{u}_j^h - \widehat{u}^{h,*}\|^2]$ quantifies the convergence of the finite dimensional steepest descent algorithm and can be estimated as in Lemma 8. In fact, for any sample $\overrightarrow{\omega} = \{\omega_i\}_{i=1}^N$ we have

$$\|\widehat{u}_j^h - \widehat{u}^{h,*}\|^2 \le (1 - \tau l + \tau^2 L^2)^j \|\widehat{u}_0^h - \widehat{u}^{h,*}\|^2 = e^{-\rho j}\|\widehat{u}_0^h - \widehat{u}^{h,*}\|^2.$$

with $\rho = -\log(1 - \tau l + \tau^2 L^2)$. Hence taking expectation w.r.t. $\overrightarrow{\omega}$,

$$\mathbb{E}[\|\widehat{u}_j^h - \widehat{u}^{h,*}\|^2] \le e^{-\rho j}\mathbb{E}[\|\widehat{u}_0^h - \widehat{u}^{h,*}\|^2].$$

The second term $\mathbb{E}[\|\widehat{u}^{h,*} - u_\star^h\|^2]$ accounts for the standard MC error and can be controlled as in Theorem 11 (applied on the FE approximation) leading to

$$\mathbb{E}[\|\widehat{u}^{h,*} - u_\star^h\|^2] \le \frac{1}{\alpha^2 N}\mathbb{E}[\|p(\widehat{u}^h)\|^2].$$

Finally, the term $\mathbb{E}[\|u_\star^h - u_\star\|^2]$ can be controlled by the result in Corollary 1, namely by

$$\|u_\star^h - u_\star\|^2 \leq C\big(\mathbb{E}[|y_\omega(u_\star)|^2_{H^{r+1}}] + \mathbb{E}[|p_\omega(u_\star)|^2_{H^{r+1}}]\big)h^{2r+2},$$

so that the claim follows. □

We conclude this Section by analyzing the complexity of the Algorithm 1 based on the optimization scheme (III.24). We assume that the primal and dual problems can be solved, using a triangulation with mesh size h, in computational time $C_h = O(h^{-d\gamma})$. Here, $\gamma \in [1,3]$ is a parameter representing the efficiency of the linear solver used (e.g. $\gamma = 3$ for a direct solver and $\gamma = 1$ up to a logarithm factor for an optimal multigrid solver), while n is the dimension of the physical space. Hence the overall computational work W of j gradient iterations is proportional to $W \simeq 2Njh^{-d\gamma}$.

Corollary 2. *In order to achieve a given tolerance $O(tol)$, i.e. to guarantee that $\mathbb{E}[\|\widehat{u}_j^h - u\|^2] \lesssim tol^2$, the total required computational work is bounded by*

$$W \lesssim tol^{-2-\frac{d\gamma}{r+1}} |\log(tol)|.$$

Proof. To achieve a tolerance $O(tol)$, we can equidistribute the precision tol^2 over the three terms in (III.30). This leads to the choices given in Algorithm 1:

$$j_{max} \simeq -\log(tol), \quad h \simeq tol^{\frac{1}{r+1}}, \quad N \simeq tol^{-2}.$$

Hence the total cost for computing a solution $\widehat{u}_{j_{max}}^h$ that achieves the required tolerance is $W \simeq 2Nj_{max}h^{-d\gamma} = tol^{-2-\frac{d\gamma}{r+1}}|\log(tol)|$ as claimed. □

We propose a description of the algorithm used in this Section, in Algorithm 1.
The second (MC) term in the error bound (III.30) C_2/N is numerically a problem/limitation to compute efficiently a solution. That is why in the following Section we combine the first two terms, using Stochastic Gradient techniques.

III.F. Stochastic Gradient with fixed mesh size.

As an alternative to the fixed MC gradient method (III.24) considered in Section III.E, in which the sample size N is fixed beforehand and a full sample average is computed at each iteration, here we consider a variant, known in literature as Stochastic Approximation (SA) or Stochastic Gradient (SG) [RM51, PJ92, SDR09, SRB13, DB16].

The classic version of such a method, the so-called Robbins-Monro method, works as follows. Within the steepest descent algorithm the exact gradient $\nabla J = \nabla\mathbb{E}[f] = \mathbb{E}[\nabla f]$

Algorithm 1: Steepest descent method for fully discrete problem

```
Data:
Given a desired tolerance $tol$:
Choose $\tau < \frac{l}{L^2}$, $j_{max} \simeq -\log(tol)$, $N_{MC} \simeq tol^{-2}$, $h \simeq tol^{\frac{1}{r+1}}$
Generate $N_{MC}$ iid realizations of the random field $a_i = a(\cdot, \omega_i)$, $i = 1, \dots, N_{MC}$.
initialization:
$u = 0$;
for $j = 1, \dots, j_{max}$ do
    $\widehat{p} = 0$;
    for $i = 1, \dots, N_{MC}$ do
        solve primal problem by FE $\rightarrow y(a_i, u)$
        solve dual problem by FE $\rightarrow p(a_i, u)$
        update $\widehat{p} = \widehat{p} + p(a_i, u)/N_{MC}$
    end
    $\widehat{\nabla J} = \alpha u + \widehat{p}$
    $u = u - \tau \widehat{\nabla J}$
end
```

is replaced by $\nabla f(\cdot, \omega_j)$, where the random variable ω_j is re-sampled independently at each iteration of the steepest-descent method:

$$u_{j+1} = u_j - \tau_j \nabla f(u_j, \omega_j). \tag{III.31}$$

Here, τ_j is the step-size of the algorithm and is decreasing as $1/j$ in the usual approach. We consider a generalization of this method, in which the point-wise gradient $\nabla f(\cdot, \omega_j)$ is replaced by a sample average over N_j iid realizations which are drawn independently of the previous iterations. More precisely, let $\overrightarrow{\omega_j} = (\omega_j^{(1)}, \cdots, \omega_j^{(N_j)})$, then we define the recursion as

$$u_{j+1} = u_j - \tau_j E_{MC}^{\overrightarrow{\omega_j}}[\nabla f(u_j, \omega)], \tag{III.32}$$

where $E_{MC}^{\overrightarrow{\omega_j}}[\nabla f(u, \omega)] = \frac{1}{N_j} \sum_{i=1}^{N_j} \nabla f(u, \omega_j^{(i)})$ is the usual Monte Carlo estimator. Notice that the Robbins-Monro method is a special case of this scheme, namely with $N_j = 1$ for all j. In what follows, we investigate optimal choices of the sequences $\{\tau_j\}_j$ and $\{N_j\}_j$, and the overall computational complexity of the corresponding algorithm. First we analyze the continuous version (i.e. no Finite Element discretization).

Theorem 13. *Let $u_\star$ be the solution of the continuous OCP* (III.10) *and denote by u_j the j-th iterate of* (III.32). *Then it holds that*

$$\mathbb{E}[\|u_{j+1} - u_\star\|^2] \leq c_j \mathbb{E}[\|u_j - u_\star\|^2] + \frac{2\tau_j^2}{N_j} \mathbb{E}[\|\nabla f(u_\star, \omega)\|^2], \tag{III.33}$$

with $c_j := 1 - \tau_j l + L^2 \Big(1 + \frac{2}{N_j}\Big) \tau_j^2$.

Proof. Using inequalities (III.26) and (III.28), we can formulate a recursive formula to control the error between successive iterations. As each iteration uses an independent sample, we need to keep track of the history of the sampling $\omega_{[j-1]} = \{\overrightarrow{\omega_1}, \dots, \overrightarrow{\omega_{j-1}}\}$ to be able to define u_j. Thus we introduce the conditional expectation $G[\cdot] = \mathbb{E}[\cdot | \omega_{[j-1]}]$. Using $\mathbb{E}[\nabla f(u_\star, \omega)] = 0$, we have:

$$\begin{aligned} u_{j+1} - u_\star &= u_j - u_\star - \tau_j E_{MC}^{\overrightarrow{\omega_j}}[\nabla f(u_j, \overrightarrow{\omega_j})] + \tau_j \mathbb{E}[\nabla f(u_\star, \omega)] \\ &= u_j - u_\star - \tau_j G[\nabla f(u_j, \omega)] + \tau_j \mathbb{E}[\nabla f(u_\star, \omega)] + \tau_j \big(G[\nabla f(u_j, \omega)] - E_{MC}^{\overrightarrow{\omega_j}}[\nabla f(u_j, \overrightarrow{\omega_j})]\big) \\ &= u_j - u_\star - \tau_j T_1 + \tau_j T_2, \end{aligned}$$

with $T_1 := G[\nabla f(u_j, \omega)] - \mathbb{E}[\nabla f(u_\star, \omega)]$ and $T_2 := G[\nabla f(u_j, \omega)] - E_{MC}^{\overrightarrow{\omega_j}}[\nabla f(u_j, \overrightarrow{\omega_j})]$. Hence,

$$\begin{aligned} \|u_{j+1} - u_\star\|^2 =& \|u_j - u_\star\|^2 + \tau_j^2 \|T_1\|^2 + \tau_j^2 \|T_2\|^2 \\ & - 2\tau_j \langle u_j - u_\star, T_1 \rangle + 2\tau_j \langle u_j - u_\star, T_2 \rangle - 2\tau_j^2 \langle T_1, T_2 \rangle. \end{aligned}$$

Moreover, by definition of T_1, we find:

$$\begin{aligned} \|T_1\|^2 &= \|G[\nabla f(u_j, \omega)] - \mathbb{E}[\nabla f(u_\star, \omega)]\|^2 \\ &= \|G[\nabla f(u_j, \omega) - \nabla f(u_\star, \omega)]\|^2 \qquad [\overrightarrow{\omega_j} \text{ being independent of } \omega_{[j-1]}] \\ &= \int_D \big(G[\nabla f(u_j, \omega) - \nabla f(u_\star, \omega)]\big)^2 \mathrm{d}x \\ &\leq \int_D G[|\nabla f(u_j, \omega) - \nabla f(u_\star, \omega)|^2] \mathrm{d}x \qquad [\text{Jensen}'\text{s inequality}] \\ &= G[\|\nabla f(u_j, \omega) - \nabla f(u_\star, \omega)\|^2] \\ &\leq L^2 G[\|u_j - u_\star\|^2], \end{aligned}$$

where we have used Jensen's inequality for conditional expectation: $\phi(G[X]) \leq G[\phi(X)]$ for ϕ convex. See e.g.[Wil91].

Then taking the expectation over all the history sampling $\omega_{[j-1]}$, we have:

$$\begin{aligned} \mathbb{E}[\|T_1\|^2] &\leq L^2 \mathbb{E}[G[\|u_j - u_\star\|^2]] \\ &= L^2 \mathbb{E}[\|u_j - u_\star\|^2], \end{aligned}$$

and

$$\begin{aligned}\mathbb{E}[\langle u_j - u_\star, T_1\rangle] &= \mathbb{E}[\langle u_j - u_\star, G[\nabla f(u_j,\omega) - \nabla f(u_\star,\omega)]\rangle]\\ &= \mathbb{E}[G[\langle u_j - u_\star, \nabla f(u_j,\omega) - \nabla f(u_\star,\omega)\rangle]]\\ &\geq \mathbb{E}[G[\frac{l}{2}\|u_j - u_\star\|^2]] && \text{[Strong Convexity (III.28)]}\\ &= \frac{l}{2}\mathbb{E}[\|u_j - u_\star\|^2].\end{aligned}$$

Concerning the term T_2, it holds that,

$$\|T_2\|^2 = \|G\left[\nabla f(u_j,\omega)\right] - E_{MC}^{\overrightarrow{\omega_j}}\left[\nabla f(u_j,\omega)\right]\|^2 = \int_D \Big(G[\nabla f(u_j,\omega)] - E_{MC}^{\overrightarrow{\omega_j}}\left[\nabla f(u_j,\omega)\right]\Big)^2 \mathrm{d}x.$$

Again, taking the expectation w.r.t. $\omega_{[j]}$ yields

$$\begin{aligned}\mathbb{E}\left[\|T_2\|^2\right] &= \mathbb{E}\left[\int_D \left(\frac{1}{N_j}\sum_{i=1}^{N_j}\Big(G[\nabla f(u_j,\omega)] - \nabla f\left(u_j,\omega_j^{(i)}\right)\Big)\right)^2 \mathrm{d}x\right]\\ &= \mathbb{E}\left[\int_D \frac{1}{N_j^2}\sum_{i,l=1}^{N_j}\Big(\nabla f\left(u_j,\omega_j^{(i)}\right) - G\left[\nabla f\left(u_j,\omega\right)\right]\Big)\Big(\nabla f\left(u_j,\omega_j^{(l)}\right) - G\left[\nabla f\left(u_j,\omega\right)\right]\Big)\mathrm{d}x\right]\\ &= \int_D \frac{1}{N_j^2}\sum_{i,l=1}^{N_j}\mathbb{E}\left[\Big(\nabla f\left(u_j,\omega_j^{(i)}\right) - G\left[\nabla f\left(u_j,\omega\right)\right]\Big)\Big(\nabla f\left(u_j,\omega_j^{(l)}\right) - G\left[\nabla f\left(u_j,\omega\right)\right]\Big)\right]\mathrm{d}x\\ &= \int_D \frac{1}{N_j^2}\sum_{i,l=1}^{N_j}\mathbb{E}\left[G\left[\Big(\nabla f\left(u_j,\omega_j^{(i)}\right) - G\left[\nabla f(u_j,\omega)\right]\Big)\Big(\nabla f\left(u_j,\omega_j^{(l)}\right) - G[\nabla f(u_j,\omega)]\Big)\right]\right]\mathrm{d}x.\end{aligned}$$

Observe that, conditional upon $\omega_{[j-1]}$, the random variables $Y_i = \nabla f(u_j,\omega_j^{(i)}) - G[\nabla f(u_j,\omega)]$, $i = 1,\ldots,N_j$, are mutually independent and have zero mean, i.e. $\mathbb{E}\left[Y_i|\omega_{[j-1]}\right] = G[Y_i] = 0$

and $G(Y_iY_j)=0$ when $i\neq j$. Therefore it follows that

$$\begin{aligned}
\mathbb{E}\left[\|T_2\|^2\right] &= \int_D \frac{1}{N_j^2}\sum_{i=1}^{N_j}\mathbb{E}\left[G\left[\left(\nabla f\left(u_j,\omega_j^{(i)}\right)-G\left[\nabla f\left(u_j,\omega\right)\right]\right)^2\right]\right]\mathrm{d}x\\
&=\mathbb{E}\left[\int_D \frac{1}{N_j}G\left[\left(\nabla f\left(u_j,\omega\right)-G\left[\nabla f(u_j,\omega)\right]\right)^2\right]\mathrm{d}x\right]\\
&\leq \mathbb{E}\left[\int_D \frac{1}{N_j}G\left[\nabla f^2(u_j,\omega)\right]\mathrm{d}x\right]\\
&=\frac{1}{N_j}\mathbb{E}\left[\|\nabla f(u_j,\omega)\|^2\right]\\
&\leq \frac{2}{N_j}\mathbb{E}\left[\|\nabla f(u_j,\omega)-\nabla f(u_\star,\omega)\|^2+\|\nabla f(u_\star,\omega)\|^2\right] \qquad \text{[Lipschitz condition (III.26)]}\\
&\leq \frac{2L^2}{N_j}\mathbb{E}\left[\|u_j-u_\star\|^2\right]+\frac{2}{N_j}\mathbb{E}\left[\|\nabla f(u_\star,\omega)\|^2\right].
\end{aligned}$$

Finally, we have that

$$\begin{aligned}
\mathbb{E}[\langle u_j-u_\star,T_2\rangle] &= \mathbb{E}[G[\langle u_j-u_\star,T_2\rangle]]\\
&=\mathbb{E}[\langle u_j-u_\star,G[T_2]\rangle]\\
&=\frac{1}{N_j}\sum_{i=1}^{N_j}\mathbb{E}[\langle u_j-u_\star,G[Y_i]\rangle]\\
&=0,
\end{aligned}$$

and, similarly, $\mathbb{E}[\langle T_1,T_2\rangle]=\mathbb{E}[G[\langle T_1,T_2\rangle]]=\mathbb{E}[\langle T_1,G[T_2]\rangle]=0$, which concludes the proof. □

We now consider the FE version of (III.32) and focus on the common setting $(\tau_j,N_j)=(\tau_0/j,\overline{N})$, which is a generalization of Robbins-Monro method:

$$u_{j+1}^h=u_j^h-\frac{\tau_0}{j}E_{MC}^{\overrightarrow{\omega_j}}[\nabla f^h(u_j^h,\omega)] \tag{III.34}$$

with $\overrightarrow{\omega_j}:=(\omega_j^{(1)},\cdots,\omega_j^{(\overline{N})})$.

Theorem 14. *Suppose that the assumptions of Corollary 1 hold and let u_j^h denote the j-th iterate of* (III.34). *For the choice $(\tau_j,N_j)=(\tau_0/j,\overline{N})$ with $\tau_0>1/l$ we have*

$$\mathbb{E}[\|u_j^h-u_\star\|^2]\leq D_1j^{-1}+D_2h^{2r+2}\ , \tag{III.35}$$

for suitable constants $D_1,D_2>0$ independent of j and h.

Proof. The factor c_j in (III.33) becomes in this case

$$c_j = 1 - \frac{\tau_0 l}{j} + \frac{\tau_0^2 L^2}{j^2}\Big(1 + \frac{2}{N}\Big).$$

We use the recursive formula (III.33) and set as before $u_\star^h$ the exact optimal control for the FE problem defined in (III.11). We emphasize that (III.11) has no approximation in the probability space. Setting $a_j = \mathbb{E}[\|u_j^h - u_\star^h\|^2]$ and $\beta_j = \frac{2\tau_j^2}{N_j}\mathbb{E}[\|\nabla f(u_\star^h,\omega)\|^2]$, from (III.33) applied to the sequence of Finite Element solutions $\{u_j^h\}_{j>0}$ we find

$$\begin{aligned} a_{j+1} \leq & c_j a_j + \beta_j \\ \leq & c_j c_{j-1} a_{j-1} + c_j \beta_{j-1} + \beta_j \\ \leq & \cdots \\ \leq & \underbrace{\Big(\prod_{i=1}^{j} c_i\Big)}_{=\kappa_j} a_1 + \underbrace{\sum_{i=1}^{j} \beta_i \prod_{l=i+1}^{j} c_l}_{=\mathcal{B}_j}\,. \end{aligned} \tag{III.36}$$

For the first term κ_j, computing its logarithm, we have,

$$\log(\kappa_j) = \sum_{i=1}^{j} \log(1 - \frac{\tau_0 l}{i} + \frac{M}{i^2}) \leq \sum_{i=1}^{j} \frac{-\tau_0 l}{i} + \sum_{i=1}^{j} \frac{M}{i^2},$$

where we have set $M = \tau_0^2 L^2\big(1 + \frac{2}{N}\big)$. Thus

$$\log(\kappa_j) \leq -\tau_0 l \log j + M', \ \text{ with } \ M' = \sum_{i=1}^{\infty} \frac{M}{i^2},$$

and $\kappa_j \lesssim j^{-\tau_0 l}$. For the second term $\mathcal{B}_j$ in (III.36) we have:

$$\mathcal{B}_j = \sum_{i=1}^{j} \beta_i \prod_{k=i+1}^{j} c_k \leq \sum_{i=1}^{j} \frac{S}{i^2} \underbrace{\prod_{k=i+1}^{j} \Big(1 - \frac{\tau_0 l}{k} + \frac{\tau_0^2 L^2}{k^2}\Big)}_{=K_{ij}}, \ \text{ with } \ S = \frac{2\tau_0^2}{\overline{N}}\mathbb{E}[\|\nabla f(u_\star^h,\omega)\|^2].$$

For the term K_{ij} we can proceed as follow:

$$\begin{aligned}\log(K_{ij}) &= \sum_{k=i+1}^{j} \log\left(1 - \frac{\tau_0 l}{k} + \frac{M}{k^2}\right)\\ &\leq \sum_{k=i+1}^{j} \Big(- \frac{\tau_0 l}{k} + \frac{M}{k^2}\Big)\\ &\leq -\tau_0 l(\log(j+1) - \log(i+1)) + M\left(\frac{1}{i} - \frac{1}{j}\right),\end{aligned}$$

which shows that

$$K_{ij} \leq (j+1)^{-\tau_0 l}(i+1)^{\tau_0 l} \exp\left(M\left(\frac{1}{i} - \frac{1}{j}\right)\right).$$

It follows that

$$\begin{aligned}\mathcal{B}_j &\leq (j+1)^{-\tau_0 l} \underbrace{\exp\left(-\frac{M}{j}\right)}_{\leq 1} \sum_{i=1}^{j} S i^{\tau_0 l-2} \underbrace{\exp\left(\frac{M}{i}\right)}_{\leq \exp(M)}\\ &\leq S\exp(M)(j+1)^{-\tau_0 l} \sum_{i=1}^{j} i^{\tau_0 l-2} \lesssim j^{-1},\end{aligned}$$

for $\tau_0 > 1/l$. Eventually, we obtain the following upper bound, for two constants $D_3 > 0$ and $D_4 > 0$:

$$a_{j+1} \leq D_3 j^{-\tau_0 l} a_1 + D_4 j^{-1}. \tag{III.37}$$

From the condition $\tau_0 > \frac{1}{l}$, we conclude that

$$a_{j+1} \leq D_1 j^{-1}, \tag{III.38}$$

with D_1 possibly depending in $\|u_0^h - u_\star^h\|$. Finally splitting the error as

$$\mathbb{E}[\|u_j^h - u_\star\|^2] \leq 2\mathbb{E}[\|u_j^h - u_\star^h\|^2] + 2\mathbb{E}[\|u_\star^h - u_\star\|^2],$$

and using (III.18) to bound the second term, the claim follows. □

We propose a description of the SG algorithm 2 with fixed mesh size, used in Section III.F.

We conclude this section by analyzing the complexity of the Algorithm 2.

Corollary 3. *To achieve a given tolerance $O(tol)$, i.e. to guarantee that* $\mathbb{E}[\|u_j^h - u_\star\|^2] \lesssim$

```
Algorithm 2: Stochastic Gradient with fixed mesh size algorithm, with $\overline{N} = 1$.
Data:
Given a desired tolerance $tol$, choose $\frac{1}{l} < \tau_0$, $j_{max} \simeq tol^{-2}$, and $h \simeq tol^{\frac{1}{r+1}}$
initialization:
$u = 0$;
for $j = 1, \ldots, j_{max}$ do
    sample one realization $a_j = a(\cdot, \omega_j)$ of the random field
    solve primal problem $\to y(a_j, u)$ using FE on mesh $h$
    solve dual problem $\to p(a_j, u)$ using FE on mesh $h$
    $\widehat{\nabla J} = \alpha u + p(a_j, u)$
    $u = u - \tau_j \widehat{\nabla J}$
end
```

tol^2, the total required computational work is bounded by

$$W \lesssim tol^{-2-\frac{d\gamma}{r+1}}.$$

Here, we recall that the primal and dual problems can be solved, using a triangulation with mesh size h, in computational time $C_h = O(h^{-d\gamma})$, and r is the degree of the continuous FE that we use.

Proof. To achieve a tolerance $O(tol^2)$ for the error $\mathbb{E}[\|u_j^h - u_\star\|^2]$, we can equidistribute the precision tol^2 over the two terms in (III.35). This leads to the choice:

$$j_{max} \simeq tol^{-2}, \quad h \simeq tol^{\frac{1}{r+1}}.$$

The cost for solving one deterministic PDE with the FE method is proportional to $h^{-d\gamma}$. Hence the total cost for computing a solution u_j^h that achieves the required tolerance is

$$W \simeq 2\overline{N} j h^{-d\gamma} = O(tol^{-2-\frac{d\gamma}{r+1}}),$$

as claimed. □

Remark 9. *Other choices of (τ_j, N_j) have been investigated. For example we have studied the SG with step-size $\tau_j = \tau_0/j$, $\tau_0 l - 1 > 0$ and increasing the MC sample size $N_j \sim j^{\tau_0 l - 1}$. With this choice the estimate in* (III.35) *becomes*

$$a_j \leq D_4 j^{-\tau_0 l} \log(j)\ , \tag{III.39}$$

which leads to the choice $j_{max} \simeq tol^{-\frac{2}{\tau_0 l}} |\log(tol)|^{\frac{1}{\tau_0 l}}$ and a final complexity

$$W \simeq 2\sum_{i=1}^{j} i^{\tau_0 l - 1} h^{-d\gamma} \simeq 2 j^{\tau_0 l} h^{-d\gamma} = O(tol^{-2-\frac{d\gamma}{r+1}} |\log(tol)|).$$

Fixed MC gradient	SG - Variable step-size	SG - Variable step-size and N_j
$\tau_j = \tau_0$	$\tau_j = \tau_0/j$	$\tau_j = \tau_0/j$
$N \simeq tol^{-2}$	$N_j = \overline{N}$	$N_j = j^{\tau_0 l - 1}$
$h \simeq tol^{\frac{1}{r+1}}$	$h \simeq tol^{\frac{1}{r+1}}$	$h \simeq tol^{\frac{1}{r+1}}$
$j_{max} \simeq -\log(tol)$	$j_{max} \simeq tol^{-2}$	$j_{max} \simeq tol^{-\frac{2}{\tau_0 l}} \lvert \log(tol) \rvert^{\frac{1}{\tau_0 l}}$
$W \lesssim tol^{-2-\frac{d\gamma}{r+1}} \lvert \log(tol) \rvert$	$W \lesssim tol^{-2-\frac{d\gamma}{r+1}}$	$W \lesssim tol^{-2-\frac{d\gamma}{r+1}} \lvert \log(tol) \rvert$

Table III.1 – Complexity analysis overview for different optimization methods

The proof of the bound (III.39) *is detailed in Appendix III.K for completeness.*

Remark 10. *Since the constant l may be challenging to estimate in practice, it is often difficult to fulfill the condition $\tau_0 > 1/l$. To bypass this difficulty, one could consider the* Averaged Stochastic Gradient method *[SRB13] instead, in which the step size $\tau_j = \tau_0/j^\eta$, $\eta \in (0,1)$ is chosen, with $N_j = \overline{N}$ and the averaged control $\frac{1}{j}\sum_{i=1}^{j} u_i$ is considered. The analysis of this alternative method is postponed to a future work.*

Table III.1 summarizes the results obtained in both the fixed sample size and increasing sample size regimes. There, the total work (W) to achieve a given tolerance (tol) is presented. We see from the table that the two considered SG versions improve the complexity only by a logarithmic factor compared to the fixed gradient algorithm. The advantage we see in the SG version w.r.t. the fixed gradient, is that we do not have to fix in advance the sample size N and we can just monitor the convergence of the SG iteration until a prescribed tolerance is reached. However, in Algorithm 2, we do have to choose in advance the FE mesh size. It is therefore natural to look at a further variation of the SG algorithm in which the FE mesh is refined during the iterations until a prescribed tolerance is reached. This is detailed in the next Section.

III.G. Stochastic Gradient with variable mesh size

In this section, we refine the mesh used for our FE approximation, while running the optimization routine. The new mesh size h_j is now depending on the iteration j. Here we study only sequences of nested meshes of size $h_j = 2^{-\ell(j)}$ with $\ell : \mathbb{N} \to \mathbb{N}$ being an increasing function. The optimization procedure then reads:

$$u_{j+1}^{h_{j+1}} = u_j^{h_j} - \tau_j E_{MC}^{\overrightarrow{\omega_j}}[\nabla f^{h_j}(u_j^{h_j}, \omega)], \tag{III.40}$$

with $\overrightarrow{\omega_j} := (\omega_j^{(1)}, \cdots, \omega_j^{(N_j)})$. Notice that if non-nested meshes are used, a projection operator should be added in (III.40) to transfer information from one mesh to another. We first derive an error recurrence formula in the spirit of (III.33) for the particular recurrence (III.40) with a decreasing mesh-size h_j.

Theorem 15. *Denoting by* $u_{j+1}^{h_{j+1}}$ *the approximated control obtained using the recursive definition* (III.40)*, and* $u_\star$ *the exact control for the continuous optimal problem* (III.10)*, we have:*

$$\begin{aligned}\mathbb{E}[\|u_{j+1}^{h_{j+1}} - u^*\|^2] \\ &\leq c_j\mathbb{E}[\|u_j^{h_j} - u^*\|^2] + \frac{4\tau_j^2}{N_j}\mathbb{E}[\|\nabla f(u_\star,\omega)\|^2] + 4\tau_j\big(\tau_j(1+\frac{2}{N_j}) + \frac{1}{l}\big)Ch_j^{2r+2},\end{aligned} \tag{III.41}$$

with $c_j = 1 - \frac{\tau_j l}{2} + \tau_j^2 L^2\big(2 + \frac{2}{N_j}\big)$.

Proof. Subtracting the optimal continuous control $u_\star$ from both sides of the recurrence formula (III.40), we get

$$\begin{aligned}u_{j+1}^{h_{j+1}} - u^* =& u_j^{h_j} - u^* - \tau_j E_{MC}^{\overrightarrow{\omega_j}}[\nabla f^{h_j}(u_j^{h_j},\omega)] \pm \tau_j\mathbb{E}[\nabla f^{h_j}(u^*)] \pm \tau_j G[\nabla f^{h_j}(u_j^{h_j})] + \tau_j\mathbb{E}[\nabla f(u^*)]\\ =& u_j^{h_j} - u^* + \tau_j\left(\mathbb{E}[\nabla f^{h_j}(u^*)] - G[\nabla f^{h_j}(u_j^{h_j})]\right)\\ &+ \tau_j\left(G[\nabla f^{h_j}(u_j^{h_j})] - E_{MC}^{\overrightarrow{\omega_j}}[\nabla f^{h_j}(u_j^{h_j},\omega)]\right) + \tau_j\left(\mathbb{E}[\nabla f(u^*) - \nabla f^{h_j}(u^*)]\right).\end{aligned}$$

Then setting as in proof of Theorem 13:

$$\begin{aligned}&T_1 := G[\nabla f^{h_j}(u_j^{h_j})] - \mathbb{E}[\nabla f^{h_j}(u^*)],\\ &T_2 := G[\nabla f^{h_j}(u_j^{h_j})] - E_{MC}^{\overrightarrow{\omega_j}}[\nabla f^{h_j}(u_j^{h_j},\omega)],\\ &T_3 := \mathbb{E}[\nabla f(u_\star) - \nabla f^{h_j}(u^*)],\end{aligned}$$

we can rewrite the last equality as:

$$u_{j+1}^{h_{j+1}} - u^* = u_j^{h_j} - u^* - \tau_j T_1 + \tau_j T_2 + \tau_j T_3.$$

We compute the mean of the squared norm of $u_{j+1}^{h_{j+1}} - u^*$ as

$$\begin{aligned}\mathbb{E}[\|u_{j+1}^{h_{j+1}} - u^*\|^2] =& \mathbb{E}[\|u_j^{h_j} - u^*\|^2] + \tau_j^2\mathbb{E}[\|T_1\|^2] + \tau_j^2\mathbb{E}[\|T_2\|^2] + \tau_j^2\mathbb{E}[\|T_3\|^2]\\ &- 2\tau_j\mathbb{E}[\langle u_j^{h_j} - u^*, T_1\rangle] + 2\tau_j\mathbb{E}[\langle u_j^{h_j} - u^*, T_2\rangle] + 2\tau_j\mathbb{E}[\langle u_j^{h_j} - u^*, T_3\rangle]\\ &- 2\tau_j^2\mathbb{E}[\langle T_1, T_2\rangle] + 2\tau_j^2\mathbb{E}[\langle T_2, T_3\rangle] - 2\tau_j^2\mathbb{E}[\langle T_1, T_3\rangle].\end{aligned} \tag{III.42}$$

Next, we will bound each of these ten terms to find a recursive formula on $\mathbb{E}[\|u_j^{h_j} - u^*\|^2]$. First, the term $\tau_j^2\mathbb{E}[\|T_1\|^2]$ can be bounded as in the proof of Theorem 13 leading to:

$$\tau_j^2\mathbb{E}[\|T_1\|^2] \leq \tau_j^2 L_{h_j}^2\mathbb{E}[\|u_j^{h_j} - u^*\|^2],$$

with L_{h_j} being the Lipschitz constant for the function f^{h_j}, which is bounded by L (see Lemma 6). For the term $\tau_j^2\mathbb{E}[\|T_3\|^2]$, we find,

$$\begin{aligned}
\tau_j^2\mathbb{E}[\|T_3\|^2] &= \tau_j^2\|\mathbb{E}[\nabla f(u_\star) - \nabla f^{h_j}(u^*)]\|^2\\
&= \tau_j^2\|\mathbb{E}[p(u_\star) - p^{h_j}(u_\star)]\|^2\\
&\leq \tau_j^2\mathbb{E}[\|p(u_\star) - p^{h_j}(u_\star)\|^2]\\
&\leq 2\tau_j^2\mathbb{E}[\|p(u_\star) - \widetilde{p}^{h_j}(u_\star)\|^2] + 2\tau_j^2\mathbb{E}[\|\widetilde{p}^{h_j}(u_\star) - p^{h_j}(u_\star)\|^2]\\
&\leq 2C\tau_j^2\mathbb{E}[|p(u_\star)|^2_{H^{r+1}}]h^{2r+2} + 2C\tau_j^2\mathbb{E}[|y(u_\star)|^2_{H^{r+1}}]h^{2r+2} \quad \text{[using Céa's Lemma]}\\
&\leq 2\tau_j^2 C(y(u_\star), p(u_\star))h^{2r+2}.
\end{aligned}$$

Next, for $\tau_j^2\mathbb{E}[\|T_2\|^2]$ we use the same steps as in Theorem 13 to find

$$\tau_j^2\mathbb{E}[\|T_2\|^2] \leq \frac{2\tau_j^2 L_{h_j}^2}{N_j}\mathbb{E}\Big[\|u_j^{h_j} - u_\star\|^2\Big] + \frac{2\tau_j^2}{N_j}\mathbb{E}\Big[\|\nabla f^{h_j}(u_\star,\omega)\|^2\Big].$$

Then we bound the second term of the right hand side uniformly w.r.t. h_j by

$$\begin{aligned}
\|\nabla f^{h_j}(u_\star,\omega)\|^2 &\leq 2\|\nabla f^{h_j}(u_\star,\omega) - \nabla f(u_\star,\omega)\|^2 + 2\|\nabla f(u_\star,\omega)\|^2\\
&\leq 4C(y(u_\star), p(u_\star))h^{2r+2} + 2\|\nabla f(u_\star,\omega)\|^2,
\end{aligned}$$

where we have used the same steps as for T_3 to bound $\|\nabla f_{h_j}(u_\star,\omega) - \nabla f(u_\star,\omega)\|$. Finally, for the cross terms we have

$$\begin{aligned}
2\tau_j\mathbb{E}[\langle u_j^{h_j} - u^*, T_1\rangle] &= 2\tau_j\mathbb{E}[\langle u_j^{h_j} - u^*, G[\nabla f^{h_j}(u_j^{h_j}) - \nabla f^{h_j}(u^*)]\rangle]\\
&= 2\tau_j\mathbb{E}[G[\langle u_j^{h_j} - u^*, \nabla f^{h_j}(u_j^{h_j}) - \nabla f^{h_j}(u^*)\rangle]] \quad \text{[using Strong convexity]}\\
&\geq \tau_j l\mathbb{E}[\|u_j^{h_j} - u^*\|^2],
\end{aligned}$$

and as in Theorem 9,

$$2\tau_j\mathbb{E}[\langle u_j^{h_j} - u_\star, T_2\rangle] = 2\tau_j^2\mathbb{E}[\langle T_1, T_2\rangle] = 2\tau_j^2\mathbb{E}[\langle T_2, T_3\rangle] = 0.$$

Moreover

$$\begin{aligned}
2\tau_j\mathbb{E}[\langle u_j^{h_j} - u_\star, T_3\rangle] &\leq 2\tau_j\frac{l}{4}\mathbb{E}[\|u_j^{h_j} - u_\star\|^2] + \frac{2\tau_j}{l}\mathbb{E}[\|T_3\|^2]\\
&\leq 2\tau_j\frac{l}{4}\mathbb{E}[\|u_j^{h_j} - u_\star\|^2] + \frac{4\tau_j}{l}C(y(u_\star), p(u_\star))h^{2r+2},
\end{aligned}$$

and finally

$$\begin{aligned}
2\tau_j^2\mathbb{E}[\langle T_1, T_3\rangle] &\leq \tau_j^2\mathbb{E}[\|T_1\|^2] + \tau_j^2\mathbb{E}[\|T_3\|^2]\\
&\leq \tau_j^2 L_{h_j}^2\mathbb{E}[\|u_j^{h_j} - u^*\|^2] + 2\tau_j^2 C(y(u_\star), p(u_\star))h^{2r+2}.
\end{aligned}$$

```
Algorithm 3: Stochastic Gradient with variable mesh size algorithm
Data:
Given a desired tolerance $tol$, choose $\frac{1}{l} < \tau_0$, $h_0$ and $j_{max} \simeq tol^{-2}$ initialization:
$u = 0$
for $j = 1, \dots, j_{max}$ do
    update mesh size to $h = h_0 2^{-\lceil \frac{\ln_2 j - \ln_2 \tau_0 l}{2r+2} \rceil}$
    sample one realization $a_j = a(\cdot, \omega_j)$ or the random field
    solve primal problem $\rightarrow y(a_j, u)$ on mesh $h$
    solve dual problem $\rightarrow p(a_j, u)$ on mesh $h$
    $\widehat{\nabla J} = \alpha u + p(a_j, u)$
    $u = u - \tau_j \widehat{\nabla J}$
end
```

Putting everything together, we finally obtain (III.41), as claimed. □

A natural choice to tune the parameters τ_j, N_j and h_j would be to set, guided by the usual Robbins-Monro theory, $\tau_j = \tau_0/j$, $N_j = \overline{N}$ and balancing all terms on right hand side of (III.41).

Theorem 16. *Suppose that the assumptions of Corollary 1 hold and let $u_j^{h_j}$ denote the j-th iterate of* (III.40). *For the particular choice $(\tau_j, N_j, h_j) = (\tau_0/j, \overline{N}, h_0 2^{-\ell(j)})$, with $\ell(j) = \lceil \frac{\ln_2(j) - \ln_2(\tau_0 l)}{2r+2} \rceil$, and assuming $\tau_0 > 1/l$, we have:*

$$\mathbb{E}[\|u_j^{h_j} - u^*\|^2] \leq F_1 j^{-1} \tag{III.43}$$

for a suitable constant F_1 independent of j.

Proof. With the choice of τ_j, N_j and $\ell(j)$ in the statement of the theorem, the two last terms $\frac{4\tau_j^2}{N_j}\mathbb{E}[\|\nabla f_{h_j}(u_\star, \omega)\|^2]$ and $4\tau_j\big(\tau_j(1 + \frac{2}{N_j}) + \frac{1}{l}\big)Ch_j^{2r+2}$ in the inequality (III.41) have the same order $O(j^{-2})$. Then, we apply the same reasoning as in Theorem 14 to conclude the proof. □

Now we present the idea of the SG algorithm 3 with variable mesh size.

Concerning the complexity of Algorithm 3, one can derive the following complexity result.

Corollary 4. *In order to achieve a given tolerance $O(tol)$, i.e. to guarantee that $\mathbb{E}[\|u_j^{h_j} - u_\star\|^2] \lesssim tol^2$, the total required computational work W is bounded by:*

$$W \lesssim tol^{-2 - \frac{d\gamma}{r+1}}$$

Proof. To achieve $tol^2 \lesssim j_{max}^{-1}$ requires $j_{max} \simeq tol^{-2}$. Then the total work required is bounded by

$$W = \sum_{p=1}^{j_{max}} 2\overline{N} h_p^{-d\gamma} = 2\overline{N} \sum_{p=1}^{j_{max}} 2^{d\gamma \lceil \frac{\ln_2 p - \ln_2 \tau_0 l}{2r+2} \rceil}$$

But as $\lceil \frac{\ln_2 p - \ln_2 \tau_0 l}{2r+2} \rceil \leq \frac{\ln_2 p - \ln_2 \tau_0 l}{2r+2} + 1$, one can bound:

$$\begin{aligned} W &\leq 2\overline{N} \sum_{p=1}^{j_{max}} 2^{d\gamma \left(\frac{\ln_2 p - \ln_2 \tau_0 l}{2r+2} + 1 \right)} \leq 2^{n\gamma+1} \overline{N} \{\tau_0 l\}^{\frac{-d\gamma}{2r+2}} \sum_{p=1}^{j_{max}} p^{\frac{d\gamma}{2r+2}} \\ &\leq 2^{n\gamma+1} \overline{N} \{\tau_0 l\}^{\frac{-d\gamma}{2r+2}} \frac{2r+2}{2r+2+d\gamma} (j_{max}+1)^{\frac{d\gamma}{2r+2}+1} \end{aligned}$$

But as $j_{max} \simeq tol^{-2}$, we finally bound the computational work by

$$W \lesssim tol^{-2-\frac{d\gamma}{r+1}}.$$

□

We notice that the asymptotic complexity remains the same as in the Stochastic Gradient algorithm with fixed mesh size. However, as we only use computations on coarse meshes for the first iterations, we thus expect an improvement due to reducing the constant. We will compute this constant, based on numerical examples, in the Section III.H.

III.H. Numerical results

In this section we verify the assertions of Theorems 12, 15, and 16, as well as the computational complexity derived in the corresponding Corollaries. Specifically, we illustrate the order of convergence for the three versions of the steepest descent algorithm presented in Sections III.E, III.F, and III.G respectively. For this purpose, we consider the optimal control problem (III.19) with a MC approximation of the expectation. We consider problem (III.2) in the domain $D = (0,1)^2$ with $g = 1$ and the random diffusion coefficient

$$a(x_1, x_2, \boldsymbol{\xi}) = 1 + 0.1 \left(\xi_1 \cos(\pi x_2) + \xi_2 \cos(\pi x_1) + \xi_3 \sin(2\pi x_2) + \xi_4 \sin(2\pi x_1) \right), \quad \text{(III.44)}$$

with $(x_1, x_2) \in D$ and $\boldsymbol{\xi} = (\xi_1, \dots, \xi_4)$ with $\xi_i \overset{iid}{\sim} \mathcal{U}([-1,1])$. Figure III.2 shows four typical realizations of the random field. The target function z_d has been chosen as $z_d(x,y) = \sin(2\pi x)\sin(2\pi y)$ (see Fig. III.1b) and we have taken $\alpha = 0.1$ in the objective function $J(u)$ in (III.3). For the FE approximation, we have considered a structured

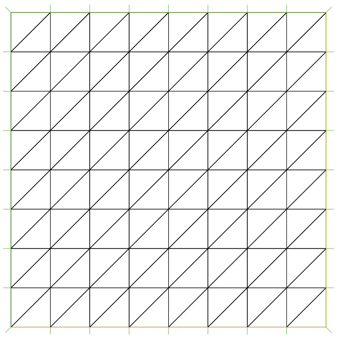


Figure III.1a – Structured mesh triangulation with $h = 2^{-3}$

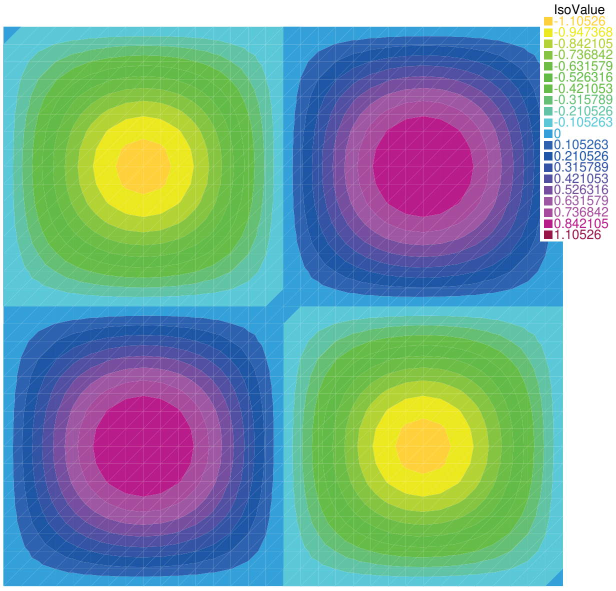
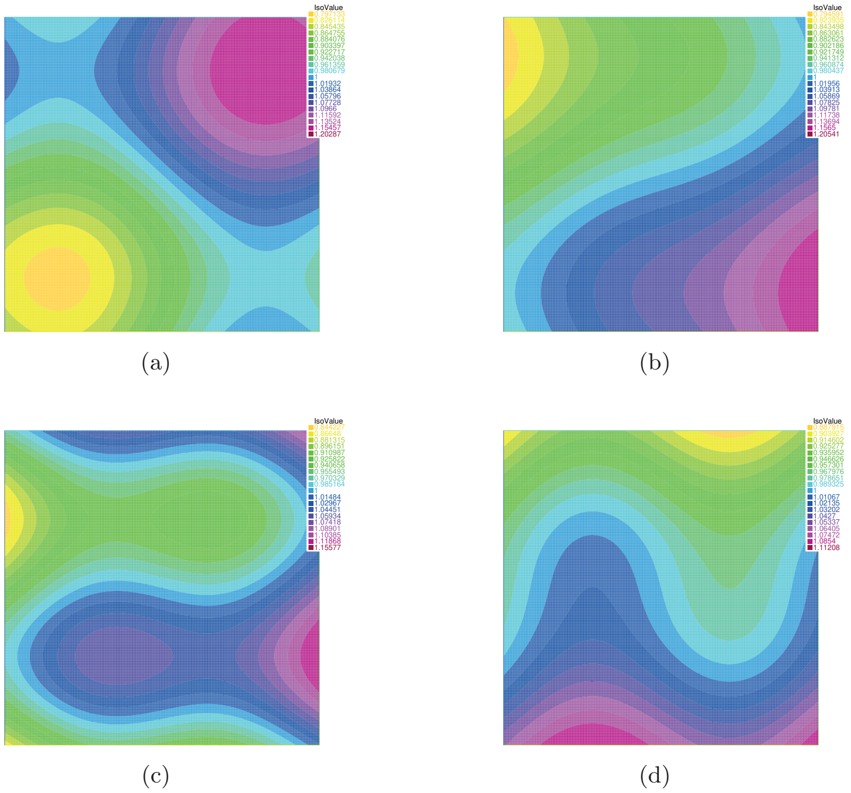


Figure III.1b – Target function z_d for the optimal control problem

triangular grid of size h (see Fig. III.1a) where each side of the domain D is divided into $1/h$ sub-intervals and used piece-wise linear FE (i.e. $r = 1$). All calculations have been performed using the FE library Freefem++[Hec12].

III.H.1. Reference solution

To compute a reference solution of problem (III.2), we use a full tensorized Gaussian Legendre (GL) quadrature grid with 5 points in each direction and a fine triangulation with $h = 2^{-8}$ (see, e.g., references [SSS11, BSSvW10] and Appendix III.J.2 for a formal error estimate). As this approximated problem is now deterministic with fixed Gaussian nodes, we used a stopping condition on the gradient. In Figure III.3 we show the optimal control obtained after $j = 6$ iterations when the stopping criterion $\|E^{GL}_{(5,5,5,5)}[\nabla J(u^h_j)]\| \leq 10^{-8}$



(a)

(b)

(c)

(d)

Figure III.2 – Four realizations of the diffusion random field (III.44). Values of parameters ξ_i, $i \in \{1, 2, 3, 4\}$ are reported in Table III.2.

was met, where u_j^h is the j-th iterate of (III.19) and $\widehat{E}$ in (III.19) is a full tensorized Gaussian Legendre (GL) quadrature approximation of the expectation. The steepest descent step size was chosen as $\tau_0 = 10$. The L^2-norm of the final control using this Gaussian quadrature is $\|\widehat{u}_{j=6}^{h=2^{-8}}\| = 0.0663345$.

III.H.2. Steepest descent algorithm with fixed discretization

We investigate here the convergence of the method defined in (III.24), for which we recall the error bound (III.30) in the case of piece-wise linear FE (i.e. $r = 1$):

$$\mathbb{E}[\|\widehat{u}_j^h - u\|^2] \leq C_1 e^{-\rho j} + \frac{C_2}{N} + C_3 h^4 \, .$$

For each tolerance tol, using formula (III.30), we compute the optimal mesh size $h = f_1(tol)$, the optimal sample size in the MC approximation $N = f_2(tol)$, and, finally, the minimum number of iterations we need in the steepest descend method, $j_{max} = f_3(tol)$. Here, the three functions f_i are introduced to emphasize that these parameters are completely determined by the prescribed tolerance goal tol. In what follows, we compare

realization	(a)	(b)	(c)	(d)
ξ_1	-0.717883	0.804387	-0.502862	0.799162
ξ_2	-0.666988	-0.966216	-0.777745	-0.174098
ξ_3	-0.383946	0.210821	0.969271	-0.337877
ξ_4	-0.35036	0.300546	0.562564	0.331761

Table III.2 – Realization values of the diffusion random field of Fig. III.2.

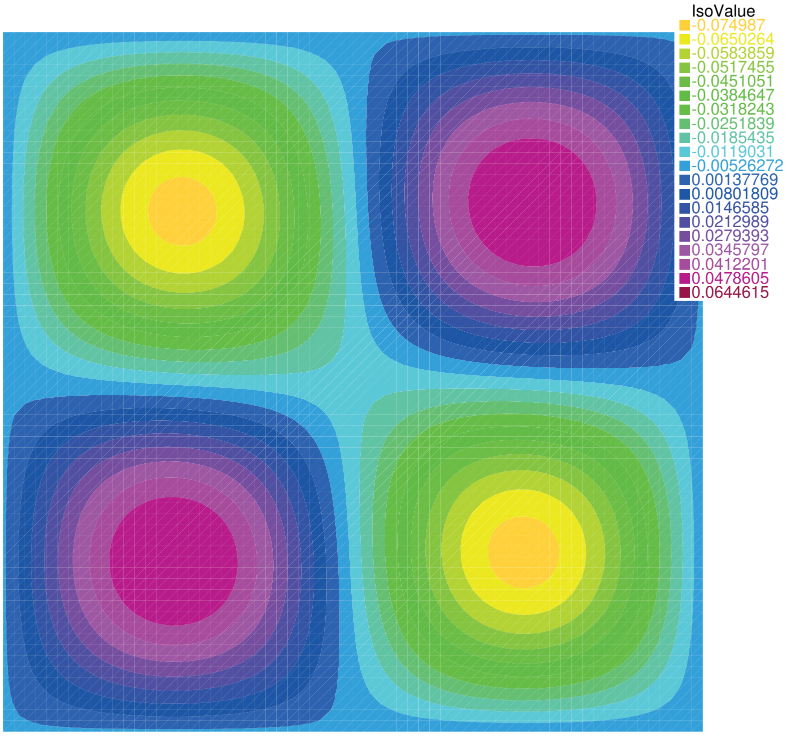


Figure III.3 – Optimal control reference solution computed with $h = 2^{-8}$ on tensorized Gauss-Legendre quadrature formula with $N = 9^4$ nodes.

the actual error on the optimal control obtained from the algorithm (measured w.r.t. the reference solution) with the prescribed tolerance.

To have a precise idea of the functions f_i, we have estimated the constants in (III.30) numerically.

- In order to estimate C_1 we used the same finest mesh as the one used to compute our reference solution with $h = 2^{-8}$, and we used also the same Gaussian 5 points for the quadrature. We computed numerically the squared error between the optimal control after i iterations and the reference solution computed above. We then only see the first term in (III.30), and running the algorithm for the first 10 iterations of the steepest descent method, we estimated a constant $C_1 \approx 10^{-3}$ and $\rho \approx 3.2$.

- To estimate the second constant C_2, we used again the same finest mesh as the one used to compute our reference solution with $h = 2^{-8}$. We ran the steepest descent method up to 10 iterations, using a MC estimator for the mean of the gradient with a sample size N_{MC} of $N_{MC} = 2^0, 2^1, \cdots, 2^5$. Finally, for every sample size N_{MC} of the MC estimator, we averaged the final error squared on the control over 10 independent realizations. As we go up to 10 iterations, the error term is of order $C_1 e^{-3.2\times 10} = 1.27 \times 10^{-17}$. That is, as long as the term C_2/N stays bigger than 10^{-15}, i.e. $C_2 > 10^{-14}$, we effectively only see the C_2/N term. We numerically found $C_2 \approx 3.16 \times 10^{-5}$, what is coherent with the last condition.
- Finally, to compute the third term, we used different mesh sizes $h = 2^{-1}, \cdots, 2^{-5}$, and we used a steepest descent algorithm with sufficiently many iterations with a Gaussian quadrature with 5 points in each direction. We found $C_3 \approx 5.01 \times 10^{-1}$.

Figure III.4 shows the convergence of the error on the control (in the L^2-norm), versus the discretization parameter h (that is directly linked to N and j_{max} using the functions f_i, $i = 1, 2, 3$). The bars denote plus one standard deviation, estimated by repeating the simulation 20 times.

We observe a convergence rate of h^{-4} on the squared error, which is consistent with the theoretical result (III.30). Figure III.5 shows the corresponding computational complexity. Here we have used the theoretical computational cost $W = 2N j_{max} h^{-2}$ (which assumes an optimal linear algebra solver with $\gamma = 1$).

The observed slope is consistent with our theoretical result $W \sim tol^{-3}$ up to logarithmic terms.

III.H.3. Stochastic Gradient with fixed mesh size h

We implemented here the Stochastic Gradient method described in Section III.F using $\overline{N} = 1$ sample at each iteration (recall that the complexity does not depend on $\overline{N}$). As the error result (III.35) is in the mean squared sense, we ran the simulation 10 times and averaged the obtained errors, in order to estimate this mean.

Also for the SG method with a fixed mesh size we have estimated the constants in (III.35).

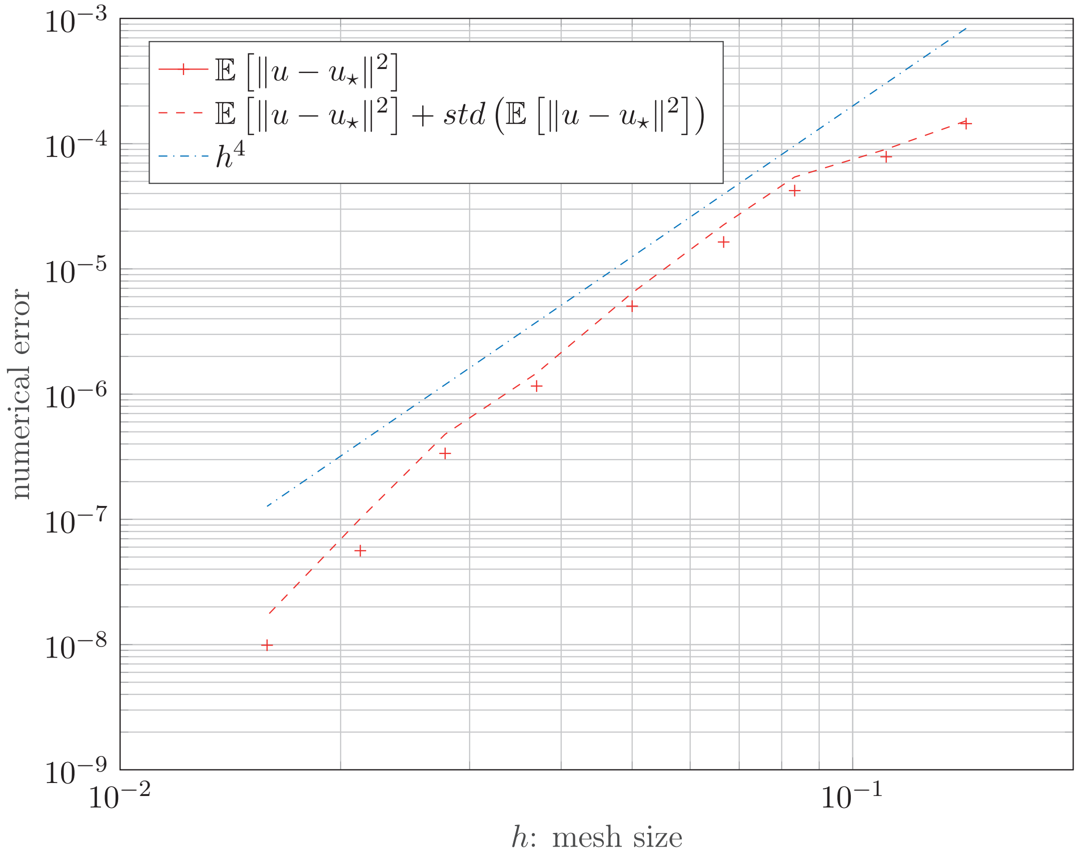


Figure III.4 – Steepest descent Algorithm 1 with fixed discretization over iterations. Error $\mathbb{E}[\|u-u_\star\|^2]$ as a function of the mesh size h. (—+—) estimated mean over 20 repetitions. (- - -) maen plus one estimated standard deviation.

- To numerically estimate the constant D_1, we simply used the finest mesh of size $h = 2^{-8}$ and plotted the squared error on the control versus the i-th iteration using a Stochastic Gradient technique. We repeated the procedure 10 times to compute a MC estimator of the expectation of this squared error. We found effectively a slope of -1 and the constant $D_1 \approx 2.51 \times 10^{-6}$.
- Again as for the fixed MC procedure, to estimate the second term constant D_2, we used different mesh sizes $h = 2^{-1}, \cdots, 2^{-5}$, and a Stochastic Gradient algorithm with sufficiently many iterations. We found $D_2 \approx 6.31 \times 10^{-1}$, which is very close to the C_3 constant, estimated earlier.

Figures III.6 presents the squared error on the control for different desired tolerances tol, i.e. different mesh sizes, using the SG steepest descent method with resampling. The theoretical rate is thus verified for $r = 1$ and $d = 2$.

Figure III.7 and III.8 show the estimated mean squared error, using Algorithm 2, as a

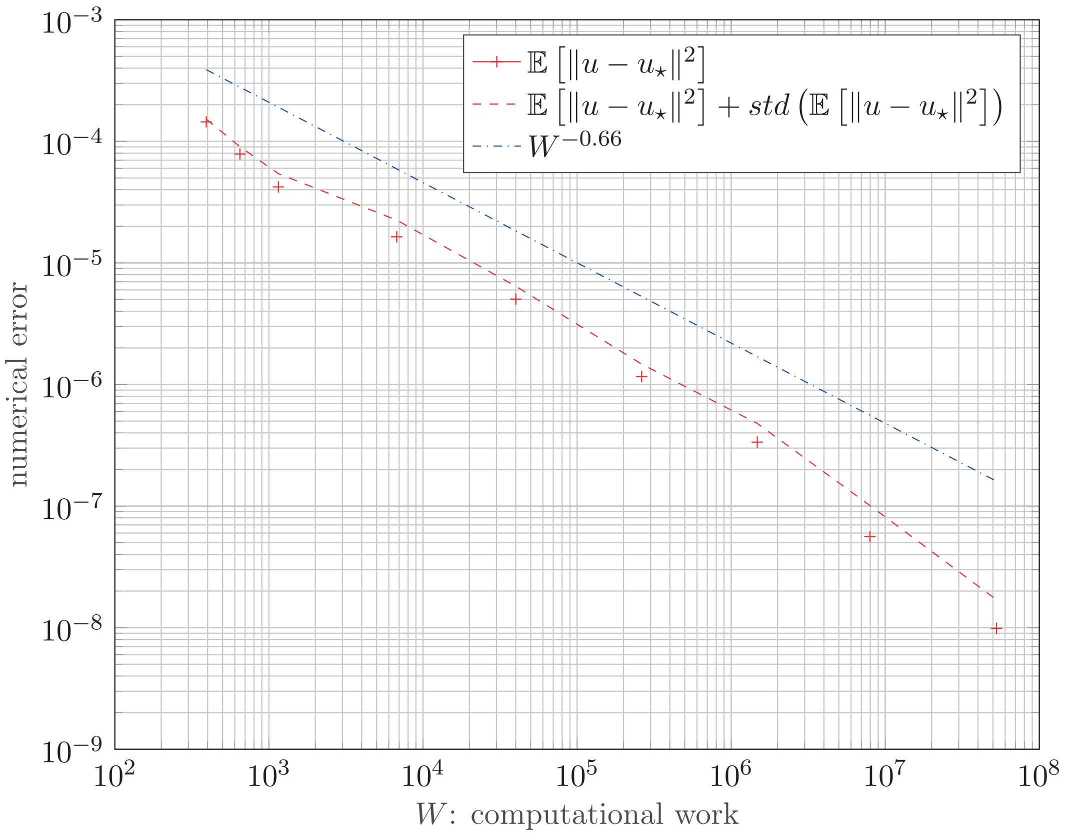


Figure III.5 – Steepest descent Algorithm 1 with fixed discretization over iterations. Error $\mathbb{E}[\|u-u_\star\|^2]$ as a function of the theoretical computational work W. (—+—) estimated mean over 20 repetitions (only 2 repetition in the last two points). (- - -) mean plus one estimated standard deviation.

function of the theoretical cost $W = 2j_{max}h^{-2}$. The slope is the one expected, namely $W \lesssim tol^{-3}$.

III.H.4. Stochastic Gradient with decreasing mesh size h_j

We illustrate here the Stochastic Gradient method described in Section III.G. As the error result (III.43) is in mean-squared sense, we ran the simulation 20 times up to iteration $j_{max} = 4000$. We then average every error at every iteration over these 20 simulation. In Figure III.9 we plot the averaged errors obtained versus the iteration of the SG recursion. In fact, the plot shows the mean squared errors and the mean squared errors plus one standard deviation, both obtained using once more all the 20 simulations. As we refine using only embedded mesh, we do see a refinement drop at iterations $j = 16, 256, 4096$. Notice that the next refinement would be at iteration $j = 65536$, which however is computationally prohibitive.

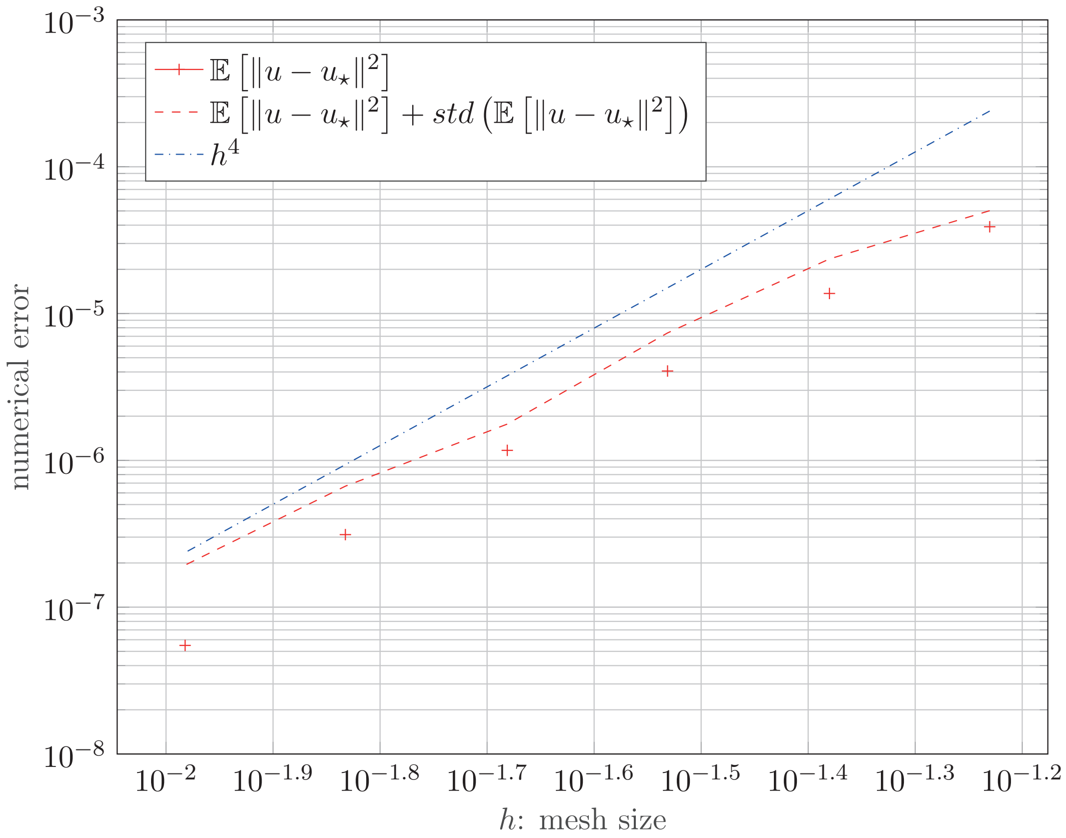


Figure III.6 – SG Algorithm 2 with fixed space discretization over iterations. Error $\mathbb{E}[\|u-u_\star\|^2]$ as a function of the mesh size h. (—+—) estimated mean over 10 repetitions (only 2 repetitions in the last two points). (- - -) mean plus one estimated standard deviation.

In practice, in order to estimate the parameter h_0, we set a desired final tolerance tol, which is directly related to h_{final} through the constants D_1 and D_2 estimated previously. Based on j_{max} linked to the tolerance tol and expression (III.43), we can thus determine the initial mesh size h_0. That is, with the initial mesh size h_0 fixed, we then run the algorithm with this h_0, ensuring that the algorithm will terminate at iteration j_{max} with final mesh size $h_{j_{max}}$.

In Figure III.10 we plot the averaged numerical error versus the computational cost W for the three algorithm studied in the previous Sections: the fixed MC gradient, the SG with fixed mesh, and the SG with variable mesh size. For the fixed MC gradient and the SG with fixed mesh, we ran 20 iid simulations for every tolerance (i.e. every point and every square in the Figure) and then averaged them to estimate the mean. For the SG with variable mesh size we show 3 different realization of error versus computational

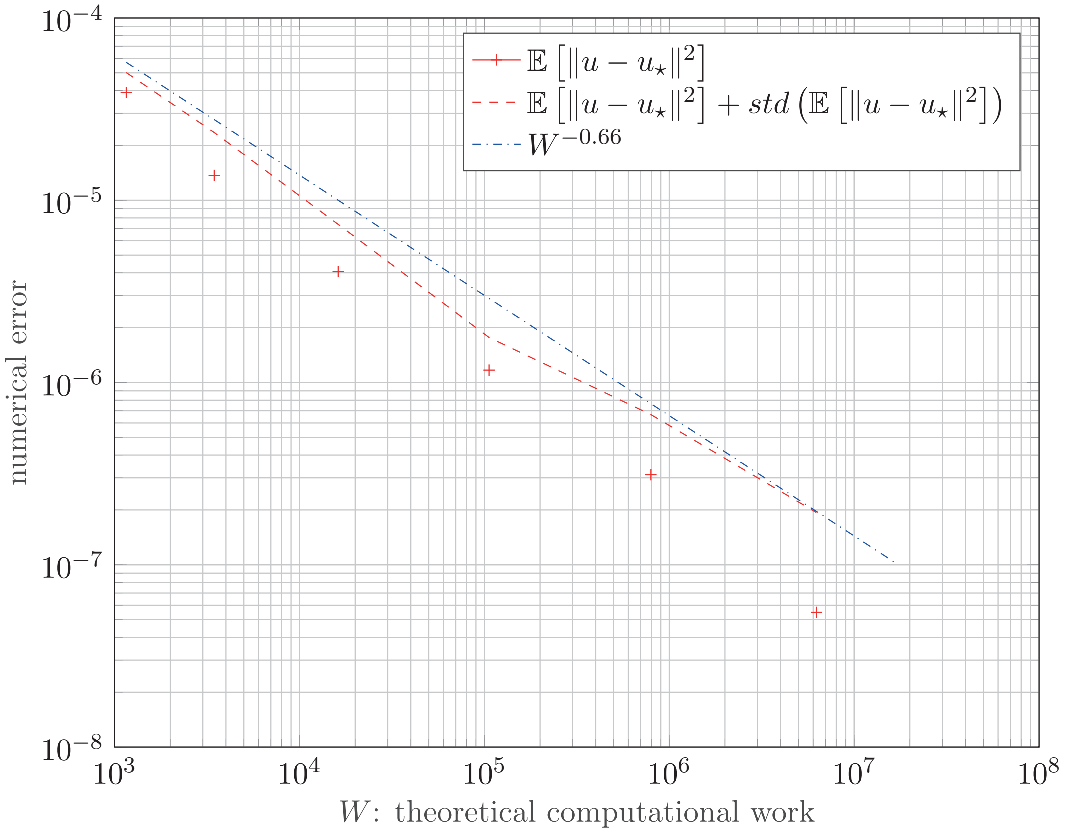


Figure III.7 – SG Algorithm 2 with fixed space discretization over iterations. Error $\mathbb{E}[\|u-u_\star\|^2]$ as a function of the theoretical computational work W. (—+—) estimated mean over 10 repetitions (only 2 repetitions in the last two points). (- - -) mean plus one estimated standard deviation.

work (with the same initial mesh size h_0). As discussed before, the SG is more efficient than the fixed MC gradient, but only by a logarithmic factor (which is difficult to observe in Figure III.10). All three algorithms follow a slope of $tol^2 \sim W^{-2/3}$, as predicted by our theoretical complexity analysis. The proportionality constant is smaller for the SG compared to the fixed MC gradient, and seems to further reduce for the variable mesh size SG version at least in the range of computational works considered. This is consistent with our intuition that computational work is saved in the earlier iterations in this version of the SG method.

III.I. Conclusions

In this work, we have analyzed and compared the complexity of three versions of the gradient method for the numerical solution of a mean-based risk-averse optimal control problem for an elliptic PDE with random coefficients, where a Finite Element discretization

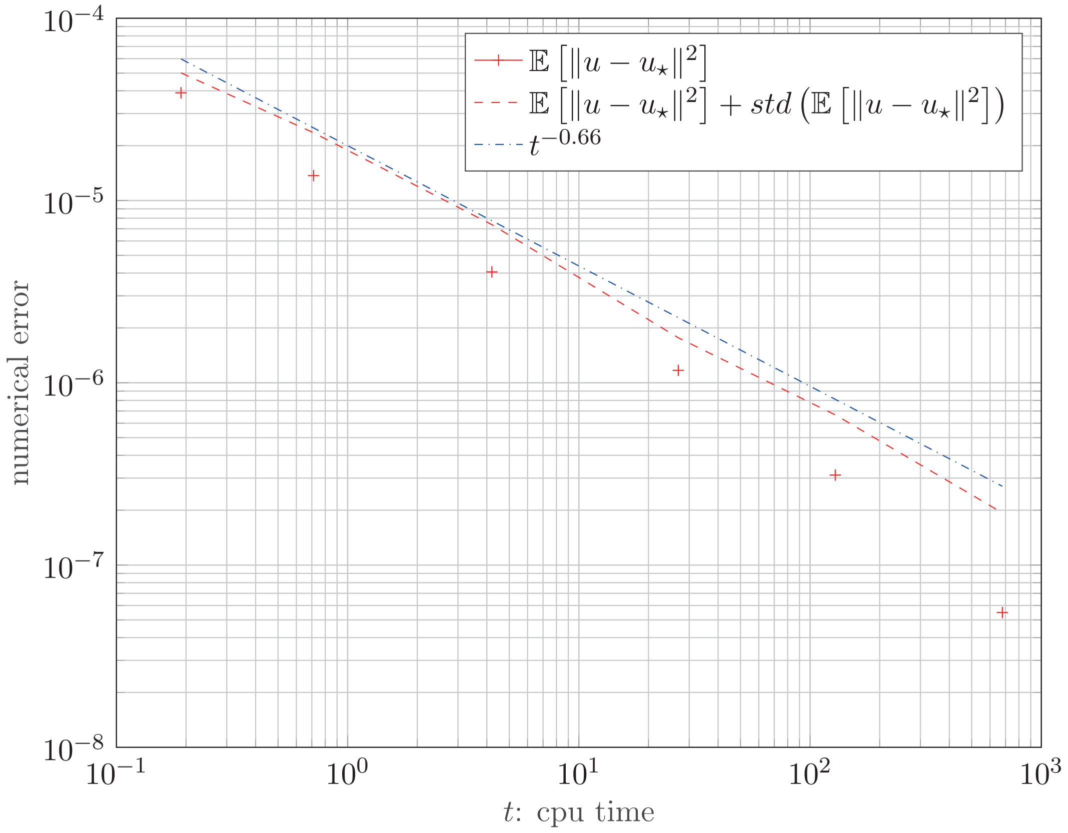


Figure III.8 – SG Algorithm 2 with fixed space discretization over iterations. The error $\mathbb{E}[\|u-u_\star\|^2]$ as a function of the average CPU time is plotted. (—+—) estimated mean over 10 repetitions (only 2 repetition in the last two points). (- - -) one estimated standard deviation.

is used to approximate the underlying PDEs and a Monte Carlo sampling is used to approximate the expectation in the risk measure. In the first version the FE mesh and Monte Carlo sample are chosen initially and kept fixed over the iterations. In the second version, a Stochastic Gradient method, the finite element discretization is still kept fixed over the iterations, however the expectation in the objective function is re-sampled independently at each iteration, with a small (fixed) sample size. Finally, the third version is again a stochastic gradient method, but now with successively refined FE meshes over the iterations. We have shown in particular, that the stochastic versions of the gradient method improve the computational complexity by log factors. Our complexity analysis is based on a priori error estimates and a priori choices of the FE mesh size, the Monte Carlo sample size, and the gradient iterations to obtain a prescribed tolerance.

Beside the improved complexity, another interest in looking at stochastic versions of the gradient method is that they are more amenable to adaptive versions, in which the mesh size and possibly the Monte Carlo sample size are refined over the iterations based on

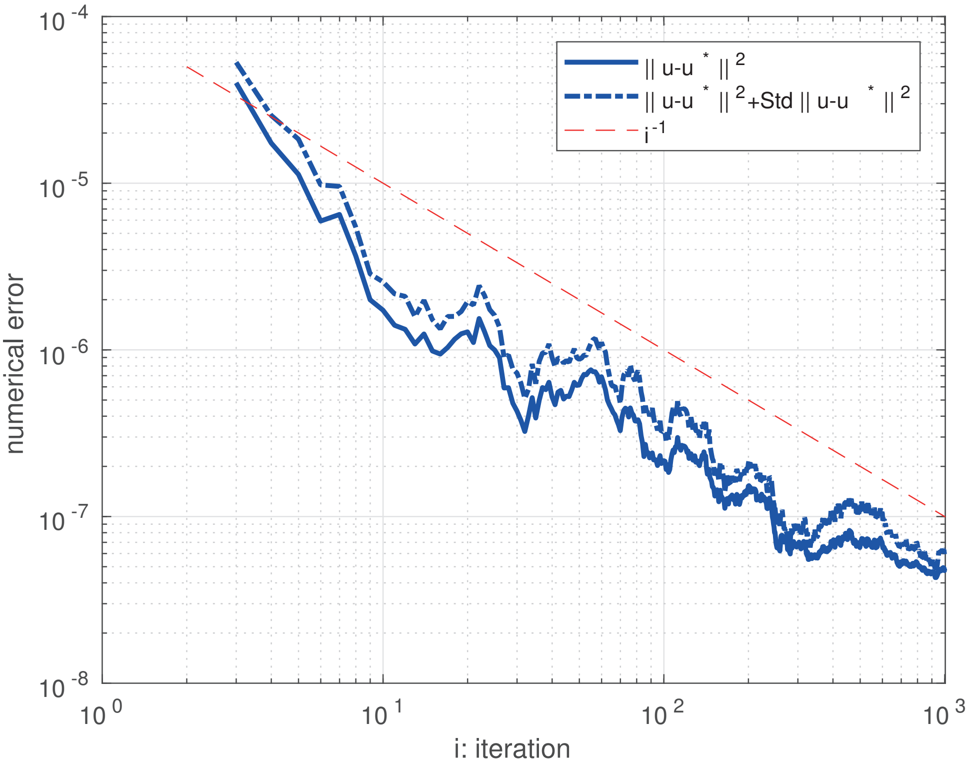


Figure III.9 – SG Algorithm 3 with variable mesh size over iterations. The error $\mathbb{E}[\|u - u_\star\|^2]$ as a function of iteration index is plotted.

suitable *a posteriori* error indicators. The study of such adaptive versions is postponed to future work.

Another interesting direction is the extension of stochastic gradient methods to more general risk measures. We mention that Stochastic Gradient methods have been already used in combination with the CVaR risk measure [BFP09], although not in the context of PDE-constrained optimal control problems.

III.J. Appendix: Reference solution by Stochastic Collocation

III.J.1. Optimal Control Problem with quadrature

In this appendix, we describe the computation of the reference solution used in the numerical result of Section III.H, by the Stochastic Collocation method on a tensor grid of Gauss Legendre points and provide an error estimate for such reference solution. In the setting of Section III.H, with only 4 random variables, we show here that the Stochastic Collocation approximation is exponentially convergent and a very accurate

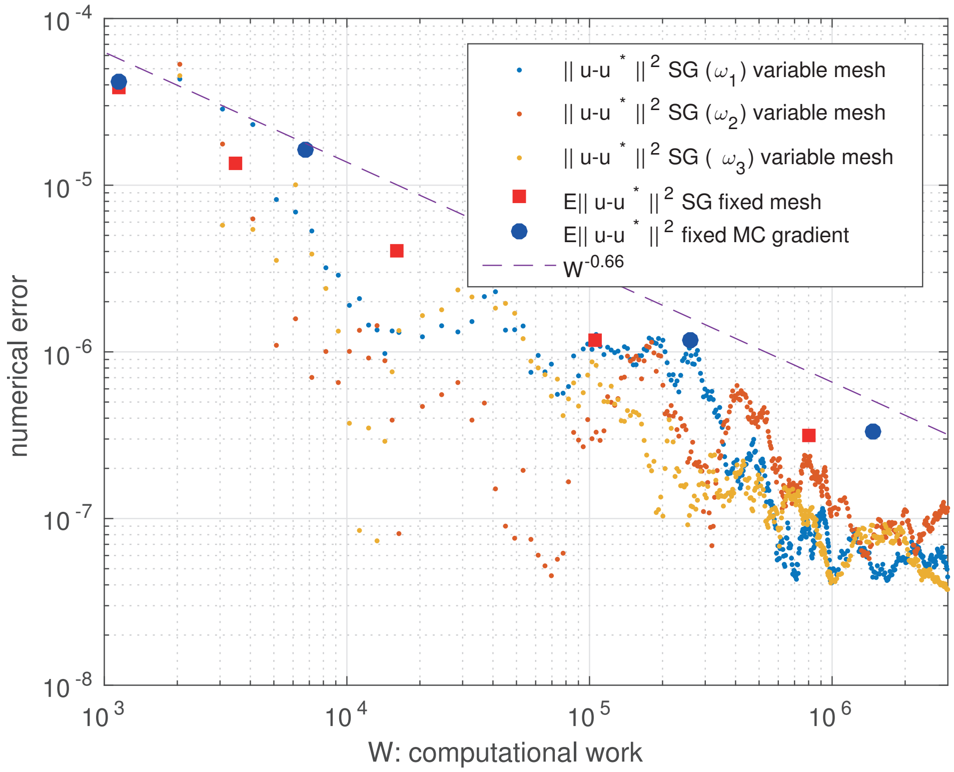


Figure III.10 – Comparison between Algorithm 1, 2, 3. The estimated mean squared error $\mathbb{E}[\|u - u_\star\|^2]$ is plotted as a function of the theoretical computation work W for fixed MC gradient and SG with fixed mesh, versus 3 different realizations of the SG with variable mesh size algorithm.

solution can be obtained with a moderate number of collocation points (5^4 were used in the numerical results). We suppose here that our expectation estimator is not random, but uses deterministic points ξ_i, for $i = 1, \dots, N$. The estimated optimal control $\widehat{u}$ is then deterministic as well. The following theorem derives an error bound when we estimate the exact expectation $\mathbb{E}$ in (III.3) by a deterministic quadrature formula $\widehat{E}$.

Theorem 17. *Denoting by $u_\star$ the optimal control solution of the exact problem* (III.10) *and by $\widehat{u}$ the solution of the semi-discrete collocation problem* (III.19)*, we have*

$$\frac{\alpha}{2}\|\widehat{u} - u_\star\|^2 + \mathbb{E}[\|y(u_\star) - y(\widehat{u})\|^2] \leq \frac{1}{2\alpha}\|\mathbb{E}[p(\widehat{u})] - \widehat{E}[p(\widehat{u})]\|^2. \tag{III.45}$$

Proof. The expressions of the gradient of J and $\widehat{J}$ are given by $\nabla J(u_\star) = \alpha u_\star + \mathbb{E}[p(u_\star)]$, $\nabla \widehat{J}(\widehat{u}) = \alpha\widehat{u} + \widehat{E}[p(\widehat{u})]$. From the optimality condition (III.5) for J, we derive the optimality condition for $\widehat{J}$ as:

$$\langle \nabla \widehat{J}(\widehat{u}), v' - \widehat{u}\rangle \geq 0 \quad \forall v' \in U. \tag{III.46}$$

Then choosing $v = \widehat{u}$ in (III.5) and $v' = u_\star$ in (III.46) and combining both, we have

$$\langle \alpha(u_\star - \widehat{u}) + \mathbb{E}[p(u_\star)] - \widehat{E}[p(\widehat{u})], \widehat{u} - u_\star \rangle \geq 0,$$

that is,

$$\alpha \|u_\star - \widehat{u}\|^2 \leq \langle \mathbb{E}[p(u_\star)] - \mathbb{E}[p(\widehat{u})] + \mathbb{E}[p(\widehat{u})] - \widehat{E}[p(\widehat{u})], \widehat{u} - u_\star \rangle. \tag{III.47}$$

In order to bound the first part of the error in (III.47), $\langle \mathbb{E}[p(u)] - \mathbb{E}[p(\widehat{u})], \widehat{u} - u \rangle$, we take one random realization ω and we use the primal-dual equations to obtain:

$$\begin{aligned} \langle \widehat{u} - u_\star, p_\omega(u_\star) - p_\omega(\widehat{u}) \rangle &= b_\omega(y_\omega(\widehat{u}) - y_\omega(u_\star), p_\omega(u) - p_\omega(\widehat{u})) \\ &= \langle y_\omega(u_\star) - y_\omega(\widehat{u}), y_\omega(\widehat{u}) - y_\omega(u_\star) \rangle \\ &= -\|y_\omega(u_\star) - y_\omega(\widehat{u})\|^2. \end{aligned}$$

Then taking the (exact) expectation over all the realizations ω, we find:

$$\langle \mathbb{E}[p(u_\star)] - \mathbb{E}[p(\widehat{u})], \widehat{u} - u_\star \rangle = -\mathbb{E}[\|y(u_\star) - y(\widehat{u})\|^2].$$

For the second contribution, $\langle \mathbb{E}[p(\widehat{u})] - \widehat{E}[p(\widehat{u})], \widehat{u} - u_\star \rangle$, we simply use Young's inequality, yielding

$$\langle \mathbb{E}[p(\widehat{u})] - \widehat{E}[p(\widehat{u})], \widehat{u} - u_\star \rangle \leq \frac{1}{2\alpha} \|\mathbb{E}[p(\widehat{u})] - \widehat{E}[p(\widehat{u})]\|^2 + \frac{\alpha}{2} \|\widehat{u} - u_\star\|^2,$$

from which the claim eventually follows. □

III.J.2. Collocation on tensor grid of Gaussian Legendre quadrature

The quantification of the quadrature error $\mathbb{E}[p(\widehat{u})] - \widehat{E}[p(\widehat{u})]$, i.e. the right hand side in (III.45), heavily depends on the smoothness of the dual function in the stochastic variables. The numerical example considered in Section III.H has a diffusion coefficient of the form

$$a(x, \xi) = a_0(x) + \sum_{i=1}^{M} \sqrt{\lambda_i} \xi_i b_i(x) \ ,$$

with $a_0 > 0$ a.e. in D, $\|b_i\|_{L^\infty(D)} = 1$, $\sum_{i=1}^M \sqrt{\lambda_i} < \operatorname{essinf}_{x \in D} a_0(x)$ and $\xi_i \sim \mathcal{U}([-1,1])$ iid uniform random variables. We denote by $\xi = (\xi_1, \cdots, \xi_M)$ the corresponding random vector. Hence, in this case the probability space $(\Gamma, \mathcal{F}, P)$ is $\Gamma = [-1,1]^M$, $\mathcal{F} = \mathcal{B}(\Gamma)$ the Borel σ-algebra on Γ, and $\mathbb{P}(d\xi) = \otimes_{i=1}^M \frac{d\xi_i}{2}$ the uniform product measure on Γ. In this case we chose as a quadrature formula the tensor Gaussian quadrature built on Gauss-Legendre quadrature points. In particular, we consider a tensor grid with q_i

points in the i-th variable and denote the corresponding quadrature by $E_q^{GL}[\cdot]$, where $q = (q_1, \cdots, q_M) \in \mathbb{N}^M$ is a multi-index.

To any vector of indexes $[k_1, \ldots, k_M] \in \prod_{i=1}^M \{1, \cdots, q_i\}$ we associate the global index

$$k = k_1 + q_1(k_2 - 1) + q_1 q_2(k_3 - 1) + \ldots,$$

and we denote by y_k the point $y_k = [y_{1,k_1}, y_{2,k_2}, ..., y_{M,k_M}] \in \Gamma$. We also introduce, for each $n = 1, 2, \ldots, N$, the Lagrange basis $\{l_{n,j}\}_{j=1}^{q_n}$ of the space $P_{q_{n-1}}$,

$$l_{n,j} \in P_{q_{n-1}}(\Gamma_n), \quad l_{n,j}(y_{n,k}) = \delta_{jk}, \quad j, k = 1, \ldots, q_n,$$

where δ_{jk} is the Kronecker symbol, and $P_{q-1}(\Gamma) \subset L^2(\Gamma)$ is the span of tensor product polynomials with degree at most $q - 1 = (q_1 - 1, \ldots, q_M - 1)$; i.e., $P_{q-1}(\Gamma) = \bigotimes_{i=1}^M P_{q_i-1}(\Gamma_i)$. Hence the dimension of P_{q-1} is $N_q = \prod_{i=1}^N (q_i)$. Finally we set $l_k(y) = \prod_{n=1}^N l_{n,k_n}(y_n)$.

For any continuous function $g : \Gamma \to \mathbb{R}$ we introduce the Gauss Legendre quadrature formula $E_q^{GL}[g]$ approximating the integral $\int_\Gamma g(y)\,\mathrm{d}y$ as

$$E_q^{GL}[g] = \sum_{k=1}^{N_q} \omega_k g(y_k), \quad \omega_k = \prod_{n=1}^M \omega_{k_n}, \quad \omega_{k_n} = \int_{\Gamma_n} l_{k_n}^2(y)\,\mathrm{d}y \tag{III.48}$$

We now analyze the error introduced by the quadrature formula. The first step is to investigate the smoothness of the map $\xi \mapsto p(\widehat{u}, \xi)$. For this, it is convenient to extend the primal and dual problems to the complex domain. To do so, let us define

$$a(x, z) = a_0(x) + \sum_{i=1}^M \sqrt{\lambda_i} z_i b_i(x)$$

with $z = (z_1, \cdots, z_M) \in \mathbb{C}^M$ and let

$$\mathcal{U}_0 = \{z \in \mathbb{C}^M : \mathcal{R}e(a(x, z)) > 0 \quad a.e. \quad \text{in} \quad D\}.$$

We consider the primal and dual problems extended to the complex domain: $\forall z \in \mathcal{U}_0$ find $y(\cdot, z) \in H_0^1(D; \mathbb{C})$ s.t.

$$\int_D a(x, z)\nabla y(x, z)\nabla v(x)\mathrm{d}x = \int_D (\widehat{u}(x) + g(x))v(x)\mathrm{d}x \quad \forall v \in H_0^1(D; \mathbb{C}) \,, \tag{III.49}$$

and find $p(\cdot, z) \in H_0^1(D; \mathbb{C})$ s.t.

$$\int_D a(x, z)\nabla p(x, z)\nabla v(x)\mathrm{d}x = \int_D (y(x, z) - z_d(x))v(x)\mathrm{d}x \quad \forall v \in H_0^1(D; \mathbb{C}) \,. \tag{III.50}$$

It is well known that problem (III.49) and (III.50) are well posed in $\mathcal{U}_0$. Let now $\Sigma \subset \mathcal{U}_0$

be

$$\Sigma := \{z \in \mathbb{C}^N : \sum_{i=1}^{M} \sqrt{\lambda_i}|z_i| \leq \frac{a_{min}}{2}\}$$

with $a_{min} = ess\inf_{x\in D} a_0(x)$. The next Lemma states that both $z \mapsto y(\cdot, z)$ and $z \mapsto p(\cdot, z)$ are holomorphic functions in $\mathcal{U}_0$ with uniform bounds on Σ. The result for $z \mapsto y(\cdot, z)$ is well known and can be found in reference [CD15] fro example, so that we only give the proof for $z \mapsto p(\cdot, z)$.

Lemma 9. *Both functions $z \mapsto y(\cdot, z)$ and $z \mapsto p(\cdot, z)$ are holomorphic on $\mathcal{U}_0$, and both have a uniform bound on Σ, in the sense that*

$$\max_{z\in\Sigma} \|y(\cdot, \xi)\|_{H_0^1} \leq C_P \frac{\|g + \widehat{u}\|}{a_{min}} \tag{III.51}$$

and

$$\max_{z\in\Sigma} \|p(\cdot, z)\|_{H_0^1} \leq C_P \frac{\|z_d\|}{a_{min}} + C_P^3 \frac{\|g + \widehat{u}\|}{a_{min}^2} \ . \tag{III.52}$$

Proof. It is well known (see e.g. [CD15]) that the function $z \mapsto y(\cdot, z)$ is holomorphic on $\mathcal{U}_0$ with bound (III.51). This property translates to the dual function $z \mapsto p(\cdot, z) \in H_0^1(D; \mathbb{C})$ which is holomorphic in $\mathcal{U}_0$ as well with bound

$$\begin{aligned} \max_{z\in\Sigma} \|p(\cdot, z)\|_{H^1} &\leq C_P \max_{z\in\Sigma} \frac{\|y(\cdot, z) - z_d\|}{a_{min}} \\ &\leq C_P \frac{\|z_d\|}{a_{min}} + C_P \max_{z\in\Sigma} \frac{\|y(\cdot, z)\|}{a_{min}} \\ &\leq C_P \frac{\|z_d\|}{a_{min}} + C_P^3 \frac{\|g + \widehat{u}\|}{a_{min}^2} \ . \end{aligned}$$

□

Based on the last regularity result and following [BNT10], we can state the following error estimate for the quadrature error.

Theorem 18. *Denoting by $\widehat{u}$ the solution of the semi-discrete (in probability) optimal control problem* (III.19) *with $\widehat{E} = E_q^{GL}[\cdot]$ and $p(\widehat{u})$ the corresponding adjoint function, there exists $C > 0$ and $\{r_1, \cdots, r_M\}$ independent of q s.t.*

$$\|\mathbb{E}[p(\widehat{u})] - E_q^{GL}[p(\widehat{u})]\|^2 \leq C \sum_{n=1}^{M} e^{-r_n q_n} \ ,$$

with q_n the number of points used in the quadrature in direction n.

III.K. Proof for increasing Monte Carlo sampling in SG

Here we detail the proof of the bound (III.39) in remark 9. The factor c_j in (III.33) becomes

$$c_j := 1 - \tau_j l + L^2\Big(1 + \frac{2}{N_j}\Big)\tau_j^2 = 1 - \frac{\tau_0 l}{j} + L^2\Big(1 + 2j^{1-\tau_0 l}\Big)\frac{\tau_0^2}{j^2}\ ,$$

for $\tau_j = \tau_0/j$ and $N_j \sim i^{\tau_0 l - 1}$ with $\tau_0 l - 1 > 0$. We use the recursive formula (III.33) and set, as before, $u_\star^h$ to be the exact optimal control for the FE problem defined in (III.11). We emphasize that (III.11) has no approximation in probability space. Setting $a_j = \mathbb{E}[\|u_j^h - u_\star^h\|^2]$ and $\beta_j = \frac{2\tau_j^2}{N_j}\mathbb{E}[\|\nabla f(u_\star^h, \omega)\|^2]$, we have from (III.33), applied to the sequence of FE solutions $\{u_j^h\}_{j>0}$,

$$\begin{aligned} a_{j+1} \leq & c_j a_j + \beta_j \\ \leq & c_j c_{j-1} a_{j-1} + c_j \beta_{j-1} + \beta_j \\ \leq & \cdots \\ \leq & \underbrace{\Big(\prod_{i=1}^{j} c_i\Big)}_{=\kappa_j} a_1 + \underbrace{\sum_{i=1}^{j} \beta_i \prod_{l=i+1}^{j} c_l}_{=\mathcal{B}_j}\,. \end{aligned} \tag{III.53}$$

For the first term κ_j, computing its logarithm, we have

$$\log(\kappa_j) \leq \sum_{i=1}^{j} \log(1 - \frac{\tau_0 l}{i} + \frac{M'}{i^2}) \leq \sum_{i=1}^{j} \frac{-\tau_0 l}{i} + \sum_{i=1}^{j} \frac{M'}{i^2},$$

where we have set $M' = 3\tau_0^2 L^2$ as we have $1 - \tau_0 l < 0$ and thus $j^{1-\tau_0 l} \leq 1$ for every $j \geq 1$. Therefore

$$\log(\kappa_j) \leq -\tau_0 l \log j + M'', \ \text{ with } \ M'' = \sum_{i=1}^{\infty} \frac{M'}{i^2}$$

and $\kappa_j \lesssim j^{-\tau_0 l}$. For the second term $\mathcal{B}_j$ in (III.53) we have

$$\mathcal{B}_j = \sum_{i=1}^{j} \beta_i \prod_{k=i+1}^{j} c_k \leq \sum_{i=1}^{j} S' i^{-\tau_0 l - 1} \underbrace{\prod_{k=i+1}^{j} \Big(1 - \frac{\tau_0 l}{k} + \frac{3\tau_0^2 L^2}{k^2}\Big)}_{=K_{ij}}, \ \text{ with } \ S' = 2\tau_0^2 \mathbb{E}[\|\nabla f(u_\star^h, \omega)\|^2]\ .$$

For the term K_{ij} we find that

$$\begin{aligned}\log(K_{ij}) &= \sum_{k=i+1}^{j} \log\left(1 - \frac{\tau_0 l}{k} + \frac{M'}{k^2}\right)\\ &\leq \sum_{k=i+1}^{j} \Big(- \frac{\tau_0 l}{k} + \frac{M'}{k^2}\Big)\\ &\leq -\tau_0 l(\log(j+1) - \log(i+1)) + M'\left(\frac{1}{i} - \frac{1}{j}\right),\end{aligned}$$

which shows that

$$K_{ij} \leq (j+1)^{-\tau_0 l}(i+1)^{\tau_0 l} \exp\left(M'\left(\frac{1}{i} - \frac{1}{j}\right)\right) .$$

It follows that

$$\begin{aligned}\mathcal{B}_j &\leq (j+1)^{-\tau_0 l} \underbrace{\exp\left(-\frac{M'}{j}\right)}_{\leq 1} \sum_{i=1}^{j} S' i^{-\tau_0 l - 1}(i+1)^{\tau_0 l} \underbrace{\exp\left(\frac{M'}{i}\right)}_{\leq \exp(M')}\\ &\leq S' \exp(M')(j+1)^{-\tau_0 l} \sum_{i=1}^{j} (i+1)^{-1} \lesssim j^{-\tau_0 l} \log(j),\end{aligned}$$

for $\tau_0 > 1/l$. Eventually, we obtained the following upper bound for two constants $D_3 > 0$ and $D_4 > 0$:

$$a_{j+1} \leq D_3 j^{-\tau_0 l} a_1 + D_4 j^{-\tau_0 l} \log(j). \tag{III.54}$$

We conclude that

$$a_{j+1} \leq D_4 j^{-\tau_0 l} \log(j), \tag{III.55}$$

with D_4 possibly depending on $\|u_0^h - u_\star^h\|$. Finally, splitting the error as

$$\mathbb{E}[\|u_j^h - u_\star\|^2] \leq 2\mathbb{E}[\|u_j^h - u_\star^h\|^2] + 2\mathbb{E}[\|u_\star^h - u_\star\|^2]$$

and using (III.18) to bound the second term, the claim follows.

IV Multilevel Stochastic Gradient method for PDE-constrained OCPs with uncertain parameters

This Chapter is essentially the same as [MN19], in preparation.

IV.A. Introduction

In this paper, we present a multilevel Monte Carlo (MLMC) Stochastic Gradient (SG) approach, to solve an Optimal Control Problem (OCP) with PDE constraints with random parameters. The deterministic control acts as a forcing term in the random PDE and is chosen so as to minimize some expected cost functional. We use a Stochastic Gradient approach to compute the optimal control and independent MLMC estimators when computing the steepest descent direction, of the expected cost, at each iteration. The accuracy of the MLMC estimator is increased over the iterations, with consequent increases in computational cost. The refinement strategy is chosen a-priori as a function of the iteration counter.

The Stochastic Gradient algorithm has been introduced by Robbins and Monro [RM51] and is widely used to solve robust optimization problems, involving the optimization of an expected loss function, which can be stated as

$$u_\star \in \arg\min_{u \in U} J(u), \quad J(u) = \mathbb{E}_\omega[f(u,\omega)]$$

where $\omega \in \Gamma$ denotes a random elementary event and f a "loss" or objective function. The usual SG algorithm then writes:

$$u_{j+1} = u_j - \tau_j \nabla f(u_j, \omega_j)$$

where the sequence of ω_j is independent and identically distributed (iid) and the step-size is usually chosen as $\tau_j = \tau_0/j$, with τ_0 sufficiently large. In particular, when $u \mapsto f(u,\omega)$ is strongly convex for a.e. $\omega \in \Gamma$, the SG algorithm guarantees an algebraic convergence rate on the mean squared error, i.e. $\mathbb{E}[\|u_j - u_\star\|^2] \lesssim 1/j$, where $u_\star$ denotes the exact optimal control. In our setting, the evaluation of the loss functional $f(u_j, \cdot)$ involves the solution of a PDE, which can not be done exactly, except in very simple cases and requires a discretization step. This, in turn, induces an error in the evaluation of the cost functional, which can be kept small at the price of a high computational cost.

The multilevel Monte Carlo method has been proven to be very effective in reducing the cost of computing expectations of output functionals of differential models, compared to classic Monte Carlo estimators, by exploiting a hierarchy of discretizations, with increasing accuracy and computational costs. Essentially, the MLMC estimator makes most of the computation on very coarse (and cheap) discretizations and corrects it with only few evaluations on finer (and more expensive) discretizations.

The multilevel paradigm has been introduced by Heinrich [Hei00] for parametric integration. It has been extended to weak approximations of stochastic differential equations (SDEs) in [Gil08] and has shown its efficiency, as a tool in numerical computations. Recently, the application of MLMC methods to uncertainty quantification problems involving PDEs with random data has been investigated from the mathematical point of view in a number of works [BSZ11, BLS13, CST13, CGST11, MSŠ12, TSGU13, NT15].

In the most favorable cases, it has been shown that the cost W of computing the expected value of some output quantity of a stochastic differential model with accuracy tol, scales as $W \lesssim tol^{-2}$, and does not see the cost of solving the problem on fine discretizations. In this work, we use MLMC within a SG algorithm, by replacing the single realization $\nabla f(u_j, \omega_j)$ by a MLMC estimator, based on a hierarchy of finite element (FE) discretization. In particular, we present a full convergence and complexity analysis of the resulting MLSG algorithm in the case of a quadratic, strongly convex, OCP. By reducing progressively the bias and the variance of the MLMC estimator over the iterations at a proper rate, we are able to recover an optimal complexity $W \lesssim tol^{-2}$ in the computation of the optimal control, analogous to the one for the computation of a single expectation. This result considerably improves the one in our previous work [MKN18] where we have studied the SG method in which a single realization $\nabla f(u_j, \omega_j)$ is taken at each iteration, and computed on progressively finer discretization over the iterations. An alternative way to consider a MLMC estimator with a Stochastic Gradient algorithm, although not in the context of PDE constrained OCPs, has been proposed in [Fri16], which also leads to the optimal complexity $W \lesssim tol^{-2}$ in favorable cases. The idea in [Fri16] is to build a sequence of coupled standard Robbins-Monro algorithm with different discretization levels to construct a multilevel estimator of the optimal control. We propose, instead, to construct a single SG algorithm that uses at each iteration a multilevel estimate for $\mathbb{E}[\nabla f(u, \cdot)]$, with increasing accuracy over the iterations (a similar approach can be found in [DM15]).

We also propose a randomized version of the MLSG algorithm, which uses the unbiased multilevel Monte Carlo algorithm proposed in [RG12, RG15] (see also [Gil15, Section

2.2]). In this randomized version, we replace the full MLMC sampler at each iteration j of the Stochastic Gradient algorithm by only one evaluation of the difference of the objective function on levels l_j and $l_j - 1$ where the level l_j is drawn randomly (and independently at each iteration) from a suitable probability mass function over all levels. We show that this version of the MLSG algorithm achieves optimal complexity $W \lesssim tol^{-2}$. The main advantage of this randomized version, w.r.t. the one that uses a full MLMC estimator at each iteration, is that it requires less parameters to tune, which is preferable, from a numerical point of view.

The outline of the Chapter is as follow. In Section IV.B, we present the problem setting, and recall some results about existence and uniqueness of the optimal control, and its finite element approximation. We also recall the multilevel Monte Carlo estimator. Then in Section IV.C, we introduce the multilevel Monte Carlo Stochastic Gradient method. Our algorithm uses a hierarchy of uniformly refined meshes with mesh sizes forming a geometric sequence. This is justified from the arguments in [HAvST16]. We discuss the optimal strategy to reduce the bias and variance of the MLMC estimator over the iterations and derive a complexity result. In Section IV.D we present the randomized version of the MLSG algorithm and derive a complexity result. Section IV.E presents a numerical example, and assesses theoretical results from Sections IV.C and IV.D. Finally, Section IV.F presents some conclusions and future perspectives of this work.

IV.B. Problem setting

We start by introducing the primal problem that will be part of the OCP discussed in the following. We consider the problem of finding the solution $y : D \times \Gamma \to \mathbb{R}$ of the elliptic random PDE

$$\begin{cases} -\operatorname{div}(a(x,\omega)\nabla y(x,\omega)) = g(x) + u(x), & x \in D, \quad \omega \in \Gamma, \\ y(x,\omega) = 0, & x \in \partial D, \quad \omega \in \Gamma, \end{cases} \tag{IV.1}$$

where $D \subset \mathbb{R}^d$ denotes the physical domain and $(\Gamma, \mathcal{F}, \mathbb{P})$ is a complete probability space. The diffusion coefficient a is a random field, g is a deterministic source term and u is the deterministic control. The solution of (IV.1) for a given control u will be equivalently denoted $y_\omega(u)$, or simply $y(u)$ in what follows. Let $U = L^2(D)$ be the set of all admissible control functions and $Y = H_0^1(D)$ the space of the solutions of (IV.1) endowed with the norm $\|v\|_{H_0^1(D)} = \|\nabla v\|$ where $\|\cdot\|$ denotes the $L^2(D)$-norm induced by the inner product $\langle \cdot, \cdot \rangle$. The ultimate goal is to determine an optimal control u^*, in the sense that:

$$u_\star \in \operatorname*{arg\,min}_{u \in U} J(u), \quad \text{s.t.} \quad y_\omega(u) \in Y \quad \text{solves} \quad \text{(IV.1)} \quad \text{almost surely (a.s.) in } \Gamma. \tag{IV.2}$$

Here, $J(u) := \mathbb{E}[f(u,\omega)]$ is the objective function with $f(u,\omega) = \frac{1}{2}\|y_\omega(u) - z_d\|^2 + \frac{\beta}{2}\|u\|^2$ and z_d is the target function that we would like the state y to approach as close as possible. We use assumptions and results from [MKN18, Sections 2 and 3] to guarantee well posedness of (IV.2) and regularity of solutions.

Assumption 14. *The diffusion coefficient $a \in L^\infty(D \times \Gamma)$ is bounded and bounded away from zero a.e. in $D \times \Gamma$, i.e.*

$$\exists \quad a_{\min}, a_{\max} \in \mathbb{R} \quad \text{such that} \quad 0 < a_{\min} \leq a(x,\omega) \leq a_{\max} \quad \text{a.e. in } D \times \Gamma.$$

Assumption 15. *The regularization parameter β is strictly positive, i.e. $\beta > 0$ and the deterministic source term is such that $g \in L^2(D)$.*

In what follows, we denote the $L^2(D)$-functional representation of the Gateaux derivative of J, by $\nabla J(u)$, namely

$$\int_D \nabla J(u) \delta u \, \mathrm{d}x = \lim_{\epsilon \to 0} \frac{J(u + \epsilon \delta u) - J(u)}{\epsilon} \quad \forall \delta u \in L^2(D).$$

Then existence and uniqueness of the OCP (IV.2) can be stated as follows.

Theorem 19. *Under Assumptions 14 and 15, the OCP* (IV.2) *admits a unique control $u_\star \in U$. Moreover*

$$\nabla J(u) = \mathbb{E}[\nabla f(u,\cdot)] \ \text{ with } \ \nabla f(u,\omega) = \beta u + p_\omega(u), \tag{IV.3}$$

where $p_\omega(u) = p$ is the solution of the adjoint problem (a.s. in Γ)

$$\begin{cases} -\operatorname{div}(a(\cdot,\omega)\nabla p(\cdot,\omega)) &= y(\cdot,\omega) - z_d & \text{in } D, \\ p(\cdot,\omega) &= 0 & \text{on } \partial D. \end{cases} \tag{IV.4}$$

We continue recalling the weak formulation of (IV.1), which reads

$$\text{find } y_\omega \in Y \ s.t. \ b_\omega(y_\omega, v) = \langle g + u, v \rangle \quad \forall v \in Y \qquad \text{for a.e. } \omega \in \Gamma, \tag{IV.5}$$

where $b_\omega(y,v) := \int_D a(\cdot,\omega) \nabla y \nabla v dx$. Similarly, the weak form of the adjoint problem (IV.4) reads:

$$\text{find } p_\omega \in Y \ s.t. \ b_\omega(v, p_\omega) = \langle v, y_\omega - z_d \rangle \quad \forall v \in Y \qquad \text{for a.e. } \omega \in \Gamma. \tag{IV.6}$$

We can thus rewrite the OCP (IV.2) equivalently as:

$$\begin{cases} \min_{u \in U} J(u), \quad J(u) = \frac{1}{2}\mathbb{E}[\|y_\omega(u) - z_d\|^2] + \frac{\beta}{2}\|u\|^2 \\ \text{s.t.} \quad y_\omega(u) \in Y \quad \text{solves} \\ b_\omega(y_\omega(u), v) = \langle g + u, v \rangle \quad \forall v \in Y \qquad \text{for a.e. } \omega \in \Gamma. \end{cases} \tag{IV.7}$$

Following [MKN18], we now recall two regularity results about Lipschitz continuity and strong convexity of f in the particular setting of the problem considered here.

Lemma 10 (Lipschitz continuity)**.** *The random functional f is such that:*

$$\|\nabla f(u,\omega) - \nabla f(v,\omega)\| \leq Lip\|u-v\| \quad \forall u,v \in U \text{ and a.e. } \omega \in \Gamma, \tag{IV.8}$$

with $Lip = \beta + \frac{C_p^4}{a_{min}^2}$, where C_p is the Poincaré constant, $C_p = \sup_{v\in Y/\{0\}} \frac{\|v\|}{\|\nabla v\|}$.

Lemma 11 (Strong convexity)**.** *The random functional f is such that:*

$$\frac{l}{2}\|u-v\|^2 \leq \langle \nabla f(u,\omega) - \nabla f(v,\omega), u-v\rangle \quad \forall u,v \in U \text{ and a.e. } \omega \in \Gamma, \tag{IV.9}$$

with $l = 2\beta$.

IV.B.1. Finite Element approximation

In order to compute numerically an optimal control we consider a Finite Element (FE) approximation of the infinite dimensional OCP (IV.7). Let us denote by $\{\tau_h\}_{h>0}$ a family of regular triangulations of D and choose Y^h to be the space of continuous piece-wise polynomial functions of degree r over τ_h that vanish on ∂D, i.e. $Y^h = \{y \in C^0(\overline{D}) : y|_K \in \mathbb{P}_r(K) \quad \forall K \in \tau_h, y|_{\partial D} = 0\} \subset Y$, and $U^h = Y^h$. We reformulate the OCP (IV.7) as a finite dimensional OCP in the FE space:

$$\begin{cases} \min_{u^h\in U^h} J^h(u^h), \quad J^h(u^h) = \frac{1}{2}\mathbb{E}[\|y^h_\omega(u^h) - z_d\|^2] + \frac{\beta}{2}\|u^h\|^2 \\ \text{s.t. } y^h_\omega \in Y^h \text{ and} \\ b_\omega(y^h_\omega(u^h), v^h) = \langle u^h + g, v^h\rangle \quad \forall v^h \in Y^h \quad \text{for a.e. } \omega \in \Gamma. \end{cases} \tag{IV.10}$$

Under the following regularity assumption on the domain and diffusion coefficient:

Assumption 16. *The domain $D \subset \mathbb{R}^d$ is polygonal convex and the random field $a \in L^\infty(D\times\Gamma)$ is such that $\nabla a \in L^\infty(D\times\Gamma)$,*

the following error estimate has been obtained in [MKN18]. In order to lighten the notation, we omit the subscript ω in $y_\omega(\cdot)$ and $p_\omega(\cdot)$ from now on.

Theorem 20. *Let $u_\star$ be the optimal control, solution of problem* (IV.7)*, and denote by $u^h_\star$ the solution of the approximate problem* (IV.10)*. Suppose that $y(u_\star), p(u_\star) \in L^2_{\mathbb{P}}(\Gamma; H^{r+1}(D))$ and Assumption 16 holds; then*

$$\begin{aligned} \|u_\star - u^h_\star\|^2 + \mathbb{E}[\|y(u_\star) - y^h(u^h_\star)\|^2] + h^2\mathbb{E}[\|y(u_\star) - y^h(u^h_\star)\|^2_{H^1_0}] \\ \leq A_1 h^{2r+2}\left(\mathbb{E}[|y(u_\star)|^2_{H^{r+1}}] + \mathbb{E}[|p(u_\star)|^2_{H^{r+1}}]\right), \end{aligned} \tag{IV.11}$$

with a constant A_1 independent of h.

Notice that the OCP (IV.10) can be equivalently formulated in U instead of U^h:

$$\begin{cases} \min_{u \in U} J^h(u), \quad J^h(u) = \frac{1}{2}\mathbb{E}[\|y^h_\omega(u) - z_d\|^2] + \frac{\beta}{2}\|u\|^2 \\ \text{s.t. } y^h_\omega \in Y^h \text{ and} \\ b_\omega(y^h_\omega(u), v^h) = \langle u + g, v^h \rangle \quad \forall v^h \in Y^h, \quad \text{for a.e. } \omega \in \Gamma. \end{cases} \tag{IV.12}$$

Indeed, if we decompose any $u \in U$ into $u = u_h + w$ with $u_h \in U^h$ and $\langle w, v^h \rangle = 0, \quad \forall v^h \in U^h$, it follows that

$$J^h(u) = \frac{1}{2}\mathbb{E}[\|y^h_\omega(u^h) - z_d\|^2] + \frac{\beta}{2}\|u^h\|^2 + \frac{\beta}{2}\|w\|^2, \tag{IV.13}$$

so clearly the optimal control $\widetilde{u}^*$ of (IV.12) satisfies $\widetilde{u}^* = u^{h*} + w^*$ with $w^* = 0$ and u^{h*} solution of (IV.10), i.e. the optimal control of (IV.12) is indeed a FE function in U^h. For later developments it will be more convenient to consider the formulation (IV.12) rather than (IV.10).

Following analogous developments as in [MKN18], it is straightforward to show that $\forall u \in U$

$$\nabla J^h(u) = \mathbb{E}[\nabla f^h(u, \cdot)] \text{ with } \nabla f^h(u, \omega) = \beta u + p^h_\omega(u) \in U, \tag{IV.14}$$

where $p^h_\omega(u)$ solves the FE adjoint problem which reads

$$\text{find } p^h_\omega(u) \in Y^h \; s.t. \; b_\omega(v^h, p^h_\omega(u)) = \langle v^h, y^h_\omega(u) - z_d \rangle \quad \forall v^h \in Y^h, \quad \text{for a.e. } \omega \in \Gamma, \tag{IV.15}$$

and $y^h_\omega(u)$ solves the primal problem formulated in (IV.12). Moreover, $\nabla f^h(u, \cdot)$ satisfies Lipschitz and convexity properties analogous to those in Lemmas 10 and 11, with the same constants:

Lemma 12 (Lipschitz continuity)**.**

$$\|\nabla f^h(u, \omega) - \nabla f^h(v, \omega)\| \leq Lip\|u - v\| \quad \forall u, v \in U \quad \forall h > 0, \quad \text{for a.e. } \omega \in \Gamma.$$

Lemma 13 (Strong convexity)**.**

$$\frac{l}{2}\|u - v\|^2 \leq \langle u - v, \nabla f^h(u, \omega) - \nabla f^h(v, \omega) \rangle \quad \forall u, v \in U \quad \forall h > 0, \quad \text{for a.e. } \omega \in \Gamma.$$

We conclude this section by stating an error bound on the FE approximation of the functional f when evaluated at the optimal control $u_\star$.

Lemma 14. *Let $u_\star$ be the solution of the OCP* (IV.7) *and assume $y(u_\star), p(u_\star) \in L^2_{\mathbb{P}}(\Gamma, H^{r+1}(D))$. Then, there exists $C(u_\star) > 0$ such that, $\forall h > 0$:*

$$\mathbb{E}[\|\nabla f^h(u_\star, \cdot) - \nabla f(u_\star, \cdot)\|^2] \leq C(u_\star) h^{2r+2}.$$

Proof. A proof of this result can be found for example in [MKN18, Corollary 1]. □

IV.B.2. Multilevel Monte Carlo (MLMC) estimator

The OCP (IV.7), or its FE approximation (IV.12), involves computing the expectation of the cost functional $f(u, \omega)$. For this, in this work, we consider a Multilevel Monte Carlo estimator. The key idea of MLMC [Gil15] is to estimate the mean of a random quantity P related to the PDE problem (IV.1), by simultaneously using Monte Carlo (MC) estimators on several approximations P_l, $l = 0, \dots, L$ of P that are built on a hierarchy of computational grids τ_{h_l} with discretization parameters $h_L < \cdots < h_1 < h_0$ (FE mesh sizes in our context). Considering N_l iid realizations $\omega_{l,i}$, $i = 1, \dots, N_l$ of the system's random input parameters on each level $0 \leq l \leq L$, the MLMC estimator for the expected value $\mathbb{E}[P]$ is given by

$$E^{\mathrm{MLMC}}_{L,\overrightarrow{N}}[P] = \frac{1}{N_0}\sum_{i=1}^{N_0} P_0(\omega_{0,i}) + \sum_{l=1}^{L}\frac{1}{N_l}\sum_{i=1}^{N_l}\left(P_l(\omega_{l,i}) - P_{l-1}(\omega_{l,i})\right) \tag{IV.16}$$

$$= \sum_{l=0}^{L}\frac{1}{N_l}\sum_{i=1}^{N_l}\left(P_l(\omega_{l,i}) - P_{l-1}(\omega_{l,i})\right), \tag{IV.17}$$

where we have set $P_{-1} = 0$, and $\overrightarrow{N} = \{N_0, N_1, \dots, N_L\}$. We recall the main complexity result from [Gil15]. Denote by V_l the variance of $P_l - P_{l-1}$ that is $V_l = \mathbb{V}ar[P_l - P_{l-1}]$ and by C_l the expected cost of generating one realization of (P_l, P_{l-1}), $C_l = \mathbb{E}[Cost(P_l(\omega_{l,i}), P_{l-1}(\omega_{l,i}))]$ and consider a sequence of uniform meshes with $h_l = h_0\delta^{-l}$, $\delta > 1$. If there exist positive constants $q_w, q_s, q_c, c_w, c_s, c_c$ such that $q_w \geq \frac{1}{2}\min(q_s, q_c)$ and

$$\left|\mathbb{E}[P_l - P]\right| \leq c_w 2^{-q_w l}, \tag{IV.18}$$

$$V_l \leq c_s 2^{-q_s l}, \tag{IV.19}$$

$$C_l \leq c_c 2^{q_c l}, \tag{IV.20}$$

then there exists a positive constant c_0 such that for any $\epsilon < e^{-1}$ there are values $L = L(\epsilon)$ and $N_l = N_l(\epsilon)$, $l = 0, \dots, L(\epsilon)$, for which the multilevel estimator (IV.16) has a mean-squared-error MSE with bound

$$MSE = \sum_{l=0}^{L}\frac{V_l}{N_l} \leq \epsilon^2,$$

and an expected computational cost $C = \sum_{l=0}^{L} C_l N_l$ with bound

$$C \leq \begin{cases} c_0 \epsilon^{-2}, & \text{if } q_s > q_c, \\ c_0 \epsilon^{-2} \left(\log \epsilon\right)^2, & \text{if } q_s = q_c, \\ c_0 \epsilon^{-2-(q_c-q_s)/q_w}, & \text{if } q_s < q_c. \end{cases}$$

For a fixed variance V_l, the optimal number of samples at level l that minimize the cost C_l, is given by the formula:

$$N_l^*(\epsilon) = \left\lceil \epsilon^{-2} \sqrt{\frac{V_l}{C_l}} \sum_{k=0}^{L(\epsilon)} \sqrt{V_k C_k} \right\rceil.$$

IV.C. Multilevel Stochastic Gradient (MLSG) algorithm

First introduced by Robbins and Monro in 1961, Stochastic Approximation (SA) techniques, such as Stochastic Gradient (SG) [RM51, PJ92, SDR09, SRB13, DB16] are popular techniques in the machine learning community, to minimize a sum of functions (say with q terms), and allowing the user to only compute the gradient on batches of size $p \ll q$ at each iteration, these batches being selected at random and independently. The classic version of such a method, the so-called Robbins-Monro (RM) method, works as follows. Within the steepest descent algorithm, the exact gradient $\nabla J = \nabla \mathbb{E}[f] = \mathbb{E}[\nabla f]$[1] is replaced by $\nabla f(\cdot, \omega_j)$, where the random variable ω_j (i.e. a random point $\omega_j \in \Gamma$ with distribution $\mathbb{P}$) is re-sampled independently at each iteration of the steepest-descent method:

$$u_{j+1} = u_j - \tau_j \nabla f(u_j, \omega_j). \tag{IV.21}$$

Here, τ_j is the step-size of the algorithm and is decreasing as τ_0/j in the usual setting. The RM method applied to the OCP (IV.7) has been analyzed in [MKN18]. In this paper, we consider a generalization of this method, in which the "point-wise" gradient $\nabla f(\cdot, \omega_j)$ is replaced by a MLMC estimator, which is independent of the ones used in the previous iterations. Specifically, we introduce a hierarchy of refined meshes $\tau_{h_0}, \tau_{h_1}, \ldots$ with $h_0 > h_1 > \ldots$ and corresponding FE spaces Y^{h_l}, $l = 0, 1, \ldots$, and $U^{h_l} = Y^{h_l}$ and use, at each iteration, the MLMC estimator defined in Section IV.B.2, $E^{\text{MLMC}}_{L_j, \overrightarrow{N}_j}[P(u_j)]$, where $P(u) = \nabla f(u, \cdot)$ and $P_l(u) = \nabla f^{h_l}(u, \cdot)$. The total number of levels L_j and samples per level $\overrightarrow{N}_j = \{N_{j,0}, N_{j,1}, \ldots, N_{j,L_j}\}$ are allowed to depend on j. In particular, we require L_j and $N_{j,l}$ to be non decreasing sequences in j and $L_j, \sum_{l=0}^{L_j} N_{j,l} \to \infty$ as $j \to \infty$. We study only sequences of *nested* meshes of size $h_l = h_0 2^{-l}$ (i.e. with $\delta = 2$). This implies, in particular, $Y^{h_{l-1}} \subset Y^{h_l}$, $\forall l \geq 1$. In this restricted context, the optimization procedure

[1] the fact that the gradient and the expectation can be exchanged for the OCP (IV.2) has been proven in [MKN18]

reads:

$$\begin{aligned} u_{j+1} &= u_j - \tau_j E^{\mathrm{MLMC}}_{L_j,\overrightarrow{N}_j}[\nabla f(u_j,\cdot)], \\ &= u_j - \tau_j \left(\beta u_j + E^{\mathrm{MLMC}}_{L_j,\overrightarrow{N}_j}[p(u_j,\cdot)]\right) \quad \forall j \geq 1, \quad u_1 \in Y^{h_0}, \end{aligned} \tag{IV.22}$$

with

$$E^{\mathrm{MLMC}}_{L_j,\overrightarrow{N}_j}[p(u_j,\cdot)] = \sum_{l=0}^{L_j} \frac{1}{N_{j,l}} \sum_{i=1}^{N_{j,l}} \left(p^{h_l}(u_j,\omega^j_{l,i}) - p^{h_{l-1}}(u_j,\omega^j_{l,i})\right). \tag{IV.23}$$

Notice that $u_j \in Y^{h_{L_{j-1}}}$ for any $j \geq 1$. This can be shown by a simple induction argument noticing that if $u_j \in Y^{h_{L_{j-1}}}$, then $\widehat{\nabla J}_{MLSG} := E^{\mathrm{MLMC}}_{L_j,\overrightarrow{N}_j}[\nabla f(u_j,\cdot)] \in Y^{h_{L_j}}$ which implies $u_{j+1} \in Y^{h_{L_j}}$ thanks to the nestedness assumption, $Y^{h_p} \subset Y^{h_q}$ if $p \leq q$.

Remark 11. *In the case where non-nested meshes were used, we would need to introduce suitable interpolation operators $I^{h_q}_{h_p} : Y^{h_p} \to Y^{h_q}$ with $p < q$ and modify the MLSG algorithm as follows*

$$u_{j+1} = (1 - \beta\tau_j) I^{h_{L_j}}_{h_{L_{j-1}}}(u_j) - \tau_j E^{MLMC}_{L_j,\overrightarrow{N}_j}[p(u_j,\cdot)],$$

with

$$E^{MLMC}_{L_j,\overrightarrow{N}_j}[p(u_j,\cdot)] = \sum_{l=0}^{L_j} \frac{1}{N_{j,l}} I^{h_{L_j}}_{h_l} \sum_{i=1}^{N_{j,l}} \left(p^{h_l}(u_j,\omega^j_{l,i}) - I^{h_l}_{h_{l-1}} p^{h_{l-1}}(u_j,\omega^j_{l,i})\right).$$

IV.C.1. Convergence analysis

Let us denote by

$$F_j = \sigma(\{\omega^k_{l,i}\},\ k = 1,\dots,j-1;\ l = 0,\dots,L_k;\ i = 1,\dots,N_{k,l})$$

the σ-algebra generated by all the random variables $\{\omega^k_{l,i}\}$ up to iteration $j-1$. Notice that the control u_j is measurable with respect to F_j, i.e. the process $\{u_j\}$ is $\{F_j\}$-adapted. We denote the conditional expectation to F_j by $\mathbb{E}[\cdot|F_j]$.

Theorem 21. *For any non decreasing deterministic or random F_j-adapted sequence $\{L_j,\ j \geq 0\}$, $\{N_{j,l},\ j \geq 0,\ l \in \{0,\dots,L_j\}\}$, denoting by u_j the approximated control obtained at iteration j using the recursive formula* (IV.22)*, and $u_\star$ the exact control for the continuous OCP* (IV.7)*, we have:*

$$\mathbb{E}[\|u_{j+1} - u_\star\|^2|F_j] \leq c_j\|u_j - u_\star\|^2 + 2\tau_j^2\sigma_j^2 + \left(2\tau_j^2 + \frac{2\tau_j}{l}\right)\epsilon_j^2 \tag{IV.24}$$

with $c_j = 1 - \frac{l}{2}\tau_j + 2Lip^2\left(1 + 4\sum_{l=0}^{L_j}\frac{1}{N_{j,l}}\right)\tau_j^2$
$\sigma_j^2 = \sigma^2(L_j, \overrightarrow{N}_j) = \mathbb{E}[\|E^{MLMC}_{L_j,\overrightarrow{N}_j}[\nabla f(u_\star)] - \mathbb{E}[\nabla f^{h_{L_j}}(u_\star)|F_j]\|^2|F_j]$ *the variance term*
$\epsilon_j^2 = \epsilon^2(L_j) = \|\mathbb{E}[\nabla f^{h_{L_j}}(u_\star)|F_j] - \mathbb{E}[\nabla f(u_\star)]\|^2$ *the squared bias term.*

Proof. Using the optimality condition

$$\mathbb{E}[\nabla f(u_\star)] = 0,$$

and the fact that the expectation of the MLMC estimator for any deterministic, or random F_j-measurable $u \in U$, is

$$\mathbb{E}[E^{\text{MLMC}}_{L_j,\overrightarrow{N}_j}[\nabla f(u)]|F_j] = \mathbb{E}[\nabla f^{h_{L_j}}(u)|F_j],$$

we can decompose the error at iteration $j+1$ as

$$\begin{aligned}\|u_{j+1} - u_\star\|^2 = \Big\| u_j - u_\star - \tau_j \underbrace{\left(E^{\text{MLMC}}_{L_j,\overrightarrow{N}_j}[\nabla f(u_j,\cdot)] - \mathbb{E}[\nabla f^{h_{L_j}}(u_j,\cdot)|F_j]\right)}_{=B} \\ - \tau_j \underbrace{\left(\mathbb{E}[\nabla f^{h_{L_j}}(u_j,\cdot)|F_j] - \mathbb{E}[\nabla f^{h_{L_j}}(u_\star,\cdot)|F_j]\right)}_{=A} - \tau_j \underbrace{\left(\mathbb{E}[\nabla f^{h_{L_j}}(u_\star,\cdot)|F_j] - \mathbb{E}[\nabla f(u_\star,\cdot)]\right)}_{=C} \Big\|^2.\end{aligned} \tag{IV.25}$$

Then, taking the conditional expectation of (IV.25) to the σ-algebra F_j, we obtain the 10 following terms

$$\begin{aligned}\mathbb{E}[\|u_{j+1} - u_\star\|^2|F_j] = \mathbb{E}[\|u_j - u_\star\|^2|F_j] + \tau_j^2\mathbb{E}[\|B\|^2|F_j] + \tau_j^2\mathbb{E}[\|A\|^2|F_j] + \tau_j^2\mathbb{E}[\|C\|^2|F_j] \\ - 2\tau_j\langle u_j - u_\star, \mathbb{E}[B|F_j]\rangle - 2\tau_j\langle u_j - u_\star, \mathbb{E}[A|F_j]\rangle - 2\tau_j\langle u_j - u_\star, \mathbb{E}[C|F_j]\rangle \\ + 2\tau_j^2\mathbb{E}[\langle A, C\rangle|F_j] + 2\tau_j^2\mathbb{E}[\langle A, B\rangle|F_j] + 2\tau_j^2\mathbb{E}[\langle B, C\rangle|F_j].\end{aligned} \tag{IV.27}$$

The second term can be bounded, using the expression of the variance of the MLMC estimator, and the Lipschitz property in Lemma 12, as

$$\begin{aligned}
\mathbb{E}[\|B\|^2|F_j] &= \mathbb{E}[\|E^{\mathrm{MLMC}}_{L_j,\vec{N}_j}[\nabla f(u_j,\cdot)] - \mathbb{E}[\nabla f^{h_{L_j}}(u_j,\cdot)|F_j]\|^2|F_j] \\
&= \mathbb{E}\left[\left\|\sum_{l=0}^{L_j}\frac{1}{N_{j,l}}\sum_{i=1}^{N_{j,l}}\underbrace{\Big(\nabla f^{h_l}(u_j,\omega^j_{i,l}) - \nabla f^{h_{l-1}}(u_j,\omega^j_{i,l}) - \mathbb{E}[\nabla f^{h_l}(u_j,\cdot) - \nabla f^{h_{l-1}}(u_j,\cdot)|F_j]\Big)}_{=E_{i,l}(u_j)}\right\|^2 |F_j\right] \\
&= \mathbb{E}\left[\sum_{l=0}^{L_j}\sum_{i=1}^{N_{j,l}}\sum_{l'=0}^{L_j}\sum_{i'=1}^{N_{j,l'}}\frac{1}{N_{j,l}}\frac{1}{N_{j,l'}}\langle E_{i,l}(u_j), E_{i',l'}(u_j)\rangle|F_j\right] \\
&= \sum_{l=0}^{L_j}\sum_{i=1}^{N_{j,l}}\sum_{l'=0}^{L_j}\sum_{i'=1}^{N_{j,l'}}\frac{1}{N_{j,l}}\frac{1}{N_{j,l'}}\underbrace{\mathbb{E}\left[\langle E_{i,l}(u_j), E_{i',l'}(u_j)\rangle|F_j\right]}_{=0 \text{ if } i\neq i' \text{ or } l\neq l'} \\
&= \sum_{l=0}^{L_j}\sum_{i=1}^{N_{j,l}}\frac{1}{N_{j,l}^2}\mathbb{E}\left[\|E_{i,l}(u_j)\|^2|F_j\right] \\
&= \sum_{l=0}^{L_j}\frac{1}{N_{j,l}}\mathbb{E}\left[\|E_{1,l}(u_j)\|^2|F_j\right] \\
&\leq 2\sum_{l=0}^{L_j}\frac{1}{N_{j,l}}\mathbb{E}[\|E_{1,l}(u_j) - E_{1,l}(u_\star)\|^2|F_j] + 2\sum_{l=0}^{L_j}\frac{1}{N_{j,l}}\mathbb{E}[\|E_{1,l}(u_\star)\|^2|F_j] \\
&\leq 8Lip^2\left(\sum_{l=0}^{L_j}\frac{1}{N_{j,l}}\right)\|u_j - u_\star\|^2 + 2\,\underbrace{\mathbb{E}[\|E^{\mathrm{MLMC}}_{L_j,\vec{N}_j}[\nabla f(u_\star,\cdot)] - \mathbb{E}[\nabla f^{h_{L_j}}(u_\star,\cdot)|F_j]\|^2|F_j]}_{=\sigma_j^2}.
\end{aligned}$$

Finally

$$\tau_j^2\mathbb{E}[\|B\|^2|F_j] \leq \tau_j^2 8Lip^2\left(\sum_{l=0}^{L_j}\frac{1}{N_{j,l}}\right)\|u_j - u_\star\|^2 + 2\tau_j^2\sigma_j^2. \qquad \text{(IV.28)}$$

The third term can be bounded as

$$\tau_j^2\mathbb{E}[\|A\|^2|F_j] = \tau_j^2\mathbb{E}[\|\nabla f^{h_{L_j}}(u_j,\cdot) - \nabla f^{h_{L_j}}(u_\star,\cdot)\|^2|F_j] \leq \tau_j^2 Lip^2\|u_j - u_\star\|^2. \qquad \text{(IV.29)}$$

The fourth term $\mathbb{E}[\|C\|^2|F_j] = \epsilon_j^2$ is just the squared bias term.
The fifth term is

$$-2\tau_j\langle u_j - u_\star, \underbrace{\mathbb{E}[B|F_j]}_{=0}\rangle = 0.$$

The sixth term, using strong convexity of f^h uniform in h, is

$$\begin{aligned}-2\tau_j\langle u_j - u_\star, \mathbb{E}[A|F_j]\rangle &= -2\tau_j\langle u_j - u_\star, \mathbb{E}[\nabla f^{h_{L_j}}(u_j,\cdot) - \nabla f^{h_{L_j}}(u_\star,\cdot)|F_j]\rangle \\ &\leq -l\tau_j\|u_j - u_\star\|^2.\end{aligned}$$

The seventh term of (IV.27) can be bounded as

$$-2\tau_j\mathbb{E}\left[\langle u_j - u_\star, C\rangle|F_j\right] \leq \frac{l\tau_j}{2}\|u_j - u_\star\|^2 + \frac{2\tau_j}{l}\mathbb{E}\left[\|C\|^2|F_j\right] = \frac{l\tau_j}{2}\|u_j - u_\star\|^2 + \frac{2\tau_j}{l}\epsilon_j^2.$$

The eight term can be decompose as:

$$2\tau_j^2\mathbb{E}[\langle A, C\rangle|F_j] \leq \tau_j^2\mathbb{E}[\|A\|^2|F_j] + \tau_j^2\epsilon_j^2$$

The ninth and tenth terms vanish since A and C are measurable on F_j and $\mathbb{E}[B|F_j] = 0$. Hence

$$2\tau_j^2\mathbb{E}[\langle B, A\rangle|F_j] = 2\tau_j^2\mathbb{E}[\langle B, C\rangle|F_j] = 0.$$

In conclusion, we have

$$\mathbb{E}[\|u_{j+1}-u_\star\|^2|F_j] = \left(1 - \frac{l}{2}\tau_j + Lip^2\left(2 + 8\sum_{l=0}^{L_j}\frac{1}{N_{j,l}}\right)\tau_j^2\right)\|u_j-u_\star\|^2+2\tau_j^2\sigma_j+\left(2\tau_j^2 + \frac{2}{l}\tau_j\right)\epsilon_j^2. \tag{IV.30}$$

□

From now on, we consider only deterministic sequences $\{L_j\}, \{N_{j,l}\}$, i.e. chosen in advance and not adaptively during the algorithm. In this case the quantities σ_j and ϵ_j defined in Theorem 21 are deterministic as well. From Theorem 21, taking the full expectation $\mathbb{E}[\cdot]$ in (IV.24) leads to the recurrence

$$a_{j+1} \leq c_j a_j + \lambda\tau_j^2\sigma_j^2 + \mu\tau_j\epsilon_j^2, \tag{IV.31}$$

where a_j denotes the MSE $a_j = \mathbb{E}[\|u_j - u_\star\|^2]$, c_j, σ_j^2, ϵ_j^2 are defined in Theorem 21 and $\lambda = 2$, $\mu = 2\tau_0 + \frac{2}{l}$.

We now derive error bounds on the variance term $\sigma^2(L, \overrightarrow{N})$ and the bias term $\epsilon^2(L)$ as a function of the total number of levels L and the sample sizes $\overrightarrow{N} = \{N_0, N_1, \ldots, N_L\}$.

Lemma 15. *For a sufficiently smooth optimal control $u_\star$ and primal and dual solutions $y(u_\star)$, $p(u_\star)$, the bias term $\epsilon^2(L)$ associated to the MLMC estimator* (IV.22) *with L levels satisfies*

$$\epsilon^2(L) \leq C(u_\star)h_L^{2r+2}$$

for some constant $C(u_\star) > 0$.

Proof. Following the definition of the MLMC estimator, and using its telescoping property, we have:

$$\begin{aligned}\epsilon^2(L) &= \|\mathbb{E}[E^{\text{MLMC}}_{L,\overrightarrow{N}}[\nabla f(u_\star)]] - \mathbb{E}[\nabla f(u_\star)]\|^2 \\ &= \|\mathbb{E}[\nabla f^{h_L}(u_\star,\cdot)] - \mathbb{E}[\nabla f(u_\star,\cdot)]\|^2 \\ &\leq \mathbb{E}[\|\nabla f^{h_L}(u_\star,\cdot) - \nabla f(u_\star,\cdot)\|^2] \\ &\leq C(u_\star) h_L^{2r+2}.\end{aligned}$$

where in the last step we have used Lemma 14. □

Lemma 16. *For a sufficiently smooth optimal control $u_\star$ and primal and dual solutions $y(u_\star)$, $p(u_\star)$, the variance term $\sigma^2(L, \overrightarrow{N})$ associated to the MLMC estimator* (IV.22) *using L levels and $\overrightarrow{N} = \{N_0, N_1, \ldots, N_L\}$ sample sizes, satisfies*

$$\sigma^2(L, \overrightarrow{N}) \leq \sum_{l=0}^{L} \frac{2}{N_l} \widetilde{C}(u_\star) h_l^{2r+2}$$

for some constant $\widetilde{C}(u_\star) > 0$, namely $\widetilde{C}(u_\star) = \max\{C(u_\star)\left(1 + 2^{2r+2}\right), \frac{\mathbb{E}\left[\|\nabla f^{h_0}(u_\star)\|^2\right]}{2h_0^{2r+2}}\}$ and $C(u_\star)$ as in Lemma 15.

Proof. We use again the notation $E_{i,l}(u_\star) = \nabla f^{h_l}(u_\star, \omega_{i,l}) - \nabla f^{h_{l-1}}(u_\star, \omega_{i,l}) - \mathbb{E}[\nabla f^{h_l}(u_\star,\cdot) - \nabla f^{h_{l-1}}(v,\cdot)]$, with $\omega_{i,l}$ iid with distribution $\mathbb{P}$ on Γ and denote $V_l = \mathbb{E}\left[\|E_{i,l}(u_\star)\|^2\right]$. For $l = 0, \ldots, L$, $i = 1, \ldots, N_l$, we have

$$\begin{aligned}\sigma^2(L, \overrightarrow{N}) =& \mathbb{E}\|E^{\text{MLMC}}_{L,\overrightarrow{N}}[\nabla f(u_\star,\cdot)] - \mathbb{E}[\nabla f^{h_L}(u_\star,\cdot)]\|^2 \\ =& \mathbb{E}\|\sum_{l=0}^{L} \frac{1}{N_l} \sum_{i=1}^{N_l} E_{i,l}(u_\star)\|^2 \\ =& \sum_{l,l'} \sum_{i,i'} \frac{1}{N_l N_{l'}} \mathbb{E}\langle E_{i,l}(u_\star), E_{i',l'}(u_\star)\rangle \\ =& \sum_{l=0}^{L} \sum_{i=1}^{N_l} \frac{1}{N_l^2} \mathbb{E}[\|E_{i,l}(u_\star)\|^2] = \sum_{l=0}^{L} \frac{V_l}{N_l}\end{aligned}$$

with the following bounds for the quantity V_0

$$\begin{aligned}V_0 &\leq \mathbb{E}\left[\left\|\nabla f^{h_0}(u_\star,\cdot) - \mathbb{E}\left[\nabla f^{h_0}(u_\star,\cdot)\right]\right\|^2\right] \\ &\leq \mathbb{E}\left[\left\|\nabla f^{h_0}(u_\star,\cdot)\right\|^2\right],\end{aligned}$$

and V_l, for $l \geq 1$:

$$\begin{aligned}
V_l \leq & 2\mathbb{E}[\|\nabla f^{h_l}(u_\star,\cdot) - \nabla f(u_\star,\cdot) - \mathbb{E}[\nabla f^{h_l}(u_\star,\cdot) - \nabla f(u_\star,\cdot)]\|^2] \\
& + 2\mathbb{E}\left[\left\|\nabla f^{h_{l-1}}(u_\star,\cdot) - \nabla f(u_\star,\cdot) - \mathbb{E}\left[\nabla f^{h_{l-1}}(u_\star,\cdot) - \nabla f(u_\star,\cdot)\right]\right\|^2\right] \\
\leq & 2\left(\mathbb{E}\left[\|\nabla f^{h_l}(u_\star,\cdot) - \nabla f(u_\star,\cdot)\|^2\right] + \mathbb{E}[\|\nabla f^{h_{l-1}}(u_\star,\cdot) - \nabla f(u_\star,\cdot)\|^2]\right) \\
\leq & 2C(u_\star)\left(h_l^{2r+2} + h_{l-1}^{2r+2}\right) = 2C(u_\star)\left(1 + 2^{2r+2}\right)h_l^{2r+2}.
\end{aligned}$$

Then combining the last two bounds, we reach:

$$\sigma^2(L, \overrightarrow{N}) \leq \sum_{l=0}^{L} \frac{2}{N_l}\widetilde{C}(u_\star)h_l^{2r+2}$$

with $\widetilde{C}(u_\star) = \max\{C(u_\star)\left(1 + 2^{2r+2}\right), \frac{\mathbb{E}\left[\|\nabla f^{h_0}(u_\star)\|^2\right]}{2h_0^{2r+2}}\}$. □

From Lemmas 15 and 16 we see that the bias $\epsilon_j^2 = \epsilon^2(L_j)$ and the variance $\sigma_j^2 = \sigma^2(L_j, \overrightarrow{N}_j)$ terms go to zero if $L_j, N_{j,l} \to \infty$ as $j \to \infty$. The rate at which L_j and $N_{j,l}$ should diverge to ∞ as $j \to \infty$ should be chosen so as to minimize the total computational cost to achieve a prescribed accuracy.

We express here a final bound obtained on the MSE a_j after j iterations when using the MLSG algorithm (IV.22). Specifically, in the next Lemma we assume algebraic convergence rates for the two terms $\tau_j^2\sigma_j^2 \sim j^{-\eta_1}$ and $\tau_j\epsilon_j^2 \sim j^{-\eta_2}$ and discuss in Lemma 18 how the parameters (η_1, η_2) should be chosen, to obtain the best complexity, while guaranteeing a given MSE.

Lemma 17. *Assuming that we can choose L_j and $\overrightarrow{N}_j$ such that the bias and variance terms decay as*

$$\sum_{l=0}^{L_j} \frac{2}{N_{j,l}}\widetilde{C}(u_\star)h_l^{2r+2} \sim \sigma_0^2 j^{-\eta_1+2}, \quad \eta_1 \in]2, \frac{\tau_0 l}{2} + 1], \tag{IV.32}$$

$$C(u_\star)h_{L_j}^{2r+2} \sim \epsilon_0^2 j^{-\eta_2+1}, \quad \eta_2 \in]1, \frac{\tau_0 l}{2} + 1], \tag{IV.33}$$

with $\sigma_0^2 > 0$, $\epsilon_0^2 > 0$ constants, and taking $\tau_j = \tau_0/j$ with $\tau_0 l > 2$, then we can bound the MSE $a_j = \mathbb{E}[\|u_j - u_\star\|^2]$ for the MLSG algorithm (IV.22) *after j iterations as*

$$a_j \leq C_1(a_1)j^{-\frac{\tau_0 l}{2}} + C_2\begin{cases} j^{1-\min\{\eta_1,\eta_2\}}, & \textit{if } \eta_1 < \frac{\tau_0 l}{2} + 1 \textit{or } \eta_2 < \frac{\tau_0 l}{2} + 1 \\ \log(j)j^{-\frac{\tau_0 l}{2}}, & \textit{if } \eta_1 = \eta_2 = \frac{\tau_0 l}{2} + 1 \end{cases} \tag{IV.34}$$

with C_1 and C_2 independent of j.

Proof. We first notice that, as $N_{j,l} \geq 1$, under assumption (IV.33),

$$\sum_{l=0}^{L_j} \frac{1}{N_{j,l}} \leq L_j + 1 \leq \widetilde{c}\log(j+1), \tag{IV.35}$$

for some constant $\widetilde{c} > 0$. From (IV.31) we obtain by induction

$$\begin{aligned} a_{j+1} \leq & c_j a_j + \lambda\tau_j^2\sigma_j^2 + \mu\tau_j\epsilon_j^2 \\ \leq & c_j c_{j-1} a_{j-1} + c_j(\lambda\tau_{j-1}^2\sigma_{j-1}^2 + \mu\tau_{j-1}\epsilon_{j-1}^2) + \lambda\tau_j^2\sigma_j^2 + \mu\tau_j\epsilon_j^2 \\ \lesssim & \cdots \\ \lesssim & \underbrace{\Big(\prod_{i=1}^{j} c_i\Big)}_{=\kappa_{0,j}} a_1 + \underbrace{\sum_{i=1}^{j}(\lambda\tau_i^2\sigma_i^2 + \mu\tau_i\epsilon_i^2) \prod_{l=i+1}^{j} c_l}_{=\mathcal{B}_j} . \end{aligned} \tag{IV.36}$$

with $\lambda = 2$ and $\mu = 2\tau_0 + \frac{2}{l}$. For the first term $\kappa_{0,j}$, computing its logarithm, we have,

$$\log(\kappa_{0,j}) = \sum_{i=1}^{j} \log(1 - \frac{\tau_0 l}{2i} + Lip^2\left(2 + 8\widetilde{c}\log(i+1)\right)\frac{\tau_0^2}{i^2}) \leq \sum_{i=1}^{j} \frac{-\tau_0 l}{2i} + \widehat{c}\sum_{i=1}^{j} \frac{\log(i+1)}{i^2},$$

with $\widehat{c} = Lip^2\tau_0^2\left(\frac{2}{\log 2} + 8\widetilde{c}\right)$. Thus

$$\log(\kappa_{0,j}) \leq -\frac{\tau_0 l}{2}\log(j+1) + M, \text{ with } M = \widehat{c}\sum_{i=1}^{\infty}\frac{\log(i+1)}{i^2},$$

and $\kappa_{0,j} \lesssim j^{-\frac{\tau_0 l}{2}}$. For the second term $\mathcal{B}_j$ in (IV.36) we have:

$$\mathcal{B}_j \leq \sum_{i=1}^{j}\left(\lambda\tau_0^2\sigma_0^2 i^{-\eta_1} + \mu\tau_0\epsilon_0^2 i^{-\eta_2}\right)\underbrace{\prod_{k=i+1}^{j} c_k}_{=\kappa_{i,j}} .$$

For the term $\kappa_{i,j}$ we can proceed as follows:

$$\begin{aligned}
\log(\kappa_{i,j}) &= \sum_{k=i+1}^{j} \log\left(c_k\right) \\
&= \sum_{k=i+1}^{j} \log\left(1 - \frac{\tau_0 l}{2k} + \widehat{c}\frac{\log(k+1)}{k^2}\right) \\
&\leq \sum_{k=i+1}^{j} \left(-\frac{\tau_0 l}{2k} + \widehat{c}\frac{\log(k+1)}{k^2}\right) \\
&\leq -\frac{\tau_0 l}{2}\left(\log(j+1) - \log(i+1)\right) + M,
\end{aligned}$$

which shows that

$$\kappa_{i,j} \leq (j+1)^{-\frac{\tau_0 l}{2}}(i+1)^{\frac{\tau_0 l}{2}} \exp\left(M\right).$$

It follows, in the case $\max\{\eta_1, \eta_2\} < \frac{\tau_0 l}{2} + 1$ and using $(i+1)^{\tau_0 l/2} \leq (2i)^{\tau_0 l/2}$ for $i \geq 1$, that

$$\begin{aligned}
\mathcal{B}_j &\leq (j+1)^{-\frac{\tau_0 l}{2}} \exp\left(M\right) \sum_{i=1}^{j} \left(\lambda\tau_0^2\sigma_0^2 i^{\frac{\tau_0 l}{2}-\eta_1} + \mu\tau_0\epsilon_0^2 i^{\frac{\tau_0 l}{2}-\eta_2}\right) 2^{\tau_0 l/2} \\
&\lesssim j^{1-\min\{\eta_1,\eta_2\}},
\end{aligned}$$

whereas in the case $\eta_1 = \eta_2 = \frac{\tau_0 l}{2} + 1$

$$\begin{aligned}
\mathcal{B}_j &\leq \exp(M)(j+1)^{-\frac{\tau_0 l}{2}} \sum_{i=1}^{j} \left(\lambda\tau_0^2\sigma_0^2 i^{\frac{\tau_0 l}{2}-\eta_1} + \mu\tau_0\epsilon_0^2 i^{\frac{\tau_0 l}{2}-\eta_2}\right) 2^{\tau_0 l/2} \\
&\leq \exp(M)(j+1)^{-\frac{\tau_0 l}{2}} \left(\lambda\tau_0^2\sigma_0^2 + \mu\tau_0\epsilon_0^2\right)(1+\log(j))2^{\tau_0 l/2} \\
&\lesssim \log(j) j^{-\frac{\tau_0 l}{2}}.
\end{aligned}$$

The remaining cases where $\min\{\eta_1, \eta_2\} < \max\{\eta_1, \eta_2\} = \frac{\tau_0 l}{2} + 1$ are treated analogously. □

We introduce now some computational cost assumptions, to be able to use the MLMC results, recalled in Section IV.B.2. Let us assume that the primal and dual problems can be solved using a triangulation with mesh size h, in computational time $C_h \lesssim h^{-d\gamma}$. Here, $\gamma \in [1,3]$ is a parameter representing the efficiency of the linear solver used (e.g. $\gamma = 3$ for a direct solver and $\gamma = 1$ up to a logarithmic factor for an optimal multigrid solver), while d is the dimension of the physical space, $D \subset \mathbb{R}^d$. In the particular context presented in this work and using the results in Lemma 16, the variance V_l at level l can be bounded as

$$V_l = \mathbb{E}[\|\nabla f^{h_l}(u_\star,\cdot) - \nabla f^{h_{l-1}}(u_\star,\cdot)\|^2] \leq 2\widetilde{C}(u_\star)h_l^{2r+2} \lesssim 2^{-l(2r+2)}.$$

We can estimate the computational cost C_l to generate one realization of $\nabla f^{h_l}(u_\star,\cdot) - \nabla f^{h_{l-1}}(u_\star,\cdot)$ by:

$$C_l = Cost(\nabla f^{h_l}, \nabla f^{h_{l-1}}) \leq 2Cost(\nabla f^{h_l}) \lesssim h_l^{-\gamma d} \lesssim 2^{l\gamma d}$$

Hence the costs and variances match the assumptions in Section IV.B.2 with $q_s = 2r+2$ and $q_c = \gamma d$. For a problem in dimension $d \leq 3$ solved with an optimal solver ($\gamma = 1$ up to log-terms) and using $\mathbb{P}1$ finite elements ($r = 1$) we fall in the case $q_s > q_c$ and should expect the MLMC estimator with optimal sample sizes to achieve an optimal complexity tol^{-2} at each iteration. We show in Theorem 22 that this optimality is preserved when MLMC is used within a stochastic gradient method to solve the OCP (IV.7) when the rates η_1, η_2 are properly chosen. We start with a preliminary result.

Lemma 18. *In the case where $2r+2 > \gamma d$, where γ is such that the cost C_l of computing one realization of $\nabla f^{h_l}(u,\cdot) - \nabla f^{h_{l-1}}(u,\cdot)$ is bounded by $C_l \lesssim 2^{l\gamma d}$, and choosing*

- *the step-size $\tau_j = \frac{\tau_0}{j}$ with $\tau_0 > \frac{2}{l}$;*
- *the sequence of levels $\{L_j\}_j$ such that*

$$L_j = \left\lceil \frac{-1}{\log(2)} \log\left(\frac{1}{h_0}\left(\frac{\epsilon_0^2 j^{1-\eta_2}}{C(u_\star)}\right)^{\frac{1}{2r+2}}\right)\right\rceil,$$

for some $\eta_2 \in]1, \frac{\tau_0 l}{2}+1]$ so that the bias term in (IV.33) satisfied $\epsilon_j^2 \leq \epsilon_0^2 j^{1-\eta_2}$;

- *the sequence of sample sizes $\{N_{j,l}\}_{\{j,l\}}$ as*

$$N_{j,l} = \left\lceil \Upsilon(L_j) 2^{-l\frac{2r+2+\gamma d}{2}} j^{\eta_1 - 2}\right\rceil$$

with $\Upsilon(L_j) = 2\widetilde{C}(u_\star)\sigma_0^{-2} h_0^{2r+2} \sum_{k=0}^{L_j} 2^{-k\frac{2r+2-\gamma d}{2}}$ for some $\eta_1 \in]2, \frac{\tau_0 l}{2}+1]$ so that the variance term in (IV.32) satisfies $\sigma_j^2 \leq \sigma_0^2 j^{2-\eta_1}$;

then, using the MLMC estimator $E^{MLMC}_{L_i,\overrightarrow{N}_i}$ in (IV.22) at each iteration $i = 1, \dots, j$, the total required computational work W_j to compute u_j, is bounded by:

$$W_j \lesssim j^{\max\{\eta_1 - 2, (\eta_2 - 1)\frac{\gamma d}{2r+2}\}+1}.$$

Proof. In the case $2r+2 > \gamma d$ we know that we are in the first case of MLMC algorithm, i.e. $q_s > q_c$ what guarantees an optimal computational complexity $\widehat{W_j}$ for each MLMC estimator $E^{\mathrm{MLMC}}_{L_j,\overrightarrow{N}_j}$ (with optimal $\overrightarrow{N}_j$ and L_j), with bound

$$\begin{aligned}
\widehat{W_j} = \sum_{l=0}^{L_j} C_l N_{l,j} &\lesssim \sum_{l=0}^{L_j} 2^{l\gamma d}\left(1+\left(2^{-l\frac{2r+2+\gamma d}{2}} j^{\eta_1-2}\sum_{k=0}^{L_j} 2^{-k\frac{2r+2-\gamma d}{2}}\right)\right)\\
&\lesssim j^{\eta_1-2}\left(\sum_{k=0}^{L_j} 2^{-k\frac{2r+2-\gamma d}{2}}\right)^2 + \sum_{l=0}^{L_j} 2^{l\gamma d}\\
&\lesssim j^{\eta_1-2}\left(\sum_{k=0}^{L_j} 2^{-k\left(\frac{2r+2-\gamma d}{2}\right)}\right)^2 + 2^{\gamma d L_j}\\
&\lesssim j^{\eta_1-2} + j^{\frac{(\eta_2-1)\gamma d}{2r+2}}
\end{aligned}$$

The first inequality just recalls that $N_{l,j}$ must be an integer, so the optimal value is rounded up to the nearest integer and can not be smaller than 1. Finally, we get a total cost, for computing u_j as:

$$W_j = \sum_{i=1}^{j} \widehat{W_i} \lesssim j^{\max\{\eta_1-2,(\eta_2-1)\frac{\gamma d}{2r+2}\}+1}$$

□

We conclude the complexity analysis, investigating the optimal choice of parameters (η_1,η_2) to use in the MLSG algorithm. Such optimal choice as well as the resulting complexity are stated in the next Theorem.

Theorem 22. *With the same notation and assumptions as in Lemma 18, the choice* $\eta_1=\eta_2=\eta\geq\frac{2(2r+2)-\gamma d}{2r+2-\gamma d}$ *,* $\tau_0>\frac{2(\eta-1)}{l}$ *is optimal and the MLSG algorithm* (IV.22)*, requires a computational work* $W(tol)$ *to reach a MSE* $=O(tol^2)$*, that is bounded by:*

$$W(tol)\lesssim tol^{-2}.$$

Proof. We consider first the case $(\eta_1,\eta_2)\in\mathcal{S}_{ad}=]2,\frac{\tau_0 l}{2}+1[\times]1,\frac{\tau_0 l}{2}+1[$. From Lemma 17, we have that the MSE $a_j=\mathbb{E}[\|u_j-u_\star\|^2]$ decays as $a_j\lesssim j^{1-\min\{\eta_1,\eta_2\}}$. We want now to find the best choice $(\eta_1,\eta_2)\in\mathcal{S}_{ad}$ that minimizes the total computational work $W_j\lesssim j^{\max\{\eta_1-2,(\eta_2-1)\frac{\gamma d}{2r+2}\}+1}$ to compute u_j, under the constraint that the MSE $a_j=\mathbb{E}[\|u_j-u_\star\|^2]$ is $O\left(tol^2\right)$. Fixing $j\sim tol^{\frac{2}{1-\min\{\eta_1,\eta_2\}}}$ we are lead to the minimization problem

$$\min_{(\eta_1,\eta_2)\in\mathcal{S}_{ad}} \frac{1+\max\{\eta_1-2,(\eta_2-1)\frac{\gamma d}{2r+2}\}}{\min\{\eta_1,\eta_2\}-1} = \min_{(\eta_1,\eta_2)\in\mathcal{S}_{ad}} \frac{\phi(\eta_1,\eta_2)}{\psi(\eta_1,\eta_2)}. \tag{IV.37}$$

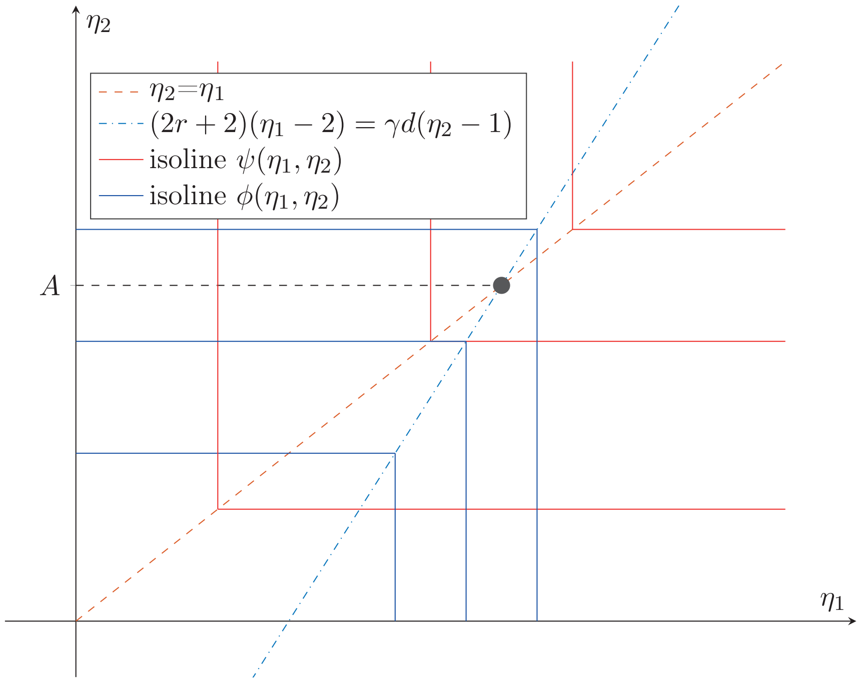


Figure IV.1 – Isocurves for problem (IV.37) w.r.t. (η_1,η_2)

Figure IV.1 shows the isolines of the function $(\eta_1,\eta_2)\mapsto\psi(\eta_1,\eta_2)$ in red (—), which are L-shaped with corners on the bisectrice $\eta_1=\eta_2$ (red dashed line (- - -)), as well as the isolines of the function $(\eta_1,\eta_2)\mapsto\phi(\eta_1,\eta_2)$ in blue (—), which have corners on the line

$$\eta_1-2=(\eta_2-1)\frac{\gamma d}{2r+2}\quad\Leftrightarrow\quad\eta_2=\underbrace{\frac{2r+2}{\gamma d}}_{=a>1}\eta_1+\underbrace{1-2\frac{2r+2}{\gamma d}}_{=b<-1}$$

(blue dashed line (-·-·)). Problem (IV.37) can be equivalently written as

$$\min_{c\in\left]0,\frac{\tau_0 l}{2}\right[}\ \min_{\substack{(\eta_1,\eta_2)\in\mathcal{S}_{ad}\\ \psi(\eta_1,\eta_2)=c}}\frac{\phi(\eta_1,\eta_2)}{c}. \tag{IV.38}$$

For the inner minimization, we see that on a isoline $\psi(\eta_1,\eta_2)=c$, the minimum of $\phi(\eta_1,\eta_2)$ is always achieved when $\eta_1=\eta_2$ (the minimizer is not unique and corresponds to a whole horizontal or vertical segment depending on whether $c>A-1$ or $c<A-1$, see Figure IV.1). Finally (IV.38) can be rewritten as

$$\min_{\eta\in\left]2,\frac{\tau_0 l}{2}+1\right[}\frac{1+\max\left\{\eta-2,(\eta-1)\frac{\gamma d}{2r+2}\right\}}{\eta-1}=\min_{\eta\in\left]2,\frac{\tau_0 l}{2}+1\right[}\max\left\{1,\frac{1+(\eta-1)\frac{\gamma d}{2r+2}}{\eta-1}\right\}, \tag{IV.39}$$

from which we see that, since $\gamma d < 2r+2$, the optimum is achieved when $\eta \geq \frac{2(2r+2)-\gamma d}{2r+2-\gamma d}$ provided τ_0 is chosen such that $\tau_0 > \frac{2(\eta-1)}{l}$. The condition $\eta \geq \frac{2(2r+2)-\gamma d}{2r+2-\gamma d}$ implies in particular that

$$\max\{\eta-2, (\eta-1)\frac{\gamma d}{2r+2}\} = \eta - 2,$$

hence $W_j \lesssim j^{\eta-1}$ where j should be chosen as small as possible while satisfying the constraints $j^{1-\eta} \leq tol^2$, which leads to $j \sim tol^{\frac{2}{1-\eta}}$ and finally $W(tol) \lesssim tol^{-2}$. The remaining case $\eta = \frac{\tau_0 l}{2} + 1$ always leads to a worse bound. □

IV.C.2. Implementation of MLSG

Here we present an effective implementation of the MLSG algorithm

The MLSG algorithm requires estimating the constants $C(u_\star)$ and $\widetilde{C}(u_\star)$. This could be done by a screening phase replacing the optimal control $u_\star$ by e.g. the initial control u_1. Hence, for instance, for M iid initial random inputs $\widetilde{\omega}_j, j = 1, \ldots, M$ distributed as $\mathbb{P}$ on Γ, we could estimate

$$\widehat{E}_l = \frac{1}{M}\sum_{j=1}^{M} \left\| \nabla f^{h_l}(u_1, \widetilde{\omega}_j) - \nabla f^{h_{l-1}}(u_1, \widetilde{\omega}_j) \right\|^2$$

and then approximate the constant $C(u_\star)$ by least squares fit of the model $\widehat{E}_l = C(u_\star)h_l^{2r+2}$. For the constant $\widetilde{C}(u_\star)$ we have always taken in our simulation $\widetilde{C}(u_\star) = C(u_\star)$. Notice that the choice of the constants $C(u_\star)$ and $\widetilde{C}(u_\star)$ does not affect the asymptotic complexity result. A good choice of such constants, however, leads to a good balance of the error contributions in the MLMC estimator, notably its bias and variance. On the same vein, the parameters σ_0 and ϵ_0 should be chosen so that the two error contributions in the recurrence (IV.31), namely $\lambda\tau_0^2\sigma_0^2 j^{-\eta}$ and $\mu\tau_0\epsilon_0^2 j^{-\eta}$ are equilibrated. For instance, one could fix $\sigma_0 = \sqrt{\frac{\mu}{\lambda\tau_0}}\epsilon_0$.

IV.D. Randomized multilevel Stochastic Gradient algorithm

We present in this section a modified version of the MLSG algorithm, namely the randomized multilevel stochastic gradient (RMLSG) algorithm, where we avoid computing a full MLMC estimator at each iteration, but we rather randomize the choice of the level used to compute the gradient direction. It corresponds to using at each iteration the randomized MLMC algorithm proposed in [RG15, RG12] (see also [Gil15]). Specifically, at each iteration j, we sample an index l_j following a suitable discrete probability measure on $\{0, \ldots, L_j\}$ with probability mass function $\{p_l^j\}_{l=0}^{L_j}$ possibly changing at each iteration,

Algorithm 4: MLSG algorithm

```
Data:
Choose $\eta \geq \frac{2(2r+2)-\gamma d}{2r+2-\gamma d}$, $\tau_0 > \frac{2(\eta-1)}{l}$, $h_0, \sigma_0, \epsilon_0$,
generate the sequence $L_j = \left\lceil \frac{-1}{\log(2)} \log\left( \frac{1}{h_0} \left( \frac{\epsilon_0^2 j^{1-\eta}}{C(u_\star)} \right)^{\frac{1}{2r+2}} \right) \right\rceil$,
compute $N_{j,l} = \left\lceil \sigma_0^{-2} j^{\eta-2} 2\widetilde{C}(u_\star) h_0^{2r+2} 2^{-l\frac{2r+2+\gamma d}{2}} \sum_{k=0}^{L_j} 2^{-k\frac{2r+2-\gamma d}{2}} \right\rceil$.
Initialize  $u = 0$;
for $j \geq 1$ do
    initialize  $\widehat{p} = 0$;
    generate $\sum_{l=0}^{L_j} N_{j,l}$ iid realizations of the random field $a_{l,i}^j = a(\cdot, \omega_{l,i}^j)$,
     $i = 1, \dots, N_{j,l}$, $l = 0, \dots, L_j$.
    for $l = 0, \dots, L_j$ do
        initialize  $\widehat{p_l} = 0$;
        for $i = 1, \dots, N_{j,l}$ do
            solve primal problem by FE on mesh $h_{l-1}$ and realization
            $a_{i,l}^j \to y^{h_{l-1}}(a_{i,l}^j, u)$
            solve dual problem by FE on mesh $h_{l-1}$ and realization
            $a_{i,l}^j \to p^{h_{l-1}}(a_{i,l}^j, y^{h_{l-1}})$
            solve primal problem by FE on mesh $h_l$ and realization
            $a_{i,l}^j \to y^{h_l}(a_{i,l}^j, u)$
            solve dual problem by FE on mesh $h_l$ and realization
            $a_{i,l}^j \to p^{h_l}(a_{i,l}^j, y^{h_l})$
            update $\widehat{p_l} = \widehat{p_l} + \frac{1}{N_{i,l}} \left( p^{h_l}(a_{i,l}^j, u) - p^{h_{l-1}}(a_{i,l}^j, u) \right)$
        end
        update $\widehat{p} = \widehat{p} + \widehat{p_l}$
    end
    $\widehat{\nabla J} = \beta u + \widehat{p}$
     $u = u - \frac{\tau_0}{j} \widehat{\nabla J}$
end
```

and we set

$$u_{j+1} = u_j - \tau_j E_{L_j, \overrightarrow{p}^j}^{\mathrm{RMLMC}}[\nabla f(u_j, \cdot)], \quad j \geq 1, \quad u_1 \in Y^{h_0}, \tag{IV.40}$$

with

$$E_{L_j, \overrightarrow{p}^j}^{\mathrm{RMLMC}}[\nabla f(u_j, \cdot)] := \frac{1}{p_{l_j}^j} \left(\nabla f^{h_{l_j}}(u_j, \omega_j) - \nabla f^{h_{l_j-1}}(u_j, \omega_j) \right), \quad l_j \sim \{p_l^j\}, \quad \omega_j \sim \mathbb{P}, \tag{IV.41}$$

where all random variables $\{l_j, \omega_j\}_{j \geq 1}$ are mutually independent. We recall now from [Gil15, RG12, RG15] that the optimal choice for the discrete probability mass function

$\{p_l^j\}_{l=0}^{L_j}$, under the condition $q_s > q_c$ is $p_l^j = 2^{-l\frac{q_s+q_c}{2}}\left(\sum_{k=0}^{L_j} 2^{-k\frac{q_s+q_c}{2}}\right)^{-1}$ which, in our setting with $q_s = 2r+2$ and $q_c = \gamma d$, and assuming $2r+2 > \gamma d$, reads

$$p_l^j = 2^{-l\frac{2r+2+\gamma d}{2}}\left(\sum_{k=0}^{L_j} 2^{-k\frac{2r+2+\gamma d}{2}}\right)^{-1}. \tag{IV.42}$$

Remark 12. *The formula* (IV.42) *allows one to take $L_j = \infty$, $\forall j \geq 1$. This leads to an unbiased estimator* (IV.41)*, and corresponds to the estimator proposed in [RG12, RG15]. However, in this work, we prefer the biased version $L_j < \infty$, $\forall j \geq 1$ with L_j suitably increasing in j, as it leads to a RMLSG algorithm with smaller variance of the computational cost (see Theorem 25 below).*

The next Lemma quantifies the bias of the estimator (IV.41) for a fixed control u.

Lemma 19. *For any $L_j \geq 0$ and any probability mass function $\left\{p_l^j\right\}_{l=0}^{L_j}$ on $\{0,\dots,L_j\}$, we have*

$$\mathbb{E}\left[E_{L_j,\vec{p}^j}^{RMLMC}[\nabla f(u,\cdot)]\right] = \mathbb{E}\left[\nabla f^{h_{L_j}}(u,\cdot)\right]. \tag{IV.43}$$

Proof. Conditioning on the value taken by the random variable l_j, we have

$$\begin{aligned}\mathbb{E}\left[E_{L_j,\vec{p}^j}^{\mathrm{RMLMC}}[\nabla f(u,\cdot)]\right] &= \sum_{l=0}^{L_j}\mathbb{E}\left[\frac{1}{p_{l_j}^j}\left(\nabla f^{h_{l_j}}(u,\cdot) - \nabla f^{h_{l_j-1}}(u,\cdot)\right)|l_j = l\right]\underbrace{\mathbb{P}(l_j = l)}_{=p_l^j}\\ &= \sum_{l=0}^{L_j}\mathbb{E}\left[\nabla f^{h_l}(u,\cdot) - \nabla f^{h_{l-1}}(u,\cdot)\right]\\ &= \mathbb{E}\left[\nabla f^{h_{L_j}}(u,\cdot)\right].\end{aligned}$$

□

So we conclude that the estimator (IV.41) has the same bias as the full MLMC estimator (IV.23).

IV.D.1. Convergence analysis

Let us denote by

$$F_j := \sigma\left(\{\omega_i\},\{l_i\},\ i = 1,\dots,j-1\right)$$

the σ-algebra generated by all the random variables $\{\omega_i\}$ and the sampled indexs $\{l_i\}$ up to iteration $j-1$. Following the same procedure as in Section IV.C.1 we first derive a recurrence relation for the error $\|u_j - u_\star\|$ at iteration j.

Theorem 23. *For any deterministic or random F_j-adapted sequence $\{L_j,\, j \geq 1\}$, $\{p^j_l,\, j \geq 1,\, l \in \{0, \dots, L_j\}\}$, with $\{L_j\}$ non decreasing, denoting by u_j the approximate control obtained at iteration j using the recursive formula* (IV.40)*, and $u_\star$ the exact control for the continuous OCP* (IV.7)*, we have:*

$$\mathbb{E}[\|u_{j+1} - u_\star\|^2 | F_j] \leq \overline{c_j} \|u_j - u_\star\|^2 + 2\tau_j^2 \overline{\sigma}_j^2 + \left(2\tau_j^2 + \frac{2\tau_j}{l}\right) \overline{\epsilon}_j^2 \tag{IV.44}$$

with $\overline{c_j} = 1 - \frac{l}{2}\tau_j + 2Lip^2\tau_j^2 \left(4 + 3\sum_{l=0}^{L_j} \frac{2}{p^j_l}\right)$

$\overline{\sigma}_j^2 = \overline{\sigma}^2(L_j, \overrightarrow{p}^j) = \mathbb{E}\left[\left\|E^{RMLMC}_{L_j, \overrightarrow{p}^j}[\nabla f(u_\star)] - \mathbb{E}[\nabla f^{h_{L_j}}(u_\star)|F_j]\right\|^2 |F_j\right]$ *the variance term*

$\overline{\epsilon}_j^2 = \overline{\epsilon}^2(L_j) = \left\|\mathbb{E}[\nabla f^{h_{L_j}}(u_\star)|F_j] - \mathbb{E}[\nabla f(u_\star)]\right\|^2$ *the squared bias term.*

Proof. Using the optimality condition

$$\mathbb{E}[\nabla f(u_\star)] = 0,$$

and the fact that the expectation of the randomized MLMC estimator for any deterministic, or random F_j-measurable $u \in U$, is

$$\mathbb{E}[E^{\text{RMLMC}}_{L_j, \overrightarrow{p}^j}[\nabla f(u)]|F_j] = \mathbb{E}[\nabla f^{h_{L_j}}(u)|F_j],$$

we can decompose the error at iteration $j+1$ as

$$\begin{aligned}
\|u_{j+1} - u_\star\|^2 &= \left\| u_j - u_\star - \tau_j \frac{1}{p^j_{l_j}} \left(\nabla f^{h_{l_j}}(u_j, \omega_j) - \nabla f^{h_{l_j - 1}}(u_j, \omega_j)\right)\right\|^2 \\
&= \Bigg\| u_j - u_\star - \tau_j \underbrace{\left(\frac{1}{p^j_{l_j}} \left(\nabla f^{h_{l_j}}(u_j, \omega_j) - \nabla f^{h_{l_j - 1}}(u_j, \omega_j)\right) - \mathbb{E}\left[\nabla f^{h_{L_j}}(u_j, \cdot)|F_j\right]\right)}_{=B(u_j)} \\
&\quad - \tau_j \underbrace{\mathbb{E}\left[\nabla f^{h_{L_j}}(u_j, \cdot) - \nabla f^{h_{L_j}}(u_\star, \cdot)|F_j\right]}_{=A} - \tau_j \underbrace{\mathbb{E}\left[\nabla f^{h_{L_j}}(u_\star, \cdot) - \nabla f(u_\star, \cdot)|F_j\right]}_{=C} \Bigg\|^2 \\
&= \|u_j - u_\star\|^2 + \tau_j^2 \|A\|^2 + \tau_j^2 \|B(u_j)\|^2 + \tau_j^2 \|C\|^2 \\
&\quad - 2\tau_j \langle u_j - u_\star, A\rangle - 2\tau_j \langle u_j - u_\star, B(u_j)\rangle - 2\tau_j \langle u_j - u_\star, C\rangle \\
&\quad + 2\tau_j^2 \langle A, B(u_j)\rangle + 2\tau_j^2 \langle A, C\rangle + 2\tau_j^2 \langle B(u_j), C\rangle
\end{aligned}$$

Taking conditional expectation w.r.t. the σ-algebra F_j and using the fact that $\mathbb{E}[B(u_j)|F_j] = 0$, which implies $\mathbb{E}[\langle u_j - u_\star, B(u_j)\rangle |F_j] = \mathbb{E}[\langle A, B(u_j)\rangle|F_j] = \mathbb{E}[\langle C, B(u_j)\rangle|F_j] = 0$, we get

$$\begin{aligned}\mathbb{E}\left[\|u_{j+1}-u_\star\|^2|F_j\right] =&\|u_j-u_\star\|^2+\tau_j^2\mathbb{E}[\|A\|^2|F_j]+\tau_j^2\mathbb{E}[\|B(u_j)\|^2|F_j]+\tau_j^2\mathbb{E}[\|C\|^2|F_j]\\ &-2\tau_j\langle u_j-u_\star,\mathbb{E}[A|F_j]\rangle-2\tau_j\langle u_j-u_\star,\mathbb{E}[C|F_j]\rangle+2\tau_j^2\mathbb{E}[\langle A,C\rangle|F_j]\\ \leq&\|u_j-u_\star\|^2+\tau_j^2\mathbb{E}[\|A\|^2|F_j]+\tau_j^2\mathbb{E}[\|B(u_j)\|^2|F_j]+\tau_j^2\mathbb{E}[\|C\|^2|F_j]\\ &-\tau_j l\|u_j-u_\star\|^2+\frac{l}{2}\tau_j\|u_j-u_\star\|^2+\frac{2}{l}\tau_j\mathbb{E}[\|C\|^2|F_j]\\ &+\tau_j^2\mathbb{E}[\|A\|^2|F_j]+\tau_j^2\mathbb{E}[\|C\|^2|F_j]\\ \leq&\left(1-\frac{l}{2}\tau_j+2Lip^2\tau_j^2\right)\|u_j-u_\star\|^2+\tau_j^2\mathbb{E}[\|B(u_j)\|^2|F_j]+\left(2\tau_j^2+\frac{2\tau_j}{l}\right)\mathbb{E}[\|C\|^2|F_j],\end{aligned}$$

where we have exploited the Lipschitz continuity and strong convexity of $f^{h_{L_j}}$ to bound $\mathbb{E}[\|A\|^2|F_j]\leq Lip^2\|u_j-u_\star\|^2$ as well as $\langle u_j-u_\star,\mathbb{E}[A|F_j]\rangle\geq\frac{l}{2}\|u_j-u_\star\|^2$ (see also the proof of Theorem 21). Splitting the term $B(u_j)$ as $(B(u_j)-B(u_\star))+B(u_\star)$, we get:

$$\begin{aligned}\mathbb{E}\left[\|u_{j+1}-u_\star\|^2|F_j\right]\leq&\left(1-\frac{l}{2}\tau_j+2Lip^2\tau_j^2\right)\|u_j-u_\star\|^2+2\tau_j^2\mathbb{E}[\|B(u_j)-B(u_\star)\|^2|F_j]\\ &+2\tau_j^2\underbrace{\mathbb{E}[\|B(u_\star)\|^2|F_j]}_{=\bar{\sigma}_j^2}+\left(2\tau_j^2+\frac{2\tau_j}{l}\right)\underbrace{\mathbb{E}[\|C\|^2|F_j]}_{=\bar{\epsilon}_j^2}.\end{aligned}$$

We can further split the term $B(u_j)-B(u_\star)$ in 3 parts, and use the Lipschitz continuity:

$$\begin{aligned}\|B(u_j)-B(u_\star)\|^2\leq&3\left\|\frac{1}{p_{l_j}^j}\left(\nabla f^{h_{l_j}}(u_j,\cdot)-\nabla f^{h_{l_j}}(u_\star,\cdot)\right)\right\|^2\\ &+3\left\|\frac{1}{p_{l_j}^j}\left(\nabla f^{h_{l_j-1}}(u_j,\cdot)-\nabla f^{h_{l_j-1}}(u_\star,\cdot)\right)\right\|^2+3\left\|\mathbb{E}[\nabla f^{h_{L_j}}(u_j,\cdot)-\nabla f^{h_{L_j}}(u_\star,\cdot)|F_j]\right\|^2,\end{aligned}$$

so that its conditional expectation reads

$$\mathbb{E}[\|B(u_j)-B(u_\star)\|^2|F_j]\leq3Lip^2\left(1+\sum_{l=0}^{L_j}\frac{2}{p_l^j}\right)\|u_j-u_\star\|^2.$$

Finally, we obtain:

$$\mathbb{E}\left[\|u_{j+1}-u_\star\|^2|F_j\right]\leq\left(1-\frac{l}{2}\tau_j+2Lip^2\tau_j^2\left(4+3\sum_{l=0}^{L_j}\frac{2}{p_l^j}\right)\right)\|u_j-u_\star\|^2+2\tau_j^2\bar{\sigma}_j^2+\left(2\tau_j^2+\frac{2\tau_j}{l}\right)\bar{\epsilon}_j^2. \tag{IV.45}$$

□

From now on, we consider only deterministic sequences $\{L_j\}, \{p_l^j\}$, i.e. chosen in advance and not adaptively during the algorithm. In this case the quantities $\overline{\sigma}_j$ and $\overline{\epsilon}_j$ defined in Theorem 23 are deterministic as well. From Theorem 23, taking the full expectation $\mathbb{E}[\cdot]$ in (IV.45) leads to the recurrence

$$a_{j+1} \leq \overline{c_j} a_j + \lambda \tau_j^2 \overline{\sigma}_j^2 + \mu \tau_j \overline{\epsilon}_j^2 \tag{IV.46}$$

where a_j denotes the MSE $a_j = \mathbb{E}[\|u_j - u_\star\|^2]$, $\overline{c_j}$, $\overline{\sigma}_j^2$, $\overline{\epsilon}_j^2$ are defined in Theorem 23 and $\lambda = 2$, $\mu = 2\tau_0 + \frac{2}{l}$ as in (IV.31). We now derive bounds on the bias term $\overline{\epsilon}^2(L)$ and the variance term $\overline{\sigma}^2(L, \{p_l\}_{l=0}^L)$ as a function of the total number of levels L and the probability mass function (pmf) $\{p_l\}_{l=0}^L$ on $\{0, \ldots, L\}$.

Lemma 20. *For a sufficiently smooth optimal control $u_\star$ and primal and dual solutions $y(u_\star)$, $p(u_\star)$, the bias term $\overline{\epsilon}^2(L)$ associated to the randomized MLMC estimator* (IV.41) *with L levels satisfies*

$$\overline{\epsilon}^2(L) \leq C(u_\star) h_L^{2r+2}.$$

with $C(u_\star)$ as in Lemma 15.

Proof. Since, from Lemma 19, we have $\mathbb{E}\left[E_{L,\overrightarrow{p}}^{\mathrm{RMLMC}}[\nabla f(u,\cdot)]\right] = \mathbb{E}\left[\nabla f^{h_L}(u,\cdot)\right]$, the proof follows verbatim that of Lemma 15. □

Lemma 21. *For a sufficiently smooth optimal control $u_\star$ and primal and dual solutions $y(u_\star)$, $p(u_\star)$, the variance term $\overline{\sigma}^2(L, \{p_l\}_{l=0}^L)$ associated to the randomized MLMC estimator* (IV.41) *using L levels and the pmf $\{p_l\}_{l=0}^L$, satisfies*

$$\overline{\sigma}^2(L, \{p_l\}_{l=0}^L) \leq \sum_{l=0}^L 2\widetilde{C}(u_\star) \frac{h_l^{2r+2}}{p_l}$$

with $\widetilde{C}(u_\star)$ as in Lemma 16. If the pmf $\{p_l\}_{l=0}^L$ is chosen optimally as $p_l \propto 2^{-l\frac{2r+2+\gamma d}{2}}$, and $2r+2 > \gamma d$ then

$$\overline{\sigma}^2(L, \{p_l\}_{l=0}^L) = O(1), \quad \text{w.r.t. } L.$$

Proof. For the first part of the Lemma, we start showing that

$$\overline{\sigma}^2(L, \{p_l\}_{l=0}^L) \leq \sum_{k=0}^L \frac{V_k}{p_k}$$

with $V_k = \mathbb{E}\left[\left\|\nabla f^{h_k}(u_\star,\cdot) - \nabla f^{h_{k-1}}(u_\star,\cdot)\right\|^2\right]$. We can write, following [Gil15]

$$\begin{aligned}
\overline{\sigma}^2(L,\{p_l\}_{l=0}^L) =&\mathbb{E}\left[\left\|E_{L,\overrightarrow{p}}^{\text{RMLMC}}[\nabla f(u_\star,\cdot)] - \mathbb{E}\left[\nabla f^{h_L}(u_\star,\cdot)\right]\right\|^2\right]\\
=&\mathbb{E}\left[\left\|\frac{1}{p_{l_R}}\left(\nabla f^{h_{l_R}}(u_\star,\cdot) - \nabla f^{h_{l_R-1}}(u_\star,\cdot)\right)\right\|^2\right] - \left\|\mathbb{E}\left[\nabla f^{h_L}(u_\star,\cdot)\right]\right\|^2\\
\leq&\mathbb{E}\left[\left\|\frac{1}{p_{l_R}}\left(\nabla f^{h_{l_R}}(u_\star,\cdot) - \nabla f^{h_{l_R-1}}(u_\star,\cdot)\right)\right\|^2\right]\\
\leq&\sum_{k=0}^{L}\frac{V_k}{p_k}.
\end{aligned}$$

The variance terms V_k can be bounded as in Lemma 16 as

$$V_k \leq 2\widetilde{C}(u_\star)h_k^{2r+2}, \quad k = 0,\dots,L$$

leading to the final result

$$\overline{\sigma}^2(L,\{p_l\}_{l=0}^L) \leq \sum_{k=0}^{L}\frac{2}{p_k}\widetilde{C}(u_\star)h_k^{2r+2}.$$

Replacing the probability mass function $p_l \propto 2^{-l\frac{2r+2+\gamma d}{2}}$, as defined in (IV.42), we get:

$$\begin{aligned}
\overline{\sigma}^2(L,\{p_l\}_{l=0}^L) \leq \sum_{l=0}^{L}2\widetilde{C}(u_\star)\frac{h_l^{2r+2}}{p_l} &= \sum_{l=0}^{L}2\widetilde{C}(u_\star)h_0^{2r+2}2^{-l(2r+2)}2^{l(\frac{2r+2+\gamma d}{2})}\frac{2^{-(L+1)\frac{2r+2+\gamma d}{2}}-1}{2^{-\frac{2r+2+\gamma d}{2}}-1}\\
&\leq 2\widetilde{C}(u_\star)h_0^{2r+2}\sum_{l=0}^{L}2^{-l\frac{2r+2-\gamma d}{2}}\left(1-2^{-\frac{2r+2+\gamma d}{2}}\right)^{-1}\\
&= O(1).
\end{aligned}$$

Since $2r+2-\gamma d > 0$ and the series is convergent. □

Analogously to the non-randomized MLSG algorithm, we enforce an algebraic decrease of the bias as a function of j, i.e.

$$C(u_\star)h_0^{2r+2}2^{-L_j(2r+2)} \sim \overline{\epsilon}_0^2 j^{1-\eta}, \quad \text{for some } \eta > 1. \tag{IV.47}$$

Under the further condition $\eta < \frac{3(2r+2)+\gamma d}{2r+2+\gamma d}$, we can obtain a bound on the MSE $a_j = \mathbb{E}[\|u_j - u_\star\|^2]$ for the RMLSG algorithm, analogous to the one stated in Lemma 17 for the non-randomized version.

Lemma 22. *Assuming that we can choose L_j such that the bias term decays as*

$$C(u_\star)h_{L_j}^{2r+2} \sim \overline{\epsilon}_0^2 j^{-\eta+1}, \quad \eta \in \left]1, \frac{3(2r+2)+\gamma d}{2r+2+\gamma d}\right[, \tag{IV.48}$$

with $\overline{\epsilon}_0^2 > 0$ constant, taking $\tau_j = \tau_0/j$ with $\tau_0 l > 2$, and the probability mass function $\{p_l^j\}$ as in (IV.42)*, then we can bound the MSE $a_j = \mathbb{E}[\|u_j - u_\star\|^2]$ for the RMLSG algorithm* (IV.40) *after j iterations as*

$$a_j \leq C_1(a_1) j^{-\frac{\tau_0 l}{2}} + C_2 j^{1-\min\{2,\eta\}} \tag{IV.49}$$

with $C_1(a_1)$ and C_2 independent of j.

Proof. We notice that we can bound $\sum_{l=0}^{L_j}\left(p_l^j\right)^{-1}$ appearing in the constant $\overline{c_j}$ of Theorem 23, by

$$\sum_{l=0}^{L_j}\left(p_l^j\right)^{-1} \leq \widetilde{c} j^{\theta}, \tag{IV.50}$$

for some constant $\widetilde{c} > 0$ and $\theta = \frac{\eta-1}{2}\left(1 + \frac{\gamma d}{2r+2}\right)$. Indeed, denoting by $P_j = \sum_{l=0}^{L_j} 2^{-l\frac{2r+2+\gamma d}{2}} = O(1)$, we have

$$\begin{aligned}\sum_{l=0}^{L_j}\left(p_l^j\right)^{-1} &= \sum_{l=0}^{L_j} 2^{l\frac{2r+2+\gamma d}{2}} P_j \\ &\lesssim 2^{(L_j+1)\frac{2r+2+\gamma d}{2}} \\ &\lesssim j^{\frac{\eta-1}{2r+2}\frac{2r+2+\gamma d}{2}} \\ &\lesssim j^{\theta}.\end{aligned}$$

The condition $\eta < \frac{3(2r+2)+\gamma d}{2r+2+\gamma d}$ is equivalent to $\theta < 1$, what makes the serie of general term $\left\{\sum_{l=0}^{L_j}\left(p_l^j\right)^{-1} j^{-2}\right\}_j$ summable. From (IV.46) we obtain by induction

$$\begin{aligned}a_{j+1} \leq& \overline{c_j} a_j + \lambda\tau_j^2\overline{\sigma}_j^2 + \mu\tau_j\overline{\epsilon}_j^2 \\ \leq& \overline{c_j}\ \overline{c_{j-1}} a_{j-1} + \overline{c_j}(\lambda\tau_{j-1}^2\overline{\sigma}_{j-1}^2 + \mu\tau_{j-1}\overline{\epsilon}_{j-1}^2) + \lambda\tau_j^2\overline{\sigma}_j^2 + \mu\tau_j\overline{\epsilon}_j^2 \\ \lesssim& \cdots \\ \lesssim& \underbrace{\left(\prod_{i=1}^{j}\overline{c_i}\right)}_{=\kappa_{0,j}} a_1 + \underbrace{\sum_{i=1}^{j}(\lambda\tau_i^2\overline{\sigma}_i^2 + \mu\tau_i\overline{\epsilon}_i^2)\prod_{l=i+1}^{j}\overline{c_l}}_{=\mathcal{B}_j}.\end{aligned} \tag{IV.51}$$

with $\lambda = 2$ and $\mu = 2\tau_0 + \frac{2}{l}$. For the first term $\kappa_{0,j}$, computing its logarithm, we have,

$$\log(\kappa_{0,j}) \leq \sum_{i=1}^{j} \log\left(1 - \frac{\tau_0 l}{2i} + \tau_0^2 Lip^2 \left(8 + 12\widetilde{c} i^{\theta}\right) i^{-2}\right) \leq \sum_{i=1}^{j} \frac{-\tau_0 l}{2i} + \widehat{c}\sum_{i=1}^{j} i^{\theta-2},$$

with $\widehat{c} = Lip^2\,(8 + 12\widetilde{c})\,\tau_0^2$. Hence,

$$\log(\kappa_{0,j}) \leq -\frac{\tau_0 l}{2}\log(j+1) + M', \ \text{ with } \ M' = \widehat{c}\sum_{i=1}^{\infty} i^{\theta-2} < +\infty,$$

which implies $\kappa_{0,j} \lesssim j^{-\frac{\tau_0 l}{2}}$. For the second term $\mathcal{B}_j$ in (IV.51) we have:

$$\mathcal{B}_j \leq \sum_{i=1}^{j} \left(\lambda \tau_0^2 \overline{\sigma}_0^2 i^{-2} + \mu \tau_0 \overline{\epsilon}_0^2 i^{-\eta}\right) \underbrace{\prod_{k=i+1}^{j} \overline{c_k}}_{=\kappa_{i,j}}.$$

For the term $\kappa_{i,j}$ we can proceed as follows:

$$\begin{aligned}
\log(\kappa_{i,j}) &= \sum_{k=i+1}^{j} \log\left(\overline{c_k}\right) \\
&\leq \sum_{k=i+1}^{j} \log\left(1 - \frac{\tau_0 l}{2k} + \widehat{c}\frac{k^{\theta}}{k^2}\right) \\
&\leq \sum_{k=i+1}^{j} \left(-\frac{\tau_0 l}{2k} + \widehat{c}\frac{k^{\theta}}{k^2}\right) \\
&\leq -\frac{\tau_0 l}{2}\left(\log(j+1) - \log(i+1)\right) + M',
\end{aligned}$$

which implies

$$\kappa_{i,j} \leq (j+1)^{-\frac{\tau_0 l}{2}} (i+1)^{\frac{\tau_0 l}{2}} \exp\left(M'\right),$$

and the final bound on B_j:

$$\begin{aligned}
\mathcal{B}_j &\leq (j+1)^{-\frac{\tau_0 l}{2}} \exp\left(M'\right) \sum_{i=1}^{j} \left(\lambda \tau_0^2 \overline{\sigma}_0^2 i^{\frac{\tau_0 l}{2}-2} + \mu \tau_0 \overline{\epsilon}_0^2 i^{\frac{\tau_0 l}{2}-\eta}\right) 2^{\tau_0 l/2} \\
&\lesssim j^{1-\min\{2,\eta\}}.
\end{aligned}$$

□

We are now ready to state the final complexity result for the RMLSG algorithm.

Theorem 24. *In the case $2r + 2 > \gamma d$ with the same notation and assumptions as in*

Lemma 22, if the parameter η satisfies $\eta \in \left]1, \frac{3(2r+2)+\gamma d}{2r+2+\gamma d}\right[$, and $\tau_0 > \frac{2}{l}$, then the expected computational work $W(tol)$ of the RMLSG algorithm (IV.40) *to reach a MSE $O(tol^2)$, is bounded by:*

$$\mathbb{E}[W(tol)] \lesssim tol^{\min\left\{-2, \frac{-2}{\eta-1}\right\}}. \tag{IV.52}$$

In particular, if we choose $\eta \in \left[2, \frac{3(2r+2)+\gamma d}{2r+2+\gamma d}\right[$, we reach the optimal complexity of $\mathbb{E}[W(tol)] \lesssim tol^{-2}$.

Proof. The expected computational work of RMLSG up to iteration j, namely W_j, can be bounded by

$$\begin{aligned}
\mathbb{E}[W_j] = \sum_{i=1}^{j} \mathbb{E}[C_{l_i}] &= \sum_{i=1}^{j}\sum_{k=0}^{L_i} C_k p_k^i \\
&\lesssim \sum_{i=1}^{j}\sum_{k=0}^{L_i} 2^{k\gamma d} 2^{-k\frac{2r+2+\gamma d}{2}} \left(\sum_{p=0}^{L_j} 2^{-p\frac{2r+2+\gamma d}{2}}\right)^{-1} \\
&= \sum_{i=1}^{j}\sum_{k=0}^{L_i} 2^{-k\frac{2r+2-\gamma d}{2}} \left(\sum_{p=0}^{L_j} 2^{-p\frac{2r+2+\gamma d}{2}}\right)^{-1} \\
&\lesssim O(j).
\end{aligned}$$

Taking now $j \sim tol^{\frac{2}{1-\min\{2,\eta\}}}$ to achieve $a_j \lesssim tol^2$ we finally obtain $\mathbb{E}[W(tol)] \lesssim tol^{\frac{2}{1-\min\{2,\eta\}}}$ which can be equivalently rewritten as

$$\mathbb{E}[W(tol)] \lesssim tol^{\min\left\{-2, \frac{-2}{\eta-1}\right\}}.$$

□

The previous Theorem shows that the RMLSG algorithm achieves the optimal complexity $\mathbb{E}[W(tol)] \lesssim tol^{-2}$ (in terms of expected computational cost versus MSE), for all $\eta \in \left[2, \frac{3(2r+2)+\gamma d}{2r+2+\gamma d}\right[$. It is worth looking also at the variance of the computational work, beside its expected value. The next Lemma shows that the choice $\eta = 2$ is optimal in the sense that it minimizes the variance of the cost among all $\eta \in \left[2, \frac{3(2r+2)+\gamma d}{2r+2+\gamma d}\right[$, at least in the case $2r+2 < 3\gamma d$, and leads to a coefficient of variation $\varrho(tol) = \frac{\sqrt{\mathbb{V}ar[W(tol)]}}{\mathbb{E}[W(tol)]}$ that goes asymptotically to zero.

Theorem 25. *Let $W(tol)$ be the computational cost to reach a MSE $= O(tol^2)$ by the RMLSG algorithm* (IV.40)*, and denote by $\varrho(tol)$ the coefficient of variation of $W(tol)$,* *namely*

$$\varrho(tol) = \frac{\sqrt{\mathbb{V}ar[W(tol)]}}{\mathbb{E}[W(tol)]}.$$

Assuming that the computational cost C_l *of computing one realization of* $\nabla f^{h_l}(u,\cdot) - \nabla f^{h_{l-1}}(u,\cdot)$ *can be bounded as* $\underline{c_c}2^{l\gamma d} \leq C_l \leq \overline{c_c}2^{l\gamma d}$ *for some* $\underline{c_c}, \overline{c_c} > 0$*, then if* $2r+2 \geq 3\gamma d$*,*

$$\lim_{tol\to 0} \varrho(tol) = 0, \quad \forall \eta \in \left[2, \frac{3(2r+2)+\gamma d}{2r+2+\gamma d}\right[.$$

On the other hand, if $\gamma d < 2r+2 < 3\gamma d$*,*

$$\lim_{tol\to 0} \varrho(tol) = 0, \quad \forall \eta \in \left[2, \frac{3\gamma d + (2r+2)}{3\gamma d - (2r+2)}\right[.$$

Moreover, $\varrho(tol)$ *is minimized for* $\eta = 2$*, for which*

$$\varrho(tol) \lesssim tol^{\frac{3(2r+2-\gamma d)}{2(2r+2)}}.$$

Proof. Let us start computing the variance of the computational cost W_j, after j iterations

$$\begin{aligned}
\mathbb{V}ar\left[W_j\right] := \mathbb{V}ar\left[\sum_{i=1}^{j} C_{l_i}\right] &= \sum_{i=1}^{j} \mathbb{V}ar\left[C_{l_i}\right] \\
&\leq \sum_{i=1}^{j} \mathbb{E}\left[C_{l_i}^2\right] \\
&= \sum_{i=1}^{j}\sum_{l=0}^{L_i} C_l^2 p_l^i \\
&\lesssim \sum_{i=1}^{j}\sum_{l=0}^{L_i} 2^{2l\gamma d} 2^{-l\frac{2r+2+\gamma d}{2}} \\
&\lesssim \sum_{i=1}^{j}\sum_{l=0}^{L_i} 2^{-l\frac{2r+2-3\gamma d}{2}}.
\end{aligned}$$

Observe moreover, that under the assumption $C_l \sim 2^{l\gamma d}$, we have

$$\begin{aligned}
\mathbb{E}\left[W_j\right] &= \sum_{i=1}^{j}\sum_{l=0}^{L_i} C_l p_l^i \\
&\geq \sum_{i=1}^{j} \frac{1}{P_j} \sum_{l=0}^{L_i} \underline{c_c} 2^{2l\gamma d} 2^{-l\frac{2r+2+\gamma d}{2}} \\
&\geq \frac{\underline{c_c}}{P_\infty} j.
\end{aligned}$$

If $2r+2 > 3\gamma d$, the series $\{\sum_{l=0}^{L_i} 2^{-l\frac{2r+2-3\gamma d}{2}}\}_i$ is convergent, we end up with

$$\mathbb{V}ar[W_j] \lesssim j, \quad \forall \eta \in \left[2, \frac{3(2r+2)+\gamma d}{2r+2+\gamma d}\right[.$$

In this case, the squared coefficient of variation $\frac{\mathbb{V}ar[W_j]}{\mathbb{E}[W_j]^2} \lesssim j^{-1}$ which implies $\varrho^2(tol) \lesssim tol^2$, $\forall \eta \in \left[2, \frac{3(2r+2)+\gamma d}{2r+2+\gamma d}\right[$.
If, instead, $2r+2 = 3\gamma d$, then we have

$$\mathbb{V}ar[W_j] \lesssim \sum_{i=1}^{j} L_i \leq \sum_{i=1}^{j} \log\left(i^{\frac{\eta-1}{2r+2}}\right) \leq \frac{\eta-1}{2r+2}(j+1)\log(j+1)$$

and then

$$\varrho(tol)^2 = \frac{\mathbb{V}ar[W(tol)]}{\mathbb{E}[W(tol)]^2} \lesssim \frac{\eta-1}{2r+2} tol^2 \log(tol^{-1})$$

which is minimized for $\eta = 2$.
Finally, when $2r+2 < 3\gamma d$, we have

$$\mathbb{V}ar[W_j] \lesssim \sum_{i=1}^{j} 2^{L_i \frac{3\gamma d-(2r+2)}{2}} \lesssim \sum_{i=1}^{j} i^{\frac{\eta-1}{2r+2}\frac{3\gamma d-(2r+2)}{2}} \lesssim j^{\frac{\eta-1}{2r+2}\frac{3\gamma d-(2r+2)}{2}+1}$$

and we derive

$$\varrho(tol)^2 = \frac{\mathbb{V}ar[W(tol)]}{\mathbb{E}[W(tol)]^2} \lesssim tol^{-\frac{\eta-1}{2r+2}(3\gamma d-(2r+2))+2},$$

which shows that $\lim_{tol\to 0} \varrho(tol) = 0$, $\forall \eta \in [2, \frac{3\gamma d+(2r+2)}{3\gamma d-(2r+2)}[$. In particular, $\varrho(tol)$ is minimized for $\eta = 2$, what finishes the proof. □

We present in the following Section a description of the RMLSG algorithm.

IV.D.2. Implementation of the RMLSG algorithm

In this section, we present an effective implementation of the RMLSG algorithm.

Algorithm 5 requires estimating the constant $C(u_\star)$ which can be done in the same way as proposed in Section IV.C.2. Notice that overall this randomized version of the MLSG algorithm has less parameters to tune than the non-randomized one.

Algorithm 5: Randomized MLSG algorithm

```
Data:
Choose $\tau_0 > \frac{2}{l}, h_0, \bar{\epsilon}_0$,
generate the sequence $L_j = \left\lceil \frac{-1}{\log(2)} \log\left( \frac{1}{h_0} \left( \frac{\bar{\epsilon}_0^2 j^{-1}}{C(u_\star)} \right)^{\frac{1}{2r+2}} \right) \right\rceil \quad j \geq 1$,
compute $p_l^j = 2^{-l\frac{2r+2+\gamma d}{2}} \left( \sum_{k=0}^{L_j} 2^{-k\frac{2r+2+\gamma d}{2}} \right)^{-1} \quad j \geq 1, \quad l = 0, \dots, L_j$.
initialize $u = 0$;
for $j \geq 1$ do
    generate one iid realization of the random field $a_j = a(\cdot, \omega_j)$, $j \geq 1$.
    sample $l_j \sim \{p_l^j\}_{l=0}^{L_j}$ on $\{0, \dots, L_j\}$
    solve primal problem by FE on mesh $h_{l_j-1}$ and realization
    $a_j \to y^{h_{l_j-1}}(a_j, u)$
    solve dual problem by FE on mesh $h_{l_j-1}$ and realization
    $a_j \to p^{h_{l_j-1}}(a_j, y^{h_{l_j-1}})$
    solve primal problem by FE on mesh $h_{l_j}$ and realization
    $a_j \to y^{h_{l_j}}(a_j, u)$
    solve dual problem by FE on mesh $h_{l_j}$ and realization
    $a_j \to p^{h_{l_j}}(a_j, y^{h_{l_j}})$
    $\widehat{\nabla J} = \beta u + \frac{1}{p_{l_j}^j} \left( p^{h_{l_j}}(a_j, u) - p^{h_{l_j-1}}(a_j, u) \right)$
    $u = u - \frac{\tau_0}{j} \widehat{\nabla J}$
end
```

IV.E. Numerical results

IV.E.1. Problem setting

In this section we verify the assertions on the order of convergence and computational complexity stated in Lemmas 17, 22 and Theorem 22, 24 for the MLSG Algorithm 4 and the RMLSG Algorithm 5, respectively. For this purpose, we consider the optimal control problem (IV.7) in the domain $D = (0,1)^2$ with $g = 1$ and the following random diffusion coefficient:

$$a(x_1, x_2, \xi) = 1 + \exp\left(var\left(\xi_1 \cos(1.1\pi x_1) + \xi_2 \cos(1.2\pi x_1) + \xi_3 \sin(1.3\pi x_2) + \xi_4 \sin(1.4\pi x_2) \right) \right), \tag{IV.53}$$

with $(x_1, x_2) \in D$, $var = exp(-1.125)$ and $\xi = (\xi_1, \dots, \xi_4)$ with $\xi_i \overset{iid}{\sim} \mathcal{U}([-1,1])$ (this test case is taken from [LG17]). We have chosen $\beta = 10^{-4}$ as the price of energy (regularization parameter) in the objective functional. For the FE approximation, we have considered a structured triangular grid of mesh size h where each side of the domain D is divided into

j: iteration	$\{1,\dots,3\}$	$\{4,\dots,15\}$	$\{16,\dots,63\}$	$\{64,\dots,120\}$
L_j	0	1	2	3
h_{L_j}	2^{-3}	2^{-4}	2^{-5}	2^{-6}

Table IV.1 – One example of level refinement over the iterations for Algorithm 4

$1/h$ sub-intervals and used piece-wise linear finite elements (i.e. $r = 1$). All calculations have been performed using the FE library Freefem++ [Hec12].

IV.E.2. Reference solution

In order to compute one reference solution u_{ref}, we used a tensorized Gauss-Legendre quadrature formula with $q = 5$ Gauss-Legendre knots in each of the 4 random variables $\xi_1, \dots, \xi_4$ in (IV.53), hence $n = 5^4$ knots in total, in order to approximate the expectation $\mathbb{E}$ in the objective functional. We discretized the OCP (IV.7), using a FE method with $\mathbb{P}_1$ elements (i.e. $r = 1$), over a regular triangulation of the domain D, with a discretization parameter $h = 2^{-7}$. To compute one optimal control, we used a full gradient strategy, (requiring at each iteration to solve 2×5^4 discretized PDE) up to 20 iterations, using an adaptive (optimal in our quadratic setting) step-size. We reached a final gradient norm of $\|\nabla \widehat{J}(u_{20})\| = 6.54276e - 12$ and a final difference between two consecutive controls of $\|u_{20} - u_{19}\| = 5.05678e - 09$.

IV.E.3. MLSG algorithm

In order to assess the convergence rate of the MLSG Algorithm 4 and its computational complexity, we run 10 independent realizations of the MLSG algorithm, up to 120 iterations, using a step size $\tau_j = \tau_0/(j + 10)$, and the following parameters: $\tau_0 = 2/\beta$, $h_0 = 2^{-3}$, $\epsilon_0 = \sqrt{C(u_\star)h_0^{2r+2}}$, $\sigma_0 = \sqrt{\left(2\tau_0 + \frac{2}{l}\right)\frac{\epsilon_0^2}{2\tau_0}}$, $\eta = \frac{2(2r+2)-\gamma d}{2r+2-\gamma d}$, $r = 1$, $d = 2$, $\gamma = 1$, $l = 2\beta$ (see Lemma 11). The constant $C(u_\star)$ has been estimated in [MN18] for the same test case. Here we have taken $C(u_\star) = 0.5$. These parameters have been used in Algorithm 4 to determine the levels L_j and samples per level $N_{j,l}$, at each iteration. We report in Table IV.1 the levels L_j and the corresponding mesh sizes over the iterations In Figure IV.2, we plot the mean error on the control, $\mathbb{E}[\|u_j - u_{ref}\|]$, averaged over the 10 repetitions of the MLSG procedure, versus the iteration counter in log scale. We verify a slope of -1.09, which is consistent with the result $MSE = O(j^{1-\eta})$ stated in Lemma 17 with $\eta = 3$. Figure IV.3 shows the estimated mean error, averaged over the 10 repetitions, versus the computational cost model $W_j = \sum_{i=0}^{j}\sum_{l=0}^{L_i} 2^{l\gamma d} N_{l,i}$, which confirms the complexity result of Theorem 22.

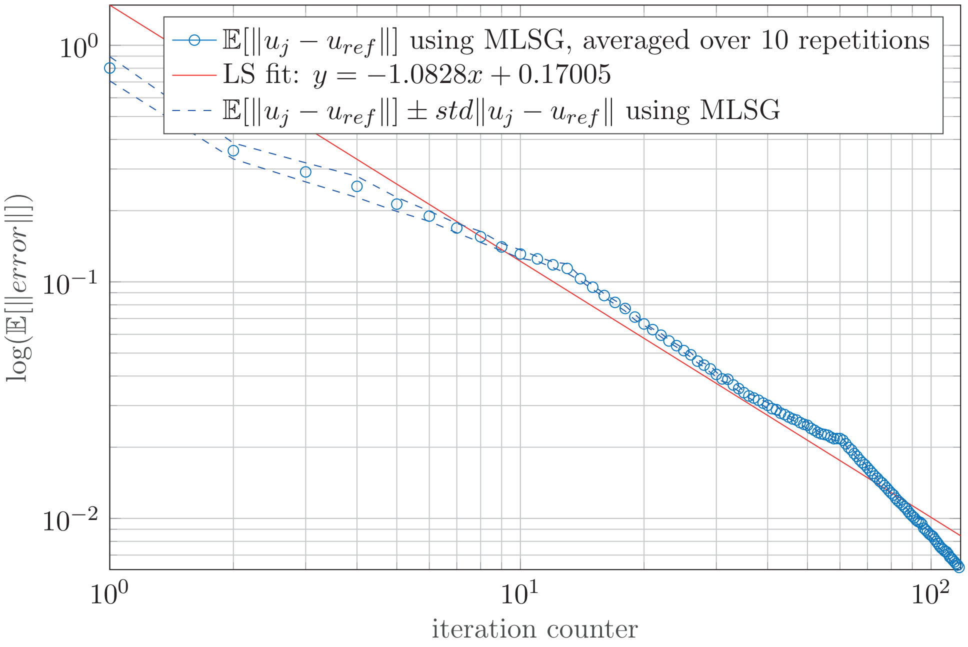


Figure IV.2 – Mean error vs iteration counter for the MLSG Algorithm 4

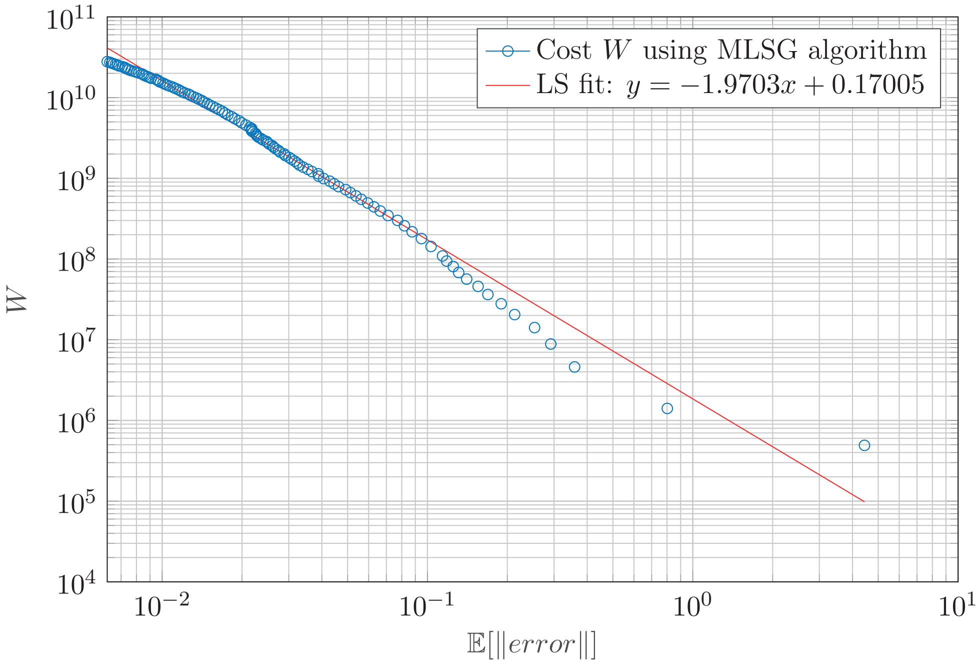


Figure IV.3 – Cost W versus mean error, using the MLSG Algorithm 4

IV.E.4. RMLSG algorithm

Using the randomized version of the MLSG Algorithm 5, we assess the convergence rate (IV.49) averaging over 1000 independent realizations, of the RMLSG algorithm, up to iteration $j = 1000$. The problem setting is the same as above and we have used the same parameters $r = 1$. $d = 2$, $\gamma = 1$, $C(u_\star) = 0.5$, $\tau_j = \tau_0/(j + 10)$ with $\tau_0 = 2/\beta$, $\overline{\epsilon}_0 = \sqrt{C(u_\star)h_0^{2r+2}}$, $\eta = 2$.We use the the *optimal* probability mass function:

$$p_l^j = 2^{-l\frac{2r+2+\gamma d}{2}} \left(\sum_{k=0}^{L_j} 2^{-k\frac{2r+2+\gamma d}{2}} \right)^{-1} \qquad j \geq 1, \quad l = 0, \dots, L_j.$$

In Figure IV.4, we plot the mean error versus the iteration counter in log-scale and observe a rate $-1/2$ which is consistent with the result in Lemma 22 with $\eta = 2$. Figure IV.5 shows the *expected* computational cost model $\mathbb{E}[W_j] = \sum_{i=1}^{j} \sum_{k=0}^{L_i} 2^{k\gamma d} p_k^i$ versus the actual mean error averaged over the 1000 repetitions and verify a slope of -2 consistent with the complexity result in Theorem 24 (with $\eta = 2$). The discontinuities in the expected computational cost in Figure IV.5 are due to the fact that the expected cost is not an increasing function of the iteration counter, as shown on Figure IV.6. Specifically, as the probability mass function $\{p_l^j\}$ is normalized by its sum $\sum_{k'=0}^{L_j} 2^{-k'\frac{2r+2+\gamma d}{2}}$, when we reach iteration j where the maximum level L_j is increased the by 1, we observe slight lower expected cost, i.e. $\mathbb{E}[W(E^{\text{RMLMC}}_{L_j,\vec{p}^j})] = \sum_{k=0}^{L_j} 2^{k\gamma d} p_k^j = \sum_{k=0}^{L_j} 2^{-k\frac{2r+2-\gamma d}{2}} \left(\sum_{k'=0}^{L_j} 2^{-k'\frac{2r+2+\gamma d}{2}}\right)^{-1}$ is **not** monotonic in j.

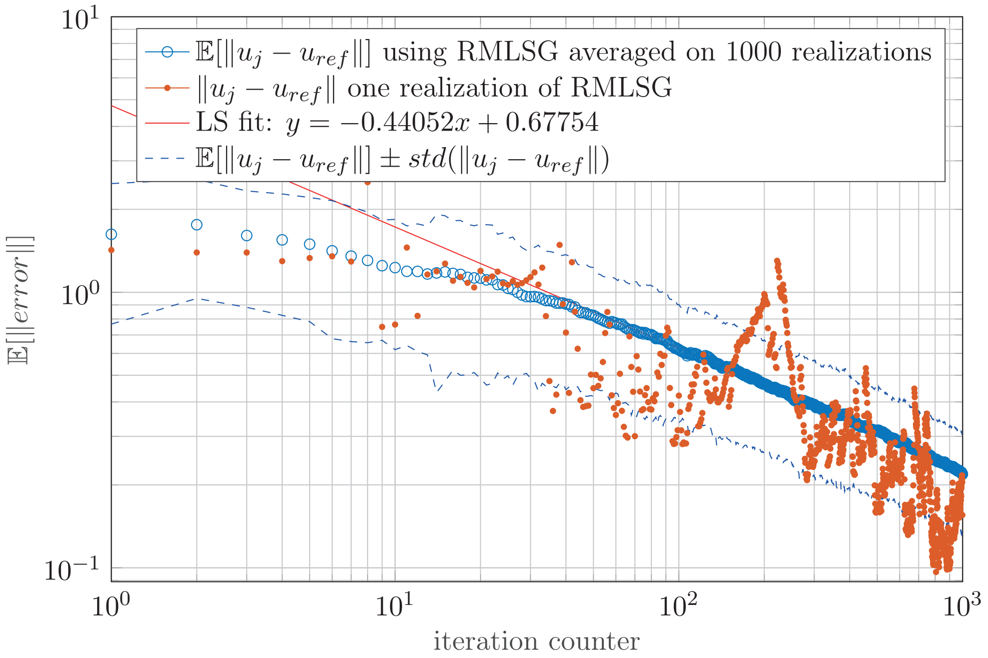


Figure IV.4 – Mean error vs iteration counter for the RMLSG Algorithm 5

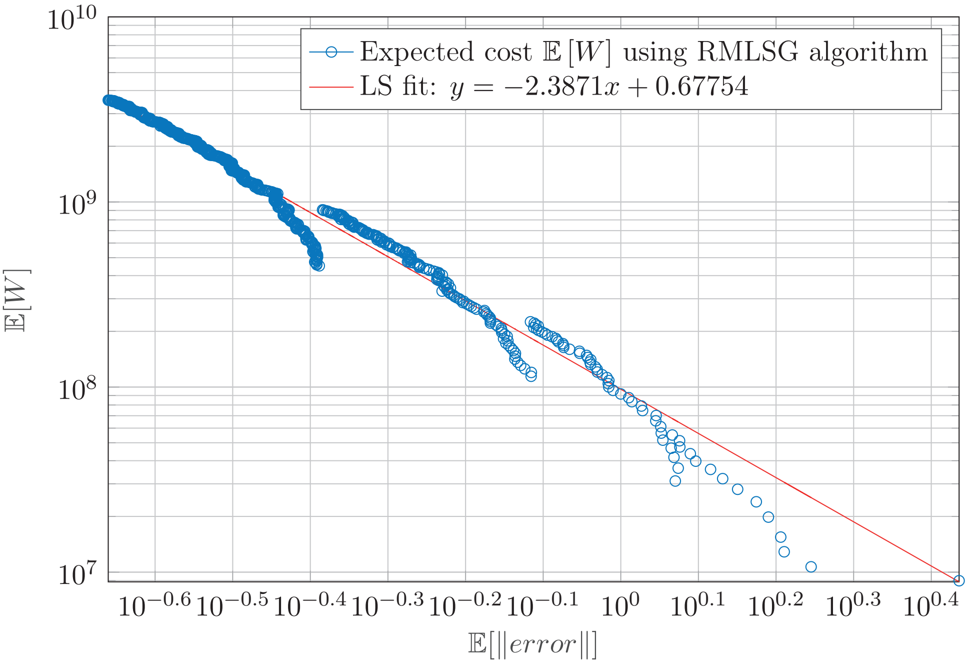


Figure IV.5 – Expected cost $\mathbb{E}[W]$ vs mean error for the randomized MLSG Algorithm 5

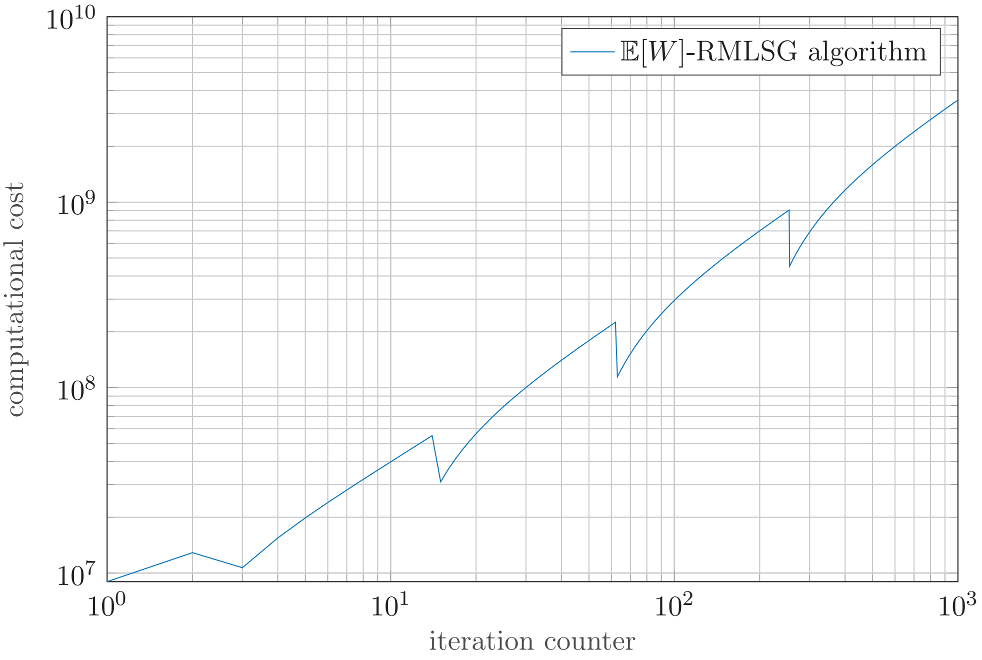


Figure IV.6 – Expected computational cost (model) vs iteration counter in log scale

IV.F. Conclusions

In this work, we presented a modified version of the Stochastic Gradient algorithm, in order to solve numerically a PDE-constrained OCP, with uncertain coefficients. The usual Robbins-Monro approach, involving a single realization estimator of the gradient is replaced by either a MLMC estimator, with increasing cost w.r.t. the iteration counter, or a randomized version of the MLMC estimator, where only one difference term of the full MLMC estimator is computed at each iteration, on a randomly drawn level, according to a probability mass function, set *a priori*. We have shown that both algorithms, when properly tuned, achieve the optimal complexity $W \lesssim tol^{-2}$. These complexity results are assessed in the numerical Section. In practice, many constants have to be tuned beforehand in these 2 MLSG algorithms, and as the randomized version, namely RMLSG, presents fewer constants/parameters to tune, it is preferred as it guarantees less variability w.r.t. the choice of the algorithm parameters.

V PDE-constrained OCPs with uncertain parameters using SAGA

This Chapter is essentially the same as the publication [MN18], submitted for publication.

V.A. Introduction

In this paper we consider a risk averse optimal control problem (OCP) for an elliptic PDE with random diffusion coefficients

$$u^* \in \operatorname*{arg\,min}_{u \in U} J(u), \quad J(u) = \mathbb{E}_\omega[f(u,\omega)] \tag{V.1}$$

where $\omega \in \Gamma$ denotes a random elementary event, $f(u,\omega) = \tilde{f}(y_\omega(u), u, \omega)$ and $y_\omega(u)$ is the solution of an elliptic PDE $\mathcal{E}(y_\omega(u), \omega) = u$ with some random coefficients. Here the right hand side u is a deterministic function, in a possibly infinite dimensional space U, that acts as a control so as to minimize the functional $f(\cdot, \omega)$, in an average sense with respect to (w.r.t.) ω. In particular, in the setting considered in this work, $u \mapsto f(u,\omega)$ is strongly convex for any $\omega \in \Gamma$.

Assuming that the randomness can be parametrized in terms of a small number M of independent random variables, the expectation appearing in the cost functional $J(u)$ can be written as a M-dimensional integral and suitably approximated by a quadrature formula as e.g. a tensorized Gaussian quadrature, leading to an approximate optimal control problem

$$\widehat{u}^* \in \operatorname*{arg\,min}_{u \in U} \widehat{J}(u), \quad \widehat{J}(u) = \sum_{j=1}^{n} \zeta_j f(u, \eta_j) \tag{V.2}$$

where η_j are the quadrature knots and ζ_j the quadrature weights with $\sum_{j=1}^n \zeta_j = 1$. For a given control u, evaluating $\widehat{J}(u)$ entails the computation of the n solutions $\{y_{\eta_j}(u)\}_{j=1}^n$ of the underlying PDE. This approach is known in the literature as *stochastic collocation* method and has been analyzed e.g. in [BNT10]. It leads, in favorable cases, to an error in the functional that converges to zero (sub)-exponentially in n, although typically exposed to the curse of dimensionality, hence acceptable only for a small number of random variables. By replacing the tensorized quadrature by a suitable sparse one (see e.g. [BG04, NTT16]), dimension free convergence rates have been demonstrated in certain cases (see e.g. [SS13, HANTT16a, EST18, ZDS18] and references therein). However, in this work, we stick to the simpler setting of a tensorized Gaussian quadrature formula and a small number of random variables.

To solve the approximate OCP (V.2), we could consider a steepest descent hereafter called Full Gradient (FG), or a Conjugate Gradient (CG) method which would converge exponentially fast in the number of iterations, i.e. $\|\widehat{u}_k - \widehat{u}_\star\| \leq C\rho^k$ for some $\rho \in (0,1)$ where $\widehat{u}_k$ is the k-th iterate of the method and the error is measured in a suitable norm. The practical limitation of this approach is that each iteration requires the evaluation of a descent direction for $\widehat{J}(u)$, which entails n solutions of the PDE and n solutions of the corresponding adjoint problem. If n is large, the cost for one single iteration may become excessively high.

A popular technique in the machine learning community to solve optimization problems of the form (V.2) is the *Stochastic Gradient* (SG) method ([RM51]) which reads

$$\widehat{u}_{k+1} = \widehat{u}_k - \tau_k \zeta_{i_k} \nabla_u f(u, \eta_{i_k})$$

where the gradient of only one term in the sum is evaluated at each iteration (corresponding to one primal and one adjoint computation) for a randomly drawn index i_k, and the convergence is achieved by reducing the step size τ_k over the iterations. This makes the cost of each iteration affordable. The convergence of the SG method for a PDE-constrained optimal control problem with uncertain parameters has been studied in the recent work [MKN18] in the context of a Monte Carlo approximation of the expectation appearing in (V.1). In particular, we have shown that the root mean squared error $\mathbb{E}[\|\widehat{u}_k - \widehat{u}^*\|]$ of the SG method converges with order $1/\sqrt{k}$, which is the same order of the Monte Carlo "quadrature" error, and leads to an optimal strategy and a slightly better overall complexity than a FG (or CG) approach. In the setting of this paper, however, the quadrature error decays (sub)-exponentially, and the convergence rate of $1/\sqrt{k}$ of the SG method would lead to a much worse complexity than a FG or a CG method.

In recent years, variants of the SG method for a finite dimensional optimization problem of the form (V.2), such as the *Stochastic Averaged Gradient* (SAG) method [SRB13] and the SAGA method [DBLJ14] have been proposed, which recover an exponential convergence in the number of iterations, by introducing a memory term which stores all previously computed gradients in the sum and overwrite a term if the corresponding index is re-drawn. The method presented in [DBLJ14] is applicable to the case of uniform weights $\zeta_j = \frac{1}{n}$, $i = 1, \dots, n$ and uniformly drawn indexs i_k over $\{1, \dots, n\}$. A variant of the SAGA method that uses a non-uniform sampling of the indexs i_k has been proposed in [SBA$^+$].

In this paper, we extend the SAGA algorithm to the infinite dimensional setting of problem (V.2) and to the case of non-uniform weights, as they appear naturally in a Gaussian quadrature formula. In particular, we propose an importance sampling strategy where the indexs i_k are drawn from a possibly non-uniform distribution, also different from the distribution induced by the weights $\{\zeta_j\}_{j=1}^n$.

Following similar steps as in [DBLJ14, SBA$^+$], we present a full theoretical convergence analysis of the generalized SAGA method for the infinite dimensional OCP (V.2). In particular we show that, asymptotically in n, the optimal sampling measure for the indexs i_k is the *uniform* measure.

We also present a complexity analysis, in terms of computational cost versus accuracy, of the generalized SAGA method to solve the original OCP (V.1), which accounts for both the Stochastic Collocation quadrature error as well as the error in solving the primal and adjoint PDEs approximately by the finite element method. The complexity of SAGA is then compared to the complexity of FG as well as SG. Our theoretical results show that the generalized SAGA method has the same asymptotic complexity as the FG method and outperforms SG.

As shown by our numerical experiments, the interest in using SAGA versus FG is in the pre-asymptotic regime, as SAGA often delivers acceptable solutions, from a practical point of view, well before performing n iterations, i.e. with far less that $2n$ PDE solves (we recall that one single FG iteration entails already $2n$ PDE solves). In a context of limited budget, SAGA represents therefore a very appealing option.

As pointed out above, in this work we have restricted our study to the case of a small number of random variables and a tensorized Gaussian type quadrature formula, the main constraint in our analysis being that we need positive weights $\{\zeta_j\}$. This leaves open the question if the methodology can be extended and applied also with other quadrature formulas, such as sparse grid quadratures whose weights are not all positive, and possibly a large number of random variables. This question needs further investigation. We believe, however, that the current work provides an important and necessary step toward further generalizations.

V.B. Problem setting

We start by introducing the primal problem that will be part of the OCP discussed in the following. Specifically, we consider the problem of finding the solution $y : D \times \Gamma \to \mathbb{R}$

of the elliptic random PDE

$$\begin{cases} -\operatorname{div}(a(x,\omega)\nabla y(x,\omega)) &= g(x) + u(x), \qquad x \in D, \quad \omega \in \Gamma, \\ y(x,\omega) &= 0, \qquad x \in \partial D, \quad \omega \in \Gamma, \end{cases} \tag{V.3}$$

where $D \subset \mathbb{R}^d$ denotes the physical domain and $(\Gamma, \mathcal{F}, \mathbb{P})$ is a complete probability space. The diffusion coefficient a is a random field, g is a deterministic source term and u is the deterministic control. The solution of (V.3) for a given control u will be equivalently denoted $y_\omega(u)$, or simply $y(u)$ in what follows. Let $U = L^2(D)$ be the set of all admissible control functions and $Y = H_0^1(D)$ the space of the solutions of (V.3), then the goal is to determine the optimal control u^*, in the sense that:

$$u_\star \in \operatorname*{arg\,min}_{u \in U} J(u), \quad \text{s.t.} \quad y_\omega(u) \in Y \quad \text{solves} \quad \text{(V.3)} \quad \text{almost surely (a.s.) in } \Gamma. \tag{V.4}$$

Here, $J(u) := \mathbb{E}[f(u,\omega)]$ is the objective function with $f(u,\omega) = \frac{1}{2}\|y_\omega(u) - z_d\|^2 + \frac{\beta}{2}\|u\|^2$ and z_d is the target function that we would like the state y to approach as close as possible. We have denoted by $\|\cdot\|$ the $L^2(D)$-norm induced by the inner product $\langle \cdot, \cdot \rangle$.

V.B.1. Existence and uniqueness result

We use results from [MKN18, Sections 2 and 3]. We recall the three assumptions from [MKN18] that guarantee well posedness of (V.4) and regularity of solutions.

Assumption 17. *The diffusion coefficient $a \in L^\infty(D \times \Gamma)$ is bounded and bounded away from zero a.e. in $D \times \Gamma$, i.e.*

$$\exists \quad a_{\min}, a_{\max} \in \mathbb{R} \quad \text{such that} \quad 0 < a_{\min} \leq a(x,\omega) \leq a_{\max} \quad \text{a.e. in } D \times \Gamma.$$

Assumption 18. *The regularization parameter β is strictly positive, i.e. $\beta > 0$ and the deterministic source term is such that $g \in L^2(D)$.*

In what follow, we denote the $L^2(D)$-functional representation of the Gateaux derivative of J, by $\nabla_u J(u)$, namely

$$\int_D \nabla_u J(u) \delta u \, \mathrm{d}x = \lim_{\epsilon \to 0} \frac{J(u + \epsilon \delta u) - J(u)}{\epsilon} \quad \forall\, \delta u \in L^2(D).$$

Existence and uniqueness of the OCP (V.4) can be stated as follows.

Theorem 26. *Under Assumptions 17 and 18, the OCP* (V.4) *admits a unique control $u_\star \in U$. Moreover*

$$\nabla_u J(u) = \beta u + \mathbb{E}[p_\omega(u)], \tag{V.5}$$

where $p_\omega(u) = p$ is the solution of the adjoint problem (a.s. in Γ)

$$\begin{cases} -\operatorname{div}(a(\cdot,\omega)\nabla p(\cdot,\omega)) &= y(\cdot,\omega) - z_d \qquad \text{in } D, \\ p(\cdot,\omega) &= 0 \qquad \text{on } \partial D. \end{cases} \tag{V.6}$$

We recall as well the weak formulation of (V.3), which reads

$$\text{find } y_\omega \in Y \; s.t. \; b_\omega(y_\omega, v) = \langle g + u, v\rangle \quad \forall v \in Y \qquad \text{for a.e. } \omega \in \Gamma, \tag{V.7}$$

where $b_\omega(y,v) := \int_D a(\cdot,\omega)\nabla y \nabla v dx$. Similarly, the weak form of the adjoint problem (V.6) reads:

$$\text{find } p_\omega \in Y \; s.t. \; b_\omega(v, p_\omega) = \langle v, y_\omega - z_d\rangle \quad \forall v \in Y \qquad \text{for a.e. } \omega \in \Gamma. \tag{V.8}$$

We can thus rewrite the OCP (V.4) equivalently as:

$$\begin{cases} \min_{u\in U} J(u), \quad J(u) = \frac{1}{2}\mathbb{E}[\|y_\omega(u) - z_d\|^2] + \frac{\beta}{2}\|u\|^2 \\ \text{s.t.} \quad y_\omega(u) \in Y \quad \text{solves} \\ b_\omega(y_\omega(u), v) = \langle g + u, v\rangle \quad \forall v \in Y \qquad \text{for a.e. } \omega \in \Gamma. \end{cases} \tag{V.9}$$

We continue recalling two regularity results, that have been proven in [MKN18], about Lipschitz property and strong convexity for f in the particular setting of the problem considered here.

Lemma 23 (Lipschitz condition)**.** *The random functional $f(u,\omega)$ is such that:*

$$\|\nabla_u f(u_1,\omega) - \nabla_u f(u_2,\omega)\| \leq L\|u_1 - u_2\| \quad \forall u_1, u_2 \in U \text{ and a.e. } \omega \in \Gamma, \tag{V.10}$$

with $L = \beta + \frac{C_p^4}{a_{min}^2}$, where C_p is the Poincaré constant, $C_p = \sup_{v\in Y/\{0\}} \frac{\|v\|}{\|\nabla v\|}$.

Lemma 24 (Strong Convexity)**.** *The (random) functional $f(u,\omega)$ is such that:*

$$\frac{l}{2}\|u_1 - u_2\|^2 \leq \langle \nabla_u f(u_1,\omega) - \nabla_u f(u_2,\omega), u_1 - u_2\rangle \quad \forall u_1, u_2 \in U \text{ and a.e. } \omega \in \Gamma, \tag{V.11}$$

with $l = 2\beta$.

V.B.2. Finite Element approximation

In order to compute numerically an optimal control we consider a Finite Element (FE) approximation of the infinite dimensional OCP (V.9). Let us denote by $\{\tau_h\}_{h>0}$ a family of regular triangulation of D and choose Y^h to be the space of continuous piece-wise polynomial functions of degree r over τ_h that vanish on ∂D, i.e. $Y^h = \{y \in C^0(\overline{D}) : y|_K \in \mathbb{P}_r(K) \quad \forall K \in \tau_h, y|_{\partial D} = 0\} \subset Y$, and $U^h = Y^h$. We reformulate the OCP (V.9) as a finite dimensional OCP in the FE space:

$$\begin{cases} \min_{u^h \in U^h} J^h(u^h), \quad J^h(u^h) = \frac{1}{2}\mathbb{E}[\|y^h_\omega(u^h) - z_d\|^2] + \frac{\beta}{2}\|u^h\|^2 \\ \text{s.t. } y^h_\omega \in Y^h \text{ and} \\ b_\omega(y^h_\omega(u^h), v^h) = \langle u^h + g, v^h \rangle \quad \forall v^h \in Y^h \quad \text{for a.e. } \omega \in \Gamma. \end{cases} \tag{V.12}$$

Under the following regularity assumption on the domain and diffusion coefficient:

Assumption 19. *The domain $D \subset \mathbb{R}^d$ is polygonal convex and the random field $a \in L^\infty(D \times \Gamma)$ is such that $\nabla a \in L^\infty(D \times \Gamma)$,*

the following error estimate has been obtained in [MKN18]. In order to lighten the notations, we omit the subscript ω in $y_\omega(\cdot)$ and $p_\omega(\cdot)$ from now on.

Theorem 27. *Let $u_\star$ be the optimal control, solution of problem* (V.9)*, and denote by $u^h_\star$ the solution of the approximate problem* (V.12)*. Suppose that $y(u_\star), p(u_\star) \in L^2_{\mathbb{P}}(\Gamma; H^{r+1}(D))$ and Assumption 19 holds; then*

$$\begin{aligned} &\|u_\star - u^h_\star\|^2 + \mathbb{E}[\|y(u_\star) - y^h(u^h_\star)\|^2] + h^2\mathbb{E}[\|y(u_\star) - y^h(u^h_\star)\|^2_{H^1_0}] \\ &\qquad\qquad \le A_1 h^{2r+2}\{\mathbb{E}[|y(u_\star)|^2_{H^{r+1}}] + \mathbb{E}[|p(u_\star)|^2_{H^{r+1}}]\}, \end{aligned} \tag{V.13}$$

with a constant A_1 independent of h.

The next step is to approximate the expectation $\mathbb{E}[\cdot]$ in (V.12) by a suitable quadrature formula $\widehat{E}[\cdot]$. This is detailed in the next section.

V.B.3. Collocation method

We describe here a semi-discrete (approximation in probability only) OCP obtained by replacing the exact expectation $\mathbb{E}[\cdot]$ in (V.9) by a suitable quadrature formula $\widehat{E}[\cdot]$. We assume that the random diffusion coefficient can be represented as a function of a finite number of independent uniformly distributed random variables:

$$a = a(x, \xi)$$

with $\xi = (\xi_1, \ldots, \xi_M)$ and $\xi_i \overset{\text{iid}}{\sim} \mathcal{U}([-1,1])$. Hence, in this case since the whole problem is parameterized by the random vector ξ, we can take a probability space $\Gamma = [-1,1]^M$, $\mathcal{F} = \mathcal{B}(\Gamma)$ the Borel σ-algebra on Γ, and $\mathbb{P}(d\xi) = \otimes_{i=1}^M \frac{d\xi_i}{2}$ the uniform product measure on Γ. In this case we chose as a quadrature formula the tensor Gaussian quadrature built on Gauss-Legendre quadrature points. In particular, if $X : \Gamma \to \mathbb{R}$, $\xi \mapsto X(\xi) = X(\xi_1, \ldots, \xi_M)$, is a random variable with finite mean, then the Gauss-Legendre quadrature formula is given by

$$\widehat{E}[X] = \sum_{j=1}^{n} \zeta_j X(\eta_j), \tag{V.14}$$

where n is the total number of points used, $\{\zeta_j\}_j$ are the positive quadrature weights and $\{\eta_j\}_j$ the associated quadrature knots. The semi-discrete collocation problem then reads:

$$\begin{cases} \min_{\widehat{u}\in U} \widehat{J}(\widehat{u}), \quad \widehat{J}(\widehat{u}) = \frac{1}{2}\widehat{E}[\|y_\xi(\widehat{u}) - z_d\|^2] + \frac{\beta}{2}\|\widehat{u}\|^2 \\ \text{s.t.} \quad y_{\eta_j}(\widehat{u}) \in Y \quad \text{and} \\ b_{\eta_j}(y_{\eta_j}(\widehat{u}), v) = \langle g + \widehat{u}, v\rangle \quad \forall v \in Y \quad j = 1, \dots, n. \end{cases} \tag{V.15}$$

An error estimate has been shown in [MKN18].

Lemma 25. *Let $u_\star$ be the optimal control, solution of* (V.9) *and $\widehat{u}_\star$ the solution of the semi-discrete OCP* (V.15)*. Then there exists $A_2 > 0$ s.t.*

$$\|u_\star - \widehat{u}_\star\|^2 + \widehat{E}[\|y(u_\star) - y(\widehat{u}_\star)\|^2] \leq A_2\|\mathbb{E}[p(\widehat{u}_\star)] - \widehat{E}[p(\widehat{u}_\star)]\|^2 \tag{V.16}$$

To quantify the convergence rate of the right hand side of (V.16), one has first to understand the smoothness of the function $\xi \mapsto p_\xi(u)$ for a generic $u \in U$. For this, we make the regularity assumption on the diffusion coefficient

Assumption 20. *The parametric diffusion coefficient $\xi \mapsto a(\cdot, \xi) \in L^\infty(D)$ is analytic in each variable $(\xi_1, \cdots, \xi_M)$ in Γ and there exist $0 < \gamma_1, \dots, \gamma_M \in \mathbb{R}$ and $A_3 > 0$ such that*

$$\left\| \frac{\partial^k a(\cdot, \xi)}{\partial \xi_j^k} \right\|_{L^\infty(D)} \leq A_3 k! \gamma_j^k \tag{V.17}$$

Then following [BNT10], it can be shown that for any $u \in U$, the primal solution $\xi \mapsto y_\xi(u) \in Y$ and the adjoint solution $\xi \mapsto p_\xi(u) \in Y$ are both analytic in Γ (see also [MKN18, Lemma 7]) and the following result holds:

Theorem 28. *Denoting by $\widehat{u}_\star$ the solution of the semi-discrete (in probability) optimal control problem* (V.15) *with $\widehat{E} = E_q^{GL}[\cdot]$ the tensor Gauss-Legendre quadrature formula with $q = (q_1, \dots, q_M)$ points in each of the variables $(\xi_1, \dots, \xi_m)$, and $p(\widehat{u}_\star)$ the corresponding adjoint solution, there exist $A_4 > 0$ and $0 < s_1, \cdots, s_M \in \mathbb{R}$ independent of q s.t.*

$$\|\mathbb{E}[p(\widehat{u}_\star)] - E_q^{GL}[p(\widehat{u}_\star)]\|^2 \leq A_4 \sum_{n=1}^{M} e^{-s_n q_n} .$$

Clearly, the discretization in space by Finite Elements (V.12) and in probability by

Gauss-Legendre formula (V.15) can be combined to obtain the fully discrete OCP:

$$\begin{cases} \min_{\widehat{u}^h \in \widehat{U}^h} \widehat{J}^h(\widehat{u}^h), \quad \widehat{J}^h(\widehat{u}^h) = \frac{1}{2}\widehat{E}[\|y^h_\omega(\widehat{u}^h) - z_d\|^2] + \frac{\beta}{2}\|\widehat{u}^h\|^2 \\ \text{s.t. } y^h_\omega \in Y^h \text{ and} \\ b_\omega(y^h_\omega(\widehat{u}^h), v^h) = \langle \widehat{u}^h + g, v^h\rangle \quad \forall v^h \in Y^h \quad \text{for a.e. } \omega \in \Gamma. \end{cases} \tag{V.18}$$

If $\widehat{u}^h_\star$ denotes the solution of OCP (V.18), the total error will satisfy

$$\|u_\star - \widehat{u}^h_\star\|^2 \leq A_5 \left(h^{2r+2} + \sum_{n=1}^{M} e^{-s_n q_n} \right), \tag{V.19}$$

for a suitable constant $A_5 > 0$ independent of h and $\{q_n\}$. The following section is dedicated to optimization techniques used to tackle such optimization problem. In particular, we focus on Stochastic Approximation methods as Stochastic Gradient and Stochastic Average Gradient. To keep the notation light, we present the different optimization algorithms and convergence estimates only for the semi-discrete problem (V.15), although all results extend straightforwardly to the fully discrete case.

V.C. Review of Stochastic Approximation methods

We recall two optimization techniques, namely Stochastic Gradient (SG) method, and Stochastic Averaged Gradient (SAG) method, mainly used in machine learning, and well adapted to solve optimization problems whose objective function is the sum of a large number of terms, as in our semi-discrete problem (V.15). We consider in this section the general optimization problem

$$\min_{u \in U} \widehat{J}(u), \quad \widehat{J}(u) = \frac{1}{n}\sum_{i=1}^{n} g_i(u). \tag{V.20}$$

where each function g_i is convex, differentiable, each gradient ∇g_i is Lipschitz-continuous, as defined in (V.10) replacing f with g_i, with a common Lipschitz constant L, and U is finite dimensional.

V.C.1. Stochastic Gradient (SG)

Known in literature as Stochastic Approximation (SA) or Stochastic Gradient (SG) [RM51, PJ92, SDR09, SRB13, DB16], the classic version of such a method, the so-called Robbins-Monro method, works as follows. Within the steepest descent algorithm the exact gradient $\nabla \widehat{J}(u) = \frac{1}{n}\sum_{i=1}^{n} \nabla g_i(u)$ is replaced by one particular term of the sum, $\nabla g_{i_k}(u)$, where i_k is chosen at random at each iteration step k of the optimization algorithm:

$$u_{k+1} = u_k - \tau_k \nabla g_{i_k}(u_k). \tag{V.21}$$

Here, $i_k \sim \mathcal{U}(\{1,\dots,n\})$ are iid uniform random variables on $\{1,\dots,n\}$. In (V.21), τ_k is the step-size of the algorithm and decreases as $1/k$ in the usual approach. For the following theorem, we assume that each g_i is strongly-convex with a common constant l defined as in (V.11) replacing f with g_i.

Theorem 29. *Let $\widehat{u}_\star$ be the solution of problem* (V.20) *and denote by u_k the k-th iterate of* (V.21). *For the choice $\tau_k = \tau_0/k$ with $\tau_0 > 1/l$, then we have:*

$$\mathbb{E}[\|u_k - \widehat{u}_\star\|^2] \leq A_6 k^{-1}, \tag{V.22}$$

for a suitable constant $A_6 > 0$ independent of k.

V.C.2. Stochastic Average Gradient (SAG)

Another optimization method, called SAG, has recently been introduced in [SRB13]. It relies on the same idea as the SG algorithm, but introduces a memory, which stores the gradients computed using older controls, and averages over them to compute the new gradient direction. The SAG memory-based scheme reads

$$u_{k+1} = u_k - \frac{\tau_k}{n} \sum_{j=1}^{n} \nabla g_j(\phi_j^{k+1}), \tag{V.23}$$

where at each iteration k an index $i_k \in \{1,\dots,n\}$ is selected *at random*, and we set

$$\phi_j^{k+1} = \begin{cases} u_k & \text{if } j = i_k, \\ \phi_j^k & \text{otherwise.} \end{cases} \tag{V.24}$$

Again, the indexs $i_k \sim \mathcal{U}(\{1,\dots,n\})$ are iid uniform random variables. If we assume that each g_i is strongly-convex with a common constant l defined as in (V.11), the authors of [SRB13] have proven that the convergence is exponential in k.

Theorem 30. *Let $\widehat{u}_\star$ be the solution of problem* (V.20) *and denote by u_k the k-th iterate of* (V.23) *with $\tau_k = \frac{1}{16L}$. Then*

$$\mathbb{E}[\|u_k - \widehat{u}_\star\|^2] \leq A_7 \left(1 - \min\left\{\frac{l}{16L}, \frac{1}{8n}\right\}\right)^k, \tag{V.25}$$

for a suitable constant $A_7 > 0$ independent of k.

First we notice that we now require to store n gradients, and to update them on the fly, one at each iteration. Depending on the memory required for one gradient storage

	SG	SAG-SAGA
convex	$\mathbb{E}[\widehat{J}(u_k)] - \widehat{J}(u_\star) = O(1/\sqrt{k})$	$\mathbb{E}[\widehat{J}(u_k)] - \widehat{J}(u_\star) = O(1/k)$
strongly-convex	$\mathbb{E}[\widehat{J}(u_k)] - \widehat{J}(u_\star) = O(1/k)$ $\mathbb{E}[\lVert u_k - u_\star \rVert^2] = O(1/k)$	$\mathbb{E}[\widehat{J}(u_k)] - \widehat{J}(u_\star) = O((1-\epsilon)^k)$ $\mathbb{E}[\lVert u_k - u_\star \rVert^2] = O((1-\epsilon)^k)$

Table V.1 – Convergence rate for SG and SAG method

(i.e. $\sim h^{-d}$ for the OCP (V.18) if h is the characteristic mesh size of the FE discretization, and d the space dimension of the problem), and on the parameter n, the memory needed can dramatically limit this algorithm. The major improvement of this method with respect to SG is its exponential $(1-\epsilon)^k$ convergence rate for strongly-convex objectives, *similar to the full gradient method*, versus an algebraic $1/k$ rate for the SG method. Table V.1 summarizes the different convergence rates, for both the objective functional and the control for these two methods.

Remark 13. *Notice that in SAG, the step-size τ_k does not necessarily decrease and remains usually fixed. The factor $(1-\epsilon)$ in the convergence rate in Table V.1 depends on the Lipschitz constant L, the strong-convexity constant l, and on the parameter n such that:*

$$\epsilon = \min\left\{\frac{l}{16L}, \frac{1}{8n}\right\}, \tag{V.26}$$

when a fixed constant step-size $\tau_k = \frac{1}{16L}$ is used (see [SRB13]).

As pointed out in [SRB13], despite the fact that n appears in the convergence rate of SAG, in the case where $n > \frac{2L}{l}$, performing n iterations, i.e. one effective pass through the quadrature knots, reduces the error by a factor $(1-1/8n)^n \leq \exp(-1/8)$, which is independent of n. Thus, in this setting, each pass through all the data reduces the error by a constant multiplicative factor as in the FG algorithm.

V.C.3. SAGA

We recall here also a slightly modified version of SAG, called SAGA, proposed in [DBLJ14] where the updated part in the gradient estimator is changed by a factor n. It makes the gradient estimator unbiased, and simplifies the proof of convergence. The SAGA iterative scheme reads:

$$u_{k+1} = u_k - \tau_k \left(\nabla g_{i_k}(u_k) - \nabla g_{i_k}(\phi^k_{i_k}) + \frac{1}{n}\sum_{j=1}^{n} \nabla g_j(\phi^k_j) \right) \tag{V.27}$$

where, as for SAG, the indexs $i_k \sim \mathcal{U}(\{1,\dots,n\})$ are drawn independently and ϕ_j^k is updated as in (V.24). For comparison, SAG can be rewritten equivalently as:

$$u_{k+1} = u_k - \tau_k \left(\frac{\nabla g_{i_k}(u_k) - \nabla g_{i_k}(\phi_{i_k}^k)}{n} + \frac{1}{n}\sum_{j=1}^{n} \nabla g_j(\phi_j^k) \right). \tag{V.28}$$

The convergence rate of SAGA remains the same as for SAG, while lightening the proof of convergence. In the next section we apply SAGA to the OCP (V.15) combined with an importance sampling strategy, and extend its convergence proof to this setting.

V.D. SA methods in the context of PDE-constrained OCP

We aim now at applying SG and SAG/SAGA to the semi-discrete OCP (V.15) (or its fully discrete counterpart). The objective function $\widehat{J}(u)$ in (V.15) reads

$$\begin{aligned} \widehat{J}(u) &= \frac{1}{2}\widehat{E}[\|y_\xi(u) - z_d\|^2] + \frac{\beta}{2}\|u\|^2 \\ &= \sum_{i=1}^{n} \zeta_i f_i(u) \end{aligned} \tag{V.29}$$

with $f_i(u) = \frac{1}{2}\|y_{\eta_i}(u) - z_d\|^2 + \frac{\beta}{2}\|u\|^2$ and $\nabla_u f_i(u) = \beta u + p_{\eta_i}(u)$, where $\{\zeta_i\}_i$ are weights of the Gauss Legendre quadrature formula and $\{\eta_i\}_i$ its knots. One possibility to apply SG or SAG/SAGA to the OCP $\min_{u\in U} \widehat{J}(u)$ is to rewrite $\widehat{J}(u)$ as

$$\widehat{J}(u) = \frac{1}{n}\sum_{i=1}^{n} g_i(u)$$

with $g_i(u) = n\zeta_i f_i(u)$. However, the functions $f_i(u)$ are naturally weighted by the non-uniform weights $\{\zeta_i\}$ and this raises the question whether the indexs i_k in the Stochastic Approximation techniques should be drawn from a uniform or *non-uniform* distribution. We take the second, more general, approach by introducing a discrete probability measure $\widetilde{\zeta}$ on $\{1,\dots,n\}$, $\widetilde{\zeta}(j) = \widetilde{\zeta}_j > 0$, with $\sum_{j=1}^{n} \widetilde{\zeta}_j = 1$ and using an importance sampling strategy.

Hence the modified Stochastic Gradient method with importance sampling for the OCP (V.15) reads:

```
Algorithm 6: SG on PDE constrained OCP, with non-uniformly sampled indexs
  given $u_k$
  sample $i_k \sim \widetilde{\zeta}$
  compute $u_{k+1} = u_k - \tau_k \frac{\zeta_{i_k}}{\widetilde{\zeta}_{i_k}} \nabla_u f_{i_k}(u_k)$.
```

Similarly the modified SAGA method with importance sampling for the OCP (V.15) reads:

```
Algorithm 7: SAGA on PDE constrained OCP, with non-uniformly sampled indexs
    given $u_k$, $\{\phi_j^k\}_{j=1}^n$
    sample $i_k \sim \widetilde{\zeta}$
    compute $u_{k+1} = u_k - \tau_k \left( \left(\nabla_u f_{i_k}(u_k) - \nabla_u f_{i_k}(\phi_{i_k}^k)\right) \frac{\zeta_{i_k}}{\widetilde{\zeta}_{i_k}} + \sum_{j=1}^n \zeta_j \nabla_u f_j(\phi_j^k) \right)$
    set $\phi_j^{k+1} = \begin{cases} u_k & \text{if } j = i_k, \\ \phi_j^k & \text{otherwise.} \end{cases}$
```

In practice, in the SAGA algorithm, we do not store the past controls ϕ_j^k, rather the past gradients $grad_j^k = \nabla_u f_j(\phi_j^k)$. Similarly, we do not recompute at each iteration k the whole sum $G_k = \sum_{j=1}^n \zeta_j \nabla_u f_j(\phi_j^k)$, rather update it using the formulas

$$G_{k+1} = G_k - \zeta_{i_{k+1}} grad_{i_{k+1}}^k + \zeta_{i_{k+1}} \nabla_u f_{i_{k+1}}(u_k)$$

and then the memory place $grad_{i_{k+1}}$ as:

$$grad_j^{k+1} = \begin{cases} \nabla_u f_j(u_k) & \text{if } j = i_{k+1}, \\ grad_j^k & \text{otherwise.} \end{cases}$$

We point out that both the SG and SAGA methods applied to the OCP (V.15) require 2 PDE's solved per iteration. Moreover SAGA requires to store $2n$ PDE solutions at all iterations.

V.D.1. Convergence and complexity analysis of the modified SG Algorithm 6

Following the analysis in [MKN18], we can bound the Mean Squared Error (MSE) of the SG iterates assuming a weighted summability property on the discrete probability $\{\widetilde{\zeta}_j\}_j$ used to sample the index i_k at iteration k:

Assumption 21 (Weights summability)**.** *Let us define*

$$\widetilde{S}_n = \sum_{j=1}^n \frac{\zeta_j^2}{\widetilde{\zeta}_j}. \tag{V.30}$$

There exist $0 < \widetilde{S} < \infty$*s.t. for every* $n \in \mathbb{N}$*,*

$$\widetilde{S}_n \leq \widetilde{S}.$$

Then, one can establish the following bound on the MSE when applying SG Algorithm 6:

Theorem 31. *Denoting by* $\widehat{u}_k^h$ *the* k*-th iteration of the modified Algorithm 6 applied to the fully discrete OCP* (V.18) *with FE mesh size* h *and Gauss-Legendre quadrature* *formula with* $(q_1, \ldots, q_M)$ *knots in each stochastic direction, then we can bound the MSE,* $\mathbb{E}[\|\widehat{u}_k^h - u_\star\|^2]$*, as:*

$$\mathbb{E}[\|\widehat{u}_k^h - u_\star\|^2] \leq B_1 k^{-1} + B_2 \sum_{n=1}^{M} e^{-s_n q_n} + B_3 h^{2r+2} \ , \tag{V.31}$$

with constants B_1, B_2, B_3 *independent of* $h, \{q_n\}_n$ *and* k*.*

We omit the proof as it follows very similar steps as in [MKN18]. We now analyze the complexity of the SG algorithm 6 in terms of computational work W versus accuracy tol.

Corollary 5. *In order to achieve a given tolerance tol, i.e. to guarantee that* $\mathbb{E}[\|\widehat{u}_k^h - u_\star\|^2] \lesssim tol^2$*, the total required computational work is bounded by*

$$W \lesssim tol^{-2-\frac{d\gamma}{r+1}}. \tag{V.32}$$

where we assume that the primal and adjoint problems can be solved, using a triangulation with mesh size h, in computational time $C_h = O(h^{-d\gamma})$*. Here,* $\gamma \in [1,3]$ *is a parameter representing the efficiency of the linear solver used (e.g.* $\gamma = 3$ *for a direct solver and* $\gamma = 1$ *up to a logarithm factor for an optimal multigrid solver), while d is the dimension of the physical space. The memory space required to store the gradient and solution at each iteration scales as*

$$storage \lesssim tol^{\frac{-d}{r+1}} \tag{V.33}$$

Proof. If we want to guarantee an error of order $O(tol)$, we can equalize the three terms on the right hand side of (V.31) to tol^2 thus obtaining:

$$h = O(tol^{\frac{1}{r+1}}), \quad q_j = \frac{2}{s_j}\log(tol^{-1}), \quad n = \left(\frac{2}{R}\log(tol^{-1})\right)^M, \quad k = O(tol^{-2}).$$

where we have set $R = \left(\prod_{j=1}^M s_j\right)^{\frac{1}{M}}$ the geometric mean of $(s_1, \ldots, s_M)$. As W denotes the computational work (proportional to time if we don't use any parallel computing strategy), we have

$$W = 2C_h k \lesssim tol^{-2} tol^{\frac{-d\gamma}{r+1}}. \tag{V.34}$$

The memory space required to store one gradient and one current control u_k, is proportional to h^{-d} thus leading to:

$$storage \lesssim tol^{\frac{-d}{r+1}} \tag{V.35}$$

□

V.D.2. Convergence analysis of the modified SAGA Algorithm 7

We prove in Theorem 32 below the exponential convergence in the number of iterations of Algorithm 7. The outcome of our analysis in that uniform sampling of the index i_k (i.e. $\widetilde{\zeta}_j = \frac{1}{n}$, $\forall j = 1, \ldots, n$) is indeed optimal in the sense that it provides the best convergence rate, asymptotically in n.

The proof is inspired from [DBLJ14] and is valid under the assumption that the weights $\{\zeta_j\}_j$ in (V.29) are positive and sum up to 1, which holds for Gaussian quadrature formulas, each f_i is Lipschitz with the same Lipschitz constant L, which is guaranteed for OCP (V.15) by Lemma 23, and $\widehat{J}(u) = \sum_{i=1}^{n} \zeta_i f_i(u)$ is strongly convex, which is guaranteed by Lemma 24. In what follows, we denote by F_k the σ-algebra generated by the random variables $i_0, i_1, \ldots, i_{k-1}$ and denote by $\mathbb{E}[\cdot|F_k]$ the conditional expectation to such σ-algebra. Moreover, in the remaining of this Section, we use the shorthand notation $f_j'(u)$ for $\nabla_u f_j(u)$. For the convergence proof, we also need to introduce the quantity

$$Q_k = \sum_{j=1}^{n} \frac{\zeta_j^2}{\widetilde{\zeta}_j} \|f_j'(\phi_j^k) - f_j'(\widehat{u}_\star)\|^2$$

where $\widehat{u}_\star$ denotes, as usual, the optimal control, solution of the semi-discrete OCP (V.15). We start our convergence analysis by few technical Lemmas.

Lemma 26. *We have the following bound on the conditional expectation* $\mathbb{E}[Q_{k+1}|F_k]$*:*

$$\mathbb{E}[Q_{k+1}|F_k] \leq \max_j (1 - \widetilde{\zeta}_j) Q_k + S_n L^2 \|u_k - \widehat{u}_\star\|^2 \text{ with } S_n = \sum_{p=1}^{n} \zeta_p^2 \tag{V.36}$$

Proof. We write the conditional expectation as a sum over the possible values $i_k = p$, $p \in \{1, \ldots, n\}$;

$$
\begin{aligned}
\mathbb{E}[Q_{k+1}|F_k] &= \mathbb{E}\left[\sum_{j=1}^{n}\frac{\zeta_j^2}{\widetilde{\zeta}_j}\|f_j'(\phi_j^{k+1}) - f_j'(\widehat{u}_\star)\|^2|F_k\right] \\
&= \sum_{p=1}^{n}\mathbb{E}\left[\sum_{j=1}^{n}\frac{\zeta_j^2}{\widetilde{\zeta}_j}\|f_j'(\phi_j^{k+1}) - f_j'(\widehat{u}_\star)\|^2|F_k, i_k = p\right]\widetilde{\zeta}_p \\
&= \sum_{p=1}^{n}\widetilde{\zeta}_p\left\{\sum_{j=1, j\neq p}^{n}\frac{\zeta_j^2}{\widetilde{\zeta}_j}\|f_j'(\phi_j^{k}) - f_j'(\widehat{u}_\star)\|^2 + \frac{\zeta_p^2}{\widetilde{\zeta}_p}\|f_p'(u_k) - f_p'(\widehat{u}_\star)\|^2\right\} \\
&= \underbrace{\sum_{p=1}^{n}\widetilde{\zeta}_p}_{=1}\left\{\sum_{j=1}^{n}\frac{\zeta_j^2}{\widetilde{\zeta}_j}\|f_j'(\phi_j^{k}) - f_j'(\widehat{u}_\star)\|^2\right\} - \sum_{p=1}^{n}\zeta_p^2\|f_p'(\phi_p^k) - f_p'(\widehat{u}_\star)\|^2 + \sum_{p=1}^{n}\zeta_p^2\|f_p'(u_k) - f_p'(\widehat{u}_\star)\|^2 \\
&= \sum_{j=1}^{n}\frac{\zeta_j^2}{\widetilde{\zeta}_j}(1-\widetilde{\zeta}_j)\|f_j'(\phi_j^{k}) - f_j'(\widehat{u}_\star)\|^2 + \sum_{p=1}^{n}\zeta_p^2\|f_p'(u_k) - f_p'(\widehat{u}_\star)\|^2 \\
&\leq \max_j\left(1-\widetilde{\zeta}_j\right)\sum_{j=1}^{n}\frac{\zeta_j^2}{\widetilde{\zeta}_j}\|f_j'(\phi_j^{k}) - f_j'(\widehat{u}_\star)\|^2 + \sum_{p=1}^{n}\zeta_p^2 L^2\|u_k - \widehat{u}_\star\|^2 \\
&\leq \max_j\left(1-\widetilde{\zeta}_j\right)Q_k + S_n L^2\|u_k - \widehat{u}_\star\|^2
\end{aligned}
$$

□

Lemma 27. *Let* $P_k = \left(f_{i_k}'(u_k) - f_{i_k}'(\phi_{i_k}^k)\right)\frac{\zeta_{i_k}}{\widetilde{\zeta}_{i_k}} + \sum_{j=1}^{n} f_j'(\phi_j^k)\zeta_j$ *and* $T_k = P_k - \nabla\widehat{J}(\widehat{u}_\star)$, *then we have the following properties:*

$$\mathbb{E}[P_k|F_k] = \nabla\widehat{J}(u_k) \tag{V.37}$$

$$\mathbb{E}[T_k|F_k] = \nabla\widehat{J}(u_k) - \nabla\widehat{J}(\widehat{u}_\star) \tag{V.38}$$

$$\mathbb{E}[\|T_k\|^2|F_k] \leq 9\widetilde{S}_n L^2\|u_k - \widehat{u}_\star\|^2 + 8Q_k \tag{V.39}$$

where $\widetilde{S}_n$ *is defined as in* (V.30).

Proof. Again, we further condition on the possible values taken by the random variable

i_k, thus obtaining:

$$\begin{aligned}
\mathbb{E}[P_k|F_k] &= \mathbb{E}\left[\left(\left(f'_{i_k}(u_k) - f'_{i_k}(\phi^k_{i_k})\right)\frac{\zeta_{i_k}}{\widetilde{\zeta}_{i_k}} + \sum_{j=1}^n f'_j(\phi^k_j)\zeta_j\right)|F_k\right] \\
&= \sum_{p=1}^n \mathbb{E}\left[\left(\left(f'_{i_k}(u_k) - f'_{i_k}(\phi^k_{i_k})\right)\frac{\zeta_{i_k}}{\widetilde{\zeta}_{i_k}} + \sum_{j=1}^n f'_j(\phi^k_j)\zeta_j\right)|F_k, i_k = p\right]\widetilde{\zeta}_p \\
&= \sum_{j=1}^n f'_j(u_k)\frac{\zeta_j}{\widetilde{\zeta}_j}\widetilde{\zeta}_j - \sum_{j=1}^n f'_j(\phi^k_j)\frac{\zeta_j}{\widetilde{\zeta}_j}\widetilde{\zeta}_j + \sum_{j=1}^n f'_j(\phi^k_j)\zeta_j \\
&= \sum_{j=1}^n f'_j(u_k)\zeta_j = \nabla\widehat{J}(u_k)
\end{aligned}$$

which proves (V.37). We see from this that P_k is an unbiased estimator of $\nabla\widehat{J}(u_k)$, when conditioned to F_k, which represents the main difference with SAG, and simplifies the convergence proof. Equation (V.38) follows straightforwardly:

$$\mathbb{E}[T_k|F_k] = \nabla\widehat{J}(u_k) - \nabla\widehat{J}(\widehat{u}_\star).$$

We prove now (V.39).

$$\begin{aligned}
\mathbb{E}[\|T_k\|^2|F_k] =&\mathbb{E}[\|T_k - \mathbb{E}[T_k|F_k]\|^2|F_k] + \|\mathbb{E}[T_k|F_k]\|^2 \\
=&\mathbb{E}[\|P_k - \nabla\widehat{J}(u_k)\|^2|F_k] + \|\nabla\widehat{J}(u_k) - \nabla\widehat{J}(\widehat{u}_\star)\|^2 \\
=&\mathbb{E}[\|\left(f'_{i_k}(u_k) - f'_{i_k}(\phi^k_{i_k})\right)\frac{\zeta_{i_k}}{\widetilde{\zeta}_{i_k}} + \sum_{j=1}^n f'_j(\phi^k_j)\zeta_j - \sum_{j=1}^n f'_j(u_k)\zeta_j\|^2|F_k] + \|\nabla\widehat{J}(u_k) - \nabla\widehat{J}(\widehat{u}_\star)\|^2 \\
\le&2\underbrace{\mathbb{E}[\|\left(f'_{i_k}(u_k) - f'_{i_k}(\phi^k_{i_k})\right)\frac{\zeta_{i_k}}{\widetilde{\zeta}_{i_k}}\|^2|F_k]}_{=A} + 2\underbrace{\mathbb{E}[\|\sum_{j=1}^n \zeta_j\left(f'_j(\phi^k_j) - f'_j(u_k)\right)\|^2|F_k]}_{=B} \\
&+\underbrace{\|\nabla\widehat{J}(u_k) - \nabla\widehat{J}(\widehat{u}_\star)\|^2}_{=C}
\end{aligned}$$

The first part A can be split as

$$\begin{aligned}
A &= \mathbb{E}[\|\underbrace{\left(f'_{i_k}(u_k) - f'_{i_k}(\phi^k_{i_k})\right)}_{\pm f'_{i_k}(\widehat{u}_\star)}\frac{\zeta_{i_k}}{\widetilde{\zeta}_{i_k}}\|^2|F_k] \\
&\le 2\underbrace{\mathbb{E}[\|\left(f'_{i_k}(u_k) - f'_{i_k}(\widehat{u}_\star)\right)\frac{\zeta_{i_k}}{\widetilde{\zeta}_{i_k}}\|^2|F_k]}_{=T_1} + 2\underbrace{\mathbb{E}[\|\left(f'_{i_k}(\widehat{u}_\star) - f'_{i_k}(\phi^k_{i_k})\right)\frac{\zeta_{i_k}}{\widetilde{\zeta}_{i_k}}\|^2|F_k]}_{=T_2}
\end{aligned}$$

with

$$T_1 \leq L^2\|u_k - \widehat{u}_\star\|^2\mathbb{E}[\frac{\zeta_{i_k}^2}{\widetilde{\zeta}_{i_k}^2}] = L^2\|u_k - \widehat{u}_\star\|^2\widetilde{S}_n$$

The term T_2 can be developed as a sum over the possible values of i_k:

$$T_2 = \mathbb{E}[\frac{\zeta_{i_k}^2}{\widetilde{\zeta}_{i_k}^2}\|f'_{i_k}(\phi_{i_k}^k) - f'_{i_k}(\widehat{u}_\star)\|^2|F_k] = \underbrace{\sum_{j=1}^n \frac{\zeta_j^2}{\widetilde{\zeta}_j}\|f'_j(\phi_j^k) - f'_j(\widehat{u}_\star)\|^2}_{=Q_k}$$

Moreover

$$\begin{aligned}
B &= \mathbb{E}[\|\sum_{j=1}^n \zeta_j\left(f'_j(\phi_j^k) - f'_j(u_k)\right)\|^2|F_k] \\
&\leq \left(\sum_{j=1}^n \zeta_j\|f'_j(\phi_j^k) - f'_j(u_k)\|\frac{\sqrt{\widetilde{\zeta}_j}}{\sqrt{\widetilde{\zeta}_j}}\right)^2 \\
&\leq \left(\sum_{j=1}^n \frac{\zeta_j^2}{\widetilde{\zeta}_j}\|\underbrace{f'_j(\phi_j^k) - f'_j(u_k)}_{\pm f'_j(\widehat{u}_\star)}\|^2\right)\underbrace{\left(\sum_{j=1}^n \widetilde{\zeta}_j\right)}_{=1} \\
&\leq 2\sum_{j=1}^n \frac{\zeta_j^2}{\widetilde{\zeta}_j}\|f'_j(\phi_j^k) - f'_j(\widehat{u}_\star)\|^2 + 2\sum_{j=1}^n \frac{\zeta_j^2}{\widetilde{\zeta}_j}\|f'_j(u_k) - f'_j(\widehat{u}_\star)\|^2 \\
&\leq 2Q_k + 2L^2\widetilde{S}_n\|u_k - \widehat{u}_\star\|^2
\end{aligned}$$

Finally, by Lemma 23

$$\begin{aligned}
C &= \|\sum_{j=1}^n \zeta_p\left(f'_p(u_k) - f'_p(\widehat{u}_\star)\right)\|^2 \\
&\leq \left(\sum_{j=1}^n |\zeta_p|\|f'_p(u_k) - f'_p(\widehat{u}_\star)\|\right)^2 \\
&\leq L^2\|u_k - \widehat{u}_\star\|^2\left(\sum_{j=1}^n |\zeta_p|\right)^2 \\
&\leq \widetilde{S}_nL^2\|u_k - \widehat{u}_\star\|^2
\end{aligned}$$

which completes the proof. □

Lemma 28. *Let $\alpha > 0$ and let Assumptions 23 and 24 hold. If u_k denotes the k-th* *iterate of SAGA Algorithm 7 and $\widehat{u}_\star$ is the solution of the OCP* (V.15)*, then there exist $D_1, D_2 \in \mathbb{R}_+$ such that:*

$$\mathbb{E}[\|u_{k+1} - \widehat{u}_\star\|^2 + \alpha Q_{k+1}|F_k] \leq D_1\|u_k - \widehat{u}_\star\|^2 + D_2\alpha Q_k$$

with $D_1 = 1 - l\tau + (\alpha S_n + 8\tau^2\widetilde{S}_n)L^2 + \tau^2 L^2$, $D_2 = 1 - \widetilde{\zeta}_{min} + \frac{8\tau^2}{\alpha}$, $\widetilde{\zeta}_{min} = \min_j \widetilde{\zeta}_j$, and S_n as in Lemma 26.

Proof. Using that $\nabla\widehat{J}(\widehat{u}_\star) = 0$, we have

$$\|u_{k+1} - \widehat{u}_\star\|^2 = \|u_k - \widehat{u}_\star - \tau\underbrace{(P_k - \nabla\widehat{J}(\widehat{u}_\star))}_{=T_k}\|^2 = \|u_k - \widehat{u}_\star\|^2 - 2\tau\langle u_k - \widehat{u}_\star, T_k\rangle + \tau^2\|T_k\|^2.$$

Let us develop now $\|u_{k+1} - \widehat{u}_\star\|^2 + \alpha Q_{k+1}$, using Lemmas 26 and 27,

$$\begin{aligned}
\mathbb{E}[\|u_{k+1} - \widehat{u}_\star\|^2 + \alpha Q_{k+1}|F_k] &= \mathbb{E}[\|u_k - \widehat{u}_\star\|^2|F_k] - 2\tau\mathbb{E}[\langle u_k - \widehat{u}_\star, T_k\rangle|F_k] + \tau^2\mathbb{E}[\|T_k\|^2|F_k] + \alpha\mathbb{E}[Q_{k+1}|F_k] \\
&= \|u_k - \widehat{u}_\star\|^2 - 2\tau\langle u_k - \widehat{u}_\star, \nabla\widehat{J}(u_k) - \nabla\widehat{J}(\widehat{u}_\star)\rangle + \tau^2\mathbb{E}[\|T_k\|^2|F_k] + \alpha\mathbb{E}[Q_{k+1}|F_k] \\
&\leq \|u_k - \widehat{u}_\star\|^2 - l\tau\|u_k - \widehat{u}_\star\|^2 + \tau^2\left(9\widetilde{S}_n L^2\|u_k - \widehat{u}_\star\|^2 + 8Q_k\right) \\
&+ \alpha\left(\max_j(1 - \widetilde{\zeta}_j)Q_k + S_n L^2\|u_k - \widehat{u}_\star\|^2\right) \\
&\leq \left(1 - l\tau + (\alpha S_n + 9\tau^2\widetilde{S}_n)L^2\right)\|u_k - \widehat{u}_\star\|^2 + \left(\max_j(1 - \widetilde{\zeta}_j) + 8\frac{\tau^2}{\alpha}\right)\alpha Q_k
\end{aligned}$$

□

We are now ready to state the final convergence result. For this, we need to find the right choice of $\alpha > 0$ and τ s.t.

$$1 - l\tau + (\alpha S_n + 9\tau^2\widetilde{S}_n)L^2 = D_1 < 1 \tag{V.40}$$

and

$$\max_j(1 - \widetilde{\zeta}_j) + \frac{8\tau^2}{\alpha} = D_2 < 1 \tag{V.41}$$

One particular choice that guarantees an exponential in k convergence rate is shown in the following Theorem.

Theorem 32. *Let Assumptions 21 holds and let us define:*

$$\widetilde{\zeta}_j = \frac{1}{n}, \quad N = 25\widetilde{S}, \quad \tau = \frac{l}{2NL^2}, \quad \alpha = 16n\tau^2.$$

with $\widetilde{S}$ as in Assumption 21. Then, we have

$$\mathbb{E}[\|u_{k+1} - \widehat{u}_\star\|^2 + \alpha Q_{k+1}] \leq \rho\mathbb{E}[\|u_k - \widehat{u}_\star\|^2 + \alpha Q_k]$$

with $\rho = \min\{1 - \frac{l^2}{4NL^2}, 1 - \frac{1}{2n}\} \in (0,1)$. Notice, in particular, that N and τ do not depend on n.

Proof. The particular choice of $\{\zeta_j\}_j$, α, τ implies $D_1 \leq 1 - \frac{l^2}{4NL^2}$, $D_2 = 1 - \frac{1}{2n}$, where we have exploited the fact that $nS_n = \widetilde{S}_n$ for $\widetilde{\zeta}_j = 1/n$. Hence,

$$\mathbb{E}[\|u_{k+1} - \widehat{u}_\star\|^2 + \alpha Q_{k+1}|F_k] \leq D_1\|u_k - \widehat{u}_\star\|^2 + D_2\alpha Q_k \leq \underbrace{\max(D_1, D_2)}_{=\rho}\{\|u_k - \widehat{u}_\star\|^2 + \alpha Q_k\}$$

The final result is obtained by taking a further expectation over $(i_0, \ldots, i_{k-1})$. □

Corollary 6. *Under Assumption 21 if u_k denotes the k-th iterate of SAGA described in Algorithm 7 and $\widehat{u}_\star$ is the solution of the semi-discrete OCP* (V.15), *then there exists $D_3 > 0$ such that:*

$$\mathbb{E}\left[\|u_k - \widehat{u}_\star\|^2\right] \leq D_3(1-\epsilon)^k \text{ with } \epsilon = \min\left\{\frac{l^2}{4NL^2}, \frac{1}{2n}\right\} \tag{V.42}$$

Proof. This result is a direct application of Theorem 32. □

Remark 14. *Theorem 32 generalized for any $\tau \in (0, \frac{l}{NL^2})$, in which case $D_1(\tau) = 1 - l\tau + \tau^2 L^2 N \in (1 - \frac{l^2}{4NL^2}, 1)$.*

We finish this subsection by showing that Assumption 21 holds, in the case of Gauss-Legendre quadrature.

Lemma 29. *In the setting of uniform random variables ξ and tensorized Gauss-Legendre quadrature formulas, choosing $\widetilde{\zeta}_p = \frac{1}{n}$, then Assumption 21 holds.*

Proof. As shown in [Sze39, page 353, (15.3.10)], the weights of the Gauss-Legendre quadrature formula satisfy

$$\zeta_p \lesssim \frac{1}{n}.$$

Hence, for a tensor quadrature with $(n_1, \ldots, n_M)$ points in each variables, and a multi-index $p = (p_1, \ldots, p_M)$, with $1 \leq p_i \leq n_i$ as $n = \prod_i n_i$ we have

$$\zeta_p = \prod_{i=1}^{M} \zeta_{p_i} \lesssim \prod_{i=1}^{M} \frac{1}{n_i}$$

and

$$\begin{aligned}
\sum_p \frac{\zeta_p^2}{\widetilde{\zeta}_p} = \sum_p \zeta_p^2 n &\lesssim \sum_{p_1=1}^{n_1} \cdots \sum_{p_M=1}^{n_N} \left(\prod_{i=1}^{M} \frac{1}{n_i} \right)^2 \prod_{i=1}^{M} n_i \\
&= \sum_{p_1=1}^{n_1} \cdots \sum_{p_M=1}^{n_N} \prod_{i=1}^{M} \frac{1}{n_i} \\
&= \prod_{i=1}^{M} \frac{1}{n_i} \sum_{p_1=1}^{n_1} \cdots \sum_{p_M=1}^{n_N} \\
&= \prod_{i=1}^{M} \frac{1}{n_i} \prod_{i=1}^{M} n_i \\
&= 1
\end{aligned}$$

with constant in the symbol $\lesssim$ independent of $(n_1, \ldots, n_M)$, but depending exponentially on M.

□

Remark 15. *Result of Lemma 29 still holds for Gauss-Jacobi abscissas, as proven in [Sze39, page 353, eq. (15.3.10)].*

V.D.3. Complexity analysis of SAGA

The convergence result stated in Theorem 28 for the semi-discrete OCP (V.15) applies equally well to the discrete OCP (V.18) with the same constants (thanks to the fact that the FE approximation functions $f_i^h(u^h)$ satisfy strong convexity and Lipschitz continuity inequalities as in Lemmas 23 and 24 with the same constants). We now analyze the complexity of the SAGA algorithm 7 in terms of computational work W versus accuracy tol. The complexity analysis is based on the following error splitting into FE discretization error, Gauss-Legendre quadrature error and SAGA optimization error when stopping the SAGA algorithm at iteration k leading to the following result.

Theorem 33. *With same notations of Corollary 6, if $\widehat{u}^h_{SAGA_k}$ denotes the approximated optimal control computed by using successively a FE approximation, full tensor Gauss-Legendre quadrature formula and SAGA method, then the MSE is bounded by:*

$$\mathbb{E}[\|\widehat{u}^h_{SAGA_k} - \widehat{u}_\star\|^2] \leq C_1(1-\epsilon)^k + C_2 \sum_{n=1}^{M} e^{-s_n q_n} + C_3 h^{2r+2} \tag{V.43}$$

with $\epsilon = \frac{1}{2n} = \frac{1}{2}\prod_{n=1}^{M} q_n^{-1}$, assuming that $n > \frac{50\widetilde{S}L^2}{l^2}$, and with constants C_1, C_2 and C_3 independent of $k, \{q_n\}_n$ and h.

Proof. We can decompose the total error using the three successive approximations presented in Section V.B and V.D: FE discretization, quadrature formula and SAGA optimization procedure:

$$\mathbb{E}[\|\widehat{u}^h_{SAGA_k} - \widehat{u}_\star\|^2] \leq 3\underbrace{\mathbb{E}[\|\widehat{u}^h_{SAGA_k} - \widehat{u}^h_\star\|^2]}_{SAGA} + 3\underbrace{\|\widehat{u}^h_\star - u^h_\star\|^2}_{quadrature} + 3\underbrace{\|u^h_\star - \widehat{u}_\star\|^2}_{FE} \tag{V.44}$$

where $\widehat{u}^h_\star$ is the optimal solution of the fully-discrete OCP (V.18) and $u^h_\star$ is the optimal control of the FE discretized OCP (V.12). The result is straightforward using the bounds in Corollary 6, Theorem 27 and Theorem 28. □

Corollary 7. *In order to achieve a given tolerance $O(tol)$, i.e. to guarantee that $\mathbb{E}[\|\widehat{u}^h_{SAGA_k} - \widehat{u}_\star\|^2] \lesssim tol^2$, the total required computational work is bounded by*

$$W \lesssim \left(\log(tol^{-1})\right)^{M+1} tol^{\frac{-d\gamma}{r+1}}. \tag{V.45}$$

where we assume that the primal and dual problems can be solved, using a triangulation with mesh size h, in computational time $C_h = O(h^{-d\gamma})$. Here, $\gamma \in [1,3]$ is a parameter representing the efficiency of the linear solver used (e.g. $\gamma = 3$ for a direct solver and $\gamma = 1$ up to a logarithm factor for an optimal multigrid solver), while d is the dimension of the physical space $D \subset \mathbb{R}^d$. The memory space required to store the history of the computed gradients scales as

$$storage = O\left(\left(\log(tol^{-1})\right)^M tol^{\frac{-d}{r+1}}\right) \tag{V.46}$$

Proof. Using Theorem 33, as we want to guarantee an error of order $O(tol)$, we can equalize the three terms on the right hand side of (V.43) to tol^2 and finally get:

$$h = O(tol^{\frac{1}{r+1}}), \quad q_j = \frac{2}{s_j}\log(tol^{-1}), \quad n = \left(\frac{2}{R}\log(tol^{-1})\right)^M, \quad k = \frac{2\log(tol^{-1})}{-\log(1-\frac{1}{2n})}$$

with $R = \left(\prod_j s_j\right)^{1/M}$. So we obtain asymptotically

$$k \sim 4n\log(tol^{-1}) \sim 4\left(\frac{2}{R}\log(tol^{-1})\right)^M \log(tol^{-1}) = O\left(\left(\log(tol^{-1})\right)^{M+1}\right)$$

If W denotes the computational work (proportional to time if we do not use any parallel computing strategy), we have

$$W = O\left(\left(\log(tol^{-1})\right)^{M+1} tol^{\frac{-d\gamma}{r+1}}\right). \tag{V.47}$$

The memory space required to store the history of all the n computed gradients is

$O(\cdot)$	FG	SG	SAGA
W	$\left(\log(tol^{-1})\right)^{M+1} tol^{\frac{-d\gamma}{r+1}}$	$tol^{-2-\frac{-d\gamma}{r+1}}$	$\left(\log(tol^{-1})\right)^{M+1} tol^{\frac{-d\gamma}{r+1}}$
storage	$tol^{\frac{-d}{r+1}}$	$tol^{\frac{-d}{r+1}}$	$\left(\log(tol^{-1})\right)^{M} tol^{\frac{-d}{r+1}}$

Table V.2 – Computational work and required storage memory for the modified Algorithm 6 and 7 to solve the OCP V.9.

proportional to nh^{-d}, so:

$$storage = O\left(\left(\log(tol^{-1})\right)^{M} tol^{\frac{-d}{r+1}}\right) \tag{V.48}$$

□

The computational work and storage requirements for SAGA stated in Corollary 7 are reported in Table V.2. For comparison, we state in the same Table also the complexity and storage requirement of the standard Stochastic Gradient algorithm, as well as the Full Gradient algorithm, both based on the same quadrature formula and FE approximation as for SAGA (we refer to [MKN18] where these results have been derived in the context of a Monte Carlo approximation). A naive implementation of the FG algorithm would require to store the gradient computed in each quadrature point, hence a storage of $O(nh^{-d})$. Alternatively, one can store only the partial weighted sum of the gradients and update it as soon as the gradient in a new quadrature point has been computed, which brings down the storage to $O(h^{-d})$.

V.E. Numerical results

In this section we verify the assertions on the order of convergence and computational complexity stated in Theorem 33 and Corollary 7. For this purpose, we consider the optimal control problem (V.9) in the domain $D = (0,1)^2$ with $g = 1$ and the following random diffusion coefficient:

$$\begin{aligned} a(x_1, x_2, \boldsymbol{\xi}) = 1 + \exp\left(var\left(\xi_1 \cos(1.1\pi x_1) + \xi_2 \cos(1.2\pi x_1)\right.\right. \\ \left.\left. + \xi_3 \sin(1.3\pi x_2) + \xi_4 \sin(1.4\pi x_2)\right)\right), \end{aligned} \tag{V.49}$$

with $(x_1, x_2) \in D$, $var = \exp(-1.125)$ and $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_4)$ with $\xi_i \overset{iid}{\sim} \mathcal{U}([-1,1])$ (this test case is taken from [LG17]). We have chosen $\beta = 10^{-4}$ as the price of energy (regularization parameter) in the objective functional. For the FE approximation, we have considered a structured triangular grid of mesh size h where each side of the domain D is divided into $1/h$ sub-intervals and used piece-wise linear finite elements (i.e. $r = 1$). For the approximation of the expectation in the objective functional, we have used a full tensor

Gauss-Legendre quadrature formula with the same number q of quadrature knots in each random variable ξ_j, $j \in \{1, 2, 3, 4\}$. All calculations have been performed using the FE library Freefem++ [Hec12].

V.E.1. Performance of SAGA and comparison with FG

In this subsection, we consider the SAGA method using a fixed mesh size $h = 2^{-3}$ and study its convergence for different levels of the full tensor Gauss-Legendre quadrature formula, i.e. a different number q of points in each random variable (the total number of quadrature points being q^4). For each q, we compute a reference solution using 50000 SAGA iterations (using again the same FE mesh size $h = 2^{-3}$). Then, we perform 5000 SAGA iterations and compare the error $= u_k - u_{ref}$ w.r.t. the reference solution. We repeat the computation 20 times, independently, to estimate the log-mean error $\log \mathbb{E}[\|\text{error}\|]$ (hereafter $\log(\cdot)$ refers to the base 10 logarithm). In all cases we have used a step-size $\tau = 1000$. We show in Figure V.1 the convergence plots of $\log(\mathbb{E}[\|\text{error}\|])$ versus k, for $q \in \{1, \ldots, 10\}$ and in Figure V.2 a zoom on the first 500 iterations. We can observe two regimes: a first one over the first 20 iterations, of faster exponential convergence, and a second one afterwards, of slower, but still exponential convergence. Then, in order to verify the exponential rate $1 - \epsilon$ of equation (V.43), we plot in Figure V.3 (top), the

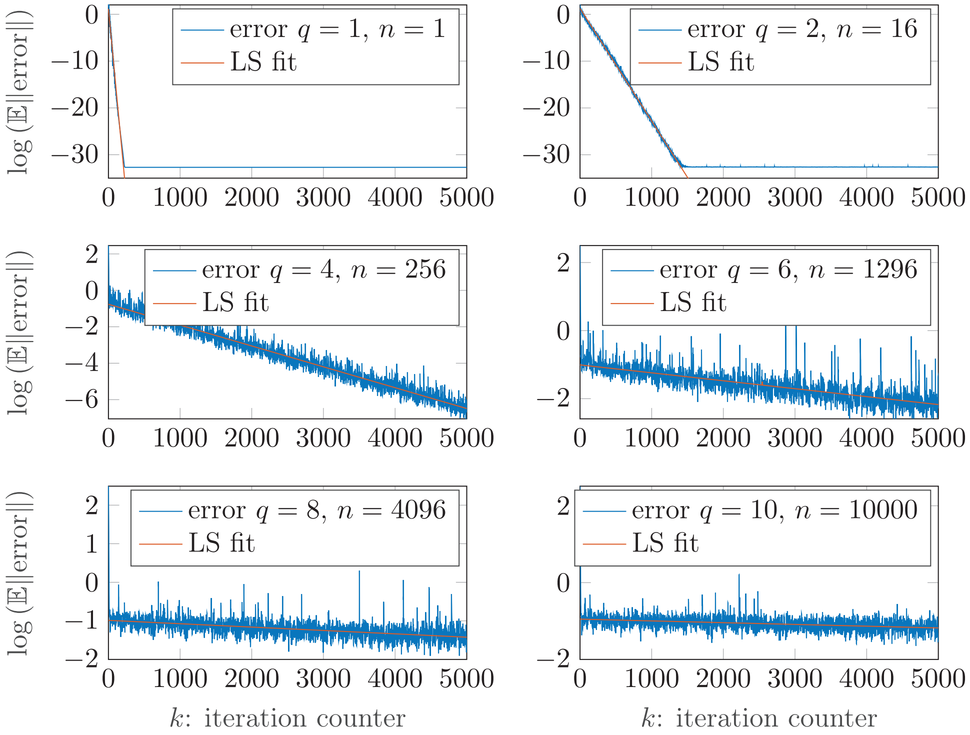


Figure V.1 – Convergence of SAGA for different q and fixed FE mesh (reference solution computed with the same FE mesh and quadrature with q^4 nodes).

estimated convergence rate, ϵ_{est} divided by the theoretical one $\epsilon_{th} = \frac{1}{2n} = \frac{1}{2q^4}$ versus q. Similarly, we plot in Figure V.3 (bottom) the estimated constant C_1 of equation (V.43) versus q. In both cases, the plotted quantity varies very little for $q \in \{3, \ldots, 10\}$ which confirms the validity of our theoretical analysis.

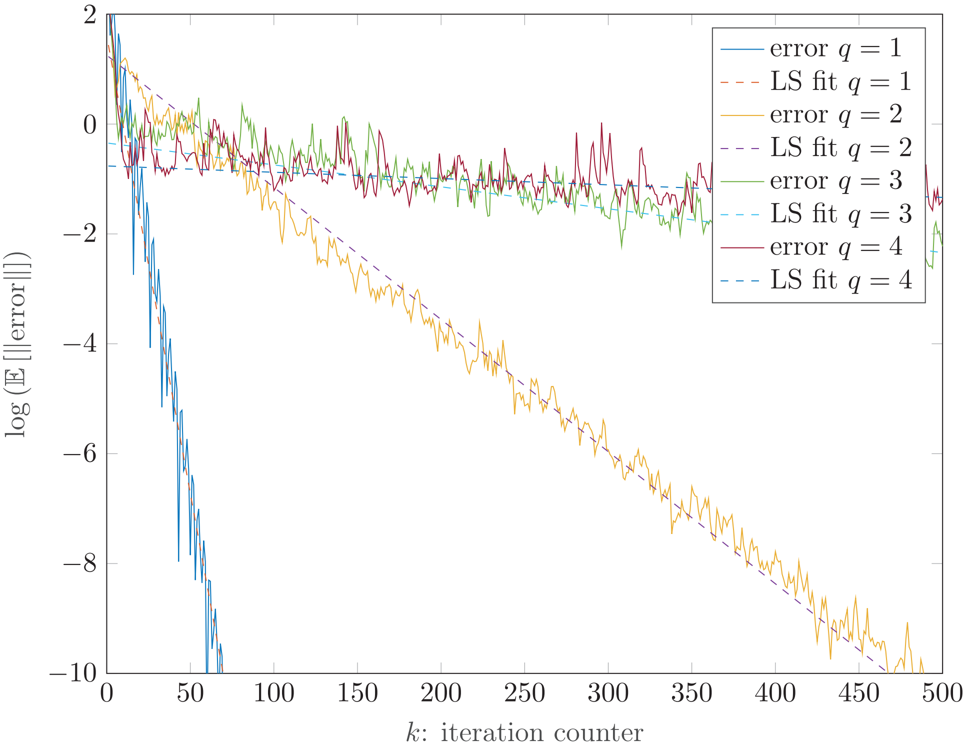


Figure V.2 – Convergence of SAGA algorithm, for different values of q. Zoom on the first 500 iterations.

We analyze next the sensitivity of the SAGA algorithm w.r.t. the step-size. In Figure V.4, we plot the L^2-norm of the error (averaged over 20 experiments) versus the iteration counter, for $q = 5$. With a high step-size, $\tau = 2000$, the method diverges to infinity. Progressively decreasing τ to 1000, the method starts converging, with the expected exponential rate, although slowly. Among the different values that we have tried, a step size $\tau = 100$ seems to provide the fastest convergence. Further decreasing τ to 10 makes SAGA converge poorly.

In Figure V.5 we compare the convergence rate of SAGA, with $\tau = 100$, with that of a full gradient method using the optimal step-size for a quadratic optimization problem, i.e.

$$\tau_k = \frac{\|\nabla \widehat{J}(u_k)\|^2}{\beta \|\nabla \widehat{J}(u_k)\|^2 + \|\widehat{E}[y_\xi(\nabla \widehat{J}(u_k) - g)]\|^2} \tag{V.50}$$

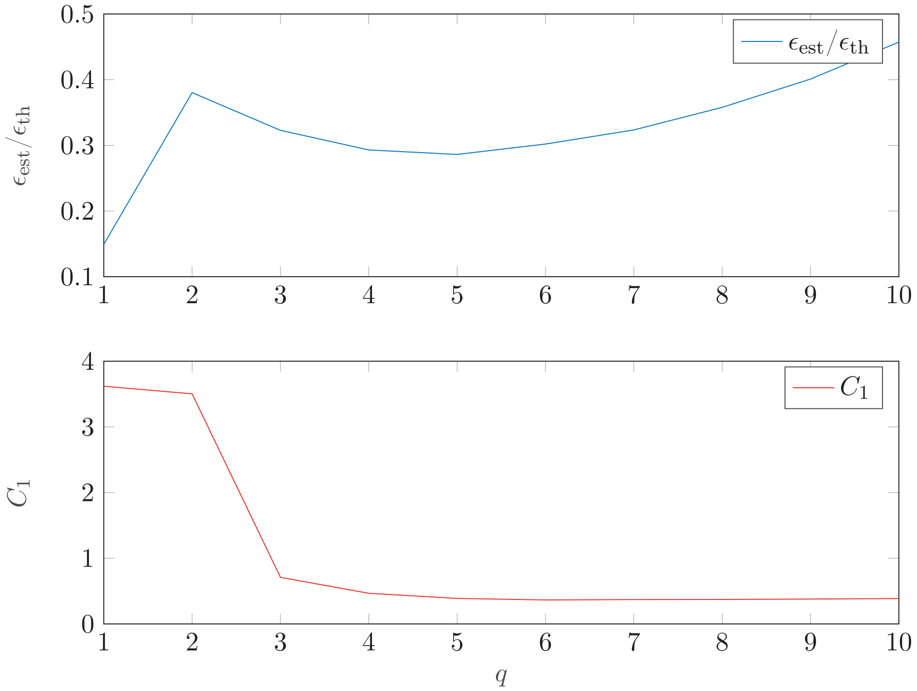


Figure V.3 – Assessment of SAGA convergence rate $C_1(1-\epsilon)^k$. Top (—): estimated rate $\epsilon_{est}/\epsilon_{th} = \epsilon_{est}2q^4$. Bottom (—): estimated constant C_1.

where $\widehat{E}$ is the Gauss-Legendre quadrature formula defined in (V.14), and $y_\xi(\cdot)$ is the solution of (V.3). The error in both cases is plotted against the number of PDE solves. The plot shows a fastest convergence of FG than SAGA, asymptotically. However, SAGA features a smaller error in the pre-asymptotic regime, and delivers an acceptable solution, from a practical point of view, already before two full iterations of FG (i.e. having solved 2500 PDEs). This makes it attractive in a limited budget context.

V.E.2. Complexity results for the SAGA algorithm

We investigate here the convergence of the method defined in Algorithm 7, for which we recall the error bound (V.43) in the case of piece-wise linear FE (i.e. $r=1$) and a 4-dimensional stochastic variable (i.e. $M=4$):

$$\mathbb{E}[\|\widehat{u}^h_{SAGA_k} - u_\star\|^2] \leq C_1(1-\epsilon(q))^k + C_2 e^{-sq} + C_3 h^4 \tag{V.51}$$

where s is the rate of exponential convergence of the quadrature formula, and q the number of knots used in each stochastic variable (isotropic case).

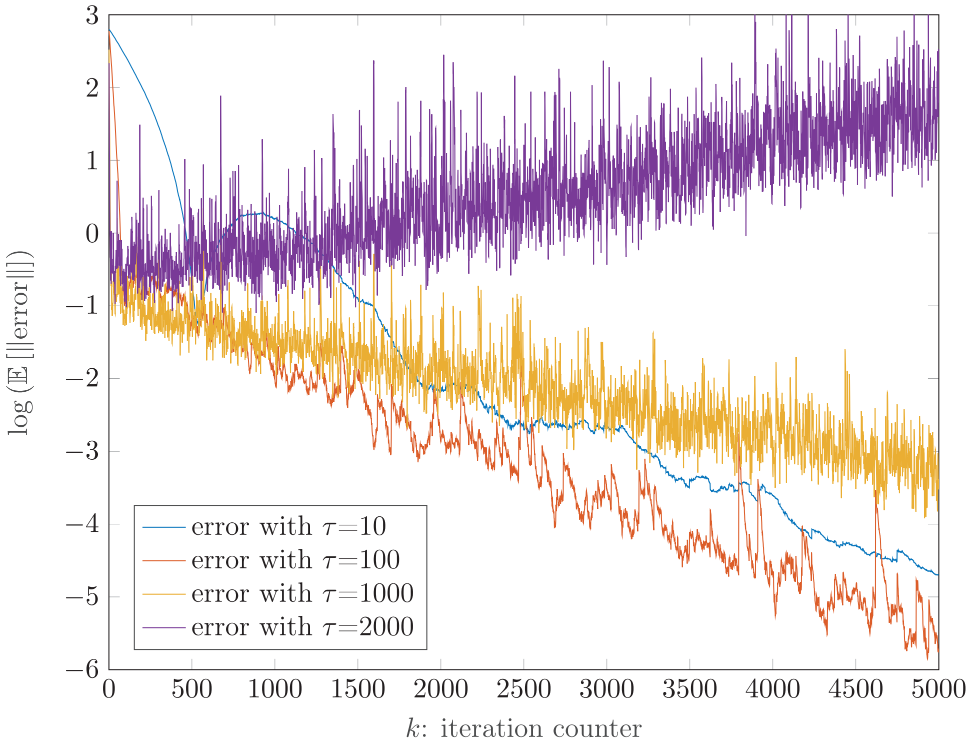


Figure V.4 – SAGA sensitivity to τ for $q = 5$.

We have estimated the constants in (V.51) numerically.

- In order to estimate C_1 and ϵ in (V.51), we used a fixed mesh with $h = 2^{-3}$ and a fixed quadrature formula for both reference solution and SAGA iterations. By doing so, we remove the two last terms in the bound (V.51), and only keep the first one of which we want to estimate the constants C_1 and $\epsilon(q)$. Using Figures V.1 and V.3 previously described, we estimate $C_1 \in [1200, 4500]$ and $\epsilon(q) \approx \frac{0.2}{q^4}$.

- To estimate the constant C_2 and s in (V.51), we use a mesh of size $h = 2^{-4}$ for both the reference solution and the optimal control with quadrature. First, we computed the reference solution for a fine Gauss-Legendre quadrature formula, i.e. $q = 8$, using the FG algorithm up to iteration 300. Then we computed the error for the approximated optimal control using only $q \in \{1, \ldots, 5\}$ points in each random variable, using again the FG algorithm up to iteration 100. In both cases we have used a step-size $\tau = 2000$. Results are detailed in Table V.3 and plotted in Figure V.6. The error is the difference between the estimated optimal control using $q \in \{1, \ldots, 5\}$ knots in the quadrature formula, and the optimal control computed for $q = 8$. We estimate $C_2 \approx 57.4$ and $s \approx 5.89$.

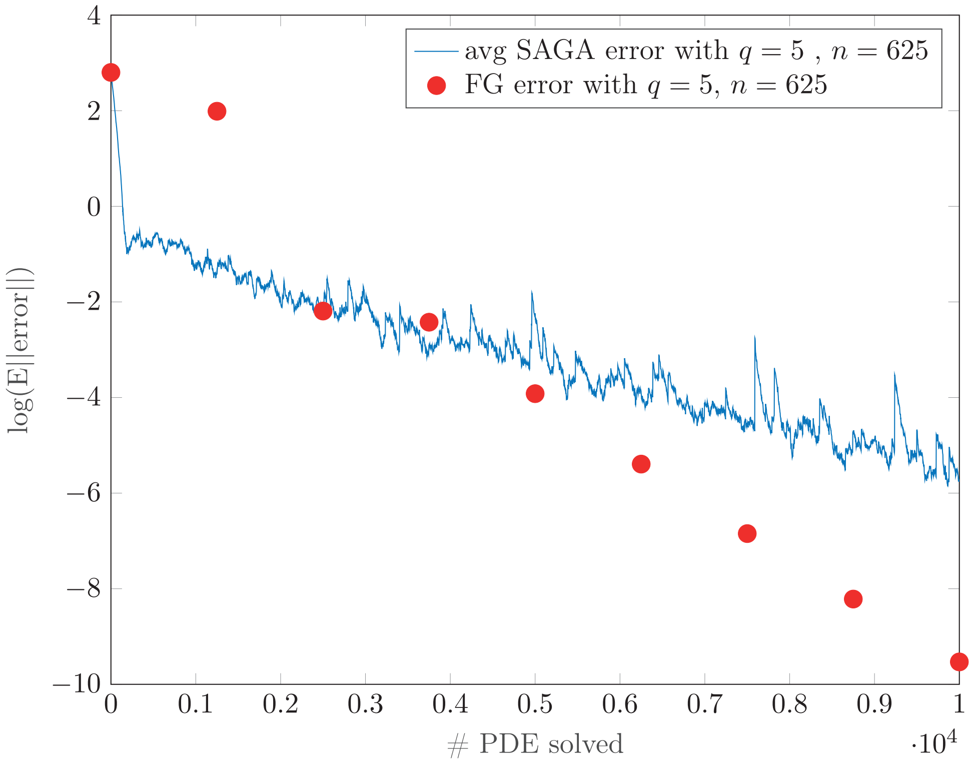


Figure V.5 – Comparison of SAGA with $\tau = 100$ vs FG using the optimal step-size (V.50), for $q = 5$.

- Finally, to estimate the third term C_3, we used the FG algorithm up to iteration 300, with a step-size $\tau = 2000$, on a quadrature formula with $q = 1$ knot in each random variable, and computed the reference solution on a fine mesh with $h = 2^{-7}$. Then we run FG with the same step size $\tau = 2000$ and quadrature ($q = 1$) on coarser meshes with $h = 2^{-1}, \cdots, 2^{-6}$. Results are shown in Table V.4. The Table confirms a convergence $O(h^4)$ of the squared error with an estimated constant $C_3 \approx 1170$. To double check our estimate, we repeated the estimation using the SAGA algorithm: for the reference solution, we used a mesh size $h = 2^{-9}$, and $q = 2$ points in the quadrature formula, and SAGA algorithm up to iteration 20000. Then we computed 10 repetitions of SAGA using mesh sizes $h = 2^{-1}, \cdots, 2^{-7}$, up to iteration 10000 and computed the average error. Results are shown in the third column of Table V.4. The results are almost identical to those obtained with the FG algorithm, which makes us believe that our estimation of C_3 is reliable.

In order to assess the complexity of the SAGA algorithm, we set a target tolerance *tol*. For each target tolerance *tol*, we compute the optimal mesh size $h(tol)$, the optimal

q	$\|\text{error}\|$
1	1.22e-1
2	5.27e-4
3	1.45e-6
4	3.35e-9
5	7.53e-12

Table V.3 – Quadrature error on the optimal control, versus the number of knots q used in each random variable.

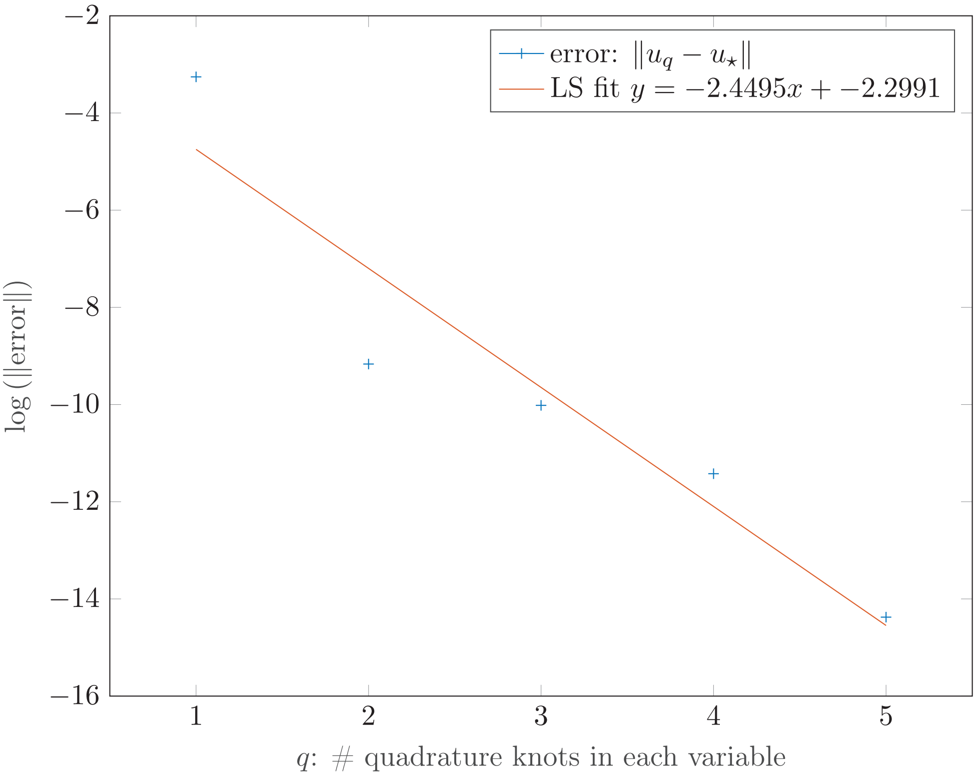


Figure V.6 – Fitting the quadrature (squared) error model $C_2 e^{-sq}$ in (V.51).

number of Gauss-Legendre points in the quadrature formula $q(tol)$, and the optimal number of iterations $k_{max}(tol)$, using the constants C_1, C_2 and C_3 and rates $\epsilon(q)$ and s estimated in the previous subsection. Then we run 20 independent realizations of the SAGA algorithm up to iteration k_{max}, all of them on a mesh of size $h(tol)$, using a quadrature formula with $q(tol)$ points in each random variable, and estimate the average error on the optimal control. The reference solution has been computed using the FG method, on a mesh of size $h = 2^{-9}$, with a quadrature formula with $q = 6$, and for more than 50 iterations. Figure V.7 shows the estimated average error versus the computational cost model $W = k_{max} h^{-d}$, which confirms the complexity result of Corollary 7. Table V.5

h^{-1}	error (FG, $q=1$)	error (SAGA, $q=1$)
2	7.83	7.82
4	2.19	2.19
8	5.46E-01	5.48E-01
16	1.36E-01	1.37E-01
32	3.34E-02	3.42E-02
64	8.34E-03	8.47E-03
128	2.09E-03	2.04E-03

Table V.4 – FE discretization error on the optimal control, using FG, or SAGA, versus the characteristic mesh size h^{-1}.

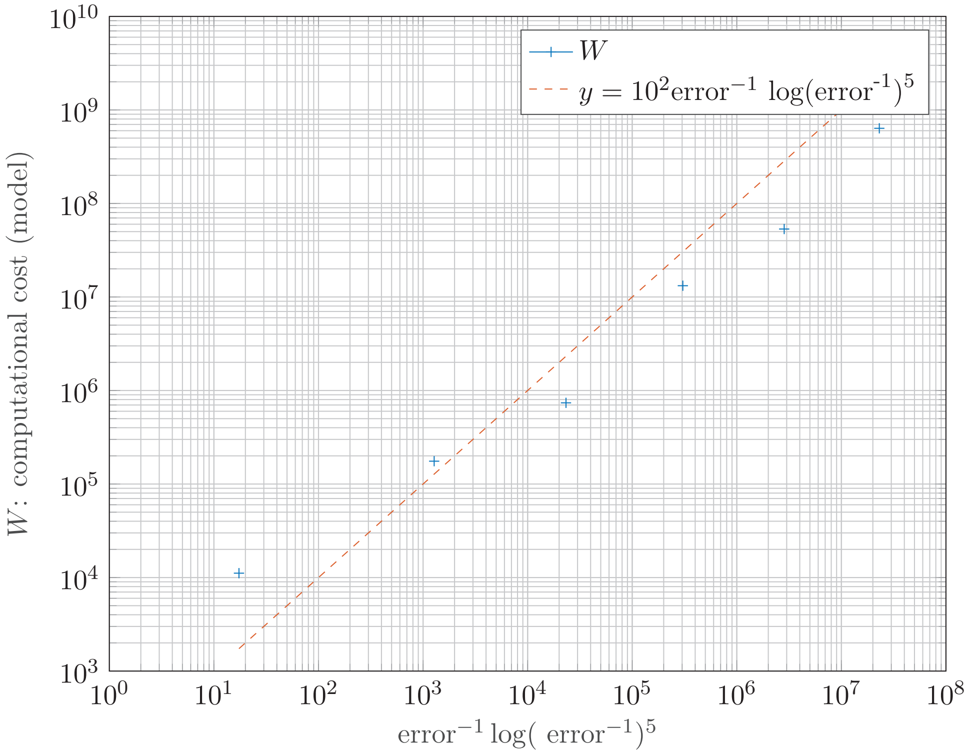


Figure V.7 – Computational cost (model) vs averaged SAGA error.

gives details on the optimal discretization parameters h, q, k_{max} as well as the required memory space, estimated based on the model $M = nh^{-d}$, for each considered tolerance.

The main limitation we see for this method is the required memory space, since the storage increases as the desired tolerance gets smaller and will reach at some point the memory limit of the employed machine.

tol	$1/h$	q	n	k_{max}	W comp. cost	storage	avg error
3.16E-01	11	1	1	46	1.11E+04	1.21E+02	2.59E-01
1.00E-01	19	2	16	243	1.75E+05	5.78E+03	8.02E-02
3.16E-02	33	2	16	339	7.38E+05	1.74E+04	2.68E-02
1.00E-02	59	3	81	1896	1.32E+07	2.82E+05	8.29E-03
3.16E-03	105	3	81	2417	5.33E+07	8.93E+05	2.62E-03
1.00E-03	185	4	256	9298	6.36E+08	8.76E+06	8.00E-04
3.16E-04	329	4	256	18947	4.74E+09	2.77E+07	2.80E-04

Table V.5 – Final error over 20 i.i.d. realizations of SAGA, versus the target tolerance tol.

V.F. Conclusions

In this work, we have proposed a SAGA algorithm to solve numerically a quadratic risk-averse optimal control problem for an elliptic PDE with random coefficients, where the expectation in the objective functional has been approximated by a Gauss-Legendre quadrature formula whereas the elliptic PDE has been discretized by finite elements. The SAGA algorithm is a Stochastic Gradient type algorithm with a fixed-length memory term, which computes at each iteration the gradient of the objective functional in only one quadrature point, randomly chosen from a possibly non-uniform distribution. We have shown that the asymptotically optimal sampling distribution is indeed the uniform one, over the quadrature points. We have also shown that, when equilibrating the three sources of errors, namely the finite element discretization error, the quadrature error and the error due to the SAGA optimization algorithm, the overall complexity, in terms of computational work versus prescribed tolerance, is asymptotically the same as the one of a full gradient method (i.e. a gradient method that sweeps over all quadrature points at each iteration), as the tolerance goes to zero. However, as illustrated by our numerical experiments, the advantage of SAGA with respect to FG is in the pre-asymptotic regime, as acceptable solutions may be obtained already before a full sweep over all quadrature points.

The full tensor Gauss-Legendre quadrature formula considered in this work is affected by the curse of dimensionality, hence applicable only to problems for which the randomness can be described in terms of a small number of random variables. To overcome such curse of dimensionality, one could use sparse quadratures instead [BNT10, LG17, Bor10], whose weights, however, are not all positive. The result in Lemma 28 is still valid, as long as the approximate functional $\widehat{J}$ satisfies a strong convexity condition,

$$\frac{l}{2}\|u_1 - u_2\|^2 \leq \langle \nabla_u \widehat{J}(u_1) - \nabla_u \widehat{J}(u_2), u_1 - u_2 \rangle \quad \forall u_1, u_2 \in U,$$

which might not be guaranteed for a given number of quadrature points. Also, because of the presence of negative weights, the quantity $\widetilde{S}_n$ might not be uniformly bounded in n, therefore, the results in Theorem 32 and Corollary 6 might not apply to this case. These issues will be further investigated in a future work.

VI Conclusions and perspectives

VI.A. Conclusions

In this thesis we analyzed and developed stochastic approximation methodologies to efficiently solve PDE-constrained optimal control problems with uncertain parameters. Specifically, we compared the complexity of different versions of stochastic gradient methods to compute the numerical solutions of mean-based risk-averse OCPs. We started by introducing a finite element discretization, used to approximate the underlying PDEs, as well as a collocation formula, to approximate the expectation in the risk measure, and derived theoretical error bounds on the approximated control.

In the first analyzed algorithm, the FE mesh and a Monte Carlo estimator are chosen initially and kept fixed over the iterations, whereas in a second algorithm, a Stochastic Gradient method is used, with the FE discretization still kept fixed over the iterations; however the expectation in the objective function is re-sampled independently at each iteration, with a fixed small sample size. Then, we generalized such approach into a stochastic gradient method, with successively refined FE meshes over the iterations.

Later we proposed a modified version of the SG algorithm, where the usual Robbins-Monro approach, involving a single realization estimator of the gradient is replaced by either a MLMC estimator, with increasing cost w.r.t. the iteration counter, or a randomized version of the MLMC estimator, where only one difference term of the full MLMC estimator is computed at each iteration, on a randomly drawn level, according to a probability mass function.

Finally, we have proposed a SAGA algorithm to solve numerically the risk-averse OCPs, where the expectation in the objective functional has been approximated by a Gauss-Legendre quadrature formula. The SAGA algorithm is a SG type algorithm, with a fixed-length memory term, which computes at each iteration the gradient of the objective functional in only one quadrature point, randomly chosen.

Our complexity analysis is based on *a priori* error estimates, and *a priori* choices of the FE mesh size, the MC sample size or quadrature formula size, and the maximum number of iterations of the gradient method, to obtain a prescribed tolerance.

After assessing the effectiveness of the stochastic versions of the gradient method, improving the computational complexity by log factors, w.r.t. the full gradient algorithm based on fixed MC estimator, we replace the re-sampled MC estimator, by a re-sampled MLMC estimator, and show that we achieve the optimal complexity $W \lesssim tol^{-2}$.

Lastly, using the SAGA method, we show that the overall complexity, in terms of computational work versus prescribed tolerance, is asymptotically the same as the one of a full gradient method (i.e. a gradient method that sweeps over all quadrature points at each iteration), as the tolerance goes to zero. However, as illustrated by our numerical experiments, the advantage of SAGA with respect to FG is in the pre-asymptotic regime, as acceptable solutions may be obtained already before a full sweep over all quadrature points. It is noteworthy that the full tensor Gauss-Legendre quadrature formula, considered in this work, is affected by the curse of dimensionality, hence applicable only to problems for which the randomness can be described in terms of a small number of random variables.

VI.B. Perspectives

To overcome the curse of dimensionality of the SAGA algorithm that uses a full tensor quadrature formula, one could use sparse quadratures instead [BNT10, LG17, Bor10], whose weights, however, are not all positive, implying that the presented results might not apply to this case.

Another interesting direction is the extension of stochastic gradient methods to more general risk measures. We mention that Stochastic Gradient methods have been already used in combination with the CVaR risk measure [BFP09], although not in the context of PDE-constrained optimal control problems. These issues may be further investigated.

The analysis, in this thesis work, is based on *a priori* error estimates. All tunable parameters have been chosen *a priori*, based on some preliminar computations. For example, the iterations at which we refine the mesh size, in the SG with variable mesh algorithm, or the estimation of the MLMC parameters $q_w, q_s, q_c,$, or the number of levels and number of sample per level, for each iteration counter, are such *a priori* based quantity. This is based on *a priori* error estimates, that are possible due to the fact that we consider a simple elliptic PDE, with a quadratic functional. Nevertheless, in real world problems, the setting would not be as ideal, and one may require to estimate the different error contributions alongside the iterations, i.e. on the fly, and construct adaptative algorithms. This analysis would be based on adaptative error estimates, where the problem parameters, e.g. the MLMC parameters, would be estimated, while iterating the optimization scheme. This construction is postpone to future work.

In this work we have only considered and analyzed the stochastic gradient method. It would be interesting to understand if other optimization methods such as Quasi-Newton of Conjugate Gradient can be reformulated in a stochastic framework, for PDE-constrained OCP under uncertainty.

Finally, in order to accelerate the numerical iterative methods, one could think at introducing preconditioner, in order to reach a problem that is more suitable for numerical computations. How to do this in the context of stochastic gradient iterations is a topic we have not looked into and deserves further investigations.

Bibliography

[AAUH17] A. Ahmad Ali, E. Ullmann, and M. Hinze. Multilevel Monte Carlo analysis for optimal control of elliptic PDEs with random coefficients. *SIAM/ASA J. Uncertain. Quantif.*, 5(1):466–492, 2017. doi:10.1137/16M109870X.

[ADEH99] P. Artzner, F. Delbaen, J.-M. Eber, and D. Heath. Coherent measures of risk. *Math. Finance*, 9(3):203–228, 1999. doi:10.1111/1467-9965.00068.

[APSG17] A. Alexanderian, N. Petra, G. Stadler, and O. Ghattas. Mean-variance risk-averse optimal control of systems governed by PDEs with random parameter fields using quadratic approximations. *SIAM/ASA J. Uncertain. Quantif.*, 5(1):1166–1192, 2017. doi:10.1137/16M106306X.

[Ber97] D. P. Bertsekas. A new class of incremental gradient methods for least squares problems. *SIAM J. Optim.*, 7(4):913–926, 1997. doi:10.1137/S1052623495287022.

[BFP09] O. Bardou, N. Frikha, and G. Pagès. Computing VaR and CVaR using stochastic approximation and adaptive unconstrained importance sampling. *Monte Carlo Methods Appl.*, 15(3):173–210, 2009. doi:10.1515/MCMA.2009.011.

[BG04] H-J. Bungartz and M. Griebel. Sparse grids. *Acta Numer.*, 13:147–269, 2004. doi:10.1017/S0962492904000182.

[BHG07] D. Blatt, A. O. Hero, and H. Gauchman. A convergent incremental gradient method with a constant step size. *SIAM J. Optim.*, 18(1):29–51, 2007. doi:10.1137/040615961.

[BLS13] A. Barth, A. Lang, and C. Schwab. Multilevel Monte Carlo method for parabolic stochastic partial differential equations. *BIT*, 53(1):3–27, 2013. doi:10.1007/s10543-012-0401-5.

[BM11] F. Bach and E. Moulines. Non-asymptotic analysis of stochastic approximation algorithms for machine learning. In *Proceedings of the 24th International Conference on Neural Information Processing Systems*, NIPS'11, pages 451–459, USA, 2011. Curran Associates Inc.

[BNT10] I. Babuška, F. Nobile, and R. Tempone. A stochastic collocation method for elliptic partial differential equations with random input data. *SIAM review*, 52(2):317–355, 2010.

[Bor10] A. Borzì. Multigrid and sparse-grid schemes for elliptic control problems with random coefficients. *Computing and Visualization in Science*, 13(4):153–160, Apr 2010. doi:10.1007/s00791-010-0134-4.

[BOS16] P. Benner, A. Onwunta, and M. Stoll. Block-diagonal preconditioning for optimal control problems constrained by PDEs with uncertain inputs. *SIAM J. Matrix Anal. Appl.*, 37(2):491–518, 2016. doi:10.1137/15M1018502.

[BS12] A. Borzì and V. Schulz. *Computational optimization of systems governed by partial differential equations*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2012.

[BSSvW10] A. Borzì, V. Schulz, C. Schillings, and G. von Winckel. On the treatment of distributed uncertainties in PDE-constrained optimization. *GAMM-Mitteilungen*, 33(2):230–246, 2010. doi:10.1002/gamm.201010017.

[BSZ11] A. Barth, C. Schwab, and N. Zollinger. Multi-level Monte Carlo finite element method for elliptic PDEs with stochastic coefficients. *Numer. Math.*, 119(1):123–161, 2011. doi:10.1007/s00211-011-0377-0.

[BTZ04] I. Babuska, R. Tempone, and G.E. Zouraris. Galerkin finite element approximations of stochastic elliptic partial differential equations. *SIAM Journal on Numerical Analysis*, 42(2):800–825, 2004. doi:10.1137/S0036142902418680.

[BvW11] A. Borzì and G. von Winckel. A POD framework to determine robust controls in PDE optimization. *Comput. Vis. Sci.*, 14(3):91–103, 2011. doi:10.1007/s00791-011-0165-5.

[CD15] A. Cohen and R. DeVore. Approximation of high dimensional parametric PDEs. *Acta Numerica*, 24:1–159, 2015. doi:10.1017/S0962492915000033.

[CGST11] K. A. Cliffe, M. B. Giles, R. Scheichl, and A. L. Teckentrup. Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients. *Comput. Vis. Sci.*, 14(1):3–15, 2011. doi:10.1007/s00791-011-0160-x.

[CQ14] P. Chen and A. Quarteroni. Weighted reduced basis method for stochastic optimal control problems with elliptic PDE constraint. *SIAM/ASA J. Uncertain. Quantif.*, 2(1):364–396, 2014. doi:10.1137/130940517.

[CST13] J. Charrier, R. Scheichl, and A. L. Teckentrup. Finite element error analysis of elliptic PDEs with random coefficients and its application to multilevel Monte Carlo methods. *SIAM J. Numer. Anal.*, 51(1):322–352, 2013. doi:10.1137/110853054.

[DB15] A. Défossez and F. Bach. Averaged least-mean-squares: bias-variance trade-offs and optimal sampling distributions. In *Artificial Intelligence and Statistics*, pages 205–213, 2015.

[DB16] A. Dieuleveut and F. Bach. Nonparametric stochastic approximation with large step-sizes. *The Annals of Statistics*, 44(4):1363–1399, 2016.

[DBLJ14] A. Defazio, F. Bach, and S. Lacoste-Julien. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 1646–1654. Curran Associates, Inc., 2014.

[De 15] J. C. De los Reyes. *Numerical PDE-constrained optimization.* Springer, Cham, 2015. doi:10.1007/978-3-319-13395-9.

[DM15] S. Dereich and T. Mueller-Gronbach. General multilevel adaptations for stochastic approximation algorithms. *arXiv e-prints*, page arXiv:1506.05482, June 2015, 1506.05482.

[EST18] O. G. Ernst, B. Sprungk, and L. Tamellini. Convergence of sparse collocation for functions of countably many Gaussian random variables (with application to elliptic PDEs). *SIAM J. Numer. Anal.*, 56(2):877–905, 2018. doi:10.1137/17M1123079.

[Eva98] L. C. Evans. *Partial differential equations.* Graduate studies in mathematics. American Mathematical Society, 1998.

[FB15] N. Flammarion and F. Bach. From averaging to acceleration, there is only a step-size. In *Conference on Learning Theory*, pages 658–695, 2015.

[Fri16] N. Frikha. Multi-level stochastic approximation algorithms. *Ann. Appl. Probab.*, 26(2):933–985, 2016. doi:10.1214/15-AAP1109.

[FS12] M. P. Friedlander and M. Schmidt. Hybrid deterministic-stochastic methods for data fitting. *SIAM J. Sci. Comput.*, 34(3):A1380–A1405, 2012. doi:10.1137/110830629.

[Gil08] M. Giles. Multilevel Monte Carlo path simulation. *Operations Research*, 56(3):607–617, 2008.

[Gil15] M. Giles. Multilevel Monte Carlo methods. *Acta Numerica*, 24:259–328, 2015.

[GLL11] M. D. Gunzburger, H.-C. Lee, and J. Lee. Error estimates of stochastic optimal Neumann boundary control problems. *SIAM J. Numer. Anal.*, 49(4):1532–1552, 2011. doi:10.1137/100801731.

[GWZ14] M. D. Gunzburger, C. G. Webster, and G. Zhang. Stochastic finite element methods for partial differential equations with random input data. *Acta Numerica*, 23:521–650, 2014. doi:10.1017/S0962492914000075.

[Hac16] W. Hackbusch. *Iterative solution of large sparse systems of equations*, volume 95 of *Applied Mathematical Sciences*. Springer, [Cham], second edition, 2016. doi:10.1007/978-3-319-28483-5.

[HANTT16a] A-L. Haji-Ali, F. Nobile, L. Tamellini, and R. Tempone. Multi-index stochastic collocation convergence rates for random PDEs with parametric regularity. *Found. Comp. Math.*, 16(6):1555–1605, 2016. doi:10.1007/s10208-016-9327-7.

[HANTT16b] A.-L. Haji-Ali, F. Nobile, L. Tamellini, and R. Tempone. Multi-index stochastic collocation for random PDEs. *Comput. Methods Appl. Mech. Engrg.*, 306:95–122, 2016. doi:10.1016/j.cma.2016.03.029.

[HAvST16] A.-L. Haji-Ali, F. , Nobile, E. von Schwerin, and R. Tempone. Optimization of mesh hierarchies in multilevel Monte Carlo samplers. *Stoch. Partial Differ. Equ. Anal. Comput.*, 4(1):76–112, 2016. doi:10.1007/s40072-015-0049-7.

[Haz10] S. B. Hazra. *Large-scale PDE-constrained optimization in applications*. Springer-Verlag, Berlin, 2010. doi:10.1007/978-3-642-01502-1.

[Hec12] F. Hecht. New development in freefem++. *J. Numer. Math.*, 20(3-4):251–265, 2012.

[Hei00] S. Heinrich. The multilevel method of dependent tests. In *Advances in stochastic simulation methods (St. Petersburg, 1998)*, Stat. Ind. Technol., pages 47–61. Birkhäuser Boston, Boston, MA, 2000.

[HPUU09] M. Hinze, R. Pinnau, M. Ulbrich, and S. Ulbrich. *Optimization with PDE constraints*. Mathematical Modelling: Theory and Applications 23. Springer, New York, 2009. doi:10.1007/978-1-4020-8839-1.

[HTF01] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001.

[KHRvBW13] D. P. Kouri, M. Heinkenschloss, D. Ridzal, and B. G. van Bloemen Waanders. A trust-region algorithm with adaptive stochastic collocation for PDE optimization under uncertainty. *SIAM J. Sci. Comput.*, 35(4):A1847–A1879, 2013. doi:10.1137/120892362.

[Kou12] D. P. Kouri. *An approach for the adaptive solution of optimization problems governed by partial differential equations with uncertain coefficients.* ProQuest LLC, Ann Arbor, MI, 2012. Thesis (Ph.D.)–Rice University.

[KS13] A. Kunoth and C. Schwab. Analytic regularity and GPC approximation for control problems constrained by linear parametric elliptic and parabolic PDEs. *SIAM J. Control Optim.*, 51(3):2442–2471, 2013. doi:10.1137/110847597.

[KS16] D. P. Kouri and T. M. Surowiec. Risk-averse PDE-constrained optimization using the conditional value-at-risk. *SIAM J. Optim.*, 26(1):365–396, 2016. doi:10.1137/140954556.

[KS18] D. P. Kouri and T. M. Surowiec. Existence and optimality conditions for risk-averse PDE-constrained optimization. *SIAM/ASA J. Uncertain. Quantif.*, 6(2):787–815, 2018. doi:10.1137/16M1086613.

[KY97] H. J. Kushner and G. G. Yin. *Stochastic approximation algorithms and applications*, volume 35 of *Applications of Mathematics (New York).* Springer-Verlag, New York, 1997. doi:10.1007/978-1-4899-2696-8.

[LBE$^+$14] G. Leugering, P. Benner, S. Engell, A. Griewank, H. Harbrecht, M. Hinze, R. Rannacher, and S. Ulbrich, editors. *Trends in PDE constrained optimization.* Birkhäuser/Springer, Cham, 2014.

[LG17] H.-C. Lee and M. D. Gunzburger. Comparison of approaches for random pde optimization problems based on different matching functionals. *Computers and Mathematics with Applications*, 73(8):1657 – 1672, 2017. doi:10.1016/j.camwa.2017.02.002.

[Lio71] J. L. Lions. *Optimal control of systems governed by partial differential equations.* Grundlehren der mathematischen Wissenschaften. Springer-Verlag, 1971.

[LPS14] G. J. Lord, C. E. Powell, and T. Shardlow. *An introduction to computational stochastic PDEs.* Cambridge Texts in Applied Mathematics. Cambridge University Press, 2014.

[MKN18] M.C. Martin, S. Krumscheid, and F. Nobile. Analysis of stochastic gradient methods for PDE-constrained optimal control problems with uncertain parameters. MATHICSE Technical Report 04.2018, École Polytechnique Fédérale de Lausanne, 2018.

[MN18] M. C. Martin and F. Nobile. PDE-constrained optimal control problems with uncertain parameters using SAGA. arXiv:1808.03112, 2018.

[MN19] M. C. Martin and F. Nobile. Multilevel stochastic gradient method for PDE-constrained optimal control problems with uncertain parameters. In preparation, 2019.

[MSŠ12] S. Mishra, Ch. Schwab, and J. Šukys. Multi-level Monte Carlo finite volume methods for nonlinear systems of conservation laws in multi-dimensions. *J. Comput. Phys.*, 231(8):3365–3388, 2012. doi:10.1016/j.jcp.2012.01.011.

[Mur12] K. P. Murphy. *Machine learning: a probabilistic perspective.* The MIT Press, 2012.

[NJLS09] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization*, 19(4):1574–1609, 2009. doi:10.1137/070704277.

[NT15] F. Nobile and F. Tesei. A multi level Monte Carlo method with control variate for elliptic PDEs with log-normal coefficients. *Stoch. Partial Differ. Equ. Anal. Comput.*, 3(3):398–444, 2015. doi:10.1007/s40072-015-0055-9.

[NTT16] F. Nobile, L. Tamellini, and R. Tempone. Convergence of quasi-optimal sparse grid approximation of Hilbert-space-valued functions: application to random elliptic PDEs. *Numer. Math.*, 134(2):343–388, 2016. doi:10.1007/s00211-015-0773-y.

[NW99] J. Nocedal and S. J. Wright. *Numerical optimization.* Springer Series in Operations Research. Springer-Verlag, New York, 1999. doi:10.1007/b98874.

[PJ92] B. T. Polyak and A.B. Juditsky. Acceleration of stochastic approximation by averaging. *SIAM Journal on Control and Optimization*, 30(4):838–855, 1992.

[Pol90] B. T. Polyak. A new method of stochastic approximation type. *Avtomat. i Telemekh.*, (7):98–107, 1990.

[Pow73] M. J. D. Powell. On search directions for minimization algorithms. *Math. Programming*, 4:193–201, 1973. doi:10.1007/BF01584660.

[Qua09] A. Quarteroni. *Numerical models for differential problems.* MS&A. Springer, Milano, 2009.

[RG12] C.-H. Rhee and P. W. Glynn. A new approach to unbiased estimation for SDE's. In *Proceedings of the Winter Simulation Conference*, WSC '12, pages 17:1–17:7. Winter Simulation Conference, 2012.

[RG15] C.-H. Rhee and P. W. Glynn. Unbiased estimation with square root convergence for SDE models. *Oper. Res.*, 63(5):1026–1043, 2015. doi:10.1287/opre.2015.1404.

[RM51] H. Robbins and S. Monro. A stochastic approximation method. *Ann. Math. Statist.*, 22(3):400–407, 09 1951. doi:10.1214/aoms/1177729586.

[RS06] A. Ruszczyński and A. Shapiro. Optimization of convex risk functions. *Math. Oper. Res.*, 31(3):433–452, 2006. doi:10.1287/moor.1050.0186.

[RS07] A. Ruszczyński and A. Shapiro. Corrigendum to: “Optimization of convex risk functions” [Math. Oper. Res. **31** (2006), no. 3, 433–452. *Math. Oper. Res.*, 32(2), 2007. doi:10.1287/moor.1070.0265.

[RSN90] F. Riesz and B. Sz.-Nagy. *Functional analysis.* Dover Books on Advanced Mathematics. Dover Publications, Inc., New York, 1990. Translated from the second French edition by Leo F. Boron, Reprint of the 1955 original.

[RU02] R. T. Rockafellar and S. Uryasev. Conditional value-at-risk for general loss distributions. *J. Bank. Financ.*, 26(7):1443–1471, 2002. doi:10.1016/S0378-4266(02)00271-6.

[Rup88] D. Ruppert. Efficient estimations from a slowly convergent robbins-monro process. Technical report, Cornell University Operations Research and Industrial Engineering, 1988.

[RW12] E. Rosseel and G. N. Wells. Optimal control with stochastic PDE constraints and uncertain controls. *Comput. Methods Appl. Mech. Engrg.*, 213/216:152–167, 2012. doi:10.1016/j.cma.2011.11.026.

[SBA$^+$] M. W. Schmidt, R. Babanezhad, M. O. Ahmed, A. Defazio, A. Clifton, and A. Sarkar. Non-uniform stochastic average gradient method for training conditional random fields. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2015, San Diego, California, USA, May 9-12, 2015.*

[SDR09] A. Shapiro, D. Dentcheva, and A. Ruszczyński. *Lectures on stochastic programming.* Society for Industrial and Applied Mathematics, 2009. doi:10.1137/1.9780898718751.

[SRB13] M. W. Schmidt, N. Le Roux, and F. R. Bach. Minimizing finite sums with the stochastic average gradient. *CoRR*, abs/1309.2388, 2013.

[SS13] C. Schillings and C. Schwab. Sparse, adaptive Smolyak quadratures for Bayesian inverse problems. *Inverse Problems*, 29(6):065011, 28, 2013. doi:10.1088/0266-5611/29/6/065011.

[SSS11] C. Schillings, S. Schmidt, and V. Schulz. Efficient shape optimization for certain and uncertain aerodynamic design. *Computers & Fluids*, 46(1):78 – 87, 2011. doi:10.1016/j.compfluid.2010.12.007. 10th ICFD Conference Series on Numerical Methods for Fluid Dynamics (ICFD 2010).

[Sze39] G. Szeg. *Orthogonal polynomials*, volume 23. American Mathematical Soc., 1939.

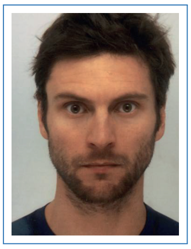
[TKXP12] H. Tiesler, R. M. Kirby, D. Xiu, and T. Preusser. Stochastic collocation for optimal control problems with stochastic PDE constraints. *SIAM J. Control Optim.*, 50(5):2659–2682, 2012. doi:0.1137/110835438.

[TSGU13] A. L. Teckentrup, R. Scheichl, M. B. Giles, and E. Ullmann. Further analysis of multilevel Monte Carlo methods for elliptic PDEs with random coefficients. *Numer. Math.*, 125(3):569–600, 2013. doi:10.1007/s00211-013-0546-4.

[VBV18] A. Van Barel and S. Vandewalle. Robust optimization of pdes with random coefficients using a Multilevel Monte Carlo method. *SIAM-ASA Journal on Uncertainty Quantification*, 2018.

[Wil91] D. Williams. *Probability with martingales*. Cambridge mathematical textbooks. Cambridge University Press, 1991.

[ZDS18] J. Zech, D. Dung, and C. Schwab. Multilevel approximation of parametric and stochastic PDEs. Technical Report SAM Report 2018-05, ETHZ, 2018.



178 rue de la tour
73 140 St Martin de la Porte, France
☎ +33 (0)6 42 64 08 66
✉ matthieu.martin.mm@gmail.com
single, 29 years old, french nationality

Matthieu MARTIN

Education

07.2015–03.2019 **PhD in applied mathematics**, *École Polytechnique Fédérale de Lausanne*, Lausanne, Switzerland.
Title: Stochastic approximation methods for PDE-constrained optimal control problems with uncertain parameters. Focus: Machine learning optimization techniques, Multi Level Monte Carlo approximation, High dimensional collocation approximation.

09.2010–08.2014 **Master degree in applied mathematics**, *École Centrale de Paris*, Paris, France.
top French engineering school in **optimization, data mining** and **finance**

09.2013–08.2014 **Master degree in mathematics**, *Pierre and Marie Curie University – École Polytechnique*, Paris, France.
Major: **game theory**, **optimization** and **economics**

09.2007-08.2010 **Classes préparatoires MPSI/MP***, *Lycée Louis le Grand*, Paris, France.
Preparatory school for national competitive exams to enter French engineering Schools, first year in Lycée du Parc

09.2004–08.2007 **French Baccalaureate, Scientific Program**, *Lycée Vaugelas*, Chambéry, France.
16.9/20 cum Laude

Work Experience

09.2014–06.2015 **Data scientist**, *Ridesurfing*, Sydney, Australia.
Business intelligence on different projects such as forecasting client demand to anticipate our supply. Enhance the **user experience** and business outcomes using cohort analysis. Investigate patterns in the data using machine learning. Develop a suite of KPI reports on forecasting efficiency.

02.2013–10.2013 **Data Scientist**, *Coregas*, Yennora NSW, Australia.
Worked as an intern on different projects such as **forecasting client demand** to anticipate our supply or dynamic optimization between depots and branches to reduce transport cost. Enhance the user experience and business outcomes. Investigate patterns in the data using machine learning. Develop a suite of KPI reports on scanning efficiency by branch. (**Data-Mining, Predictive Analytics**)

06.2012–01.2013 **Data Analyst**, *Yogiplay*, Menlo Park, CA, USA.
Implement as an intern machine learning methods on the user's data, crossing Google analytics with server log files to find some user practices and improve our applications for devices. Business intelligence: AB-testing **optimization to increase retention**. Use of Pentaho Data Integration, MySQL and shell script (awk) in log files to extract useful information/typical behavior. Cronjobs and automation.

09.2010–08.2011 **Pricing project Internship**, *Legrand Company*, Paris, France.
Company project to expand into domestic automation market. Study of the current pricing of electronic components to better manage inventory, optimize its supply and ultimately to minimize its costs. Final stage of « Project enjeux » competition at École Centrale de Paris.

Publications

1. M. C. Martin, S. Krumscheid, and F. Nobile. Analysis of stochastic gradient methods for PDE-constrained optimal control problems with uncertain parameters. Submitted, 2018.
2. M. C. Martin and F. Nobile. PDE-constrained optimal control problems with uncertain parameters using saga. Submitted, 2018.
3. M. C. Martin and F. Nobile. Multilevel stochastic gradient method for PDE-constrained optimal control problems with uncertain parameters. In preparation, 2019.

Presentations in Conferences and Workshops

01.2016 **Poster presentation**, *at SRI UQ WORKSHOP 2016*, Kaust, UAE.

28.04.2017 **Poster presentation**, *at Colloque Numérique Suisse / Schweizer Numeric Colloquium*, Basel, Switzerland.

18.07.2017–21.07.2017 **Poster presentation**, *QUIET 2017 - Quantification of Uncertainty: Improving Efficiency and Technology*, Trieste, Italy, SISSA, International School for Advanced Studies, Main Campus.

6.09.2017–8.09.2017 **Talk**, *at the Workshop Frontiers of Uncertainty Quantification in Engineering (FrontUQ)*, Munich, Germany.

28.05.2018–01.06.2018 **Talk**, *44e Congrès National d'Analyse Numérique (CANUM)*, Cap d'Agde, France.

06.2018 **Conference**, *13th World Congress in Computational Mechanics*, NYC, USA.

Conferences, Workshops, and Schools attended

01.2017 **Winter school**, *on Optimization*, Zinal, Switzerland.

31.01.2017–1.02.2017 **Workshop**, *on Multiscale methods for stochastic dynamics*, Geneva, Switzerland.

7.02.2017–10.02.2017 **Winter School**, *on Optimal Control*, Engelberg, Switzerland.

11.2017 **Workshop**, *46th SpeedUp Workshop on ”Uncertainty Quantification and HPC”*, University of Bern, Switzerland.

01.2018 **Winter school**, *on Optimization*, Zinal, Switzerland.

06.2018 **Workshop**, *Analysis of Adaptive Stochastic Gradient and MCMC Algorithms*, Alan Turing Institute, London, England.

Courses taken during the PhD

01.2016 **Optimization Methods and Models**, *grade: 6*, credits: 4.

01.2016 **Optimal control of systems governed by PDEs**, *grade: Passed*, credits: 2.

01.2016 **High Dimensional Approximation for PDEs with random parameters**, *grade: 5*, credits: 2.

01.2016 **Deep learning**, *grade: 5*, credits: 4.

Languages

English	Fluent	
French	Native speaker	
German	Reading & Conversational	*5 months spent in Berlin with Erasmus program*

Computer Skills

Linux, MacOS, Windows	extensive		
Python	extensive, incl. PyTorch & Tensorflow	Matlab	extensive
Obj. Caml	good	Php	good
LaTeX	extensive	C/C++	good
R	good	Freefem++	extensive

Extracurricular activity

Sport	Road bike, Triathlon, Ski touring, in competition
Music	played the Tuba for 12 years at Chambéry conservatory

Ce document a été imprimé au Centre d'impression EPFL,
imprimerie climatiquement neutre, certifiée myClimate.