Path Integral Control

Proefschrift

ter verkrijging van de graad van doctor
aan de Radboud Universiteit Nijmegen
op gezag van de rector magnificus prof. dr. J.H.J.M. van Krieken,
volgens besluit van het college van decanen
in het openbaar te verdedigen op
maandag 28 november 2016,
om **16.30 uur** precies

door

Sep Anton Thijssen

geboren op 27 juli 1983
te Groningen

Promotor:

Prof. dr. H.J. Kappen

Manuscriptcommissie:

Prof. dr. E.A. Cator	
Prof. dr. J.D.M. Maassen	Universiteit van Amsterdam
Dr. G.N.J.C. Bierkens	Technische Universiteit Delft

This work was funded by Thales Research & Technology NL, and supported by the European Community Seventh Framework Programme (FP7/2007-2013) under grant agreement 270327 (CompLACS).

© Sep Thijssen
ISBN 978-94-6284-079-9
Uitgeverij BOXPress, *Proefschriftmaken.nl*

Preface

I discovered that finishing a PhD is not easy. I owe a lot of people for the time, money and effort to make this work possible.

Bert, I learned a lot from you. You taught me about machine learning, control theory and related scientific topics. Also, you taught me many things outside our field of science that were crucial when doing research; most of these things I would never have figured out myself. Thank you for your trust, patience and the freedom you gave me.

Vicenç and Andrew, thank you for the cooperation on the project with the quadcoptors, which has led to a direct contribution to my thesis. It was a real multi agent experience, where all of us were dependent on the others.

I would like to thank all my colleagues at the Radboud University Nijmegen. I really enjoyed the active atmosphere at the Biophysics department, where there is always someone up for a discussion, both on- and off-topic. Special thanks go to my colleagues in SNN. Annet, Willem, Wim, Bart, Mohammed, Alberto, Vicenç, Kevin, Joris, Dominik and Hans, thank you for your company and for all that you taught me.

Also, special thanks go to Joris for all the valuable comments on my work. Although you are not listed as co-promotor, you really have been a tutor to me. As a fellow mathematician you knew how to inspire me ("you should read these text books"), leading to my crucial first result in path integral control.

The support I received outside of academia is perhaps just as important, and I would like to thank all my friends and family for their company. Leonie, I love you. Karel is awesome.

Sep,
October 2016

Contents

Preface iii

Page

1 Introduction 1

2 Stochastic optimal control 5
2.1 Introduction . . . 5
2.2 Definition . . . 6
2.3 The Hamilton-Jacobi-Bellman equation . . . 7

3 Path integral control theory 9
3.1 Introduction . . . 9
3.2 Definition . . . 10
3.3 Alternative formulation . . . 12
3.4 Linearization of the HJB . . . 14
3.5 The optimal cost is not random . . . 16
3.6 A path integral for the optimal expected cost . . . 17
3.7 A path integral for the optimal control . . . 18
3.8 A path integral for the optimal control gradient . . . 19
3.9 Path integral variance . . . 22

4 Kullback Leibler control theory 27
4.1 Introduction . . . 27
4.2 Definition . . . 28
4.3 KL solution . . . 28
4.4 KL and path integral control . . . 30

5 Adaptive multiple importance sampling 33
5.1 Introduction . . . 33
5.2 The generic AMIS . . . 35
5.3 Consistency of flat-AMIS . . . 37
5.4 AMIS with discarding . . . 38

5.5 Consistent AMIS for diffusion processes 41
5.6 The choice of proposal 42
5.7 Numerical example 43
5.8 Proofs and definitions 47

6 The path integral control algorithm **49**
6.1 Introduction 49
6.2 Parametrized control 50
6.3 Control computations 51
6.4 Example 53

7 Real-time stochastic optimal control for multi-agent quadrotor systems **57**
7.1 Introduction 57
7.2 Related work on UAV planning and control 59
7.3 Path integral control for multi-UAV planning 61
7.3.1 Path integral control 61
7.3.2 Multi-UAV planning 62
7.3.3 Low level control 64
7.3.4 Simulator platform 65
7.4 Results 65
7.4.1 Scenario I: Drunken Quadrotor 66
7.4.2 Scenario II: holding pattern 68
7.4.3 Scenario III: cat and mouse 71
7.5 Conclusions 72

Bibliography **81**

Index **83**

Summary **85**

Samenvatting **87**

Curriculum Vitae **89**

Chapter 1

Introduction

Using an external input in order to move a system into a desired state is a very common objective that naturally occurs in many areas of science. It arises in a wide variety of practical problems. In robotics, the problem may be to plan a sequence of actions that yield a motor behavior such as walking or grasping an object [RTM+12a, KUD13]. In finance, the problem may be to devise a sequence of buy and sell actions to optimize a portfolio of assets, or to determine the optimal option price [GH99]. In many of these situations one faces the extra difficulty of uncertainty due to model imperfections and unpredictable external influences. The theory of stochastic optimal control studies such problems.

The dynamic programming approach to stochastic optimal control problem yields a partial differential equation that is known as the Hamilton-Jacobi-Bellman (HJB) Equation [Øks85, FS06]. In general the HJB is impossible to solve analytically, and numerical solutions are intractable due to the curse of dimensionality. One way to proceed is to consider the class of control problems in which the HJB can be linearized, and, consequently, expressed as a path integral [Kap05a]. This approach has led to efficient computational methods that have been successfully applied to control, for example, multi agent systems and robot movement [BWK08a, AM12, SM11, TBS10]. Despite it's success, some key aspects of path integral control have not yet been addressed, such as:

- The optimal control depends on the state, and in a stochastic setting the future state is uncertain. As a consequence, the optimal control is a so called feedback function: it needs to react to random disturbances of the system. Practical applications of path integral control methods to, for instance, robotics have largely ignored this issue and the resulting 'open loop' controllers are independent of state. It is well-known that such a control solution is not stable with respect to disturbances.
- One of the main challenges in applications is to obtain a good numerical

estimate of the path integrals involved. This can be a very difficult task because of the weighting with path costs, which can severely reduce the number of effective sample paths. To mitigate this, it has been suggested to use an exploring control, which introduces an importance sampling scheme. Numerical evidence shows that this can improve sampling [Kap05b] but there are no theoretical results to back this up.

- A very efficient way to implement importance sampling is via an adaptive and mixed scheme known as Adaptive Multiple Importance Sampling (AMIS), [CMMR12]. In AMIS, samples are generated sequentially: at iteration k we draw N_k samples from a proposal distribution P^{u_k} that is parametrized by u_k. The idea of AMIS is that the parameters u_k are adapted sequentially, perhaps using samples from previous iterations, such that the successive proposals improve over successive iterations. Although AMIS has been shown to be a very efficient sampling method, the consistency for the AMIS estimator has only been established in a very restricted case [MPS12], where it is assumed that the parameter u_k is only updated using the last N_k samples (instead of all $N_1+N_2+\ldots+N_k$ samples at stage k), and N_k is assumed to grow to infinity as k does. Furthermore, the high computational complexity of the re-weighting in AMIS makes it unsuitable for applications involving diffusion processes, and hence for path integral control.

- A satisfactory numerical approximation of a path integral can require millions of sample paths. In contrast, it might be prohibitively expensive to obtain a hundred samples, especially when said samples need to be drawn using real world experiments. Even computer simulations might not be able to satisfy the demand for samples if a complex problem is to be solved in real time.

In this thesis we will report recent theoretical and practical developments that address these problems. These will then be applied to investigate whether path integral methods can be used to control multiple agents in complicated cooperative tasks. Below follows a short description of the Chapters in this thesis.

In Chapter 2 we give an introduction to stochastic optimal control theory, of which path integral control is a special case. We give a derivation of the HJB Equation that is used to build the path integral control theory in Chapter 3.

In Chapter 3 we extend the theory of path integral control. We prove a theorem – the Main Path Integral Control Theorem – which can be used to construct parametrized state dependent feedback controllers. The optimal parameters can be expressed in terms of the generalized path integral formulas. Furthermore, we derive an upper bound on the variance of the path weights in terms of distance between the exploring- and optimal control. This means that both the control and

the sampling problem have the same solution. As a consequence, we can iteratively improve control estimates with an increasingly effective sampling procedure.

Connected to this Chapter is the publication [TK15] by the author. We furthermore included a unpublished result about the so called second order path integral in Section 3.8.

In Chapter 4 we expand the theory of path integral control with results from the more general notion of KL-control. This chapter forms a bridge between the path integral control problem, as treated in Chapter 3, and the problem of efficient Monte Carlo sampling, as treated in Chapter 5.

Included in this chapter is a new proof of the Main Path Integral Control Theorem by means of the Girsanov Theorem.

In Chapter 5 we consider sequential and adaptive importance sampling for diffusion processes. One of the key differences with standard AMIS from [CMMR12], is that we propose to use a different re-weighting scheme. Whereas standard AMIS uses the balance heuristic[1] [VG95, OZ00] for re-weighting, which modifies the denominator of the importance weights to a mixture of all proposal distributions, we propose a much simpler discarding-re-weighting scheme. The discarding-re-weighting is of lower computational complexity than balance heuristic re-weighting, and we prove that the resulting AMIS estimate is consistent. Using numerical experiments, we demonstrate that discarding-re-weighting performs very similar to the balance heuristic, but at a fraction of the computational cost.

Connected to this Chapter is the intended publication [TK16] by the author.

In Chapter 6 we combine the results from Chapter 3 and 5 in order to formulate the Path Integral Control Algorithm.

In Chapter 7 we investigate whether path integral methods can be used to control multiple agents in complicated cooperative tasks. In a complicated scenario it is tempting to define subtasks, solve them, and combine those by hand. Unfortunately this leads to problem specific solutions. Instead we show that cooperative strategies can also arise automatically when using one joint target cost function for all agents.

More specifically, we present in Chapter 7 a novel method for controlling teams of unmanned aerial vehicles using Stochastic Optimal Control (SOC) theory. The approach consists of a centralized high-level planner that computes optimal state trajectories as velocity sequences, and a platform-specific low-level controller which ensures that these velocity sequences are met. The planning task is expressed as a centralized path-integral control problem, for which optimal control computation corresponds to a probabilistic inference problem that can be solved by efficient sampling methods. Through simulation we show that our SOC approach (a) has

[1] Balance heuristic is also referred to as *deterministic multiple mixture* in the literature.

significant benefits compared to deterministic control and other SOC methods in multimodal problems with noise-dependent optimal solutions, (b) is capable of controlling a large number of platforms in real-time, and (c) yields collective emergent behavior in the form of flight formations. Finally, we show that our approach works for real platforms, by controlling a team of three quadrotors in outdoor conditions.

Connected to this Chapter is the publication [GTS+15].

Chapter 2

Stochastic optimal control

2.1 Introduction

In stochastic optimal control we try to minimize a cost function that is constrained by a controlled and randomly disturbed dynamical system. In contrast to a deterministic system, the path of future states of a randomly disturbed system cannot be deduced by the initial state. This has an important consequence: since the optimal control input depends on the system state, it will be a feedback controller, whereas in deterministic control, the optimal solution can be described by an open-loop feed forward control signal that only depends on the initial state.

The dynamics, that describe the evolution of the states, will be given as a stochastic differential equation (SDE). A realization of the SDE is sometimes called a (random) path. Each path has an attributed cost, which is a random variable depending both on the path and the control input. The dependence on path and control in the cost will in general give rise to conflicting optimization targets. For example, there might be a large cost for strong control inputs, while at the same time a strong control input might decrease the path dependent part of the cost.

In this Chapter we will define the stochastic optimal control problem for finite time horizon diffusion processes. In order to solve the control problem we take a formal approach using the dynamic programming method, similar as in [FR75, FS06, Øks85], which will lead to a differential equation, know as the Hamilton Jacobi Bellman Equation.

We will assume that the reader is to some degree familiar with concepts such as Brownian motion, stochastic differential equation and stochastic (Itô) integral. We advice [Øks85, KS91] for more background on stochastic calculus.

2.2 Definition

In this section we define the finite time horizon stochastic optimal control problem. Such a problem is given in terms of a stochastic differential equation (SDE) that depends on a control and a cost term, expressed in terms of the solution of the SDE, that should be minimized with respect to the control.

More precisely, we consider the following controlled n-dimensional SDE

$$dX_t^u = \mu^u\big(t, X_t^u\big)dt + \sigma^u\big(t, X_t^u\big)dW_t, \tag{2.1}$$

with $t \in [t_0, t_1]$, initial condition $X_{t_0}^u = x_0 \in \mathbb{R}^n$, and W_t a standard m-dimensional Brownian motion. The functions $\mu^u : [t_0, t_1] \times \mathbb{R}^n \to \mathbb{R}^n$ and $\sigma^u : [t_0, t_1] \times \mathbb{R}^n \to \mathbb{R}^{n\times m}$ should be such that a solution $(X_t^u)_{t_0 \le t \le t_1}$ of the SDE exists; for conditions that ensure this, see for example [FR75, FS06]. The superscript u that appears in the SDE is a Markov feedback control law, [FS06, Øks85], $u = u(t, x)$, which is a function $u : [t_0, t_1] \times \mathbb{R}^n \to \mathbb{R}^k$. The coefficients of the SDE are such that $\mu^u(t, x) = \mu(t, x, u(t, x))$ and $\sigma^u(t, x) = \sigma(t, x, u(t, x))$.

For a given Markov control law u we define the cost-to-go as

$$S_t^u = \Phi\left(X_{t_1}^u\right) + \int_t^{t_1} L\left(\tau, X_\tau^u, u(\tau, X_\tau^u)\right) d\tau.$$

Here $L(\cdot,\cdot,\cdot)$ is know as the immediate cost-function, and $\Phi(\cdot)$ as the end-cost-function. We remark that $\left(S_t^u\right)_{t\in[t_0,t_1]}$ is a random process that is not adapted, which means, roughly speaking, that at time t the cost S_t is not independent of future states X_τ^u for $\tau > t$, see also [KS91, Øks85].

Next, if it exists, we define the expected cost to go function as

$$J^u(t, x) = \mathbb{E}\left[S_t^u \mid X_t^u = x\right].$$

The goal in stochastic optimal control is to minimize the expected cost with respect to the control:

$$\begin{aligned} J^\star(t, x) &= \min_u J^u(t, x), \\ u^\star(\cdot,\cdot) &= \arg\min_u J^u(t_0, x_0). \end{aligned}$$

Generally speaking, there is a two step approach in order to ensuring that a minimum $J^\star$, and corresponding minimizer $u^\star$ exist. In the first step, one assumes that there exists a solution of the dynamic programming equation (see Eq. (2.6)). If this solution is suitably well behaved, then it also solves the minimization problem. In the literature this is called a verification theorem [FR75, Øks85] and it can be interpreted as a proof of *sufficiency* of the the dynamic programming equation. In the second step, one gives conditions that ensure that a solution of the dynamic programming

equation exist. Unfortunately such conditions are very restrictive [FS06] and exclude many systems of interest. In this work, however, we will mostly be interested in a special case of of stochastic optimal control, called path integral control, see Chapter 3. For path integral control, and the related KL-control, see Chapter 4, it is much simpler to state necessary and sufficient conditions for existence of a solution, see [BK14] for an extensive treatment of this subject. Because the focus of this work is on path integral control, we sill skip these issues in the general case, and simply assume that $J^\star$ and $u^\star$ exist.

Based on existence of $u^\star$ and $J^\star$, we will present in the next section a formal derivation of the dynamic programing equation, which can be interpreted as a proof of its *necessity*.

2.3 The Hamilton-Jacobi-Bellman equation

We continue with a formal derivation of the Hamilton-Jacobi-Bellman (HJB) equation for the stochastic optimal control problem that is given in the previous section. The derivation is similar as those given in [Øks85, FR75]. The starting point for this derivation, is the assumption that the process $(X_t^u)_{t_0 \le r \le t_1}$, as well as the optimal cost- and control-functions $J^\star$ and $u^\star$, exist and are suitably well behaved.

A great deal of information about a stochastic processes $(X_t^u)_{t_0 \le t \le t_1}$ from Eq. (2.1) is encoded by the backward evolution operator $\mathscr{A}$, known as the (infinitesimal) generator. The operator $\mathscr{A}$ is defined on functions $\phi : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}$ of class $\mathscr{C}^{1,2}$, i.e. that are twice differentiable with continuous second order derivatives in the second (state) variable, and continuously differentiable with respect to the first (time) variable). $\mathscr{A}$ is defined by

$$\mathscr{A}^u \phi = \lim_{h \downarrow 0} h^{-1} \big(\mathbb{E}\big[\phi(t+h, X_{t+h}^u) \,|\, X_t^u = x \big] - \phi(t,x) \big).$$

Because in our case the underlying process X_t^u is a diffusion process, $\mathscr{A}^u$ takes the form of a second order partial differential operator, see [KS91, Øks85] for details,

$$\mathscr{A}^u \phi = \partial_t \phi + (\partial_x \phi)^\top \mu^u + \frac{1}{2} \operatorname{trace}\big(\sigma^u (\sigma^u)^\top (\partial_{xx} \phi) \big), \tag{2.2}$$

where $()^\top$ denotes the transpose, $\partial_{xx}\phi$ denotes the Hessian of ϕ, and μ^u and σ^u are the coefficients from Eq. (2.1).

Substituting a random state X_t^u for x in $J^\star(t,x)$, we obtain a stochastic process $J_t^\star = J^\star(t, X_t^u)$. If we assume that $J^\star \in \mathscr{C}^{1,2}$, then according to Itô's Lemma [Øks85, KS91, FR75] $J_t^\star$ satisfies the following SDE, which involves the generator

$$dJ_t^\star = \mathscr{A}^u J_t^\star dt + (\partial_x J_t^\star)^\top \sigma^u(t, X_t^u) dW_t, \tag{2.3}$$

where we use the notation $\mathscr{A}^u J_t^\star = (\mathscr{A}^u J^\star)\big(t, X_t^u\big)$ and $\partial_x J_t^\star = (\partial_x J^\star)\big(t, X_t^u\big)$.

Now we put Eq. (2.3) in integral notation (over times $[t,s]$) and take the expected value conditioned on $X_t^u = x$, so that

$$\mathbb{E}\left[J^\star(s,X_s^u)\right] = J^\star(t,x) + \mathbb{E}\int_t^s \mathscr{A}^u J_r^\star dr + \int_t^s (\partial_x J_r^\star)^\top \sigma_r^u dW_r.$$

Assuming that $\mathbb{E}\int_t^s \|(\partial_x J_r^\star)^\top \sigma_r^u\|^2 dr < \infty$, the stochastic integral is a Martingale [Øks85], such that

$$\mathbb{E}\left[J^\star(s,X_s^u)\right] = J^\star(t,x) + \mathbb{E}\int_t^s \mathscr{A}^u J_r^\star dr. \tag{2.4}$$

This result is know as Dynkin's formula, see also [Øks85].

Define the feedback control function $\tilde{u}(r,y)$ by

$$\tilde{u}(r,y) = \begin{cases} u(r,y) & \text{if } t \le r \le s \\ u^\star(r,y) & \text{if } s < r \end{cases}. \tag{2.5}$$

Using optimality of $J^\star$ and conditional expectation, we get according to the dynamic programming principle that

$$J^\star(t,x) \le J^{\tilde{u}}(t,x) = \mathbb{E}\left[J^\star(s,X_s^u)\right] + \mathbb{E}\int_t^s L\left(r,X_r^u,u_r\right) dr,$$

where we have denoted $u_r = u\left(r,X_r^u\right)$. Intuitively, the dynamic programming equation says that the expected cost on the right-hand side is sub-optimal because it used sub-optimal controls from time t to time s. As a consequence, this cost is larger than the optimal cost (on the right-hand side). Note that the above is an equality when $u = u^\star$. Combining this with (2.4) we get

$$0 \le \mathbb{E}\int_t^s L_r + \mathscr{A}^u J_r^\star dr,$$

where we have denoted $L_r = L\left(r,X_r^u,u_r\right)$. The expectation is over an integral that depends on the process X_r^u form time t to s. In order to get rid of the expectation we divide by $s-t$ and take the limit $s \downarrow t$, which results in

$$0 \le L(t,x,u(t,x)) + \mathscr{A}^u J^\star(t,x).$$

Recall that this is an equality when $u = u^\star$. We now have derived the following partial differential equation, known as the Hamilton-Jacobi-Bellman (HJB) equation, for stochastic optimal control

$$0 = \min_u \left[L(t,x,u) + \mathscr{A}^u J^\star(t,x)\right], \tag{2.6}$$

that should be solved with the end condition $J^\star(t_1,x) = \Phi(x)$. In general the HJB equation is impossible to solve analytically, and numerical solutions are intractable due to the problem of dimensionality. In order to proceed we will consider in Chapter 3 the class of control problems in which the HJB equation can be linearized.

Chapter 3

Path integral control theory

3.1 Introduction

Optimal control theory provides an elegant mathematical framework for obtaining an optimal controller using the Hamilton-Jacobi-Bellman (HJB) equation. In general the HJB equation is impossible to solve analytically, and numerical solutions are intractable due to the problem of dimensionality. As a result, often a suboptimal linear feedback controller such as a proportional-integral-derivative (PID) controller [Ste94] or another heuristic approach is used instead. The use of suboptimal controllers may be particularly problematic for nonlinear stochastic problems, where noise affects the optimality of the controller.

One way to proceed is to consider the class of control problems in which the HJB equation can be linearized. Such problems can be divided into two closely related cases [TT12]. The first considers infinite-time-average cost problems, while the second considers finite-time problems. Approaches of the first kind [FM95, Fle82] solve the control problem as an eigenvalue problem. This class has the advantage that the solution also provides a feedback signal, but the disadvantage that a discrete representation of the state space is required, [HDB14, KUD13]. In the second case the optimal control solution is given as a path integral [Kap05a]. This case will be the subject of this chapter. Path integral approaches have led to efficient computational methods that have been successfully applied to multiagent systems and robot movement [RTM+12a, BWK08a, AM12, SM11, TBS10].

Despite its success, two key aspects have not yet been addressed.

1. The issue of state feedback has been largely ignored in path integral approaches and the resulting "open-loop" controllers are independent of the state; they are possibly augmented with an additional PID controller to ensure stability [RTM+12a].

2. The path integral is computed using Monte Carlo sampling. The use of an exploring control as a type of importance sampling has been suggested to improve the efficiency of the sampling [Kap05b, GH99] but there appear to be no theoretical results to back this up.

These two aspects are related because the exploring controls are most effective if they are state feedback controls. In this chapter we propose solutions to these two issues. To achieve this, we derive a path integral control formula that can be utilized to construct parametrized state-dependent feedback controllers. In Chapter 6 we show how a feedback controller might be obtained using path integral control computations that in a sense approximates the optimal control within the limits of the parametrization. The parameters for all future times can be computed using a single set of Monte Carlo samples.

We derive the key property that the path integral is independent of the importance sampling when using infinite samples. However, importance sampling strongly affects the efficiency of the sampler. In Theorem 3.12 we derive a bound which implies that, when the importance control approaches the optimal control, the variance in the estimates reduces to zero and the effective sample size becomes maximal. This allows us to improve the control estimates sequentially by using better and better importance sampling with increasing effective sample size.

Outline. This chapter is structured as follows. In Section 3.2 and 3.3 we review path integral control and we extend the existing theory in Section 3.4 and 3.5. In Sections 3.6 to 3.8 we use the new theory to derive three path integral formulas. Furthermore, in Section 3.9, we derive upper- and lower bounds on the variance of the weights that appear in the path integral formulas.

The main application of the theory in this chapter is of such importance that we treat it separately in Chaper 6. There we will apply the path integral formula from Section 3.7 in order to construct a feedback controller, and describe how to compute it efficiently.

Most of the new results in this Chapter are based on [TK15].

3.2 Definition

The classical Path Integral control problem that we shall consider is a special instance of a control problem as defined in Section 2.2, where the dynamics and cost(-to-go)

are of the form

$$dX_t^u = b\left(t, X_t^u\right) dt + \sigma\left(t, X_t^u\right)\left[\left(u(t, X_t^u\right) dt + dW_t\right], \tag{3.1}$$

$$S_t^u = Q\left(X_{t_1}^u\right) + \int_t^{t_1} R\left(\tau, X_\tau^u\right) + \frac{1}{2} u\left(\tau, X_\tau^u\right)^\top u\left(\tau, X_\tau^u\right) d\tau + \int_t^{t_1} u\left(\tau, X_\tau^u\right)^\top dW_\tau, \tag{3.2}$$

where $t \in [t_0, t_1]$ and $X_{t_0}^u = x_0$. Here W_t is an m-dimensional standard Brownian motion, and

$$\begin{aligned} b &: [t_0, t_1] \times \mathbb{R}^n \to \mathbb{R}^n, \\ \sigma &: [t_0, t_1] \times \mathbb{R}^n \to \mathbb{R}^{n \times m}, \\ u &: [t_0, t_1] \times \mathbb{R}^n \to \mathbb{R}^m. \end{aligned}$$

Note that S_t^u depends on future ($\tau > t$) values of X and is therefore not adaptive [Øks85, FS06] with respect to the Brownian motion. Note furthermore that we have included a stochastic integral with respect to Brownian motion in the cost S. This is somewhat unusual because the stochastic integral vanishes when taking the expected value. However, when performing a change of measure with a drift u, such a term appears naturally (see Sections 3.7 and 3.5), and hence it is convenient to include it now.

For brevity we shall often suppress dependence on state in the notation of b, σ, u, and R. For example, Eq. (3.1) and Eq. (3.2) can also be described by

$$\begin{aligned} dX_t^u &= b_t dt + \sigma_t (u_t dt + dW_t), \\ S_t^u &= Q\left(X_{t_1}^u\right) + \int_t^{t_1} R_\tau + \frac{1}{2} u_\tau^\top u_\tau d\tau + \int_t^{t_1} u_\tau^\top dW_\tau. \end{aligned}$$

The goal in stochastic optimal control, as we also described in Section 2.2, is to minimize the expected cost with respect to the control.

$$J^\star(t, x) = \min_u J^u(t, x) = \min_u \mathbb{E}\left[S_t^u \mid X_t^u = x\right], \tag{3.3}$$

$$u^\star(\cdot, \cdot) = \arg\min_u J^u(t_0, x_0). \tag{3.4}$$

Here $\mathbb{E}$ denotes the expected value with respect to the stochastic process from Eq. (3.1). The following, previously established result [Kap05a, TBS10] gives a solution of the control problem in terms of path integrals.

Theorem 3.1. *The solution of the control problem Eqs. (3.3, 3.4) is given by:*

$$J^\star(t, x) = -\log \mathbb{E}\left[e^{-S_t^u} \mid X_t^u = x\right], \tag{3.5}$$

$$u^\star(t, x) - u(t, x) = \lim_{\tau \downarrow t} \frac{\mathbb{E}\left[(W_\tau - W_t) e^{-S_t^u} \mid X_t^u = x\right]}{\mathbb{E}\left[(\tau - t) e^{-S_t^u} \mid X_t^u = x\right]}. \tag{3.6}$$

Here $u(t, x)$ is an arbitrary Markov control.

Proof. Eq. (3.5) will be proven in Corollary 3.6 and Eq. (3.6) will be proven in Corollary 3.8. □

Because the solution of the control problem is given in terms of a path integral Eqs. (3.5, 3.6), the control problem Eqs. (3.1, 3.2) is referred to as a path integral control problem. Since these paths are conditioned on $X_t^u = x$, the solutions $u^\star$ and $J^\star$ must be recomputed for each t, x separately. This issue will be partly resolved in the Main Path Integral Control Theorem 3.7, where we show that all expected optimal future controls can be expressed using a single path integral.

The optimal control solution holds for any function u. In particular, it holds for $u = 0$ in which case we refer to Eq. (3.1) as the uncontrolled dynamics. Computing the optimal control in Eq. (3.6) with $u \neq 0$ implements a type of importance sampling, which is further discussed in Section 3.9 and in Chapter 5.

3.3 Alternative formulation

In this section we will give an alternative formulation of the path integral problem that at first glance appears to be more general than the path integral problem as given in Section 3.2. The alternative form might give the feeling of more power to express a control problem as one of the path integral type, and perhaps therefore it was used in works as [TBS10]. We will show, however, that the alternative form is equivalent to the definition given in Section 3.2.

Definition 3.2. The *alternative form* of the path integral problem has, instead of Eqs. (3.1, 3.2), the following respective dynamic and cost

$$\begin{aligned} dX_t &= b_t dt + g_t u_t dt + \sigma_t dW_t, \\ S &= Q\big(X_{t_1}\big) + \int_{t_0}^{t_1} \left(R_t + \frac{1}{2} u_t^\top V_t u_t \right) dt, \end{aligned}$$

with $g_t = g(t, X_t) \in \mathbb{R}^{n \times m}$ and a non-singular $V_t = V(t, X_t) \in \mathbb{R}^{m \times m}$ that satisfy the following restriction

$$\sigma_t \sigma_t^\top = c\, g_t V_t^{-1} g_t^\top, \tag{3.7}$$

for some scalar $c \in \mathbb{R}$. We furthermore assume that $m \leq n$ [1], and that g and σ are of rank m, i.e., of full column rank. The terms b, u, R, Q, σ and W are similarly defined as in Section 3.2.

[1] If $m > n$ we can consider a system where g, of rank n is replaced with an $n \times n$ matrix $\hat{g}$ satisfying $\hat{g}^\top \hat{g} = g g^\top$. This system with $m = n$ has the same uncontrolled dynamics and cost, and therefore the same optimal cost and associated optimal distribution.

Note that, in contrast to Eq. (3.2), we did not include a stochastic integral part (the $\int u^\top dW$ term) in the in the alternative form of the cost. The $\int u^\top dW$ term is a Martingale that vanishes when taking the expectation, so therefore it does not affect the solution of the control problem. The reason to in-/exclude such a term is therefore based on convenience and elegance. In the alternative form we aim to state the problem as convenient as possible for an application, and therefore we exclude the $\int u^\top dW$ term. In contrast, the form of Eq. (3.2) is such that the development of the theory will be as elegant as possible, and the $\int u^\top dW$ term plays a crucial role, as explained in Section 3.5.

In the proceeding we shall need that σ and g have a left inverse. Note that because σ and g are assumed to have independent columns, they indeed do have a left inverse – the so called Moore-Penrose pseudo inverse – defined by $g^+ = \left(g^\top g\right)^{-1} g^\top$, and similarly for σ. This left inverse has the properties that $g^+ g = 1_m$ and $\left(g^\top\right)^+ = (g^+)^\top$, and similarly for σ.

The next proposition shows that the problem in the alternative form from above is equivalent to a control problem as defined in Section 3.2.

Proposition 3.3. *Consider the control problem with dynamics and cost*

$$d\tilde{X}_t = b_t dt + \sigma_t [\tilde{u}_t dt + dW_t],$$
$$\tilde{S} = \tilde{Q}\left(\tilde{X}_{t_1}\right) + \int_{t_0}^{t_1} \left(\tilde{R}_t + \frac{1}{2}\tilde{u}_t^\top \tilde{u}_t\right) dt + \int_{t_0}^{t_1} \tilde{u}_t^\top \, dW_t,$$

that is of the regular form as defined in Eqs. (3.1, 3.2) in Section 3.2, where we take $\tilde{u} = \sigma^\top (g^+)^\top V u/c$, $\tilde{Q} = Q/c$ *and* $\tilde{R} = R/c$. *Then* $\tilde{X} = X$ *and* $\mathbb{E}[\tilde{S}] = \mathbb{E}[S]/c$. *In particular, if* $J^\star$, $u^\star$ *are the solution of the problem in alternative form, then* $\tilde{J}^\star = J^\star/c$, $\tilde{u}^\star = \sigma^\top (g^+)^\top V u^\star/c$ *are the solution of the problem in regular form.*

Proof. By definition of $\tilde{u}$ we have

$$\sigma\tilde{u} = \sigma\sigma^\top (g^+)^\top V u/c,$$

which according to Eq. (3.7) is equal to

$$\sigma\tilde{u} = gu.$$

It follows that the respective stochastic differential equations that define X and $\tilde{X}$ are the same.

In order to prove that $\mathbb{E}[\tilde{S}] = \mathbb{E}[S]/c$, it suffices to show that $\tilde{u}^\top \tilde{u} = u^\top V u/c$, because we have already chosen $\tilde{Q} = Q/c$ and $\tilde{R} = R/c$. Again, by definition of $\tilde{u}$ we get that

$$\begin{aligned}\tilde{u}^\top \tilde{u} &= \left(\sigma^\top (g^+)^\top V u/c\right)^\top \sigma^\top (g^+)^\top V u/c \\ &= u^\top V^\top g^+ \sigma\sigma^\top (g^+)^\top V u/c^2.\end{aligned}$$

Now use that $V = V^\top$ and, from Eq. (3.7), that $V^{-1} = g + \sigma\sigma^\top (g^+)^\top / c$, so that

$$\tilde{u}^\top \tilde{u} = u^\top V u / c. \qquad \square$$

3.4 Linearization of the HJB

In this section we start to solve the path integral control problem. The starting point is the HJB equation (2.6), which we will analyze for the specific form of the path integral problem. The next crucial step in the theory is a linearization trick via a logarithmic transform. These ideas were first explored by [Fle82], and used in the context of path integral control in [Kap05b]. With the linearization of the HJB we prove a useful lemma – the Main Lemma 3.4 – involving the controlled process. This lemma we will be used to derive various results, including the Main Path Integral Control Theorem, in the remainder of this Chapter.

The solution of the optimal control is described by the HJB Eq. (2.6). In case of path integral control, i.e. when the cost and dynamics are of the form Eqs. (3.1, 3.2) from Section 3.2, this is (suppressing dependence on time for brevity)

$$\begin{aligned} 0 &= \min_u \left[R + \frac{1}{2} u^\top u + \mathscr{A}^u(J^\star) \right] \\ &= \min_u \left[R + \frac{1}{2} u^\top u + \partial_t J^\star + (b + \sigma u)^\top \partial_x J^\star + \frac{1}{2} \mathrm{Tr}\left(\sigma\sigma^\top \partial_{xx} J^\star\right) \right], \end{aligned}$$

with boundary condition $J^\star(t_1, x) = Q(x)$. In case of path integral control the minimization can be solved for u, resulting in a partial differential equation for the optimal expected cost to go $J^\star$

$$u^\star = -\sigma^\top \partial_x J^\star \tag{3.8}$$

$$0 = R + \partial_t J^\star - \frac{1}{2} (\partial_x J^\star)^\top \sigma\sigma^\top \partial_x J^\star + b^\top \partial_x J^\star + \frac{1}{2} \mathrm{Tr}\left(\sigma\sigma^\top \partial_{xx} J^\star\right). \tag{3.9}$$

Throughout the rest of this work we reserve the symbol $\psi(t,x)$ for the function

$$\psi(t,x) = e^{-J^\star(t,x)},$$

and the symbol ψ_t, for the process

$$\psi_t = \psi(t, X_t^u).$$

When it is clear from the context that we are discussing either the function or the process, we will simply use the symbol ψ. Note that in terms of ψ, the optimal control formula Eq. (3.8) can also be expressed by

$$u^\star \psi = -\sigma^\top \partial_x (J^\star) \psi = \sigma^\top \partial_x \psi. \tag{3.10}$$

Applying the generator of the uncontrolled process to the function $\psi(t,x)$ results in the following equation

$$\mathscr{A}^0\psi = \left(-\partial_t J^\star + \frac{1}{2}(\partial_x J^\star)^\top \sigma\sigma^\top \partial_x J^\star - b^\top \partial_x J^\star - \frac{1}{2}\mathrm{Tr}\left(\sigma\sigma^\top \partial_{xx} J^\star\right)\right)\psi.$$

Combining with Eq. (3.9), we obtain a linear equivalent of the HJB in terms of the function ψ

$$\mathscr{A}^0\psi = R\psi, \tag{3.11}$$

with boundary condition $\psi(t_1,x) = e^{-Q(x)}$. This linear relation is very useful, and it allows us to prove the following Lemma.

Lemma 3.4 (Main Lemma)**.** *Let $u(s,y)$ be any Markov control function, and let $t \in [t_0,t_1]$. Let $J_t^{\star,u} = J^\star(t,X_t^u)$ denote the process that gives the optimal expected cost to go, initialized at time t in a random and sub-optimally controlled future state X_t^u. Then*

$$e^{-S_t^u} = e^{-J_t^{\star,u}} + e^{-S_t^u}\int_t^{t_1} e^{S_\tau^u - J_\tau^\star}\left[u_\tau^\star - u_\tau\right]^\top dW_\tau. \tag{3.12}$$

Proof. For $\tau \in [t,t_1]$ let

$$Z_\tau = S_t^u - S_\tau^u = \int_t^\tau R_s + \frac{1}{2}u_s^\top u_s dt + \int_t^\tau u_s^\top dW_s$$

denote the cost that is made from time t to time τ. Then Z_τ is – in contrast to S_τ^u – adapted, and hence satisfies the following SDE

$$dZ_\tau = \left(R_\tau + \tfrac{1}{2}u_\tau^\top u_\tau\right)d\tau + u_\tau^\top dW_\tau.$$

Let $\phi_\tau = e^{-Z_\tau}$, then, by Itô's Lemma [Øks85, FS06]

$$\begin{aligned} d\phi_\tau &= -\phi_\tau dZ_\tau + \tfrac{1}{2}\phi_\tau d[Z,Z]_\tau \\ &= -\phi_\tau\left(R_\tau d\tau + \tfrac{1}{2}u_\tau^\top u_\tau d\tau + u_\tau^\top dW_\tau\right) + \tfrac{1}{2}\phi_\tau u_\tau^\top u_\tau d\tau \\ &= -\phi_\tau\left(R_\tau d\tau + u_\tau^\top dW_\tau\right) \end{aligned}$$

Similarly, we apply Itô's Lemma in order to obtain a SDE for the process $\psi_\tau = \psi(\tau, X_\tau^u)$

$$d\psi_\tau = \mathscr{A}^u(\psi_\tau)d\tau + \partial_x(\psi)_\tau^\top \sigma_\tau dW_\tau.$$

This can be simplified by using the linearized HJB Eq. (3.11). Note that $\mathscr{A}^u = \mathscr{A}^0 + (\sigma u)^\top \partial_x$, so that

$$d\psi_\tau = R_\tau\psi_\tau d\tau + \partial_x(\psi)_\tau^\top \sigma_\tau(u_\tau d\tau + dW_\tau).$$

Combining with the analytical expression Eq. (3.10) for the optimal control we obtain

$$d\psi_\tau = \psi_\tau \left(R_\tau d\tau + u_\tau^{\star\top} u_\tau d\tau + u_\tau^{\star\top} dW_\tau \right).$$

Using the product rule from stochastic calculus [Øks85] we obtain

$$\begin{aligned} d(\phi_\tau \psi_\tau) &= \psi_\tau d\phi_\tau + \phi_\tau d\psi_\tau + d[\phi, \psi]_\tau \\ &= \phi_\tau \psi_\tau \left(-R_\tau d\tau - u_\tau^\top dW_\tau \right) \\ &\quad + \phi_\tau \psi_\tau \left(R_\tau d\tau + u_\tau^{\star\top} u_\tau d\tau + u_\tau^{\star\top} dW_\tau \right) \\ &\quad - \phi_\tau \psi_\tau u_\tau^\top u_\tau^\star d\tau \\ &= \phi_\tau \psi_\tau \left(u_\tau^\star - u_\tau \right)^\top dW_\tau. \end{aligned} \tag{3.13}$$

Integrating the above from t to t_1 gives

$$\phi_{t_1} \psi_{t_1} - \phi_t \psi_t = \int_t^{t_1} \phi_\tau \psi_\tau [u_\tau^\star - u_\tau]^\top dW_\tau.$$

From the definitions of ϕ and ψ we get $\phi_\tau \psi_\tau = e^{S_\tau^u - S_t^u - J_\tau^{\star,u}}$, and because $J_{t_1}^{\star,u} = S_{t_1}^u = Q\left(X_{t_1}^u\right)$, we obtain

$$e^{-S_t^u} - e^{-J_t^{\star,u}} = \int_t^{t_1} e^{S_\tau^u - S_t^u - J_\tau^{\star,u}} [u_\tau^\star - u_\tau]^\top dW_\tau.$$

Rearranging terms gives the statement of the lemma. □

3.5 The optimal cost is not random

The Main Lemma has the following immediate consequence

Corollary 3.5. *Let us denote $S_t^\star = S_t^{u^\star}$. When initializing with $X_{t_0} = x_0$ at time t_0, then, at time t, the optimal cost to go is equal to the optimal expected cost to go:*

$$S_t^\star = J_t^{\star,u^\star} = J^\star\left(t, X_t^{u^\star}\right).$$

In particular, when we condition on $X_t^{u^\star} = x$, we get

$$S_t^\star = J^\star(t, x).$$

Proof. Take $u = u^\star$ in Eq. (3.12). □

This result can be interpreted as follows. Let us define an optimally controlled random path as an instance of Eq. (3.1) with $u = u^\star$, i.e. this path is $X^\star = \left(X_t^\star\right)_{t_0 \le t \le t_1}$, where we have used the shorthand notation $X_t^\star = X_t^{u^\star}$. Although $X^\star$ is random, it's attributed cost $S_{t_0}^\star = J^\star(t_0, x_0)$ has zero variance, because the initial condition $X_{t_0}^\star = x_0$ has zero variance.

Looking back at the definition of S in Eq. (3.2), we now see that, in order to obtain this result, it was critical to include the stochastic integral $\int u^\top dW$. This can be explained intuitively as follows. The control $u_t dt$ of the system is disturbed by the noise dW_t. So if u_t and dW_t go in opposite directions this might be considered as bad luck because the noise pushes the system away from the direction you want to control it into. Fortunately, this bad luck is compensated for in the cost, because $\int u^\top dW$ gives a negative contribution to the cost when u and dW go in opposite directions. Similarly, if u and dW go in the same direction, you can be considered lucky, but you are penalized for that by the positive $\int u^\top dW$ term in the cost. The Corollary shows that this compensation is fair in the following sense: when controlling optimally the cost is always the same, regardless of good/bad luck.

3.6 A path integral for the optimal expected cost

With the following Corollary of the Main Lemma, we see that the optimal expected cost to go can be computed with a path integral.

Corollary 3.6. *Let $t \in [t_0, t_1]$. Then*

$$\psi_t = e^{-J_t^{\star,u}} = \mathbb{E}\left[e^{-S_t^u} \mid \mathscr{F}_t\right], \tag{3.14}$$

and the optial expected cost to go is given by

$$J^\star(t,x) = -\log \mathbb{E}\left[e^{-S_t^u} \mid X_t^u = x\right]. \tag{3.15}$$

Here $u(t,x)$ is an arbitrary Markov control, $\mathbb{E}$ denotes the expected value with respect to the stochastic process from Eq. (3.1), and the filtration $\mathscr{F}_t$ denotes that we are taking the expected value conditioned on events up to time t.

Proof. Take $\mathbb{E}[\cdot \mid \mathscr{F}_t]$ on both sides of Eq. (3.12). When conditioning on $X_t^u = x$, note that $J_t^\star = J^\star\left(t, X_t^u\right) = J^\star(t,x)$, and recall that $J^\star = -\log\psi$. □

We remark that when $u = 0$, Eq. (3.15) is equivalent to

$$\psi(t,x) = \mathbb{E}\left[\psi\left(t_1, X_{t_1}^0\right) e^{-\int_t^{t_1} R_\tau d\tau} \mid X_t^0 = x\right],$$

which is known as the Feynman-Kac formula [Øks85, FS06] for the PDE of Eq. (3.11). Therefore the result from the Corollary can also be obtained by using the Feynman-Kac theorem, and subsequently performing a change of measure from X^0 to X^u. The

correction term for this change of measure is given by the Radon-Nikodym derivative $e^{-\int(\frac{1}{2}u^\top u dt + u^\top dW)}$, which explains why we included $\int u^\top dW$ in the definition of S.

3.7 A path integral for the optimal control

In this chapter we present a new theorem from [TK15]: The Main Path Integral Control Theorem. This theorem is a generalization of Theorem 3.1 and gives a solution of the control problem in terms of path integrals. The disadvantage of Theorem 3.1 is that it requires us to recompute the optimal control for each t, x separately. Here, we show that we can also compute the *expected* optimal future controls using a single set of trajectories with initialization $X^u_{t_0} = x_0$. We furthermore generalize the path integral expressions by considering the product with some function $g(t,x)$. In the Chapter 6 we utilize this result to approximate a good feedback controller. Here we proceed with the statement and the proof of the generalized path integral formula.

Theorem 3.7 (Main Path Integral Control Theorem)**.** *Let $t \in [t_0, t_1]$, and let $u(\tau, x)$ be any Markov control function. Let $g : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}^l$, and consider the process $g_\tau = g(\tau, X^u_\tau)$, then*

$$\mathbb{E}\left[e^{-S^u_t}\int_t^{t_1} g_\tau dW_\tau^\top\right] = \mathbb{E}\left[e^{-S^u_t}\int_t^{t_1} g_\tau (u^\star_\tau - u_\tau)^\top d\tau\right]. \tag{3.16}$$

Here $\mathbb{E}$ denotes the expected value w.r.t. the process given in Eq. (3.1).

Proof. Consider the Main Lemma, multiply with $\int_t^{t_1} g_t dW_t^\top$, and take the expected value:

$$\mathbb{E}\left[e^{-S^u_t}\int_t^{t_1} g_\tau dW_\tau^\top\right] = \mathbb{E}\left[e^{-S^u_t}\int_t^{t_1} e^{S^u_\tau - J^\star_\tau} g_\tau (u^\star_\tau - u_\tau)^\top d\tau\right].$$

On the right-hand side the term $e^{-J^\star_t}\int_t^{t_1} g dW^\top$ has vanished because $e^{-J^\star_t}$ is adapted, and hence, independent from the stochastic integral that has zero mean. Note that the term $e^{-S^u_t}\int_t^{t_1} g dW^\top$ on the left-hand side has not vanished, since S^u_t is not adapted. We have furthermore used the Itô Isometry on the right-hand side.

The left-hand side is now as in the statement of the theorem. To see that the right-hand side is also as in the statement we apply Eq. (3.14), interchange $\mathbb{E}$ and

$\int$, and then use the Law of total expectation. This gives

$$\begin{aligned}\mathbb{E}\left[e^{-S_t^u}\int_t^{t_1} g_\tau dW_\tau^\top\right] &= \int_t^{t_1}\mathbb{E}\left[e^{S_\tau^u - J_\tau^\star - S_t^u} g_\tau (u_\tau^\star - u_\tau)^\top\right] d\tau \\ &= \int_t^{t_1}\mathbb{E}\left[e^{S_\tau^u - S_t^u}\mathbb{E}\left[e^{-S_\tau^u}\mid \mathscr{F}_\tau\right] g_\tau (u_\tau^\star - u_\tau)^\top\right] d\tau \\ &= \mathbb{E}\left[e^{-S_t^u}\int_t^{t_1} g_\tau (u_\tau^\star - u_\tau)^\top d\tau\right]. \qquad \square\end{aligned}$$

Corollary 3.8. *The above can be used to prove Eq. (3.6) of Theorem 3.1.*

Proof. Let t, τ be such that $t_0 \leq t \leq \tau \leq t_1$, and take Eq. (3.16) with $g(s,y) = \mathbb{1}_{s\in[t,\tau]}$. Dividing by $\tau - t$ and taking the limit $\tau \downarrow t$ we obtain

$$\lim_{\tau\downarrow t}\frac{1}{\tau - t_0}\mathbb{E}\left[e^{-S_t^u}(W_\tau - W_t)^\top\right] = \mathbb{E}\left[e^{-S_t^u}(u_t^\star - u_t)^\top\right].$$

If we condition this on $X_t^u = x$, then the expression $u_t^\star - u_t = u^\star(t,x) - u(t,x)$ can be pulled outside the expectation. Dividing by $\mathbb{E}\left[e^{-S_t^u}\mid X_t^u = x\right]$ we get Eq. (3.6). $\square$

3.8 A path integral for the optimal control gradient

If we were to call Eq. (3.14) and Eq. (3.16) respectively the zeroth- and first-order path integral, then we will derive in this section the second order path integral. The first order path integral is used in Chapter 6 in order to compute the parameter A of a parametrized control $u(t,x) = Ag(t,x)$. A is computed via a linear matrix equation, and the resulting control $u = Ag$ is in a certain way an approximation of $u^\star$. Using the second order path integral from this section in a similar way results in a quadratic equation for A. Unfortunately we have not been able to apply this quadratic equation to improve the control computations, and it remains unclear if there are other theoretical implications. Nevertheless we think that this unpublished result is interesting.

The path integral formulas that we will derive are very long, and would not fit the width of the paper if we were to use the same notation as in the previous sections. Therefore we shall simplify the notation somewhat by suppressing notation for dependence on time or dependence on importance control. Furthermore we will use the notation $\tilde{u} = u^\star - u$. E.g. the result from the Main Path Integral Control Theorem is denoted more briefly by

$$\mathbb{E}\left[e^{-S_t}\int_{\delta(t)} g\tilde{u}^\top d\tau\right] = \mathbb{E}\left[e^{-S_t}\int_{\delta(t)} g dW^\top\right], \tag{3.17}$$

where $\delta(t)$ is the time interval $\delta(t) = [t, t_1]$. We obtained this from the Main Lemma by multiplying with $\int dW$ and taking the expected value. If we instead multiply with $\iint dW dW^\top$, we get a new path integral formula.

Theorem 3.9. *Let $u(\tau, x)$ be any Markov control function, and write $\tilde{u} = u^\star - u$. Let $t \in [t_0, t_1]$, and let $\Delta(t)$ be the triangular set of times $\Delta(t) = \{(\tau, \rho) \in \mathbb{R}^2 \mid t \leq \rho \leq \tau \leq t_1\}$, then*

$$\begin{aligned}&\mathbb{E}\Big[e^{-S_t} \iint_{\Delta(t)} dW dW^\top\Big] \\ &\quad = \mathbb{E}\Big[e^{-S_t} \iint_{\Delta(t)} \big\{\tilde{u}\tilde{u}^\top + (\partial_x \tilde{u})\sigma + (\mathscr{A}^u(\tilde{u}) + (\partial_x \tilde{u})\sigma\tilde{u}) \int dW^\top\big\} d\tau^2\Big]. \qquad (3.18)\end{aligned}$$

Here the double integration is in the following order $\iint_{\Delta(t)} dW dW^\top = \int_t^{t_1} \big(\int_t^\tau dW_\rho\big) dW_\tau$, and $\iint_{\Delta(t)} \ldots d\tau^2 = \int_t^{t_1} \big(\int_t^\tau \ldots d\rho\big) d\tau$.

Proof. In this proof we will use the process ϕ_τ as defined in the proof of the Main Lemma. Recall that this is the process $\phi_\tau = e^{S_\tau - S_t}$. The result of this lemma can be expressed in the brief notation as

$$e^{-S_t} = \psi_t + \int_{\delta(t)} \phi\psi\tilde{u}^\top dW, \qquad (3.19)$$

where $\delta = [t, t_1]$. We multiply with $\iint_{\Delta(t)} dW dW^\top$ and take the expected value:

$$\mathbb{E}\Big[e^{-S_t} \iint_{\Delta(t)} dW dW^\top\Big] = \mathbb{E}\Big[\int_{\delta(t)} \phi\psi\tilde{u}^\top dW \iint_{\Delta(t)} dW dW^\top\Big].$$

On the right-hand side $\psi_t \iint_{\Delta(t)} dW dW^\top$ vanishes when taking the expectation, because ψ_t is independent from the Martingale $\iint_{\Delta(t)} dW dW^\top$. We proceed by applying the Itô Isometry on the right-hand side

$$\mathbb{E}\Big[e^{-S_t} \iint_{\Delta(t)} dW dW^\top\Big] = \mathbb{E}\Big[\int_{\delta(t)} \phi\psi\tilde{u} \int dW^\top d\tau\Big].$$

Similar as in the proof of the Main Path Integral Control Theorem, the next step is to swap $\mathbb{E}$ and $\int$ on the right-hand side and use that $\mathbb{E}[\phi_\tau\psi_\tau \mid \mathscr{F}_\tau] = \mathbb{E}[e^{S_\tau - S_t} e^{-S_\tau} \mid \mathscr{F}_\tau] = e^{-S_t}$ for all $\tau \in \delta(t)$, in combination with the law of total expectation

$$\begin{aligned}\mathbb{E}\Big[e^{-S_t} \iint_{\Delta(t)} dW dW^\top\Big] &= \int_{\delta(t)} \mathbb{E}\Big[\phi\psi\tilde{u} \int dW^\top\Big] d\tau \\ &= \int_{\delta(t)} \mathbb{E}\Big[\mathbb{E}[\phi\psi \mid \mathscr{F}_\tau]\tilde{u} \int dW^\top\Big] d\tau \\ &= \int_{\delta(t)} \mathbb{E}\Big[e^{-S_t}\tilde{u} \int dW^\top\Big] d\tau \\ &= \mathbb{E}\Big[e^{-S_t} \int_{\delta(t)} \tilde{u} \int dW^\top d\tau\Big].^2 \qquad (3.20)\end{aligned}$$

Because the integrand at the right-hand side evaluates to 0 at the time $\tau = t$, we can rewrite it as $\tilde{u}_\tau \int_t^\tau dW_\rho^\top = \int_t^\tau d\left(\tilde{u}\int dW^\top\right)_\rho$. Next, we try to find the SDE of this term. Itô's product rule gives

$$d\left(\tilde{u}\int dW^\top\right) = \tilde{u}dW^\top + d\tilde{u}\int dW^\top + d\left[\tilde{u}, \int dW^\top\right]. \tag{3.21}$$

where $[\cdot,\cdot]$ denotes the quadratic variation process. Next, we require the SDE of $\tilde{u}$, which by Itô's Lemma is given by

$$d\tilde{u}_\rho = \mathscr{A}^u(\tilde{u})d\rho + \partial_x(\tilde{u})\sigma dW.$$

Using this with Eq. (3.21) we obtain

$$d\left(\tilde{u}\int dW^\top\right)_\rho = \tilde{u}dW^\top + (\mathscr{A}^u(\tilde{u})d\rho + \partial_x(\tilde{u})\sigma dW)\int dW^\top + \partial_x(\tilde{u})\sigma d\rho.$$

We use this to rewrite the integrand on the right-hand side of Eq. (3.20), which results in

$$\begin{aligned}&\mathbb{E}\left[e^{-S_t}\iint_{\Delta(t)} dW dW^\top\right]\\ &= \mathbb{E}\left[e^{-S_t}\iint_{\Delta(t)} \tilde{u}dW_\rho^\top + \left(\mathscr{A}^u(\tilde{u})d\rho + \partial_x(\tilde{u})\sigma dW_\rho\right)\int dW^\top + \partial_x(\tilde{u})\sigma d\rho\; d\tau\right].\end{aligned}$$

Now we apply the Eq. (3.17) to replace the dW_ρ integrators from the inner integral with $\tilde{u}d\rho$, so that

$$\begin{aligned}&\mathbb{E}\left[e^{-S_t}\iint_{\Delta(t)} dW dW^\top\right]\\ &= \mathbb{E}\left[e^{-S_t}\iint_{\Delta(t)} \tilde{u}\tilde{u}^\top d\rho + (\mathscr{A}^u(\tilde{u})d\rho + \partial_x(\tilde{u})\sigma\tilde{u}d\rho)\int dW^\top + \partial_x(\tilde{u})\sigma d\rho\; d\tau\right]. \square\end{aligned}$$

Corollary 3.10. *Let t, r such that $t_0 \le t \le r \le t_1$, and define the triangular set $\Delta(t,r) = \{(\tau,\rho) \in \mathbb{R}^2 \mid t \le \rho \le \tau \le r)\}$, then*

$$\begin{aligned}&[u^\star(t,x) - u(t,x)][u^\star(t,x) - u(t,x)]^\top + \partial_x[u^\star(t,x) - u(t,x)]\sigma(t,x)\\ &= \lim_{r\downarrow t} \frac{2}{(r-t)^2}\frac{\mathbb{E}\left[e^{-S_t^u}\iint_{\Delta(t,r)} dW dW^\top \mid X_t^u = x\right]}{\mathbb{E}\left[e^{-S_t^u} \mid X_t^u = x\right]}.\end{aligned}$$

Proof. Consider Eq. (3.18), and replace t_1 with r. To get rid of the $\iint_{\Delta(t,r)} \ldots d\tau^2$ on the right-hand side, we multiplying with $2/(r-t)^2$ and take the limit $r \downarrow t$, so that

$$\lim_{r\downarrow t} \frac{2}{(r-t)^2}\mathbb{E}\left[e^{-S_t^u}\iint_{\Delta(t,r)} dW dW^\top\right] = \mathbb{E}\left[e^{-S_t^u}\left\{\tilde{u}_t\tilde{u}_t^\top + (\partial_x\tilde{u}_t)\sigma_t\right\}\right]. \tag{3.22}$$

[2] So far we have followed the exact same steps as in the Main Path Integral Control Theorem. So perhaps it is not unsurprising that this intermediate result can also be obtained directly, from Eq. (3.17) by choosing $g = \int dW$. Strictly speaking this is only allowed when we generalize the Main Path Integral Control Theorem to adaptive processes g_t instead of Markov control functions $g(t,x)$ with their attributed process $g_t = g(t, X_t)$.

The term $\left(\mathscr{A}_r^u(\tilde{u}_r) + (\partial_x \tilde{u}_r)\sigma_r \tilde{u}_r\right)\int_t^r dW^\top$ has vanished on the right hand side, because $\int_t^r dW$ evaluates to zero when $r = t$. If we condition this on $X_t^u = x$, we get

$$\begin{aligned}\tilde{u}_t\tilde{u}_t^\top + (\partial_x \tilde{u}_t)\sigma_t \\ = [u^\star(t,x) - u(t,x)][u^\star(t,x) - u(t,x)]^\top + \partial_x[u^\star(t,x) - u(t,x)]\sigma(t,x).\end{aligned}$$

Since this term is not random, it can be pulled outside the expectation in Eq. (3.22). Dividing by $\mathbb{E}\left[e^{-S_t^u} \mid X_t^u = x\right]$ we get the statement of the corollary. □

3.9 Path integral variance

A Monte Carlo approximation of the optimal control solution Eq. (3.6) is a weighted average, where the weight depends on the path cost. If the variance of the weights is high, then a lot of samples are required to obtain a good estimate. Critically, Eq. (3.6) holds for all u, so that it can be chosen to reduce the variance of the path weights. This induces a change of measure and an importance sampling scheme. By the Girsanov Theorem [Øks85, FS06], the change in measure does not affect the weighted average (for a more detailed description in the context of path integral control, see [TT12]). The Radon-Nikodym derivative $\exp\left(-\int \frac{1}{2}u^\top u dt + u^\top dW\right)$ is the correction term for importance sampling with u, which explains why we included $\int u^\top dW$ in the definition of S.

We have seen in Section 3.5 that the optimal u for sampling purposes is $u^\star$. In this section we will furthermore show that the variance will decrease as u gets closer to $u^\star$. This motivates adaptive sampling, which we treat in Chapter 5, in which increasingly better estimates u of $u^\star$ improve sampling so that even better approximations of $u^\star$ might be obtained.

Definition 3.11. Given a feedback control $u(t,x)$, and a realization of the cost $S_{t_0}^u$, we define:

1. The weight of a path is $\alpha^u = \dfrac{e^{-S_{t_0}^u}}{\mathbb{E}\left[e^{-S_{t_0}^u}\right]}$.
2. The fraction λ^u of effective samples is $\lambda^u = \dfrac{1}{\mathbb{E}[(\alpha^u)^2]}$.

Because $\mathrm{Var}(\alpha^u) + 1 = \mathbb{E}[(\alpha^u)^2]$, the fraction of effective samples as defined in Definition 3.11.2 satisfies $0 < \lambda^u \leq 1$. It has been suggested [Liu08] that this fraction can be used to determine how well one can compute a sample estimate of a weighted average. Loosely speaking, the idea behind the effective fraction of samples can be explained as follows. If the variance is high, then the path weight of most samples in the estimate is negligible, with a few exceptional 'lucky-samples'

that will have a relatively large path weight. In that case, only the lucky-samples will contribute effectively to the weighted sample estimate. On the other hand, if the variance is low, then all samples get roughly the same path weight, and they can all be considered effective.

Next we present a novel theorem from [TK15] that effectively gives a connection between sampling efficiency and the control. More precisely, the theorem gives an upper and lower bound on the variance of the path weight in terms of the control u. An important consequence of this theorem is given in Corollary 3.13: if a control u is close to the optimal $u^\star$, then also the fraction of effective samples is close to optimal.

Theorem 3.12. *We have the following upper and lower bounds for the variance of the path weight:*

$$\mathrm{Var}(\alpha^u) \leq \int_{t_0}^{t_1} \mathbb{E}\left[(u_t^\star - u_t)^\top (u_t^\star - u_t)(\alpha^u)^2\right] dt, \tag{3.23}$$

$$\mathrm{Var}(\alpha^u) \geq \int_{t_0}^{t_1} \mathbb{E}\left[(u_t^\star - u_t)\alpha^u\right]^\top \mathbb{E}\left[(u_t^\star - u_t)\alpha^u\right] dt. \tag{3.24}$$

Let $\|\cdot\|_\infty$ denote the L^∞-norm that is defined by

$$\|u\|_\infty = \inf\{A \geq 0 : |u(t,X)| < A \text{ for almost every } t, X\}.$$

Corollary 3.13. *If* $\|u^\star - u\|_\infty \leq \sqrt{\epsilon/(t_1 - t_0)}$*, then*

$$\lambda^u \geq 1 - \epsilon.$$

Proof. Combining $\|u^\star - u\|_\infty \leq \sqrt{\epsilon/(t_1 - t_0)}$ with Eq. (3.23) gives

$$\mathrm{Var}(\alpha^u) \leq \epsilon \mathbb{E}\left[(\alpha^u)^2\right].$$

Recall that $\mathrm{Var}(\alpha^u) = \mathbb{E}\left[(\alpha^u)^2\right] - 1$, so that

$$\mathbb{E}\left[(\alpha^u)^2\right] - 1 \leq \epsilon \mathbb{E}\left[(\alpha^u)^2\right],$$

$$1 - \frac{1}{\mathbb{E}[(\alpha^u)^2]} \leq \epsilon,$$

and thus, by definition of λ^u,

$$\lambda^u \geq 1 - \epsilon. \qquad \square$$

Proof of Theorem 3.12. From Corollary 3.6 it follows that the denominator in the definition of α^u is $\mathbb{E}\left[e^{-S_{t_0}^u}\right] = e^{-J_{t_0}^\star} = \psi_{t_0}$. So if we consider the Main Lemma with $t = t_0$, and divide by ψ_{t_0} we get

$$\alpha^u = 1 + \int_{t_0}^{t_1} \alpha^u e^{S_t^u - J_t^\star} [u_t^\star - u_t]^\top dW_t.$$

Using the Itô Isometry [Øks85], we see that the variance is given by

$$\mathrm{Var}(\alpha^u) = \mathbb{E}\int_{t_0}^{t_1} \left(\alpha^u e^{S_t^u - J_t^\star}\right)^2 [u_t^\star - u_t]^\top [u_t^\star - u_t] dt. \tag{3.25}$$

For the upper bound we consider Eq. (3.14), squared, and then we apply Jensen's inequality

$$e^{-J_t^\star 2} = \mathbb{E}\left[e^{-S_t^u} \mid \mathscr{F}_t\right]^2 \leq \mathbb{E}\left[e^{-2S_t^u} \mid \mathscr{F}_t\right].$$

Substituting in Eq. (3.25) and using the Law of total expectation we obtain Ineq. (3.23).

For the lower bound we use Jensen's Inequality on the whole integrand of Eq. (3.25) to obtain

$$\mathrm{Var}(\alpha^u) \geq \int_{t_0}^{t_1} \mathbb{E}\left[\alpha^u e^{S_t^u - J_t^\star} \left(u_t^\star - u_t\right)^\top\right] \mathbb{E}\left[\alpha^u e^{S_t^u - J_t^\star} \left(u_t^\star - u_t\right)\right] dt.$$

Using Eq. (3.14) and the Law of total expectation we obtain Ineq. (3.24). □

We conclude that the optimal control problem is equivalent to the optimal sampling problem. An important consequence, which is given in Corollary 3.13, is that if the importance control is close to optimal, then so is the sampling efficiency.

Next we give a illustration of Theorem 3.12. For this, we consider the following control problem, of which we know the analytical solution.

Example 3.14 (Geometric Brownian Motion)**.** The path integral control problem with dynamics and cost given respectively by

$$\begin{aligned} dX_t^u &= X_t^u\left((\tfrac{1}{2} + u_t)dt + dW_t\right), \\ S^u &= 5\log(X_1^u)^2 + \frac{1}{2}\int_0^1 u_t^2 dt + \int_0^1 u_t^\top dW_t, \end{aligned}$$

with $0 \leq t \leq 1$ and initial state $X_0 = 1/2$, has solution

$$\begin{aligned} J^\star(t,x) &= \frac{5\log(x)^2}{10(1-t)+1} + \frac{1}{2}\log(10(1-t)+1), \\ u^\star(t,x) &= \frac{-10\log(x)}{10(1-t)+1}. \end{aligned}$$

To verify that this is indeed the solution, it can readily be checked that $J^\star$ satisfies Eq. (3.9) and that $u^\star$ satisfies Eq. (3.8), with $R = 0$, $b = x/2$, and $\sigma = x$.

In order to visualize Theorem 3.12 we consider a range of sub-optimal importance controls $u^\epsilon(t,x) = u^\star(t,x) + \sqrt{\epsilon}$. Each u^ϵ yields a path weight $\alpha^\epsilon := \alpha^{u^\epsilon}$. Because $(u^\star - u^\epsilon)^2 = \epsilon$, Theorem 3.12 implies that $\epsilon \leq \mathrm{Var}(\alpha^\epsilon) \leq \frac{\epsilon}{1-\epsilon}$. The results are reported in Figure 3.1.

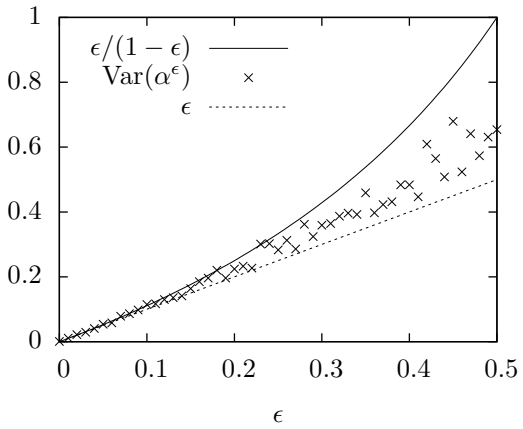

Figure 3.1: Estimate of $\mathrm{Var}(\alpha^\epsilon)$, where $\alpha^\epsilon := e^{-S^{u^\epsilon}(t_0)}/\psi(t_0, x_0)$ with upper and lower bounds from Theorem 3.12 with respect to the control problem in Example 3.14. Here we considered a range of sub-optimal importance controls $u^\epsilon(t,x) = u^\star(t,x) + \sqrt{\epsilon}$. The estimate of the variance is based on a MC estimate of 10^4 paths that were generated approximately with the Euler-Maruyama method with time increment $dt = 0.001$.

Chapter 4

Kullback Leibler control theory

4.1 Introduction

In the path integral control problem that we defined in Chapter 3, and more generally, in the stochastic optimal control problem from Chapter 2, the randomness comes from a stochastic process with a finite time horizon that is driven by a Brownian motion. The theory of stochastic control, however, is not restricted to these sources of randomness, as we shall see in this chapter. The main result is that the path integral control problem can be generalized to a minimization over probability measures that are regularized by a Kullback-Leibler (KL) divergence. Consequently, the more general stochastic control problem is known as KL control problem.

This chapter is not included for the sake a generalization alone. It is included because a treatment of path integral- without KL-control is not complete. Three critical contributions of the KL control generalization are that: the more the general setup (1) allows for more mathematical rigor, (2) gives a broader perspective on the matter, which makes it easier to make connections to related problems such as efficient Monte Carlo sampling, as treated in Chapter 5, and (3) sheds a new light on the more specific theory of path integral control, resulting in more elegant proofs of established results.

Some of the key elements in KL control theory have been developed in the closely related field large deviation, see for example [MD98]. For more recent developments see in KL control, we refer to [Leh13, BK14]. A new result that is included in this chapter is a new proof of the Main Path Integral Control Theorem, at the end of Section 4.4.

Outline. The rest of this chapter is structured as follows. In Section 4.2 we formulate the KL-control problem, that we subsequently analyze and solve in

Section 4.3. These results are then interpreted in the specific case of path integral control in Section 4.4.

4.2 Definition

Throughout the rest of this chapter we use the following objects. Let $(\Omega, \mathscr{F}, Q)$ be probability space with a random variable C, which is called the *(state) cost* . We call Q the *uncontrolled probability measure*. Other probability measures on $(\Omega, \mathscr{F})$ are referred to as *controlled probability measures*, or more briefly by *controlled measure*. Controlled probability measures will often be denoted by the letter P.

The goal will be to minimize $\mathbb{E}_P[C]$ w.r.t. the controlled measure P, while at the same time minimizing the distance from Q to Q. To be more precise, if for all $A \in \Omega$ we have $Q(A) = 0 \Rightarrow P(A) = 0$, then the measure P is said to be absolutely continuous with respect to Q, which is denoted by $P \ll Q$. If this is the case then, by the Radon-Nikodym Theorem [Nie97], P has a density with respect to Q and this density is called the Radon-Nikodym derivative, which is denoted by $\frac{dP}{dQ}$. The measure for distance that we will use is

$$\mathrm{D}_{\mathrm{KL}}(P\|Q) = \int \log\left(\frac{dP}{dQ}\right) dP,$$

which is known as the Kullback-Leibler divergence from Q to P, or as the relative entropy of P with respect to Q. Now we can define respectively the *total (random) cost* and the *total expected cost*. These combine the state cost with the KL-divergence.

$$\begin{aligned} S(P) &= C + \log \tfrac{dP}{dQ}, \\ J(P) &= \mathbb{E}_P\left[S(P)\right] = \mathbb{E}_P[C] + \mathrm{D}_{\mathrm{KL}}(P\|Q). \end{aligned}$$

The goal in the KL-control is to compute $J^\star := \inf_P J(P)$, and if it exists, to construct a minimizing measure $P^\star$ such that $J(P^\star) = J^\star$.

4.3 KL solution

In this section we will describe the solution of the KL control problem. This solution is based on the work of [BK14] and is found without using the Dynamic Programming Principle. Furthermore, since the randomness in KL control does not necessarily involve stochastic processes, there is no equivalent of the HJB equation. Instead, the solution directly gives a result that, in a sense, generalizes the path integral formula Eq. (3.15) for the optimal cost.

Theorem 4.1. *Suppose that* $\mathbb{E}_Q\left[e^{-C}\right] < \infty$. *Then a measure* $P^\star$ *exists, such that* $P^\star \ll Q$ *and such that* $S(P^\star) = J(P^\star) = -\log \mathbb{E}_Q\left[e^{-C}\right]$. *Furthermore, the measure* $P^\star$

is optimal in the following sense: if P is a measure with $P \ll P^\star$, then $J(P^\star) \leq J(P)$, with equality if and only if $P = P^\star$ almost everywhere.

Proof. Because $\mathbb{E}_Q[e^{-C}] < \infty$, we can define a measure $P^\star$ by

$$\frac{dP^\star}{dQ} = \frac{e^{-C}}{\mathbb{E}_Q[e^{-C}]}. \tag{4.1}$$

For this measure we have $P^\star \ll Q$, and so it's total cost is defined, and it satisfies

$$\begin{aligned} S(P^\star) &= C + \log\left(\tfrac{dP^\star}{dQ}\right) \\ &= C + \log\left(\tfrac{e^{-C}}{\mathbb{E}_Q[e^{-C}]}\right) \\ &= -\log \mathbb{E}_Q\left[e^{-C}\right]. \end{aligned}$$

Clearly $S(P^\star) = J(P^\star)$; the total random cost equals the total expected cost because it has zero variance. Now suppose that P is another measure with $P \ll P^\star$. Then also $P \ll Q$, so that we can consider the total cost w.r.t. P, and rewrite it as

$$\begin{aligned} S(P) &= C + \log \tfrac{dP}{dQ} \\ &= C + \log\left(\tfrac{dP^\star}{dQ}\tfrac{dP}{dP^\star}\right) \\ &= C + \log\left(\tfrac{e^{-C}}{\mathbb{E}_Q[e^{-C}]}\tfrac{dP}{dP^\star}\right) \\ &= -\log \mathbb{E}_Q\left[e^{-C}\right] + \log\left(\tfrac{dP}{dP^\star}\right). \end{aligned}$$

So the total expected cost is equal to

$$J(P) = \mathbb{E}_P[S(P)] = -\log \mathbb{E}_Q\left[e^{-C}\right] + \mathrm{D}_{\mathrm{KL}}(P\|P^\star). \tag{4.2}$$

Optimality of $P^\star$ now follows from Gibbs' inequality. □

The condition $\mathbb{E}_Q\left[e^{-C}\right] < \infty$ is sufficient to guarantee that the measure $P^\star$ exists, including a total expected cost $J^\star = J(P^\star) = -\log \mathbb{E}_Q\left[e^{-C}\right]$. However, there are cases where the solution $P^\star$ has the peculiar property that $\mathbb{E}_{P^\star}[C] = -\infty$ while at the same time $\mathbb{E}_{P^\star}\left[\log \frac{dP^\star}{dQ}\right] = \infty$. To exclude these cases, it is sufficient [BK14] to assume that the following conditions hold: $\mathbb{E}_Q[\mathbb{1}_{C<\infty}] > 0$ and $\mathbb{E}_Q\left[|C|e^{-C}\right] < \infty$. These conditions also imply $\mathbb{E}_Q\left[e^{-C}\right] < \infty$.

In the next section we will analyze the special case when the KL control problem is a path integral control problem. With this in mind, we can already see some connections between the results of Theorem 4.1 and the theory developed in Chapter 3. In both cases the optimal cost $S^\star$ has zero variance, and is equal to the optimal expected cost $J^\star$. Furthermore the optimal cost is given by $J^\star = \mathbb{E}_Q[e^{-C}]$, which we also found in Eq. (3.15) if we take $u = 0$.

4.4 KL and path integral control

KL control is a generalization of path integral control. If the randomness from the KL control comes from a Brownian motion over a finite time interval, however, the KL control problem becomes, loosely speaking, a path integral control problem. In this Section we shall make the transition form KL control to path integral control. The key connection is made by the Girsanov Theorem: the Radon-Nikodym derivative attributed to a change of measure is exactly the exponentiated control cost of the drift that is added in the change of measure. We use this idea in this section in order to construct controlled measures P^u for given a suitable control process u_t. Subsequently we shall analyze what the consequences are for the optimal control and measure. In particular, we shall give a new proof of the Main Path Integral Control Theorem.

Let W_t be a standard m-dimensional Brownian motion on a filtered probability space $(\Omega, (\mathscr{F}_t)_{t\in[t_0,t_1]}, Q)$. Let and X_t an be n-dimensional Itô process of the form

$$dX_t = b_t dt + \sigma_t dW_t,$$

where $b_t \in \mathbb{R}^n$ and $\sigma_t \in \mathbb{R}^{n\times m}$. We call X a path, or more specifically, an uncontrolled path when considered with respect to the measure Q. In order to control X, we shall consider control processes:

Definition 4.2. We call an m-dimensional measurable adapted process $(u_t)_{t_0\le t\le t_1}$ an *(admissible) control process* when $Z_t = e^{\int_{t_0}^t u_s dW_s - \frac{1}{2}\int_{t_0}^t u_s^\top u_s ds}$ is a square integrable Martingale.

The process Z_t from the definition above is known as the Doléans-Dade exponential of $\int u dW$, and is also denoted by $Z_t = \mathscr{E}\left(\int_{t_0}^t u_s dW_s\right)$. More generally, $\mathscr{E}(v_t)$ is defined as the solution $Y_t = \mathscr{E}(v_t)$ of the SDE given by $dY_t = Y_t dv_t$, with initial condition $Y_{t_0} = 0$. A well known condition that ensures that the process $Z_t = \mathscr{E}\left(\int_{t_0}^t u_s dW_s\right)$ is a Martingale, is the so called Novikov Condition, which states that $\mathbb{E}\left[e^{\frac{1}{2}\int_{t_0}^{t_1} u_t^\top u_t dt}\right] < \infty$.

If u_t is a control process, then we get by standard Girsanov theory that $dP^u = Z_t dQ$ defines a probability measure P^u on $(\Omega, (\mathscr{F}_t)_{t\in[t_0,t_1]})$ that is equivalent to Q. We will refer to P^u as the controlled measure. The connection with the control process u_t is given by the Girsanov Theorem: $dB_t = dW_t - u_t dt$ defines a P^u-Brownian motion B_t, so that the stochastic representation of X in terms of P^u is

$$dX_t = b_t dt + \sigma_t(u_t dt + dB_t),$$

which is a controlled path. Similarly, we get that

$$e^{-S(P^u)} = e^{-C}\frac{dQ}{dP^u} = e^{-C-\int_{t_0}^{t_1} u_t dW_t + \frac{1}{2}\int_{t_0}^{t_1} u_t^\top u_t dt} = e^{-C-\int_{t_0}^{t_1} u_t dB_t - \frac{1}{2}\int_{t_0}^{t_1} u_t^\top u_t dt}.$$

The connection with path integral control from Chapter 3 becomes apparent when we choose the state cost of the form $C = Q(X_{t_1}) + \int_{t_0}^{t_1} R(t, X_t) dt$.

Next we consider the optimal measure $P^\star$ as defined in Section 4.3, which exists when $\mathbb{E}_Q\left[e^{-C}\right] < \infty$. The first question is whether a an optimal control process exists, which is a control process $u_t^\star$ such that $P^\star = P^{u^\star}$.

Theorem 4.3. *Suppose that $Z_t = \mathbb{E}_Q\left[\frac{e^{-C}}{\mathbb{E}_Q[e^{-C}]} \mid \mathscr{F}_t\right]$ is square integrable martingale w.r.t. Q. Then there is an adapted $u_t^\star$ such that $P^\star = P^{u^\star}$.*

Proof. By the Martingale Representation Theorem [Øks85], we get that there exists a process V_t such that $dZ_t = V_t dW_t$. Now take a process $u_t^\star$ such that $V_t = u_t^\star Z_t$. Then Z_t is the solution of $dZ_t = u_t^\star Z_t dW_t$, in other words, it is the Doléan-Dade exponential $Z_t := \mathscr{E}\left(\int_{t_1}^t u_s^\star dW_s\right)$. Because Z_t is a square integrable Martingale, the process $u_t^\star$ is an admissible control process. The theorem follows by the defining property Eq. (4.1) of $P^\star$ and the construction of P^u as described above. □

Using the construction above we present a new result form [TK16], which is a more elegant proof of the Main Path Integral Control Theorem than that of Section 3.7.

Theorem 4.4 (Main Path Control Integral Theorem)**.** *Let u_t be a control process, and let B_t be a P^u-Brownian motion. Suppose that an optimal control $u^\star$ exists. Let g_t be a k-dimensional measurable, square integrable and adapted process, then*

$$\mathbb{E}_{P^u}\left[e^{-S(P^u)} \int_{t_0}^{t_1} g_t u_t^{\star\top} dt\right] = \mathbb{E}_{P^u}\left[e^{-S(P^u)} \int_{t_0}^{t_1} g_t (dB_t + u_t dt)^\top\right].$$

Proof. Because $u^\star$ is an optimal control we have $P^\star = P^{u^\star}$. Furthermore, the equation

$$dB_t^\star + u_t^\star dt = dB_t + u_t dt.$$

defines a $P^\star$-Brownian motion $B_t^\star$. If we multiply this equation with the g, integrate, and take the expected value w.r.t. $P^\star$, we obtain

$$\mathbb{E}_{P^\star}\left[\int_{t_0}^{t_1} g_t u_t^{\star\top} dt\right] = \mathbb{E}_{P^\star}\left[\int_{t_0}^{t_1} g_t (u_t dt + dB_t)^\top\right].$$

On the left hand side the $\int g_t dB^\star$ term vanished because it is a Martingale w.r.t. $P^\star$ because g is square integrable. The variable $e^{-S^\star} = e^{-C}\frac{dQ}{dP^\star} = \mathbb{E}_Q\left[e^{-C}\right]$ has zero variance, so it is safe to multiply the equation above with it

$$\mathbb{E}_{P^\star}\left[e^{-S^\star} \int_{t_0}^{t_1} g_t u_t^{\star\top} dt\right] = \mathbb{E}_{P^\star}\left[e^{-S^\star} \int_{t_0}^{t_1} g_t (u_t dt + dB_t)^\top\right].$$

Changing the measure from $P^\star$ to P^u, and using that $e^{-S^\star}\frac{dP^\star}{dP^u} = e^{-S(P^u)}$, we obtain

$$\mathbb{E}_{P^u}\left[e^{-S(P^u)}\int_{t_0}^{t_1} g_t u_t^{\star\top} dt\right] = \mathbb{E}_{P^u}\left[e^{-S(P^u)}\int_{t_0}^{t_1} g_t(u_t dt + dB_t)^\top\right].^{1} \qquad \square$$

The theory in this chapter could be generalized somewhat if we drop the Martingale condition in the definition of control process and in Theorem 4.3. In that case one has, for example, to deal with the fact that $P^\star$ and Q might not be equivalent measures. In such a situation the standard Girsanov Theorem is not valid, and it should be replaced by a version that only requires that $P^\star \ll Q$. We refer the reader to [BK14] for a treatment of the general case.

[1] Note that although B_t is a P^u-Brownian motion, the martingale term $\int g_t dB_t$ does not vanish on the right hand side when taking the expected value w.r.t. P^u. The reason is that the martingale is multiplied by $e^{-S(P^u)}$, which is not adapted.

Chapter 5

Adaptive multiple importance sampling

5.1 Introduction

Monte Carlo (MC) integration is a broadly applied method to numerically compute integrals that might be difficult to evaluate otherwise, due to, for example, high dimensions. The main shortcoming of MC integration is perhaps that the estimator can have a high variance, which has led to techniques such as Importance Sampling (IS). The idea behind IS is reducing the variance of an estimator by drawing samples from a chosen proposal distribution that puts more emphasis on “important" regions. This will in general introduce a bias, which has to be corrected with an importance weight.

There are, generally speaking, two different motivations for implementing IS.

First, one might be interested in an expected value over a distribution Q from which it is impossible to draw samples (efficiently). In this case a proposal distribution can be constructed in order to generate samples [CGMR04, MPS12, CMMR12]. When the density $q = dQ/dx$ is only known up to a factor, the normalization constant needs to be estimated as well. For this reason, it is common to choose a proposal distribution close to Q.

The second motivation to use IS, is whenever sampling from Q is possible but very inefficient for the purpose of MC integration. This is, for example, typically the case with conditioned diffusions [Doo57] or stochastic control problems [KR16], which have important applications of IS in, for example, robotics, [TBS10]. Our motivation to use IS is of the second kind.

In cases where it is difficult to choose a single proposal distribution that covers all the important regions, one can resort to a mixture of proposal distributions, This technique is known as Multiple Importance Sampling (MIS) [OZ00]. An important

problem in MIS is the choice or construction of good proposal distributions. Roughly, there are two approaches: either the proposals are carefully chosen in advance of the sampling procedure [VG95], or the proposals are optimized during the sampling procedure [OB92, CMMR12]. The advantage of the former is that it is clearly consistent, because, in contrast to the latter, all samples are independent. The advantage of the latter is that the optimization scheme might yield better sampling efficiency.

A particular instance of MIS with optimization of the proposals during sampling is the so-called Adaptive Multiple Importance Sampling (AMIS) algorithm [CMMR12]. In AMIS the samples and their associated importance weights are combined according to the balance heuristic. Although the balance heuristic is optimal in the sense of variance reduction when the number of samples goes to infinity [VG95], it also introduces a complicated dependence between the samples from the various proposals. As a consequence, consistency for AMIS is a non trivial proposition, which only recently been established, and only in restricted cases [MPS12].

An aspect of AMIS, or more generally of MIS, that has not been addressed by the literature, is that of the additional computational overhead that is caused by re-weighting of the samples. This overhead is proportional to the cost of computing a likelihood ratio. In some scenarios, for example when sampling requires real world interaction, this cost might be negligible. However, in MC sampling this cost will be roughly proportional to the cost of drawing a sample. This becomes an issue when the re-weighting scheme has a higher computational complexity than the drawing process, because in that case the algorithm will eventually spend more time on re-weighting than on drawing samples. Critically, this is the case when re-weighting uses the balance heuristic, which has a complexity of $\mathcal{O}(K^2M)$, where K is the number proposal distributions, and M the number of samples per proposal. Note that this is larger than the complexity $\mathcal{O}(KM)$ of drawing all the MK samples, particularly with many proposal distributions.

In this chapter we propose a new re-weighting scheme, called discarding-re-weighting, and addresses the issues described above. In particular, discarding-re-weighting will have a complexity of $\mathcal{O}(KM)$. Furthermore we will provide a consistency proof of the corresponding discarding-AMIS, without any restrictions, aside from the usual, on the proposals distributions.

In this work we are mainly interested in sampling over diffusion processes. For diffusion processes a natural proposal distribution arises by adding a drift term, which can be interpreted as a control input that steers the diffusion process. Here starts, loosely speaking, the connection with path integral control. In case of a more general measure, there is a similar connection with KL control. We have already seen in previous chapters that the solution to the stochastic control problem can expressed as an expected value, which in turn could be computed via a MC sampling method. In this chapter we shall see that the best proposal distribution

for sampling is also the solution to a stochastic optimal control problem. In a sense, the sampling- and control-problem are mathematically indistinguishable.

Outline. The remainder of this chapter is structured as follows. In Section 5.2 we review the generic AMIS method. Sections 5.3–5.5 consider the re-weighting scheme, where in Section 5.3 we treat consistency, in Section 5.4 introduce discarding-re-weighting, which we apply in Section 5.5 to sampling over diffusion processes. In Section 5.6 we propose a specific proposal update in the context of diffusion processes. This update is used in Section 5.7 to compare our new re-weighting scheme with the balance heuristic.

The new results in this Chapter are from [TK16].

5.2 The generic AMIS

In this section we briefly review IS, MIS and AMIS for MC integration. In particular we shall give a description of a generic AMIS.

Let $(\Omega, \mathscr{F}, Q)$ be a probability space with an E-valued random variable X, and an $\mathbb{R}$-valued function $h(X)$. The goal is to calculate

$$\psi = \mathbb{E}_Q[h(X)],$$

using a MC estimate. In particular we will be interested in variance reduction that can be achieved via importance sampling. Let P be another probability measure on $(\Omega, \mathscr{F})$, and let dQ/dP denote a density of Q relative to P, then

$$\hat{\psi}^P = \frac{1}{N}\sum_{n=1}^{N} h(X_n)\tfrac{dQ}{dP}(X_n), \qquad \text{where } X_n \sim P, \tag{5.1}$$

is an unbiased estimator for ψ, provided that for all events $A \in \mathscr{F}$

$$P(A) = 0 \quad \Longrightarrow \quad h = 0 \text{ on } A, Q\text{-almost surely.} \tag{5.2}$$

Often condition (5.2) is replaced by the stronger assumption of absolute continuity, $Q \ll P$, so that the importance weight dQ/dP exists everywhere. Regarding importance sampling, however, we only require that dQ/dP exists whenever $h \neq 0$.

Instead of using one proposal, P, the MC estimate can also be based on a mixture of proposals. For $k = 1, \dots, K$ let P_k be probability measures on $(\Omega, \mathscr{F})$ all satisfying Condition (5.2). The Multiple IS (MIS) estimator is defined as

$$\hat{\psi}^{\text{MIS}} = \frac{1}{N}\sum_{k=1}^{K}\sum_{n=1}^{N_k} h(X_n^k)\tfrac{dQ}{dP_k}(X_n^k)w_k(X_n^k), \tag{5.3}$$
$$X_n^k \sim P_k, \qquad \text{for } n = 1, \dots, N_k,$$

where $N = \sum_{k=1}^{K} N_k$ is the total number of samples. If the X_n^k are independent, and the re-weighting functions $w_k(x)$ satisfy

$$h(x) \neq 0 \quad \Rightarrow \quad \frac{1}{N} \sum_{k=1}^{K} N_k w_k(x) = 1,$$

then $\hat{\psi}^{\text{MIS}}$ is an unbiased estimate [VG95]. Remarkably there are many choices for w_k. A particularly simple choice would be $w_k = 1$, which will henceforth be referred to as flat re-weighting. Another scheme that is of interest is the so called balance-heuristic, which is also called deterministic multiple mixture. It is defined by

$$w_k(x) = \frac{1}{\frac{1}{N} \sum_{l=1}^{K} N_k \frac{dP_l}{dP_k}(x)}.$$

The advantage of balance heuristic over flat re-weighting is that the former results in lower variance mixed estimates when combining, for example, a (good) proposal that gives low variance estimates with a (bad) proposal that gives high variance estimates. The reason is, roughly speaking, that the reciprocal of the variance of the balance mix is the reciprocal of the harmonic mean of the variance of the individual proposals, while for the flat re-weighted mix this is the standard arithmetic mean. For a study on the relative merits of various related re-weighting schemes for MIS see [VG95, OZ00].

In order to improve the efficiency of a MIS algorithm, one can adapt the proposals sequentially. This idea was first mentioned in [OB92] with the name Adaptive Importance Sampling (AIS), and more recently in [CMMR12] with the name Adaptive Multiple Importance Sampling (AMIS). Both of these methods adapt the proposals at iteration k by adapting a parameter that is estimated using all samples that are draw up to iteration k. The two methods differ in the re-weighting: AMIS uses the balance-heuristic, while AIS uses flat re-weighting. If we instead consider the idea of adaptive sequential updates without specifying the form of the proposal or the re-weighting scheme we obtain a generic AMIS [EMLB15a, MELC15]:

The computational complexity of the generic AMIS will depend on the specifics of both the adaptation and re-weighting step. For example, AMIS with K iterations that uses the balance-heuristic has a complexity of $\mathcal{O}(MK^2)$, when $N_k = M$ samples are used at each iteration k, while for flat re-weighting this is only $\mathcal{O}(MK)$.

The unbiasedness and consistency from MIS does in general not carry over to the generic AMIS. The adaptation step introduces dependencies between samples from different iterations. Furthermore, the re-weighting might introduce extra correlations. Consistency has been established for a specific AMIS in [MPS12] under the assumption that the adaptation is only based on the last N_k samples and that N_k grows at least as fast as k. The downside of this method is that the proposal cannot be updated very frequently, and only while using a subset of all the

Algorithm 1 generic AMIS

- At iteration $k = 1, \dots, K$ do

  Adaptation. Construct a measure P_k, possibly depending on X_n^l and w_n^l with $1 \le l < k$, $1 \le n \le N_l$

  Generation. For $n = 1, \dots, N_k$ draw $X_n^k \sim P_k$

  Re-weighting. For $n = 1, \dots, N_k$ construct w_n^k.
  For $l = 1, \dots, k-1$ and $n = 1, \dots, N_l$, update w_n^l.

  Output. Return $\hat{\psi}_k = \frac{1}{\sum_{l=1}^k N_l} \sum_{l=1}^k \sum_{n=1}^{N_l} \frac{dQ}{dP_l}(X_n^l) h(X_n^l) w_n^l$

available samples. In the next section we will establish consistency of AMIS with flat re-weighting (flat-AMIS) for generic proposal adaptations without any such restrictions.

5.3 Consistency of flat-AMIS

In this section we will prove that flat-AMIS is consistent. Consistency can only be established when we make some assumptions on the proposals (see Example 5.2), but these assumptions will be quite general and they often do not pose any restrictions in practice.

Let $\mathscr{P}$ be the class of proposal distributions. Let $\|X\|_r = (\mathbb{E}[|X|^r])^{1/r}$ denote the L^r-norm. We will require that there are constants $r > 1$ and $C > 0$, such that for all $P \in \mathscr{P}$

$$\left\| h(X) \tfrac{dQ}{dP} \right\|_r < C, \qquad X \sim P. \tag{5.4}$$

The following theorem is a new result from [TK16].

Theorem 5.1 (Flat-AMIS is consistent)**.** *Let $\hat{\psi}_k$ be defined as in the output step of Algorithm 1 using flat re-weighting, i.e. with $w_n^k = 1$. Suppose that both Eq. (5.2) and (5.4) are satisfied, then*

$$\hat{\psi}_k \to \psi \qquad a.s. \tag{5.5}$$

when $\sum_{l \le k} N_l \to \infty$.

Proof. Let $i(n,k) = n + \sum_{l<k} N_l$ denote the total number of samples so far, and define $Y_i = h(X_n^k) \frac{dQ}{dP_k}(X_n^k)$. Then by Eq. (5.2) we obtain that Y_i is an unbiased estimator of ψ, when conditioning on all samples up to i, i.e. $\mathbb{E}[Y_i \mid X_j, j < i] = \psi$. Therefore

$\{Y_i - \psi\}_{i>0}$ is a martingale difference sequence (see Definition 5.6). Furthermore, by the Minkowski inequality, we get that $\|Y_i - \psi\|_r \leq \|Y_i\|_r + \|\psi\|_r < C + \psi$, where we used Eq. (5.4) for the last inequality. We conclude that $Y_i - \psi$ is bounded uniformly in the L^r-norm. By Theorem 5.7, we obtain $I^{-1}\sum_{i=1}^{I} Y_i \to \psi$ almost surely as $I \to \infty$. Now note that $\hat{\psi}_k = I^{-1}\sum_{i=1}^{I} Y_i$ when $I = I(N_k, k) = \sum_{l \leq k} N_l$. □

Note that in the proof above we did not make any assumptions about the relative size between k and N_k. In particular the result is valid in the two extreme cases when $N_k = 1$ for all k and $K \to \infty$, or when K is finite and $N_k \to \infty$ for any k.

Example 5.2. Here we show that the condition of Eq. (5.4) in Theorem 5.1 is not redundant, by giving a sequence of proposals that will not yield a consistent estimate. Specifically, we consider the sampling problem that is given by

$$q(x) = \frac{dQ}{dx} = \frac{1}{\sqrt{2\pi}} \exp\left(-\tfrac{1}{2}x^2\right)$$
$$h(x) = \exp\left(-\tfrac{1}{2}x^2\right).$$

We will consider the class $\mathscr{P} = \{P^u : u \in \mathbb{R}\}$ of proposal distributions, where

$$p^u(x) = \frac{dP^u}{dx} = \frac{1}{\sqrt{2\pi}} \exp\left(-\tfrac{1}{2}(x-u)^2\right).$$

In Figure 5.1 we give a graphical representation of the importance sampling situation, with parameters $u = 1, 2, 3, 7$. Here you can see that the value of $h(x)q(x)/p_u(x)$ gets smaller in regions where $p^u(x)$ is large, when u increases (compare the dashed and the solid line). Indeed, it is not difficult to prove that for all $\gamma > 0$ we have $\lim_{u\to\infty} P^u(|h(X)\frac{dQ}{dP^u}(X)| > \gamma) = 0$. So if we take $u_k = k$, and $N_k = 1$ and consider the flat-AMIS estimate $\hat{\psi}_K = \sum_{k=1}^{K} h(X^k)\frac{dQ}{dP^k}(X^k)$, where $X^k \sim P^k = P^{u_k}$, then also $\lim_{K\to\infty} \Pr(\hat{\psi}_K > \gamma) = 0$ for all γ. In words: the estimator $\hat{\psi}_K$ goes to zero in probability when $K \to \infty$. In contrast, for all u, we have $\psi = \mathbb{E}_{P^u}[h(X)\frac{dQ}{dP^u}] = 1/\sqrt{2}$ (area under the dotted line in Figure 5.1). So, per definition, $\hat{\psi}_K$ is not a consistent estimator of ψ.

Theorem 5.1 holds for the generic proposal adaptation step in Algorithm 1, and condition Eq. (5.4) is the weakest that we were able to find. As a consequence, Eq. (5.4) is rather abstract and it might be hard to verify in practice. Therefore it might be sensible to replace Eq. (5.4) with a stronger condition that is easier to verify. We will do this for diffusion processes in Section 5.5.

5.4 AMIS with discarding

Flat-AMIS yields, in contrast to balance-AMIS, a provably consistent estimate, see Theorem 5.1. Furthermore, the computational complexity of flat-AMIS, is $\mathcal{O}(MK)$,

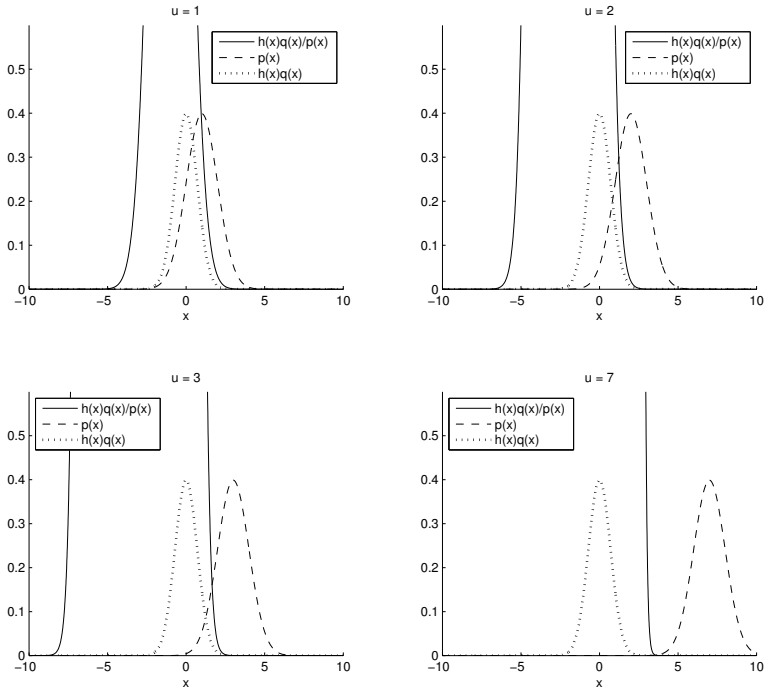

Figure 5.1: For $u = 1, 2, 3, 7$ we plot $h(x)q(x)/p^u(x)$ (solid line), $p^u(x)$ (dashed line) and the product $h(x)q(x)$ (dotted line). Although the overlap between $h(x)q(x)/p^u(x)$ and $p^u(x)$ becomes smaller for larger u, the product does not depend on u.

when $N_k = M$ for all k, which is optimal, while the complexity of balance-AMIS is $\mathcal{O}(MK^2)$. Nevertheless, balance-AMIS will outperform flat-AMIS in most practical applications. The reason is that in a flat re-weighting scheme, samples from a poor proposal typically dominate the computation, while this effect is averaged out by the balance-heuristic. In this section we will show that a simple modification of the flat re-weighting scheme results in an AMIS that is both consistent and computationally efficient.

The issue with flat re-weighting can be understood in more detail as follows. For a good proposal P_1 the terms $h\frac{dQ}{dP_1}$ do not deviate much from ψ. For a bad proposal P_2 most terms $h\frac{dQ}{dP_2}$ are close to zero, while a few will be exceptionally large compared to ψ. These large terms obviously dominate the IS estimate with P_2, but when mixing P_1 and P_2, the large terms from P_2 will also dominate over the samples from P_1. As a result, the mixing estimate might be worse than the IS estimate from the P_1 samples alone.

To improve upon flat re-weighting we therefore propose to simply ignore the samples from bad proposals. Since the idea of AMIS is that with each adaptation the proposal improves, one will expect that the variance decreases over time, and the quality of the samples improves. This brings us to the following algorithm which we will call discarding-AMIS, where we specifically choose the following re-weighting step.

Discarding-re-weighting (at iteration k)
Determine a discarding time $t_k \in \{1, 2, \dots, k-1\}$.
For $l = 1, \dots, t_k$ and $n = 1, \dots, N_l$, set $w_n^l = 0$.
For $l = t_k + 1, \dots, k$ and $n = 1, \dots, N_l$, set $w_n^l = k/(k - t_k)$.

Note that with this re-weighting the output at iteration k of discarding-AMIS is

$$\hat{\psi}_k = \frac{1}{\sum_{l=t_k+1}^{k} N_l} \sum_{l=t_k+1}^{k} \sum_{n=1}^{N_l} h(X_n^l) \frac{dQ}{dP_l}(X_n^l).$$

The discarding time t_k as given above is generic. We will now discuss two specific implementations of t_k that both have their merits.

The first choice is motivated by the consistency issue.

Remark 5.3. Theorem 5.1 still holds whenever t_k is chosen independently of the sampling process and when $\sum_{l=t_K+1}^{K} N_l \to \infty$. For example, one can take $t_k = \lceil k/2 \rceil$ so long as $k \to \infty$.

Secondly, let us consider a discarding time that aims to re-cycle the samples as efficiently as possible. When we have a measure of performance, we can utilize it to dynamically choose a discarding time that leaves us with the samples that yield the highest performance. For example, at iteration k we can calculate the

Effective Sample Size (ESS, see Eq. (5.10)) for all possible discarding times, and then choose the one that maximizes ESS. Clearly this will introduce a new level of dependence so that Theorem 5.1 no longer holds, and consistency is not guaranteed. The computational cost of checking the ESS for all discarding times at iteration k is $\mathcal{O}(Mk)$. If we do this at each iteration $k = 1, \dots, K$, we get a total complexity of $\mathcal{O}(MK^2)$, which is more than $\mathcal{O}(MK)$ for the computations of the weights of all samples over all iterations. The latter however, might have a much larger prefactor, so that in practice the cost for finding the best ESS is negligible. This is for example the case with diffusion processes. Alternatively, one could consider the ESS for a sparser set of possible discarding times, such as $t = 2^s$ for $s = 1, 2, \dots, \log(K)$, which will yield a complexity of $\mathcal{O}(MK \log(K))$.

In Section 5.7 we illustrate the difference in efficiency between $t_k = \lceil k/2 \rceil$ and ESS-optimized discarding.

5.5 Consistent AMIS for diffusion processes

In this section we apply AMIS in order to compute expected values over a diffusion process, i.e. with respect to the Wiener measure. By adding a drift to a diffusion process, we obtain a change in measure, and hence proposals that can be used for AMIS. We will give an easy to verify condition, involving the drift, that ensures consistency of flat-AMIS.

In case of Wiener noise, the target measure Q, will implicitly be given by an d-dimensional Itô process of the form

$$dX_t = \mu_t dt + \sigma_t dW_t, \tag{5.6}$$

with $(\mu_t)_{0 \le t \le T}$ and $(\sigma_t)_{0 \le t \le T}$ adapted processes of dimension d and $d \times m$ respectively, and W_t an m-dimensional Brownian motion. The function h in $\mathbb{E}_Q[h(X)]$ can be any function of the entire path: $h(X) = h\left((X_t)_{0 \le t \le T}\right)$.

If we have an adapted m-dimensional process $(u_t)_{0 \le t \le T}$, we can implement IS with the proposal P^u that is implicitly given by

$$dX_t = \mu_t dt + \sigma_t (u_t dt + dW_t). \tag{5.7}$$

Often the adapted processes are given as feedback functions: $u_t = u(t, X_t)$, $\mu_t = \mu(t, X_t)$, $\sigma_t = \sigma(t, X_t)$. Instead of an explicit formula for the densities densities dP^u/dx, dQ/dx with respect to a reference (e.g. Lebesgue) measure dx, we only have access to stochastic differential equations such as Eq. (5.7). On the upside, we will be able to generate (approximate) samples, for example by using the Euler-Maruyama method, [KP92]. So the goal in this scenario is not to generate samples close to the target Q; we can already do that by choosing $u = 0$. Instead, the aim of IS, or more generally, of AMIS, in this context, is to reduce the variance in the MC estimate of $\mathbb{E}_Q[h(X)] = \mathbb{E}_{P^u}[h(X) dQ/dP^u]$. In case of Wiener noise

we are able to compute the importance weight dQ/dP^u, which, by the Girsanov Theorem [KS91, Øks85], is given by:

$$\frac{dQ}{dP^u} = \exp\left(-\int_0^T u_t^\top dW_t - \frac{1}{2}\int_0^T u_t^\top u_t dt\right), \tag{5.8}$$

where we have used $\top$ to denote the transpose. Note that since this equation is exact, we do not have to worry about normalization.

Next, we will investigate consistency of flat-AMIS in case of Wiener noise. Let $\mathscr{U}$ be a class of of adapted processes $(u_t)_{0\le t\le T}$ and let $\mathscr{P} = \{P^u \mid u \in \mathscr{U}\}$ be the corresponding class of proposal measures. We will replace the abstract conditions Eq. (5.2, 5.4), that appear in Theorem 5.1, by some assumptions that, although stronger, are easier to verify in practice. The following theorem is a new result from [TK16].

Theorem 5.4. *Let $\mathscr{U}$ be a class of adapted processes. Suppose that*

1. *$\mathscr{U}$ is uniformly bounded in the L^∞ norm.*
2. *There is an $r > 1$ such that $h \in L^r(Q)$.*

Then flat-AMIS with proposals from the class $\mathscr{P} = \{P^u \mid u \in \mathscr{U}\}$ is consistent.

Proof. See the Proofs and definitions Section 5.8. □

If the adapted processes $u \in \mathscr{U}$ are given by feedback functions $u_t = u(t, X_t)$, then Condition 1 is, for example, satisfied when $\mathscr{U}$ is uniformly bounded. Similarly, if h is of the form $h(X) = \int_0^T H(X_t)dt$, or of the form $h(X) = H(X_T)$, for some function H, then Condition 2 is satisfied if H is bounded.

5.6 The choice of proposal

In this section, we propose a specific adaptation step for Algorithm 1 that can be used to sample over diffusion processes. We will adapt the proposal P^u by estimating a 'good' feedback function $u(t,x)$ that we can use in Eq. (5.7). Here we interpret 'good' as a function u such that P^u is close to an optimal proposal $P^\star$. For this optimal proposal to exist, we will assume for the remainder of this section that the function h is strictly positive. Note that if this is not the case, one can consider $h = (h_+ + 1) - (h_- + 1)$, where $h_+(x) = \max(h(x), 0)$ and $h_-(x) = \max(-h(x), 0)$, and compute $\mathbb{E}_Q[h_+ + 1]$ and $\mathbb{E}_Q[h_- + 1]$ separately.

Since h is strictly positive and $\mathbb{E}_Q[h(X)] < \infty$, the equation

$$\frac{dP^\star}{dQ} = \frac{h(X)}{\mathbb{E}_Q[h(X)]}, \tag{5.9}$$

defines a measure $P^\star$ that is equivalent to Q, which means, loosely speaking, that their densities have the same support. The measure $P^\star$ is the optimal proposal because IS with $P^\star$ gives zero variance estimates $h(X)dQ/dP^\star = \mathbb{E}_Q[h(X)]$. Note that, for the time being, this optimality is not of practical interest, since the definition of $P^\star$ requires $\mathbb{E}_Q[h(X)]$, which is what we want to evaluate in the first place.

One might wonder whether there exists an optimal process $(u_t^\star)_{0\leq t\leq T}$ satisfying $P^\star = P^{u^\star}$. We have seen in Chapter 4, when taking $C = -\log(h)$, that under certain conditions there indeed is such a $u^\star$. Furthermore we have seen that the solution to this problem is also the solution of a KL control problem, and in Section 3.9 we have seen that close to optimal controls yield close to zero variance. Therefore, stochastic control theory can be applied in order to find a good proposal-feedback-function $u(t,x)$ with corresponding proposal measure P^u. The idea is to use the control computations, that are further detailed in Chapter 6, for the proposal update in the **Adaptation** step in the generic AMIS as given in Algorithm 1. In Section 5.7 we give a demonstration of the path integral adaptation, where it is used to implement various AMIS algorithms for an example sampling problem over a diffusion processes.

5.7 Numerical example

In this Section we provide a numerical example in which we compare discarding-AMIS with balance-AMIS. In both cases the adaptation step will be implemented as proposed in Section 5.6, via a path integral control computation, which is detailed in Section 6.3.

We compare the various re-weighting schemes of AMIS in terms of Effective Sample Size (ESS). In the literature [CMMR12, EMLB15b] this is often defined by $N/\left(1+\mathrm{Var}_P\left[\frac{dQ}{dP}\right]\right)$. However, since our goal is to minimize the variance of $h\frac{dQ}{dP}$, we instead consider

$$\mathrm{ESS}^P = \frac{N}{1+\mathrm{Var}_P\left[h\frac{dQ}{dP}\right]\psi^{-2}} = \frac{N}{1+N\,\mathrm{Var}_P\left[\hat{\psi}^P\right]\psi^{-2}},$$

where the second equality follows from $\mathrm{Var}_P[h\frac{dQ}{dP}] = N\,\mathrm{Var}_P[\hat{\psi}^P]$,which is a consequence of the definition in Eq. (5.1). We remark that $\lambda^P = \mathrm{ESS}^P/N$ is the *fraction of effective samples*, which we analyzed in the context of diffusion processes in Section 3.9; there we prove that proposals that are close to the optimal proposal, have a fraction of effective samples that is close to 1, which is optimal.

Similar to [EMLB15b] we generalize the ESS for MIS:

$$\mathrm{ESS}^{\mathrm{MIS}} = \frac{N}{1+N\,\mathrm{Var}_P\left[\hat{\psi}^{\mathrm{MIS}}\right]\psi^{-2}},$$

which can be evaluated approximately with the following sample estimate

$$\widehat{\mathrm{ESS}} = \frac{\left(\sum_{nk} y_{nk}\right)^2}{\sum_{nk} \left(y_{nk}\right)^2}, \tag{5.10}$$

where $y_{nk} = h(X_n^k)\frac{dQ}{dP_k}(X_n^k)w_k(X_n^k)$, with $X_n^k \sim P_k$. The estimator $\widehat{\mathrm{ESS}}$ takes values between 1 (when all but one y_{nk} are zero) and N (when all y_{nk} are equal, which happens with positive probability iff $P = P^\star$).

In the following example we will describe a sampling problem that will be used to compare discarding-AMIS with balance-AMIS.

Example 5.5. This example is similar to Example 5.2, where we interpret X as a diffusion process, and generalize to $\mathbb{R}^d$. The target measure Q is implicitly given by an d-dimensional standard Brownian motion. This is the process $(X_t)_{0\le t\le 1}$ as given in Eq. (5.6) with $X_0 = 0 \in \mathbb{R}^d$ and a constant drift and diffusion equal to $\mu_t = 0, \sigma_t = 1 \in \mathbb{R}^d$. The target function is a Gaussian function centered around a target point $z \in \mathbb{R}^d$:

$$h((X)_{0\le t\le 1}) = \exp\left(-\tfrac{1}{2}(X_1 - z)^\top (X_1 - z)\right).$$

For importance sampling we will consider two different classes of proposals P^u, corresponding with two different parameterizations of u that are of the linear form $u = Ag(t,x)$, as detailed in Section 6.2.

The first case that we consider is $g = 1 \in \mathbb{R}$. The class of proposals that corresponds to $g = 1$ is in a sense the same class as we used in Example 5.2. It is a degenerate case of a diffusion process: since the control $u(t,x) = A \in \mathbb{R}^d$ is constant, all states, except the end state X_1, of the entire path $(X_t)_{0\le t\le 1}$ can be ignored, because h is only a function of X_1.

The second class is constructed with $g = (1,x) \in \mathbb{R}^{1+d}$. The corresponding $u(t,x) = Ag$ with $A \in \mathbb{R}^{(d+1)\times d}$ are linear feedback controls, making the intermediate states X_t with $t < 1$ relevant to the distribution of X_1. This more complex parametrization will give us more control over the process, and hence more flexibility in finding a good proposal. So although we need to learn a parameter A with a higher dimension, we expect a higher ESS.

We use Example 5.5 with $d = 3$, and $z = 2 \cdot \mathbb{1} \in \mathbb{R}^3$ (where $\mathbb{1}$ is the vector with all ones) in order to compare four types of AMIS algorithms. All four methods will be implemented with the same adaptation step; the path integral adaptation that is described in Section 6.3, using a parametrization based on $g = 1$. The difference between the four methods comes from the following different re-weighting schemes that they use:

- Balance re-weighting, with $N_k = 1$ sample per iteration.

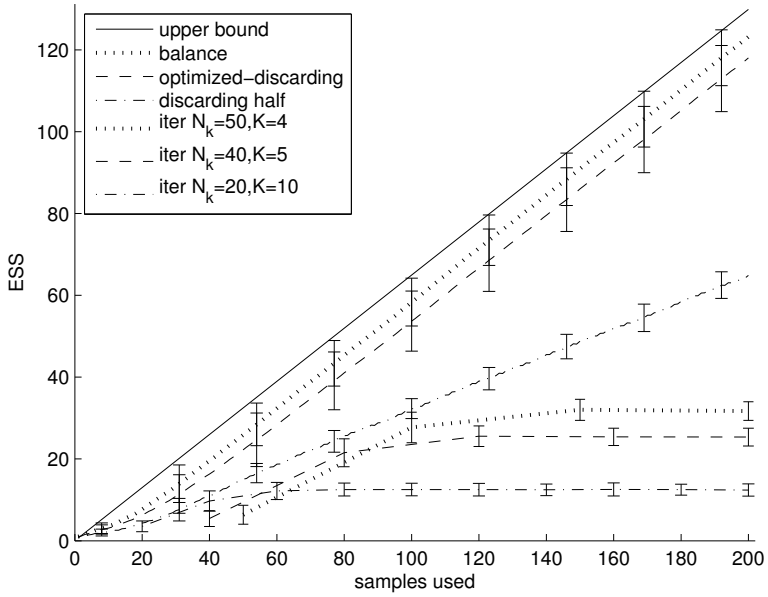

Figure 5.2: Average $\widehat{\text{ESS}}$ of AMIS for various re-weighting schemes with adaptation based on $g = 1$, taken over 100 independent runs. On the x-axis are the number of samples that are used so far in the AMIS estimate, i.e., $\sum_{i=1}^{k} N_i$ for $k = 1, \ldots, K$.

- Optimized discarding time, i.e. flat re-weighting with t_k that maximizes $\widehat{\text{ESS}}$ and $N_k = 1$ sample per iteration.
- Flat re-weighting with discarding time $t_k = \lceil k/2 \rceil$ and $N_k = 1$ sample per iteration.
- An iterative non-mixing scheme, with constant batch sizes N_k, where only the samples of the last iteration are used, i.e. with $w_K \propto 1$ and $w_k = 0$ for $k < K$.

We report how the $\widehat{\text{ESS}}$, averaged over 100 runs, increased with the number of samples that was used by each of the AMIS algorithms, see Figure 5.2. The upper bound is the optimal achievable ESS within the class of proposals corresponding to $g = 1$. The slope of this upper bound is $\max\{\text{ESS}^{P^u} : u = A,\ A \in \mathbb{R}^d\} = (3/4)^d$. The two methods that perform the best are balance-AMIS and optimized-discarding-AMIS. These two methods make optimal use of new samples when the underlying parameter A has converged to it's optimal value, as can be seen by the slope that matches the upper bound. When discarding half of the samples, the slope in average $\widehat{\text{ESS}}$ is halved as well. The iterative non-mixing schemes perform the worst: the

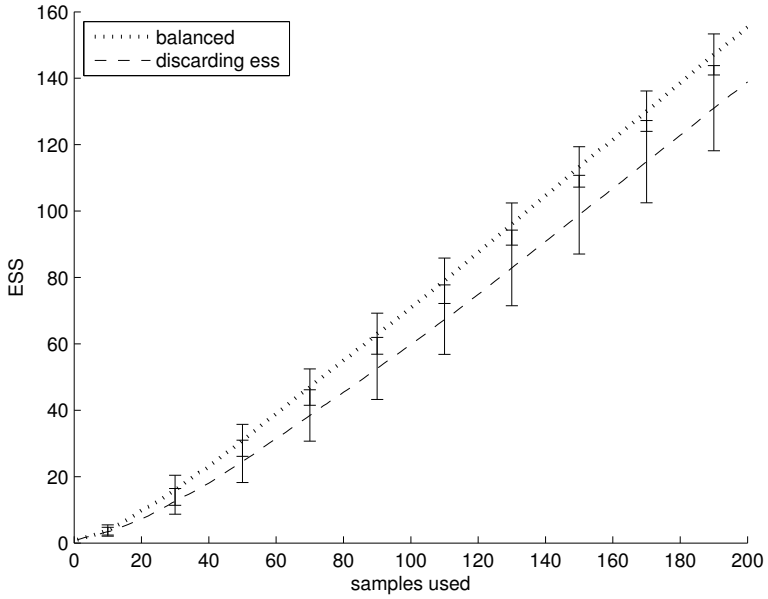

Figure 5.3: Average $\widehat{\mathrm{ESS}}$ of balance-AMIS and optimized-discarding-AMIS with adaptation based on $g = (x, 1)$, taken over 100 independent runs. On the x-axis are the number of samples that are used so far in the AMIS estimate, i.e., $\sum_{i=1}^{k} N_i$ for $k = 1, \ldots, K$.

average $\widehat{\mathrm{ESS}}$ never uses more than the N_k samples of one batch in the estimate, and therefore $\widehat{\mathrm{ESS}}$ does not increase with k, or equivalently with the total number of samples used.

A fifth method that we also tested uses flat re-weighting without discarding. Although this method seemed to perform reasonably on average, this result is not significant because of very large error bars. For these reasons this method is not included in Figure 5.2.

We make the same comparison between optimized-discarding-AMIS with balance-AMIS while using the more complex parametrization with $g = (1, x)$. The results are reported in Figure 5.3. Compared to the experiment with $g = 1$ we notice that the average $\widehat{\mathrm{ESS}}$ is higher, and that the two methods perform similar; balance-AMIS is slightly better. However, balance-AMIS also required a lot more computational resources to produce these results, see Table 5.1. From the table it appears that computation time for balance-AMIS is roughly proportional to the number of iterations. This is exactly what one would expect based on the complexity $\mathcal{O}(K^2 M) = \mathcal{O}(KN)$

Table 5.1: Computation time in seconds for 100 runs of AMIS based on $g = (1, x)$ for a fixed number $N = \sum_{k=1}^{K} N_k = 200$ of samples, but with different numbers of iterations K.

K	10	25	50	100	200
balance	46	104	200	392	780
optimized-discarding	5.9	6.2	6.4	6.9	7.9

of the adaptation step, when the total number of samples $N = 200$ is fixed. This complexity can be avoided in the adaptation step for optimized-discarding, because with that re-weighting scheme the weights are not changed in future iterations. We conclude that for a given number of samples optimized-discarding-AMIS performs almost as well as balance-AMIS, but at a fraction of the computational cost.

5.8 Proofs and definitions

Lemma 5.9 and all proofs from this section are new results from [TK16].

Definition 5.6. Let $\{X_n\}_{n>0}$ be a sequence of random variables that is adapted to the filtration $\{\mathscr{F}_n\}_{n>0}$. Then X is called a Martingale Difference Sequence w.r.t. $\mathscr{F}$ when for all $n > 0$

1. $\mathbb{E}[X_n] < \infty$,
2. $\mathbb{E}[X_{n+1} \mid \mathscr{F}_n] = 0$.

Theorem 5.7 (Generalized Strong Law of Large Numbers)**.** *Let $\{Y_n\}_{n>0}$ be a Martingale Difference Sequence relative to $\{\mathscr{F}_n\}_{n>0}$. If $\{Y_n\}_{n>0}$ is uniformly bounded in the L^r-norm for some $r > 1$, then*

$$\frac{1}{N}\sum_{n=1}^{N} Y_n \to 0 \text{ almost surely, as } N \to \infty.$$

Proof. This is proven in [Jon95] for Mixingale Sequences, which are more general than Martingale Difference Sequences. □

Remark 5.8. The theorem above does not generalize to the case $r = 1$. However, for $r = 1$ there is a weak law (convergence in probability) when the set $\{Y_n\}_{n>0}$ is uniformly integrable, see [And88].

Lemma 5.9. *Let $\mathscr{U}$ be a set of adapted processes. Suppose that $\mathscr{U}$ is uniformly bounded in the L^∞ norm, i.e. there is a constant C such that for all $u \in \mathscr{U}$,*

$$\|u\|_\infty = \inf\{A \geq 0 : |u_t| < A \text{ for all } t, P^u\text{-almost surely}\} < C.$$

Then dQ/dP^u is bounded uniformly in the L^r-norm for all $r \geq 1$.

Proof. Let $r \geq 1$ and $u \in \mathscr{U}$. Choose a such that $a > r$.

$$\begin{aligned}\|dQ/dP^u\|_r &= \left\|\exp\left(-\textstyle\int_0^T u_t^\top dW_t - \frac{1}{2}\int_0^T u_t^\top u_t dt\right)\right\|_r \\ &= \left\|\exp\left(-\textstyle\int_0^T u_t^\top dW_t - \frac{a}{2}\int_0^T u_t^\top u_t dt\right)\exp\left(-\frac{1-a}{2}\int_0^T u_t^\top u_t dt\right)\right\|_r.\end{aligned}$$

Let b be such that $\frac{1}{r} = \frac{1}{a} + \frac{1}{b}$. Then $b > 1$, and by Hölders Inequality and the bound on u we obtain respectively

$$\begin{aligned}\|dQ/dP^u\|_r &\leq \left\|\exp\left(-\textstyle\int_0^T u_t^\top dW_t - \frac{a}{2}\int_0^T u_t^\top u_t dt\right)\right\|_a \left\|\exp\left(\frac{a-1}{2}\int_0^T u_t^\top u_t dt\right)\right\|_b \\ &< \left\|\mathscr{E}\left(-a\textstyle\int_0^T u_t^\top dW_t\right)\right\|_1^{1/a} \exp\left((a-1)C^2T/2\right).\end{aligned}$$

The Doléan exponential $\mathscr{E}\left(-a\int_0^T u_t^\top dW_t\right)$ is a local martingale that is positive. Hence, it is a super martingale, so that

$$\|dQ/dP^u\|_r < \exp\left((a-1)C^2T/2\right). \qquad \square$$

Proof of Theorem 5.4. Note that Assumption 1 of the theorem implies Novikov's Condition, so that $P^u \sim Q$ for all $u \in \mathscr{U}$. So in particular we have $P^u \ll Q$ and and therefore the condition of Eq. (5.2) holds.

Now we will show that condition Eq. (5.4) also holds, so that consistency follows from Theorem 5.1. Let $r > 1$, and choose $a, b > 1$ such that $\frac{1}{r} = \frac{1}{a} + \frac{1}{b}$. Then, using Hölders inequality, we get

$$\left\|h\tfrac{dQ}{dP}\right\|_r \leq \|h\|_a \left\|\tfrac{dQ}{dP}\right\|_b.$$

Now, $\|h\|_a$ is bounded by Assumption 2 of the theorem, and $\left\|\frac{dQ}{dP}\right\|_b$ is bounded by Lemma 5.9. $\square$

Chapter 6

The path integral control algorithm

6.1 Introduction

Since in a stochastic system there is uncertainty about the future states, a state independent feed forward control cannot be optimal. Instead, the optimal control must in general be a function of the state, i.e., a feedback control. There are two ways to implement feedback. Either the optimal control action is be re-computed on the fly each time a new state is visited, or a state feedback control function is learned in advance of the problem. Although the former solution is quite resource intensive, it can be implemented relatively straightforward. In Chapter 7 we give a detailed description of this approach. In this chapter we will show how the Main Path Integral Control Theorem can be used in order to construct parametrized feedback control functions.

In order to construct our feedback controllers we shall need to evaluate path integrals, which is achieved with Monte Carlo integration. The computations that are required can be very expensive due to high variance. Fortunately we have already addressed this issue: in Section 3.9 we have shown that importance sampling can be implemented efficiently with control functions, and in Chapter 5 we analyzed how to optimize the sampling scheme. These results will all be put together in this Chapter, resulting in the Path Integral Control Algorithm.

Outline. This chapter is structured as follows. In Section 6.2 we propose a parametrization of the control, and we show how to obtain a good parameter with path integrals. In Section 6.3 we combine this with MC integration and sampling techniques, in order to create feedback control functions. We give an illustration of the constructed algorithm in Section 6.4.

6.2 Parametrized control

In this section we illustrate how the Main Path Integral Control Theorem can be used to construct a feedback controller $\hat{u}$, that will in a sense approximate $u^\star$. The starting point is that we choose a linear parametrization of the form

$$\hat{u}(t,x) = Ag(t,x).$$

Here $g(t,x) \in \mathbb{R}^l$ is an l-dimensional basis function that should be chosen by the user in advance of running the path integral control algorithm, and $A \in \mathbb{R}^{m\times l}$ is a parameter that shall be optimized by the algorithm. The idea behind the algorithm is to approximate $u^\star$ by $\hat{u}$ by optimizing over A. We can do this, for given g (and a given importance sampling control u), if we substitute $\hat{u}$ for $u^\star$ in Eq. (3.16) of the Main Path Integral Control Theorem. This will yield a system of equations that can be solved for A as follows:

$$A^\star = \mathbb{E}_{P^u}\left[e^{-S^u}\int_{t_0}^{t_1}(u_t\,dt + dW_t)g_t^\top\right]\left(\mathbb{E}_{P^u}\left[e^{-S^u}\int_{t_0}^{t_1} g_t g_t^\top dt\right]\right)^{-1}. \tag{6.1}$$

Here $\mathbb{E}_{P^u}$ denotes that the expected value is w.r.t. the process given by Eq. (3.1), and $S^u = S^u_{t_0}$ is defined as in Eq. (3.2).

The solution $A^\star$ is optimal in the sense that the corresponding $P^{\hat{u}^\star}$ (where $\hat{u}^\star = A^\star g$) minimizes the Kullback-Leibler divergence between $P^\star$ and $P^{\hat{u}}$, i.e. $\mathrm{D}_{\mathrm{KL}}(P^\star \| P^{\hat{u}})$, over the class of proposal feedback functions with the given parametrization $\{\hat{u} = Ag \mid A \in \mathbb{R}^{m\times l}\}$, see [KR16, BKMR05]. Interestingly, the optimal control problem, when seen as a KL control problem, requires us to minimize $\mathrm{D}_{\mathrm{KL}}(P^{\hat{u}} \| P^\star)$ instead; see Eq. (4.2).

The smallest possible divergence $\mathrm{D}_{\mathrm{KL}}(P^\star \| P^{\hat{u}})$ will depend on the function $g(t,x) \in \mathbb{R}^l$. Generally, complex g, i.e. with l large, yield more expressive power. In practice, however, there is a trade-off, since it is difficult to obtain good estimates of Eq. (6.1), when l is large. From a more practical point of view, it should be noted that the scenario where the algorithm is applied might put constraints on g. Whether or not that is the case, it is clear that the choice of the function g is very important, because it determines what kind of controller you will create. For example, two types of controllers, which have perhaps been applied the most, are (1) the open-loop feed forward controller, and (2) a linear feedback controller. The (time constant) open-loop controller can be realized with $g = 1$, and the linear feedback controller with $g = (1, x)$. Note that time dependence can be introduced by using piecewise time-constant controls, i.e. with functions of the form $g(t,x) = \sum_i g(x)\mathbb{1}_{t\in\Delta_i}$, where the Δ_i are small time intervals that cover $[t_0, t_1]$.

6.3 Control computations

In this section we give a detailed description of the *path integral control algorithm*. This algorithm computes MC estimates of expected values over a diffusion process. Therefore the algorithm can be used to solve a sampling problem, as described in Section 5.5, but it can also be used to solve a path integral control problem as described in Section 3.2. In order to compute the MC estimates efficiently we will use adaptive multiple importance sampling (AMIS), as descried in Chapter 5. Therefore the overall structure of the path integral control algorithm is very similar to Algorithm 1 in Section 5.2.

Algorithm 2 Path Integral Control Algorithm with AMIS

Iterate. For $k = 1, \dots, K$ do

Path Integral Adaptation. Construct a feedback control $u_k(t,x)$, possibly depending on X_n^l and w_n^l with $1 \leq l < k$ and $1 \leq n \leq N_l$.

Generation. For $n = 1, \dots, N_k$ draw sample paths $X_n^k \sim P_k$, where $P_k = P^{u_k}$, and compute the cost S_n^k of the path.

Re-weighting. For $n = 1, \dots, N_k$, construct w_n^k.
For $l = 1, \dots, k-1$ and $n = 1, \dots, N_l$, update w_n^l.

Intermediate Output. Return $\hat{J}_k^\star = -\log \frac{1}{\sum_{l=1}^k N_l} \sum_{l=1}^k \sum_{n=1}^{N_l} e^{-S_n^l} w_n^l$, and return $u_k(t,x)$.

Output. Return the optimal control estimate $u_{k+1}(t,x)$ that would be computed by the Adaptation at iteration $K+1$.

This algorithm runs $k = 1, \dots, K$ iterations, and at each iteration four steps are executed. The first step is the **Path Integral Adaptation**, that implements the control computations for importance sampling. This step contains the core of path integral control. It is explained in more detail below. The second step is **Generation**. Here N_k samples paths are drawn from the process given in Eq. (3.1) with importance control $u(t,x) = u_k(t,x)$, and the cost of each path is calculated using Eq. (3.2). Approximate samples from a diffusion process can for example be obtained with the Euler-Maruyama method, or similar higher order schemes, see [KP92]. The third step is the computation of the **Re-weighting**. This step is discussed in detail in Sections 5.3 and 5.4. In the **Intermediate Output** step we return the estimate $J^\star$, based on Eq. (3.15).

Next, we give a detailed description of the **Path Integral Adaptation**, that is given in pseudo code in Algorithm 2. Given a parametrization of the control

Algorithm 3 Path Integral Adaptation (at iteration k)

- If $k = 1$ (initialization), set:
  $A_k = 0 \in \mathbb{R}^{m \times l}$,
  $F_k = 0 \in \mathbb{R}^{m \times l}$,
  $G_k = 0 \in \mathbb{R}^{l \times l}$.

- Else, if $k > 1$,
  For $n = 1, \ldots, N_{k-1}$, set

$$
\begin{aligned}
g_t^{n,k-1} &= g\left(t, \left(X_n^{k-1}\right)_t\right), \\
u_t^{n,k-1} &= A_l g_t^{n,k-1}, \\
R_t^{n,k-1} &= R\left(t, \left(X_n^{k-1}\right)_t\right) \\
S_n^{k-1} &= Q\left(X_{t_1}\right) + \int_{t_0}^{t_1} u_t^{n,k-1\top}\left(dW_n^{k-1}\right)_t + \frac{1}{2} u_t^{n,k-1\top} u_t^{n,k-1} dt + R_t^{n,k-1} dt.
\end{aligned}
$$

  And set

$$
\begin{aligned}
G_k &= \sum_{l=1}^{k-1} \sum_{n=1}^{N_l} h\left(X_n^l\right) e^{-S_n^l} w_n^l \int_{t_0}^{t_1} g_t^{n,l} g_t^{n,l\top} dt, \\
F_k &= \sum_{l=1}^{k-1} \sum_{n=1}^{N_l} h\left(X_n^l\right) e^{-S_n^l} w_n^l \int_{t_0}^{t_1} \left(u_t^{n,l} dt + \left(dW_n^l\right)_t\right) g_t^{n,l\top} dt, \\
A_k &= F_k G_k^{-1}.
\end{aligned}
$$

  Let $u_k(t,x) = A_k g(t,x)$.
  Let P_k be the measure induced by Eq. (3.1) with $u = u_k$.

$u(t,x) = Ag(t,x)$, for a given function $g(t,x)$, Algorithm 2 computes a MC estimate of the optimal cost to go $J^\star$ as given in Eq. (3.15), and the optimal parameter $A^\star$ as given in Eq. (6.1). The MC estimates are computed sequentially, where at each iteration $k = 1,\dots,K$ we draw N_k new samples. At iteration k we use importance sampling with the estimate A_k of $A^\star$, so that the combined sample estimate over multiple iterations is an AMIS estimate. More specifically, at iteration k we draw N_k samples from the proposal distribution $P_k = P^{u_k}$, where $u_k(t,x) = A_k g(t,x)$. The parameters A_k are computed with samples from previous iterations according to Eq. (6.1), which we state here again

$$
\begin{aligned}
A^\star &= F^\star G^{\star -1},\\
F^\star &= \mathbb{E}_{P^u}\left[h(X)\frac{dQ}{dP^u}\int_{t_0}^{t_1}(u_t dt + dW_t)g_t^\top \right],\\
G^\star &= \mathbb{E}_{P^u}\left[h(X)\frac{dQ}{dP^u}\int_{t_0}^{t_1} g_t g_t^\top dt \right].
\end{aligned}
$$

At each iteration k the terms $F^\star$ and $G^\star$ will be estimated by F_k and G_k respectively.

Note that the path weights $e^{-S_n^l}$ are also required in the **Intermediate Output** step of Algorithm 2. The time integrals above could, for example, be estimated approximately with the Euler-Maruyama method. We remark that the algorithm above is of order $\mathcal{O}(MK^2)$ when $N_k = M$ for all k, because it is given for a generic re-weighting scheme. When flat- or discarding-re-weighting is used, G_k and F_k might be computed incrementally because the re-weighting does not change once set, dropping the first sum, so that the algorithm becomes $\mathcal{O}(MK)$.

6.4 Example

In this section we give an illustration of the path integral control algorithm that is described in the previous section. The focus of the illustration is on the path integral adaptation, and therefore we run Algorithm 2 with only $K = 1$ iteration, i.e. we do not use AMIS (for an example that illustrates the effect of AMIS, see Section 5.7). We will illustrate the effect of various choices of the parameterizing function $g(t,x)$ on the quality of the solution $u(t,x) = Ag(t,x)$.

For the illustration we consider the following control problem, of which we know the analytical solution.

Example 6.1 (Geometric Brownian Motion)**.** The path integral control problem with dynamics and cost given respectively by

$$
\begin{aligned}
dX_t^u &= X_t^u\left((1/2 + u_t)dt + dW_t\right),\\
S^u &= 5\log(X_1^u)^2 + \frac{1}{2}\int_0^1 u_t^2 dt + \int_0^1 u_t^\top dW_t,
\end{aligned}
$$

Table 6.1: Performance estimates of various controllers based on 10^4 sample paths. Although for numerical consistency we used 10^4 sample paths to compute the parameters, only roughly 10^2 samples are required to obtain well-performing controllers.

	$u=0$	$u^{(0)}$	$u^{(1)}$	$u^{(2)}$	$a(t)\log(x)$	$u^\star$
$\mathbb{E}[S^u_{t_0}]$	7.526	5.139	1.507	1.461	1.422	1.420
$\mathrm{Var}(\alpha^u)$	1.981	1.376	0.143	0.0506	0.0085	0.0071
ESS^u	3354	4208	8748	9518	9915	9929

with $0 \le t \le 1$ and initial state $X_0 = 1/2$, has solution

$$u^\star(t,x) = \frac{-5\log(x)}{5(1-t)+1}.$$

It is clear that a very good parametrization of the problem in Example 6.1 can be obtained with the basis functions: $g(t,x) = \log(x)\mathbb{1}_{t\in\Delta_l}$, where the Δ_l are small time intervals. This is because the state dependence, $\log(x)$, is exactly as in $u^\star$. So the path integral control algorithm only has to approximate the time dependence on the intervals Δ_l. We shall also consider three parametrizations that cannot describe $u^\star$ very well: a constant, an affine, and a quadratic function of the state. The three controllers that we obtain in this way are denoted by $u^{(0)}$, $u^{(1)}$, and $u^{(2)}$, e.g., $u^{(2)}(t,x) = a(t) + b(t)x + c(t)x^2$. The connection between the time dependent functions a, b and c with the parameterizing g is, for example in the state-independent case that $g = \mathbb{1}_{t\in\Delta_l}$ and $a(t) = \sum_l A_l \mathbb{1}_{t\in\Delta_l}$, and analogously for b and c. Here, the A_l are to be estimated (in this case, independently for each tine interval Δ_l) by the path integral adaptation.

The performance of the resulting control functions is given in Table 6.1. The row $\mathbb{E}[S^u(t_0)]$ gives the expected cost, which we want to minimize. The row $\mathrm{Var}(\alpha^u)$ gives the estimated variance of the normalized path weight $\alpha^u = e^{-S^u}/\psi$, which is directly related to the estimated ESS^u as given in Eq. 5.10, by $\mathrm{EES}^u = N/(\mathrm{Var}(\alpha^u)+1)$.

Clearly the open-loop controller $u^{(0)}(t,x) = a(t)$ improves upon the zero controller $u(t,x) = 0$. The control further improves when the affine and quadratic basis functions are subsequently considered. The best result is obtained, unsurprisingly, with the logarithmic parametrization.

In Figure 6.1 we plot the state dependence of the feedback controllers at the intermediate time $t = 1/2$. Although the parametrized functions yield a control for all x, we are mainly interested in regions of the state space that are likely to be visited by the optimal process $X^\star_t$. This is visualized by a histogram of 10^4 particles that are drawn from $X^\star_{t=1/2}$. We observe that the optimal logarithmic shape is fitted,

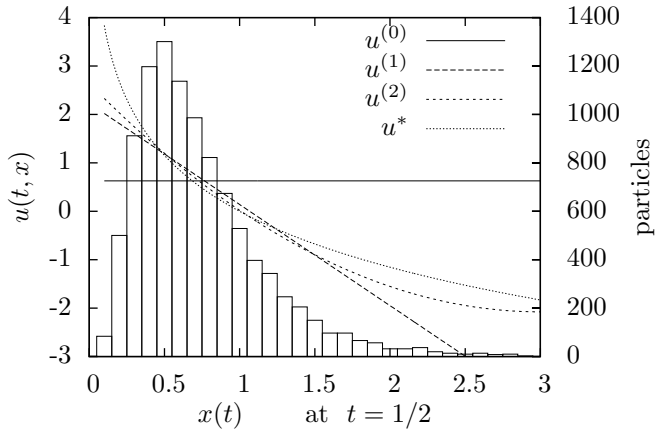

Figure 6.1: The approximate controls calculated with 10^4 sample paths in two importance sampling iterations using a time discretization of $dt = 0.001$ for numerical integration. The histogram was created with 10^4 draws from $X_t^{u^*}$ at $t = 1/2$.

and that more complex parametrizations yield a better fit in the important regions.

Chapter 7

Real-time stochastic optimal control for multi-agent quadrotor systems

7.1 Introduction

The recent surge in autonomous Unmanned Aerial Vehicle (UAV) research has been driven by the ease with which platforms can now be acquired, evolving legislation that regulates their use, and the broad range of applications enabled by both individual platforms and cooperative swarms. Example applications include automated delivery systems, monitoring and surveillance, target tracking, disaster management and navigation in areas inaccessible to humans.

Quadrotors are a natural choice for an experimental platform, as they provide a safe, highly-agile and inexpensive means by which to evaluate UAV controllers. Figure 7.1 shows a 3D model of one such quadrotor, the *Ascending Technologies Pelican*. Quadrotors have non-linear dynamics and are naturally unstable, making control a non-trivial problem.

Stochastic optimal control (SOC) provides a promising theoretical framework for achieving autonomous control of quadrotor systems. In contrast to deterministic control, SOC directly captures the uncertainty typically present in noisy environments and leads to solutions that qualitatively depend on the level of uncertainty [Kap05b]. However, with the exception of the simple Linear Quadratic Gaussian case, for which a closed form solution exists, solving the SOC problem requires solving the Hamilton-Jacobi-Bellman (HJB) equation. This equation is generally intractable, and so the SOC problem remains an open challenge.

In such a complex setting, a hierarchical approach is usually taken and the con-

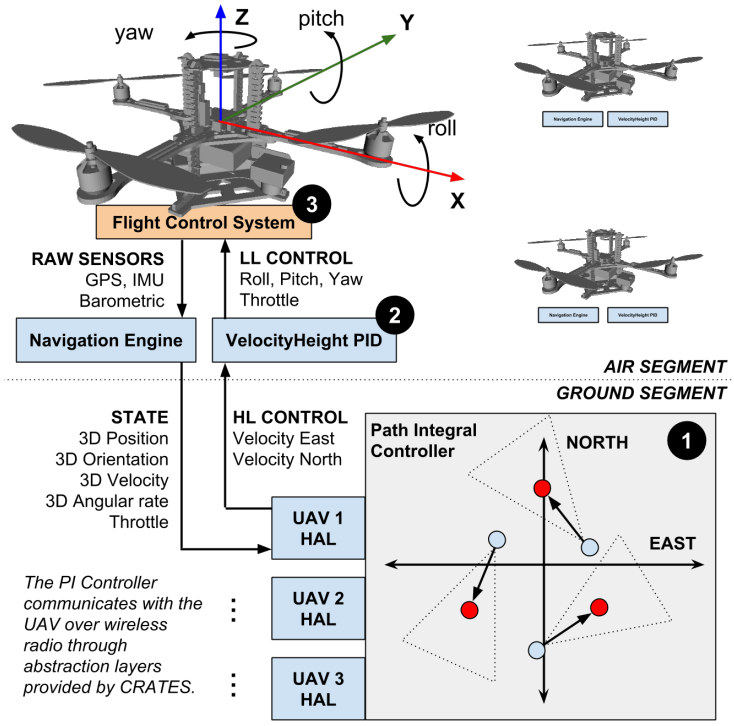

Figure 7.1: Control hierarchy: The path integral controller (1) calculates target velocities/heights for each quadrotor. These are converted to roll, pitch, throttle and yaw rates by a platform-specific Velocity Height PID controller (2). This control is in turn passed to the platform's flight control system (3), and converted to relative motor speed changes.

trol problem is reduced to follow a state-trajectory (or a set of way points) designed by hand or computed offline using trajectory planning algorithms [Ken12]. While the planning step typically involves a low-dimensional state representation, the control methods use a detailed complex state representation of the UAV. Examples of control methods for trajectory tracking are the Proportional Integral Derivative or the Linear-Quadratic regulator.

We introduced a generic class of SOC problems in Chapter 3 for which the controls and the cost function are restricted in a way that makes the HJB equation linear and therefore more efficiently solvable. This class of problems is known as path integral (PI) control, linearly-solvable controlled diffusions or Kullback-Leibler control, and it has lead to successful robotic applications, e.g. [KUD13, RTM+12b, TBS10]. A particularly interesting feature of this class of problems is that the

computation of optimal control is an inference problem with a solution given in terms of the passive dynamics. In a multi-agent system, where the agents follow independent passive dynamics, such a feature can be exploited using approximate inference methods such as variational approximations or belief propagation [KGO12, BWK08b].

In this chapter, we show how PI control can be used for solving motion planning tasks on a team of quadrotors in real time. We combine periodic re-planning with receding horizon, similarly to model predictive control, with efficient importance sampling. At a high level, each quadrotor is modelled as a point mass that follows simple double integrator dynamics. Low-level control is achieved using a standard Proportional Integral Derivative (PID) velocity controller that interacts with a real or simulated flight control system. With this strategy we can scale PI control to ten units in simulation. Although in principle there are no further limits to experiments with actual platforms, our first results with real quadrotors only include three units. To the best of our knowledge this has been the first real-time implementation of PI control on an actual multi-agent system.

In the next section we describe related work. We introduce our approach in Section 7.3. Results are shown on three different scenarios in Section 7.4. Finally, Section 7.5 concludes this chapter.

7.2 Related work on UAV planning and control

There is a large and growing body of literature related to this topic. In this section, we highlight some of the most related literature to the presented approach. An recent survey of control methods for general UAVs can be found in [Ken12].

Stochastic optimal control is mostly used for UAV control in its simplest form, assuming a linear model perturbed by additive Gaussian noise and subject to quadratic costs (LQG), e.g. [HBF+08]. While LQG can successfully perform simple actions like hovering, executing more complex actions requires considering additional corrections for aerodynamic effects such as induced power or blade flapping [HHWT11]. These approaches are mainly designed for accurate trajectory control and assume a given desired state trajectory that the controller transforms into motor commands.

Model Predictive Control (MPC) has been used to optimize trajectories in multi-agent UAV systems [SKS03]. MPC employs a model of the UAV and solves an optimal control problem at time t and state $x(t)$ over a future horizon of a fixed number of time-steps. The first optimal move $u^\star(t)$ is then applied and the rest of the optimal sequence is discarded. The process is repeated again at time $t+1$. A quadratic cost function is typically used, but other more complex functions exist.

MPC has mostly been used in indoor scenarios, where high-precision motion capture systems are available. For instance, in [KMPK13] authors generate smooth trajectories through known 3-D environments satisfying specifications on intermediate waypoints and show remarkable success controlling a team of 20 quadrotors.

Trajectory optimization is translated to a relaxation of a mixed integer quadratic program problem with additional constraints for collision avoidance, that can be solved efficiently in real-time. Examples that follow a similar methodology can be found in [TMK12, ASD12]. Similarly to our approach, these methods use a simplified model of dynamics, either using the 3-D position and yaw angle [KMPK13, TMK12] or the position and velocities as in [ASD12]. However, these approaches are inherently deterministic and express the optimal control problem as a quadratic problem. In our case, we solve an inference problem by sampling and we do not require intermediate trajectory waypoints.

In outdoor conditions, motion capture is difficult and Global Positioning System (GPS) is used instead. Existing control approaches are typically either based on Reynolds flocking [BSK11, HLV+11, VVS+14, Rey87] or flight formation [GL12, YDSZ13]. In Reynolds flocking, each agent is considered a point mass that obeys simple and distributed rules: separate from neighbors, align with the average heading of neighbors and steer towards neighborhood centroid to keep cohesion. Flight formation control is typically modeled using graphs, where every node is an agent that can exchange information with all or several agents. Velocity and/or position coordination is usually achieved using consensus algorithms.

The work in [QCH13] shares many similarities with our approach. Authors derive a stochastic optimal control formulation of the flocking problem for fixed-wings UAVs. They take a leader-follower strategy, where the leader follows an arbitrary (predefined) policy that is learned offline and define the immediate cost as a function of the distance and heading with respect to the leader. Their method is demonstrated outdoors with 3 fixed-wing UAVs in a distributed sensing task. As in this chapter, they formulate a SOC problem and perform MPC. However, in our case we do not restrict to a leader-follower setup and consider a more general class of SOC problems which can include coordination and cooperation problems.

Planning approaches

Within the planning community, [BFL14] consider search and tracking tasks, similar to one of our scenarios. Their approach is different to ours, they formulate a planning problem that uses used *search patterns* that must be selected and sequenced to maximise the probability of rediscovering the target. [APSTK15] and [CTKL13] consider a different problem: dynamic data acquisition and environmental knowledge optimisation. Both techniques use some form of replanning. While [APSTK15] uses a Markov Random Field framework to represent knowledge about the uncertain map and its quality, [CTKL13] rely on partially-observable MDPs. All these works consider a single UAV scenario and low-level control is either neglected or deferred to a PID or waypoint controller.

Recent progress in path integral control

There has been significant progress in PI control, both theoretically and in applications. Most of existing methods use parametrized policies to overcome the main limitations (see Section 7.3.1). Examples can be found in [TBS10, SS12, GKPN14]. In these methods, the optimal control solution is restricted by the class of parametrized policies and, more importantly, it is computed offline. In [RTV13], authors propose to approximate the transformed cost-to-go function using linear operators in a reproducing kernel Hilbert space. Such an approach requires an analytical form of the PI embedding, which is difficult to obtain in general. In [HDB14], a low-rank tensor representation is used to represent the model dynamics, allowing to scale PI control up to a 12-dimensional system. More recently, the issue of state-dependence of the optimal control has been addressed [TK15], where a parametrized state-dependent feedback controller is derived for the PI control class.

Finally, model predictive PI control has been recently proposed for controlling a nano-quadrotor in indoor settings in an obstacle avoidance task [WRD15]. In contrast to our approach, their method is not hierachical and uses naive sampling, which makes it less sample efficient. Additionally, the control cost term is neglected, which can have important implications in complex tasks involving noise. The approach presented here scales well to several UAVs in outdoor conditions and is illustrated in tasks beyond obstacle avoidance navigation.

7.3 Path integral control for multi-UAV planning

We first briefly review PI control theory. This is followed by a description of the proposed method used to achieve motion planning of multi-agent UAV systems using PI control.

7.3.1 Path integral control

We consider continuous time stochastic control problems, where the dynamics and cost are respectively linear and quadratic in the control input, but arbitrary in the state. More precisely, consider the following stochastic differential equation of the state vector $\mathbf{x} \in \mathbb{R}^n$ under controls $\mathbf{u} \in \mathbb{R}^m$

$$d\mathbf{x} = \mathbf{f}(\mathbf{x})dt + \mathbf{G}(\mathbf{x})(\mathbf{u}dt + d\boldsymbol{\xi}), \tag{7.1}$$

where $\boldsymbol{\xi}$ is m—dimensional Wiener noise with covariance $\Sigma_u \in \mathbb{R}^{m\times m}$ and $\mathbf{f}(\mathbf{x}) \in \mathbb{R}^n$ and $\mathbf{G}(\mathbf{x}) \in \mathbb{R}^{n\times m}$ are arbitrary functions, $\mathbf{f}$ is the drift in the uncontrolled dynamics (including gravity, Coriolis and centripetal forces), and $\mathbf{G}$ describes the effect of the control $\mathbf{u}$ into the state vector $\mathbf{x}$.

A realization $\boldsymbol{\tau} = \mathbf{x}_{0:dt:T}$ of the above equation is called a (random) path. In order to describe a control problem we define the cost that is attributed to a path

(cost-to-go) by

$$S(\boldsymbol{\tau}|\mathbf{x}_0,\mathbf{u}) = r_T(\mathbf{x}_T) + \sum_{t=0:dt:T-dt} \left(r_t(\mathbf{x}_t)dt + \frac{1}{2}\mathbf{u}_t^\top \mathbf{R}\mathbf{u}_t \right) dt, \tag{7.2}$$

where $r_T(\mathbf{x}_T)$ and $r_t(\mathbf{x}_t)$ are arbitrary state cost terms at end and intermediate times, respectively. $\mathbf{R}$ is the control cost matrix. The general stochastic optimal control problem is to minimize the expected cost-to-go w.r.t. the control

$$\mathbf{u}^\star = \arg\min_{\mathbf{u}} \mathbb{E}[S(\boldsymbol{\tau}|\mathbf{x}_0,\mathbf{u})].$$

In general, such a minimization leads to the Hamilton-Jacobi-Bellman (HJB) equation, which is a non-linear, second order partial differential equation. However, under the following relation between the control cost and noise covariance $\Sigma_u = \lambda\mathbf{R}^{-1}$, the resulting equation is *linear* in the exponentially transformed cost-to-go function. The solution is given by the Feynman-Kac Formula, which expresses optimal control in terms of a path integral, which can be interpreted as taking the expectation under the optimal path distribution [Kap05b]

$$p^\star(\boldsymbol{\tau}|\mathbf{x}_0) \propto p(\boldsymbol{\tau}|\mathbf{x}_0,\mathbf{u}) \exp(-S(\boldsymbol{\tau}|\mathbf{x}_0,\mathbf{u})/\lambda), \tag{7.3}$$

$$\left\langle \mathbf{u}_t^\star(\mathbf{x}_0) \right\rangle = \left\langle \mathbf{u}_t + (\boldsymbol{\xi}_{t+dt} - \boldsymbol{\xi}_t)/dt \right\rangle, \tag{7.4}$$

where $p(\boldsymbol{\tau}|\mathbf{x}_0,\mathbf{u})$ denotes the probability of a (sub-optimal) path under equation (7.1) and $\langle\cdot\rangle$ denotes expectation over paths distributed by $p^\star$.

The constraint $\Sigma_u = \lambda\mathbf{R}^{-1}$ forces control and noise to act in the same dimensions, but in an inverse relation. Thus, for fixed λ, the larger the noise, the cheaper the control and vice-versa. Parameter λ act as a temperature: higher values of λ result in optimal solutions that are closer to the uncontrolled process.

Equation (7.4) permits optimal control to be calculated by probabilistic inference methods, e.g., Monte Carlo. An interesting fact is that equations (7.3, 7.4) hold for all controls $\mathbf{u}$. In particular, $\mathbf{u}$ can be chosen to reduce the variance in the Monte Carlo computation of $\left\langle \mathbf{u}_t^\star(\mathbf{x}_0) \right\rangle$ which amounts to importance sampling. This technique can drastically improve the sampling efficiency, which is crucial in high dimensional systems. Despite this improvement, direct application of PI control into real systems is limited because it is not clear how to choose a proper importance sampling distribution. Furthermore, note that equation (7.4) yields the optimal control for all times t averaged over states. The result is therefore an open-loop controller that neglects the state-dependence of the control beyond the initial state.

7.3.2 Multi-UAV planning

The proposed architecture is composed of two main levels. At the most abstract level, the UAV is modeled as a 2D point-mass system that follows double integrator

dynamics. At the low-level, we use a detailed second order model that we learn from real flight data [DNH08]. We use model predictive control combined with importance sampling. There are two main benefits of using the proposed approach: first, since the state is continuously updated, the controller does not suffer from the problems caused by using an open-loop controller. Second, the control policy is not restricted by any parametrization.

The two-level approach permits to transmit control signals from the high-level PI controller to the low-level control system at a relatively low frequencies (we use 15Hz in this work). Consequently, the PI controller has more time available for sampling a large number of trajectories, which is critical to obtain good estimates of the control. The choice of 2D in the presented method is not a fundamental limitation, as long as double-integrator dynamics is used. The control hierarchy introduces additional model mismatch. However, as we show in the results later, this mismatch is not critical for obtaining good performance in real conditions.

Ignoring height, the state vector $\mathbf{x}$ is thus composed of the East-North (EN) positions and EN velocities of each agent $i = 1, \ldots, M$ as $x_i = [p_i, v_i]^\top$ where $p_i, v_i \in \mathbb{R}^2$. Similarly, the control $\mathbf{u}$ consists of EN accelerations $u_i \in \mathbb{R}^2$. Equation (7.1) decouples between the agents and takes the linear form

$$\begin{aligned} dx_i &= Ax_i dt + B(u_i dt + d\xi_i), \\ A &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \qquad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \end{aligned} \tag{7.5}$$

Notice that although the dynamics is decoupled and linear, the state cost $r_t(\mathbf{x}_t)$ in equation (7.2) can be any arbitrary function of all UAVs states. As a result, the optimal control will in general be a non-linear function that couples all the states and thus hard to compute.

Given the current joint optimal action $\mathbf{u}_t^\star$ and velocity $\mathbf{v}_t$, the expected velocity at the next time t' is calculated as $\mathbf{v}_{t'} = \mathbf{v}_t + (t' - t)\mathbf{u}_t^\star$ and passed to the low-level controller. The final algorithm optionally keeps an importance-control sequence $\mathbf{u}_{t:dt:t+H}$ that is incrementally updated. We summarize the high-level controller in Algorithm 4.

The importance-control sequence $\mathbf{u}_{t:dt:t+H}$ is initialized using prior knowledge or with zeros otherwise. Noise is dimension-independent, i.e. $\Sigma_u = \sigma_u^2 \mathrm{Id}$. To measure sampling convergence, we define the *Effective Sample Size* (ESS) as $\mathrm{ESS} := 1/\sum_{k=1}^{N} w_k^2$, which is a quantity between 1 and N. Values of ESS close to one indicate an estimate dominated by just one sample and a poor estimate of the optimal control, whereas an ESS close to N indicates near perfect sampling, which occurs when the importance- equals the optimal-control function.

Algorithm 4 PI control for UAV motion planning

function PICONTROLLER($N, H, dt, r_t(\cdot), \Sigma_u, \mathbf{u}_{t:dt:t+H}$)
  for $k = 1, \ldots, N$ **do**
    Sample paths $\tau_k = \{\mathbf{x}_{t:dt:t+H}\}_k$ with Eq. (7.5)
  end for
  Compute $S_k = S(\tau_k | \mathbf{x}_0, \mathbf{u})$ with Eq. (7.2)
  Store the noise realizations $\{\xi_{t:dt:t+H}\}_k$
  Compute the weights: $w_k = e^{-S_k/\lambda} / \sum_l e^{-S_l/\lambda}$
  for $s = t : dt : t + H$ **do**
    $\mathbf{u}^\star_s = \mathbf{u}_s + \frac{1}{dt} \sum_k w_k \left(\{\xi_{s+dt}\}_k - \{\xi_s\}_k \right)$
  end for
  Return next desired velocity: $\mathbf{v}_{t+dt} = \mathbf{v}_t + \mathbf{u}^\star_t dt$ and $\mathbf{u}^\star_{t:dt:t+H}$ for importance sampling at $t + dt$
end function

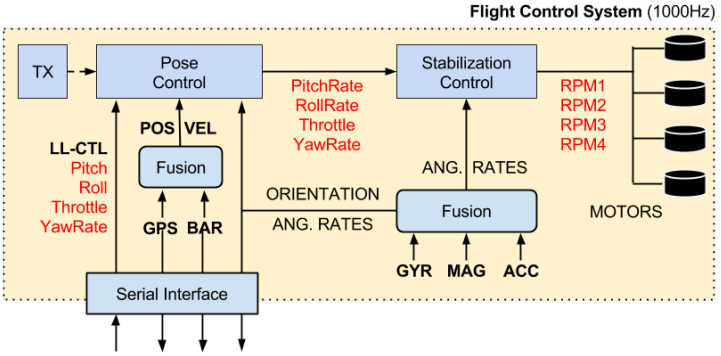

Figure 7.2: The flight control system (FCS) is comprised of two control loops: one for stabilization and the other for pose control. A low-level controller interacts with the FCS over a serial interface to stream measurements and issue control.

7.3.3 Low level control

The target velocity $\mathbf{v} = [v_E \; v_N]^\top$ is passed along with a height $\hat{p}_U$ to a Velocity-Height controller. This controller uses the current state estimate of the real quadrotor $\mathbf{y} = [p_E \; p_N \; p_U \; \phi \; \theta \; \psi \; u \; v \; w \; p \; q \; r]^\top$, where (p_E, p_N, p_U) and (ϕ, θ, ψ) denote navigation-frame position and orientation and $(u, v, w), (p, q, r)$ denote body-frame and angular velocities, respectively. It is composed of four independent PID controllers for roll $\hat{\phi}$, pitch $\hat{\theta}$, throttle $\hat{\gamma}$ and yaw rate $\hat{r}$. that send the commands to the flight control system (FCS) to achieve $\mathbf{v}$.

Figure 7.2 shows the details of the FCS. The control loop runs at 1kHz fusing

triaxial gyroscope, accelerometer and magnetometer measurements. The accelerometer and magnetometer measurements are used to determine a reference global orientation, which is in turn used to track the gyroscope bias. The difference between the desired and actual angular rates are converted to motor speeds using the model in [MKC12].

An outer pose control loop calculates the desired angular rates based on the desired state. Orientation is obtained from the inner control loop, while position and velocity are obtained by fusing GPS navigation fixes with barometric pressure (BAR) based altitude measurements. The radio transmitter (marked TX in the diagram) allows the operator to switch quickly between autonomous and manual control of a platform. There is also an acoustic alarm on the platform itself, which warns the operator when the GPS signal is lost or the battery is getting low. If the battery reaches a critical level or communication with the transmitter is lost, the platform can be configured to land immediately or alternatively, to fly back and land at its take-off point.

7.3.4 Simulator platform

We have developed an open-source framework called `CRATES` [1]. The framework is a implementation of `QRSim` [DN13, SdNJH14] in Gazebo, which uses Robot Operating System (ROS) for high-level control. It permits high-level controllers to be platform-agnostic. It is similar to the Hector Quadrotor project [MSKK12] with a formalized notion of a hardware abstraction layers.

The `CRATES` simulator propagates the quadrotor state forward in time based on a second order model [DNH08]. The equations were learned from real flight data and verified by expert domain knowledge. In addition to platform dynamics, `CRATES` also simulates various noise-perturbed sensors, wind shear and turbulence. Orientation and barometric altitude errors follow zero-mean Ornstein-Uhlenbeck processes, while GPS error is modeled at the pseudo range level using trace data available from the International GPS Service. In accordance with the Military Specification MIL-F-8785C, wind shear is modeled as a function of altitude, while turbulence is modeled as a discrete implementation of the Dryden model. `CRATES` also provides support for generating terrain from satellite images and light detection and ranging (LIDAR) technology, and reporting collisions between platforms and terrain.

7.4 Results

We now analyze the performance of the proposed approach in three different tasks. We first show that, in the presence of control noise, PI control is preferable over

[1] `CRATES` stands for 'Cognitive Robotics Architecture for Tightly-Coupled Experiments and Simulation'. Available at `https://bitbucket.org/vicengomez/crates`

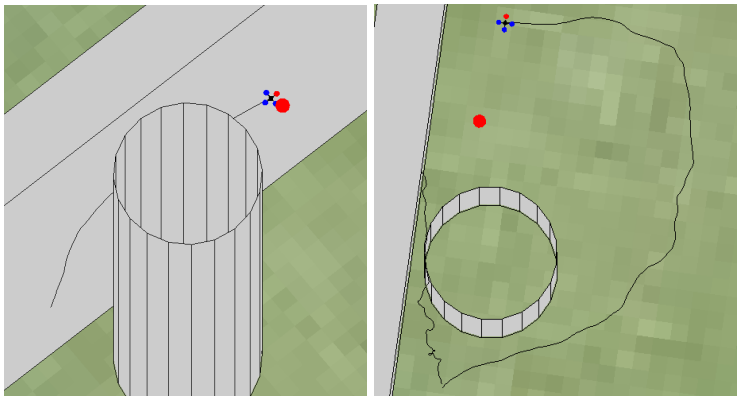

Figure 7.3: Drunken Quadrotor: a red target has to be reached while avoiding obstacles. (Left) the shortest route is the optimal solution in the absence of noise. (Right) with control noise, the optimal solution is to fly around the building.

other approaches. For clarity, this scenario is presented for one agent only. We then consider two tasks involving several units: a flight formation task and a pursuit-evasion task.

We compare the PI control method described in Section 7.3.2 with iterative linear-quadratic Gaussian (iLQG) control [TL05]. iLQG is a state-of-the-art method based on differential dynamic programming, that iteratively computes local linear-quadratic approximations to the finite horizon problem. A key difference between iLQG and PI control is that the linear-quadratic approximation is certainty equivalent. Consequently, iLQG yields a noise independent solution.

7.4.1 Scenario I: Drunken Quadrotor

This scenario is inspired in [Kap05b] and highlights the benefits of SOC in a quadrotor task. The Drunken Quadrotor is a finite horizon task where a quadrotor has to reach a target, while avoiding a building and a wall (figure 7.3). There are two possible routes: a shorter one that passes through a small gap between the wall and the building, and a longer one that goes around the building. Unlike SOC, the deterministic optimal solution does not depend on the noise level and will always take the shorter route. However, with added noise, the risk of collision increases and thus the optimal noisy control is to take the longer route.

This task can be alternatively addressed using other planning methods, such as the one proposed by [OWB13], which allow for specification of user's acceptable levels of risk using chance constraints. Here we focus on comparing deterministic and stochastic optimal control for motion planning. The amount of noise thus determines whether the optimal solution is go through the risky path or the longer

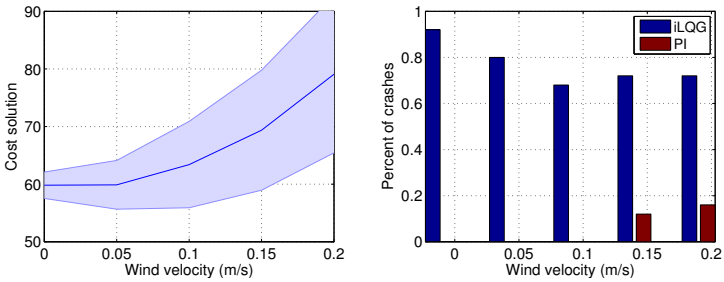

Figure 7.4: Results: **Drunken Quadrotor with wind**: For different wind velocities and fixed control noise $\sigma_u^2 = 0.5$. (Left) cost of the obtained solutions and (Right) percentage of crashes using iLQG and PI.

safer path.

The state cost in this problem consists of hard constraints that assign infinite cost when either the wall or the building is hit. PI control deals with collisions by killing particles that hit the obstacles during Monte Carlo sampling. For iLQG, the local approximations require a twice differentiable cost function. We resolved this issue by adding a smooth obstacle-proximity penalty in the cost function. Although iLQG computes linear feedback, we tried to improve it with a MPC scheme, similar as for PI control. Unfortunately, this leads to numerical instabilities in this task, since the system disturbances tend to move the reference trajectory through a building when moving from one time step to the next. For MPC with PI control we use a receding horizon of three seconds and perform re-planning at a frequency of 15 Hz with $N = 2000$ sample paths. Both methods are initialized with $\mathbf{u}_t = 0, \forall t$. iLQG requires approximately 10^3 iterations to converge with a learning rate of 0.5%.

Figure 7.3 (left) shows an example of real trajectory computed for low control noise level, $\sigma_u^2 = 10^{-3}$. To be able to obtain such a trajectory we deactivate sensor uncertainties in accelerometer, gyroscope, orientation and altimeter. External noise is thus limited to aerodynamic turbulences only. In this case, both iLQG and PI solutions correspond to the shortest path, i.e. go through the gap between the wall and the building. Figure 7.3 (right) illustrates the solutions obtained for larger noise level $\sigma_u^2 = 1$. While the optimal reference trajectory obtained by iLQG does not change, which results in collision once the real noisy controller is executed (left path), the PI control solution avoids the building and takes the longer route (right path). Note that iLQG can find both solutions depending on initialization. However, However, it will always choose the shortest route, regardless of nearby obstacles. Also, note that the PI controlled unit takes a longer route to reach the target. The reason is that the control cost $\mathbf{R}$ is set quite high in order to reach a good ESS. Alternatively, if $\mathbf{R}$ is decreased, the optimal solution could reach the target sooner, but at the cost of a decreased ESS. This trade-off, which is inherent in PI control,

can be resolved by incorporating feedback control in the importance sampling, as presented in [TK15].

We also consider more realistic conditions with noise not limited to act in the control. Figure 7.4 (a,b) shows results in the presence of wind and sensor uncertainty. Panel (a) shows how the wind affects the quality of the solution, resulting in an increase of the variance and the cost for stronger wind. In all our tests, iLQG is not able to bring the quadrotor to the other side. Panel (b) shows the percentage of crashes using both methods. Crashes occur often using iLQG control and only occasionally using PI control. With stronger wind, the iLQG controlled unit does occasionally not even reach the corridor (the unit did not reach the other side but did not crash either). This explains the difference in percentages of Panel (b). We conclude that for multi-modal tasks (tasks where multiple solution trajectories exist), the proposed method is preferable to iLQG.

7.4.2 Scenario II: holding pattern

The second scenario addresses the problem of coordinating agents to hold their position near a point of interest while keeping a safe range of velocities and avoiding crashing into each other. Such a problem arises for instance when multiple aircraft need to land at the same location, and simultaneous landing is not possible. The resulting flight formation has been used frequently in the literature [VVS+14, HBF+08, YDSZ13, FMG+12], but always with prior specification of the trajectories. We show how this formation is obtained as the optimal solution of a SOC problem.

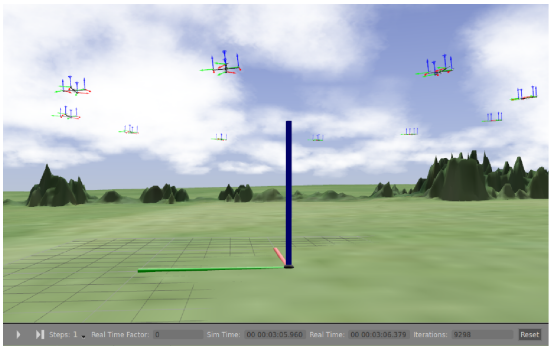

Figure 7.5: Holding pattern in the CRATES simulator. Ten units coordinate their flight in a circular formation. In this example, $N = 10^4$ samples, control noise is $\sigma_u^2 = 0.1$ and horizon $H = 1$ sec. Cost parameters are $v_{\min} = 1, v_{\max} = 3, C_{\text{hit}} = 20$ and $\mathrm{d} = 7$. Environmental noise and sensing uncertainties are modeled using realistic parameter values.

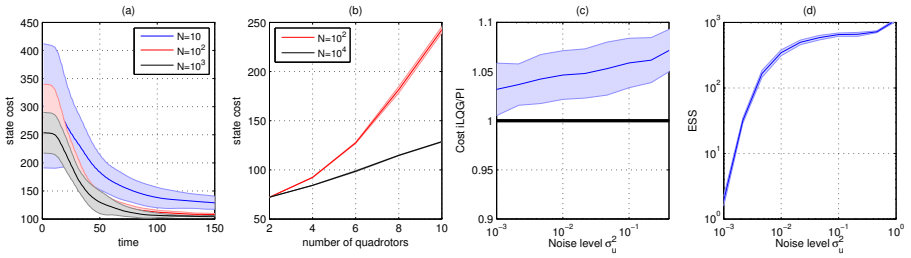

Figure 7.6: Holding pattern: (a) evolution of the state cost for different number of samples $N = 10, 10^2, 10^3$. (b) scaling of the method with the number of agents. For different control noise levels, (c) comparison between iLQG and PI control (ratios > 1 indicate better performance of PI over iLQG) and (d) Effective Sample Sizes. Errors bars correspond to ten different random realizations.

Consider the following state cost (omitting time indexes)

$$r_{\text{HP}}(x) = \sum_{i=1}^{M} \exp(v_i - v_{\max}) + \exp(v_{\min} - v_i) + \exp(\| p_i - \mathrm{d} \|_2) + \sum_{j>i}^{M} C_{\text{hit}} / \| p_i - p_j \|_2 \tag{7.6}$$

where $v_{\max}$ and $v_{\min}$ denote the maximum and minimum velocities, respectively, d denotes penalty for deviation from the origin and C_{hit} is the penalty for collision risk of two agents. $\| \cdot \|_2$ denotes ℓ-2 norm.

The optimal solution for this problem is a circular flying pattern where units fly equidistantly from each other. The value of parameter d determines the radius and the average velocities of the agents are determined from $v_{\min}$ and $v_{\max}$. Since the solution is symmetric with respect to the direction of rotation (clockwise or anti-clockwise), only when the control is executed, a choice is made and the symmetry is broken. Figure 7.5 shows a snapshot of a simulation after the flight formation has been reached for a particular choice of parameter values [2]. Since we use an uninformed initial control trajectory, there is a transient period during which the agents organize to reach the optimal configuration. The coordinated circular pattern is obtained regardless of the initial positions. This behavior is robust and obtained for a large range of parameter values.

Figure 7.6(a) shows immediate costs at different times. Cost always decreases from the starting configuration until the formation is reached. This value depends on several parameters. We report its dependence on the number N of sample

[2] Supplementary video material is available at http://www.mbfys.ru.nl/staff/v.gomez/uav.html

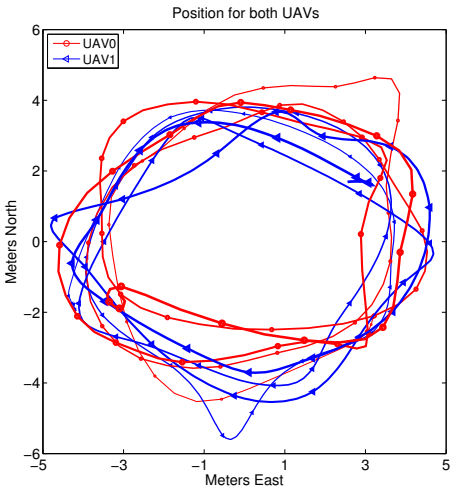

Figure 7.7: Resulting trajectories of a Holding Pattern experiment using two platforms in outdoors conditions.

paths. For large N, the variances are small and the cost attains small values at convergence. Conversely, for small N, there is larger variance and the obtained dynamical configuration is less optimal (typically the distances between the agents are not the same). During the formation of the pattern the controls are more expensive. For this particular task, full convergence of the path integrals is not required, and the formation can be achieved with a very small N.

Figure 7.6(b) illustrates how the method scales as the number of agents increases. We report averages over the mean costs over 20 time-steps after one minute of flight. We varied M while fixing the rest of the parameters (the distance d which was set equal to the number of agents in meters). The small variance of the cost indicates that a stable formation is reached in all the cases. As expected, larger values of N lead to smaller state cost configurations. For more than ten UAVs, the simulator starts to have problems in this task and occasional crashes may occur before the formation is reached due to limited sample sizes. This limitation can be addressed, for example, by using more processing power and parallelization and it is left for future work.

We also compared our approach with iLQG in this scenario. Figure 7.6(c) shows the ratio of cost differences after convergence of both solutions. Both use MPC, with a horizon of 2s and update frequency of 15Hz. Values above 1 indicate that PI control consistently outperforms iLQG in this problem. Before convergence, we also found, as in the previous task, that iLQG resulted in occasional crashes while PI control did not. The Effective Sample Size (ESS) is shown in Figure 7.6(d).

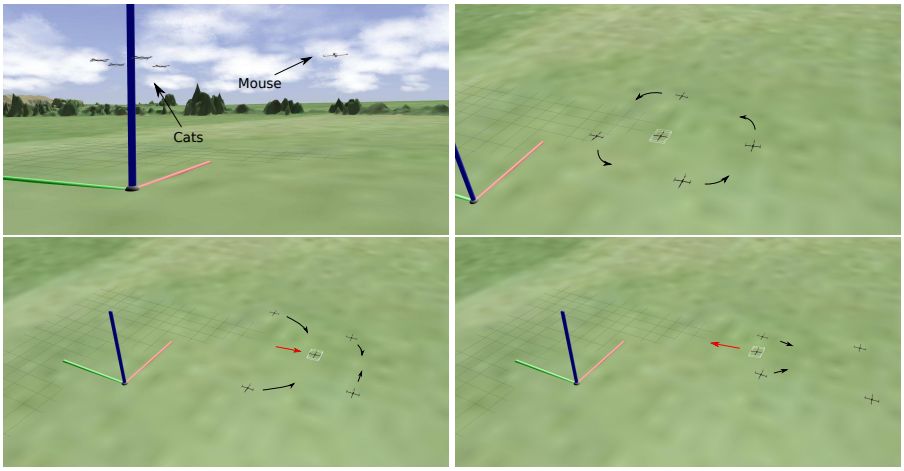

Figure 7.8: Cat and mouse scenario: **(Top-left)** four cats and one mouse. **(Top-right)** for horizon time $H = 2$ seconds, the four cats surround the mouse forever and keep rotation around it. **(Bottom-left)** for horizon time $H = 1$ seconds, the four cats chase the mouse but **(bottom-right)** the mouse manages to escape. With these settings, the multi-agent system alternates between these two dynamical states. Number of sample paths is $N = 10^4$, noise level $\sigma_u^2 = 0.5$. Other parameter values are $\mathrm{d} = 30, v_{\min} = 1, v_{\max} = 4, v_{\min} = 4$ and $v_{\max\ \mathrm{mouse}} = 3$.

We observe that higher control noise levels result in better exploration and thus better controls. We can thus conclude that the proposed methodology is feasible for coordinating a large team of quadrotors.

For this task, we performed experiments with the real platforms. Figure 7.7 shows real trajectories obtained in outdoor conditions (see also the video that accompanies this chapter for an experiment with three platforms). Despite the presence of significant noise, the circular behavior was also obtained. In the real experiments, we used a Core i7 laptop with 8GB RAM as base station, which run its own ROS messaging core and forwarded messages to and from the platforms over a IEEE 802.11 2.4GHz network. For safety reasons, the quadrotors were flown at different altitudes.

7.4.3 Scenario III: cat and mouse

The final scenario that we consider is the cat and mouse scenario. In this task, a team of M quadrotors (the cats) has to catch (get close to) another quadrotor (the mouse). The mouse has autonomous dynamics: it tries to escape the cats by moving at velocity inversely proportional to the distance to the cats. More precisely, let

p_{mouse} denote the 2D position of the mouse, the velocity command for the mouse is computed (omitting time indexes) as

$$v_{\text{mouse}} = v_{\underset{\text{mouse}}{\max}} \frac{v}{\| v \|_2}, \qquad \text{where } v = \sum_{i=1}^{M} \frac{p_i - p_{\text{mouse}}}{\| p_i - p_{\text{mouse}} \|^2}.$$

The parameter $v_{\underset{\text{mouse}}{\max}}$ determines the maximum velocity of the mouse. As state cost function we use equation (7.6) with an additional penalty term that depends on the sum of the distances to the mouse

$$r_{\text{CM}}(x) = r_{\text{HP}}(x) + \sum_{i=1}^{M} \| p_i - p_{\text{mouse}} \|_2 .$$

This scenario leads to several interesting dynamical states. For example, for a sufficiently large value of M, the mouse always gets caught (if its initial position is not close to the boundary, determined by d). The optimal control for the cats consists in surrounding the mouse to prevent collision. Once the mouse is surrounded, the cats keep rotating around it, as in the previous scenario, but with the origin replaced by the mouse position. The additional video shows examples of other complex behaviors obtained for different parameter settings. Figure 7.8 (top-right) illustrates this behavior.

The types of solution we observe are different for other parameter values. For example, for $M = 2$ or a small time horizon, e.g. $H = 1$, the dynamical state in which the cats rotate around the mouse is not stable, and the mouse escapes. This is displayed in Figure 7.8 (bottom panels) and better illustrated in the video provided as supplementary material. We emphasize that these different behaviors are observed for large uncertainty in the form of sensor noise and wind.

7.5 Conclusions

In this chapter we presented a centralized, real-time stochastic optimal control algorithm for coordinating the actions of multiple autonomous vehicles in order to minimize a global cost function. The high-level control task is expressed as a path integral control problem that can be solved using efficient sampling methods and real-time control is possible via the use of re-planning and model predictive control. To the best of our knowledge, this is the first real-time implementation of path integral control on an actual multi-agent system.

We have shown in a simple scenario (Drunken Quadrotor) that the proposed methodology is more convenient than other approaches such as deterministic control or iLQG for planning trajectories. In more complex scenarios such as the Holding Pattern and the Cat and Mouse, the proposed methodology is also preferable and allows for real-time control. We observe multiple and complex group behavior

emerging from the specified cost function. Our experimental framework CRATES has been a key development that permitted a smooth transition from the theory to the real quadrotor platforms, with literally no modification of the underlying control code. This gives evidence that the model mismatch caused by the use of a control hierarchy is not critical in normal outdoor conditions. Our current research is addressing the following aspects:

Large scale parallel sampling— the presented method can be easily parallelized, for instance, using graphics processing units, as in [WRD15]. Although the tasks considered in this work did not required more than 10^4 samples, we expect that this improvement will significantly increase the number of application domains and system size.

Distributed control— we are exploring different distributed formulations that take better profit of the factorized representation of the state cost. Note that the costs functions considered in this work only require pairwise couplings of the agents (to prevent collisions). However, full observability of the joint space is still required, which is not available in a fully distributed approach.

Bibliography

[AM12] R. Anderson and D. Milutinović. A stochastic optimal enhancement of feedback control for unicycle formations. In *11th International Symposium on Distributed Autonomous Robotic Systems (DARS)*, 8–11 November 2012 2012.

[And88] Donald W. K. Andrews. Laws of large numbers for dependent non-identically distributed random variables. *Econometric Theory*, 4(3):458–467, 1988.

[APSTK15] Alexandre Albore, Nathalie Peyrard, Régis Sabbadin, and Florent Teichteil Königsbuch. An online replanning approach for crop fields mapping with autonomous uavs. In *International Conference on Automated Planning and Scheduling*, 2015.

[ASD12] F. Augugliaro, A.P. Schoellig, and R. D'Andrea. Generation of collision-free trajectories for a quadrocopter fleet: A sequential convex programming approach. In *Intelligent Robots and Systems (IROS)*, pages 1917–1922, 2012.

[BFL14] Sara Bernardini, Maria Fox, and Derek Long. Planning the behaviour of low-cost quadcopters for surveillance missions. In *International Conference on Automated Planning and Scheduling*, 2014.

[BK14] J. Bierkens and H.J. Kappen. Explicit solution of relative entropy weighted control. *Systems & Control Letters*, 72(0):36 – 43, 2014.

[BKMR05] Pieter-Tjerk de Boer, Dirk P. Kroese, Shie Mannor, and Reuven Y. Rubinstein. A tutorial on the cross-entropy method. *Annals of Operations Research*, 134(1):19–67, 2005.

[BSK11] Axel Bürkle, Florian Segor, and Matthias Kollmann. Towards autonomous micro UAV swarms. *J. Intell. Robot. Syst.*, 61(1-4):339–353, 2011.

[BWK08a] B. van den Broek, W. Wiegerinck, and H.J. Kappen. Graphical model inference in optimal control of stochastic multi-agent systems. *J. Artif. Intell. Res.*, 32(1):95–122, may 2008.

[BWK08b] Bart van den Broek, Wim Wiegerinck, and H. J. Kappen. Graphical model inference in optimal control of stochastic multi-agent systems. *J. Artif. Intell. Res.*, 32:95–122, 2008.

[CGMR04] O. Cappe, A. Guillin, J. M. Marin, and C. P. Robert. Population monte carlo. *Journal of Computational and Graphical Statistics*, 13(4):907–929, 2004.

[CMMR12] Jean-Marie Cornuet, Jean-Michel Marin, Antonietta Mira, and Christian P. Robert. Adaptive multiple importance sampling. *Scandinavian Journal of Statistics*, 39(4):798–812, 2012.

[CTKL13] Caroline Ponzoni Carvalho Chanel, Florent Teichteil-Königsbuch, and Charles Lesire. Multi-target detection and recognition by uavs using online pomdps. In *AAAI*, 2013.

[DN13] Renzo De Nardi. The QRSim Quadrotors Simulator. Technical Report RN/13/08, Department of Computer Science, University College London, March 2013.

[DNH08] R. De Nardi and O.E. Holland. Coevolutionary modelling of a miniature rotorcraft. In *10th International Conference on Intelligent Autonomous Systems (IAS10)*, pages 364 – 373, 2008.

[Doo57] J.L. Doob. Conditional brownian motion and the boundary limits of harmonic functions. *Bulletin de la Société Mathématique de France*, 85:431–458, 1957.

[EMLB15a] V. Elvira, L. Martino, D. Luengo, and M.F. Bugallo. Efficient multiple importance sampling estimators. *Signal Processing Letters, IEEE*, 22(10):1757–1761, Oct 2015.

[EMLB15b] Víctor Elvira, Luca Martino, David Luengo, and Mónica F. Bugallo. Generalized multiple importance sampling. *arXiv preprint arXiv:1511.03095*, 2015.

[Fle82] Wendell H. Fleming. *Logarithmic transformations and stochastic control*. Springer Berlin Heidelberg, Berlin, Heidelberg, 1982.

[FM95] Wendell H Fleming and William M McEneaney. Risk-sensitive control on an infinite time horizon. *SIAM Journal on Control and Optimization*, 33(6):1881–1915, 1995.

[FMG+12] Antonio Franchi, Carlo Masone, Volker Grabe, Markus Ryll, Heinrich H Bülthoff, and Paolo Robuffo Giordano. Modeling and control of UAV bearing-formations with bilateral high-level steering. *Int. J. Robot. Res.*, page 0278364912462493, 2012.

[FR75] Wendell Fleming and Raymond W. Rishel. *Deterministic and stochastic optimal control*. Applications of mathematics. Springer, New York, Berlin, Heidelberg, 1975.

[FS06] W.H. Fleming and H.M. Soner. *Controlled Markov Processes and Viscosity Solutions*. Stochastic Modelling and Applied Probability. Springer, 2006.

[GH99] P. Glasserman and P. Heidelberger. Asymptotically optimal importance sampling and stratification for pricing path-dependent options. *Math. Finance*, 9:117–152, 1999.

[GKPN14] Vicenç Gómez, H. J. Kappen, J. Peters, and G. Neumann. Policy search for path integral control. In *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD)*, volume 8724, pages 482–497, 2014.

[GL12] Josep Guerrero and Rogelio Lozano. *Flight Formation Control*. John Wiley & Sons, 2012.

[GTS+15] Vicenç Gómez, Sep Thijssen, Andrew Symington, Stephen Hailes, and H. J. Kappen. Real-time stochastic optimal control for multi-agent quadrotor systems. *arXiv preprint arXiv:1502.04548*, 2015.

[HBF+08] J.P. How, B. Bethke, A. Frank, D. Dale, and J. Vian. Real-time indoor autonomous vehicle test environment. *IEEE Contr. Syst. Mag.*, 28(2):51–64, 2008.

[HDB14] M.B. Horowitz, A. Damle, and J.W. Burdick. Linear hamilton jacobi bellman equations in high dimensions. In *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, pages 5880–5887, Dec 2014.

[HHWT11] Gabriel M. Hoffmann, Haomiao Huang, Steven L. Waslander, and Claire J. Tomlin. Precision flight control for a multi-vehicle quadrotor helicopter testbed. *Control. Eng. Pract.*, 19(9):1023 – 1036, 2011.

[HLV+11] Sabine Hauert, Severin Leven, Maja Varga, Fabio Ruini, A. Cangelosi, J.-C. Zufferey, and D. Floreano. Reynolds flocking in reality with fixed-wing robots: Communication range vs. maximum turning rate. In *Intelligent Robots and Systems (IROS)*, pages 5015–5020, 2011.

[Jon95] R. M. de Jong. Laws of large numbers for dependent heterogeneous processes. *Econometric Theory*, 11(2):347–358, 1995.

[Kap05a] H.J. Kappen. Linear theory for control of nonlinear stochastic systems. *Phys. Rev. Lett.*, 95(20):200201, 2005.

[Kap05b] H.J. Kappen. Path integrals and symmetry breaking for optimal control theory. *J. Stat. Mech.: Theory Exp.*, 2005(11):P11011, 2005.

[Ken12] Farid Kendoul. Survey of advances in guidance, navigation, and control of unmanned rotorcraft systems. *J. Field Robot.*, 29(2):315–378, 2012.

[KGO12] H. J. Kappen, V. Gómez, and M. Opper. Optimal control as a graphical model inference problem. *Mach. Learn.*, 87:159–182, 2012.

[KMPK13] Alex Kushleyev, Daniel Mellinger, Caitlin Powers, and Vijay Kumar. Towards a swarm of agile micro quadrotors. *Auton. Robot.*, 35(4):287–300, 2013.

[KP92] Peter E. Kloeden and Eckhard Platen. *Numerical Solution of Stochastic Differential Equations*. Springer-Verlag Berlin Heidelberg, 1 edition, 1992.

[KR16] H. J. Kappen and H. C. Ruiz. Adaptive importance sampling for control and inference. *Journal of Statistical Physics*, 162(5):1244–1266, 2016.

[KS91] Ioannis Karatzas and Steven E. Shreve. *Brownian Motion and Stochastic Calculus (Graduate Texts in Mathematics)*. Springer, 2nd edition, August 1991.

[KUD13] K. Kinjo, E. Uchibe, and K. Doya. Evaluation of linearly solvable markov decision process with dynamic model learning in a mobile robot navigation task. *Front. Neurorob.*, 7(7), 2013.

[Leh13] Joseph Lehec. Representation formula for the entropy and functional inequalities. *Annales de l'I.H.P. Probabilités et statistiques*, 49(3):885–899, 2013.

[Liu08] Jun S. Liu. *Monte Carlo Strategies in Scientific Computing*. Springer, corrected edition, January 2008.

[MD98] Boué Michelle and Paul Dupuis. A variational representation for certain functionals of brownian motion. *The Annals of Probability*, 26(4):1641–1659, 1998.

[MELC15] L. Martino, V. Elvira, D. Luengo, and J. Corander. Layered adaptive importance sampling. *arXiv preprint arXiv:1505.04732*, 2015.

[MKC12] R Mahony, V Kumar, and P Corke. Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor. *IEEE Robotics & Automation Magazine*, pages 20–32, September 2012.

[MPS12] Jean-Michel Marin, Pierre Pudlo, and Mohammed Sedki. Consistency of the adaptive multiple importance sampling. *arXiv preprint arXiv:1211.2548*, 2012.

[MSKK12] Johannes Meyer, Alexander Sendobry, Stefan Kohlbrecher, and Uwe Klingauf. Comprehensive Simulation of Quadrotor UAVs Using ROS and Gazebo. *Lecture Notes in Computer Science*, 7628:400–411, 2012.

[Nie97] Ole A. Nielsen. *An introduction to integration and measure theory*. Wiley, 1997.

[OB92] Man-Suk Oh and James O. Berger. Adaptive importance sampling in monte carlo integration. *Journal of Statistical Computation and Simulation*, 41(3-4):143–168, 1992.

[Øks85] Bernt Øksendal. *Stochastic Differential Equations : An Introduction with Applications*. Springer, Berlin Heidelberg, 1985.

[OWB13] Masahiro Ono, Brian C. Williams, and Lars Blackmore. Probabilistic planning for continuous dynamic systems under bounded risk. *J. Artif. Intell. Res. (JAIR)*, 46:511–577, 2013.

[OZ00] Art Owen and Yi Zhou. Safe and effective importance sampling. *Journal of the American Statistical Association*, 95(449):135–143, 2000.

[QCH13] S.A.P. Quintero, G.E. Collins, and J.P. Hespanha. Flocking with fixed-wing uavs for distributed sensing: A stochastic optimal control approach. In *American Control Conference (ACC)*, pages 2025–2031, 2013.

[Rey87] Craig W. Reynolds. Flocks, herds and schools: A distributed behavioral model. *SIGGRAPH Comput. Graph.*, 21(4):25–34, 1987.

[RTM+12a] E. Rombokas, E. Theodorou, M. Malhotra, E. Todorov, and Y. Matsuoka. Tendon-driven control of biomechanical and robotic systems: A path integral reinforcement learning approach. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 208–214, 2012.

[RTM+12b] E. Rombokas, E. Theodorou, M. Malhotra, E. Todorov, and Y. Matsuoka. Tendon-driven control of biomechanical and robotic systems: A path integral reinforcement learning approach. In *International Conference on Robotics and Automation (ICRA)*, pages 208–214, 2012.

[RTV13] K. Rawlik, M. Toussaint, and S. Vijayakumar. Path integral control by reproducing kernel Hilbert space embedding. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, pages 1628–1634. AAAI Press, 2013.

[SdNJH14] A. C. Symington, R. de Nardi, S. J. Julier, and S. Hailes. Simulating quadrotor UAVs in outdoor scenarios. In *Intelligent Robots and Systems (IROS)*, 2014.

[SKS03] David H Shim, H Jin Kim, and Shankar Sastry. Decentralized nonlinear model predictive control of multiple flying robots. In *IEEE conference on Decision and control (CDC)*, volume 4, pages 3621–3626, 2003.

[SM11] N. Sugimoto and J. Morimoto. Phase-dependent trajectory optimization for cpg-based biped walking using path integral reinforcement learning. In *International Conference on Humanoid Robots, IEEE-RAS*, pages 255–260, 26–28 October 2011 2011.

[SS12] F. Stulp and O. Sigaud. Path integral policy improvement with covariance matrix adaptation. In *International Conference Machine Learning*, 2012.

[Ste94] R.F. Stengel. *Optimal Control and Estimation*. Dover, New York, 1994.

[TBS10] E. Theodorou, J. Buchli, and S. Schaal. A generalized path integral control approach to reinforcement learning. *Journal of Machine Learning Research*, 11:3137–3181, 2010.

[TK15] Sep Thijssen and H. J. Kappen. Path integral control and state-dependent feedback. *Phys. Rev. E*, 91:032104, Mar 2015.

[TK16] Sep Thijssen and H. J. Kappen. Consistent adaptive multiple importance sampling and controlled diffusions. *arXiv preprint arXiv:??.??*, 2016.

[TL05] Emmanuel Todorov and Weiwei Li. A generalized iterative LQG method for locally-optimal feedback control of constrained nonlinear stochastic systems. In *American Control Conference, 2005. Proceedings of the 2005*, pages 300–306 vol. 1, June 2005.

[TMK12] M. Turpin, N. Michael, and V. Kumar. Decentralized formation control with variable shapes for aerial robots. In *International Conference on Robotics and Automation (ICRA)*, pages 23–30, 2012.

[TT12] E. Theodorou and E. Todorov. Relative entropy and free energy dualities: connections to path integral and kl control. In *Decision and*

Control (CDC), 2012 IEEE 51st Annual Conference on, pages 1466–1473, 10–13 December 2012 2012.

[VG95] Eric Veach and Leonidas J. Guibas. Optimally combining sampling techniques for monte carlo rendering. In *Proceedings of the 22Nd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '95, pages 419–428, New York, NY, USA, 1995.

[VVS+14] Gábor Vásárhelyi, Csaba Virágh, Gergő Somorjai, Norbert Tarcai, Tamás Szörényi, Tamás Nepusz, and Tamás Vicsek. Outdoor flocking and formation flight with autonomous aerial robots. In *Intelligent Robots and Systems (IROS)*, 2014.

[WRD15] Grady Williams, Eric Rombokas, and Tom Daniel. Gpu based path integral control with learned dynamics. *CoRR*, abs/1503.00330, 2015.

[YDSZ13] Bocheng Yu, Xiwang Dong, Zongying Shi, and Yisheng Zhong. Formation control for quadrotor swarm systems: Algorithms and experiments. In *Chinese Control Conference (CCC)*, pages 7099–7104, 2013.

Index

$()^\top$, transpose, 7
$J^\star$, optimal expected cost, 6, 11, 28
$P^\star$, optimal controlled measure, 28
$\ll$, absolute continuity, 28
$\mathscr{A}$, generator, 7
$\mathscr{E}(\cdot)$, Dol'eans-Dade exponential, 30
$\mathrm{D}_{\mathrm{KL}}(\cdot\|\cdot)$, KL divergence, 28
∂_{xx}, Hessian, 7
ψ, value function, 14
$u^\star$, optimal control, 6, 11, 31
`QRSim`, 65
`CRATES`, 65

absolute continuity, 28, 35
adapted, 6
admissible control process, 30
AIS, 36
alternative form, 12
AMIS, 36

balance heuristic, 3, 36
Brownian motion, 6

control process, 30
controlled (probability) measure, 28
cost, 10, 28
cost to go, 6, 10

deterministic multiple mixture, 3, 36
discarding time, 40
Discarding-re-weighting, 40
Doléans-Dade exponential, 30, 31
dynamic programming, 6, 8
dynamics, 10

Dynkin's formula, 8

end cost, 6
ESS, 41, 43
expected cost to go, 6

feedback, 1
Feynman-Kac, 17

generator, 7
Girsanov Theorem, 30

HJB equation, 8, 15

immediate cost, 6
importance sampling, 35
importance weight, 35
infinitesimal generator, 7
IS, 33
Itô's Lemma, 7

Kullback-Leibler divergence, 28

Law of Large Numbers, 47

Main Theorem, 18
Markov control, 6
Martingale, 30, 47
Martingale Difference Sequence, 47
MIS, 33, 35
Mixingale Sequence, 47
Multiple Importance Sampling, 33

Novikov Condition, 30, 48

path integral, 12, 17–19

Path Integral Adaptation, 51, 52
path integral control algorithm, 51
proposal, 33

Radon-Nikodym derivative, 28
Radon-Nikodym Theorem, 28
re-weighting, 36, 51
relative entropy, 28

SDE, 5, 6
SOC, 3, 57
state cost, 28

target measure, 41
total cost, 28
total expected cost, 28

uncontrolled (probability) measure, 28
uncontrolled path, 30

verification theorem, 6

Summary

The objective of stochastic control theory is to find an external input (the control) in order to move a noisy system into a desired state. There are many applications of control theory in everyday life and it appears naturally in various areas of science. In robotics, the problem may be to plan a sequence of actions that yield a motor behavior such as walking or grasping an object. In finance, the problem may be to devise a sequence of buy and sell actions to optimize a portfolio of assets, or to determine the optimal option price.

In certain cases, the optimal way to control the system can be described in a mathematically convenient manner, so that the control can be approximated by the Monte Carlo method, which relies on numerous random samples. The control takes place over a certain time interval so that the random samples look like paths. The involved computations for the optimal control are in a sense numerical approximations of integrals. Hence the name Path Integral Control.

The goal in stochastic optimal control is to control a noisy system in such a way that a given cost function is minimized in expectation. In general, this problem can be solved by applying the principle of dynamic programming. In the continuous time setting this yields a differential equation, known as the Hamilton-Jacobi-Bellman (HJB) equation, that describes the optimal expected cost function. Although, from a mathematical point of view, this can be interpreted as a solution, the HJB equation cannot be applied directly to compute a control input. In general, the HJB differential equation does not have an analytical solution and numerical solutions are intractable due to the curse of dimensionality.

In order to proceed we consider in this thesis a class of control problems in which the HJB can be linearized. For these problems the solution can be expressed with the Feynman-Kac formula as an expectation over a stochastic process, i.e. a path integral. Although this approach has been applied successfully before, we will cover in this thesis some key aspects about path integral control that were previously unknown or unclear.

In Chapter 3 we give a description of path integral control in terms of stochastic calculus. An important new result is a generalization of the Main Theorem of path integral control, which can be used to construct parametrized state dependent

feedback controllers. The state dependent feedback is key, because the optimal control is always a function of the state in a system with noise. Another new theorem gives the connection between the control that is used in the Monte Carlo simulation and the efficiency of the involved computation. The efficiency is measured in terms of the variance of the path costs and is connected to the effective sample size of the simulation. It turns out that the most efficient computations, with zero variance in the samples, are achieved when using the optimal control. The positive and self-reinforcing effect is that better controls yield better simulations, and better simulations yield more efficient computations for the control.

In Chapter 4 we make the connection between path integral control and the more general Kullback-Leibler (KL) control. This has led to some new insights about the parametrized control function that we compute; the parametrization is only an approximation of the optimal control, having the same outcome in expectation. From the connection with KL control, however, we obtain that this approximation is also an optimal control for a connected control problem in which the KL control cost term is reversed. Furthermore, we give a more elegant proof of the Main Theorem of path integral control that is based on Girsanov's Theorem.

In Chapter 5 we show that finding the optimal control and optimizing the sampling procedure are mathematically the same problem. Furthermore, we investigate how simulations that are created with different controls can be combined in one big simulation, and we give conditions that ensure that the resulting estimate is consistent. We apply this by describing a method of low computational complexity that computes an estimate that is both consistent and efficient in combining the samples. The corresponding algorithms are described in Chapter 6.

In the last chapter we show that path integral control can be used to control a team of quadcopters in a flight pattern that requires cooperation. This has been shown in an outdoor demonstration where the quadcopters had to deal with wind, turbulence and imperfect GPS signals.

Samenvatting

Het doel in stochastische regeltechniek is om met een extern signaal (de aansturing) een systeem dat onderhevig is aan ruis in een gewenste staat te krijgen. Er zijn veel toepassingen van stochastische regeltechniek in zowel het alledaagse leven als in de wetenschap; het idee om dingen zo goed mogelijk aan te sturen is heel natuurlijk. In robotica is er bijvoorbeeld de uitdaging om met het aansturen van sterke elektromotoren in een robotarm een fragiel object op te pakken. In de financiële wereld wil men een strategie voor koop- en verkoop-acties om een portfolio te optimaliseren, of men wil de prijs van een aandeel of optie bepalen.

De optimale aansturing kan in bepaalde gevallen beschreven worden op een wiskundig handige manier, zodanig dat de aansturing berekend kan worden met de Monte-Carlosimulatie methode. Deze methode maakt gebruik van vele willekeurige simulaties. Het aansturen vindt plaats gedurende een bepaalde tijd en daarom zal een simulatie eruit zien als een pad. De berekeningen voor de optimale aansturing zijn in feite slimme benaderingen van integralen. Vandaar de titel van het proefschrift: Path Integral Control (Padintegraal Aansturing).

De uitdaging in de stochastische optimale regeltechniek is om een systeem met ruis zodanig aan te sturen dat een gegeven kostenfunctie naar verwachting geminimaliseerd wordt. In het algemeen kan dit probleem opgelost worden door gebruik te maken van het principe van dynamisch programmeren. Dit levert een differentiaal vergelijking, de Hamilton-Jacobi-Bellman (HJB) vergelijking, die de optimale verwachte koste van het probleem beschrijven. Hiermee heb je weliswaar een wiskundige oplossing in handen, maar deze is niet direct toe te passen. Het is namelijk in het algemeen erg moeilijk om differentiaal vergelijkingen op te lossen. Analytische oplossingen zijn er vaak niet en numerieke oplossingen zijn moeilijk te berekenen vanwege de vloek van de dimensionaliteit.

Om het bovenstaande probleem te doorbreken, beschouwen we in dit proefschrift een bepaalde klasse van problemen waarbij de HJB vergelijking gelineariseerd kan worden. In dat geval kan de oplossing met de Feynman-Kac formule beschreven worden als een verwachtingswaarde over een stochastisch proces; oftewel een padintegraal. Alhoewel deze padintegraal aanpak al met succes is toegepast, behandelen we in dit proefschrift een aantal belangrijke aspecten van de

theorie die voorheen onduidelijk of onbekend waren.

In Hoofdstuk 3 geven we een beschrijving van padintegraal aansturing met behulp van stochastische calculus. Een belangrijk nieuw resultaat is een generalisatie van de hoofdstelling van padintegraal aansturing, die het mogelijk maakt om de verwachte optimale feedbackloop efficiënt uit te rekenen met behulp van geparametriseerde functies van de toestand. Dit is van cruciaal belang omdat in systemen met ruis de optimale aansturing altijd een feedback signaal gebruikt en dus een functie van de toestand is. Een ander nieuw resultaat geeft een verband tussen de gebruikte aansturing in de Monte-Carlo methode en de efficiëntie van de simulaties. De efficiëntie wordt gemeten met de spreiding van de padkosten en is gerelateerd aan de effectieve simulatiegrootte. De conclusie is dat de optimale aansturing de meest efficiënte berekeningen levert met een variantie van nul. Dit geeft het volgende positieve effect: betere aansturing levert betere simulaties en betere simulaties leveren betere berekeningen voor de aansturing.

In Hoofdstuk 4 maken we de koppeling tussen padintegraal aansturing en het algemenere Kullback-Leibler (KL) aansturingsprobleem. Dit heeft geleid tot meer inzichten over de geparametriseerde aansturingsfunctie die we uitrekenen; deze is slechts een benadering van de optimale aansturing die in verwachting dezelfde uitkomst geeft. Uit de nieuwe theorie volgt echter ook dat deze benadering optimaal is in een omgekeerd KL aansturingsprobleem. Verder geven we een eleganter bewijs van de hoofdstelling van padintergaal aansturing door gebruik te maken van de Stelling van Girsanov.

In Hoofdstuk 5 laten we zien dat het aansturingsprobleem en het bijbehorende simulatie probleem, wiskundig gezien dezelfde problemen zijn. Verder onderzoeken we hoe simulaties behorende bij verschillende aansturingen samengevoegd kunnen worden tot één grote simulatie en we geven voorwaarden waaronder deze samengevoegde simulatie consistent is. Als toepassing geven we een manier om de simulaties computationeel snel samen te voegen die zowel consistent is als een efficiënte samengevoegde simulatie levert. De bijbehorende algoritmen worden beschreven in Hoofdstuk 6.

In het laatste hoofdstuk laten we zien dat het mogelijk is om met padintegraal aansturing een team van quadcopters een vluchtpatroon uit te laten voeren die samenwerking vereist. Dit is ook uitgevoerd in een buitenexperiment waar de quadcopters onder andere onderhevig waren aan ruis van de buitenwind, turbulentie en een imperfect GPS signaal.

Curriculum Vitae

Sep Thijssen was born on July 27, 1983, in Groningen, the Netherlands. He grew up in Apeldoorn, where he finished high school at the Atheneum of the Koninklijke Scholengemeenschap Apeldoorn in 2001. After this period, he moved to Nijmegen to study mathematics at the Radboud University in Nijmegen. As a mathematician he showed interest in algebraic subjects such as number theory and computer algebra. In 2010 he obtained a MSc. degree in Mathematics, cum laude. That same year he became a PhD student at the department of biophysics at the Radboud University under the supervision of H.J. Kappen. Although quite different from what he had been working on during his master, he discovered new interests in stochastic optimal control theory. Currently he is working as a mathematics teacher at the HAN University of Applied Sciences.

Donders Graduate School for Cognitive Neuroscience

For a successful research Institute, it is vital to train the next generation of young scientists. To achieve this goal, the Donders Institute for Brain, Cognition and Behaviour established the Donders Graduate School for Cognitive Neuroscience (DGCN), which was officially recognised as a national graduate school in 2009. The Graduate School covers training at both Master's and PhD level and provides an excellent educational context fully aligned with the research programme of the Donders Institute.

The school successfully attracts highly talented national and international students in biology, physics, psycholinguistics, psychology, behavioral science, medicine and related disciplines. Selective admission and assessment centers guarantee the enrolment of the best and most motivated students.

The DGCN tracks the career of PhD graduates carefully. More than 50% of PhD alumni show a continuation in academia with postdoc positions at top institutes worldwide, e.g. Stanford University, University of Oxford, University of Cambridge, UCL London, MPI Leipzig, Hanyang University in South Korea, NTNU Norway, University of Illinois, North Western University, Northeastern University in Boston, ETH ZÃijrich, University of Vienna etc.. Positions outside academia spread among the following sectors: specialists in a medical environment, mainly in genetics, geriatrics, psychiatry and neurology. Specialists in a psychological environment, e.g. as specialist in neuropsychology, psychological diagnostics or therapy. Positions in higher education as coordinators or lecturers. A smaller percentage enters business as research consultants, analysts or head of research and development. Fewer graduates stay in a research environment as lab coordinators, technical support or policy advisors. Upcoming possibilities are positions in the IT sector and management position in pharmaceutical industry. In general, the PhDs graduates almost invariably continue with high-quality positions that play an important role in our knowledge economy.

For more information on the DGCN as well as past and upcoming defenses please visit:

`http://www.ru.nl/donders/graduate-school/donders-graduate/`