THE UNIVERSITY *of* EDINBURGH

On Probabilistic Inference Approaches to Stochastic Optimal Control

Konrad C. Rawlik

Doctor of Philosophy

Institute of Perception, Action and Behaviour
School of Informatics
University of Edinburgh

2013

Abstract

While stochastic optimal control, together with associate formulations like Reinforcement Learning, provides a formal approach to, amongst other, motor control, it remains computationally challenging for most practical problems. This thesis is concerned with the study of relations between stochastic optimal control and probabilistic inference. Such dualities – exemplified by the classical Kalman Duality between the Linear-Quadratic-Gaussian control problem and the filtering problem in Linear-Gaussian dynamical systems – make it possible to exploit advances made within the separate fields. In this context, the emphasis in this work lies with utilisation of approximate inference methods for the control problem.

Rather then concentrating on special cases which yield analytical inference problems, we propose a novel interpretation of stochastic optimal control in the general case in terms of minimisation of certain Kullback-Leibler divergences. Although these minimisations remain analytically intractable, we show that natural relaxations of the exact dual lead to new practical approaches. We introduce two particular general iterative methods Ψ-*Learning*, which has global convergence guarantees and provides a unifying perspective on several previously proposed algorithms, and *Posterior Policy Iteration*, which allows direct application of inference methods. From these, practical algorithms for Reinforcement Learning, based on a Monte Carlo approximation to Ψ-Learning, and model based stochastic optimal control, using a variational approximation of posterior policy iteration, are derived.

In order to overcome the inherent limitations of parametric variational approximations, we furthermore introduce a new approach for none parametric approximate stochastic optimal control based on a reproducing kernel Hilbert space embedding of the control problem.

Finally, we address the general problem of temporal optimisation, i.e., joint optimisation of controls and temporal aspects, e.g., duration, of the task. Specifically, we introduce a formulation of temporal optimisation based on a generalised form of the finite horizon problem. Importantly, we show that the generalised problem has a dual finite horizon problem of the standard form, thus bringing temporal optimisation within the reach of most commonly used algorithms.

Throughout, problems from the area of motor control of robotic systems are used to evaluate the proposed methods and demonstrate their practical utility.

Acknowledgements

There are a number of people whom I wish to thank for their support, help and advice when conducting this research and writing this thesis.

First, I would like to thank my supervisor, *Sethu Vijayakumar*, for providing me with the opportunity to conduct this research, the encouragement to pursue my ideas and guidance in completing this thesis.

Second, I would particularly like to thank *Marc Toussaint*, for providing the opportunity to visit his group at the TU Berlin. The stimulating discussions during my time there have played a significant role in clarifying my thoughts on the subject matters in this thesis and I found his advice invaluable.

Third, I would like to thank my fellow students in the SLMC group, the DTC and ANC. Their help with uncountable minor daily struggles of academic work and life will not be forgotten and they have made studying and working at Edinburgh an enjoyable experience.

Finally, and most importantly, I would like to thank my parents. Their advice and encouragement at the right times has led me here and has played an immeasurable role in allowing me to compile this thesis – a debt insufficiently repaid. Similarly I would like to thank *Jasmin Paris*, who has been very accepting of my absentmindedness induced by pondering aspects of this work and who has been a constant reminder that there are more important things in life.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Konrad C. Rawlik*)

To J.R. & T.R. and J.P.

Table of Contents

List of Notation

Below is a list of symbols and abbreviations used throughout this thesis (unless an exception is noted in the text). Entries of the form $f(\cdot)$ denote an argument should be supplied to the function f, for example where there is a direct dependency on some quantity. In addition to the terms defined here, note that we use the convention of bold upper-case letters, $\mathbf{A}$, to denote matrices, bold lower-case letters, $\mathbf{a}$, to denote vectors and normal weighted font, a, to denote scalar terms. Finally, we use the general notation $x_{i:j}$ to denote the vector of elements $(x_i, x_{i+1}, \ldots, x_j)$ and, in the specific instance of dynamic variables, the more concise $\bar{x} = x_{0:K}$ to denote entire trajectories of x.

Symbols:

$\mathbf{x}$	State
$\mathbf{u}$	Control
r	Auxiliary task variable
t, T	Time and time horizon, e.g. trajectory length
k, K	Discrete time step and total number of time steps, e.g. trajectory length
n, N	algorithm iteration and total number of iterations
m, M	sample index and total number of samples
$q_\pi(\bar{\mathbf{x}}, \bar{\mathbf{u}})$	State-control trajectory distribution under the policy π, see (3.2)
$\mathcal{C}, \mathcal{C}_T, \mathcal{C}_\bullet$	Cost, terminal cost, time stationary cost
$\mathcal{J}(\cdot)$	Stochastic optimal control objective
$\mathcal{V}(\cdot)$	Optimal value function

$\mathcal{N}(\cdot,\cdot), \mathcal{N}[\cdot,\cdot]$	Gaussian distribution in standard and canonical form respectively
$\mathbb{E}_p[f(\cdot)]$	Expectation of f with respect to p
$\mathrm{KL}(\cdot\|\cdot)$	Kullback-Leibler divergence
$\mathbf{g}^k_{\mathcal{AB}}$	The Gramian matrix $[\mathbf{G}^k_{\mathcal{AB}}]_{ij} = k(a_i, b_j)$
$\mathbf{g}^k_{\mathcal{A}}$	The feature matrix $[\mathbf{g}^k_{\mathcal{A}}]_i = k(a_i, \cdot)$
$\mathcal{H}^k$	Hilbert space with reproducing kernel k

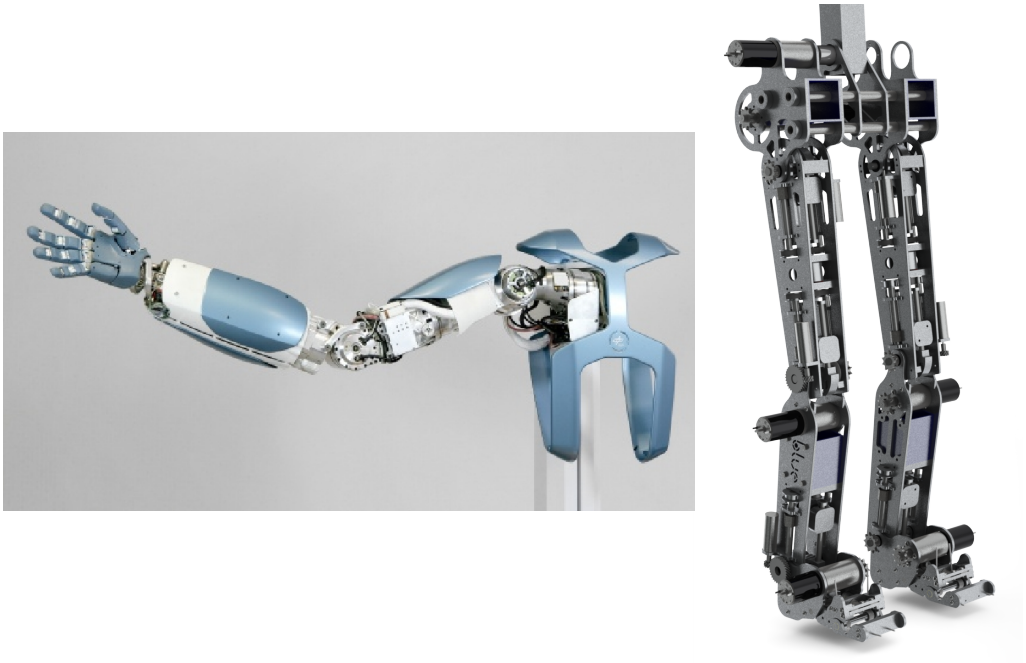
Acronyms:

AICO	Approximate Inference Control
EM	Expectation Maximisation
HJB	Hamilton-Jacobi-Bellman
iLQG	iterative Linear-Quadratic-Gaussian
KL	Kullback-Leibler
LQ	Linear-Quadratic
LQG	Linear-Quadratic-Gaussian
MAP	maximum a posteriori
MC	Monte Carlo
MDP	Markov Decision Process
ODE	ordinary differential equation
PDE	partial differential equation
PPI	Posterior Policy Iteration
PPI-T	temporal PPI
RKHS	Reproducing Kernel Hilbert Space
RL	Reinforcement Learning
SDE	stochastic differential equation
SOC	Stochastic Optimal Control

Chapter 1

Introduction

Humans and other biological systems solve the motor control problem they face with surprising ease. One just needs to consider the clumsiness of infants – or our attempts at controlling autonomous robotic systems – to appreciate how complex the task is and with what sophistication the nervous system solves it. In particular, the ability to combine efficiency with accuracy and robustness, in spite of perturbations and inherent uncertainties, has long served as an inspiration for robotics research. With advances in hardware design, we are now seeing systems which increasingly exhibit capabilities comparable to their biological counterparts. However, such systems – examples of which are illustrated in Figure 1.1 – raise new challenges. Foremost, they typically have high dimensional state & control spaces. A consequence is redundancy, i.e., the capability of achieving the same behaviour in different ways, e.g., achieving the same end effector position with different joint configurations. Such redundancy, which modern system exhibit not only on the level of kinematics but also dynamics and controls, is both a curse and an opportunity. On the one hand, it significantly complicates the control problem, requiring methodologies which can select amongst the different possible solutions. On the other hand, with redundancy comes the possibility of exploiting the system's capabilities to concurrently achieve secondary objectives. However, the problem is further complicated by the increasing difficulty to identify the dynamics of the systems under consideration – either in isolation or when interacting with an only partially observable environment – necessitating a stochastic view of the problem. The above can be summarised as a fundamental



(a) (b)

Figure 1.1: Examples of modern anthropomorphic robots aiming to emulate capabilities of biological systems, in these examples specifically, compliance and variable impedance actuation. These systems have high dimensional state and control spaces and complicated non-linear dynamics. (a) The DLR hand-arm system (Grebenstein et al., 2011), which has 26 kinematic degrees of freedom and is actuated by 52 motors. (b) The BLUE bipedal system (Enoch et al., 2012), which incorporates variable stiffness and damping actuation in each of the 6 actively controlled joints.

question in robotics: how to select the controls in a task, such as to mitigate uncertainties and fully exploit the capabilities of the plant?

Stochastic Optimal Control (SOC) provides a framework which gives a principled answer to this question. It's starting point is a formal definition of the task in terms of a cost which captures the desired behaviour. Such costs typically consists of combinations of criteria like error measures, e.g., distances to the desired target, energy consumption or movement durations. Based on the cost, controls are chosen such that the expectation of the cost with respect to the arising trajectory distribution is minimised. Importantly, the formulation incorporates both the task specification and the system dynamics. Such an integrated

procedure is in stark contrast to traditional approaches to the control problem, which typically solve the task independent of the dynamics, dividing the control problem into trajectory planning and subsequent trajectory tracking (An et al., 1988).

While the characteristics of modern systems make the application of SOC desirable, they also pose the greatest challenge to it's viability. In general, SOC remains analytically intractable beyond small classes of basic problems. As such the central focus within the field has been on formulation of appropriate frameworks which allow efficient numerical computation of approximate solutions.

In this thesis, we approach the problem from an probabilistic inference perspective. In general, probabilistic inference concerns the problem of inferring the value of unobserved variables based on stochastically related observed quantities. A typical example and application in the context of robotics is state estimation, that is, inferring the state of the plant based on noisy sensor measurements. Intuitively, the control problem can be phrased in terms of an inference problem by framing it as the question *"What does have to happen for the task to have been achieved?"*. That is, we consider the problem of inferring the intermediate (unobserved) states and controls which led to an imagined future observation of task achievement. This thesis provides a general investigation of the relations between such inference problems and SOC, formalising a general duality between the two.

Developing an understanding of the relation between SOC and probabilistic inference is not only an interesting theoretical exercise. Probabilistic inference, particularly in dynamical systems, faces many of the same challenges faced by SOC. These include the lack of analytical solutions, non-linear relations amongst variables, manipulation of distributions over continuous spaces and scaling to high dimensional problems. This has lead to the development of a rich literature on approximations, both analytical and sample based (e.g., Bishop, 2006; MacKay, 2003), many of which lack analogues in the SOC literature. A precise understanding of the correspondences between SOC and probabilistic inference allows not only for transfer of successful inference methods, it also provides a new perspective on the problem which may lead us to entirely new combinations of approaches from both fields. We illustrate the utility of such an approach in

this thesis, by deriving a series of novel algorithms based on our insights into the relations between the two problems.

1.1 Thesis Outline

The structure and contributions of this thesis can be summarised as follows.

We begin by providing an introduction to SOC in **Chapter 2**. Specifically, we first define the SOC problem and the closely related Reinforcement Learning (RL) formulation. We then review the characterisation of solutions of SOC based on Bellman's Minimum Principle in the discrete and continuous time setting. Finally, we discuss the specific application of SOC to problems in robotics, providing a brief overview of relevant previous work.

In **Chapter 3**, we discuss the formulation of SOC in terms of inference based approaches. An extensive overview of previous work is given, covering formulations based on both, maximum a posteriori (MAP) and Bayesian estimation, which have lead to dualities for restricted classes of problems. This leads us to observe a novel connection between certain Kullback-Leibler (KL) divergences and the Bellman equation, giving rise to a novel duality for the general SOC problem.

Original Contributions:

- *Review and classification of inference based dualities to the SOC problem and discussion of their limitations.*
- *Formulation of a novel discrete time duality not restricted by specific assumptions on the system dynamics or costs.*

However, the KL divergence based duality, although general, does not directly yield analytically tractable solutions. In **Chapter 4**, we therefore study a natural relaxation of the exact dual, which gives rise to iterative solutions to the finite and infinite horizon stochastic optimal control problem. On the basis of these we introduce the Ψ-Learning algorithm, a new model free RL algorithms for problems with both discrete and continuous state and action spaces. This is evaluated on standard benchmark problems.

Original Contributions:

- *Formulation of iterative policy search, based on a general policy improvement condition*
- *Characterisation of the asymptotic behaviour of the proposed iterations*
- *Formulation of model free Reinforcement Learning algorithms, for both, finite and continuous state and control spaces.*

In **Chapter 5**, we propose an alternative relaxation of our duality. This gives rise to the Posterior Policy Iteration (PPI) procedure which leads to risk sensitive SOC solutions by a process of iterated Bayesian inference. The benefit of this formulation is the potential for employing efficient approximation algorithms from Machine Learning for general SOC problems. The proposed algorithms are implemented on a state of the art arm-hand system.

Original Contributions:

- *Characterisation of a general connection between Bayesian Inference and Risk Sensitive SOC.*
- *Formulation of practical algorithms based on exact and approximate inference.*
- *Evaluation of the proposed algorithms on a modern robotic manipulator.*

In **Chapter 6**, we present an embedding of SOC problems, of the so called path integral form (also applicable to PPI) into a Reproducing Kernel Hilbert Space (RKHS). Using consistent, sample based estimates of the embedding leads to a model free, non-parametric approach for calculation of an approximate solution to the control problem. This formulation admits a decomposition of the problem into an invariant and task dependent component. Consequently, we make much more efficient use of the sample data compared to previous sample based approaches in this domain, e.g., by allowing sample re-use across tasks. Numerical examples on test problems, which illustrate the sample efficiency, are provided.

Original Contributions:

- *Formulation of a RKHS embedding of general Risk Sensitive SOC problems and certain risk neutral SOC problems.*
- *Formulation of sample based algorithms based on empirical estimates of the embedding*
- *Formulation of various efficient empirical estimators, including those based on sample transfer across tasks.*

In **Chapter 7**, we present a methodology capable of jointly optimizing temporal parameters, e.g. the movement duration or the time point of reaching an intermediate goal, in addition to the control command profiles. We propose a formulation of this extended SOC problem which can be reduced to a finite horizon problem, allowing application of all previously proposed methods. In particular, we extend PPI to this setting, providing, to the best of our knowledge, a first practical approach to tackling generic via point problems in a systematic way under a SOC framework. The advantages of such temporal optimisation are illustrated on various plants, highlighting amongst others, the importance of good temporal parameters in dynamic tasks.

Original Contributions:

- *Formulation of the problem of SOC with temporal optimisation and illustration of it's reduction to finite horizon problem.*
- *Formulation of practical algorithms for SOC with temporal optimisation based on approximate inference.*
- *Evaluation of the proposed algorithms on a modern robotic manipulator.*

Finally, in **Chapter 8**, we give conclusions and suggest directions for future work.

1.2 Publication Summary

This thesis provides an enhanced and extended presentation of some elements which have been or are expected to shortly be published. The general duality of Chapter 3 and it's relaxations in Chapter 4 and Chapter 5 have been first published in Rawlik et al. (2012). Rawlik et al. (2010) describes the Expectation Maximisation (EM) based temporal optimisation framework, while Nakanishi et al. (2011) discusses an application of temporal optimisation to periodic movements. Finally, publications based on Chapter 6 and Chapter 4 are currently under review.

Publications and Submissions:

- Rawlik, K. and Toussaint, M. and Vijayakumar, S. (2012). On Stochastic Optimal Control and Reinforcement Learning by Approximate Inference. In *Proc. Robotics: Science and Systems VIII*.
- Nakanishi, J. and Rawlik, K. and Vijayakumar, S. (2011). Stiffness and Temporal Optimization in Periodic Movements: An Optimal Control Approach. In *Proc. Int. Conf. on Intelligent Robots and Systems*.
- Rawlik, K. and Toussaint, M. and Vijayakumar, S. (2010). An Approximate Inference Approach to Temporal Optimization in Optimal Control. In *Proc. Advances in Neural Information Processing Systems*.
- Rawlik, K. and Toussaint, M. and Vijayakumar, S.. Approximate Inference Formulations of Stochastic Optimal Control and Reinforcement Learning. Submitted to *Autonomous Robots*.
- Rawlik, K. and Toussaint, M. and Vijayakumar, S.. Path Integral Control by Reproducing Kernel Hilbert Space Embedding. Submitted to *Int. Conf. on Artificial Intelligence and Statistics*.

Chapter 2

Stochastic Optimal Control

The problem of (deterministic) optimal control can, informally, be summarise as:

> *For a given dynamical system with inputs, determine a control sequence, such that the associated state trajectory minimises a given objective.*

Due to either incomplete knowledge of the system or it's corruption by noise, a situation commonly arising in practise is that the dynamical system is in fact stochastic. It is apparent that, such cases require adaptation of the above problem statement. For example, a control sequence can no longer be associated with a specific trajectory, rather giving rise to a distribution over such trajectories. The resulting problem is the SOC problem and forms the main focus of this thesis.

In this chapter we review the relevant background on SOC. Specifically we will first formalise the problem statement in Section 2.1, also discussing the special case of RL. We then proceed to review the dynamic programming approach to these problems. Finally, in Section 2.3 we discuss considerations specific to the application of SOC to robotics problems and briefly survey relevant previous work.

The presentation will necessarily be concise and, at times, where this does not affect the results, we sacrifice rigour for simplicity of presentation. This is in particular the case with regards to the treatment of stochastic processes, a more thorough treatment of which is provided by, e.g., Øksendal (2010). On aspects of control theory beyond the scope of this thesis, the reader may similarly refer to any of the many excellent text books available on the subject (e.g., Stengel,

1986; Bryson and Ho, 1975; Bertsekas, 1995). Alternative introductions to the subject matter, giving a broader overview at a level of detail comparable to ours, are provided by Todorov (2006b) and Kappen (2011).

2.1 Problem Statement

Let us begin by formalising the problem of interest to the degree necessary. A SOC problem comprises the two components apparent from the earlier statements, a system and an objective, and in addition a policy space. We examine each of these in turn, before stating the formal problem.

- **Controlled System:** This is a representation of the behaviour of the system over time under the influence of given controls, usually taking the form of dynamics in state space form.

  Formally, let $\mathcal{X} \subseteq \mathbb{R}^{D_x}$ be the state space and $\mathcal{U} \subseteq \mathbb{R}^{D_u}$ the control space. In the discrete time setting we will consider general controlled Markov processes. That is, the distribution over state trajectories, given controls, factorises as

  $$P(\mathbf{x}_{1:K}|\mathbf{u}_{0:K-1}, \mathbf{x}_0) = \prod_{k=0}^{K-1} P(\mathbf{x}_{k+1}|\mathbf{x}_k, \mathbf{u}_k) \ .$$

  The individual factors are referred to as transition distributions. In continuous time, we shall concern ourselves with systems which take the form of the general stochastic differential equation (SDE)

  $$\mathrm{d}\mathbf{x} = f(\mathbf{x}, \mathbf{u})\mathrm{d}t + g(\mathbf{x}, \mathbf{u})\mathrm{d}\omega, \qquad \mathbb{E}\left[\mathrm{d}\omega \mathrm{d}\omega^T\right] = \mathbf{I}\mathrm{d}t \ ,$$

  where $f : \mathcal{X} \times \mathcal{U} \to \mathbb{R}^{D_x}$ and $g : \mathcal{X} \times \mathcal{U} \to \mathbb{R}^{D_x \times D_\omega}$, i.e., f and g are vector and matrix valued functions respectively, with D_ω the dimension of the noise ω. Furthermore, $\mathrm{d}\omega$ is a multivariate Wiener process and formally the SDE is to be interpreted as an Itō integral.

- **Policy Space:** This defines the set of policies over which optimisation is performed, i.e., the domain of the optimisation problem.

For our purposes, a policy π shall be a (conditional) distribution[1] over $\mathcal{U}$, with a policy space some subset of all policies. For example in deterministic optimal control the policy space is usually given by *open loop policies*, that is the set of all distributions $\pi(\mathbf{u}|t)$. In SOC on the other hand, the main interest lies with closed loop Markov policies, also known as *feedback policies*, where the controls are conditioned on both time[2] and the current state, i.e., $(t, \mathbf{x}(t))$. A special subset of these is formed by *deterministic policies*, for which $\pi(\mathbf{u}|\mathbf{x}) = \delta_{\mathbf{u}=f(\mathbf{x})}$, where δ is the delta distribution and f is some function.

- **Objective:** This is the functional to be optimised and constitutes a formal description of the task one is interested in solving with the given system.

  Formally, an objective is given by any functional of a policy, initial state pairs, taking values in $\mathbb{R}_+$. Here, our interest lies foremost with the, so called, *finite horizon problem*, for which the objective takes the form

$$\mathcal{J}(\pi, \mathbf{x}(0)) = \mathbb{E}_{\mathbf{x}(\cdot),\mathbf{u}(\cdot)|\mathbf{x}(0),\pi}\Big[\underbrace{\mathcal{C}_T(\mathbf{x}(T)) + \int_0^T \mathcal{C}(\mathbf{x}(t), \mathbf{u}(t), t)\mathrm{d}t}_{=\mathcal{C}(\mathbf{x}(\cdot),\mathbf{u}(\cdot))}\Big] ,$$

  where T is a given horizon and comprises a terminal cost $\mathcal{C}_T$, e.g., in a reaching task the end-effector distance to the target, and the integration of a cost rate $\mathcal{C}$, e.g., energy consumption. As it consists of the expected value of a trajectory cost, this objective is also often referred to as *expected cost*. Despite the term *expected cost minimisation* often being used synonymously with SOC, many alternative objectives have been proposed, with Table 2.1 providing some alternative objectives based on integration of an underlying cost rates.

The SOC problem can now be formalised as, given a policy space $\mathcal{P}$, objective $\mathcal{J}$ and initial state $\mathbf{x}_0$), finding a policy $\pi^* \in \mathcal{P}$ s.t.,

$$\mathcal{J}(\pi^*, \mathbf{x}_0) = \inf_{\pi\in\mathcal{P}} \mathcal{J}(\pi, \mathbf{x}_0) \ . \tag{2.1}$$

[1] n.b. we denote by π the density function (or probability mass function, as may be the case) and as such write $\pi(\mathbf{u}|\cdot)$ rather then the usual $\mathbf{u} = \pi(\cdot)$

[2] n.b. in the interest of a more concise notation we will implicitly assume conditioning on time and write $\pi(\mathbf{u}(t)|\mathbf{x}(t))$

Name	$\mathcal{J}(\pi, \mathbf{x}_0)$	
Finite Horizon:	$\mathbb{E}\left[\mathcal{C}_T(\mathbf{x}(T)) + \int_0^T \mathcal{C}(\mathbf{x}(t), \mathbf{u}(t), t)\mathrm{d}t\right]$	
First Exit:	$\mathbb{E}\left[\mathcal{C}_T(\mathbf{x}(T)) + \int_0^T \mathcal{C}(\mathbf{x}(t), \mathbf{u}(t), t)\mathrm{d}t\right]$	$T = \inf\{t; \mathbf{x}(t) \in \mathcal{X}_{exit}\}$
Discounted:	$\mathbb{E}\left[\int_0^\infty \gamma^t \mathcal{C}_\bullet(\mathbf{x}(t), \mathbf{u}(t))\right]$	$\gamma \in [0, 1]$
Average:	$\lim_{T\to\infty} \frac{1}{T}\mathbb{E}\left[\int_0^T \mathcal{C}_\bullet(\mathbf{x}(t), \mathbf{u}(t))\mathrm{d}t\right]$	
Risk Sensitive:	$-\eta \log \mathbb{E}\left[\exp\{-\frac{1}{\eta}(\mathcal{C}_T(\mathbf{x}(t)) + \int_0^T \mathcal{C}(\mathbf{x}(t), \mathbf{u}(t))\mathrm{d}t)\}\right]$	

Table 2.1: Common objective functions for continuous time SOC. All expectations are w.r.t. to trajectories under policy π with $\mathbf{x}(0) = \mathbf{x}_0$. We use $\mathcal{C}_T$ to denote a terminal cost, while $\mathcal{C}$ and $\mathcal{C}_\bullet$ denote time varying and time stationary cost rates respectively.

Unless otherwise stated, we will from here on assume $\mathcal{J}$ to be the finite horizon objective and in the interest of uncluttered notation, will not explicitly note the dependence on $\mathbf{x}_0$, writing $\mathcal{J}(\pi)$.

On a final note, although we have assumed both states and controls to be continuous, we shall occasionally discuss discrete time problems with (finite) discrete state and control sets. We shall refer to such a problem as a Markov Decision Process (MDP), implicitly assuming continuous state and control sets when referring to a SOC problem.

2.1.1 Reinforcement Learning

A central assumption in the formulation of the SOC problem was knowledge of both the objective and the system dynamics. In the problems considered in this thesis the objective is chosen by the operator and as such may generally be assumed as known. The system dynamics however may not be available or, even if known, be corrupted by interactions with objects in the environment, e.g., lifting an object of unknown mass. Several formulations extending the SOC

problem to situations with incomplete knowledge of dynamics and objective have been proposed. Of these Reinforcement Learning (RL) provides the most general framework which subsumes alternative formulations, like adaptive control, as special cases.

A general introduction to RL is given in the classical text by Sutton and Barto (1998), while Szepesvári (2010) provides a more recent review. In it's general form, RL is concerned with the problem of learning good – w.r.t. an objective – behaviour from interactions with the environment. Typically one considers an agent in closed loop interaction with the environment. That is, the agent observes it's state and chooses a control[3], upon which he transitions to a new state, while at the same time receiving a reward associated with the state transition. The aim is to, over the course of such interaction, learn a control policy which maximises the future discounted rewards.

Throughout the literature large amounts of variation are encountered with regards to episodic nature of tasks, objective and extend of prior knowledge about the dynamics and objective. While, Togelius et al. (2009) provide a short overview of some of the variety of different assumption encountered, in this thesis, we shall use the term RL to, in general, denote solving an SOC problem based on samples. Specifically, finding π^* only having access to samples from the dynamics and cost function[4], where we leave the process by which such samples are acquired open to further specification.

2.2 Bellman's Minimum Principle

Solutions of the deterministic optimal control problem can be characterised by two alternative approaches, Pontryagin's Minimum Principle (Boltyanskiy et al., 1962), which has it's origin in the Calculus of Variation, and Bellman's Minimum Principle (Bellman, 1957). However, while Pontryagin's principle has computational benefits over Bellman's principle, e.g., it avoids exponential complexity

[3] n.b. *controls* are often referred to as *actions* in the RL literature

[4] n.b. we assume that, up to the cost functions, the general form of the objective $\mathcal{J}(\cdot,\cdot)$ is known, including any open parameters, e.g., in the discounted infinite horizon case (cf. Table 2.1) γ is known but only point wise evaluations of $\mathcal{C}_\bullet$ are available.

scaling with the problem dimensionality, it does not readily generalise to the stochastic case (although see Yong and Zhou, 1999). We therefore restrict our discussion to Bellman's principle and the so called Dynamic Programming solution arising from it.

2.2.1 Discrete Time

Dynamic Programming is based on the following observation made by Bellman (1957, Chap. III.3.)

Principle of Optimality. *An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision.*

In order to make this intuitive idea more concrete let us define the *optimal value function* $\mathcal{V}_k(\mathbf{x}_k)$ – also known as the *cost to go* – as the expected cost of starting in state $\mathbf{x}_k$ at time step k and following the optimal policy thereafter. The implication of the Principle of Optimality is, that this optimal value function can be expressed by the following recursive equation, known as the Bellman equation,

$$\mathcal{V}_k(\mathbf{x}_k) = \inf_{\mathbf{u}\in\mathcal{U}} \{\mathcal{C}(\mathbf{x}_k, \mathbf{u}) + \mathbb{E}_{\mathbf{x}_{k+1}|\mathbf{x}_k,\mathbf{u}} [\mathcal{V}_{k+1}(\mathbf{x}_{k+1})]\} \ . \tag{2.2}$$

In the MDP case, with a finite horizon objective, the Bellman equation can be directly solved backwards in time, a process referred to as Dynamic Programming. For general SOC however, analytical tractability of Dynamic Programming is not guaranteed and is in fact limited to a small number of problem classes. One such class is the Linear-Quadratic-Gaussian (LQG) problem, on which we illustrate the use of the Bellman equation in the following example.

Example 2.1 (LQG problem)**:** Consider the classical LQG problem which, in the finite horizon setting, consists of dynamics

$$\mathbf{x}_{k+1} = \mathbf{f} + \mathbf{F}^x\mathbf{x}_k + \mathbf{F}^u\mathbf{u}_k + \eta, \qquad \eta \sim \mathcal{N}(0, \mathbf{Q}) \ ,$$

and cost function

$$\mathcal{C}(\bar{\mathbf{x}}, \bar{\mathbf{u}}) = \sum_{k=0}^{K} \left(\mathbf{x}_k^T\mathbf{C}_k^x\mathbf{x}_k + \mathbf{x}_k^T\mathbf{c}_k^x + \mathbf{u}_k^T\mathbf{C}_k^u\mathbf{u}_k + \mathbf{u}_k^T\mathbf{c}_k^u\right) \ .$$

In the interest of a concise presentation however, we shall limit ourselves to the case $\mathbf{f} = 0$, $\mathbf{c}_k^x = 0$ and $\mathbf{c}_k^u = 0$.

We will demonstrate that the optimal value function takes the form

$$\mathcal{V}_k(\mathbf{x}) = \mathbf{x}^T \mathbf{V}_k \mathbf{x} + \mathbf{v}_k \ ,$$

by applying the Bellman equation (2.2) backwards in time. This is trivially true at time step K with $\mathbf{V}_K = \mathbf{C}_K^x$ and $\mathbf{v}_k = 0$. Now working inductively backwards in time

$$\mathcal{V}_k(\mathbf{x}_k) = \inf_{\mathbf{u}\in\mathcal{U}} \{\mathbf{x}_k^T \mathbf{C}_k^x \mathbf{x}_k + \mathbf{u}_k^T \mathbf{C}_k^u \mathbf{u}_k + \underbrace{\mathbb{E}_{\mathbf{x}_{k+1}|\mathbf{x}_k,\mathbf{u}} \left[\mathcal{V}_{k+1}(\mathbf{x}_{k+1})\right]}_{(\mathbf{F}^x\mathbf{x}_k+\mathbf{F}^u\mathbf{u}_k)^T\mathbf{V}_{k+1}(\mathbf{F}^x\mathbf{x}_k+\mathbf{F}^u\mathbf{u}_k)+\mathrm{Tr}(\mathbf{Q}\mathbf{V}_{k+1})+\mathbf{v}_{k+1}}\} \ . \tag{2.3}$$

As the expression is quadratic in $\mathbf{u}$ the minimisation can be performed analytically, yielding

$$\mathbf{u}^{\inf} = -(\mathbf{C}^u + \mathbf{F}^{uT}\mathbf{V}_{k+1}\mathbf{F}^u)^{-1}\mathbf{F}^{uT}\mathbf{V}_{k+1}\mathbf{F}^x\mathbf{x}_k \ . \tag{2.4}$$

Substituting back into (2.3) we find that $\mathcal{V}_k(\mathbf{x}_k)$ is indeed of the required form with

$$\begin{aligned} \mathbf{V}_k &= \mathbf{C}^x + \mathbf{F}^{xT}(\mathbf{V}_{k+1} - \mathbf{V}_{k+1}\mathbf{F}^u(\mathbf{C}^u + \mathbf{F}^{uT}\mathbf{V}_{k+1}\mathbf{F}^u)^{-1}\mathbf{F}^{uT}\mathbf{V}_{k+1})\mathbf{F}^x \\ \mathbf{v}_k &= \mathrm{Tr}\left(\mathbf{Q}\mathbf{V}_{k+1}\right) + \mathbf{v}_{k+1} \end{aligned} \tag{2.5}$$

As the optimal controls (2.4) are independent of the $\mathbf{v}$'s, these are usually not computed and the Riccati equation (2.5) is generally considered to be the solution.

Obtaining corresponding results in the more general case when $\mathbf{f} \neq \mathbf{0}$, $\mathbf{c}_k^x \neq \mathbf{0}$ or $\mathbf{c}_k^u \neq \mathbf{0}$ is straightforward. In fact the above steps may be carried out whenever evaluation of the expectation in (2.3) yields a quadratic expression in $(\mathbf{x}_k, \mathbf{u}_k)$. As such, the solution can be further generalised to problems for which $\mathbf{Q} = \mathbf{x}_k^T\mathbf{Q}^x\mathbf{x}_k + \mathbf{u}_k^T\mathbf{Q}^u\mathbf{u}_k + \mathbf{x}_k^T\mathbf{Q}^{xu}\mathbf{u}_k$, i.e., with state and control dependent noise.

Finally, in general, the particular form of the Bellman equation is specific to the considered objective, with (2.2) being associated with the finite horizon setting. For example, consider the discounted infinite horizon objective from Table 2.1.

The arising control problem is time stationary and it is easy to show that the Bellman equation takes the implicit form

$$\mathcal{V}(\mathbf{x}) = \inf_{\mathbf{u}\in\mathcal{U}} \{\mathcal{C}(\mathbf{x},\mathbf{u}) + \gamma \mathbb{E}_{\mathbf{y}|\mathbf{x},\mathbf{u}}[\mathcal{V}(\mathbf{y})]\} \ .$$

2.2.2 Continuous Time

The Principle of Optimality naturally extends to continuous time, in the sense that an optimal policy over some $[t,T]$ is also necessarily optimal over any $[t,t+\Delta] \subset [t,T]$. The limit $\Delta \to 0$ can be shown (see, e.g., Todorov, 2006b) to give rise to the partial differential equation (PDE)

$$\partial_t \mathcal{V} = \inf_{\mathbf{u}\in\mathcal{U}} \left(\mathcal{C} + (\nabla\mathcal{V})^T f + \frac{1}{2}\operatorname{Tr}\left(\mathbf{Q}\nabla^2\mathcal{V}\right)\right), \qquad \mathbf{V}(\mathbf{x},T) = \mathcal{C}_T(\mathbf{x}) \ , \tag{2.6}$$

which is the continuous time analogue of the Bellman equation. It is known as the Hamilton-Jacobi-Bellman (HJB) equation and it's form, like the Bellman equation's, is dependent on the objective, (2.6) arising from the finite horizon setting. As an illustrative example of it's use we consider the continuous time problem corresponding to Example 2.1.

Example 2.2 (Continuous Time LQG Problem)**:** Consider the finite horizon problem

$$\mathrm{d}\mathbf{x} = (\mathbf{F}^x\mathbf{x} + \mathbf{F}^u\mathbf{u})\mathrm{d}t + \mathrm{d}\omega, \qquad \mathbb{E}\left[\mathrm{d}\omega \mathrm{d}\omega^T\right] = \mathbf{Q}\mathrm{d}t \ ,$$
$$\mathcal{C}(\mathbf{x}(\cdot),\mathbf{u}(\cdot)) = \int_0^T \frac{1}{2}\mathbf{x}(t)^T\mathbf{C}^x\mathbf{x}(t) + \frac{1}{2}\mathbf{u}(t)^T\mathbf{C}^u\mathbf{u}(t)\mathrm{d}t \ .$$

We proceed by guessing the parametric form of the value function to be

$$\mathcal{V}(\mathbf{x},t) = \mathbf{x}^T\mathbf{V}(t)\mathbf{x} + \mathbf{v}(t) \ ,$$

showing that the guess satisfies the HJB equation and obtaining ordinary differential equations (ODEs)s for $\mathbf{V}, \mathbf{v}$.

The derivatives appearing in the HJB take the forms

$$\partial_t\mathcal{V}(\mathbf{x},t) = \mathbf{x}^T\dot{\mathbf{V}}(t)\mathbf{x} + \dot{\mathbf{v}}(t) \ , \quad \nabla\mathcal{V}(\mathbf{x},t) = \mathbf{V}(t)\mathbf{x} \ , \quad \nabla^2\mathcal{V}(\mathbf{x},t) = \mathbf{V}(t) \ ,$$

so that substituting we have

$$-\mathbf{x}^T\dot{\mathbf{V}}(t)\mathbf{x} - \dot{\mathbf{v}}(t)$$
$$= \inf_u \left(\mathbf{x}^T\mathbf{C}^x\mathbf{x} + \frac{1}{2}\mathbf{u}^T\mathbf{C}^u\mathbf{u} + \mathbf{x}^T\mathbf{V}(t)^T(\mathbf{F}^x\mathbf{x} + \mathbf{F}^u\mathbf{u}) + \frac{1}{2}\operatorname{Tr}\left(\mathbf{Q}\mathbf{V}(t)\right)\right) \ .$$

As in the discrete time case we can resolve the minimisation analytically by

$$\mathbf{u}^{\inf} = -\mathbf{C}^{u-1}\mathbf{F}^{uT}\mathbf{V}(t)\mathbf{x} \ ,$$

so that the HJB with resolved controls can be reduced to

$$\begin{aligned} -\mathbf{x}^T\dot{\mathbf{V}}(t)\mathbf{x} - \dot{\mathbf{v}}(t) = \mathbf{x}^T\Big(&\mathbf{C}^x + \mathbf{F}^{xT}\mathbf{V}(t) + \mathbf{V}(t)^T\mathbf{F}^x \\ &+ \mathbf{V}(t)\mathbf{F}^u\mathbf{C}^{u-1}\mathbf{F}^{uT}\mathbf{V}(t)\Big)\mathbf{x} + \frac{1}{2}\operatorname{Tr}\left(\mathbf{Q}\mathbf{V}(t)\right) \ . \end{aligned}$$

Matching terms on both sides we conclude that the optimal value function is given by the solution of the system of ODEs

$$\begin{aligned} -\dot{\mathbf{V}} &= \mathbf{C}^x + \mathbf{F}^{xT}\mathbf{V}(t) + \mathbf{V}(t)^T\mathbf{F}^x + \mathbf{V}(t)\mathbf{F}^u\mathbf{C}^{u-1}\mathbf{F}^{uT}\mathbf{V}(t) \ , \\ -\dot{\mathbf{v}} &= \frac{1}{2}\operatorname{Tr}\left(\mathbf{Q}\mathbf{V}(t)\right) \ , \end{aligned}$$

with boundary conditions $\mathbf{V}(T) = 0$, $\mathbf{v}(T) = 0$.

As for the Bellman equation, analytical solutions of the HJB equation are limited to a small number of problem classes beyond LQG (e.g., Huh and Sejnowski, 2011). One particular case are a class of problems for which the HJB can be reduced to a linear PDE. These problems have received increasing attention in the literature under the names of *Path Integral Control* (Kappen, 2005) and *Linearly Solvable Control* (Todorov, 2009b). Due to their close connection to certain inference problems (cf. Section 3.2.2.2), they are of particular interest here and we conclude with a brief discussion of their HJB equation.

We begin with the observation, that a crucial step in obtaining the results for the LQG case (cf. Example 2.2) was analytical resolution of the minimisation in the HJB equation. It is furthermore apparent, that it is sufficient for the problem to be Linear-Quadratic (LQ) in the controls only, for analytical minimisation to be possible. Specifically, for a finite horizon problem of the form

$$\begin{aligned} \mathrm{d}\mathbf{x} &= (f(\mathbf{x}) + \mathbf{F}^u\mathbf{u})\mathrm{d}t + \mathrm{d}\omega \\ \mathcal{C}(\mathbf{x}(\cdot), \mathbf{u}(\cdot)) &= \int_0^T \mathcal{C}(\mathbf{x}(t)) + \frac{1}{2}\mathbf{u}(t)^T\mathbf{C}^u\mathbf{u}(t)\mathrm{d}t \end{aligned}$$

the analytical solution to the minimisation is given by

$$\mathbf{u}^{\inf}(t) = -\mathbf{H}^{-1}\mathbf{F}^{uT}\nabla\mathcal{V}(\mathbf{x}(t), t)$$

Substitution into the HJB equation then yields the non-linear PDE

$$\partial_t \mathcal{V} = \mathcal{C} + (\nabla \mathcal{V})^T f + \frac{1}{2} \operatorname{Tr}\left(-\mathbf{F}^u \mathbf{C}^{u-1} \mathbf{F}^{uT} (\nabla \mathcal{V})(\nabla \mathcal{V})^T + \mathbf{Q} \nabla^2 \mathcal{V}\right)$$

However under the additional condition $\lambda \mathbf{F}^u \mathbf{C}^{u-1} \mathbf{F}^{uT} = \mathbf{Q}$ the linear PDE

$$-\partial_t \psi = -\frac{1}{\lambda} \mathcal{C} \psi + (\nabla \psi)^T f + \frac{1}{2} \operatorname{Tr}\left(\mathbf{F}^u \mathbf{Q} \mathbf{F}^{uT} \nabla^2 \psi\right) \tag{2.7}$$

in $\psi(\mathbf{x}, t) = \exp\{-\lambda^{-1} \mathcal{V}(\mathbf{x}, t)\}$ is obtained.

Remark 2.1: Consistent with the discussion in Section 2.1 we have throughout assumed the state (and control) space to be continuous, a fact apparent from the appearance of $\nabla \mathcal{V}$ in the HJB equation. Although continuous time MDPs are beyond the scope of this thesis, we note that they can be formulated based on Markov Jump processes and a HJB like equation can similarly be derived (Theodorou, 2011).

2.3 Applications to Robotics

Stochastic Optimal Control is an appealing general framework for robotics as, upon framing the task in terms of a cost function, it implicitly resolves any redundancies and yields controls which exploit the capabilities of the system, while taking it's limitations into account. Although it is certainly not the only general framework (see, e.g., Morimoto et al., 2003; Guigon et al., 2008) it has been widely adopted, despite it's application being by no means straightforward. The key challenges posed by SOC problems, in the context of robotics, may be summarised as follows:

- The state of a typical robot manipulator comprises, at it's most basic, the joint angles and their velocities, while controls are given by the torques or a voltage signal. Thus the problems have **continuous state and control spaces**.

- The ever increasing sophistication of robotic plants comes at the cost of ever increasing complexity, which entails **high dimensional state and control spaces**. Robotic plants typically have more then 10 kinematic

degrees of freedom, e.g., the hand-arm system used in this thesis has 15 kinematic degrees of freedom, with the number usually higher in anthropomorphic designs, e.g., an anthropomorphic hand-arm system developed at the German Aerospace Centre (DLR) has 26 kinematic degrees of freedom and is actuated by 52 motors (Grebenstein et al., 2011).

- The systems considered typically have **non-linear dynamics**. Although, for direct torque control, the dynamics remain linear in the controls, raising the possibility of obtaining solutions within the discussed class of problems with the linearisable HJB equation. However, with the recent increase in interest of more complex actuation models, like variable stiffness actuation (e.g., Braun et al., 2012), the assumption of control linearity becomes increasingly invalid. The general problem is compounded when interaction with the environment is required, where in case of contacts, the dynamics become discontinuous and constrained.

- **Task specification** in terms of a cost function poses a non trivial problem.

A consequence of the first three points, is a general lack of analytical solutions. Before discussing approximate approaches prosed in the literature, let us examine two further questions concerning the type of SOC problems of interest in robotics, whether the problem is continuous or discrete in time and what objective one should consider.

While, as physical system, a robot is intrinsically of a continuous time nature, SOC problems associated with robotics are in fact often more naturally formed in discrete time. This is due to the policy space. Input forces or torques are continuous. However, they are not usually the control variables of the SOC problem. Rather, the controls are the inputs to the actuators, e.g. motor voltages. These inputs can commonly only be set with a certain, non-negligible, frequency, due to, e.g., bandwidth limitations. As such, from the point of view of a continuous problems, policies should be restricted to a class of piecewise constant functions. However as the choice points are discrete a more convenient approach is to form the marginal problem by integrating over the decision free intervals. It is with this in mind that we shall concentrate on discrete time SOC in this thesis.

With regards to objectives, we observe that robotics problems, in particular when considering manipulators, are goal oriented, e.g. pick up object X, move X to Y, etc., and as such episodic. This has lead to the finite horizon objective becoming the most widely adopted choice. We will therefore for most of the thesis also use this objective, examining this choice in more detail in Chapter 7.

To summarise, we consider the canonical SOC problem of interest in robotics to take the form

$$\operatorname*{argmin}_{\pi} \mathbb{E}_{\bar{\mathbf{x}},\bar{\mathbf{u}}|\pi,\mathbf{x}_0}\left[\mathcal{C}_T(\mathbf{x}_K) + \sum_{k=1}^{K-1} \mathcal{C}(\mathbf{x}_k, \mathbf{u}_k)\right],$$

where $\mathcal{C}_T$ and $\mathcal{C}$ are a given terminal and per-stage cost respectively and, for control frequency $1/\Delta$, $\mathbf{x}_k$ are the marginals at Δk of some stochastic continuous time non-linear dynamics under piecewise constant control $\mathbf{u}(s) = \mathbf{u}'$, with $s \in [\Delta k, \Delta(k+1)]$ and $\mathbf{u}' \sim \pi(\cdot|\mathbf{x}_k)$.

Example 2.3: Consider the standard task of reaching with the end effector of a n-link arm to a specific point, whilst minimising the square torque. The state of the plant consists of joint angles and velocities, i.e., $\mathbf{x} = (\mathbf{q}^T, \dot{\mathbf{q}}^T)^T \in \mathbb{R}^{2n}$ and has the continuous time dynamics

$$\mathrm{d}\mathbf{x} = \begin{bmatrix} \dot{\mathbf{q}} \\ -\mathbf{M}(\mathbf{q})^{-1}(\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) - \tau(\mathbf{x},\mathbf{u})) \end{bmatrix} + \mathbf{Q}\mathrm{d}\omega$$

where $\mathbf{M}$ is a inertia matrix, $\mathbf{C}$ and $\mathbf{g}$ are terms due to Coriolis and gravitational torques and $\tau(\cdot,\cdot)$ are some actuator dynamics. With the control frequency given by $1/\Delta$, the discrete time dynamics are obtained by integrating the above SDE over intervals Δ. The trajectory cost comprises a terminal cost, measuring the error in end effector coordinates, and a per step cost which measures the integrated square torque (but see Remark 2.2). Specifically,

$$\mathcal{C}(\bar{\mathbf{x}},\bar{\mathbf{u}}) = \alpha\|\phi^* - \phi(\mathbf{q})\| + \sum_{k=0}^{K} \tau(\mathbf{x}_k,\mathbf{u}_k)^2\Delta$$

where $\phi(\cdot)$ is the mapping from joint to end-effector coordinates, ϕ^* is the target and α is a parameter governing the trade off between reaching the target and torque minimisation.

Remark 2.2: The discretisation discussed here differs conceptually from discretisations necessarily performed when numerically solving a continuous time SOC problem. They are an inherent trait of the problem and considering the discrete time problem does not constitute an approximation. The latter on the other hand lead to an approximate solution of the problem which can be improved by reducing the discretisation step length.

The observant reader may notice that discretising the integrated cost rate term in the finite horizon objective, gives rise to a sum of terms of the form $\mathcal{C}(\mathbf{x}_k, \mathbf{x}_{k+1}, \mathbf{u}_k)$. In keeping with the formulations used throughout the literature, we shall assume – with the exception of Chapter 6 – the per-stage cost term to be a function of $(\mathbf{x}_k, \mathbf{u}_k)$ only, thus in the case of a state dependent cost rate committing to an approximation. However all presented results can be easily generalised to the correct term.

Remark 2.3: The discretisation has an interesting consequences in the context of collisions. In principle collisions, and more generally non-linearities in the dynamics, do not pose a significant complication as the discrete dynamics should be obtained by integration of the underlying time continuous dynamics over the time interval controls can not be changed. Thus only a practical problem arises, in that such non linearities have to be correctly accounted for when preforming numerical integration of the non linear, potentially switching, SDE. However, note that as a consequence of the discrete nature of the problem typical controllers can not avoid obstacles almost surely. Formally, provided the noise acts in all dimensions of the state space, thus ensuring the marginal state distribution after any time interval greater then 0 has support on all of $\mathcal{X}$, for any obstacle with none zero Lebesgue measure the collision probability is none zero. Importantly, this is a consequence of the time discrete control and thus an inherent property of the problem. Furthermore, note that discrete time control is not a necessary or sufficient condition to preclude almost sure obstacle avoidance, i.e., there exist continuous time problems with the same problem, e.g., where controls are limited, and there exist discrete time problem without this problem, e.g., where gains of feedback controllers are set at discrete time intervals.

2.3.1 Relevant Approaches

As discussed, the SOC problems arising in robotics are generally, due to the continuous state and control spaces and non linear dynamics, not analytically tractable. The challenge of forming approximate solutions is also increasingly compounded by the increasing complexity of robotic systems, and the associated increase in degrees of freedoms, i.e., problem dimensionality[5]. This makes solution of the problem by discretisation of the Bellman equation not viable, as such an approach exhibits exponential scaling in the dimensionality of the state-control space – the so called curse of dimensionality. Rather the most commonly employed approaches in robotics are based on iterative local dynamic programming. The underlying idea is to calculate a local approximation of the value function in a region of interest, typically given by a nominal trajectory, compute a locally optimal policy and then iteratively update the region of interest by application of the obtained local policy to the system. The curse of dimensionality is avoided, as locally the problem is likely to, approximately at least, take an analytically tractable form. However, due to their local nature, such methods can never guarantee global convergence. Instantiations of the generic idea differ mainly in the type of approximation used for the value function and method of (approximately) evaluating the required expectation. The classical instance is the well known Differential Dynamic Programming (DDP) algorithm due to Jacobson (1967) (stochastic variant by Theodorou et al., 2010b), which uses 2^{nd} order approximations around a nominal trajectory to both dynamics and the value function. More recently, Tassa and Todorov (2011) studied the generic form and suggested the use of a generalised linear model for the value function approximation and evaluation of the expectation by cubature methods. However it is the iterative Linear-Quadratic-Gaussian (iLQG) (Li, 2006) variant which has been most widely adopted for robotics applications (e.g. Todorov et al., 2010; Mitrovic et al., 2010b; Berret et al., 2011; Braun et al., 2012). It employs 2^{nd} order value function and 1^{st} order dynamics approximations, leading to re-

[5] n.b., our focus is on approaches with specific application to stochastic problems. Numerical solutions of the deterministic control problems, on the other hand, are commonly based on Pontryagin's minimum principle and exhibit better scaling w.r.t. problem dimensionality.

duced computational complexity compared to DDP, often with little effect on the obtained solution.

A recent comprehensive survey of RL in the context of robotics has been conducted by Kober and Peters (2012) and we restrict ourselves here to only a brief summary of the principle developments. In principle, any SOC method can be applied in the RL setting, by combining it with a model learning approach. For example, Mitrovic et al. (2010a) combine iLQG with a general framework for learning of dynamics. However, as dynamics learning is a hard problem in itself, so called *model free RL* approaches, which learn the value function or policy directly, are often preferable. Of these, policy based methods offer the advantage, that even relatively simple parametric policies can yield good results, while at the same time prior knowledge can be more directly incorporated into the policy parametrisation. In particular, so called, policy gradient algorithms, based on stochastic gradient descent in $\mathcal{J}$ have been widely used in the robotics community (e.g., Peters and Schaal, 2008; Kohl and Stone, 2004). A recent development has been the application of inference based formulations, discussion of which we defer to Chapter 3.

Chapter 3

Inference based Stochastic Optimal Control

Probabilistic inference, in particular in dynamical systems, faces comparable challenges to those identified for the SOC problem in Section 2.3, in particular, scaling to large dimensional problems over continuous spaces. Addressing these has lead to the development of a wide range of exact and approximate approaches within the field of Machine Learning which have no analogue in SOC. Examples include approximate message passing algorithm like Expectation Propagation (Minka, 2001) or factored representations of MDPs (e.g., Guestrin et al., 2003). In particular, with regards to inference problems in dynamical systems, increasingly sophisticated approximation methods have been proposed (e.g., Van Der Merwe, 2004; Hartikainen et al., 2011; Deisenroth et al., 2012). It is with the hope of transferring such ideas to SOC that we strive to find a dual formulation in probabilistic inference.

On a conceptual level, the idea that SOC could be framed as a type of inference problem can be made intuitive by framing the problem as inferring the controls which achieve a certain task, i.e., which give rise to a certain observed behaviour. However, formalising an exact dual relationship in the general case has proven to be difficult. Previous efforts in this area largely require certain assumptions on the form of the dynamics, cost or both, thus limiting their scope. We overcome these limitations with a novel general dual formulation for SOC in the discrete time setting. In the following, we first introduce our formulation, subsequently

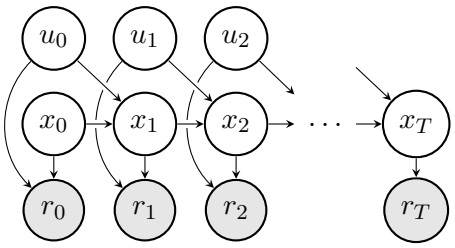


Figure 3.1: Graphical model for the posterior process in the proposed formulation.

providing an extensive review of previous work, highlighting it's relation to our approach.

Before proceeding, let us emphasise that our interest lies with correspondences between problems from the two domains, not the use of inference techniques to enhance individual steps within existing SOC algorithms. The latter is in itself an area of active research, particularly in the context of RL, where probabilistic methods can provide a principled way of accounting for the lack of knowledge about the dynamics and cost. Examples include the use of Gaussian Processes in dynamic programming by Deisenroth et al. (2009) and in policy gradient methods by Deisenroth and Rasmussen (2011) and Ghavamzadeh and Engel (2007), or the utilisation of importance sampling in policy gradient estimation by Tang and Abbeel (2010).

3.1 General Kullback-Leibler Divergence based Duality

We now turn to the formulation of a general relation between SOC and constrained minimisation of certain KL divergences. The Bayesian model underlying our approach is illustrated in Figure 3.1. In addition to the state and control variables of classical SOC, a binary dynamic random task variable r_k is introduced. The task likelihood is related to the classical cost by choosing,

$$P(r_k = 1|\mathbf{x}_k, \mathbf{u}_k) = \exp\{-\eta \mathcal{C}(\mathbf{x}_k, \mathbf{u}_k)\} \ , \tag{3.1}$$

where $\eta > 0$ is some constant in analogy with the inverse temperature of a Boltzmann distribution. Let us denote the trajectory distribution under some policy by[1] $q_\pi(\bar{\mathbf{x}}, \bar{\mathbf{u}})$, that is,

$$q_\pi(\bar{\mathbf{x}}, \bar{\mathbf{u}}) = P(\mathbf{x}_{1:K}, \mathbf{u}_{0:K}|x_0, \pi) = \prod_{k=0}^{K-1} P(\mathbf{x}_{k+1}|\mathbf{x}_k, \mathbf{u}_k)\pi(\mathbf{u}_k|\mathbf{x}_k) \tag{3.2}$$

Constructing a prior from some given policy π and assuming the artificial observations $r_{0\ldots K} = 1$, we form the posterior

$$\begin{aligned} p_\pi(\bar{\mathbf{x}}, \bar{\mathbf{u}}) &= P(\bar{\mathbf{x}}, \bar{\mathbf{u}}|\bar{r} = \mathbf{1}, \mathbf{x}_0, \pi) \\ &= \frac{1}{P(\bar{r} = \mathbf{1}|\mathbf{x}_0, \pi)} q_\pi(\bar{\mathbf{x}}, \bar{\mathbf{u}}) \prod_{k=0}^{K} \exp\{-\eta \mathcal{C}(\mathbf{x}_k, \mathbf{u}_k)\} \ . \end{aligned} \tag{3.3}$$

In the following, we will distinguish between the unknown control policy π and a prior policy π^0. We derive statements about the KL divergence[2] $\mathrm{KL}\left(q_\pi \| p_{\pi^0}\right)$ – intuitively we think of q_π as the *controlled process* which is *not* conditioned on costs (as defined in (3.2)), and p_{π^0} as the *posterior process*, which is conditioned on costs but generated via a potentially uninformed policy π^0 (as defined in (3.3)). The dual problem will be to find a control policy π such that the controlled process q_π matches the posterior process p_{π^0}. The following result establishes the basic relation between such a KL divergence and SOC

Proposition 3.1. *Let π^0 and π be an arbitrary stochastic policies, then the following identities hold*

$$\mathrm{KL}\left(q_\pi \| p_{\pi^0}\right) = Z + \eta \mathcal{J}(\pi) + \mathbb{E}_{q_\pi}\left[\mathrm{KL}\left(\pi \| \pi^0\right)\right] \tag{3.4a}$$

$$= Z + \eta \mathcal{J}(\pi) - \mathbb{E}_{q_\pi}\left[\log \pi^0(\bar{\mathbf{u}}|\bar{\mathbf{x}})\right] - \mathbb{E}_{q_\pi}\left[H(\pi)\right] \ , \tag{3.4b}$$

where $Z = \log P(\bar{r} = \mathbf{1}|\pi^0, \mathbf{x}_0)$, $H(\cdot)$ is the entropy[3], $\mathcal{J}(\pi)$ is the expected cost

$$\mathcal{J}(\pi) = \mathbb{E}_{q_\pi}\big[\sum_{k=0}^{K} \mathcal{C}(\mathbf{x}_k, \mathbf{u}_k)\big]$$

(cf. Section 2.1) and where we use the short-hand notation $\pi(\bar{\mathbf{u}}|\bar{\mathbf{x}}) = \prod_{k=0}^{K} \pi(\mathbf{u}_k|\mathbf{x}_k)$.

[1] n.b., we use the notation $\bar{a}$ to denote entire trajectories in a variable a

[2] n.b. a definition of the KL divergence and a summary of it's relevant properties is given in Appendix A

[3] n.b. the differential entropy, also known as *Shannon entropy* in the discrete case, is defined as $H(p) = -\int p(z) \log p(z)$

Proof. By definition of the KL divergence and using (3.3) and (3.1),

$$\begin{aligned}\mathrm{KL}\left(q_\pi \| p_{\pi^0}\right) &= \mathbb{E}_{q_\pi}\left[\log \frac{q_\pi(\bar{\mathbf{x}}, \bar{\mathbf{u}})}{P(\bar{r} = \mathbf{1}; \pi^0)^{-1} q_{\pi^0}(\bar{\mathbf{x}}, \bar{\mathbf{u}}) P(\bar{r} = \mathbf{1} | \bar{\mathbf{x}}, \bar{\mathbf{u}})}\right] \\ &= Z + \mathrm{KL}\left(q_\pi \| q_{\pi^0}\right) + \int_{\bar{\mathbf{x}}, \bar{\mathbf{u}}} q_\pi(\bar{\mathbf{x}}, \bar{\mathbf{u}}) \sum_{k=0}^{K} \eta \mathcal{C}(\mathbf{x}_t, \mathbf{u}_t)\end{aligned}$$

and

$$\begin{aligned}\mathrm{KL}\left(q_\pi \| q_{\pi^0}\right) &= \int_{\bar{\mathbf{x}}, \bar{\mathbf{u}}} q_\pi(\bar{\mathbf{x}}, \bar{\mathbf{u}}) \log \frac{\pi(\bar{\mathbf{u}}|\bar{\mathbf{x}})}{\pi^0(\bar{\mathbf{u}}|\bar{\mathbf{x}})} \\ &= \int_{\bar{\mathbf{x}}} q_\pi(\bar{\mathbf{x}}) \ \mathrm{KL}\left(\pi(\cdot|\bar{\mathbf{x}}) \| \pi^0(\cdot|\bar{\mathbf{x}})\right) \\ &= -\mathbb{E}_{q_\pi}\left[\log \pi^0(\bar{\mathbf{u}}|\bar{\mathbf{x}})\right] - \mathbb{E}_{q_\pi}\left[H(\pi(\cdot|\bar{\mathbf{x}}))\right]\end{aligned}$$

■

The presented identities are interesting in several respects: Equation (3.4a) tells us that finding an unconstrained policy $\pi^* = \operatorname{argmin}_\pi \mathrm{KL}\left(q_\pi \| p_{\pi^0}\right)$ is a compromise between minimized expected costs $\mathcal{J}(\pi)$ and choosing π similar to the prior policy π^0. In particular, in the limit $\eta \to \infty$ the expected cost term dominates and we retrieve a solution to the SOC problem. Further, choosing the prior policy π^0 to be uniform, the term $\mathbb{E}_{q_\pi}\left[\log \pi^0(\bar{\mathbf{u}}|\bar{\mathbf{x}})\right]$ in (3.4b) becomes constant and π^* is a compromise between minimized expected costs $\mathcal{J}(\pi)$ and *maximizing* the policy's entropy $\mathbb{E}_{q_\pi}\left[H(\pi)\right]$. This hints at a relation to risk-sensitive control, which we will discuss in more detail in Chapter 5.

Informally we may summarise the consequence of these identities with the following corollary.

Corollary 3.1. *Let π^0 be an arbitrary stochastic policy and $\mathcal{D}$ the set of deterministic policies, then the problem*

$$\pi^* = \operatorname*{argmin}_{\pi \in \mathcal{D}} \mathrm{KL}\left(q_\pi(\bar{\mathbf{x}}, \bar{\mathbf{u}}) \| p_{\pi^0}(\bar{\mathbf{x}}, \bar{\mathbf{u}})\right) \ , \tag{3.5}$$

is equivalent to the finite horizon stochastic optimal control problem with cost per stage

$$\hat{\mathcal{C}}(\mathbf{x}_k, \mathbf{u}_k) = \mathcal{C}(\mathbf{x}_k, \mathbf{u}_k) - \frac{1}{\eta} \log \pi^0(\mathbf{u}_k|\mathbf{x}_k) \ .$$

While in the case where π^0 is a distribution w.r.t. the counting measure, e.g., $\mathcal{U}$ is finite, this corollary is formally correct and follows directly from Proposition 3.1,

it is only an informal guideline in the general case. The difficulty arises from the fact that deterministic policies have distributions w.r.t. the counting measure which is not absolutely continuous w.r.t. the Lebesgue measure w.r.t. which π^0 will be formed. As such the KL divergence is formally not defined in this case (n.b. it is often taken to be defined but ∞). However considering the identity (3.4b), we may observe that a limiting interpretation of Corollary 3.1 is possible. In particular observe that we may replace $\mathcal{D}$ with a sequence of sets $\mathcal{D}_\epsilon$, each comprising all stochastic policies of entropy ϵ, i.e., $\mathcal{D}_\epsilon = \{\pi \in \Pi; H(\pi(\cdot|\mathbf{x}) = \epsilon \forall \mathbf{x} \in \mathcal{X}\}$ with Π the set of all stochastic policies. Then as $\epsilon \to 0$, we approach the set of deterministic policies while

$$\mathrm{KL}\left(q_\pi(\bar{\mathbf{x}}, \bar{\mathbf{u}}) \| p_{\pi^0}(\bar{\mathbf{x}}, \bar{\mathbf{u}})\right) \to Z + \mathbb{E}_{q_\pi(\bar{\mathbf{x}}, \bar{\mathbf{u}})}\left[\sum_{k=0}^{K} \hat{\mathcal{C}}(\mathbf{x}_k, \mathbf{u}_k)\right] .$$

We consider Corollary 3.1 as an informal statement of this limiting relation. Throughout this thesis we will refer to Corollary 3.1 without further explicitly acknowledging this complication, finding it a convenient short hand to unify various formulations and an informal guide to suggest novel approaches. However at no point will we use Corollary 3.1 directly to solve a SOC problem, but shall, as we will see, always take care to relax the restriction to deterministic policies, thus returning to a well formed expression.

Note that as an immediate consequence we may recover any given SOC problem with cost $\mathcal{C}$ by choosing $\pi^0(\cdot|x)$ to be the uniform distribution over $\mathcal{U}$[4].

Corollary 3.1 and (3.5), provide the focal point of much of the remained of this thesis and it is these we refer to when we talk of *our formulation* or the *general duality*. Let us therefore briefly emphasise the implication of these results. Corollary 3.1 tells us that SOC can be interpreted as trying to emulate the uncontrolled, but task conditioned, posterior process (3.3) with a controlled process (3.2). Importantly however not with any feasible controlled processes, but one based on deterministic controls. The consequence is that the minimisation problem in (3.5) is constrained and does in general not admit a closed form solution. Although it does, as we shall see in Chapter 4 and Chapter 5, provide a novel perspective on the problem which gives rise to new iterative solutions.

[4]n.b., formally we should require $\mathcal{U}$ to be finite or bounded, although see also Remark 3.1

However, before we turn to the problem of how to utilise Corollary 3.1, we shall in the remainder of this chapter review alternative inference based formulations of SOC and their relation to Corollary 3.1.

Remark 3.1 (Control Cost Prior Policy)**:** In the special case where the per-stage cost takes the form

$$\mathcal{C}(\mathbf{x}_k, \mathbf{u}_k) = \mathcal{C}^x(\mathbf{x}_k) + \mathcal{C}^u(\mathbf{u}_k)$$

we may equivalently consider a model with $P(r_k = 1|\mathbf{x}_k, \mathbf{u}_k) = \exp\{-\mathcal{C}^x(\mathbf{x}_k)\}$ and a prior policy $\pi^0(\mathbf{u}_k|\mathbf{x}_k) \propto \exp\{-\mathcal{C}^u(\mathbf{u}_k)\}$. For example in the common case of a quadratic control cost, such a π^0 is a zero mean Gaussian with variance given by the inverse control cost weight. This allows us, in particular in the case of unbounded control spaces, to avoid the potential problem of the uniform distribution in Corollary 3.1.

3.2 Review of Inference-Control Dualities

We now review previously observed relations between inference problems and SOC. These can be broadly divided into three classes, based on the general form of the relation. Specifically, *Maximum a Posteriori Estimation Dualities*, *Filtering Dualities* and *KL Divergence Dualities*. We will discuss each in turn, highlighting relevant relations to our formulation presented above.

3.2.1 Maximum a Posteriori Estimation Dualities

The process of Bayesian estimation does seemingly lack the nature of an optimisation problem[5] central to SOC, thus making it hard to establish an explicit connection between the two. The problem of finding the modes of the posterior, i.e., MAP estimations, on the other hand, is directly formulated as an optimisation problem, making it easier to form a conceptual connection.

Informally, the principle underlying such approaches is the formulation of a distribution over policies such that

[5]Although it is worth noting that one can derive the posterior from minimization of a variational free energy, a perspective which is commonly employed in approximate inference approaches (e.g. Yedidia et al., 2005).

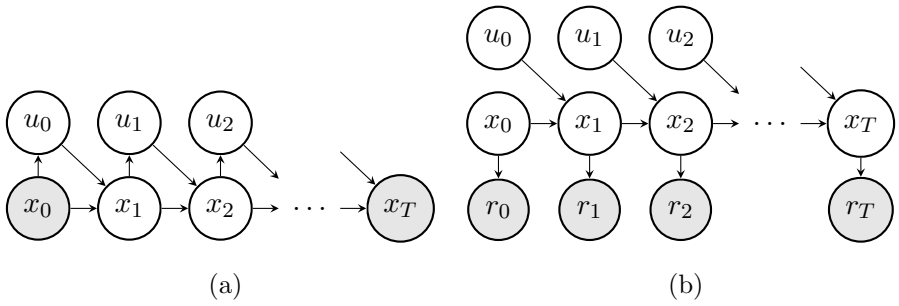


Figure 3.2: Graphical models for MAP estimation based formulations which lack an exact correspondence to SOC. **(a)** The optimistic inference control model. **(b)** The Approximate Inference Control model.

$$\underset{\pi}{\operatorname{argmin}} - \log P(\pi|r) = \underset{\pi}{\operatorname{argmin}} \mathcal{J}(\pi) \ . \tag{3.6}$$

Although principally coinciding minima are sufficient for duality, the use of solvers which only guarantee local optimality makes stronger equivalences desirable, and has led to most approaches aiming at either coinciding modes and local minima or even equivalence up to an additive constant. The constructed distribution often has a natural interpretation as a posterior distribution under conditioning on the task – indicated here by the conditioning on some variable r –, although this is not necessary. In fact many approaches are based on minimisation of a (marginal) likelihood $P(r|\pi)$, which is obviously equivalent to MAP with an improper uniform prior on π.

Formulations without Exact Duality

A direct instantiation of the generic form (3.6) for sequential finite horizon control problems was proposed by Attias (2003) in the case of MDPs and independently, under the name *optimistic inference control*, by Raiko and Tornio (2005), for problems with continuous state spaces. The formulation is, as illustrated in Figure 3.2(a), based on direct conditioning of a trajectory distribution under some fixed policy on the initial and a desired final state. In terms of (3.6), we identify

$\bar{\mathbf{u}} \rightsquigarrow \pi$ and $(\mathbf{x}_0, \mathbf{x}_K) \rightsquigarrow r$, so that the problem takes the form

$$\underset{\bar{\mathbf{u}}}{\operatorname{argmin}} - \log P(\bar{\mathbf{u}}|\mathbf{x}_0, \mathbf{x}_K) = \underset{\bar{\mathbf{u}}}{\operatorname{argmin}} - \log \int_{\mathbf{x}_{1:K-1}} q_{\pi^0}(\mathbf{x}_{1:K-1}, \bar{\mathbf{u}}|\mathbf{x}_0, \mathbf{x}_K) \qquad (3.7)$$

where we denote the trajectory distribution under a policy π by q_π. It is solved by both Attias and Raiko and Tornio through application of a smoothing algorithm and subsequent explicit minimisation. Although Attias observes that the resulting controls maximise the sum of the log probabilities of reaching the target and controls, a direct connection with general SOC is not made. An extension of this work, aiming to take a given cost function into account, is Approximate Inference Control (AICO) by Toussaint (2009b). Here, rather then conditioning the state directly, an auxiliary binary random variable is conditioned on artificial observations, as illustrated in Figure 3.2(b). The connection to a given cost per-stage, which is assumed to decompose into a state and control dependent part, is made by

$$P(r_k = 1|\mathbf{x}_k) = \exp\{-\mathcal{C}^x(\mathbf{x}_k)\} \quad \text{and} \quad \pi^0(\mathbf{u}_k) \propto \exp\{-\mathcal{C}^u(\mathbf{u}_k)\}$$

and the problem takes an equivalent form to (3.7). Toussaint notes that, using Jensen's inequality, it is easy to show that the minimised objective is a lower bound on the expected cost, and concludes that few conclusions about the relation to SOC can be made.

The model underlying our formulation is closely related to the AICO model. Specifically, the latter arises as a special case for $\eta = 1$ and under the assumptions of additive state and control costs (also cf. Remark 3.1). However, obviously the two approaches differ in their use of the model. Our approach can be seen as a clarification of the relation of AICO to SOC, which was missing from the original work. We further comment on relations of particular algorithms arising from Corollary 3.1 and AICO in Chapter 5.

Non-sequential Problems

Initial work by Sabes and Jordan (1996) and Dayan and Hinton (1997) considered a non sequential decision problems – equivalent to a discrete time SOC problem with horizon $K = 1$ – and were to our knowledge the first to propose the

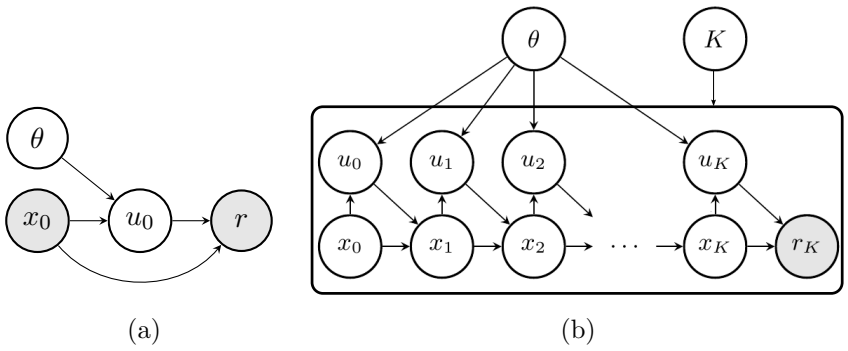


Figure 3.3: Graphical models for MAP estimation based formulations with exact correspondence to SOC. **(a)** The model implicitly underlying the formulation in the non-sequential case. **(b)** The mixture model due to Toussaint and Storkey (2006) for the sequential problem. The variable K is integrated out yielding a mixture over Markov chains of various length.

application of EM algorithms to control problems. Peters and Schaal (2007) subsequently applied these ideas to robot control, considering sequential problems with immediate reward – equivalent to a discounted cost problem with $\gamma = 0$. In both these cases the formulation can be reduced to minimisation of the negative log of

$$P(\mathbf{u}|\mathbf{x}, \theta) \propto f(\mathbb{E}_{\mathbf{y}|\mathbf{x},\mathbf{u}}[\mathcal{C}(\mathbf{y}, \mathbf{u})])\pi(\mathbf{u}|\mathbf{x}, \theta) \ ,$$

with some $f : \mathbb{R}_+ \to [0, 1]$, which is strictly monotonically decreasing, e.g., $f(\cdot) = \exp\{-(\cdot)\}$. Note that, although neither of the cited works offers such an interpretation, we may consider $f(\mathcal{C}(\cdot))$ as the likelihood of the observation of a binary auxiliary variable r. Thus, the model underlying these approaches takes the form Figure 3.3(a).

Sequential Problems with Bounded Costs - Mixture Model

The work of Toussaint and Storkey (2006) (see also Toussaint et al., 2010b) constitutes an extension of the above to sequential problems under the assumption of bounded costs. The construction combines ideas from several of the aforementioned approaches and is centred around the posterior distribution over trajecto-

ries of a given length

$$P(\mathbf{x}_{0:k}, \mathbf{u}_{0:k}|r_k = 1, k, \theta) = q_{\pi_\theta}(\mathbf{x}_{0:k}, \mathbf{u}_{0:k}) \underbrace{P(r_k = 1|\mathbf{x}_k, \mathbf{u}_k)}_{:=1-\frac{\mathcal{C}(\mathbf{x}_k,\mathbf{u}_k)}{\max_{(\mathbf{x},\mathbf{u})}\mathcal{C}(\mathbf{x},\mathbf{u})}} , \tag{3.8}$$

where we again used q_π to denote the trajectory distribution under a policy. Note that, in keeping with the non-sequential approaches and in contrast to the methods of Attias et al., an auxiliary variable with pseudo observations is utilised, rather then conditioning the final state directly. Equivalence, up to an additive constant, of the negative log likelihood and expected cost for different problem classes is then obtained by taking a mixture of (3.8) according to a prior over k. For example, $P(k) = \mathcal{U}([0, K])$ yields the expected cost for a finite horizon problem, while $P(k) = (1 - \gamma)\gamma^k$ leads to equivalence in the discounted infinite horizon case. The complete model is illustrated in Figure 3.3(b).

Sequential Problems with General Costs

Barber and Furmston (2009) suggest an alternative to the mixture model, which seemingly does not require bounded per step objectives. The observation made is that, in the case of positive rewards $\mathcal{R}(\cdot, \cdot)$ – rather then costs – the expected reward can be seen as the normalisation constant of the distribution

$$P(\mathbf{x}_{0:k}, \mathbf{u}_{0:k}, k|\theta) = \frac{\mathcal{R}(\mathbf{x}_k, \mathbf{u}_k, k) q_{\pi_\theta}(\mathbf{x}_{0:k}, \mathbf{u}_{0:k})}{\mathbb{E}_{q_{\pi_\theta}(\mathbf{x}_{0:k},\mathbf{u}_{0:k})}\left[\sum_{\tau=0}^{k} \mathcal{R}(\mathbf{x}_\tau, \mathbf{u}_\tau, \tau)\right]} . \tag{3.9}$$

This is obviously equivalent to the aforementioned mixture model with an improper uniform prior on k and $P(r_k = 1|\mathbf{x}_k, \mathbf{u}_k) = \mathcal{R}(x_k, u_k, k)$ and, reassuringly, equivalent EM updates are obtained. However, Barber and Furmston fail to note that (3.9) is only a proper distribution for θ such that $\mathcal{J}(\pi_\theta) < \infty$. In this context, we may see the bound required by Toussaint and Storkey as a sufficient condition such that this is the case for all θ. The implication for Barber and Furmston is that, formally the policy space should be restricted to $\Theta' = \{\theta \in \Theta; \mathcal{J}(\pi_\theta) < \infty\}$. Such a restriction does not affect the optimal solution, as $\mathcal{J}(\pi^*) < \infty$. In practise, relevant approaches (Barber and Furmston, 2009; Furmston and Barber, 2010; Kober and Peters, 2009; Hoffman et al., 2009a) do not guard against improper distributions, merely assuming θ remains in the restricted set.

Practical Applications

Irrespective of it's formal interpretation, AICO has been successfully utilised for trajectory planning in a variety of robotics problems (Rawlik et al., 2010; Toussaint et al., 2010a; Zarubin et al., 2012; Ivan et al., 2013).

Since their introduction in the model based setting, both approaches for sequential problems have given rise to RL algorithms. These generally focus on stochastic variants of the EM algorithm, in particular approximation of the E-Step by means of an empirical distribution. Vlassis and Toussaint (2009) initially propose the application of a stochastic variant of EM, with subsequent work (Vlassis et al., 2009) suggesting an improved sampling scheme. Hoffman et al. (2009a), concentrating on the continuous case, independently obtain similar results. Furthermore, both the Reward Weighted Regression (RWR) and Policy learning by Weighted Exploration with the Returns (PoWER) algorithms (both Kober and Peters, 2009) lead to equivalent updates for particular choices of policies. Empirical results suggest that such methods yield faster convergence than policy gradient based approaches, although generally little or no improvement in the final policy is observed. As an alternative to stochastic EM, Furmston and Barber (2010) follow a model based RL approach in the case of MDPs. Here, the suggestion is to use standard EM with the transition model given as a posterior over models conditioned on the observed data. This approach is shown to further improve on the convergence speed of the stochastic EM methods.

3.2.2 Filtering Dualities

Going back to the observation of the duality between Linear-Quadratic Regulator and Linear-Gaussian filtering by Kalman (1960) – the so called *Kalman Duality* – these approaches pre-date the previously discussed MAP estimation based formulations. They have however only recently found application in robotics, with their generalisation to a class of non-linear problems. These dualities are formed by considering the problem of estimating the state posterior of an unobserved dynamical process based on noisy observations. In particular, with $\mathbf{x}_{0:K}$ the unobserved states and $\mathbf{y}_{0:K}$ the observations, the posterior $P(\mathbf{x}_k|\mathbf{y}_{k:K})$ is called the backward filtering distribution. It is this distribution which can be related to the

value function of a specific SOC problem by

$$P(\mathbf{x}_k|\mathbf{y}_{k:K}) \propto \exp\{-\mathcal{V}_k(\mathbf{x}_k)\} \ . \tag{3.10}$$

A perhaps unsurprising fact, considering that (3.10) strongly suggests the existence of an HJB equation with resolved minimisation, is that for problems with a filtering dual closed form expressions for the optimal controls in terms of $\mathcal{V}$ exist. Thus, existence of such a dual, reduces the control problem to a Bayesian inference problem.

3.2.2.1 Kalman Duality

To illustrate the idea of such filtering dualities, let us begin with the classical result, first observed by Kalman, for the LQG case[6]. Consider the partially observable linear Gaussian Markov process

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{F}^x\mathbf{x}_k + \eta_x \quad \eta_x \sim \mathcal{N}(0, \mathbf{S}_{\eta_x}) \ , \\ \mathbf{y}_k &= \mathbf{G}\mathbf{x}_k + \eta_y \quad \eta_y \sim \mathcal{N}(0, \mathbf{S}_{\eta_y}) \ , \end{aligned}$$

with $x_0 \sim \mathcal{N}[0, 0]$. As random variables arising from linear transformations and combinations of Gaussian random variables are themselves Gaussian we conclude that the joint $P(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ is Gaussian and hence $\mathbf{x}_k|\mathbf{y}_{k:K} \sim \mathcal{N}[\mathbf{p}_k, \mathbf{P}_k]$. Furthermore, as in general,

$$P(\mathbf{x}_k|\mathbf{y}_{k:k} = 0) \propto \int_{\mathbf{x}_k} P(\mathbf{x}_{k+1}|\mathbf{x}_k)P(\mathbf{y}_k = 0|\mathbf{x}_k)P(\mathbf{x}_{k+1}|\mathbf{y}_{k+1:K} = 0) \ ,$$

the parameters $\mathbf{p}_k, \mathbf{P}_k$ can be computed recursively using standard Gaussian identities. For the specific case of an observation sequence $\mathbf{y}_{0:K} = 0$ we obtain the solution $\mathbf{p}_{0:K} = 0$ and

$$\mathbf{P}_{k+1} = \mathbf{G}^T\mathbf{S}_{\eta_y}^{-1}\mathbf{G} + \mathbf{F}^{xT}\mathbf{P}_{k+1}\mathbf{F}^x - \mathbf{F}^{xT}\mathbf{P}_{k+1}^T(\mathbf{P}_{k+1} + \mathbf{S}_{\eta_x}^{-1})^{-1}\mathbf{P}_{k+1}\mathbf{F}^x \ .$$

Recall from Example 2.1 that, the solution for the value function in a LQG problem with $\mathbf{f} = 0$, $\mathbf{c}^x = 0$ and $\mathbf{c}^u = 0$ takes the form $\mathcal{V}_k(\mathbf{x}) = \mathbf{x}^T\mathbf{V}_k\mathbf{x}$, with $\mathbf{V}_k$

[6] n.b., in (Kalman, 1960) the duality is presented for the Linear-Quadratic Regulator case, however as the solutions of the LQ Regulator and LQG problems coincide we choose to present it here in the context of the latter.

given by (2.5). Applying Woodbury identities, it is straightforward to bring (2.5) into the form

$$\mathbf{V}_k = \mathbf{C}^x + \mathbf{F}^{xT}\mathbf{V}_{k+1}\mathbf{F}^x - \mathbf{F}^{xT}\mathbf{V}_{k+1}^T(\mathbf{V}_{k+1} + (\mathbf{F}^u\mathbf{C}^{u-1}\mathbf{F}^{uT})^{-1})^{-1}\mathbf{V}_{k+1}\mathbf{F}^x \ .$$

It now becomes obvious that the equations are equivalent under the substitution summarised in the following table

Linear Quadratic Gaussian	Linear Gaussian Filter
$\mathbf{V}$	$\mathbf{P}$
$\mathbf{F}^x$	$\mathbf{F}^x$
$\mathbf{C}^x$	$\mathbf{G}^T\mathbf{S}_{\eta_y}^{-1}\mathbf{G}$
$\mathbf{F}^u(\mathbf{C}^u)^{-1}\mathbf{F}^{uT}$	$\mathbf{S}_{\eta_x}$
k	k

Remark 3.2: In the interest of a concise presentation we have limited ourselves to the case of an LQG problem with $\mathbf{f} = 0$, $\mathbf{c}^x = 0$, $\mathbf{c}^u = 0$ however the duality extends naturally to the more general case.

Remark 3.3: Derivations of the Kalman filter commonly found in the literature yield a representation of $\mathbf{x}_k|\mathbf{y}_{0:k-1}$ in standard form, i.e., of the forward filtering distribution in terms of a mean $\hat{x}_k$ and a covariance Σ (see, e.g., Kalman, 1960; Stengel, 1986). The backward-canonical form provided here is referred to as the *backwards Kalman information filter*. We choose this representation as it naturally leads to the generalised form in the non-linear case discussed below. In particular, note that the information filter form seems to provide a more intuitive duality compared to the standard form which is given by the following table

Linear Quadratic Gaussian	Linear Gaussian Filter
$\mathbf{V}$	Σ
$\mathbf{F}^x$	$\mathbf{F}^{xT}$
$\mathbf{F}^u$	$\mathbf{G}^T$
$\mathbf{C}^x$	$\mathbf{S}_{\eta_x}$
$\mathbf{C}^u$	$\mathbf{S}_{\eta_y}$
k	$K - k$

Thus the key conceptual correspondences can be summarised as

Linear Quadratic Gaussian	Information Filter	Standard Filter
Control	Process Noise	Measurements
State Cost	Measurements	Process Noise

In the information form, similar to the MAP formulations in Section 3.2.1, we may think of conditioning a state process on the task.

3.2.2.2 Generalised Kalman Duality

While undoubtedly intriguing from the mathematical point of view, the question arising with regards to practical applications, is whether such a duality can be extended beyond the case of (discrete time) LQG to more interesting non-linear problems. The similarity of the relation between process noise and control cost in the Kalman Duality and the condition encountered in the context of linearisation of the HJB equation of control LQ problems (cf. Section 2.2.2), suggests further examination of the latter case.

Let us, as previously, begin with the filtering problem. Consider the non-linear continuous time partially observed system given by

$$\begin{aligned} \mathrm{d}\mathbf{x} &= f(\mathbf{x})\mathrm{d}t + \mathrm{d}\eta_x \quad \mathbb{E}\left[\eta_x^T \eta_x\right] = \mathbf{S}_{\eta_x}\mathrm{d}t \\ \mathrm{d}\mathbf{y} &= g(\mathbf{x})\mathrm{d}t + \mathrm{d}\eta_y \quad \mathbb{E}\left[\eta_y^T \eta_y\right] = \mathbf{S}_{\eta_y}\mathrm{d}t \ . \end{aligned}$$

Now, let $\tilde{p}$ denote the un-normalised backward filtering distribution, i.e., $\tilde{p}(\mathbf{x}(t)) \propto P(\mathbf{x}(t)|\mathbf{y}(\cdot > t), \mathbf{x}(0))$. It is known (see, e.g., Bensoussan, 1992) that $\tilde{p}$ satisfies the following stochastic partial differential equation – the so called backward Duncan-Mortensen-Zakai equation –

$$-\mathrm{d}\tilde{p} = f^T \nabla \tilde{p}\mathrm{d}t + \frac{1}{2}\operatorname{Tr}\left(\mathbf{S}_{\eta_x}\nabla^2\tilde{p}\right)\mathrm{d}t + \tilde{p}g^T\mathrm{d}\mathbf{y}$$

Assuming the observations are $\mathbf{y}(\cdot) = 0$ and following Krishnamurthy and Elliott (2002), this can be transformed (cf. Appendix B) into the partial differential equation

$$-\partial_t\tilde{p} = -\frac{1}{2}\|g\|^2_{\mathbf{S}_{\eta_\mathbf{y}}}\tilde{p} + (\nabla\tilde{p})^T f + \frac{1}{2}\operatorname{Tr}\left(\mathbf{S}_{\eta_x}\nabla^2\tilde{p}\right) \tag{3.11}$$

which we recognise as an instance of the linear form of the HJB equation (2.7), with the equivalence summarised by the the following table

Hamilton-Jacobi-Bellman	Duncan-Mortensen-Zakai
$\exp\{-\mathcal{V}(\cdot)\}$	$\tilde{p}(\cdot)$
f	f
$\frac{1}{\lambda}\mathcal{C}(\cdot)$	$\frac{1}{2}\|g\|^2_{\mathbf{S}_{\eta_y}}$
$\mathbf{F}^u(\mathbf{C}^u)^{-1}\mathbf{F}^{uT}$	$\mathbf{S}_{\eta_x}$
t	t

We may note the similarity of the substitutions made to those in the previously discussed discrete time case.

Remark 3.4: Alternative derivations of the equivalence of the linear HJB equation and it's filter representation exist. To our knowledge, the first concrete mention of this duality is due to Mitter and Newton (2000), where a derivation based on the variational duality between Relative Entropy and Free Energy is provided. Alternatively, Kappen (2011) (although Theodorou (2011) provides a more thorough treatment) gives a derivation of an essentially equivalent result using the Feyman-Kac Lemmas, though a connection to the Kalman duality is not directly made.

3.2.2.3 Practical Applications

Since the popularisation of the associated class of SOC problems under the name of *Path Integral Control*, application of inference methods based on the generalised Kalman duality has increased. Kappen (2005) proposes the use of a direct Monte Carlo (MC) sampling solution, as well as an importance sampling based approach, with a proposal distribution based on a variational approximation. Theodorou et al. (2009) consider the application to robotic control and, due to the poor sample efficiency of a direct MC approach, proposes an approximation based on trajectory re-use. As an alternative to MC approaches, Mensink et al. (2010) (see also Broek et al., 2011) suggest Expectation Propagation, considering, amongst others, the problem of a robotic manipulator moving amongst obstacles.

In the context of RL, arguably the most successful application, has been the Policy Improvement by Path Integral Control (PI^2) algorithm proposed by Theodorou et al. (2010a). Rather then solving RL problems of appropriate form

directly, Theodorou et al. suggest to iteratively form local problems, which can be approximately solved using samples. The algorithm has been applied to control of various robotic systems (e.g., Kalakrishnan et al., 2012) and results indicate it outperforms both gradient based methods and the EM based algorithm discussed in Section 3.2.1.

3.2.2.4 Relation to General Duality

There is an obvious conceptual connection between filtering dualities and the duality of Corollary 3.1. As in the latter case, the solution is obtained by connecting the controlled process with and uncontrolled one (also cf. Remark 3.1). This observation can be made more explicit and allows us to interpret the Kalman dualities from the point of view of the proposed divergence.

Let us begin with the Kalman duality in the discrete time case, that is with an LQG problem. Using Bayes rule we may, in general, write $p_{\pi^0}(\bar{\mathbf{x}}, \bar{\mathbf{u}}) = P(\bar{\mathbf{x}}|r_k = 1, \pi^0)P(\bar{\mathbf{u}}|\bar{\mathbf{x}}, \pi^0)$. In the specific case under consideration, this factorisation takes the form,

$$p_{\pi^0}(\bar{\mathbf{x}}, \bar{\mathbf{u}}) = Z^{-1} + \prod_{k=0}^{K-1} \exp\{\mathbf{m}_k^T \mathbf{Q}^{-1} \mathbf{F}^u \mathbf{u}_k - \mathbf{u}_k^T \mathbf{C}^u \mathbf{u}_k)\} \cdot \underbrace{\prod_{k=0}^{K-1} \exp\{\mathbf{x}_{k+1}^T \mathbf{C}^x \mathbf{x}_{k+1}\} P(\mathbf{x}_{k+1}|\mathbf{x}_k, \mathbf{u}_k = 0)}_{=\nu(\bar{\mathbf{x}})} ,$$

where $\mathbf{m}_k = \mathbf{x}_{k+1} - \mathbf{f} - \mathbf{F}^x \mathbf{x}_k$ and Z is a constant. Thus, the KL divergence of Corollary 3.1 can be written as

$$\mathrm{KL}\left(q_\pi(\bar{\mathbf{x}}, \bar{\mathbf{u}}) \| p_{\pi^0}(\bar{\mathbf{x}}, \bar{\mathbf{u}})\right) = \log Z + \mathrm{KL}\left(q_\pi(\bar{\mathbf{x}}) \| \nu(\bar{\mathbf{x}})\right) - \mathbb{E}_{q_\pi}\left[\sum_{k=0}^{K} \mathbf{m}_k^T \mathbf{Q}^{-1} \mathbf{F}^u \mathbf{u}_k - \frac{1}{2} \mathbf{u}_k^T \mathbf{C}^u \mathbf{u}_k)\right] . \quad (3.12)$$

Now, since $\mathbf{m}_k = \mathbf{F}^u \mathbf{u}_k$, the final term vanishes under the condition $\mathbf{C}^u = \mathbf{B}^T \mathbf{Q}^{-1} \mathbf{B}$. Examining the table in Section 3.2.2.1, we may note that furthermore, under this condition, the backward process of ν corresponds exactly to the filtering process of the Kalman duality. This result can be extended by observing

that it is sufficient for the problem to be control LQ to allow for the factorisation and reduction to $\mathrm{KL}\left(q_\pi(\bar{\mathbf{x}})\|\nu(\bar{\mathbf{x}})\right)$.

As a special case, we may consider the discretisation of a continuous time control LQ problem in the limit of the discretisation step length going to zero. Such a limit is well defined, as the two processes will have the same diffusion term. Interestingly, adapting results by Todorov (2009b), one can show that under mild conditions, unlike in the discrete time case, in the continuous time limit there exists a policy which matches q_π and ν exactly. Hence, ν is the state trajectory distribution under the optimal policy, making this case a continuous state and control set generalisation of KL control.

In summary, the proposed formulation provides us with a novel view of the Kalman duality and it's generalisation. Specifically, rather then seeing the filtering distributions as a mathematical means to obtaining the value function, we may interpret the filtering processes as defining trajectory distributions the optimal control process attempts to match. Finally, this alternative view leads to the extension of the discrete time dualities beyond the LQG case, to the control LQ setting - a novel result to the best of our knowledge.

In fact, the new perspective allows for further generalisation, by dropping the requirement of $\mathbf{C}^u = \mathbf{B}^T\mathbf{Q}^{-1}\mathbf{B}$. The terms in (3.12) do no longer cancle, however, using the inverse substitution[7] $\mathbf{u}_k = (\mathbf{F}^u)^{\#}\mathbf{m}_k$, we have

$$\mathrm{KL}\left(q_\pi(\bar{\mathbf{x}},\bar{\mathbf{u}})\|p_{\pi^0}(\bar{\mathbf{x}},\bar{\mathbf{u}})\right) = \mathrm{KL}\left(q_\pi(\bar{\mathbf{x}})\|\nu'(\bar{\mathbf{x}})\right) - \log Z$$

Here, the new filtering processes ν', given by

$$\nu'(\bar{\mathbf{x}}) = \nu(\bar{\mathbf{x}}) \prod_{k=0}^{K-1} \exp\{\mathbf{m}_k^T\mathbf{Q}^{-1}\mathbf{m}_k - \frac{1}{2}\mathbf{m}_k^T(\mathbf{F}^u)^{\#T}\mathbf{C}^u(\mathbf{F}^u)^{\#}\mathbf{m}_k\}$$

arises from augmentation with a new task likelihood – or observation likelihood, depending on the point of view – which, unlike the likelihood in ν, is a function of state pairs $(\mathbf{x}_k, \mathbf{x}_{k+1})$.

[7] n.b., $\mathbf{A}^{\#}$ is the Moore-Penrose pseudoinverse of $\mathbf{A}$

3.2.3 Kullback-Leibler Divergence based Duality for MDPs

The idea underlying the general duality we propose in Corollary 3.1, can be summarised as finding a relation of the form

$$\mathrm{KL}\left(P(z|\pi)\|P(z|r)\right) = \mathcal{J}(\pi) \ . \tag{3.13}$$

As we have seen, in the general case, we were required to impose constraints on the class of policies considered. However, Todorov (2006a) (see also Todorov, 2009b) and subsequently Kappen et al. (2009) suggest a class of MDPs, so called *linearly solvable MDPs* for which a duality of the unconstrained form (3.13) exists. Let us first briefly recall the framework in the form given by Kappen et al., before discussing it's limitations.

Choose some free dynamics $\nu_0(\mathbf{x}_{k+1}|\mathbf{x}_k)$ and let the cost be given as

$$\mathcal{C}_T(\bar{\mathbf{x}}) = \ell(\bar{\mathbf{x}}) + \log \frac{\nu(\bar{\mathbf{x}})}{\nu_0(\bar{\mathbf{x}})} \ ,$$

where $\nu(\mathbf{x}_{k+1}|\mathbf{x}_k)$ is the controlled process under some policy. Then

$$\mathbb{E}_\nu\left[\mathcal{C}_T(\bar{\mathbf{x}})\right] = \mathrm{KL}\left(\nu(\bar{\mathbf{x}})\|\nu_0(\bar{\mathbf{x}})\exp\{-\ell(\bar{\mathbf{x}})\}\right) \ , \tag{3.14}$$

is minimised w.r.t. ν by

$$\nu(\mathbf{x}_{1:K}|\mathbf{x}_0) = \frac{1}{Z(\mathbf{x}_0)}\exp\{-\ell(\mathbf{x}_{1:K})\}\nu_0(\mathbf{x}_{1:K}|\mathbf{x}_0) \ , \tag{3.15}$$

and one concludes that the optimal control is given by $\nu(\mathbf{x}_{k+1}|\mathbf{x}_k)$, where the implied meaning is that $\nu(\mathbf{x}_{k+1}|\mathbf{x}_k)$ is the trajectory distribution under the optimal policy.

Although (3.15) gives a process which minimises (3.14), it is not obvious how to compute the actual controls $\mathbf{u}_k$. Specifically when given a model of the dynamics, i.e., $P(\mathbf{x}_{k+1}|\mathbf{x}_k, \mathbf{u}_k)$, and having chosen some ν_0, a non trivial, yet implicitly made, assumption is that there exists a policy implementing the required transitions $\nu(\mathbf{x}_{k+1}|\mathbf{x}_k)$, i.e., $\exists\pi$ s.t.

$$\mathrm{KL}\left(\int_{\mathbf{u}_k} P(\mathbf{x}_{k+1}|\mathbf{x}_k, \mathbf{u}_k)\pi(\mathbf{u}_k|\mathbf{x}_k)\|\nu(\mathbf{x}_{k+1}|\mathbf{x}_k)\right) = 0. \tag{3.16}$$

However, in general, such a π may not exist. This is made very explicit by Todorov (2009b) for the MDP case, who acknowledges that the method is only applicable

if the dynamics are fully controllable, in the sense that $P(\mathbf{x}_{k+1}|\mathbf{x}_k, \mathbf{u}_k)$ can be brought into any required form by the controls. Although at the same time, it is suggested that solutions to classical problems can be obtained by continuous embedding of the discrete MDP, such an approach has several drawbacks. For one, it requires solving a continuous problem even for cases which could have been otherwise represented in tabular form, but, more importantly, such an approach is obviously not applicable to problems which already have continuous state or action spaces.

3.3 Discussion

Our main aim was the formulation of a SOC problem in terms of inference, or more specifically minimisation of KL divergences. The benefit of such dualities is the possibility for application of approximation techniques from the inference literature. Of course such exact duality is not necessarily required to utilise the general principles underlying these approximate inference techniques, it does however significantly eases the processes by directly clarifying whether all assumptions are satisfied. At the same time it is important to note that any such duality will not yield a closed form solution which could not be derived in the SOC formulation. The later point is illustrated by the filtering dualities of Section 3.2.2. While these seemingly resolve the minimisation problem, the closed form resolution of the minimisation is equally possible for the SOC problem under equivalent assumptions of a control LQ form. As such, the primal purpose of such dual formulation is not the direct solution to the problem, but rather to provide a novel perspective, which in turn suggests new approaches.

The limitations of the filtering dualities and the KL based duality for MDPs of Section 3.2.3 are apparent. Both impose considerable constraints on the SOC problem. Consider the case of filtering dualities, which is of greater interest here as they allow for continuous states and actions. The problem is required to be control LQ and the quadratic control cost is linked to the noise magnitude. Furthermore it is apparent from the form of the noise variance, that noise acts through the same subspace as the controls. As discussed in Section 2.3, the

assumption of control linearity[8] is becoming increasingly problematic in robotics.

Meanwhile, both, the MAP estimation based formulation and our duality of Corollary 3.1 do not constraint the dynamics. However, while the filtering dualities directly lead to an inference problem, these formulations lead to a more challenging optimisation problem. In particular, without further assumptions on the costs, both take the form of a constrained minimisation problem. Although, Corollary 3.1 offers the advantage of subsuming the filtering dualities, hence reducing to an unconstrained problem under the right assumptions.

In the MAP estimation case, relaxation of the constraint by ignoring it has led to efficient sample based algorithms. The question therefore arises, whether suitable relaxation can similarly lead to practical algorithms in the proposed formulation. If so, it would provide the generality of MAP based methods, in conjuncture with, under suitable conditions, the simplicity of filtering based approaches. This is the question we address in the following chapters.

[8] n.b., due to the constraint on the noise, this problem can not be fixed by augmenting the state space with the controls and introducing $\dot{\mathbf{u}}$ as new control

Chapter 4

Ψ-Learning

We now examine how the general duality in Corollary 3.1, presented in the previous chapter, can be utilised to solve SOC problems. Recall that, within this duality we interpret the SOC problem as an attempt to match a cost conditioned posterior process p_{π^0} (cf. (3.3)) with the controlled process q_π (cf. (3.2)) and the problem takes the form

$$\operatorname*{argmin}_{\pi \in \mathcal{D}} \mathrm{KL}\left(q_\pi(\bar{\mathbf{x}}, \bar{\mathbf{u}}) \| p_{\pi^0}(\bar{\mathbf{x}}, \bar{\mathbf{u}})\right)$$

where $\mathcal{D}$ are deterministic policies. While unconstrained minimisation of a KL divergence $\mathrm{KL}\left(q\|p\right)$ w.r.t. q is achieved by $q = p$, the above problem is complicated by the presence of two constraints

- The policy is constrained to the deterministic set $\pi \in \mathcal{D}$
- The posterior approximation q_π is constrained to respect the system dynamics

Here, we examine the effect of relaxing the first of these constraints, while maintaining the second.

The first result is that such a relaxation yields a natural general condition for iterative policy improvement. We proceed to derive specific instances of such iterations in the finite and discounted infinite setting and study their asymptotic behaviour, demonstrating convergence in expected costs to the global optimum under mild conditions on π^0. In addition, we give an asymptotic bound on the

expected cost for a wide class of approximate iterations. We then turn to the sample based implementation of the iterations obtained in the discounted infinite horizon setting, which we term Ψ-Learning. Specifically, two algorithms are proposed, a direct implementation for finite state-control spaces and an approximation, applicable in cases of large or continuous state-control spaces. Finally, we examine the close connection of Ψ-Learning to several alternative algorithms which have been proposed.

4.1 Policy Improvement by Duality Relaxation

Relaxation of the constraint to allow minimisation over arbitrary stochastic, rather then just deterministic, policies, provides a closed form solution. Although this does not directly lead to an optimal policy, we have the following result:

Proposition 4.1. *For any* $\pi \neq \pi^0$,

$$\mathrm{KL}\left(q_\pi \| p_{\pi^0}\right) \leq \mathrm{KL}\left(q_{\pi^0} \| p_{\pi^0}\right) \Longrightarrow \mathcal{J}(\pi) < \mathcal{J}(\pi^0) \ .$$

Proof. Expanding the KL divergences we have

$$\begin{aligned} \mathrm{KL}\left(q_\pi \| q_{\pi^0}\right) - \mathbb{E}_{q_\pi}\left[\log P(r_k = 1 | \bar{\mathbf{x}}, \bar{\mathbf{u}})\right] + Z \\ \leq \mathrm{KL}\left(q_{\pi^0} \| q_{\pi^0}\right) - \mathbb{E}_{q_{\pi^0}}\left[\log P(\bar{r} = 1 | \bar{\mathbf{x}}, \bar{\mathbf{u}})\right] + Z \ , \end{aligned}$$

where $Z = \log P(\bar{r} = 1 | x_0; \pi^0)$. Subtracting Z on both sides and noting that $\mathrm{KL}\left(q_{\pi^0} \| q_{\pi^0}\right) = 0$, we obtain

$$\mathrm{KL}\left(q_\pi \| q_{\pi^0}\right) + \eta \mathcal{J}(\pi) \leq \eta \mathcal{J}(\pi^0) \ ,$$

where we used the fact $\log P(\bar{r} = 1 | \bar{\mathbf{x}}, \bar{\mathbf{u}}) = -\eta \mathcal{C}(\bar{\mathbf{x}}, \bar{\mathbf{u}})$. Hence, as $\eta \geq 0$ and $\mathrm{KL}\left(q_\pi \| q_{\pi^0}\right) \geq 0$ with equality iff $\pi = \pi^0$, the result follows. ■

This provides us with a general condition for step-wise policy improvement. In particular, with some initial π^0, the iteration

$$\pi^{n+1} \leftarrow \operatorname*{argmin}_\pi \mathrm{KL}\left(q_\pi \| p_{\pi^n}\right) \ , \tag{4.1}$$

gives rise to a chain of stochastic policies with ever decreasing expected costs. In the remainder of this section we shall examine specific instances of such iterations.

Specifically, we will begin with the finite horizon problem, where we can obtain closed form solutions to the iterates of (4.1). We subsequently study a class of approximations to (4.1), which will eventually allow us to extend the results of finite horizon case to the the discounted infinite horizon setting. While the subsequent sections provide a detailed account of the derivation of these updates and a study of the asymptotic behaviour of the associated iterations, in summary we obtain the following results. In both cases the iterates π^{n+1} take the general form of a Boltzmann like distribution

$$\pi^{n+1}(\mathbf{u}_k|\mathbf{x}_k) = \exp\{\Psi^{n+1}(\mathbf{x}_k, \mathbf{u}_k) - \bar{\Psi}^{n+1}(\mathbf{x}_k)\} \tag{4.2}$$

with energy Ψ and log partition function $\bar{\Psi}^{n+1}$, where the specific update for the two cases take the forms

- **Finite Horizon:**

$$\Psi_k^{n+1}(\mathbf{x}_k, \mathbf{u}_k) = \log \pi^n(\mathbf{u}_k|\mathbf{x}_k) - \eta\mathcal{C}(\mathbf{x}_k, \mathbf{u}_k) + \mathbb{E}_{\mathbf{x}_{k+1}|\mathbf{x}_k,\mathbf{u}_k}\left[\bar{\Psi}_{k+1}^{n+1}(\mathbf{x}_{k+1})\right]$$

- **Discounted Infinite Horizon:**

$$\Psi^{n+1}(\mathbf{x}, \mathbf{u}) = \log \pi^n(\mathbf{u}|\mathbf{x}) - \eta\mathcal{C}_\bullet(\mathbf{x}, \mathbf{u}) + \gamma\mathbb{E}_{\mathbf{y}|\mathbf{x},\mathbf{u}}\left[\bar{\Psi}^n(\mathbf{y})\right]$$

We refer to these methods collectively as Ψ-Iterations. As we shall see, the iterations in both these cases enjoy convergence to the globally optimal policy.

4.1.1 Finite Horizon Problems

Let us first study the standard finite horizon setting and derive the closed form solution to the iteration (4.1). We begin by noting the following general result which allows for unconstrained minimisation of a KL divergence with respect to a marginal of a posterior distribution

Proposition 4.2. *Let a, b, c be random variables with joint*

$$P(a, b, c) = P(a)P(b|a)P(c|b, a) \ ,$$

$\mathcal{P}$ the set of distributions over a and consider the minimisation[1]

$$z = \min_{q \in \mathcal{P}} \mathrm{kl}\left(q(a)P(b|a) \| P(a, b, c = \hat{c})\right) \ .$$

The minimum is attained at

$$q^*(a) \propto P(a) \exp\{\int_b P(b|a) \log P(c = \hat{c}|a, b)\} \tag{4.3}$$

and

$$z = -\log \int_a P(a) \exp\{\int_b P(b|a) \log P(c = \hat{c}|a, b)\} \ . \tag{4.4}$$

Proof. We form the Lagrangian

$$\mathcal{L} = \lambda \left[\int_a q(a) - 1\right] + \mathrm{kl}\left(q(a)P(b|a) \| P(a, b, c = \hat{c})\right)$$

where λ are the Lagrange multipliers. Setting the partial derivatives w.r.t. $q(a)$ to 0 gives

$$\begin{aligned} 0 &= \log \frac{q(a)}{P(a)} + 1 - \int_b P(b|a) \log P(c = \hat{c}|a, b) + \lambda \\ &= \log \frac{q(a)}{Z(\lambda)P(a) \exp\{\int_b P(b|a) \log P(c = \hat{c}|a, b)\}} \ , \end{aligned}$$

where Z is a function of the Lagrange multiplier. The result in (4.3) now directly follows and more specifically the minimizer is

$$q^*(a) = \frac{P(a) \exp\{\int_b P(b|a) \log P(c = \hat{c}|a, b)\}}{\int_a P(a) \exp\{\int_b P(b|a) \log P(c = \hat{c}|a, b)\}} \ .$$

Substituting q^* back into the minimised expression, we have

$$\begin{aligned} z &= \int_a q^*(a) \log \frac{q^*(a)}{P(a)} - \int_{a,b} q^*(a)P(b|a) \log P(c = \hat{c}|a, b) \\ &= \int_a q^*(a) \log \frac{\exp\{\int_b P(b|a) \log P(c = \hat{c}|a, b)\}}{Z} - \int_a q^*(a) \int_b P(b|a) \log P(c = \hat{c}|a, b) \\ &= \int_a q^*(a) \int_b P(b|a) \log P(c = \hat{c}|a, b) + \int_a q^* a \log \frac{1}{Z} - \int_a q^*(a) \int_b P(b|a) \log P(c = \hat{c}|a, b) \\ &= \int_a q^*(a) \log \frac{1}{Z} \\ &= -\log Z \ , \end{aligned}$$

with $Z = \int_a P(a) \exp\{\int_b P(b|a) \log P(c = \hat{c}|b)\}$. ∎

[1] n.b., we use kl to denote an expression of the form of a KL divergence where the arguments are not necessarily normalised, i.e., $\mathrm{kl}\left(f(x) \| g(x)\right) = \int f(x) \log \frac{f(x)}{g(x)}$.

Utilising this results, allows the solution to (4.1) to be obtained iteratively backwards in time and specifically we have

Proposition 4.3. *Let $\mathcal{P}$ be the set of stochastic policies on $(\mathcal{X},\mathcal{U},\mathcal{T})$ then the solution of*

$$\pi^{n+1} = \operatorname*{argmin}_{\pi\in\mathcal{P}} \mathrm{KL}\left(q_\pi \| p_{\pi^n}\right) \ ,$$

is given by the Boltzmann like distribution,

$$\pi^{n+1}(\mathbf{u}_k|\mathbf{x}_k) = \exp\{\Psi_k^{n+1}(\mathbf{x}_k,\mathbf{u}_k) - \bar{\Psi}_k^{n+1}(\mathbf{x}_k)\} \ , \tag{4.5}$$

with energy

$$\Psi_k^{n+1}(\mathbf{x}_k,\mathbf{u}_k) = \log \pi^n(\mathbf{u}_k|\mathbf{x}_k) - \eta\mathcal{C}(\mathbf{x}_k,\mathbf{u}_k) + \mathbb{E}_{\mathbf{x}_{k+1}|\mathbf{x}_k,\mathbf{u}_k}\left[\bar{\Psi}_{k+1}^{n+1}(\mathbf{x}_{k+1})\right]$$

and log partition function

$$\bar{\Psi}_k^{n+1}(\mathbf{x}_k) = \log \int_u \exp\{\Psi_k^{n+1}(\mathbf{x}_k,u)\} \ .$$

Proof. We begin by noting that

$$\operatorname*{argmin}_{\pi\in\mathcal{P}} \mathrm{KL}\left(q_\pi \| p_{\pi^n}\right) = \operatorname*{argmin}_{\pi\in\mathcal{P}} \mathrm{kl}\left(q_\pi \| P(\bar{\mathbf{x}},\bar{\mathbf{u}},\bar{r}=\mathbf{1}|\mathbf{x}_0,\pi^n)\right) \ ,$$

where $P(\bar{\mathbf{x}},\bar{\mathbf{u}},\bar{r}=\mathbf{1}|\mathbf{x}_0,\pi^n)$ is the none normalised posterior (3.3), as the normalisation constant $P(\bar{r}=\mathbf{1}|\mathbf{x}_0,\pi^n)$ is independent of π. The proof now proceeds by induction backwards in time using Proposition 4.2. Specifically, consider the policy of time $K-1$, i.e., π_{K-1}, obtained by minimizing

$$\mathrm{kl}\left(q_\pi(\mathbf{x}_K,\mathbf{u}_{K-1:K}|\mathbf{x}_{K-1}) \| P(\mathbf{x}_K,\mathbf{u}_{K-1:K},r_{K-1:K}=\mathbf{1}|\mathbf{x}_{K-1},\pi^n)\right) \ .$$

w.r.t. $\pi(\cdot|\mathbf{x}_{K-1})$ for each $\mathbf{x}_{K-1}$ independently. Applying Proposition 4.2 with $a=\mathbf{u}_{K-1}|\mathbf{x}_{K-1}$, $b=\mathbf{x}_K$ and $P(c=\hat{c}|b)=\exp\{-\eta\mathcal{C}_K(\mathbf{x}_K)+\mathcal{C}(\mathbf{x}_{K-1},\mathbf{u}_{K-1})\}$ leads to the base case. For the inductive step we observe that we may in general write the kl term in a recursive form as

$$\begin{aligned}
&\mathrm{kl}(q_{\pi^{n+1}}(\mathbf{x}_{k+1:K},\mathbf{u}_{k:K}|\mathbf{x}_k)\|\tilde{p}_{\pi^n}(\mathbf{x}_{k+1:K},\mathbf{u}_{k:K})) \\
&\quad= \int_{\mathbf{u}_k} \pi^{n+1}(\mathbf{u}_k|\mathbf{x}_k)\Big[\log\frac{\pi^{n+1}(\mathbf{u}_k|\mathbf{x}_k)}{\pi^n(\mathbf{u}_k|\mathbf{x}_k)P(r_k=1|\mathbf{x}_k,\mathbf{u}_k)} \\
&\qquad+ \mathbb{E}_{\mathbf{x}_{k+1}|\mathbf{x}_k,\mathbf{u}_k}\left[\mathrm{kl}\left(q_{\pi^{n+1}}(\mathbf{x}_{k+2:K},\mathbf{u}_{k+1:K}|\mathbf{x}_1)\|\tilde{p}_\pi(\mathbf{x}_{k+2:K},\mathbf{u}_{k+1:K})\right)\right]\Big] \ ,
\end{aligned}$$

where $\tilde{p}_\pi(\mathbf{x}_{k+1:K}, \mathbf{u}_{k:K}) = P(\mathbf{x}_{k+1:K}, \mathbf{u}_{k:K}, r_{k:K} = \mathbf{1}|\mathbf{x}_k, \pi)$. This allows Proposition 4.2 to be applied recursively with $a = \mathbf{u}_k|\mathbf{x}_k$, $b = \mathbf{x}_{k+1}$ and

$$P(c = \hat{c}|b, a) = P(r_k = 1|\mathbf{x}_k, \mathbf{u}_k) \exp\{-\bar{\Psi}_{k+1}^{n+1}(\mathbf{x}_{k+1})\} ,$$

where we used the fact that, by the second part of Proposition 4.2, the minimised nested kl term reduces to $-\bar{\Psi}_{k+1}^{n+1}$. ■

Convergence Analysis

Proposition 4.1 guarantees that the policies obtained from Proposition 4.3 have non-increasing expected costs and hence, as the expected cost is naturally bounded from below, converge with respect to the expected cost. However, obviously the question whether the procedure leads to a policy associated which a local or even global minimum of the expected cost is of particular interest. We proceed to show that the sequence of expected costs associated with the sequence of policies generated by Proposition 4.3 converges, under weak assumptions on π^0, to the expected cost of the globally optimal policy. Furthermore, the rate of convergence is a function of the KL divergence of trajectories under π^0 and an optimal policy.

We begin by stating the following result on the relation between successive policies.

Proposition 4.4. *Let $\{\pi^n\}$ be a sequence of policies generated by (4.1), then*

$$\eta\mathcal{J}(\pi^{n+1}) \leq -\bar{\Psi}_0^{n+1}(x_0) \leq \eta\mathcal{J}(\pi^n)$$

Proof. Let $\tilde{p}_{\pi^n}$, be the un-normalised posterior, i.e., $\tilde{p}_{\pi^n}(\bar{\mathbf{x}}, \bar{\mathbf{u}}) = P(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{r} = 1|\mathbf{x}_0, \pi^n)$, we have

$$\mathrm{kl}\left(q_{\pi^{n+1}} \| \tilde{p}_{\pi^n}\right) = \eta\mathcal{J}(\pi^{n+1}) + \mathrm{kl}\left(q_{\pi^{n+1}} \| q_{\pi^n}\right) \geq \eta\mathcal{J}(\pi^{n+1}) ,$$

where the inequality follows from $\mathrm{KL}\left(q_{\pi^{n+1}} \| q_{\pi^n}\right) \geq 0$. Also from (4.1) we have

$$\begin{aligned}
\mathrm{kl}\left(q_{\pi^{n+1}} \| \tilde{p}_{\pi^n}\right) &\leq \mathrm{kl}\left(q_{\pi^n} \| \tilde{p}_{\pi^n}\right) \\
&\leq \mathrm{KL}\left(q_{\pi^n}(\bar{\mathbf{x}}, \bar{\mathbf{u}}) \| q_{\pi^n}(\bar{\mathbf{x}}, \bar{\mathbf{u}})\right) - \mathbb{E}_{q_{\pi^n}}\left[\log P(\bar{r} = 1|\bar{\mathbf{x}}, \bar{\mathbf{u}})\right] \\
&\leq \eta\mathcal{J}(\pi^n) .
\end{aligned}$$

It follows that

$$\eta\mathcal{J}(\pi^{n+1}) \leq \mathrm{kl}\left(q_{\pi^{n+1}} \| \tilde{p}_{\pi^n}\right) \leq \eta\mathcal{J}(\pi^n) \ .$$

Now by assumption π^{n+1} is generated by (4.1) and hence minimises $\mathrm{KL}\left(q_{\pi^{n+1}} \| p_{\pi^n}\right)$ and thus also $\mathrm{kl}\left(q_{\pi^{n+1}} \| \tilde{p}_{\pi^n}\right)$. Therefore, by Proposition 4.2 (also cf. proof of Proposition 4.3) we have

$$\mathrm{kl}\left(q_{\pi^{n+1}} \| \tilde{p}_{\pi^n}\right) = -\bar{\Psi}_0^{n+1}(\mathbf{x}_0)$$

and the result follows. ■

Making use of this bound, we now bound the progress of the trajectory posterior under policy π^n towards the corresponding distribution under some chosen $\hat{\pi}$, obtaining

Proposition 4.5. *Let the sequence $\{\pi^n\}$ be generated by (4.1) and let $\hat{\pi}$ be an arbitrary (stochastic) policy. Then*

$$\mathrm{KL}\left(q_{\hat{\pi}} \| q_{\pi^{n+1}}\right) - \mathrm{KL}\left(q_{\hat{\pi}} \| q_{\pi^n}\right) \leq \eta\mathcal{J}(\hat{\pi}) - \eta\mathcal{J}(\pi^{n+1}) \ . \tag{4.6}$$

Proof. Let $\hat{\pi}$ be an arbitrary policy and consider

$$\begin{aligned}
\mathrm{KL}(q_{\hat{\pi}} \| q_{\pi^{n+1}}) - \mathrm{KL}\left(q_{\hat{\pi}} \| q_{\pi^n}\right) &= \int_{\bar{\mathbf{x}},\bar{\mathbf{u}}} q_{\hat{\pi}}(\bar{\mathbf{x}},\bar{\mathbf{u}}) \log \frac{q_{\pi^n}(\bar{\mathbf{x}},\bar{\mathbf{u}})}{q_{\pi^{n+1}}(\bar{\mathbf{x}},\bar{\mathbf{u}})} \\
&= \int_{\bar{\mathbf{x}},\bar{\mathbf{u}}} q_{\hat{\pi}}(\bar{\mathbf{x}},\bar{\mathbf{u}}) \log \prod_{k=0}^{K} \frac{\pi^n(\mathbf{u}_k|\mathbf{x}_k)}{\pi^{n+1}(\mathbf{u}_k|\mathbf{x}_k)} \\
&= \int_{\bar{\mathbf{x}},\bar{\mathbf{u}}} q_{\hat{\pi}}(\bar{\mathbf{x}},\bar{\mathbf{u}}) \sum_{t=0}^{T} \eta\mathcal{C}(\mathbf{x}_k,\mathbf{u}_k) - g \ ,
\end{aligned}$$

with

$$\begin{aligned}
g &= -\sum_{k=0}^{K} \int_{\bar{\mathbf{x}},\bar{\mathbf{u}}} q_{\hat{\pi}}(\bar{\mathbf{x}},\bar{\mathbf{u}}) \bar{\Psi}_k^{n+1}(\mathbf{x}_k) + \sum_{k=0}^{K} \int_{\bar{\mathbf{x}},\bar{\mathbf{u}}} q_{\hat{\pi}}(\bar{\mathbf{x}},\bar{\mathbf{u}}) \mathbb{E}_{x'|\mathbf{x}_k,\mathbf{u}_k}\left[\bar{\Psi}_{k+1}^{n+1}(x')\right] \\
&= -\bar{\Psi}_0^{n+1}(x_0) + \int_{\bar{\mathbf{x}},\bar{\mathbf{u}}} q_{\hat{\pi}}(\bar{\mathbf{x}},\bar{\mathbf{u}}) \mathbb{E}_{x'|\mathbf{x}_k,\mathbf{u}_k}\left[\bar{\Psi}_{K+1}^{n+1}(x')\right] \\
&= -\bar{\Psi}_0^{n+1}(x_0) \geq \eta\mathcal{J}(\pi^{n+1}) \ ,
\end{aligned}$$

where in the final line we used Proposition 4.4. ■

Summing the above bound over $0 \dots N$, we can compute the bound

$$\frac{1}{N}\sum_{n=1}^{N+1}\mathcal{J}(\pi^n) \leq \mathcal{J}(\hat{\pi}) + \frac{1}{\eta N}\,\mathrm{KL}\left(q_{\hat{\pi}} \| q_{\pi^0}\right) \ , \tag{4.7}$$

on the average expected cost of the policies $\pi^1 \dots \pi^{n+1}$. Now, since Proposition 4.1 guarantees that the expected cost for each π^n is non increasing with n, using (4.7), we can obtain the following stronger convergence result.

Proposition 4.6. *Let $\{\pi^n\}$ be a sequence of policies generated by (4.1), with π^0 s.t. $\pi^0(\cdot|x \in \mathcal{X})$ has support $\mathcal{U}$. Then*

$$\lim_{n\to\infty}\mathcal{J}(\pi^n) = \min_{\pi}\mathcal{J}(\pi) \ .$$

Proof. Summing the bound in Proposition 4.5 over $0 \dots N$, we have

$$\begin{aligned}\frac{1}{N}\sum_{n=1}^{N+1}\mathcal{J}(\pi^n) &\leq \mathcal{J}(\hat{\pi}) + \frac{1}{\eta N}\left[\mathrm{KL}\left(q_{\hat{\pi}} \| q_{\pi^0}\right) - \mathrm{KL}\left(q_{\hat{\pi}} \| q_{\pi^N}\right)\right] \\ &\leq \mathcal{J}(\hat{\pi}) + \frac{1}{\eta N}\,\mathrm{KL}\left(q_{\hat{\pi}} \| q_{\pi^0}\right) \ ,\end{aligned}$$

where the last line follows from $\mathrm{KL}\left(q_{\hat{\pi}} \| q_{\pi^N}\right) \geq 0$. Noting that $\hat{\pi}$ was chosen arbitrarily we may now choose, for any $\epsilon > 0$, $\hat{\pi}$ to be an ϵ-good policy π^*, i.e.,

$$\hat{\pi} = \pi^* \in \{\pi; \mathcal{J}(\pi) - \inf_{\pi'}\mathcal{J}(\pi') < \epsilon\} \ .$$

so that we have

$$\frac{1}{N}\sum_{n=1}^{N+1}\mathcal{J}(\pi^n) \leq \mathcal{J}(\pi^*) + \frac{1}{N\eta}\,\mathrm{KL}\left(q_{\pi^*} \| q_{\pi^0}\right) \ .$$

Note that, as the left-hand expression is the average expected cost over $\pi^1 \dots \pi^{N+1}$ there exists some $n \in 1 \dots N$ s.t.

$$\mathcal{J}(\pi^{N+1}) \leq \mathcal{J}(\pi^n) \leq \mathcal{J}(\pi^*) + \frac{1}{\eta N}\,\mathrm{KL}\left(q_{\pi^*} \| q_{\pi^0}\right) \ ,$$

with the first inequality following from Proposition 4.1.

Now, as by assumption on π^0, $\mathrm{KL}\left(q_{\pi^*} \| q_{\pi^0}\right) < \infty$, for any $\delta > 0$ there exists a pair $N_\delta < \infty$ s.t. $\frac{1}{N_\delta \eta}\,\mathrm{KL}\left(q_{\pi^*} \| q_{\pi^0}\right) < \frac{\delta}{2}$ and hence, with $\epsilon = \frac{\delta}{2}$,

$$\mathcal{J}(\pi^{N_\epsilon}) \leq \mathcal{J}(\pi^*) + \frac{\delta}{2} \leq \inf_{\pi}\mathcal{J}(\pi) + \delta \ ,$$

which gives the required result. ∎

Remark 4.1: In the case of continous control spaces we formally require the expected cost to be well behaved on the policy space. Specifically, note that in the proof of Proposition 4.6 we choose an ϵ good policy π^* s.t. $\mathrm{KL}\left(q_{\pi^*} \| q_{\pi^0}\right) < \infty$, lest the bound becomes trivial. This requires that π^* is absolutly continous w.r.t. to the base measure of π^0, i.e., in the continous case generally the Lebesgue measure. As such π^* can not be a deterministic policy. Hence, we require the expected cost to be well behaved in the sense that for any $\epsilon > 0$, there exists an ϵ good *stochastic* policy.

Remark 4.2: The condition on π^0 given in Proposition 4.6 is a sufficient condition. A necessary condition is that for any $\epsilon > 0$, there exists π' s.t. $\mathrm{KL}\left(q_{\pi'} \| q_{\pi^0}\right) < \infty$ and $\mathcal{J}(\pi') - \inf_\pi \mathcal{J}(\pi) < \epsilon$. In general,

$$\lim_{n\to\infty} \mathcal{J}(\pi^n) \leq \min_{\pi \in \mathcal{S}_{\pi^0}} \mathcal{J}(\pi) \qquad \mathcal{S}_{\pi^0} = \{\pi; \mathrm{KL}\left(q_{\pi'} \| q_{\pi^0}\right) < \infty\}$$

This result can be easily obtained by taking $\hat{\pi} = \inf_{\pi \in \mathcal{S}_{\pi^0}} \mathcal{J}(\pi)$ in the proof of Proposition 4.6.

4.1.2 Approximate Iterations

Performing the exact update required by (4.1) is in general hard, leading us to study the effect of replacing the exact minimisation of Section 4.1.1, with an approximation thereof. Specifically, let us consider the case where π_k^n of (4.5) is replaced with an approximation $\tilde{\pi}_k^n$. While we do not make any specific assumptions on the form of this approximation, we shall assume it is not arbitrary poor, in the sense that, for all $\mathbf{x}_k \in \mathcal{X}$, $\tilde{\pi}_k^n(\cdot|\mathbf{x}_k) = 0$, if and only if $\pi_k^n(\cdot|\mathbf{x}_k) = 0$. That is, we assume $\tilde{\pi}_k^n$ and π_k^n have the same support. Under this assumption we may define

$$\epsilon_k^n(\mathbf{u}_k, \mathbf{x}_k) = \begin{cases} \tilde{\pi}_k^n(\mathbf{u}_k|\mathbf{x}_k)/\pi_k^n(\mathbf{u}_k|\mathbf{x}_k) & \text{if } \mathbf{u}_k \in sup(\pi_k^n) \\ 1 & \text{else} \end{cases},$$

which we can interpret as a relative error measure of the approximation. Analogous to the case of an additive error measure, where the approximation is given as the sum of the true value and the error, we can now write the approximation

as a product of the true value and the relative error. Specifically, we have

$$\begin{aligned}\tilde{\pi}_k^n(\mathbf{u}_k|\mathbf{x}_k) &= \epsilon_k^n(\mathbf{u}_k,\mathbf{x}_k)\pi_k^n(\mathbf{u}_k|\mathbf{x}_k) \\ &= \pi_k^{n-1}(\mathbf{u}_k|\mathbf{x}_k)e^{-\hat{\mathcal{C}}^n(\mathbf{x}_k,\mathbf{u}_k)+\mathbb{E}_{\mathbf{x}_{k+1}|\mathbf{x}_k,\mathbf{u}_k}\left[\bar{\Psi}_{k+1}^n(\mathbf{x}_{k+1})\right]} ,\end{aligned}$$

where

$$\hat{\mathcal{C}}^n(\mathbf{x}_k,\mathbf{u}_k) = \mathcal{C}(\mathbf{x}_k,\mathbf{u}_k) + \log \epsilon_k^n(\mathbf{x}_k,\mathbf{u}_k)^{-1} . \tag{4.8}$$

It now becomes apparent that, rather then treat $\tilde{\pi}^n$ as an approximation of π^n, we may consider it to be the exact solution of an problem with augmented cost per stage $\hat{\mathcal{C}}^n$. Unlike in the case of exact iterations, this cost varies with iterations, however the tools employed for convergence analysis in the iteration stationary case can be easily adapted.

In general consider the problem with a iteration dependent cost $\mathcal{C}^n(\mathbf{x}_k,\mathbf{u}_k)$, with associated iteration dependent expected cost of a policy $\mathcal{J}^n(\pi)$. In this case the per step bound of Proposition 4.5 can be easily adapted as follows.

Proposition 4.7. *Let the sequence $\{\pi^n\}$ be generated by (4.1) under iteration varying costs and let $\hat{\pi}$ be an arbitrary (stochastic) policy. Then*

$$\mathrm{KL}\left(q_{\hat{\pi}}\|q_{\pi^{n+1}}\right) - \mathrm{KL}\left(q_{\hat{\pi}}\|q_{\pi^n}\right) \leq \eta\mathcal{J}^n(\hat{\pi}) - \eta\mathcal{J}^n(\pi^{n+1}) .$$

Proof. We follow the proof of Proposition 4.5. Thus, it is sufficient to show that a lower bound analogues to Proposition 4.4 holds. This is the case, as

$$\mathrm{kl}\left(q_{\pi^{n+1}}\|\tilde{p}_{\pi^n}\right) = \eta\mathcal{J}^n(\pi^{n+1}) + \mathrm{KL}\left(q_{\pi^{n+1}}\|q_{\pi^n}\right) \geq \eta\mathcal{J}^n(\pi^{n+1})$$

where $\tilde{p}_{\pi^n} = P(\bar{\mathbf{x}},\bar{\mathbf{u}},r=1|\mathbf{x}_0,\pi^n)$ is the un-normalised posterior based on n^{th} iteration costs. ∎

Summing this bound over $0\dots N$ yields

$$\frac{1}{N}\sum_{n=1}^{N+1}\mathcal{J}^n(\tilde{\pi}^n) \leq \frac{1}{N}\sum_{n=1}^{N+1}\mathcal{J}^n(\hat{\pi}) + \frac{1}{\eta N}\mathrm{KL}\left(q_{\hat{\pi}}\|q_{\pi^0}\right) ,$$

which in contrast to (4.7) involves on the r.h.s. an average over expected costs under the fixed policy $\hat{\pi}$.

In the specific case where the costs are given by $\hat{\mathcal{C}}^n$ as defined in (4.8), we have

$$\begin{aligned}\mathcal{J}^n(\pi) &= \mathbb{E}_{p_\pi}\left[\sum_{k=0}^{K} \mathcal{C}(\mathbf{x}_k, \mathbf{u}_k) + \log \epsilon_k^n(\mathbf{x}_k, \mathbf{u}_k)^{-1}\right] \\ &= \underbrace{\mathbb{E}_{p_\pi}\left[\sum_{k=0}^{K} \mathcal{C}(\mathbf{x}_k, \mathbf{u}_k)\right]}_{=\mathcal{J}(\pi)} + \underbrace{\mathbb{E}_{p_\pi}\left[\sum_{k=0}^{K} \log \epsilon_k^n(\mathbf{x}_k, \mathbf{u}_k)^{-1}\right]}_{=\mathcal{J}_\epsilon^n(\pi)} .\end{aligned}$$

That is, the expected none stationary cost splits into the expected cost of the underlying problem and an expected cost like term arising from the error. Hence, picking, as previously, $\hat{\pi} = \pi^*$, we obtain the bound on the average expected cost under the sequence of approximate policies

$$\frac{1}{N}\sum_{n=1}^{N+1} \mathcal{J}(\tilde{\pi}^n) \leq \mathcal{J}(\pi^*) + \frac{1}{\eta N}\,\mathrm{KL}\left(q_{\hat{\pi}} \| q_{\pi^0}\right) + \frac{1}{N}\sum_{n=1}^{N+1}\left(\mathcal{J}_\epsilon^n(\pi^*) - \mathcal{J}_\epsilon^n(\tilde{\pi}^n)\right) . \quad (4.9)$$

In particular, if $\tilde{\pi}^n$ converges to some $\tilde{\pi}^\infty$ w.r.t. $\mathcal{J}(\tilde{\pi}^n)$, i.e., for any $\kappa > 0$ there exists $N_\kappa < \infty$ such that

$$|\mathcal{J}(\tilde{\pi}^n) - \mathcal{J}(\tilde{\pi}^\infty)| < \kappa \qquad \forall n > N_\kappa$$

then

$$\mathcal{J}(\tilde{\pi}^\infty) \leq \mathcal{J}(\pi^*) + \frac{1}{N}\sum_{n=1}^{\infty}\left(\mathcal{J}_\epsilon^n(\pi^*) - \mathcal{J}_\epsilon^n(\tilde{\pi}^n)\right)$$

Although note that, in general, convergence of $\tilde{\pi}^n$ can not be guaranteed without making further assumptions on the sequence ϵ^n.

The obtained bound is interesting as it involves the average of a function of the approximation errors made. Thus, the proposed algorithm is forgiving, by virtue of only requiring the iteration to be good on average. In particular it is implicitly able to, eventually, recover if an error is made in an individual iteration. The later might be expected to prove a beneficial characteristic for sample based updates, which are correct on average. Although, obviously, if errors are made systematically the results can still be arbitrarily poor.

4.1.3 Infinite Horizon Problems

We shall now consider the infinite horizon setting with a discounted cost objective (cf. Table 2.1). It is easy to show that the time stationary analogue of (4.5) can

be obtained with

$$\Psi^{n+1}(x,u) = \log \pi^n(u|x) - \eta \mathcal{C}_\bullet(x,u) + \gamma \mathbb{E}_{y|x,u}\left[\bar{\Psi}^{n+1}(y)\right] \ . \tag{4.10}$$

However, due to the form of $\bar{\Psi}^{n+1}$, this does not yield Ψ^{n+1} in closed form. To obtain a practical solution, we make use of the relatively weak conditions given by Proposition 4.1 for obtaining a lower expected cost. This allow us to consider the minimisation in (4.1) over some iteration dependent subset $\mathcal{P}^n$ of the set of all (stochastic) policies. In particular we have

Proposition 4.8. *Let $\mathcal{P}$ be the set over all (stochastic) policies, if $\mathcal{P}^n \subseteq \mathcal{P}$ s.t. $\pi^n \in \mathcal{P}^n$ for all n, then the policies in the sequence generated by*

$$\pi^{n+1} \leftarrow \operatorname*{argmin}_{\pi \in \mathcal{P}^n} \mathrm{KL}\left(q_\pi \| p_{\pi^n}\right)$$

have non increasing expected costs.

Proof. As $\pi^n \in \mathcal{P}^n$, $\mathrm{KL}\left(q_{\pi^{n+1}} \| p_{\pi^n}\right) \leq \mathrm{KL}\left(q_{\pi^n} \| p_{\pi^n}\right)$ and hence either Proposition 4.1 applies or $\pi^n = \pi^{n+1}$. ■

Such iterations admit asynchronous updates as an interesting case, i.e., updating one or several time steps of the policy at each iteration in any particular order. Formally, we choose a schedule of time step sets $\mathcal{T}^0, \mathcal{T}^1, \ldots$ and let $\mathcal{P}^n = \{\pi; \forall k \notin \mathcal{T}^n, \pi_k = \pi_k^n\}$. Specifically, we will consider the schedule for such updates given by $\mathcal{T}^n = \{0, \ldots, n-1\}$, i.e., in each iteration we consider finite horizon problems with increasing horizon. Such a schedule leads to the update

$$\pi_k^{n+1} = \begin{cases} \pi_{k-1}^n & \text{if } k > 0 \\ \exp\{\Psi_0^{n+1}(\mathbf{x},\mathbf{u}) - \bar{\Psi}_0^{n+1}(\mathbf{x})\} \end{cases} \tag{4.11}$$

where

$$\Psi_0^{n+1}(x,u) = \Psi_0^n(x,u) - \bar{\Psi}_0^n(x) - \eta \mathcal{C}_\bullet(x,u) + \gamma \mathbb{E}_{x'|x,u}\left[\bar{\Psi}_0^n(x')\right] \ , \tag{4.12}$$

and $\bar{\Psi}_0^{n+1}(\cdot)$ is the log partition function.

Remark 4.3: Note that, (4.12) can be seen as a fixed point iteration for solving the initial, implicit, equation (4.10).

Convergence Analysis

As the iteration of Proposition 4.8 in general and in particular (4.11) are weaker then the iteration in the finite horizon case, i.e., Proposition 4.3, we are faced by the question whether the convergence properties of the latter have been maintained. Fortunately, essentially equivalent results to those for the finite horizon case can be obtained for the asynchronous algorithm (4.11) in the infinite horizon setting. To this end we shall make use of the results from Section 4.1.2, showing that the error introduced by performing incomplete updates goes to zero.

Note that, for iteration N the update of policies for time steps $k = 0 \dots N$ is exact while all subsequent updates are approximated by leaving policy π^0 invariant. Hence the relative error can be written as

$$\epsilon_k^n(\mathbf{u}_k, \mathbf{x}_k) = \begin{cases} \exp\{\mathcal{C}_\bullet(\mathbf{u}_k, \mathbf{x}_k) - \gamma \mathbb{E}_{\mathbf{x}_{k+1}|\mathbf{x}_k,\mathbf{u}_k}\left[\bar{\Psi}^0(\mathbf{x}_{k+1})\right] + \bar{\Psi}^1(\mathbf{x}_k)\} & \text{if } k > n \\ 1 & \text{else} \end{cases}$$

and the associated expected error induced cost under some policy is

$$\mathcal{J}_\epsilon^n(\pi) = \mathbb{E}_{q_\pi}\left[\sum_{k=n}^{\infty} \gamma^k \log \epsilon_k^n(\mathbf{x}_k, \mathbf{u}_k)^{-1}\right] \leq c \sum_{k=n}^{\infty} \gamma^k = c\frac{\gamma^{n-1}}{1-\gamma} ,$$

where the bound follows due the bounded costs. Substituting the bound into (4.9), we thus obtain

$$\frac{1}{N}\sum_{n=1}^{N+1} \mathcal{J}(\tilde{\pi}^n) \leq \mathcal{J}(\hat{\pi}) + \frac{2c}{N}\sum_{n=1}^{N+1} \frac{\gamma^{n-1}}{1-\gamma} + \frac{1}{\eta N} \mathrm{KL}\left(q_{\hat{\pi}} \| q_{\pi^0}\right)$$

Although informally, picking $\hat{\pi} = \pi^*$ and taking the limit $n \to \infty$ seems to provide the required result, a difficulty arises as $\mathrm{KL}\left(q_{\hat{\pi}} \| q_{\pi^0}\right)$ is not necessarily finite, due to the infinite horizon. However, as the cost is discounted we may work with an appropriate series of finite horizon bounds to obtain the following result.

Proposition 4.9. *Let $\{\pi^n\}$ be a sequence of policies generated by (4.11), with*[2] *π^0 s.t. $\pi^0(\cdot|x \in \mathcal{X})$ has support $\mathcal{U}$. Then*

$$\lim_{n\to\infty} \mathcal{J}(\pi^n) = \inf_\pi \mathcal{J}(\pi) .$$

[2] n.b. similar to Proposition 4.6, Remark 4.2 applies

Proof. Let $\mathcal{J}_K(\cdot)$ denote the expected cost over some finite horizon K, i.e.,

$$\mathcal{J}_K(\pi) = \mathbb{E}_{q_\pi(\mathbf{x}_{1:K}, \mathbf{u}_{0:K})} \left[\sum_{k=0}^{K} \gamma^k \mathcal{C}_\bullet(\mathbf{x}_k, \mathbf{u}_k) \right]$$

Follow the derivations in Section 4.1.1 and 4.1.2, it is straightforward to show that

$$\frac{1}{N} \sum_{n=1}^{N+1} \mathcal{J}_K(\pi^n) \leq \mathcal{J}_K(\pi^*) + \frac{2c}{N} \sum_{n=1}^{N+1} \frac{\gamma^{n-1}}{1-\gamma} + \frac{1}{\eta N} \mathrm{KL}\left(q_{\pi^*}(\mathbf{x}_{0:K}, \mathbf{u}_{0:K}) \| q_{\pi^0}(\mathbf{x}_{0:K}, \mathbf{u}_{0:K}) \right)$$

holds, where π^n are the none time stationary infinite horizon policies generated by (4.11) and π^* is the optimal policy[3] of the infinite horizon problem. The KL divergence is finite, if π^* is on the support of π^0. Hence for any $\epsilon > 0$, there exists $N_\epsilon < \infty$, s.t., for all $N \geq N_\epsilon$

$$\frac{1}{N} \sum_{n=1}^{N+1} \mathcal{J}_K(\pi^N) \leq \mathcal{J}_K(\pi^*) + \epsilon \tag{4.13}$$

Furthermore,

$$\mathcal{J}_K(\pi) = \mathcal{J}(\pi) - \underbrace{\mathbb{E}_{\mathbf{x}_{>K}, \mathbf{u}_{>K} | \mathbf{x}_0, \pi} \left[\sum_{k=K}^{\infty} \gamma^k \mathcal{C}_\bullet(\mathbf{x}_k, \mathbf{u}_k) \right]}_{:= \mathcal{J}_{>K}(\pi)} \tag{4.14}$$

and as the costs are bounded, $0 \leq \mathcal{J}_{>T}(\pi) \leq \bar{c} \frac{\gamma^{K-1}}{1-\gamma}$, for some constant $\bar{c}$. Substituting on both sides of (4.13), and observing that by Proposition 4.8, $\mathcal{J}(\pi^n)$ is non-increasing, we obtain

$$\mathcal{J}(\pi^{N_\epsilon}) \leq \mathcal{J}(\pi^*) + \bar{c} \frac{\gamma^{K-1}}{1-\gamma} + \epsilon \tag{4.15}$$

Hence, for any $\delta > 0$, there exists a pair $N_\delta < \infty$,$K < \infty$, s.t., for all $n \geq N_\delta$

$$\mathcal{J}(\pi^n) \leq \mathcal{J}(\pi^*) + \delta \tag{4.16}$$

and the result follows. ■

[3] n.b., in the interest of a simpler exposition, we assume $\pi^* = \operatorname{argmin}_\pi \mathcal{J}(\pi)$ exists, the argument can easily be adapted along similar lines to Proposition 4.6 by picking an ϵ-good policy.

Note, that the proof given here is for the none time stationary policy arising from the updates. In practise as the problem is time stationary one would follow the time stationary policy $\pi_\bullet(u|x) = \pi_0^n(u|x)$, i.e., the first step policy generated by (4.11). This circumvents the problem of having to store the ever growing non stationary policy, which is given by the history of first step policies. It is furthermore justified, as for any k, $\pi_{k-1}^n = \pi_k^{n+1}$ (cf. (4.11)) while by Proposition 4.8 π_k^{n+1} is a better policy in terms of expected cost.

Finally, for completeness, an alternative direct proof of global convergence, which does not utilise the results of Section 4.1.2 and which we previously presented (Rawlik et al., 2012), is given in Appendix C.

4.2 Reinforcement Learning

We now turn to the application of the proposed Ψ-iterations in the RL setting (cf. Section 2.1.1). This is motivated by the intractable nature of the updates derived above and the observation that some of the difficulty can be circumvented by making use of Monte Carlo evaluations. In the following we will concentrate on the discounted cost infinite horizon case, more commonly encountered in the RL literature, discussed in Section 4.1.3, with the understanding that equivalent steps can be taken in the finite horizon setting.

We first study a direct Monte Carlo implementation of Ψ-Iteration in the case of small finite state and control spaces, subsequently extending the approach to the continuous setting by means of function approximation. In general we will refer to the arising methods in analogy to $\mathcal{Q}$-Learning as Ψ-Learning.

4.2.1 Discrete-Finite State and Control Spaces

We observe that, for any given $\mathbf{x}, \mathbf{u}$ the update of (4.12) can be written as an expectation with respect to the transition probability $P(\mathbf{y}|\mathbf{x}, \mathbf{u})$, and hence, may be approximated from a set of sampled transitions. In particular for a given $(\mathbf{x}, \mathbf{u})$ and a set $\{\mathbf{y}_{1:M}\}$ of i.i.d. samples from $P(\mathbf{y}|\mathbf{x}, \mathbf{u})$, we have the unbiased estimator

$$\Psi(\mathbf{x}, \mathbf{u}) \leftarrow \Psi(\mathbf{x}, \mathbf{u}) - \bar{\Psi}(\mathbf{x}) - \mathcal{C}_\bullet(\mathbf{x}, \mathbf{u}) + \gamma \frac{1}{M} \sum \bar{\Psi}(\mathbf{y}_m) \tag{4.17}$$

In the RL setting obtaining such an i.i.d. sample is rarely practical, as the data is generated by the agent directly interacting with the world. Instead, assume, we are given a single sample $(\mathbf{x}, \mathbf{u}, \mathcal{C}_\bullet(\mathbf{x}, \mathbf{u}), \mathbf{y})$ of a transition from $\mathbf{x}$ to $\mathbf{y}$ under control $\mathbf{u}$, incurring cost $\mathcal{C}_\bullet$, we may perform the approximate update

$$\Psi(\mathbf{x}, \mathbf{u}) \leftarrow \Psi(\mathbf{x}, \mathbf{u}) + \alpha \left[\mathcal{R} + \gamma \bar{\Psi}(\mathbf{y}) - \bar{\Psi}(\mathbf{x})\right] \ , \tag{4.18}$$

where $\mathcal{R} = -\mathcal{C}_\bullet$ and we introduce α as a learning rate parameter. For trajectory data we then apply such an update for each tuple $(\mathbf{x}_k, \mathbf{u}_k, \mathcal{C}_\bullet(\mathbf{x}_k, \mathbf{u}_k), \mathbf{x}_{k+1})$ individually.

The convergence of (4.17) can be directly motivated from the convergence result for approximate iterations in Section 4.1.2. Informally, as the additional cost at convergence is the expectation of the error term, substituting an un-biased estimate of the successive policy does not affect the global convergence. Note in particular, in contrast to stochastic approximation algorithms this generally the case without requiring an asymptotically vanishing learning rate. We do however not provide an explicit formal proof for the convergence behaviour of either (4.17) and (4.18) beyond this informal motivation for the particular suitability of Ψ-Iterations for RL problems and the empirical observation of convergence for (4.18) (with constant learning rate). In particular (4.18) was formulated in analogy to $\mathcal{Q}$-Learning (see also Section 4.3.1) and formal analysis of it's convergence behaviour remains an open problem.

Example 4.1: We evaluate the proposed algorithm on the Grid-World problem (Sutton and Barto, 1998). The state space is given by a $N \times N$ grid. In each state, the control set consists of moves to any adjacent cell, such a move succeeding with probability 0.8. Additionally, some of the cells are occupied by obstacles and moves to these cells fail with probability 1. Finally, a set $\mathcal{A} \subseteq \mathcal{X}$ of absorbing target states is defined and the agent incurs a cost of 1 at all states other than the target, i.e., $C(x, u) = \delta_{x \notin \mathcal{A}}$ with δ the Kronecker delta. The cost is not discounted.

We compare performance against the standard based line of tabular **Q**-Learning (Sutton and Barto, 1998). Both algorithms were given data from episodes generated with controls sampled from an uninformed policy. Once a target state was reached, or if the target wasn't reached within 100 steps,

the state was reset randomly. The learning rate for Q-learning decayed as $\alpha = c/(c+m)$ with m the number of transitions sampled and c a constant which was optimised manually. The learning rate for Ψ-Learning was set to one.

As Ψ- and $\mathcal{Q}$-Learning, learn a policy and a state-control value function respectively, we choose to compare their performance in terms of the learned approximation to the value function. This is given by $\max_{\mathbf{u}} \mathcal{Q}(\mathbf{x}, \mathbf{u})$ in the case of $\mathcal{Q}$-Learning and, as we show in Section 4.3.1, by $\bar{\Psi}$ for Ψ-Learning. We plot the approximation error

$$e_{\mathcal{J}} = \frac{\max_x |\mathcal{V}(x) - \hat{\mathcal{V}}(x)|}{\max_x \mathcal{V}(x)} \ .$$

Representative results are illustrated in Figure 4.1. We observe that while, both algorithms achieved the same error at convergence, Ψ-Learning consistently requires fewer samples than Q-learning for convergence.

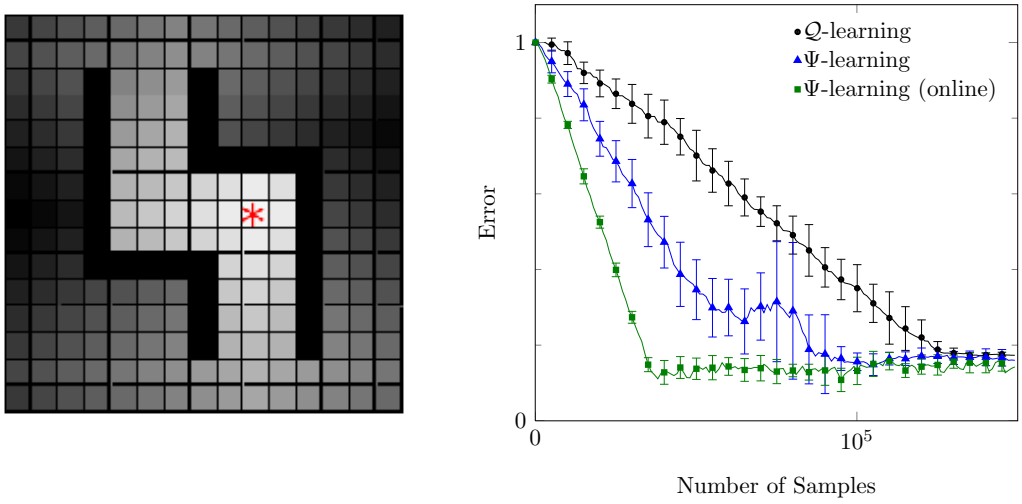
We additionally consider a online variant of Ψ-learning where the controls are sampled from the policy given by the current Ψ, i.e. $\pi(u|x) = \exp\{\Psi(x,u) - \bar{\Psi}(x)\}$. As expected, the online version outperforms sampling using an uninformed policy.

4.2.2 Large or Infinite State and Control Spaces

One needs to use parametric representations (Sutton and Barto, 1998) to store Ψ, when tabular means are no longer viable or efficient, as is the case with high dimensional or large discrete problems, or for continuous state and control spaces. Similar to numerous previous approaches (e.g., Lagoudakis and Parr, 2003; Kober and Peters, 2009; Peters et al., 2010; Azar et al., 2011), we use a linear basis function model to approximate the quantity of interest, which in our case is Ψ. Specifically, we write

$$\Psi(\mathbf{x}, \mathbf{u}) \approx \tilde{\Psi}(\mathbf{x}, \mathbf{u}, \mathbf{w}) = \sum_{l=0}^{L} w_l \phi_l(\mathbf{x}, \mathbf{u})$$

where $\phi_l : \ \mathcal{X} \times \mathcal{U} \ \rightarrow \ \mathbb{R}$ are a set of given basis functions and $\mathbf{w} = (w_1, \ldots, w_L)$ is the vector of parameters that are optimised. For such an approximation, and



(a) (b)

Figure 4.1: Results from the Grid-World problem. **(a)** Evolution of the mean error, averaged over 10 trials with error bars indicating the s.d. **(b)** Optimal value function (white low expected cost - black high expected cost) of the problem. Obstacles are black and the target state is indicated by *.

given a set of samples $(\mathbf{x}_{1\ldots M}, \mathbf{u}_{1\ldots M}, \mathcal{R}_{1\ldots M}, \mathbf{y}_{1\ldots M})$, the updates (4.18) can be written in matrix notation as

$$\Phi \mathbf{w}^{n+1} = \Phi \mathbf{w}^n + \mathbf{z} \ ,$$

where Φ is the $M \times L$ matrix with entries $\Phi_{i,j} = \phi_i(\mathbf{x}_j, \mathbf{u}_j)$ and $\mathbf{z}$ is the vector with elements

$$\mathbf{z}_m = \gamma \bar{\Psi}(\mathbf{y}_m) + \mathcal{R}_m - \bar{\Psi}(\mathbf{x}_m) \ .$$

This suggests the update rule of the form

$$\mathbf{w} \leftarrow \mathbf{w} + (\Phi^T \Phi)^{-1} \Phi^T \mathbf{z} \ .$$

The choice of basis functions is somewhat complicated by the need to evaluate the log partition function of the policy $\bar{\Psi}$, i.e. $\log \int_{\mathbf{u}} \exp\{\tilde{\Psi}(\mathbf{x}, \mathbf{u})\}$, when forming the vector $\mathbf{z}$. In cases where $\mathcal{U}$ is a finite set, arbitrary basis functions can be chosen as the integral reduces to a finite sum. However, for problems with continuous (or infinite discrete) control spaces, bases need to be chosen such that, the resulting integral is analytically tractable, i.e. the partition function of the

stochastic policy can be evaluated. One class of such basis sets is given by those $\tilde{\Psi}(\mathbf{x}, \mathbf{u}, \mathbf{w})$ that can be brought into the form

$$\tilde{\Psi}(\mathbf{x}, \mathbf{u}, \mathbf{w}) = -\frac{1}{2}\mathbf{u}^T\mathbf{A}(\mathbf{x}, \mathbf{w})\mathbf{u} + \mathbf{u}^T\mathbf{a}(\mathbf{x}, \mathbf{w}) + a(\mathbf{x}, \mathbf{w}) \ , \tag{4.19}$$

where $\mathbf{A}(\mathbf{x}, \mathbf{w})$ is a positive definite matrix-valued function, $\mathbf{a}(\mathbf{x}, \mathbf{w})$ is a vector-valued function and $a(\mathbf{x}, \mathbf{w})$ a scalar function. For such a set, the integral is of the Gaussian form and the closed form solution

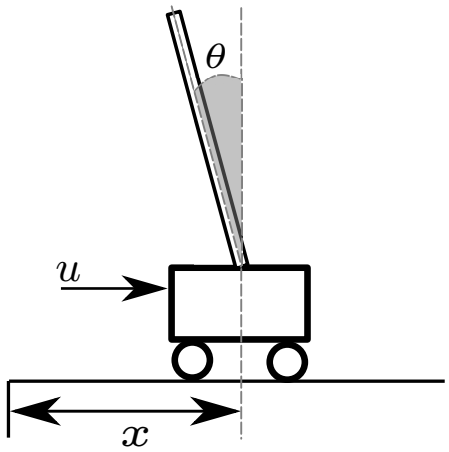
$$\log \int_{\mathbf{u}} \exp\{\tilde{\Psi}(\mathbf{x}, \mathbf{u}, \mathbf{w})\} = -\log|\mathbf{A}| - \frac{1}{2}\mathbf{a}'\mathbf{A}^{-1}\mathbf{a} + a + constant$$

is obtained. This gives us a recipe to employ basis functions that lead to tractable computations and the policy can be computed as $\pi(\mathbf{u}|\mathbf{x}, \mathbf{w}) = \mathcal{N}(\mathbf{u}|\mathbf{A}^{-1}\mathbf{a}, \mathbf{A}^{-1})$. The main implication of (4.19) is therefore the restriction to uni-modal policies. Note however that, as $\mathbf{a}$ is an arbitrary function of $\mathbf{x}$ the assumption of (4.19) does not pose a serious constraint on the complexity of the policies which can be learned.

Example 4.2: We consider the classical Cart-Pole problem (Sutton and Barto, 1998), which has been repeatedly used as a benchmark in reinforcement learning. The plant, illustrated in Figure 4.2, consists of a inverted pendulum which is mounted on a cart and is controlled by exerting forces on the latter. Formally, the state space is given by $\mathbf{x} = (x, \dot{x}, \theta, \dot{\theta})$, with x the position of the cart, θ the pendulum's angular deviation from the upright position and $\dot{x}, \dot{\theta}$ their respective temporal derivatives. Neglecting the influence of friction, the continuous time dynamics of the state are given by

$$\begin{aligned}\ddot{\theta} =& \frac{g\sin(\theta) + \cos(\theta)\left[-c_1 u - c_2\dot{\theta}^2\sin(\theta)\right]}{\frac{4}{3}l - c_2\cos^2(\theta)} \\ \ddot{x} =& c_1 u + c_2\left[\dot{\theta}^2\sin(\theta) - \ddot{\theta}\cos(\theta)\right] \ ,\end{aligned}$$

with $g = 9.8m/s^2$ the gravitational constant,constants $c_1 = (M_p + M_c)^{-1}$ and $c_2 = lM_p(M_p + M_c)^{-1}$ and parameters l,M_p,M_c summarised in Figure 4.2 The control interval is $0.02s$ and the dynamics are simulated using the fourth order Runga-Kutta method. Stochasticity is introduced by adding zero mean Gaussian noise, with small diagonal covariance, to the new state. These settings correspond to those used by Riedmiller et al. (2007) in a comparative evaluations of policy gradient RL method.



Parameter	Value
M_c (Cart mass)	1kg
M_p (Pendulum mass)	0.1kg
l (Pendulum length)	0.5m

Figure 4.2: The Cart-Pole plant and it's parameters. The pendulum is controlled by exerting a force $\mathbf{u}$ on the cart.

The task, which consists of stabilising the pendulum in the upright position, while simultaneously keeping the cart at the center of the track, has the cost function

$$\mathcal{C}_\bullet(\mathbf{x},\mathbf{u}) = \begin{cases} 0 & \text{if } (x,\theta) \text{ in target set} \\ \omega_\theta\theta^2 + \omega_x x^2 & \text{else} \end{cases} ,$$

where the target is given by $x \in [-0.05m, 0.05m]$ and $\theta \in [-0.05rad, 0.05rad]$ and the discount rate is $\gamma = 0$. We choose this cost as we found it to give better results for uniformed initial policies, for which the piecewise constant cost used by Riedmiller et al. provides little information.

The linear policy learned by Riedmiller et al. corresponds to a second order polynomial basis for Ψ in Ψ-Learning. Specifically, we use the basis set

$$\{u^2, ux, u\dot{x}, u\theta, u\dot{\theta}, x^2, x\dot{x}, x\theta, x\dot{\theta}, \dot{x}^2, \dot{x}\theta, \dot{x}\dot{\theta}, \theta^2, \theta\dot{\theta}, \dot{\theta}^2\} ,$$

which is of the form (4.19) and indeed only constitutes an approximation to the true Ψ as the problem is non-LQG.

Episodes were sampled with starting states drawn such that

$$\theta \in [-0.2rad, 0.2rad] \text{ and } x \in [-0.5m, 0.5m] , \tag{4.20}$$

and controls were sampled from the stochastic policy given by the current parameters. During training, episodes were terminated if the plant left the

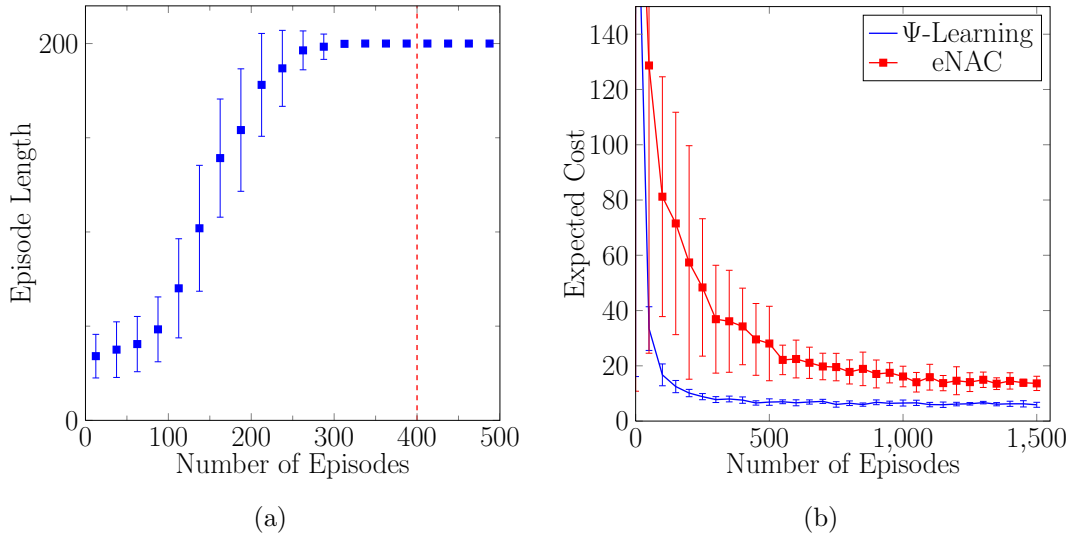


Figure 4.3: Results for RL with continuous state and action spaces on the Cart-Pole balancing task. The Cart-Pole system is illustrated in Figure 4.2. **(a)** Length of training episodes, averaged over blocks of 25 episodes, for Ψ-Learning, when initialized with an uninformed policy. The dashed red line indicates the point at which initial policies for the results in the subsequent comparison experiment were picked. Error bars indicate s.d. **(b)** Comparison of evolution of the expected cost between eNAC and Ψ-Learning. Both methods are initialised with the same stabilising policies (cf. (a)) and results averaged over 10 trials with error bars indicating s.d.

acceptable region defined by (4.20) or after 200 steps. Policy parameters were updated every 10 episodes and every 5 updates policies were evaluated by sampling 50 episodes of 500 step length using the mean of the policy. All results were averaged over 10 trials. The learning rate parameter for policy gradient methods was adjusted manually for best results.

Despite the change in cost function, like Riedmiller et al., we were not able to reliably obtain good policies from uninformed initialisation when using policy gradient methods. Our method on the other hand, when initialised with an uninformed policy, i.e., zero mean and a variance of 10, was able to learn a stabilising policy within 400 training episodes. This is illustrated in Figure 4.3(a) where the average length of training episodes is shown. In order to be able to compare to the episodic Natural Actor Critic (eNAC) method,

which produced the best result obtained by Riedmiller et al., we used the policies obtained by Ψ-Learning after 400 training episodes as initial policies. By this stage, the average expected cost of the policies was 239.35 compared to the initial cost which had been of the order 3×10^5. Figure 4.3(b) shows the evolution of the expected cost for both methods with such an initialisation and as can be seen Ψ-Learning outperformed eNAC both in terms of convergence speed and attained expected cost.

As the quality of the obtained policy will depend on how well the basis set can approximate the true Ψ, we also considered a more complex set of bases. Specifically, while keeping $\mathbf{A}$ in (4.19) a set of non-zero constant basis functions, we represented $\mathbf{a}(\mathbf{x}, \mathbf{w})$ and $a(\mathbf{x}, \mathbf{w})$ using the general and commonly used squared exponential bases which are of the form

$$\phi(\mathbf{x}) = \exp\{-(\mathbf{x} - m_\phi)^T \Sigma_\phi (\mathbf{x} - m_\phi)\} \ ,$$

with center m_ϕ and metric Σ_ϕ. The centers were sampled randomly from a region given by the acceptable region specified earlier and

$$\dot{x} \in [-1m/s, 1m/s] \text{ and } \dot{\theta} \in [-1rad/s, 1rad/s]$$

and Σ_ϕ was chosen to be diagonal. For this setting we were not able to obtain good policies using eNAC, while in the case of Ψ-Learning this choice did not outperform the polynomial basis, yielding a best policy with expected cost 26.4.

4.3 Relation to Previous Methods

4.3.1 Tabular Methods

$\mathcal{Q}$-Learning and TD(0)

Tabular Ψ-Learning, i.e., (4.18), has interesting relations to two classical algorithms, $\mathcal{Q}$-learning and TD(0), details of both of which can be found in the standard text by Sutton and Barto (1998). Before discussing these, let us first establish some basic properties, which follow directly from the proof of convergence in the infinite horizon case.

First, recollect that (i) the policy $\pi(\mathbf{u}|\mathbf{x}) = \exp\{\Psi(\mathbf{x},\mathbf{u}) - \bar{\Psi}(\mathbf{x})\}$ is of Boltzmann type with energy $\Psi(\mathbf{x},\mathbf{u})$, (ii) we proved convergence of Ψ-Learning to the optimal policy, and (iii) we know that optimal policies in MDPs are deterministic.[4] From these facts it follows that the energy $\Psi(\mathbf{x},\mathbf{u})$ converges to $-\infty$ for non-optimal actions $\mathbf{u}$ in each state $\mathbf{x}$.

Second, note that the log-$\int_u$-exp operation in the partition function $\bar{\Psi}(\mathbf{x}) = \log \int_{\mathbf{u}} \exp\{\Psi(\mathbf{x},\mathbf{u})\}$ degenerates to a $\max_u$-operation if $\Psi(\mathbf{x},\mathbf{u})$ is $-\infty$ for all actions except the optimal one in $\mathbf{x}$. Therefore it follows that, if Ψ has converged, $\bar{\Psi}(\mathbf{x}) = \max_{\mathbf{u}} \Psi(\mathbf{x},\mathbf{u})$.

Third, when replacing $\bar{\Psi}(\mathbf{x}) = \max_{\mathbf{u}} \Psi(\mathbf{x},\mathbf{u})$, Equation (4.18) becomes equivalent to standard $\mathcal{Q}$-Learning update

$$\mathcal{Q}(\mathbf{x},\mathbf{u}) \leftarrow \mathcal{Q}(\mathbf{x},\mathbf{u}) + \alpha \left[\mathcal{R} + \gamma \max_{\mathbf{u}'} \mathcal{Q}(\mathbf{y},\mathbf{u}') - \mathcal{Q}(\mathbf{x},\mathbf{u})\right] , \tag{4.21}$$

Therefore it follows that, if Ψ has converged, $\Psi(\mathbf{x},\mathbf{u}) = Q^*(\mathbf{x},\mathbf{u})$ for optimal actions $\mathbf{u}$ in each $\mathbf{x}$ (while still being $-\infty$ for non-optimal).

We summarize these findings in

Corollary 4.1. *At the point of convergence of Ψ-Learning (4.18) we have*

$$\Psi(\mathbf{x},\mathbf{u}) = \begin{cases} -\infty & \mathbf{u} \textit{ is non-optimal in } \mathbf{x} \\ \mathcal{V}(x) & \mathbf{u} \textit{ is optimal in } \mathbf{x} \end{cases}$$

$$\bar{\Psi}(\mathbf{x}) = \max_{\mathbf{u}} \Psi(\mathbf{x},\mathbf{u}) = \mathcal{V}(\mathbf{x}) ,$$

where $\mathcal{V}(\mathbf{x})$ is the optimal value function.

Note that these statements concern Ψ *after* convergence of Ψ-Learning. During learning the Ψ is not directly related to the classical $\mathcal{Q}$-function and the course of convergence is generally different. In the following we will highlight differences between Ψ-Learning and the classical RL algorithms in this respect.

Reconsider the $\mathcal{Q}$-learning algorithm (4.21). Note that it employs information from the current command and the single best future command under current knowledge. Ψ-Learning on the other hand uses a soft-max operation by employing $\bar{\Psi}$, averaging over information about the future according to the current

[4] to be more precise, an optimal policies has for any $\mathbf{x}$, support only on optimal controls for said $\mathbf{x}$

belief about the control distribution, hence taking uncertainty arising from, e.g., sampling into account.

The TD(0) algorithm, learns value function of the sampling policy π_s, that is, it learns $\mathcal{V}^{\pi_s}(\mathbf{x}) = \mathcal{J}(\pi_s, \mathbf{x})$. It has updates of the form

$$\mathcal{V}^{\pi_s}(x) \leftarrow \mathcal{V}^{\pi_s}(x) + \alpha \left[\mathcal{R} + \gamma \mathcal{V}^{\pi_s}(y) - \mathcal{V}^{\pi_s}(x)\right] \ ,$$

with α again a learning rate. Since it can be shown that $\bar{\Psi}$ converges to the value function of the optimal policy, the proposed update converges towards the TD(0) update for samples generated under the optimal policy. In particular, while TD(0) is an on-policy method and learns the value function of the policy used to generate samples, the proposed method learns the value function of the optimal policy directly.

z-Learning

The task in Example 4.1 corresponds to that of Todorov (2006a), who proposes an algorithm, equivalent to Ψ-Learning, in the context of linearly solvable MDPs (cf. Section 3.2.3). As in this context controls are implicit, Todorov's algorithm operates in the state space only and Todorov argues it is this fact, that leads to the improvement over $\mathcal{Q}$-Learning. However, Ψ-Learning yields comparable improvements despite working in the product space of states and actions, as necessitated by considering the unrestricted SOC problem. Based on the previous discussion, we propose that these improvements are due to Ψ-Learning – and implicitly Todorov's algorithm – taking current uncertainty in the estimate into account.

4.3.2 Parametric Policy Search

Natural Gradient Descent:

We first examine the connection between Ψ-Learning and the natural policy gradient descent approach, which, amongst others, underlies eNAC. Consider the case of optimisation over some parametrised familiy of policies $\{\pi_\theta; \theta \in \Theta\}$. This

can be seen as a special case of Proposition 4.8, leading to the iteration

$$\theta^{n+1} = \operatorname*{argmin}_{\theta\in\Theta} \mathrm{KL}\left(q_{\pi_\theta} \| p_{\pi_{\theta^n}}\right)$$

The KL divergence can, by (3.4a), be expanded to yield a problem of the following form

$$\operatorname*{argmin}_{\theta\in\Theta} \mathrm{KL}\left(q_{\pi_\theta} \| p_{\pi^n}\right) = \operatorname*{argmin}_{\theta\in\Theta} \eta\mathcal{J}(\pi) + \mathrm{KL}\left(q_{\pi_\theta} \| q_{\pi^n}\right) \tag{4.22}$$

The algorithms can therefore, as mentioned previously, be interpreted as minimisation of the expected cost under a penalty promoting the solution to be close, in terms of the trajectory distribution, to the current policy. A simple approach to computing an approximate solution of (4.22) is to take first and second order approximations to the expected cost and penalty terms respectively, in which case

$$\begin{aligned}\theta^{n+1} &\approx \operatorname*{argmin}_{\theta\in\Theta} \eta\left(\mathcal{J}(\theta^n) + (\theta-\theta^n)^T\nabla\mathcal{J}(\theta^n)\right) + (\theta-\theta^n)^T\mathbf{M}(\theta-\theta^n) \\ &= \theta^n + \eta\mathbf{M}^{-1}\nabla\mathcal{J}(\theta^n)\end{aligned} \tag{4.23}$$

where

$$\mathbf{M} = \mathbb{E}_{q_{\pi^n}}\left[\nabla \log q_{\pi^n} (\nabla \log q_{\pi^n})^T\right]$$

which we recognise as the Fisher Information metric. The update (4.23) is then exactly the natural policy gradient descent update of Kakade (2001), with η taking the role of the step size.

Relative Entropy Policy Search:

The Relative Entropy Policy Search (REPS) algorithm has been recently suggested by Peters et al. (2010) (also Daniel et al., 2012). Is is based on constrained minimisation of a penalised objective, which can be brought into the form

$$\pi^{n+1} = \frac{1}{\eta} \operatorname*{argmin}_{\pi} \eta\mathcal{J}(\pi) + \mathrm{KL}\left(q_\pi \| q_{\pi^n}\right)$$

under the constrain $\mathrm{KL}\left(q_\pi \| q_{\pi^n}\right) < \epsilon/\eta$ for some fixed ϵ. By Proposition 3.1 minimisation of the penalised objective corresponds directly to minimisation of the KL divergence in Ψ-Learning. Hence for $\epsilon \to \infty$ the REPS update is guaranteed to correspond to the Ψ-Learning update and in practise, if no large steps are

encountered, both algorithms will trace the same trajectory in policy space. We therefore expect that the convergence analysis for Ψ-Learning can be extended to the case of REPS for which, so far, no such results have been given.

Finally, note that REPS has been developed in the context of the average cost objective (cf. Table 2.1) and in the case of none tabular RL an explicit numerical minimisation, rather then the here proposed least squares projection was used.

Dynamic Policy Programming:

The concurrently introduced Dynamic Policy Programming (DPP) algorithm of Azar et al. (2011) is closely related to the formalism described here. Specifically, while the update equations (4.12) coincide, we provide a more general view of DPP by deriving it as a special case of the novel result in Proposition 4.1. In addition, Section 4.1.1 provides the direct extension of DPP to finite horizon problems, while the convergence proofs of Section 4.1.3 extend those given by Azar et al. to continuous state and action spaces.

4.4 Discussion

We have extended our understanding of the previously introduced general duality, by demonstrating that, a suitable relaxation gives rise to iterative solutions of SOC problems. We also analyse the asymptotic behaviour of the proposed iterations, showing their global convergence.

Interestingly, several independently proposed algorithms can be summarised as instances of this approach. In these cases, our formulation provides a novel motivation for these algorithms. Furthermore, our convergence results can be directly applied to these methods. This either extends previous results or gives the first results concerning asymptotic behaviour for these algorithms.

While we have demonstrated the applicability of the proposed formulation to RL problems, their application to model based SOC remains challenging. The arising expressions, Proposition 4.3 and (4.11), provide little scope for direct application of approximate inference techniques. In the following chapter we therefore turn to alternative relaxations of the duality which address this problem.

Finally, we would like to note that the conditions imposed by Proposition 4.1, in order to guarantee a policy improvement, are relatively weak and while we have mainly explore strong versions, i.e., full minimisations, the results opens up the opportunity for weaker forms of iterations to be explored in the future.

Chapter 5

Posterior Policy Iteration

At the beginning of Chapter 4, we observed that the general duality Corollary 3.1 gives rise to minimisation of a KL divergence under two constraints. By relaxing one of these, we were able to derive the Ψ-Learning methods. While these have attractive properties, they do not directly lead to an inference problem and hence, limit the application of Machine Learning techniques. We therefore now turn to relaxation of the second of these constraints, leading to iterative solutions based on a standard inference problem.

Our aims are two fold. On the one hand, we establish a connection between the obtained iterations and SOC, by means of risk seeking control. On the other hand, we establish a connection with the class of MAP Estimation based dualities discussed in Section 3.2.1. Combining these results provides us with a formal interpretation of the policies obtained by the latter approaches as a type of optimal risk seeking policies. Importantly, it also highlights the possibility of improving their results at negligible computational cost by adjustment of the risk seeking behaviour. We confirm our observations empirically on, both, a standard benchmark – the Cart-Pole swing up task – and on a robotic manipulator.

5.1 Formulation

Recall that our general duality (Corollary 3.1) results in a problem of the form

$$\underset{\pi \in \mathcal{D}}{\operatorname{argmin}} \operatorname{KL}\left(q_\pi(\bar{\mathbf{x}}, \bar{\mathbf{u}}) \| p_{\pi^0}(\bar{\mathbf{x}}, \bar{\mathbf{u}})\right) \ ,$$

where $\mathcal{D}$ are deterministic policies. Note that, q_π is constrained both by it's form as the trajectory distribution of the problem dynamics and by the restriction to deterministic policies. We now relax both of these constraints and consider the problem of minimising $\mathrm{KL}\left(q(\bar{\mathbf{x}}, \bar{\mathbf{u}}) \| p_{\pi^0}(\bar{\mathbf{x}}, \bar{\mathbf{u}})\right)$ with respect to an arbitrary distribution q. It is well known that the minimum is given by $q^* = p_{\pi^0}$, which implicitly defines the posterior policy,

$$\begin{aligned}\pi(\mathbf{u}_k|\mathbf{x}_k) &= p_{\pi^0}(\mathbf{u}_k|\mathbf{x}_k) \\ &\propto \pi^0(\mathbf{u}_k|\mathbf{x}_k)\mathbb{E}_{p_{\pi^0}(\bar{\mathbf{x}},\bar{\mathbf{u}}|\mathbf{x}_k,\mathbf{u}_k)}\left[\prod_{k'=k}^{K} P(r_{k'}|\mathbf{x}_{k'}, \mathbf{u}_{k'})\right] . \end{aligned} \tag{5.1}$$

As in the case of the relaxation leading to Ψ-Learning (cf. Chapter 4) we now turn to iteration of the updates, thus obtaining the PPI procedure

$$\pi^{n+1}(\mathbf{u}_k|\mathbf{x}_k) = p^n_\pi(\mathbf{u}_k|\mathbf{x}_k) . \tag{5.2}$$

The question naturally arising is whether such an iteration leads to a policy which optimises an interpretable objective. Informally, we may think of π as the parameter of the model and interpret the procedure as iterated Bayesian inference for the sequence of pseudo observation $\bar{r}^n = 1$ at iteration n. Under such an interpretation, we may expect π to converge to the maximum (marginal) likelihood parameter, i.e.,

$$\pi^n \to \operatorname*{argmax}_\pi \int_{\bar{\mathbf{x}},\bar{\mathbf{u}}} P(\bar{r} = 1, \bar{\mathbf{x}}, \bar{\mathbf{u}}|\pi) . \tag{5.3}$$

Indeed in Appendix D, we show this to be the case for MDPs.

We may now connect the policy arising from PPI to SOC, by observing that

$$\operatorname*{argmax}_\pi \int_{\bar{\mathbf{x}},\bar{\mathbf{u}}} P(\bar{r} = 1, \bar{\mathbf{x}}, \bar{\mathbf{u}}|\pi) = \operatorname*{argmin}_\pi \underbrace{-\frac{1}{\eta}\log \mathbb{E}_{q_\pi}\left[\exp\{-\eta\mathcal{C}(\bar{\mathbf{x}}, \bar{\mathbf{u}})\right]}_{=\mathcal{J}_\eta(\pi)}$$

Thus, the objective being minimized is exactly the risk sensitive objective of Marcus et al. (1997), which has been recently also used in the path integral approach to SOC by Broek et al. (2010). The risk sensitive nature can be made intuitive. Specifically, taking first order series expansions to log and exp around $\eta = 0$ yields

$$\mathcal{J}_\eta(\pi) = \mathbb{E}_{q_\pi}\left[\mathcal{C}(\bar{\mathbf{x}}, \bar{\mathbf{u}})\right] - \frac{\eta}{2}\big(\underbrace{\mathbb{E}_{q_\pi}\left[\mathcal{C}(\bar{\mathbf{x}}, \bar{\mathbf{u}})^2\right] - \mathbb{E}_{q_\pi}\left[\mathcal{C}(\bar{\mathbf{x}}, \bar{\mathbf{u}})\right]^2}_{=\mathrm{Var}(\mathcal{C}(\bar{\mathbf{x}},\bar{\mathbf{u}}))}\big) + \mathcal{O}(\eta^2) .$$

Hence, $\mathcal{J}_\eta$ constitutes a trade off between expected cost and the cost variance. In particular, as $\eta > 0$, for increasing η, optimisation of $\mathcal{J}_\eta$ aims to minimise the cost expectation while maximising the variance. That is the controls are risk seeking – aiming for occasional excellent results over a better average. Furthermore, there exist two conditions under which the classical SOC problem with risk neutral objective $\mathcal{J}$ is recovered

- when the limit $\eta \to 0$ is taken
- when the problem is deterministic

5.2 Relation to Previous Approaches

Ψ-Learning

We can make a connection to the iterations of Ψ-Learning by writing the policy update (5.2) in a Boltzmann form as

$$\pi^{n+1}(\mathbf{u}_k|\mathbf{x}_k) = \exp\{Z_k^{n+1}(\mathbf{x}_k, \mathbf{u}_k) - \bar{Z}_k^{n+1}(\mathbf{x}_k)\}$$

with energy

$$Z_k^{n+1}(\mathbf{x}_k, \mathbf{u}_k) = \log \pi^n(\mathbf{u}_k|\mathbf{x}_k) - \eta\mathcal{C}(\mathbf{x}_k, \mathbf{u}_k) + \log \mathbb{E}_{\mathbf{x}_{k+1}|\mathbf{x}_k,\mathbf{u}_k}\left[e^{\bar{Z}_{k+1}^{n+1}(\mathbf{x}_{k+1})}\right]$$

and log partition function $\bar{Z}_k^n = \log \int_{\mathbf{u}} \exp\{Z_k^n(\mathbf{x}_k, \mathbf{u})\}$. Recall that in the finite horizon case, the Ψ-Iterations also yielded a Boltzmann policy with energy (cf. Proposition 4.3)

$$\Psi_k^{n+1}(\mathbf{x}_k, \mathbf{u}_k) = \log \pi^n(\mathbf{u}_k|\mathbf{x}_k) - \eta\mathcal{C}(\mathbf{x}_k, \mathbf{u}_k) + \mathbb{E}_{\mathbf{x}_{k+1}|\mathbf{x}_k,\mathbf{u}_k}\left[\bar{\Psi}_{k+1}^{n+1}(\mathbf{x}_{k+1})\right]$$

We may now observe that the only difference lies in the log-exp transform in the final term. The log and exp cancel if the dynamics $P(\mathbf{x}_{k+1}|\mathbf{x}_k, \mathbf{u}_k)$ are deterministic. This is to be expected, from the previously made observations, since in this case the risk sensitive controls collapse to the standards risk neutral SOC controls.

Maximum a Posteriori Estimation Dualities

As (5.3) suggests, PPI is closely related to the MAP dualities discussed in Section 3.2.1. In particular, take an MDP problem with the natural parametrisations of a policy by a $|\mathcal{X}| \times |\mathcal{U}|$ matrix. Then (5.2) corresponds to the EM procedure of Toussaint and Storkey (2006) (and similarly, Barber and Furmston, 2009), combining the E- and M-Step into a single operation. The difference lies in the task liklihoods used – Toussaint and Storkey use the untransformed reward (cf. (3.8)), while PPI's model uses exponentiated negative cost (cf. (3.1)).

A perhaps more interesting insight afforded by the above discussion of PPI concerns the MAP dualities of Attias (2003), Raiko and Tornio (2005) and Toussaint (2009b). Recall from Section 3.2.1 that these lacked an interpretation within standard SOC. To our knowledge, we are the first to highlight that the objectives optimised by these methods can be related to risk sensitive control. In particular recall that the AICO formulation of Toussaint coincides with the model underlying PPI in the specific case $\eta = 1$. As AICO seeks the MAP control trajectory[1], it can now be seen as finding the optimal open loop policy with a risk seeking objective. As such, we may see PPI as the natural extension of AICO to feedback policies. Furthermore, our insight suggest that the solution obtained by AICO can trivially be improved by setting $\eta < 1$.

5.3 Algorithms

We now turn to the practical implementation of PPI. Our aim here is not to present a novel inference method – we propose a novel methodology in Chapter 6 – but rather to use the framework provided by AICO. The latter has proven successful in robotic applications (e.g., Toussaint et al., 2010a) and has seen a number of extensions (e.g., Toussaint, 2009a; Ivan et al., 2013) which lead to interesting applications in the robotics domain. Our motivation is to verify insights from the above discussion and apply them to improve the results obtained

[1] n.b. AICO is formulated to seek the MAP *state* trajectory by Toussaint (2009b), however, as the system is assumed to be control LQ, seeking the MAP *state* or *control* trajectory is equivalent.

by these approaches.

With this in mind, our presentation follows that of AICO by Toussaint (2009b), adapting it to the PPI setting as necessary. Specifically, we begin with the formulation of a message passing based solution in the LQG case, subsequently extending it to the general case by approximate message passing. We also briefly discuss the potential application of alternative inference methods.

5.3.1 Linear-Quadratic-Gaussian Problems

Classically, the LQG case plays an important role as a perturbation model and as an ingredient in iterative solution methods for both inference problems in dynamical systems and control problems. In the following, we shall therefore outline the derivation of the closed form solution to (5.2) in this particular case. This will, on the one hand, allow us to illustrate the results in a concrete setting and will further serve as the basis of the subsequently proposed approximations.

Recall from Example 2.1 that, in the LQG case, the control process is linear with Gaussian noise,

$$P(\mathbf{x}_{k+1}|\mathbf{x}_k, \mathbf{u}_k) = \mathcal{N}(\mathbf{x}_{k+1}|\mathbf{f}_k + \mathbf{F}_k^x \mathbf{x}_k + \mathbf{F}_k^u \mathbf{u}_k, \mathbf{Q}_k) \ ,$$

while the costs take the form[2],

$$\mathcal{C}(\mathbf{x}_k, \mathbf{u}_k) = \mathbf{x}_k^T \mathbf{C}_k^x \mathbf{x}_k - 2\mathbf{c}_k^{xT} \mathbf{x}_k + \mathbf{u}_k^T \mathbf{C}_k^u \mathbf{u}_k - 2\mathbf{c}_k^{uT} \mathbf{u}_k \ . \tag{5.4}$$

For notational convenience, in the following we shall drop the subscript $_k$ for $\mathbf{f}$, $\mathbf{F}^x$, $\mathbf{F}^u$, $\mathbf{Q}$, $\mathbf{C}^x$, $\mathbf{c}^x$, $\mathbf{C}^u$ and $\mathbf{c}^u$ if and only if we refer to time k.

With a cost of the form (5.4), the task likelihood takes the convenient form of a product of Gaussians, specifically,[3]

$$P(r_k = 1|\mathbf{x}_k, \mathbf{u}_k) \propto \mathcal{N}[\mathbf{x}_k|\eta\mathbf{c}^x, \eta\mathbf{C}^x]\mathcal{N}[\mathbf{u}_k|\eta\mathbf{c}^u, \eta\mathbf{C}^u] \ .$$

[2] n.b., without loss of generality with respect to Example 2.1, the factor 2 is introduced for convenience of the following derivation

[3] n.b., we use $\mathcal{N}[x|a, A]$ to denote the Gaussian in canonical form, that is,

$$\mathcal{N}[\mathbf{x}|\mathbf{a}, \mathbf{A}] = \frac{\exp\{-\frac{1}{2}\mathbf{a}^T\mathbf{A}^{-1}\mathbf{a}\}}{|2\pi\mathbf{A}^{-1}|^{\frac{1}{2}}} \exp\{-\frac{1}{2}\mathbf{x}^T\mathbf{A}\mathbf{x} + \mathbf{a}^T\mathbf{x}\} \ .$$

As further, assuming a Gaussian policy $\pi^n = \mathcal{N}(\mathbf{u}_k|\mathbf{l}_k + \mathbf{L}_k\mathbf{x}_k, \mathbf{W}_k)$, the dynamics prior is a Gaussian process, all random variables are jointly Gaussian and exact inference is trivial using a Kalman smoothing technique. With the extension to non-linear problems in mind, it is convenient to express the solution in terms of a message passing procedure like belief propagation (Yedidia et al., 2003). In particular, we may write the state marginals as

$$p_{\pi^n}(\mathbf{x}_k) \propto \mu_{x_{k-1}\to\mathbf{x}_k}(\mathbf{x}_k)\mu_{r_k\to\mathbf{x}_k}(\mathbf{x}_k)\mu_{\mathbf{x}_{k+1}\to\mathbf{x}_k}(\mathbf{x}_k) \ , \tag{5.5}$$

where the messages are

$$\begin{aligned}
\mu_{x_{k-1}\to\mathbf{x}_k}(\mathbf{x}_k) &= \int_{x_{k-1},u_{k-1}} \mu_{x_{k-2}\to x_{k-1}}(x_{k-1})\mu_{r_{k-1}\to x_{k-1}}(x_{k-1})\nu(\mathbf{x}_k|\mathbf{x}_{k-1}) \\
\mu_{\mathbf{x}_{k+1}\to\mathbf{x}_k}(\mathbf{x}_k) &= \int_{\mathbf{x}_{k+1},\mathbf{u}_k} \nu(\mathbf{x}_{k+1}|\mathbf{x}_k)\mu_{x_{k+2}\to\mathbf{x}_{k+1}}(\mathbf{x}_{k+1})\mu_{r_{k+1}\to\mathbf{x}_{k+1}}(\mathbf{x}_{k+1}) \\
\mu_{r_k\to\mathbf{x}_k}(\mathbf{x}_k) &= P(r_k|\mathbf{x}_k) \ .
\end{aligned}$$

Here, ν are the control marginal dynamics given by

$$\begin{aligned}
\nu(\mathbf{x}_{k+1}|\mathbf{x}_k) &= \int_{\mathbf{u}_k} P(\mathbf{x}_{k+1}|\mathbf{x}_k,\mathbf{u}_k)\pi^n(\mathbf{u}_k|\mathbf{x}_k)P(r_k|\mathbf{u}_k) \\
&= \mathcal{N}(\mathbf{x}_{k+1}|\hat{\mathbf{f}} + \hat{\mathbf{F}}\mathbf{x}_k, \mathbf{Q} + \mathbf{F}^u\hat{\mathbf{H}}^{-1}\mathbf{F}^{uT}) \ ,
\end{aligned}$$

with

$$\begin{aligned}
\hat{\mathbf{F}} &= \mathbf{F}^x + \mathbf{F}^u(\mathbf{I} + \eta\mathbf{C}^u\mathbf{W})^{-1}\mathbf{L} \\
\hat{\mathbf{f}} &= \mathbf{f} + \mathbf{F}^u(\mathbf{I} + \eta\mathbf{C}^u\mathbf{W})^{-1}\mathbf{l} \\
\hat{\mathbf{H}} &= \mathbf{W}^{-1} + \eta\mathbf{C}^u \ .
\end{aligned}$$

We omit the straightforward, though tedious, derivation of the explicit form of these messages[4] and we summarise the results as

$$\mu_{\mathbf{x}_k\to\mathbf{x}_{k+1}}(\mathbf{x}_{k+1}) = \mathcal{N}[\mathbf{x}_{k+1}|\mathbf{s}_{k+1},\mathbf{S}_{k+1}] \tag{5.6a}$$

$$\begin{aligned}
\mathbf{s}_{k+1} &= \hat{\mathbf{f}} + \hat{\mathbf{F}}(\mathbf{S}_k + \mathbf{C}^x)^{-1}(\mathbf{S}_k\mathbf{s}_k + \mathbf{c}^x) \\
\mathbf{S}_k &= \mathbf{Q} + \mathbf{F}^u\hat{\mathbf{H}}^{-1}\mathbf{F}^{uT} + \hat{\mathbf{F}}(\mathbf{S}_k + \mathbf{C}^x)^{-1}\hat{\mathbf{F}}^T
\end{aligned}$$

$$\mu_{\mathbf{x}_{k+1}\to\mathbf{x}_k}(\mathbf{x}_k) = \mathcal{N}(\mathbf{x}_k|\mathbf{v}_k,\mathbf{V}_k) \tag{5.6b}$$

$$\begin{aligned}
\mathbf{v}_k &= -\hat{\mathbf{F}}^{-1}\hat{\mathbf{f}} + \hat{\mathbf{F}}^{-1}(\mathbf{V}_{k+1}^{-1} + \mathbf{C}_{k+1}^x)^{-1}(\mathbf{V}_{k+1}^{-1}\mathbf{v}_{k+1} + \mathbf{c}_{k+1}^x) \\
\mathbf{V}_k &= \hat{\mathbf{F}}^{-1}[\mathbf{Q} + \mathbf{F}^u\hat{\mathbf{H}}^{-1}\mathbf{F}^{uT} + (\mathbf{V}_{k+1}^{-1} + \mathbf{C}_{k+1}^x)^{-1}]\hat{\mathbf{F}}^{-T} \ .
\end{aligned}$$

[4] n.b., one may proceed along the lines of Toussaint (2009b) or Beal (2003, Chap. 3)

The pair-wise posterior is accordingly given by

$$\begin{aligned} p(\mathbf{x}_{k+1}, \mathbf{x}_k) & \\ & \propto \mu_{x_{t-1}\to\mathbf{x}_k}\mu_{r_k\to\mathbf{x}_k} \cdot \nu(\mathbf{x}_{k+1}|\mathbf{x}_k) \cdot \mu_{r_k\to\mathbf{x}_{k+1}}\mu_{x_{k+2}\to\mathbf{x}_{k+1}} \\ & = p_{\pi^n}(\mathbf{x}_k)\mathcal{N}[\mathbf{x}_{k+1}|\bar{\mathbf{v}}_{t+1} + \bar{\mathbf{Q}}(\mathbf{A}x_t + \mathbf{a}),\ \bar{\mathbf{V}}_{t+1} + \bar{\mathbf{Q}}] \ , \end{aligned}$$

where $\bar{\mathbf{V}} = \mathbf{V} + \eta\mathbf{C}^x$, $\bar{\mathbf{v}} = \mathbf{v} + \eta\mathbf{c}^x$ and $\bar{\mathbf{Q}} = (\mathbf{Q} + \mathbf{F}^u\hat{\mathbf{H}}^{-1}\mathbf{F}^{uT})^{-1}$. We can thus obtain the joint state-control posterior, from which the posterior policy follows, as

$$p_{\pi^n}(\mathbf{u}_k, \mathbf{x}_k) = p_{\pi^n}(\mathbf{x}_k)\mathcal{N}(\mathbf{u}_k|\mathbf{l}' + \mathbf{L}'\mathbf{x}_k, \mathbf{W}') \ ,$$

with

$$\mathbf{l}' = (\mathbf{B}^T\mathbf{V}_*\mathbf{B} + \hat{\mathbf{H}})^{-1}\mathbf{B}^T\mathbf{V}_*(\bar{\mathbf{V}}_{t+1}^{-1}\bar{\mathbf{v}}_{t+1} - \hat{\mathbf{a}}) \tag{5.7a}$$

$$\mathbf{L}' = -((\mathbf{B}^T\mathbf{V}_*\mathbf{B} + \hat{\mathbf{H}})^{-1}\mathbf{B}^T\mathbf{V}_*\hat{\mathbf{A}} \tag{5.7b}$$

$$\mathbf{W}' = (\mathbf{B}^T\mathbf{V}_*\mathbf{B} + \hat{\mathbf{H}})^{-1} \ , \tag{5.7c}$$

where $\mathbf{V}_* = (\mathbf{Q} + \bar{\mathbf{V}}_{t+1}^{-1})^{-1}$. We, hence, obtain closed form iterations for the posterior policy iteration method, as $\pi^{n+1}(\mathbf{u}_k|\mathbf{x}_k) = \mathcal{N}(u_t|\mathbf{l}' + \mathbf{L}'\mathbf{x}_k, \mathbf{W}')$ is of the same Gaussian form as π^n.

Note that in the case of the LQG problem we may, rather then iterating, calculate the eventual policy π^∞ directly backwards in time. Specifically, although not immediately apparent from (5.7) but consistent with the previous discussion of PPI (cf. (5.1)), if $\pi^n(u_{t'}|x_{t'})$ is fixed for all $t' > t$, $\pi^n(\mathbf{u}_k|\mathbf{x}_k)$ converges to the MAP policy, under the improper prior $\pi^0(\mathbf{u}_k|\mathbf{x}_k) = \mathcal{N}[\mathbf{u}_k|0, 0]$. Hence, π^∞ can be calculated backwards in time using the above recursions.

Remark 5.1 (Relation to Riccati equations)**:** As previously observed by Toussaint (2009b) in the special case of AICO, a close relation exists between the Riccati equations of the LQG problem, i.e., (2.5), and the message equations discussed above. Specifically, let us define the backward pre-message

$$\begin{aligned} \bar{\mu}_{\mathbf{x}_k\to x_{k-1}}(\mathbf{x}_k) &= \mu_{r_k\to\mathbf{x}_k}(\mathbf{x}_k)\mu_{\mathbf{x}_{k+1}\to\mathbf{x}_k}(\mathbf{x}_k) \\ &= \mathcal{N}[\mathbf{x}_k|\underbrace{\mathbf{V}^{-1}\mathbf{v} + \mathbf{r}}_{:=\bar{\mathbf{v}}}, \underbrace{\mathbf{V}^{-1} + \mathbf{R}}_{:=\bar{\mathbf{V}}}] \ . \end{aligned}$$

Applying a Woodbury identity (see, e.g., Petersen and Pedersen, 2008), we can express the pre-messages as

$$\begin{aligned}\bar{\mathbf{V}}_k &= \mathbf{R}_k + (\mathbf{F}_k^{xT} - \mathbf{K})\bar{\mathbf{V}}_{k+1}\mathbf{F}^x \\ \bar{\mathbf{v}}_k &= \mathbf{r}_k + (\mathbf{F}_k^{xT} - \mathbf{K})(\bar{\mathbf{v}}_{k+1} - \bar{\mathbf{V}}_{k+1}\mathbf{f}_k) \ ,\end{aligned}$$

with

$$\mathbf{K} = \mathbf{F}^{xT}\bar{\mathbf{V}}_{k+1}(\bar{\mathbf{V}}_{k+1} + (\mathbf{Q} + \mathbf{F}^u\mathbf{C}^{u-1}\mathbf{F}^{uT})^{-1})^{-1} \ .$$

The similarity of these expressions to the Riccatti equations is immediately apparent. The main differences are the influence of the prior policy and the interaction of the noise covariance with the control cost, which we might expect due to the risk sensitive nature of the computed controls.

Note in particular, that we may informally confirm the observations previously made in the general case regarding recovery of the classical, risk neutral, optimal control solution. Specifically, consider the case of a uniform prior policy by setting $\pi^0(\mathbf{u}_k|\mathbf{x}_k) = \mathcal{N}(\mathbf{u}_k|0,\infty)$. Then, the dependence on the prior policy disappears and we exactly recover the Riccatti equations for either a deterministic system – informally setting $\mathbf{Q} = 0$ – or in the case $\eta \to 0$.

5.3.2 Variational Approximation

Analytical tractability of the posterior p_{π^n} encountered in the LQG case is unfortunately the exception rather then the norm. As such we have to resort to approximate inference methods.

In general, there is a wide range of possibilities for approximate inference in the non-LQG case. Concerning sampling methods, rejection sampling – based on forward simulation of the prior process (as suggested by the path integral methods of Kappen (2005)) – or particle methods – as those suggested in the context of MDPs by Hoffman et al. (2009b) – are an option. Such sample based approaches are particularly relevant in the context of RL. While in Chapter 6 we propose a sample based approach which is also applicable to PPI, here our interest chiefly lies with model based SOC.

The simplest extension to the LQG case presented above is to use extended Kalman methods for iteratively updating Gaussian messages $\mu_{\cdot\to\cdot}$ and the ap-

Algorithm 1: Posterior Policy Iteration (PPI)

Input: start state $\mathbf{x}_0$ functions $\mathbf{f}_k(\cdot)$, $\mathbf{F}^x_k(\cdot)$, $\mathbf{F}^u_k(\cdot)$, $\mathbf{C}^x_k(\cdot)$, $\mathbf{c}^x_k(\cdot)$, $\mathbf{C}^u_k(\cdot)$ convergence rate α, threshold ϵ
Output: policy π,
initialize messages $\mu^{fwd}_0 = \mathcal{N}[x_0, 10^{10}]$, $\mu^{bwd}_{0:K} = \mathcal{N}[0, 0]$, $\mu^{cost}_{0:K} = \mathcal{N}[0, 0]$,
repeat
  for $k = 0 \dots K$ **do** *// forward sweep*
    update μ^{fwd}_k and μ^{bwd}_k using (5.6a) and (5.6b)
    get $\mathbf{f}_k(\hat{\mathbf{x}}_k)$, $\mathbf{F}^x_k(\hat{\mathbf{x}}_k)$, $\mathbf{F}^u_k(\hat{\mathbf{x}}_k)$, $\mathbf{C}^x_k(\hat{\mathbf{x}}_k)$, $\mathbf{c}^x_k(\hat{\mathbf{x}}_k)$, $\mathbf{C}^u_k(\hat{\mathbf{x}}_k)$
    compute the belief $q(x_t)$ using (5.5), let b_t be its mode
    $\hat{\mathbf{x}}_k \leftarrow (1-\alpha)\hat{\mathbf{x}}_k + \alpha b_k$
    if $\|\hat{\mathbf{x}}_k - b_k\|^2 > \epsilon$ then $k \leftarrow k-1$ *// repeat this time slice*
  end for
  for $k = (K-1) \dots 0$ **do** *// backward sweep*
    ..same as above...
  end for
until convergence
extract policy using (5.7) backwards in time

proximate belief $\hat{p}(\mathbf{x}_k)$ given by (5.5). By "extended Kalman" we mean that we decide on a point of linearisation $\hat{\mathbf{x}}_k$ for each k and then, use these to compute approximate Gaussian messages as in extended Kalman smoothing. Practically, this means that, instead of having full access to $P(x'|u, x)$ and $P(r|u, x)$, the only interface we require is that we are able to specify a point of linearisation $\hat{\mathbf{x}}_k$ and the system simulator returns the local system matrices (or vectors) $\mathbf{f}_k(\hat{\mathbf{x}}_k)$, $\mathbf{F}^x_k(\hat{\mathbf{x}}_k)$ $\mathbf{F}^u_k(\hat{\mathbf{x}}_k)$, etc.. We choose the point of linearisation based on the mode of the current belief. The algorithm loops back-and-forth over k; for each k it iterates updating the messages and the belief $\hat{p}(\mathbf{x}_k)$, updating the point of linearisation $\hat{\mathbf{x}}_k$ based on the new belief, and recomputing the system matrices.

While SOC has classically relied on Taylor series truncations for such local approximations (e.g., Jacobson, 1967; Li, 2006), for stochastic systems approximations based on the entire distribution, rather then just the mean, have led to significant improvement in filtering and smoothing applications (e.g., but not

limited to, Hartikainen et al., 2011; Deisenroth and Ohlsson, 2010). The perspective of posterior inference opens up the possibility to leverage these advances. Hartikainen et al. (2011) provide an overview over relevant approaches to such statistical linearisations, mainly based on numerical cubature. Deisenroth and Ohlsson (2010) on the other hand, proposes a Bayesian perspective and suggest approximations based on local inference for the message parameters.

Another alternative is to use Expectation Propagation to update messages. For example, Toussaint (2009a) discusses better approximation of hard task constraints in AICO, namely the case when $P(r_k|\mathbf{x}_k, \mathbf{u}_k)$ is identically zero in prohibited regions of the task space. A typical example for this scenario are collision or joint limit constraints, where classically we would want to associate "infinite" costs with prohibited states $\mathbf{x}$. In our framework, we can define $P(r = 1|x) = 0$ for prohibited states. When otherwise beliefs are represented as Gaussian such task messages imply a truncation of the Gaussian which, as Toussaint shows, can again be approximated by a Gaussian task message.

As a further alternative, we may also use variational approaches like, e.g., the Gaussian Process based variational approximation to SDEs proposed by Archambeau et al. (2007) or the variational natural gradient descent based inference approach of Honkela et al. (2010).

Irrespective of the method used, once the algorithm has converged we obtain a local Gaussian approximation to $p_{\pi^0}(\bar{\mathbf{x}}, \bar{\mathbf{u}})$ with π^0 the improper prior $\mathcal{N}[\mathbf{u}_k|0, 0]$. Now rather then explicitly following the PPI formalism, i.e., computing the (local) posterior policy and repeating inference, we utilise the observations made in the LQG case. Specifically, we compute the eventual policy at convergence directly by backward recursion in k. The general procedure is outlined in algorithm 1 based on the extended Kalman idea discussed above. Note that, except for the final step of policy extraction, this procedure corresponds directly to the AICO algorithm – with η adjusted cost. As such, we may directly make use of the various extensions proposed for the latter (e.g., Zarubin et al., 2012; Ivan et al., 2013).

5.4 Experiments

We evaluate the discussed procedure – specifically algorithm 1 – first in a standard benchmark system and subsequently on a (simulated) robotic manipulator. Our main objective is to verify the prediction regarding behaviour under varying η.

5.4.1 Cart-Pole Swing-up

The Cart-Pole system previously introduced in Example 4.2 is considered. However, rather then balancing, the task is now to perform a swing up – the pendulum has to be moved from a hanging down to an upright position and balanced. The trajectory cost for this task is given by

$$\mathcal{C}(\bar{\mathbf{x}}, \bar{\mathbf{u}}) = \sum_{k=0}^{K} \left(\omega_1 \theta^2 + \omega_2 \dot{\theta}^2 + \omega_3 \mathbf{u}_k^2 \right) \ ,$$

where ω is a vector of weights. The time horizon was $T = 3s$, but note that, since a cost is incurred in each time step for pendulum positions away from rest in the upright position, a rapid swing up followed by holding is encouraged.

In Figure 5.1, we plot the expected costs and the cost variances, both estimated by sampling under the obtained policies, for different values of the parameter η. For reference, we also show the expected cost from the policy obtained using the iLQG algorithm of Li (2006) (also cf. Section 2.3.1) which also computes an approximately optimal linear policy. We first observe that, as predicted η acts to control the risk seeking behaviour of the policy. In particular, for increasing values of η the cost variance increases substantially. Furthermore, we note that the choice of $\eta = 1$, which, as discussed, corresponds to the AICO setting[5], leads to results substantially different from the case of classical (risk neutral) optimal control. However, reducing η leads rapidly to policies obtained by approximate inference which exhibit similar performance to those obtained by classical approximate methods.

[5] n.b., as previously discussed, formally AICO computes open loop controls, here closed loop policies where used

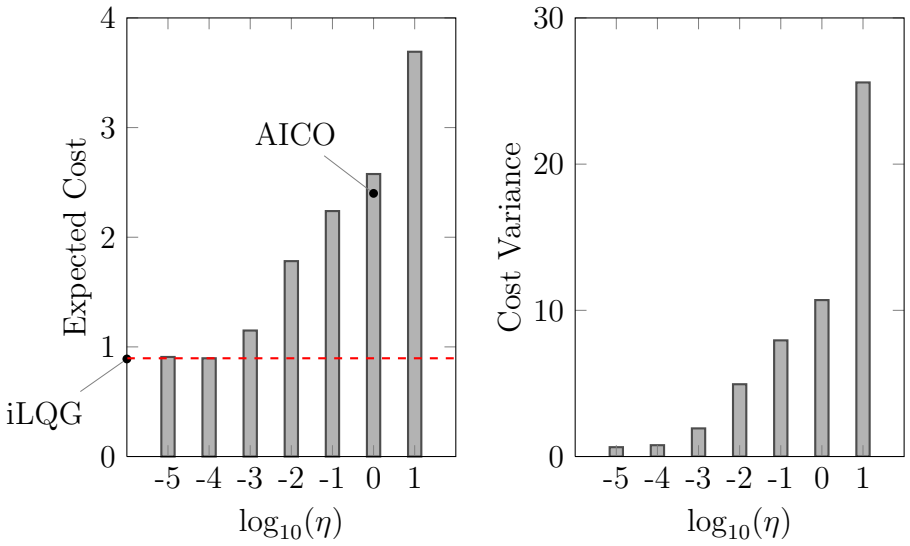
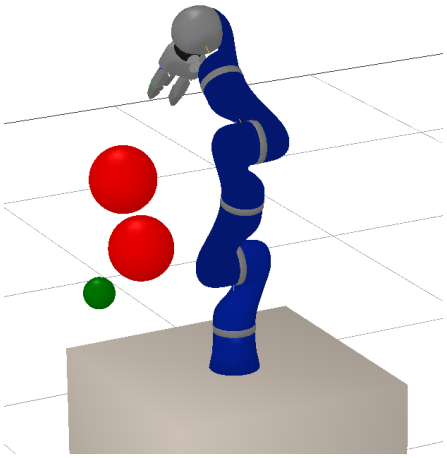


Figure 5.1: Results for PPI on the Cart-Pole swing-up task. **(a)** Expected cost achieved by policies obtained for different values of the parameter η. Red dashed line indicates expected cost of policy obtained using iLQG. All values estimated from 1000 trajectories sampled using the respective policy. **(b)** Variance of the costs achieved by the same policies as in (a).

5.4.2 Robot Manipulator

We now consider planning with a simulation of the 7-DOF Kuka lightweight robot (LWR-III). Both AICO and iLQG have been used for planning with this or very similar manipulator systems by Toussaint et al. (2010a) and Mitrovic et al. (2010b) respectively. Our aim is to first, on a simpler task, demonstrate that PPI improves upon AICO and leads to results competitive with iLQG. We then turn to a more complex task where direct application of either AICO, iLQG or PPI fails. However, exploiting the possibility of combining inference in multiple state representations, as suggested by Zarubin et al. (2012) in the context of AICO, allows solutions to be obtained by AICO and PPI, with the latter again improving upon the former.

The state of the plant is given by $\mathbf{x} = (\mathbf{q}, \dot{\mathbf{q}})$, with $\mathbf{q} \in \mathbb{R}^7$ the joint angles and $\dot{\mathbf{q}} \in \mathbb{R}^7$ the associated angular velocities. The controls $\mathbf{u} \in \mathbb{R}^7$ are the joint



Method	Relative $\mathcal{J}(\pi)$ (median)
PPI ($\eta = 0.001$)	1.000
PPI ($\eta = 0.01$)	0.996
PPI ($\eta = 0.1$)	3.030
PPI ($\eta = 1$)	3.926
PPI ($\eta = 10$)	4.980
iLQG	0.989

Figure 5.2: The simple obstacle task. The manipulator has to reach with it's end-effector to the target ● whilst avoiding the obstacles ●. The task is randomised by sampling both the target and the obstacle positions. The table provides the median over 50 task instances of the expected cost relative to that obtained by PPI($\eta = 0.001$).

space accelerations. We also added some i.i.d. noise with diagonal covariance. The trajectory cost takes the general form

$$\mathcal{C}(\bar{\mathbf{x}}, \bar{\mathbf{u}}) = \sum_{k=0}^{K} \left(\sum_{m=1}^{M} \delta_{k \in \mathcal{K}_m} \|\phi_m(\mathbf{x}_k) - \mathbf{y}_m^*\|_{\Lambda_m}^2 + \mathbf{u}_k^T \mathbf{C}^u \mathbf{u}_k \right) , \tag{5.8}$$

where the tuplets $(\phi_m, \mathcal{K}_m, \Lambda_m, \mathbf{y}_m^*)$ define the task variables, consisting of a task space mapping, a time step set, a diagonal weight matrix and the desired state in task space.

Simple Obstacle Reaching Task

We first consider a standard reaching task with obstacles. The task is defined via the following set of task variables

- **Reaching:** with $\phi_1(\mathbf{x}) \in \mathbb{R}^6$ the arm's end effector position and velocity. The cost is incurred in the final time step only, i.e., $\mathcal{K}_1 = \{K\}$, and $\mathbf{y}^*$ indicates the desired end-effector positions with zero velocities.

- **Joint Limits:** with $\phi_2(\mathbf{x}) \in \mathbb{R}$ a scalar indicating danger of violating joint

limits. Specifically,

$$\phi_2(\mathbf{x}) = \sum_j \mathcal{H}(d_j - \epsilon)^2 \ , \tag{5.9}$$

with d_j the distance to the joint limit of joint j, $\mathcal{H}$ the heavy-side function and margin $\epsilon = 0.1$rad. This task variable is considered throughout the trajectory, i.e. $\mathcal{K}_2 = \{1, 2, \ldots, K\}$.

- **Collisions:** with $\phi_2(\mathbf{x}) \in \mathbb{R}$ a scalar indicating proximity to obstacles. Specifically ϕ_2 takes the general form (5.9) with d_j the shortest distance between a pair j of collidable objects, i.e., the set of links of the arm and obstacles, and margin $\epsilon = 0.02$m. Like the joint limits, this task variable is also considered throughout the trajectory.

Although the resulting finite cost functions can not guarantee that collisions with obstacles or joint limits will not occur (see also the general discussion in Remark 2.3), such approximations are typical in the literature (e.g., Toussaint, 2009b; Ivan et al., 2013) and lead, under appropriate weighting between the reaching and collision components, to good results with low collision probability.

We consider a randomised task with two spherical obstacles, an example configuration being illustrated in Figure 5.2. Specifically, both the target and obstacle positions are randomly sampled, the latter so as to place the obstacles near the direct path to the target to allow them to influence the solution. The table in Figure 5.2 summarise the results. As different task instances can give rise to very different expected costs, we compare expected costs relative to PPI ($\eta = 0.01$), i.e., the improvement of the methods over the baseline given. The expected costs are estimated from sampled trajectories and we consider 50 task instances. We report the median as both the mean and standard deviation become meaningless due to the influence of outliers arising from cases where an individual method gets stuck in a inferior local minimum. As can be seen, similar to the previous Cart-Pole experiment in Section 5.4.1, lowering η from 1 – the value corresponding to AICO – successfully decreases the expected costs (decreases the ratio) and we achieve performance comparable to iLQG.

Complex Obstacle Reaching Task

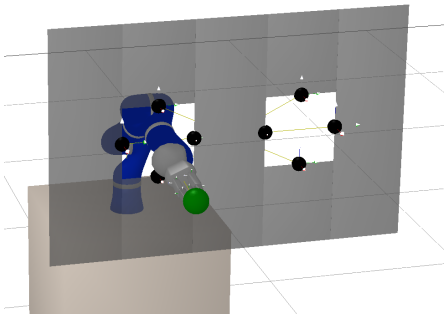
We now consider a generic instance of a task involving manipulation in constrained spaces. It comprises the same basic task variables as used with the simple obstacle above. However, instead of using spherical obstacles we use a wall with two holes as illustrated in Figure 5.3. The end-effector starts reaching through one of the holes and the reaching target lies in the other hole. Due to their local nature direct application of either iLQG or AICO fails in this task. However, in the context of AICO Zarubin et al. (2012) suggest using parallel inference in the normal state space and a abstract topological representations to overcome limitations of local planning in such tasks. With a suitable topological representation the task becomes nearly linear in the alternative representation, which serves to regularise further inference in the plant's state space. Here we use the interaction mesh representation suggested by Zarubin et al., a scale and position invariant representation of relative positions of the plant and markers in the environment. This representation has been used for this task by Ivan et al. (2013) who also used AICO.

We again randomise the task, sampling the position of the wall relative to the manipulator. As again the costs can vary significantly over task instances, we compare the relative expected costs averaged over 50 task instances. These are summarised in the table in Figure 5.3. As can be seen, and as predicted, PPI again improves upon AICO.

5.5 Discussion

We have presented a relaxation of the general duality which, similar to Ψ-Learning in Chapter 4, gives rise to an iterative algorithm we named PPI. Although PPI does not directly give rise to the SOC solution, it can be related to SOC by risk sensitive control. As such, it may be employed to obtain near risk neutral SOC solutions.

In the process we have endowed the MAP estimation based approaches with an novel interpretation within SOC. This connection is of importance as these formulations have formed the basis for application of approximate inference ap-



Method	Relative $\mathcal{J}(\pi)$ (median)
PPI ($\eta = 0.01$)	1.0000
PPI ($\eta = 0.1$)	2.0201
AICO	9.1315

Figure 5.3: The complex obstacle task. The manipulator has to, starting in one whole reach to the target ● whilst avoiding collisions with the wall. The position of the wall is randomised. The table provides the median of the expected cost ratio between the three approaches and PPI ($\eta = 0.01$) over 50 task instances.

proaches in the control problem. Our results lead us to well founded adjustments of these algorithms which improve their results in the SOC sense. This has been confirmed by our experiments.

On a more general note, we would like to emphasise that the solutions provided by PPI – or AICO – do not fundamentally differ from those of iLQG. In fact taking the limit $\eta \to 0$ the (backward) update equations of PPI will reduce to the Ricatti equations underlying iLQG. While Toussaint (2009b) shows that the message passing implementation converges faster – due to updating individual time slice believes till convergence before progressing to the next one – such changes in evaluation order can obviously also be easily incorporated into iLQG. Similarly, the concurrent conditioning on two alternative state representations used by Zarubin et al. (2012) (also Ivan et al., 2013) can easily be incorporated into iLQG or similar iterative local dynamic programming approaches (cf. Section 2.3.1). However, such extensions are, in some sense, natural in the context of inference. Therefore, we see the role of these results – relating SOC and inference – not as the solution to the SOC problem. Rather we consider it as a new perspective which allows us to think about the problem in a different context and provides a guide to new approaches.

Chapter 6

Reproducing Kernel Hilbert Space Embedding

Using the Kalman filtering dualities, we may transform computation of the value function of certain SOC problems into an inference problem. In the more general case, PPI (cf. Chapter 5) allows for a similar transformation to obtain an approximation of the SOC problem. While previously, we concentrated on deterministic approximations, we now turn to sample based approaches. This is facilitated by the observation that value function evaluation can be accomplished by forward integration of the exponential cost weighted dynamics under a certain policy, yielding a path integral.

In the special case of linear dynamics and quadratic costs, the required path integral can be evaluated analytically based on linear operators acting on state vectors. Here, we show that, analogously, a suitable embedding of the path integral into a RKHS allows for it's evaluation in terms of covariance operators acting on elements of the Hilbert space. While this in itself does not yield a tractable solution to the SOC problem, consistent estimators of the required operators give rise to efficient non-parametric algorithms.

The change of perspective, from the direct estimation of the path integral (which previous applications of Monte Carlo methods aimed at) to estimation of operators, allows us to overcome several shortcomings of previous methods, while maintaining many of their advantages. Most importantly, it can significantly reduce the sample complexity by splitting the problem appropriately into

an invariant and task varying component, allowing efficient sample re-use across tasks and leading to a form of transfer learning – contrast this to the situation where any changes in the task including, e.g., different start states, necessitate acquiring new samples (Theodorou et al., 2009, 2010a). Additionally, the approach remains model free, allowing it's application to the Reinforcement Learning setting. This is in contrast to variational (Mensink et al., 2010) or function approximation (Zhong and Todorov, 2011a,b) approaches, from which it is further distinguished through convergence guarantees. The RKHS embedding makes the operators state-dimensionality independent, leading to better scalability, while prior knowledge about both tasks and dynamics can be effectively incorporated by informing choices of sampling procedures and kernels.

Before proceeding, let us briefly state the problem. We will concentrate on the case of the generalised Kalman duality – referred to as *Path Integral Control* in the literature. However the methods proposed are equally applicable in the general case of PPI. Recall from Section 2.2.2, that for a SOC problem of the form

$$\mathrm{d}\mathbf{x} = (f(\mathbf{x}) + \mathbf{F}^u\mathbf{u})\mathrm{d}t + \mathrm{d}\omega \qquad \mathbb{E}\left[\mathrm{d}\omega\mathrm{d}\omega^T\right] = \mathbf{Q}\mathrm{d}t$$
$$\mathcal{C}(\mathbf{x}(\cdot), \mathbf{u}(\cdot)) = \mathcal{C}_T(\mathbf{x}(T)) + \int_0^T \mathcal{C}(\mathbf{x}(t)) + \frac{1}{2}\mathbf{u}(t)^T\mathbf{C}^u\mathbf{u}(t)\mathrm{d}t$$

where f, $\mathbf{F}^u$, $\mathbf{Q}$, $\mathcal{C}$ and $\mathcal{C}_T$ may be non-linear functions and $\lambda\mathbf{F}^u\mathbf{C}^{u-1}\mathbf{F}^{uT} = \mathbf{Q}$ for some $\lambda > 0$, the value function can be related to an un-normalised backward filtering distribution Ψ, by $\mathcal{V}(\mathbf{x}, t) = -\lambda \log \Psi(\mathbf{x}, t)$. Furthermore Ψ satisfies the linear PDE (3.11). As observed by, e.g., Kappen (2011), we may reverse the direction of computation and express Ψ as a forward integration with respect to the diffusion of the state in the dual filtering problem. Specifically, we have the path integral

$$\Psi(\mathbf{x}, t) = \mathbb{E}_{X^0(t\to T)|\mathbf{x}}\left[e^{-\int_t^T \frac{1}{\lambda}\mathcal{C}(X^0(s),s)\mathrm{d}s}\Psi(X^0(T), T)\right] ,$$

with $\Psi(\cdot, T) = \exp\{-\mathcal{C}_T(\cdot)/\lambda\}$ and where by $X^0(t \to T)$ we denote the uncontrolled path of the dynamics of the SOC problem, i.e., under controls $\mathbf{u}(t) = 0$, starting in $\mathbf{x}$. Assuming we are only interested in controls at certain time points, say $\{t_{1,\dots,n}\}$ with $t_n = T$, it is sufficient to compute the set $\Psi_i(x) = \Psi(x, t_i)$ and the path integral admits a representation in terms of the finite dimensional dis-

tribution $X = (X^0(t_0), \cdots, X^0(t_n))$. Specifically using the Markov property of $X^0(t)$ and marginalising intermediate states we obtain the recursive expression

$$\Psi_i(\mathbf{x}_{t_i}) = \mathbb{E}_{X_{i+1}|\mathbf{x}_{t_i}} \left[\Phi_i(\mathbf{x}_{t_i}, X_{i+1}) \cdot \Psi_{i+1}(X_{i+1})\right] \ . \tag{6.1}$$

Here,

$$\Phi_i(\mathbf{x}_{t_i}, \mathbf{x}_{t_{i+1}}) = \mathbb{E}_{X^0(t_i \to t_{i+1})|\mathbf{x}_{t_i}, \mathbf{x}_{t_{i+1}}} \left[e^{-\frac{1}{\lambda}\int_{t_i}^{t_{i+1}} \mathcal{C}(X^0(s), s)\mathrm{d}s}\right] \ , \tag{6.2}$$

where the expectation is taken w.r.t. uncontrolled path from $\mathbf{x}_{t_i}$ to $\mathbf{x}_{t_{i+1}}$. Notice that, (6.1) defines Ψ_i in terms of a linear transformation of Ψ_{i+1}. It is this operator which we estimate based on samples. Also observe that, $-\lambda \log \Phi_i$ can be seen as the (optimal) expected cost for the problem of going from $\mathbf{x}_{t_i}$ to $\mathbf{x}_{t_{i+1}}$ over the time horizon $[t_i, t_{i+1}]$ under dynamics and running costs corresponding to those of the overall problem.

The remainder of this chapter is structured as follows. After introducing the necessary background, we proceed to describe the embedding. We then discuss the issue of obtaining controls based on Ψ, followed by discussion of various extensions to the basic estimator, necessary to make it practically viable. We conclude with illustrative examples of the application of the proposed approach.

6.1 Background

We begin by introducing the necessary concepts to develop our approach. The presentation will be necessarily brief and relatively informal. Berlinet and Thomas-Agnan (2004) provide a thorough treatment of the necessary material, while Hofmann et al. (2008) (or Bishop (2006, Chap. 6)) provide an introduction with a focus on Machine Learning.

6.1.1 Reproducing Kernel Hilbert Spaces

A Hilbert space can be seen as the natural generalisation of a finite dimensional euclidean vector space. It is an abstract vector space – of finite or infinite dimension – endowed with an inner product and the metric generated by said inner product. Our main concern will be with Hilbert spaces of functions $\mathcal{Z} \to \mathbb{R}$, where $\mathcal{Z}$ shall typically be the state space $\mathcal{X}$ of our SOC problem.

A *Reproducing Kernel Hilbert Space*, $\mathcal{H}^k$, of functions $\mathcal{Z} \to \mathbb{R}$, is a Hilbert space, for which there exists a positive semi-definite kernel $k : \mathcal{Z} \times \mathcal{Z} \to \mathbb{R}$, such that

$$\langle h, k(z, \cdot)\rangle = h(z) \quad \forall h \in \mathcal{H}^k, z \in \mathcal{Z}$$

where we write $\langle \cdot, \cdot \rangle$ for the inner product in $\mathcal{H}^k$. This reproducing property has made these spaces extremely useful in Machine Learning. It is commonly employed by defining an embedding of elements $z \in \mathcal{Z}$ into $\mathcal{H}^k$, given by a operator $\mathcal{E}^k : \mathcal{Z} \to \mathcal{H}^k$ such that

$$\left\langle h, \mathcal{E}^k [z]\right\rangle = h(z) \quad \forall h \in \mathcal{H}^k \tag{6.3}$$

From the reproducing property, it is easy to see that $\mathcal{E}^k [z] = k(z, \cdot)$ and more importantly $\left\langle \mathcal{E}^k [z_1], \mathcal{E}^k [z_2]\right\rangle = k(z_1, z_2)$. Hence any algorithm which can be written in terms of inner products in $\mathcal{Z}$ may be lifted to $\mathcal{H}^k$, at the relatively minor expense of kernel evaluations.

6.1.2 Embedding of Distributions

Smola et al. (2007) (but also, Berlinet and Thomas-Agnan, 2004) extend the embedding operator to random variables over $\mathcal{Z}$. Specifically, let $\mathcal{P}^{\mathcal{Z}}$ be the set of random variables on $\mathcal{Z}$ we define the embedding operator $\mathcal{E}^k : \mathcal{P}^{\mathcal{Z}} \to \mathcal{H}^k$ by

$$\left\langle h, \mathcal{E}^k [Z]\right\rangle = \mathbb{E}_Z [h(Z)] \quad \forall Z \in \mathcal{P}^{\mathcal{Z}}, h \in \mathcal{H}^k \tag{6.4}$$

Note that the two operators are consistent, that is, in the case of $Z \sim \delta_z$, where δ_z is a delta distribution, we have $\mathcal{E}^k [Z] = \mathcal{E}^k [z]$. Indeed, using the reproducing property it is easy to show that, analogous to $\mathcal{E}^k [z] = k(z, \cdot)$ in the deterministic case,

$$\mathcal{E}^k [Z] = \mathbb{E}_Z [k(Z, \cdot)]$$

for random variables, i.e. the embedding is given by a mean of Z in the RKHS and is therefore also often referred to as the *mean map*.

Considering the case of a conditional random variable $Z|y$, observe that, for a fixed y, $Z|y$ is a random variable over $\mathcal{Z}$. This allows for a straightforward application of (6.4). However, as $Z|y$ represents a map $\mathcal{Y} \to \mathcal{P}^{\mathcal{Z}}$ – yielding

random variables over $\mathcal{Z}$ given a value $y \in \mathcal{Y}$ – an operator, mapping y into the embedding of $Z|y$ in $\mathcal{H}^k$, becomes of interest. It is convenient to define such an operator in terms of a conditional embedding operator $\mathcal{U}^{lk} : \mathcal{H}^l \to \mathcal{H}^k$ s.t.

$$\mathcal{E}^k [Z|y] = \mathcal{U}^{lk} \circ \mathcal{E}^l [y] \tag{6.5}$$

An explicit form of the operator $\mathcal{U}$ is given by Song et al. (2009) by means of covariance operators, which are generalizations of covariance matrices. Specifically the uncentered covariance operator $\mathcal{C}^{kl}_{ZY}$ for the joint random variable (Z, Y) is given by

$$\mathcal{C}^{kl}_{ZY} = \mathbb{E}_{(Z,Y)} [k(Z, \cdot) \otimes l(Y, \cdot)] \tag{6.6}$$

where $\otimes$ denotes the tensor product. Note that we can see $\mathcal{C}^{kl}_{ZY}$ as an embedding of (Z, Y) into the tensor product space $\mathcal{H}^w = \mathcal{H}^k \otimes \mathcal{H}^l$, which is the RKHS of the product kernel $w((z, y), (z', y')) = k(z, z')l(y, y')$. Now,

$$\mathcal{U}^{lk} = \mathcal{C}^{kl}_{ZY} \left(\mathcal{C}^{ll}_{YY}\right)^{-1} \tag{6.7}$$

satisfies (6.5).

Finite Sample Estimates

Evaluation of $\mathcal{U}$ requires evaluation of expectations of kernels and remains therefore, in most cases, intractable. However as the operators are expressed in terms of expectations, it is straightforward to form empirical estimates, leading to practical algorithms.

Given a set $\mathcal{D} = \{(z, y)_{0...m}\}$ of i.i.d. samples from (Z, Y), an estimate of (6.6) is given by

$$\hat{\mathcal{C}}^{kl}_{\mathcal{D}} = \frac{1}{m} \sum_{i=1}^{m} k(\cdot, z_i) \otimes l(\cdot, y_i) \ . \tag{6.8}$$

Using the latter in conjunction with (6.7), a regularized estimate of $\mathcal{U}^{lk}$ is given by

$$\hat{\mathcal{U}}^{lk}_{\mathcal{D}} = \mathbf{g}^k_{\mathcal{Z}} (\mathbf{G}^l_{\mathcal{Y}\mathcal{Y}} + \epsilon m \mathbf{I})^{-1} \mathbf{g}^l_{\mathcal{Y}} \ , \tag{6.9}$$

where ϵ represents a regularization parameter and $\mathbf{g}^k_{\mathcal{A}}$, $\mathbf{G}^k_{\mathcal{A}\mathcal{B}}$ represents the vector of embeddings and the Gramian respectively, i.e. $[\mathbf{g}^k_{\mathcal{A}}]_i = k(a_i, \cdot)$ and $[\mathbf{G}^k_{(\mathcal{A},\mathcal{B})}]_{ij} = k(a_i, b_j)$, for given sets $\mathcal{A}, \mathcal{B}$ and kernel k. These estimates have been shown to be consistent. Specifically,

Proposition 6.1 (Song et al., 2010)**.** *Assume the operator $\mathcal{C}_{YX}\mathcal{C}_{XX}^{-\frac{3}{2}}$ is Hilbert-Schmidt, then*

$$\|\hat{\mathcal{U}} - \mathcal{U}_{Y|X}\|_{HS} = \mathcal{O}(\epsilon^{\frac{1}{2}} + \epsilon^{-\frac{3}{2}} m^{-\frac{1}{2}}) \tag{6.10}$$

In particular if the regularization term ϵ satisfies $\epsilon \to 0$ and $m\epsilon^3 \to \infty$, then $\|\hat{\mathcal{U}}_{\mathcal{D}}^{lk} - \mathcal{U}^{lk}\|_{HS}$ converges in probability.

6.2 Embedding of the Path Integral

As discussed previously, evaluation of the Ψ_i's can be expressed in terms of recursive application of a linear operator (cf. equation (6.1)). This suggests, that the procedure may be lifted into a RKHS. In this section we first develop the analytical form of this embedding, before presenting empirical estimates, which are shown to be consistent.

6.2.1 Analytical One Step Path Integral Embedding

Assume we have two RKHSs $\mathcal{H}^{\psi}$ and $\mathcal{H}^{\phi}$, such that, $\Psi \in \mathcal{H}^{\psi}$ and $\Phi \in \mathcal{H}^{\phi}$. Note that, $\mathcal{H}^{\phi}$ is a space of functions $\mathbb{R}^{d_x} \times \mathbb{R}^{d_x} \to \mathbb{R}$, while $\mathcal{H}^{\psi}$ contains functions $\mathbb{R}^{d_x} \to \mathbb{R}$. To account for this mismatch in the arity of functions in these spaces, we may trivially, extend $\mathcal{H}^{\psi}$ to $\mathcal{H}^{\psi'}$, a space of functions $\mathbb{R}^{d_x} \times \mathbb{R}^{d_x} \to \mathbb{R}$, using the kernel $\psi'((u,v),(u',v')) = \psi(u,u')$, i.e., we identify $\mathcal{H}^{\psi}$ and its tensor product with the RKHS of constant functions. Unlike h in (6.4), the argument of the expectation in (6.1), specifically of Φ, is not only a function of the random variable, i.e., X_{i+1}, but also of the conditioning $\mathbf{x}_i$, and we can not apply the previous results directly. We therefore proceed by introducing an auxiliary random variable $\tilde{X}$ such that $P(\tilde{X}, X_{i+1}|\mathbf{x}_i) = P(X_{i+1}|\mathbf{x}_i)\delta_{\tilde{X}=\mathbf{x}_i}$ with δ the delta distribution. Hence, taking the embedding of $X_{i+1}, \tilde{X}|x_i$ into $\mathcal{H}^w = \mathcal{H}^{\phi} \otimes \mathcal{H}^{\psi'}$ in which the product function of Φ_i, Ψ_{i+1} resides, using (6.1) and further applying (6.4) and (6.5) we have

$$\begin{aligned}
\Psi_i(\mathbf{x}) &= \mathbb{E}_{X_{i+1}|X_i=\mathbf{x}}\left[\Phi_i(X_{i+1},\mathbf{x}) \cdot \Psi_{i+1}(X_{i+1})\right] \\
&= \left\langle \Phi_i \otimes \Psi_{i+1}, \mathcal{E}^w\left[X_{i+1}, \tilde{X}|X_i = \mathbf{x}\right]\right\rangle \\
&= \left\langle \Phi_i \otimes \Psi_{i+1}, \mathcal{U}^{wk} \circ \mathcal{E}^k\left[\mathbf{x}\right]\right\rangle \ ,
\end{aligned} \tag{6.11}$$

where k is some kernel over $\mathbb{R}^{d_x}$ of our choosing. As will become apparent, for computational reasons, it is convenient to take k to be ψ. This will allow for re-use of pre-computed matrices over the recursive evaluation of estimates of Ψ.

Remark 6.1 (Alternative Embedding)**:** An alternative representation to the embedding (6.11) exists, which, although formally equivalent, yields different empirical estimates. Let us briefly derive the analytical form of this embedding, discussing the empirical estimates further in Remark 6.3.

Observe that for the purposes of the expectation the conditioning variable is fixed and Φ is in fact only a function of it's second argument. This makes it possible to apply (6.5), embedding $X_{i+1}|x_i$ into the tensor space in which the product of Ψ and the partially evaluated Φ resides. Formally define the operator for partial evaluation on $\mathcal{H}^\phi$

$$\mathcal{R}_{\mathbf{x}}[h] = h(\mathbf{x}, \cdot) \quad \forall \mathbf{x} \in \mathbb{R}^{D_{\mathbf{x}}}, h \in \mathcal{H}^\phi$$

In particular, note that $\mathcal{R}_{\mathbf{x}} : \mathcal{H}^\phi \to \mathcal{H}^{\phi_{\mathbf{x}}}$ where $\phi_{\mathbf{x}} = \phi((\mathbf{x}, \cdot), (\mathbf{x}, \cdot))$. We can now write $\Phi_i(\mathbf{x}, \cdot) = \mathcal{R}_{\mathbf{x}}[\Phi_i] \in \mathcal{H}^{\phi_{\mathbf{x}}}$ and application of (6.4) and (6.5) to (6.1) leads to

$$\begin{aligned}
\Psi_i(\mathbf{x}) &= \mathbb{E}_{X_{i+1}|X_i=\mathbf{x}}\left[\mathcal{R}_{\mathbf{x}}[\Phi_i](X_{i+1}) \cdot \Psi_{i+1}(X_{i+1})\right] \\
&= \left\langle \mathcal{R}_{\mathbf{x}}[\Phi_i] \otimes \Psi_{i+1}, \mathcal{E}^w[X_{i+1}|X_i = \mathbf{x}] \right\rangle \\
&= \left\langle \mathcal{R}_{\mathbf{x}}[\Phi_i] \otimes \Psi_{i+1}, \mathcal{U}^{wk} \circ \mathcal{E}^k[\mathbf{x}] \right\rangle
\end{aligned} \tag{6.12}$$

where $\mathcal{H}^w = \mathcal{H}^{\phi_{\mathbf{x}}} \otimes \mathcal{H}^\psi$ and k is some kernel of our choosing on $\mathbb{R}^{D_x}$ which again we take to be ψ.

6.2.2 Finite Sample Estimates

We may now form an empirical estimate of (6.11), based on $\mathcal{D} = \{(x, x')_{1\ldots m}\}$ sampled i.i.d. from a joint distribution $P(X', X) = p_{\pi^0}(X'|X)\mu(X)$, where $p_{\pi^0}(X'|X)$ is the distribution of $X_{i+1}|X_i$ and μ is a free prior. Specifically, assume the representation of Φ_i in $\mathcal{H}^\phi$ is $\mathbf{g}^\phi_{\mathcal{B}}\beta$, which we do not assume to be finite dimensional. Then, given a empirical estimate $\hat{\Psi}_{i+1} = \mathbf{g}^\psi_{\mathcal{A}}\alpha_{i+1}$, based on some set $\mathcal{A}$, we obtain the estimate

$$\hat{\Psi}_i = \mathbf{g}^\psi_{\mathcal{X}}\alpha_i \qquad \text{with} \qquad \alpha_i = \left[\mathbf{G}^\phi_{\mathcal{D}\mathcal{B}}\beta \odot \mathbf{G}^\psi_{\mathcal{X}'\mathcal{A}}\alpha_{i+1}\right]^T \left(\mathbf{G}^\psi_{\mathcal{X}\mathcal{X}} + \epsilon m \mathbf{I}\right)^{-1}, \tag{6.13}$$

where $\odot$ denotes the Hadamard product. The term $\mathbf{G}^{\phi}_{\mathcal{DB}}\beta$ takes – but see Remark 6.2 – the particularly simple form

$$\mathbf{G}^{\phi}_{\mathcal{DB}}\beta = \Phi(\mathcal{X}, \mathcal{X}') = (\Phi(x_1, x_1'), \Phi(x_1, x_2'), \dots)^T \ . \tag{6.14}$$

Hence, obtaining an explicit representation of Φ, or indeed choosing $\mathcal{H}^{\phi}$, is not necessary.

Importantly, note that $\hat{\Psi}_i$ is a finite weighted sum of kernels, hence, $\hat{\Psi}_i \in \mathcal{H}^{\psi}$, which directly allows a recursive computation of all $\hat{\Psi}_{1\dots n}$. Furthermore, all required matrices are functions of the sample data only and as such can be pre-computed. Finally, the estimates are consistent. Specifically,

Proposition 6.2. *Under the assumptions in the main text, the assumptions of Proposition 6.1 and assuming all relevant kernels satisfy $0 \leq k(x, x') \leq 1$, the estimator $\hat{\Psi}_i$ is consistent, i.e., $\|\hat{\Psi}_i - \Psi_i\|_{\mathcal{H}}$ converges in probability.*

Proof. Let $\tilde{\Psi}_i = \hat{\mathcal{U}}^* [\Phi \otimes \Psi_{i+1}]$ where $\hat{\mathcal{U}}^*$ is the adjoint of $\hat{\mathcal{U}}$, i.e., $\tilde{\Psi}_i$ captures the approximation arising due to the empirical embedding. We now use the general relation $\|\mathcal{T}h\|_{\mathcal{H}} \leq \|\mathcal{T}\|_2 \|h\|_{\mathcal{H}} \leq \|\mathcal{T}\|_{HS} \|h\|_{\mathcal{H}}$, where $\mathcal{T}$ is any operator and $\|\cdot\|_2$ and $\|\cdot\|_{HS}$ are the operator and Hilbert-Schmidt norm[1] respectively. This yields the bound

$$\begin{aligned}
\|\tilde{\Psi}_i - \Psi_i\|_{\mathcal{H}} =& \|\hat{\mathcal{U}}^* [\Phi \otimes \Psi_{i+1}] - \mathcal{U}^* [\Phi \otimes \Psi_{i+1}] \|_{\mathcal{H}} \\
\leq& \|\Phi \otimes \Psi_{i+1}\|_{\mathcal{H}} \|\hat{\mathcal{U}}^* - \mathcal{U}^*\|_{HS} \\
=& \|\Phi\|_{\mathcal{H}} \underbrace{\|\Psi_{i+1}\|_{\mathcal{H}} \|\hat{\mathcal{U}}^* - \mathcal{U}^*\|_{HS}}_{=:\epsilon_i} \ .
\end{aligned}$$

Now

$$\begin{aligned}
\|\hat{\Psi}_i - \Psi_i\|_{\mathcal{H}} \leq& \|\hat{\Psi}_i - \tilde{\Psi}_i\|_{\mathcal{H}} + \|\tilde{\Psi}_i - \Psi_i\|_{\mathcal{H}} \\
\leq& \|\hat{\mathcal{U}}^* \left[\Phi \otimes \hat{\Psi}_{i+1}\right] - \hat{\mathcal{U}}^* [\Phi \otimes \Psi_{i+1}] \|_{\mathcal{H}} + \|\Phi\|_{\mathcal{H}} \epsilon_i \\
\leq& \|\Phi \otimes \hat{\Psi}_{i+1} - \Phi \otimes \Psi_{i+1}\|_{\mathcal{H}} \|\hat{\mathcal{U}}^*\|_2 + \|\Phi\|_{\mathcal{H}} \epsilon_i \\
\leq& \|\Phi\|_{\mathcal{H}} \|\hat{\Psi}_{i+1} - \Psi_{i+1}\|_{\mathcal{H}} + \|\Phi\|_{\mathcal{H}} \epsilon_i \ ,
\end{aligned}$$

[1] for an operator $A : \mathcal{H} \to \mathcal{H}'$ the Hilbert-Schmidt norm is defined via

$$\|A\|^2_{HS} = \sum_{i,j=1}^{\infty} \langle \psi_j, A[\phi_i] \rangle^2_{\mathcal{H}'} \ ,$$

where the ψ_j's and ϕ_i's form any complete orthonormal system for $\mathcal{H}$ and $\mathcal{H}'$ respectively.

where in the last line we used $0 \leq k(x, x') \leq 1 \Rightarrow \|\hat{\mathcal{U}}^*\|_2 \leq 1$. Furthermore, using Proposition 6.1 and the union bound, construct ϵ s.t. with probability $1-\delta$ simultaneously for all ϵ_i, $\epsilon_i \leq \epsilon$. The result then follows by induction. ■

Remark 6.2: In (6.14) we make the assumption $\Phi \in \mathcal{H}^\phi$. This is justified as we may choose $\mathcal{H}^\phi$ freely, hence, only require existence of $\mathcal{H}^\phi$. Considering that Φ is an expected cost, we may reasonably expect it to be well behaved. Otherwise, we may discard the problem, as any approach based on approximations will struggle to find an adequate solution. In general, if such a $\mathcal{H}^\phi$ does not exist, we may consider choosing an alternative Φ' close to Φ which is in some $\mathcal{H}^\phi$.

Remark 6.3 (Alternative Embedding)**:** As indicated in Remark 6.1 an alternative form of the embedding exists which leads to different empirical estimates. Specifically, applying (6.9) to (6.12) we obtain $\hat{\Psi}_i(\mathbf{x}) = \mathbf{G}^{\psi}_{\mathbf{x}\mathcal{X}}\alpha(\mathbf{x})$ with

$$\alpha(\mathbf{x}) = \left[\mathbf{G}^{\phi}_{(\mathbf{x}\mathcal{Y})\mathcal{R}}\beta \odot \mathbf{G}^{\psi'}_{\mathcal{Y}\mathcal{X}'}\alpha'\right]^T \left(\mathbf{G}^{\psi}_{\mathcal{X}\mathcal{X}} + \epsilon n\mathbf{I}\right)^{-1} .$$

Hence, although this approach allows us to evaluate $\hat{\Psi}_i$ at specific points, we do not directly obtain a finite dimensional representation of $\hat{\Psi}_i$ in some RKHS. Furthermore, due to the dependence on the evaluation point, the Gram matrix $\mathbf{G}^{\phi}_{(\mathbf{x}\mathcal{X})\mathcal{R}}$ can in general not be pre-computed. None the less, this form may have it's applications for a forward-backwards algorithm where $\mathbf{G}^{\phi}_{(\mathbf{x}\mathcal{X})\mathcal{R}}$ is used for selection of an active set $\mathcal{X}$ for which α's are computed in a backwards pass.

6.3 Computing Controls

As the SOC problem is control LQ, optimal controls can be obtained in closed form as

$$\mathbf{u}^*(\mathbf{x}, t) = -\mathbf{C}^{u-1}\mathbf{F}^u(\mathbf{x})^T\nabla_{\mathbf{x}}\mathcal{V}(\mathbf{x}, t) = \mathbf{C}^{u-1}\mathbf{F}^u(\mathbf{x})^T\frac{\lambda\nabla_{\mathbf{x}}\Psi(\mathbf{x}, t)}{\Psi(\mathbf{x}, t)} .$$

However, while for bounded expected costs, convergence of $\hat{\Psi}$ implies convergence of the estimate of the expected cost (cf. Proposition E.2 in Appendix E), convergence of the latter can be slow for large values due to the log transform. This leads, in practice, to poor policies in regions where Ψ is small. We would like to

emphasize that this problem is not limited to the methods proposed here, but is a characteristic of any approach based on estimation of Ψ, e.g., as also noted by Zhong and Todorov (2011b). To overcome this problem in practice, we form a Laplace approximation to $\hat{\Psi}$ at a local mode. The approximation is used where $\hat{\Psi}$ is small – this corresponds to a local quadratic approximation of the value function, resulting in a linear policy which steers the system towards regions of high $\hat{\Psi}$.

A general problem arising in applications of filtering dualities to RL problems is that while the value function can be estimated on samples, computation of optimal controls requires knowledge of $\mathbf{F}^u$, $\mathbf{C}^u$ and λ. Theodorou et al. (2010a) suggest to only consider deterministic problems and define stochastic pseudo-actuator dynamics so that the overall system has the required form. Specifically, consider a problem with

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}) \tag{6.15}$$

$$\mathcal{C}(\mathbf{x}(\cdot), \mathbf{u}(\cdot)) = \mathcal{C}_T(\mathbf{x}(T)) + \int_0^T \mathcal{C}(\mathbf{x}(t), \mathbf{u}(t))\mathrm{d}t \ . \tag{6.16}$$

We define some stochastic actuator dynamics so that

$$\mathrm{d}\mathbf{u} = g(\mathbf{x}, \mathbf{u}) + \mathbf{F}^{\mathbf{u}'}\mathbf{u}' + \mathrm{d}\omega \qquad \mathbb{E}\left[\mathrm{d}\omega \mathrm{d}\omega^T\right] = \mathbf{Q}(\mathbf{x}, \mathbf{u}) \ . \tag{6.17}$$

If in addition, a cost term of the form $\mathbf{u}'^T\mathbf{C}^{\mathbf{u}'}\mathbf{u}'$ is introduced, the problem with augmented state $\mathbf{x}' = (\mathbf{x}, \mathbf{u})$ and controls $\mathbf{u}'$ is of the required form. Importantly, the underlying deterministic problem can be treated as unknown, as only quantities introduced during augmentation are required for computation of the controls. In our applications of the the presented methods, we have assumed one follows this approach and $\mathbf{F}^u$, $\mathbf{C}^u$ and λ are known.

Remark 6.4: The use of empirical RKHS operators provides an alternative approach to computation of controls in the RL setting. Specifically, we may estimate the embedding of the forward dynamics $\mathcal{E}^{\psi}\left[X_{i+1}|\mathbf{x}_i, \mathbf{u}_i\right]$. The optimal controls can now be found by solving

$$\mathbf{u}^*(\mathbf{x}) = \underset{\mathbf{u}}{\operatorname{argmin}} \left\langle \Phi_i \otimes \hat{\Psi}_{i+1}, \hat{\mathcal{E}}^k_{\mathcal{D}}\left[X_{i+1}|(\mathbf{u}, \mathbf{x})\right] \right\rangle$$

This problem reduces to minimisation of a weighted kernel mixture and which for certain kernels can, as Song et al. (2009) point out, be solved efficiently.

6.4 Efficient Estimators

6.4.1 Low rank Approximation

First, we address the computational complexity of (6.13), which is $\mathcal{O}(m^3)$ for the matrix inversion, which may be precomputed, and $\mathcal{O}(m^2)$ for subsequent computations. Although such costs are acceptable for reasonably sized problems, they may prove prohibitive for application to realistic robotic systems. However, we can apply a Gram-Schmidt orthogonalisation of $\mathbf{g}^k_{\mathcal{X}}$, $\mathbf{g}^k_{\mathcal{X}'}$, as proposed by Song et al. (2011). Summarising we approximate $\mathbf{g}^k_{\mathcal{X}} \approx \mathbf{g}^k_{\mathcal{Y}}\mathbf{W}_x$ and $\mathbf{g}^k_{\mathcal{X}'} \approx \mathbf{g}^k_{\hat{\mathcal{X}}'}\mathbf{W}_{x'}$, where $\mathcal{Y} \subseteq \mathcal{X}$, $\mathcal{Y}' \subseteq \mathcal{X}'$ and $\mathbf{W}_x, \mathbf{W}_{x'}$ are weight matrices. Substituting into (6.13) we may then obtain the alternative estimator

$$\alpha_i = \left[\mathbf{G}^{\phi}_{\mathcal{D}'\mathcal{B}}\beta \odot \mathbf{G}^{\psi}_{\mathcal{Y}'\mathcal{A}}\alpha_{i+1}\right]^T \mathbf{W}_{x'}\mathbf{W}_x^T \left(\mathbf{W}_x\mathbf{W}_x^T + \epsilon m \mathbf{G}^{\psi}_{\mathcal{Y}\mathcal{Y}}{}^{-1}\right)^{-1} \mathbf{G}^k_{\mathcal{Y}\mathcal{Y}}{}^{-1}$$

This is computationally advantageous as with $|\mathcal{Y}| = \hat{m} \ll m$ the complexity reduces to $\mathcal{O}(\hat{m}^3 + \hat{m}^2 m)$ and $\mathcal{O}(\hat{m}^2)$ for required pre-computations and per iteration respectively, often with minimal effects on the obtained results.

More recently, Grünewälder et al. (2012a) propose an alternative low rank approximation, which significantly improves the Gram-Schmidt procedure, further reducing the computation complexity.

6.4.2 Importance Sampling

The estimator (6.13) is based on a sample from the distribution $p_{\pi^0}(X'|X)\mu(X)$. While we are free to choose μ, p_{π^0} is specified by $X_{i+1}|X_i$, i.e. the uncontrolled dynamics. In practice it may be impractical to sample according to the uncontrolled dynamics, e.g., we may wish to improve the policy sequentially collecting new sample following the already learned, rather then the uninformed, policy. To address such situation we follow the importance sampling approach. Specifically note that,

$$\mathcal{C}^{kl}_{ZY} = \mathbb{E}_{(Z',Y')}\left[\frac{P(Z',Y')}{Q(Z',Y')}k(Z,\cdot) \otimes l(Y,\cdot)\right] ,$$

where P, Q are the p.d.f.s of the two joints (Z, Y), (Z', Y') and we assume $Q(z, y) = 0 \Rightarrow P(z, y) = 0$. Hence, given a i.i.d. sample from (Z', Y'), an

empirical estimate of $\mathcal{C}_{ZY}^{kl}$ is given by

$$\hat{\mathcal{C}}_{\mathcal{D}}^{kl} = \sum_{i=1}^{m} w_i k(\cdot, z_i) \otimes l(\cdot, \mathbf{y}_i) \quad \text{, with} \quad w_i = P(z_i, y_i)/Q(z_i, y_i) \ .$$

Such estimates may now be applied to (6.7), in order to obtain an empirical estimate of $\mathcal{U}$. It is easy to show that, when based on a sample from $p_\pi(X'|X)\mu(X)$, where π is an alternative policy, the estimator $\hat{\Psi}_i = \mathbf{g}_{\mathcal{X}}^{\psi}\alpha_i$ with

$$\alpha_i = \left[\mathbf{G}_{\mathcal{D}\mathcal{B}}^{\phi}\beta \odot \mathbf{G}_{\mathcal{X}'\mathcal{A}}^{\psi}\alpha_{i+1}\right]^T \mathbf{W}(\mathbf{G}_{\mathcal{X}\mathcal{X}}^{k} + \epsilon n\mathbf{I})^{-1} \tag{6.18}$$

is obtained. Here, $\mathbf{W}$ is the diagonal matrix with $\mathbf{W}_{ii} = p_{\pi^0}(x_i'|x_i)/p_\pi(x_i'|x_i)$ and we again assume that $p_\pi(x'|x) = 0 \Rightarrow p_{\pi^0}(x'|x) = 0$.

6.4.3 Transfer Learning via Transition Sample Re-use

A limitation of the estimator encountered in practice is the necessity of evaluating Φ at the training transitions (cf. (6.13) and (6.14)). This, in general, may not be viable or feasible. It is therefore desirable to obtain an estimator based on evaluation of Φ on a separate, ideally arbitrary, data set $\mathcal{D}'$. To this end, observe that

$$\begin{aligned}\mathbf{G}_{\mathcal{D}\mathcal{B}}^{\phi}\beta = \langle\Phi, \phi(\mathcal{D}, \cdot)\rangle =& \langle\Phi, \mathcal{C}_{ZZ}^{\phi\phi}\left(\mathcal{C}_{ZZ}^{\phi\phi}\right)^{-1}\phi(\mathcal{D}, \cdot)\rangle \\ \approx& \underbrace{\beta^T\mathbf{G}_{\mathcal{B}\mathcal{D}'}^{\phi}}_{\Phi(\mathcal{D}')}(\mathbf{G}_{\mathcal{D}'\mathcal{D}'}^{\phi} + \epsilon m'\mathbf{I})^{-1}\mathbf{G}_{\mathcal{D}'\mathcal{D}}^{\phi} \ ,\end{aligned}$$

where Z is an some free random variable with support on $\mathbb{R}^{d_x} \times \mathbb{R}^{d_x}$ and we used an empirical estimator based on a data set $\mathcal{D}' = \{(x, x')_{1\ldots m'}\}$ of i.i.d. samples from Z (often in practice $\mathcal{D}' \subseteq \mathcal{D}$). As indicated, evaluation of the final expression only requires evaluation of Φ at elements of $\mathcal{D}'$. Hence, substituting into (6.13) gives the desired result. In particular, we are now able to pre-compute and re-use the inverse matrix of (6.13) across changing tasks and, assuming time stationary dynamics, across different time steps. This is of importance for efficient estimation. For example, in the RL setting incurred costs are known only at observed transitions. Similarly, in cases where Φ can be freely evaluated but it is expensive to do so, while generating large sets of transition samples may

be comparatively cheap, e.g., the case of simple kinematic control where cost evaluation requires collision detection. Note that, this form makes explicit use of the kernel ϕ, and while we may not be able to guarantee $\Phi \in \mathcal{H}^\phi$, by choosing a kernel such that the projection of Φ onto $\mathcal{H}^\phi$ is close to Φ, we can expect good results.

6.4.4 Task augmented sampling

We now turn to the question of the sampling distribution. While in general samples are required from the task agnostic dynamics X^0, a task often induces regularities which suggest more suitable sampling distributions. In particular, considering the role Φ takes in (6.13) as a weight vector, it appears desirable, akin to importance sampling, to concentrate samples in regions of high Φ. Obviously Φ can be used to guide the choice of the prior μ.

Formally, let the task instance be given by a tuple $\{x_0(\theta), \mathcal{C}(x, \theta, t), \mathcal{C}_T(x, \theta)\}$ parametrised by θ, and further assume task instances are picked according to some distribution $P(\theta)$. While the results of Section 6.4.3 allow the re-use of transition samples between different tasks, we are interested in further exploiting regularities induced by the task distribution. We may, for example, choose μ to sample from the time marginal distribution of the diffusion generated by the Fokker Plank equation

$$\nu(x, 0) = \mathbb{E}_{P(\theta)}\left[\delta_{x=x_0(\theta)}\right] \qquad \partial_t \nu = -\frac{\mathbb{E}_{P(\theta)}\left[\mathcal{C}\right]}{\lambda}\nu - \nabla_x\left(f\nu\right) + \nabla_x^2\left(\mathbf{B}\mathbf{B}^T\nu\right)$$

In the context of repeated tasks we can go further and incorporate Φ partly into the sampling process allowing, amongst others, for incremental learning of the task. Consider the specific situation where one wishes to execute several task instances of a generic skill. This situation is often characterised by an invariant cost component relating to the skill and a task specific cost component – taking walking as an example, we wish to stay balanced in each step but the foot placement target will differ from step to step. Formally assume the state cost decomposes as

$$\mathcal{C}(x, \theta, t) = \mathcal{C}_{skill}(\mathbf{x}, t) + \mathcal{C}_{task}(\mathbf{x}, \theta, t) \ , \tag{6.19}$$

where θ parametrises the task. In this case, we may write the path integral as

$$\Psi = \mathbb{E}_{X^\nu(t\to T)|x_t}\left[e^{-\int_t^T \frac{1}{\lambda}\mathcal{C}_{task}(X^\nu(t),\theta,t)}\Psi(X^\nu(T),T)\right] \ . \tag{6.20}$$

The expectation is now taken w.r.t. path of X^ν, which are modified dynamics, absorbing the invariant skill component of the cost. Specifically, they bias the path dynamics based on the Fokker-Plank equation with additional drift term

$$\partial_t \nu = -\frac{\mathcal{C}_{skill}}{\lambda}\nu - \nabla_x\left(f\nu\right) + \nabla_x^2\left(\mathbf{B}\mathbf{B}^T\nu\right) \ . \tag{6.21}$$

In other words, the augmented dynamics tends to restrict the solutions to lie on, or at least stay close to, some skill space.

A practical approach for exploiting the induced structure is to learn the relevant subspace from a few sampled example demonstrations, using, e.g., the approach by Havoutis and Ramamoorthy (2010), and sample $\mathcal{D}$ on the learned space. Such explicit learning of the space has several advantages. Foremost, we can use knowledge of the space to choose an appropriate kernel. Also, while $\mathcal{C}_{task}$ is generally well defined by specific objectives we wish to achieve, the skill component often takes a more abstract form, e.g., we may desire movements to overall appear 'natural', and may only be given implicitly by expert demonstrations of desired movements. In these cases, the proposed framework allows (6.20) to be used to perform optimal control without explicitly referring to the implicit costs.

6.5 Experiments

6.5.1 Double Slit

We first consider the double slit problem, previously studied by Kappen (2005) to demonstrate Monte Carlo approaches to path integral control. The problem is sufficiently simple to allow a closed form solution for Ψ to be obtained, but complex enough to highlight the shortcomings of some previous approaches. The task concerns a particle moving with constant velocity in one coordinate, while noise and controls affects it's position in an orthogonal direction. The aim is to minimise the square error to a target position at some final time, while also avoiding obstacles at some intermediate time, as illustrated in Figure 6.1(a).

Specifically, the one dimensional dynamics are $\mathrm{d}x = u + \mathrm{d}\xi$ and the cost is given by

$$\mathcal{C}_T(x) = \omega(x - x_{target})^2 \quad \text{and} \quad \mathcal{C}(x,t) = \begin{cases} 10^4 & \text{if } t = \frac{T}{2} \text{ and } x \in Obstacle \\ 0 & \text{else} \end{cases}, \tag{6.22}$$

where ω is a weight. We considered a discretisation with time step $0.02s$, i.e. 100 time steps.

We compare the true optimal policy to those obtained using two variants of the proposed estimator, $\hat{\Psi}_{\mathrm{OC}}$ and $\hat{\Psi}_{\mathrm{RL}}$. The latter is based on a Reinforcement learning setting, learning from trajectory data without access to the cost, and uses the approach for sample sharing across time steps discussed in Section 6.4.3. Meanwhile, $\hat{\Psi}_{\mathrm{OC}}$ is based on single transitions from uniformly sampled start states and uses knowledge of the cost function to evaluate Φ in each step. In both cases we use the low rank approximation and square exponential kernels $\psi(x,y) = \exp\{(x-y)^2/\lambda\}$ with λ set to the median distance of the data. For comparison, we also consider two alternative approaches – firstly, the trajectory based Monte Carlo approach of Theodorou et al. (2009), using the same number of trajectories as used in the Reinforcement Learning setting and on the other hand, a variational approximation, specifically a Laplace approximation to the true Ψ to obtain a linear approximation of the optimal policy. As can be seen in Figure 6.1(b), the proposed approach leads to policies which significantly improve upon those based on the alternative Monte Carlo approach and which are comparable to those obtained from the variational approximation. However, the latter, was computed based on knowledge of the true Ψ. In particular, note that the proposed approach makes better use of the sample provided, finding a policy which is applicable for varying starting positions, as illustrated in Figure 6.1(a). As seen from the trajectories in Figure 6.1(a), the Monte Carlo approach on the other hand fails to capture the multi-modality of the optimal policy leading to severely impoverished results when applied to starting point B without sampling a new data set (cf. Figure 6.1(b)). The variational approximation similarly requires re-computation for each new starting location, without which results would also be significantly affected.

To illustrate the dependence of the estimate on the sample size we compare, in Figure 6.1(d), the evolution of the L_1 error of the estimates of Ψ at time $t = 0$. Sample size refers to total number of transition samples seen, hence for $\hat{\Psi}_{\mathrm{RL}}$ the number of trajectories was the sample size divided by 100. In order to also highlight the advantages of the sample re-use afforded by the approach in Section 6.4.3, we also compare with $\hat{\Psi}$, the basic estimator given data of the same form as $\hat{\Psi}_{\mathrm{RL}}$, i.e. recursive application of (6.13) without sample sharing across time steps.

6.5.2 Arm Subspace Reaching Task

We consider reaching tasks on a subspace of the end-effector space of a torque controlled 5dof arm, simulating constrained tasks such as, e.g., drawing on a whiteboard or pushing objects around on a table. Here the skill component consists of moving with the end-effector staying close to a two dimensional task space, while the task instances are given by specific reaching targets. The task space used is a linear subspace of the end effector space – n.b., hence, a non linear subspace of the joint space. The cost comprises the two components

$$\mathcal{C}_{skill}(\mathbf{x}, t) = \omega_{skill}\|\mathbf{J}\varphi(\mathbf{x}) - \mathbf{j}\|^2 \quad \text{and} \quad \mathcal{C}_{task}(\mathbf{x}, \theta) = \omega_{task}\|\varphi(\mathbf{x}) - \theta\|^2 \,, \quad (6.23)$$

where $\varphi(\cdot)$ is the mapping from joint to end-effector coordinates, $\mathbf{J}$ & $\mathbf{j}$ define the task subspace, θ specifies the reaching target and ω's are weights. We again consider position control over a 2s horizon with a 0.02s discretisation.

This task is challenging for sample based approaches as the low cost trajectories are restricted to a small subspace, necessitating large sample sizes to obtain good results for an individual reaching target. This is the case, even if, as suggested by Theodorou et al. (2009) and done here, an inverse dynamics policy is used which significantly improves end-effector exploration. However, concentrating on the case of changing targets, we exploit the ideas from Section 6.4.4 by assuming the operators have been estimated under the skill augmented dynamics[2] (cf. (6.20)), and consider subsequent learning for a novel task using the

[2]n.b., while here such a sample is generated explicitly, the more time consuming approach of using the importance sample based estimator and collecting a sample under X^0 could be used

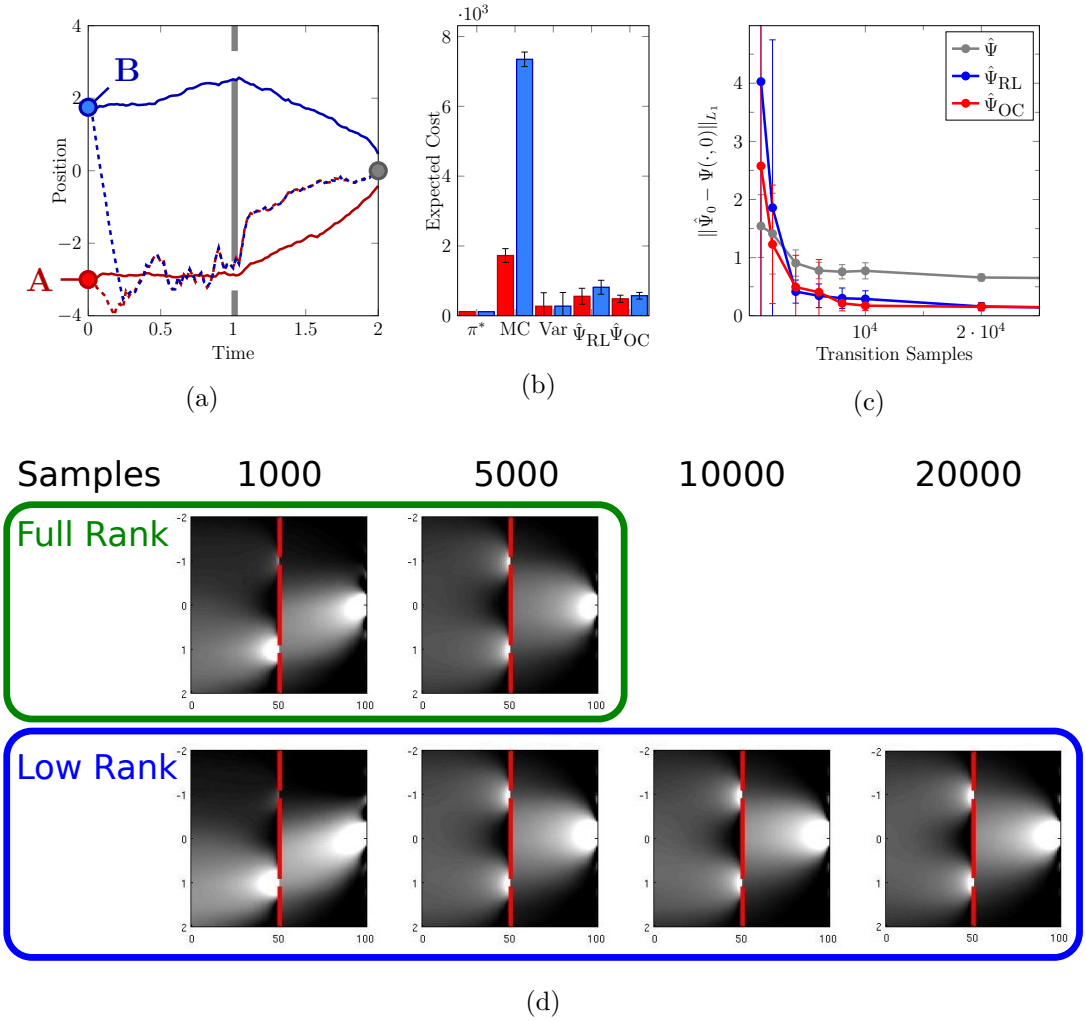


Figure 6.1: Results for the double slit problem. **(a)** Problem setup and mean trajectories from policies MC and $\hat{\Psi}_{\mathsf{RL}}$ for two start points are shown. Obstacles and target are shown in gray. **(b)** Empirical expected cost for policies based on various methods for the two start states. **(c)** The L_1 error of estimates of $\Psi(\cdot, 0)$ as a function of (transition) sample size, n.b. in case of $\hat{\Psi}$ and $\hat{\Psi}_{\mathsf{RL}}$ data was sampled as 100 step trajectories, for various estimators. **(d)** Illustration of the estimates $\hat{\Psi}_{\mathsf{OC}}$ for increasing sample set sizes. *Full Rank* is based on direct application of the estimator derived in Section 6.2.2, *Low Rank* is based on the approximate low rank estimator discussed in Section 6.4.1

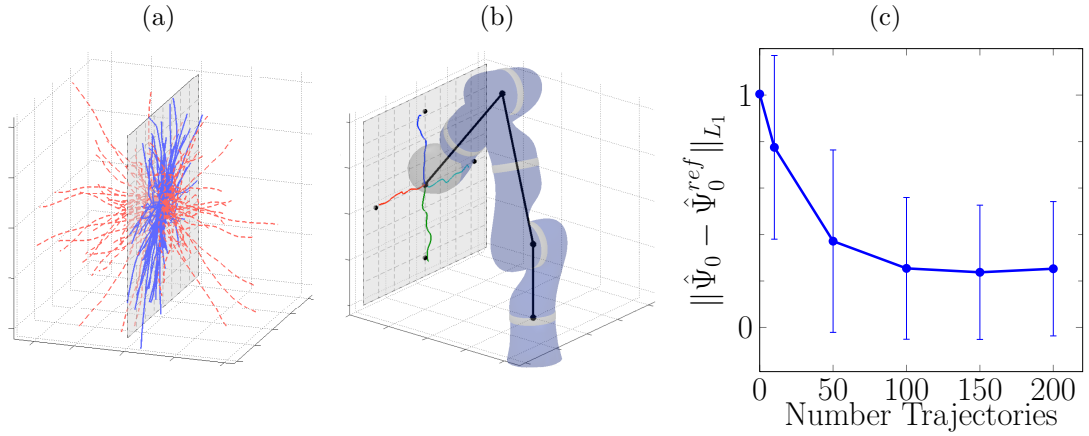


Figure 6.2: Results in the reaching task. **(a)** Training trajectories under the skill augumented policy (solid blue) and π^0 (dashed red) with task space. **(b)** Illustration of the task setup and example trajectories of policies after 100 training trajectories for a set of reaching tasks. The black dots show individual reaching targets with the arm shown in it's initial pose. **(c)** The L_1 error of estimates of $\Psi(\cdot, 0)$ as a function of training trajectories measured with respect to an estimate trained on 5000 trajectories. The data point coresponding to $\#\text{traj} = 0$ is based on the estimate is of $\hat{\Psi}$ taking only $\mathcal{C}_{skill}$ into account (see text for details).

estimator from Section 6.4.3. The already estimated operators are utilised in two ways. On the one hand, they are directly used in the calculation of $\hat{\Psi}$. On the other hand, by sampling under the policy arising when considering $\mathcal{C}_{skill}$ only, i.e., the skill policy associated with $\hat{\Psi}$ computed using the given operators and $\mathcal{C}_{task}(\cdot) = 0$. This is possible, as these trajectories are only required to learn the the cost component, i.e., to obtain $\mathcal{D}'$ in Section 6.4.3, hence do not have to be sampled under a specific policy.

The advantage of sampling under the skill policy is illustrated in Figure 6.2(a), where sample trajectories under both the skill and null policy are shown. It demonstrates that, the former more effectively explores the the task relevant subspace. Mean trajectories for policies learned from 100 trajectories for a set of tasks are illustrated in Figure 6.2(b). In Figure 6.2(c) we plot the L_1 error of $\hat{\Psi}$ as a function of trajectories averaged over ten θ. As the true Ψ is not available for this task we show the error w.r.t. a $\hat{\Psi}$ computed from 5000 trajectories,

principally to illustrate the rapid convergence of the estimator.

6.6 Discussion

We have presented a novel approach for solving stochastic optimal control problems which are of the path integral control form – and related formulations which reduce to an equivalent underlying problem. It uses Monte Carlo estimates of operators arising from a RKHS embedding of the problem, leading to a consistent estimate of Ψ. While direct application of Monte Carlo estimation to point evaluation of Ψ also yields a consistent estimate, it is impractical for computation of controls for anything but simple problems, requiring a trajectory sample for each state at which an action is to be computed. Although previous work, e.g., Theodorou et al. (2009; 2010a), has suggested approaches to overcome the problem of sample complexity, these sacrifice consistency in the process and we demonstrate that the proposed approach significantly improves upon them in terms of generalization of a policy (cf. results in Figure 6.1(a)(b)). We furthermore show that the presented estimators allow for sample re-use in situations which previously required an entirely novel sample set. In particular we consider transfer in cases where execution of several, potentially related, tasks on the same plant is required, demonstrating that it is possible to exploit samples from all tasks to learn invariant aspects.

At present, evaluation of Φ has been performed under the assumption of a fine enough discretisation which allows for approximation by point evaluation of the cost. However, for larger time steps it requires solving a local optimal control problem in itself. Although the proposed approach could be again applied, we note that the arising control problems are local in nature and may be expected to be much simpler. Therefore simpler approaches, e.g., the deterministic methods of Chapter 5, may be sufficient to give good estimates. Hence overall, an alternative perspective on the proposed method is, as a principled approach to combining solutions to local control problem to solve a more complex large scale problem. An approach which warrants further investigation in the future.

A further subject we haven't elaborated on is the choice of kernels. There has been a growing interest in exploiting prior task knowledge to define appropriate

problem representations in which the solution takes a simpler form (e.g., Zarubin et al., 2012). However, choices of such representation have been mainly driven by intuition. In the proposed approach, the choice of kernel corresponds to choosing a representation and furthermore desired properties of the kernel can be easily formalised, e.g., Ψ has a low norm in the RKHS. Thus, the presented approach could provide an avenue to formalise the choice of representations and even allow for learning of these.

Concurrent to our work, Grünewälder et al. (2012b) propose the application of RKHS embeddings of distributions to evaluate the expectation operator in the Bellman equation (2.2). Such an approach has the drawback of not embedding the entire computation. Specifically, the addition of the per step cost to the result of the embedded expectation operator results in the value function lacking a guarantee to remain in the RKHS. This is in contrast to Ψ in our formulation which remains in the RKHS. As a consequence, while we obtain Proposition 6.2, Grünewälder et al. fail to obtain a general consistency result beyond the MDP case.

Chapter 7

Temporal Optimisation

Control of sensorimotor systems, artificial or biological, is inherently both a spatial and temporal process. Not only do we have to specify *where* the plant has to move to but also *when* it reaches that position. In some control schemes the temporal component is implicit. For example, with a infinite horizon, discounted cost based controller movement duration results from the application of the feedback loop. In other cases it is explicit, like for example in finite horizon objective based formulations, where the time horizon is set explicitly as a parameter of the problem.

Although control based on an optimality criterion is certainly attractive, practical approaches for stochastic systems are currently limited to the finite horizon objective, as much of the work in the preceding chapters, or the first exit time objective given in Table 2.1. The former does not optimize temporal aspects of the movement, i.e., duration or the time when costs for specific sub goals of the problem are incurred, assuming them as given *a priori*. However, how should one choose these temporal parameters? This question is non trivial and important even while considering a simple reaching problem. The solution generally employed in practice is to use an *a priori* fixed duration, chosen experimentally. This can result in not reaching the goal, having to use an unrealistic range of control commands or excessive (wasteful) durations for short distance tasks. The alternative first exit time formulation, on the other hand, either assumes specific exit states in the cost function and computes the shortest duration trajectory which fulfils the task or assumes a time stationary task cost function and com-

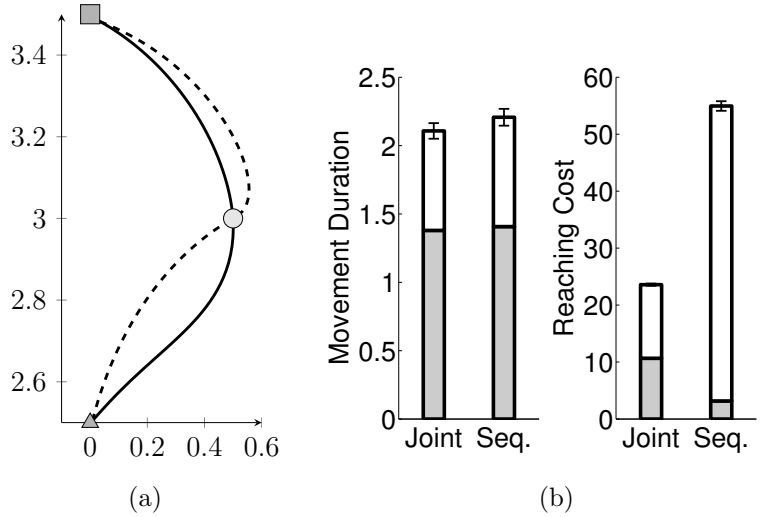


Figure 7.1: Joint (—) vs. sequential (- -) optimisation using our approach on a via-point task as descibed in Example 7.1. **(a)** Task space trajectories for the fixed start point △. Via-point and target are indicated by ○ and □, respectively. **(b)** The movement durations and reaching costs for 10 random start points. The mean proportion of the movement duration spend before the via point is shown in light grey.

putes the control which minimizes the joint cost of movement duration and task cost (Toussaint and Storkey, 2006; Barber and Furmston, 2009; Todorov, 2009a; Kulchenko and Todorov, 2011). This formalism is thus only directly applicable to tasks which do not require sequential achievement of multiple goals. Although this limitation could be overcome by chaining together individual time optimal single goal controllers, such a sequential approach has several drawbacks. First, if we are interested in placing a cost on overall movement duration, we are restricted to linear costs if we wish to remain time optimal. A second, more important, flaw is that future goals should influence our control even before we have achieved the previous goal.

To further motivate temporal optimisation, we highlight it's benefits on the following two concrete examples from our work.

Example 7.1 (Via-Point Reaching)**:** Take the conceptual task of picking up and moving an object to a designated location. Typically, one is only interested in the eventual result and – within reason – the exact duration

of the movement and when the object is picked up are free. Abstracting the details, we consider a via point task with a planar 2-link arm – the end effector has to be moved from a start position to a target, passing through some via point on the way[1]. In order to highlight the shortcomings of sequential time optimal control, we compare planning a complete movement, referred to as joint optimisation, to planning a sequence of individual optimal movements.

Figure 7.1 summarises the results. As can be seen in Figure 7.1(a) the two approaches lead to solutions with substantially different end-effector trajectories in task space. The joint optimisation, accounting for the need to continue to the eventual target after the via-point, yields a different approach angle. The profound effect this has on the incurred cost can be seen in Figure 7.1(b). While joint planning incurs higher cost before the via-point, the overall cost is more then halved. Importantly, as the plot of the movement durations illustrates, this reduction in cost is not achieved by an increase in movement duration, with both approaches leading to not significantly different durations. However, one should note that this effect would be less pronounced if the cost required stopping at the via-point, as it is the velocity away from the end target which is the main problem for the sequential planner.

Example 7.2 (Brachiation[2])**:** We consider the example of the plant in Figure 7.2(a). It consists of a two-link system with grippers on either end and an actuated central joint, moving along a horizontal ladder by brachiating, i.e., swinging between the rungs. Figure 7.2(b) shows an example of a successful movement sequence, including an initial swing-up. This problem is challenging as the system is underactuated – it has fewer actuators then degrees of freedom – and needs to interact with and exploit it's environment, specifically gravity, to achieve good results. Similar tasks have been studied by, e.g., Spong (1995) or Nakanishi et al. (2000), however these approaches generally relied on hand tuned controllers.

Figure 7.2(c)&(d) summarise the effect of temporal optimisation for an individual swing. As Figure 7.2(c) illustrates, small changes in the duration

[1] n.b., the plant and cost functions used were as in the via point tasks in Section 7.3.1

[2] n.b., experiments in this example where implemented and conducted by J. Nakanashi based on collaborative work on their formulation (Nakanishi et al., 2011, see also,)

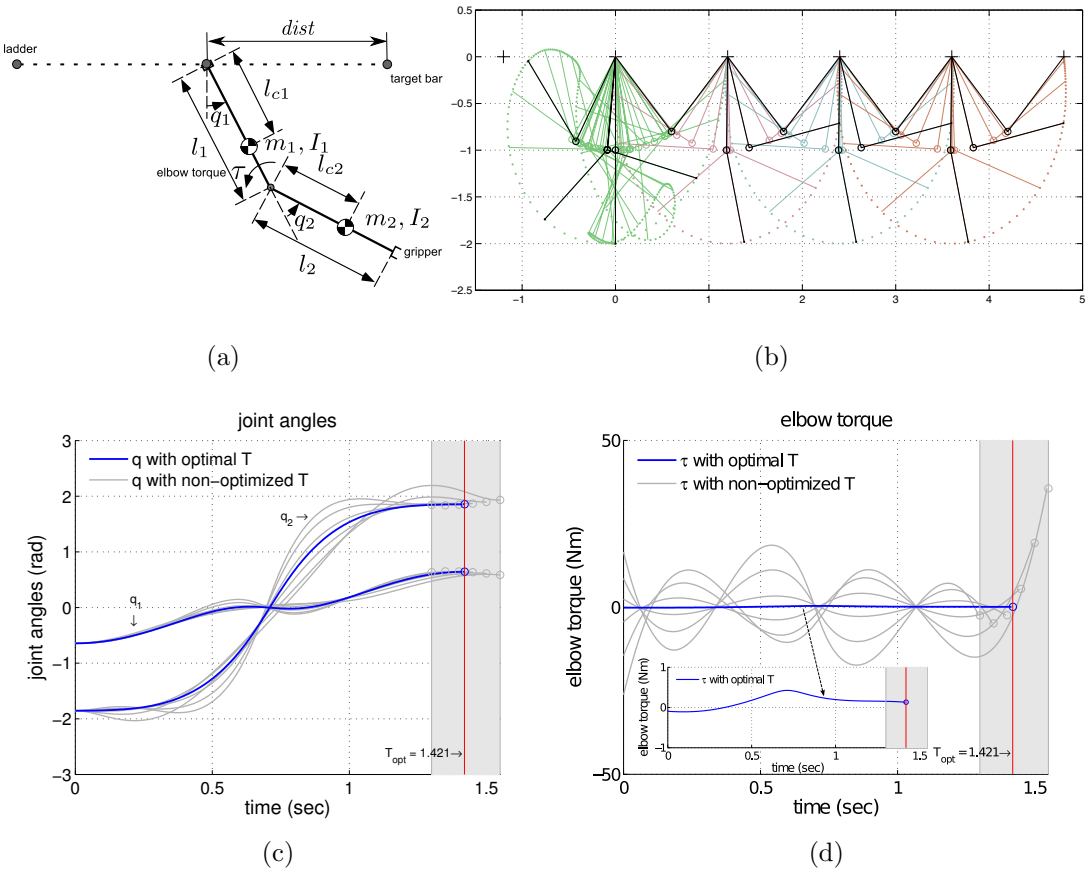


Figure 7.2: Temporal optimization of brachiation as discussed in Example 7.2. **(a)** The plant considered. **(b)** A movement sequence consisting of, starting on the left, a swing-up and three swing locomotions. **(c)** Comparison of the joint angles during an individual swing. Shown are trajectories obtained with temporal optimisation ($T_{\text{opt}} = 1.421(sec)$) and for a range of finite horizon objectives with fixed T in the range $T = [1.3(sec), ..., 1.55(sec)]$. **(d)** Comparison of the elbow torque command trajectories corresponding to the movements in (c). With temporal optimisation, the torque is limited to the interval [-0.107, 0.425] during the entire movement.

have little effect on the joint space trajectory, the task being easily fullfilled for a range of finite horizons. However, the effect on the torque profiles is profound as can be seen in Figure 7.2(d). While our method leads to a movement which exploits gravity, requiring very little active torque, even small deviation from the duration identified by our approach yield significant increases in required

torque.

In the remainder of this chapter we will first introduce our formulation of temporal optimisation, which is based on a generalisation of the finite horizon objective. We then proceed to propose practical extensions to the PPI methods of Chapter 5, before concluding with application of the proposed methods in a series of manipulation task.

7.1 Problem formulation

As indicated previously, the finite horizon objective is commonly used to frame problems in the robotics domain. It does however suffer from it's explicit reference to temporal parameters, in particular the movement duration T. In the following a generalisation is presented which removes this direct reference to T. This will allow for implicit optimisation over time horizons and, subsequently, will provide us with a framework for explicit optimisation of temporal aspects of tasks.

In the following we will be considering the general continuous time system of the form

$$\mathrm{d}\mathbf{x} = f(\mathbf{x}, \mathbf{u})\mathrm{d}t + \mathrm{d}\xi \quad \mathbb{E}\left[\mathrm{d}\xi \mathrm{d}\xi^T\right] = \mathbf{Q} \;, \tag{7.1}$$

with non-linear dynamics f and Brownian motion ξ. We also recall from Chapter 2 that, the standard finite horizon problem takes the form

$$\pi^* = \underset{\pi}{\operatorname{argmin}}\, \mathbb{E}_{\mathbf{x}(\cdot),\mathbf{u}(\cdot)|\mathbf{x}(0),\pi}\Big[\underbrace{\mathcal{C}_T(\mathbf{x}(T)) + \int_0^T \mathcal{C}(\mathbf{x}(t), \mathbf{u}(t), t)\mathrm{d}t}_{=\mathcal{C}(\mathbf{x}(\cdot),\mathbf{u}(\cdot))}\Big] \;, \tag{7.2}$$

where T is a priori fixed time horizon.

7.1.1 Generalised Finite Horizon Problems

The idea underlying the generalisation is to consider problems where the optimisation is performed over some limited quantity. In particular, rather then considering optimisation over a fixed time horizon, we now introduce a none diminishing *resource variable* β which we shall bound. Specifically, assume that β is consumed with a rate h – potentially dependent on state, controls and time –

so that

$$\beta(\mathbf{x}(\cdot), \mathbf{u}(\cdot), t) = \int_0^t h(\mathbf{x}(s), \mathbf{u}(s), s)\mathrm{d}s \qquad h > 0$$

where, without loss of generality, we fix $\beta(\cdot, \cdot, 0) = 0$ and optimisation is performed over a fixed horizon, say β_f, of β. That is we define

$$T = \inf\{t|\beta > \beta_f\}$$

as the resource exhaustion time and use it as horizon in the objective (7.2).

Remark 7.1: Clearly, with the substitution $t \rightsquigarrow \beta$, the standard finite horizon formulation is recovered.

Remark 7.2: The generalised finite horizon formulation is related to the first exit time formulation and represents a special case of this class of problems. To see this define the augmented state variable $\hat{\mathbf{x}} = (\mathbf{x}, \beta)^T$ and dynamics

$$\mathrm{d}\hat{\mathbf{x}} = \begin{bmatrix} f(\mathbf{x}, \mathbf{u}) \\ h(\mathbf{x}, \mathbf{u}, t) \end{bmatrix} \mathrm{d}t + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \mathrm{d}\xi \quad \mathbb{E}\left[\mathrm{d}\xi \mathrm{d}\xi^T\right] = \mathbf{Q}$$

and consider minimisation of the first exit time objective

$$\mathcal{C}(\mathbf{x}(\cdot), \mathbf{u}(\cdot)) = \int_0^{t_{\inf}} \mathcal{C}(\mathbf{x}(t), \mathbf{u}(t), t)\mathrm{d}t \qquad t_{\inf} = \inf\{t; \mathbf{x}(t) \in \mathcal{X}_f\}$$

where the set of exit states is $\mathcal{X}_f = \{\hat{\mathbf{x}} = (\cdot, \beta); \beta = \beta_f\}$

While the generalised problem is seemingly more complex then the classical formulation, we may in fact, under additional reasonable assumption on the resource rate h, obtain a problem of the standard form. To this end the problem is analytically transformed, expressing it relative to the resource variable rather then time. Specifically we define the right-inverse of $\beta(t)$ as

$$\alpha(\mathbf{x}(\cdot), \mathbf{u}(\cdot), s) = \inf\{t|\beta(\mathbf{x}(\cdot), \mathbf{u}(\cdot), t) > s\} \tag{7.3}$$

Using standard results of calculus and the time change theorem for Itô integrals (Øksendal, 2010, Chap. 8.5) we obtain the problem with dynamics

$$\mathrm{d}\mathbf{x} = f(\mathbf{x}, \mathbf{u})\frac{\partial \alpha}{\partial t}\mathrm{d}\beta + \sqrt{\frac{\partial \alpha}{\partial t}}\mathrm{d}\xi \tag{7.4}$$

and objective

$$\mathcal{C}(\mathbf{x}(\cdot), \mathbf{u}(\cdot)) = \int_0^{\beta_f} \mathcal{C}(\mathbf{x}, \mathbf{u}, \alpha(\beta))\frac{\partial \alpha}{\partial t}\mathrm{d}\beta \ , \tag{7.5}$$

where

$$\frac{\partial}{\partial t}\alpha(\mathbf{x}(\cdot),\mathbf{u}(\cdot),t)\Big|_{t=s} = \frac{1}{h(\mathbf{x}(s),\mathbf{u}(s),\alpha(\mathbf{x}(\cdot),\mathbf{u}(\cdot),s))} \ .$$

This problem is now of the standard finite horizon form and can be solved using standard algorithms for this class of problems. However, such solutions will yield policies indexed by β rather then time, although these may be easily transformed to the more familiar time indexed equations using (7.3).

Note that, the above formulation can be seen as the natural generalisation to the stochastic setting, of previously proposed time scaling approaches for trajectory planning (e.g., Sahar and Hollerbach, 1986; Fu et al., 2008).

Remark 7.3: It is worth noting that, while we have shown that the generalised finite horizon problem is not principally harder then the finite horizon problem, the transformation may lead to problems without analytical solution even for those case where the primal dynamics and costs would ordinarily allow for such solutions, e.g., they were of the LQG form.

7.1.2 Explicit Temporal Optimisation

As illustrated, the generalised finite horizon formulation allows implicit optimisation over task durations. In the following we now demonstrate how it also allows for explicit temporal optimisation. Before proceeding, let us first clarify what we refer to by *explicit temporal optimisation*.

In practical robotics applications cost can generally be divided into

- **goals**, these being costs dependent only on state and incurred at specific time instances.
- **skills**, these being costs independent of time incurred throughout the movement and dependent on both states and controls

Thus the trajectory cost in (7.2) takes the form

$$\mathcal{C}(\mathbf{x}(\cdot),\mathbf{u}(\cdot)) = \sum_{t_n\in\mathcal{T}} \mathcal{C}_n(\mathbf{x}(t_n)) + \int_0^{t_f} \mathcal{C}(\mathbf{x}(t),\mathbf{u}(t))\mathrm{d}t \tag{7.6}$$

where $\mathcal{T}$ is a set of time instances at which specific goals, captured by the corresponding $\mathcal{C}_n$'s, are to be fulfilled. For instance, in a reaching movement a cost

which is a function of the distance to the target is incurred only at the final time T, while collision costs are independent of time and incurred throughout the movement. In explicit temporal optimisation, our objective shall be the optimisation the set $\mathcal{T}$. Note that this objective is broader then that of duration optimisation, i.e., choice of T, aimed for by alternative formulations.

To achieve our objective we introduce a *canonical time* t' letting it take the role of the resource variable, i.e., $t' = \beta$, and choose the simple state and control independent rate function $h(\mathbf{x}(\cdot), u(\cdot), t) = 1/\delta(t)$. Formulating the goal related aspects of the costs in canonical time we obtain the problem with dynamics

$$\mathrm{d}\mathbf{x} = f(\mathbf{x}, \mathbf{u})\delta(\alpha(t'))\mathrm{d}t' + \sqrt{\delta(\alpha(t'))}\mathrm{d}\xi \quad \mathbb{E}\left[\mathrm{d}\xi \mathrm{d}\xi^T\right] = \mathbf{Q} \tag{7.7}$$

and cost

$$\mathcal{C}(\mathbf{x}(\cdot), \mathbf{u}(\cdot), \delta(\cdot)) = \sum_{t'_n \in \mathcal{T}'} \mathcal{C}_n(\mathbf{x}(t'_n)) + \int_0^{T'} \mathcal{C}(\mathbf{x}(s), \mathbf{u}(s))\delta(\alpha(s)) + \mathcal{C}_\delta(\delta(\alpha(s)))\mathrm{d}s \ , \tag{7.8}$$

where, in contrast to (7.6), the set $\mathcal{T}' = \{t'_{1\ldots N}\}$ consists of time points in canonical time. While the fulfilment time of the individual costs is now fixed in canonical time, treating $\delta(\cdot)$ as an additional control allows the time point when they are incurred in real time to be varied and in particular optimised. That is, rather then solving for just an optimal policy we solve

$$(\pi^*, \delta^*) = \underset{\pi,\delta}{\operatorname{argmin}}\, \mathbb{E}_{\mathbf{x}(\cdot),\mathbf{u}(\cdot)|\mathbf{x}(0),\pi,\delta}\left[\mathcal{C}(\mathbf{x}(\cdot), \mathbf{u}(\cdot), \delta(\cdot))\right] \ . \tag{7.9}$$

In practise the flexibility offered by the above general formulation is not required, hence we will assume δ to have a parametrised form with parameter set Θ and consider optimisation over these. Specifically it is convenient to, without loss of generality, take $\delta_\Theta(\cdot)$ to be constant between individual goals, i.e.

$$\delta_\Theta(t) = \theta_n/(t'_{n+1} - t'_n) \quad \text{for } t \in [\alpha(t'_n), \alpha(t'_{n+1})]) \ , \tag{7.10}$$

with the parameter set given by individual durations $\Theta = \{\theta_0, \ldots, \theta_{N-1}\}$.

Remark 7.4: The choice of the rate δ to be state and control independent is obviously formally unnecessary in the context of either the generalised finite horizon problem or it's interpretation as a control. Indeed, it leads to

an asymmetry in (7.9), where optimisation for the classical controls $\mathbf{u}$ is over feedback policies, while δ is a open loop policy. Certainly making δ state dependent, hence a feedback policy, would in principle lead to lower expected cost. However we refrain from such a formulation for practical reasons. Foremost, experience has shown that naive optimisation in the augmented problem although possible generally fails, leading to the development of the algorithms described below which are simplified by the choice of an open loop δ.

7.2 Inference based Temporal Optimisation

We now turn to an implementation of temporal optimisation as described above within PPI (cf. Chapter 5). To this end, we discretise the problem in the canonical time t'. The following discussion will be based on the assumption that the resulting problem takes the general form,

$$\begin{aligned} \mathbf{x}_{k+1} &= f(\mathbf{x}_k, \mathbf{u}_k, \Theta) + \epsilon \qquad \epsilon \sim \mathcal{N}(0, \mathbf{Q}(\Theta)) \\ \mathcal{C}(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \Theta) &= \sum_{k_n} \mathcal{C}_n(\mathbf{x}_{k_n}) + \sum_{k=0}^{K} \mathcal{C}(\mathbf{x}_k, \mathbf{u}_k)\delta_k + \mathcal{C}_\delta(\delta_k) \end{aligned} \tag{7.11}$$

although the ideas presented can be easily adapted to alternative forms.

Recall from Chapter 5, that the PPI formulation is based on computation of the posterior

$$P(\Theta, \bar{\mathbf{z}}|\bar{r} = 1) = P(\mathbf{x}_0) \prod_{k=0}^{K} P(\mathbf{x}_{k+1}|\mathbf{x}_k, \mathbf{u}_k, \Theta) \exp\{-\mathcal{C}(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \Theta)\}$$

where $\bar{\mathbf{z}} = (\bar{\mathbf{x}}, \bar{\mathbf{u}})$. From this the MAP policy, and in this case MAP Θ, are extracted. As this problem will in general be intractable, we proceed in two steps

$$\Theta^{\mathrm{MAP}} = \operatorname*{argmax}_{\Theta} \log \int_{\bar{\mathbf{z}}} P(\Theta, \bar{\mathbf{z}}|\bar{r} = 1) \tag{7.12a}$$

$$\pi^{\mathrm{MAP}} = \operatorname*{argmax}_{\pi} \log \int_{\bar{\mathbf{z}}} P(\pi, \bar{\mathbf{z}}|\Theta^{\mathrm{MAP}}, \bar{r} = 1) \tag{7.12b}$$

Note that the second step reduces exactly to standard PPI and may be solved with any of the methods discussed in Chapter 5. The main focus in the following

is therefore on solving (7.12a). The proposed approach is based on an iterative procedure alternating between approximation of the distribution $P(\bar{\mathbf{x}}, \bar{\mathbf{u}}|\Theta^n, \bar{r} = 1)$ and utilisation of this distribution to obtain an improved Θ^{n+1}. We call this general method temporal PPI (PPI-T). Two alternative forms of the improvement step are proposed, one gradient and one EM based. The relative merits of these two methods are then discussed in Section 7.2.3

7.2.1 Gradient Descent

We first consider direct optimisation of (7.12a) by gradient descent. Let $\mathcal{L}(\Theta) = \log \int_{\bar{\mathbf{z}}} P(\Theta, \bar{\mathbf{z}}|\bar{r} = 1)$ and note that

$$\nabla\mathcal{L}(\Theta) = (\int_{\bar{\mathbf{z}}} P(\bar{\mathbf{z}}|\bar{r} = 1, \Theta))^{-1}\nabla \int_{\bar{\mathbf{z}}} P(\bar{\mathbf{z}}|\bar{r} = 1, \Theta) - \nabla\mathcal{C}_\delta(\Theta)$$

In the general case $\mathsf{P}(\bar{r} = 1, \bar{\mathbf{z}}|\Theta)$ will not be tractable. We therefore propose taking – similar to the standard PPI algorithms – a Gaussian approximation $\tilde{p}(\bar{\mathbf{z}}|\Theta) \approx P(\bar{r} = 1, \bar{\mathbf{z}}|\Theta)$. The latter admits closed form marginalisation of $\bar{\mathbf{z}}$ and allows us to compute the approximate gradient

$$\nabla_\Theta \int_{\bar{\mathbf{z}}} P(\bar{r} = 1, \bar{\mathbf{z}}|\Theta) \approx \nabla_\Theta \int_{\bar{\mathbf{z}}} \tilde{p}(\bar{\mathbf{z}}|\Theta) \ .$$

The exact form of the approximate gradient will depend on the specific form of the approximations taken to form $\tilde{p}$. We shall derive its general form, assuming a state,control LQ approximation has been obtained. That is to say, approximations of the form

$$\begin{aligned} f(\mathbf{z}_k) \approx& \mathbf{a}_k(\Theta) + \mathbf{A}_k(\Theta)\mathbf{z}_k \qquad \mathbf{Q}_k = \mathbf{Q}(\Theta) \\ \mathcal{C}(\mathbf{z}_k, k) \approx& \frac{1}{2}\mathbf{z}_k^T\mathbf{C}_k(\Theta)\mathbf{z}_k - \mathbf{c}_k(\Theta)^T\mathbf{z}_k \end{aligned}$$

have been taken, where all terms may depend non-linearly on Θ. In the interest of an uncluttered notation we will not further note this dependence explicitly. We now define

$$\tilde{p}(\bar{\mathbf{z}}|\Theta) = \underbrace{\mathcal{N}(\bar{\mathbf{z}}|\mu, \mathbf{\Sigma})}_{\text{dynamics prior}} \cdot \underbrace{\mathcal{N}[\bar{\mathbf{z}}|\mathbf{c}, \mathbf{C}]}_{\text{cost likelihood}} \ ,$$

where $\mathbf{c} = (\mathbf{c}_1, ..., \mathbf{c}_K)^T$ and $\mathbf{C} = \text{diag}(\mathbf{C}_1, ..., \mathbf{C}_K)$, while the elements of μ are given by

$$\mu_i = (\mathbf{A}_0 \cdots \mathbf{A}_{i-1})\, \mathbf{z}_0 + \sum_{k=1}^{i-1} (\mathbf{A}_{k+1} \cdots \mathbf{A}_{i-1})\, \mathbf{a}_k$$

and $\mathbf{\Sigma}$ is the symmetric matrix with

$$\mathbf{\Sigma}_{ij} = \mathbf{\Sigma}_{ji}^T = (\mathbf{A}_{j-1} \cdots \mathbf{A}_i) \sum_{k=0}^{i-1} (\mathbf{A}_{i-1} \cdots \mathbf{A}_k) \mathbf{Q}_k (\mathbf{A}_{i-1}^T \cdots \mathbf{A}_k)^T)$$

for $i \leq j$. Now let us define $\hat{\mathbf{z}}$ to be the subset of $\bar{\mathbf{z}}$ of states which have an associated cost, i.e., $\hat{\mathbf{z}} = \{[\mathbf{z}_k]_i; [\mathbf{c}_k]_i \neq 0, [\mathbf{C}_k]_{(i,\cdot)} \neq 0\}$ Marginalising first over states not in $\hat{\mathbf{z}}$ and using standard Gaussian identities, we have

$$\begin{aligned}
\int_{\bar{\mathbf{z}}} \tilde{p}(\bar{\mathbf{z}}|\Theta) &= \int_{\hat{\mathbf{z}}} \mathcal{N}(\hat{\mathbf{z}}|\hat{\mu}, \hat{\mathbf{\Sigma}}) \mathcal{N}[\hat{\mathbf{z}}|\hat{\mathbf{c}}, \hat{\mathbf{C}}] \\
&= \int_{\hat{\mathbf{z}}} \mathcal{N}[\hat{\mathbf{z}}|\hat{\mathbf{\Sigma}}^{-1}\hat{\mu} + \hat{\mathbf{c}}, \hat{\mathbf{\Sigma}}^{-1} + \hat{\mathbf{C}}] \mathcal{N}(\hat{\mu}|\hat{\mathbf{C}}^{-1}\hat{\mathbf{c}}, \hat{\mathbf{\Sigma}} + \hat{\mathbf{C}}^{-1}) \\
&= \mathcal{N}(\hat{\mu}|\hat{\mathbf{C}}^{-1}\hat{\mathbf{c}}, \hat{\mathbf{\Sigma}} + \hat{\mathbf{C}}^{-1})
\end{aligned}$$

where $\hat{\mu}$ and $\hat{\mathbf{\Sigma}}$ denote the appropriate sub-vector and -matrix of μ and Σ respectively. Hence, with $\mathbf{m} := \hat{\mathbf{C}}^{-1}\hat{\mathbf{c}}$ and $\mathbf{M} := \hat{\mathbf{\Sigma}} + \hat{\mathbf{C}}^{-1}$, the approximate derivatives take the general form

$$\frac{\partial}{\partial \theta_n} \int_{\bar{\mathbf{z}}} \tilde{p}(\bar{\mathbf{z}}|\Theta) = \mathcal{N}(\hat{\mu}|\mathbf{m}, \mathbf{M}) \left[\mathbf{g}^T \left(\frac{\partial \mathbf{m}}{\partial \theta_n} - \frac{\partial \hat{\mu}}{\partial \theta_n} \right) - \frac{1}{2} \text{Tr}\left(\mathbf{M}^{-1}\right) \frac{\partial \mathbf{M}}{\partial \theta_n} + \frac{1}{2} \mathbf{g}^T \frac{\partial \mathbf{M}}{\partial \theta_n} \mathbf{g} \right]$$

where $\mathbf{g} = (\hat{\mathbf{\Sigma}} + \hat{\mathbf{C}}^{-1})^{-1}(\hat{\mu} - \hat{\mathbf{C}}^{-1}\hat{\mathbf{c}})$.

Combining the results, the overall approximation to the derivatives is obtained as

$$\frac{\partial}{\partial \theta_n} \mathcal{L}(\Theta) \approx -\frac{\partial}{\partial \theta_n} \mathcal{C}_\delta(\Theta^n) + \left[\mathbf{g}^T \left(\frac{\partial \mathbf{m}}{\partial \theta_n} - \frac{\partial \hat{\mu}}{\partial \theta_n} \right) - \frac{1}{2} \text{Tr}\left(\mathbf{M}^{-1}\right) \frac{\partial \mathbf{M}}{\partial \theta_n} + \frac{1}{2} \mathbf{g}^T \frac{\partial \mathbf{M}}{\partial \theta_n} \mathbf{g} \right] . \quad (7.13)$$

This can be used in any gradient based scheme to obtain a new Θ^{n+1}, which in turn gives rise to a new approximation.

Remark 7.5: In certain cases it may be possible to perform parts of the marginalisation of $\bar{\mathbf{z}}$ in closed form. In particular if the problem is control LQ, all controls can be integrated out in closed form. This does not affect the derivation and the resulting gradient takes the same form, with suitable substitutions.

Example 7.3 (Dynamical reaching task[3])**:** To make the above concrete, let us consider the special case of optimisation of movement duration. Specifically, control of a simple 2nd order dynamical system under a cost including a quadratic control cost and only a single terminal goal. Let $\mathbf{x} = (q, \dot{q})$ be the state with discretised dynamics

$$\mathbf{x}_{k+1} = \underbrace{\begin{bmatrix} \mathbf{I} & \theta\mathbf{I} \\ 0 & \mathbf{I} \end{bmatrix}}_{:=\mathbf{A}} \mathbf{x}_k + \underbrace{\begin{bmatrix} \theta^2\mathbf{I} \\ \mathbf{I} \end{bmatrix}}_{:=\mathbf{B}} \mathbf{u}_k + \epsilon \qquad \epsilon \sim \mathcal{N}\Big(0, \underbrace{\begin{bmatrix} \theta\mathbf{Q}^q & 0 \\ 0 & \theta\mathbf{Q}^{\dot{q}} \end{bmatrix}}_{:=\mathbf{Q}}\Big)$$

and trajectory cost

$$\mathcal{C}(\bar{\mathbf{x}}, \bar{\mathbf{u}}) = \mathcal{C}_T(\mathbf{x}_K) + \sum_{k=0}^{K} \mathbf{u}_k \mathbf{C}^u \mathbf{u}_k \theta$$

Note that as the system is control LQ, we may marginalise $\bar{\mathbf{u}}$ analytically. This yields the associated marginal dynamics

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \epsilon' \qquad \epsilon \sim \mathcal{N}(0, \underbrace{\mathbf{Q} + \mathbf{B}(\theta\mathbf{C}^u)^{-1}\mathbf{B}^T}_{:=\mathbf{W}})$$

Marginalising $\bar{\mathbf{u}}$ has the beneficiary effect of removing all intermediate costs. Hence the computation reduces to computing the gradients of μ_K and Σ_{KK}. Furthermore, the matrix $\mathbf{A}$ is time invariant and due to its form we have in general

$$(\mathbf{A}_i \cdots \mathbf{A}_j) = \begin{bmatrix} \mathbf{I} & |i-j|\theta\mathbf{I} \\ 0 & \mathbf{I} \end{bmatrix} .$$

Consequently we have

$$\frac{\partial \mu_K}{\partial \theta} = \mathbf{A}^{K-1}\mathbf{x}_0 = \begin{bmatrix} (K-1)\dot{q}_0 \\ 0 \end{bmatrix} .$$

[3]n.b., the gradients for this specific case were derived by D. Zarubin (personal communications)

In order to compute the gradient of $\mathbf{\Sigma}_{KK}$ observe that

$$\begin{aligned}\mathbf{\Sigma}_{KK} &= \sum_{k=0}^{K} \mathbf{A}^k \mathbf{W} (\mathbf{A}^k)^T \\ &= \sum_{k=0}^{K} \begin{bmatrix} \mathbf{I} & k\theta\mathbf{I} \\ 0 & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{W}_{11} & \mathbf{W}_{12} \\ \mathbf{W}_{21} & \mathbf{W}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{I} & 0 \\ k\theta\mathbf{I} & \mathbf{I} \end{bmatrix} \\ &= \begin{bmatrix} s_0\mathbf{W}_{11} + 2\theta s_1\mathbf{W}_{12} + \theta^2 s_2\mathbf{W}_{22} & s_0\mathbf{W}_{12} + \theta s_1\mathbf{W}_{22} \\ s_0\mathbf{W}_{12} + \theta s_1\mathbf{W}_{22} & s_0\mathbf{W}_{22} \end{bmatrix},\end{aligned}$$

where $s_0 = \sum_0^{K-1} 1$, $s_1 = \sum_{k=1}^{K-1} k$ and $s_2 = \sum_{k=1}^{K-1} k^2$. Substituting the terms for $\mathbf{W}$ we hence obtain,

$$\frac{\partial}{\partial\theta}\mathbf{\Sigma}_{KK} = \begin{bmatrix} \theta^2(\mathbf{C}^u)^{-1}s_{(0,1,2)} + s_0\mathbf{Q}^q + \theta^2 s_2\mathbf{Q}^{\dot{q}} & \theta(\mathbf{C}^u)^{-1}s_{(0,1)} + \theta s_1\mathbf{Q}^{\dot{q}} \\ \theta(\mathbf{C}^u)^{-1}s_{(0,1)} + \theta s_1\mathbf{Q}^{\dot{q}} & s_0((\mathbf{C}^u)^{-1} + \mathbf{Q}^{\dot{q}}) \end{bmatrix},$$

with $s_{(0,1,2)} = s_0 + 2s_1 + s_2$ and $s_{(0,1)} = s_0 + s_1$.

Substituting $\frac{\partial\mu_K}{\partial\theta}$ and $\frac{\partial}{\partial\theta}\mathbf{\Sigma}_{KK}$ into (7.13) now gives the required gradient.

7.2.2 Expectation Maximisation

The solution to (7.12a) can alternatively we obtained using an Expectation Maximisation approach. Specifically, we form the bound

$$\log \int_{\bar{\mathbf{z}}} P(\bar{\mathbf{z}}, \bar{r} = 1|\Theta) > \int_{\bar{\mathbf{z}}} \underbrace{P(\bar{\mathbf{z}}|\bar{r} = 1, \Theta)}_{p(\bar{\mathbf{z}})} \log P(\bar{r} = 1, \bar{\mathbf{z}}|\Theta) \ , \tag{7.14}$$

which is alternately maximised with respect to p and Θ, in an E- and M-step.

E-Step

In the E-Step we aim to calculate the posterior over the unobserved variables, i.e. the trajectories, given the current parameter values Θ^i,

$$p^n(\bar{\mathbf{z}}) = P(\bar{\mathbf{z}}|\bar{r} = 1, \Theta^n) \ .$$

However, as this is not tractable, we form an approximation $\tilde{p}^n$ using PPI.

M-Step

In the M-Step, we solve

$$\Theta^{i+1} = \underset{\Theta}{\operatorname{argmin}}\, \mathbb{E}_{\tilde{p}^n}\left[\log P(\bar{r}=1, \bar{\mathbf{z}}|\Theta)\right] \ ,$$

where $\tilde{p}^n$ is the approximation calculated in the E-Step based on Θ^n. We may expand the objective as

$$\mathbb{E}_{\tilde{p}^n}\left[\log P(\bar{r}=1, \bar{\mathbf{z}}|\Theta)\right] = \sum_{k=0}^{K-1} \left(\mathbb{E}\left[\log P(\mathbf{z}_{k+t}|\mathbf{z}_k, \delta_k)\right] - \mathbb{E}\left[\mathcal{C}(\mathbf{z}_k, \delta_k)\right]\right) + \mathit{constant} \ ,$$

where $\mathbb{E}[\cdot]$ denotes the expectation with respect to $\tilde{q}^n$. The required expectations, $\mathbb{E}\left[\mathcal{C}(\mathbf{z}_k, \delta_k)\right]$ and

$$\mathbb{E}\left[\log P(\mathbf{z}_{k+t}|\mathbf{z}_k, \delta_k)\right] = -\frac{1}{2}\mathbb{E}\left[(\mathbf{z}_{k+t} - f(\mathbf{z}_k))^\top \mathbf{Q}_k^{-1}(\mathbf{z}_{k+1} - f(\mathbf{z}_k))\right] - \frac{D_{\mathbf{z}}}{2}\log|\mathbf{Q}_k| \ ,$$

are in general not tractable. As previously, we therefore resort again to a LQ approximation. This leads, in the general case, to an expression which can not be maximised analytically w.r.t. Θ. However, if the approximation and discretisation are chosen such that the system is also linear in δ, i.e.,

$$\begin{aligned} f(\mathbf{z}_k) \approx& (\mathbf{a}_k + \mathbf{A}_k\mathbf{z}_k)\delta_k \qquad \mathbf{Q}_k = \mathbf{Q}\delta_k \\ \mathcal{C}(\mathbf{z}_k, k) \approx& (\frac{1}{2}\mathbf{z}_k^T\mathbf{C}_k\mathbf{z}_k - \mathbf{c}_k^T\mathbf{z}_k)\delta_k \end{aligned}$$

it can be shown that,

$$\frac{\partial}{\partial\delta_k}\mathbb{E}_{\tilde{p}^n}\left[\log P(\bar{r}=1, \bar{\mathbf{z}}|\Theta)\right] = \delta_k^{-2}g_2 + \delta_k^{-1}g_1 + g_0 \ , \tag{7.15}$$

with

$$\begin{aligned} g_2 =& \frac{1}{2}\delta_k^{-2}\operatorname{Tr}\left(\mathbf{Q}_k^{-1}(\mathbb{E}\left[\mathbf{z}_{k+1}\mathbf{z}_{k+1}^T\right] - 2\mathbb{E}\left[\mathbf{z_{k+1}}\mathbf{z}_k^T\right] + \mathbb{E}\left[\mathbf{z}_k\mathbf{z}_k^T\right])\right) \\ g_1 =& -\frac{D_z^2}{2} \\ g_0 =& -\frac{1}{2}\Bigg[\operatorname{Tr}(\mathbf{A}_k\mathbf{Q}_k^{-1}\mathbf{A}_k^T\mathbb{E}\left[\mathbf{z}_k\mathbf{z}_k^T\right]) + \mathbf{a}_k^T\mathbf{Q}_k^{-1}\mathbf{a}_k + 2\mathbf{a}_k^T\mathbf{Q}_k^{-1}\mathbf{A}_k\mathbb{E}\left[\mathbf{x}_k\right] \\ & + 2\left.\frac{\mathrm{d}}{\mathrm{d}\delta}\mathcal{C}_\delta\right|_{\delta_k} + \operatorname{Tr}(\mathbf{C}_k\mathbb{E}\left[\mathbf{z}_k\mathbf{z}_k^T\right]) - 2\mathbf{c}_k^T\mathbb{E}\left[\mathbf{z}_k\right]\Bigg] \ . \end{aligned}$$

In the general case we may now use gradient ascent to improve the θ's. However, in the specific case where $\mathcal{C}_\delta$ is a linear function of δ and the parametrisation discussed in Section 7.1.2 is used, (7.15) is quadratic in θ_k^{-1} and the unique extremum under the constraint $\theta_k > 0$ can be found analytically.

7.2.3 Comparison of Gradient and EM Updates

The two proposed methods have different merits. From the point of view of computational complexity the EM based updates are certainly preferable. They only require computation of the pair marginals $(\mathbf{z}_k, \mathbf{z}_{k+1})$ and operate entierly on matrices which are the size of $\mathbf{z}_k$'s dimension. Contrast this with the gradient method, which requires computation of the covariance of all cost conditioned states and controls. Due to the inversion of this matrix, gradient updates are usually more expensive to compute. The exception are problems which are control LQ with no running costs in the states. In such cases analytical marginalisation of controls means that $\hat{\mathbf{z}}$ only contain the few goal states.

While computationally attractive, EM updates suffer from numerical instability in many problems. In general, the deficiency of EM algorithms in near deterministic regimes is a well known problem (e.g., Barber and Furmston, 2009). In our case it leads to instability, when $\mathbf{Q} \approx 0$. The problem arises in the M-Step, which may be written as

$$\underset{\Theta}{\operatorname{argmax}} - \operatorname{KL}\left(p(\bar{\mathbf{z}}|\Theta^n) \| P(\bar{\mathbf{z}}|\bar{r}=1,\Theta)\right) + \log \int_{\bar{\mathbf{z}}} P(\bar{r}=1,\bar{\mathbf{z}}|\Theta)$$

It is now apparent that for deterministic dynamics no change in Θ is possible, lest the KL divergence becomes infinite. Unfortunatly, in cases when $\mathbf{Q}$ is near zero the updates do not get stuck, rather they diverge, due to numerical errors.

7.2.4 Practical Considerations

The performance of the algorithm can be greatly enhanced by using the result of the previous E-Step as initialisation for the next one. As this is likely to be near the optimum with the new temporal trajectory, approximate inference converges within only a few iterations.

In the gradient case substantial gains in computation time can be made, by incrementally increasing the subset of costs considered. As discussed above, the gradient computation scales badly with the number of cost conditioned states and controls. We may therefore begin by only considering the cost on the final state. Once converged, additional cost are added, till eventually the complete gradient is computed. As the solution with a subset of the costs is again likely to be near the overall solution, fewer computationally expensive iterations will be necessary.

The proposed algorithms lead to a variation of discretization step length which can be a problem. For one, the approximation error increases with the step length which may lead to unstable results. On the other hand, the algorithm may lead to control frequencies which are not achievable in practice. In general, a fixed control signal frequency may be prescribed by the hardware system. In practice θ's can be kept in a prescribed range by adjusting the number of discretization steps K after an new Θ has been computed.

Finally, although we have chosen to express the time cost in terms of a function of the θ's, often it may be desirable to consider a cost directly over the duration T. Noting that $T = \sum \theta_k$, all that is required is to replace $\frac{d}{d\theta}\mathcal{C}_\delta$ with $\frac{\partial}{\partial \theta_k}\mathcal{C}_\delta(\sum \theta)$ in the relevant equations.

7.3 Experiments

7.3.1 EM Based Updates

We first evaluate the proposed algorithm with EM updates in simulation on a simple plant. As a basic plant, we used a simulation of a 2 degrees of freedom planar arm, consisting of two links of equal length. The state of the plant is given by $\mathbf{x} = (\mathbf{q}, \dot{\mathbf{q}})$, with $\mathbf{q} \in \mathbb{R}^2$ the joint angles and $\dot{\mathbf{q}} \in \mathbb{R}^2$ associated angular velocities. The controls $\mathbf{u} \in \mathbb{R}^2$ are the joint space accelerations. We also added some noise with diagonal covariance.

For all experiments, we used a trajectory cost of the form

$$\mathcal{C}(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \Theta) = \mathcal{C}(\bar{\mathbf{x}}) + \sum_{k=0}^{K} \mathbf{u}_k^T (\delta_k(\Theta)\mathbf{C}^u)\mathbf{u}_k + \alpha\delta_k(\Theta) \tag{7.16}$$

where $\mathbf{C}^u = 10^4 \cdot \mathbf{I}$. The state dependent cost was

$$\mathcal{C}(\bar{\mathbf{x}}) = \sum_n (\phi_n(\mathbf{x}_{\hat{k}_n}) - \mathbf{y}_n^*)^T \Lambda_n (\phi_n(\mathbf{x}_{\hat{k}_n}) - \mathbf{y}_n^*) \ , \tag{7.17}$$

where the tuplets $(\hat{k}_n, \phi_n, \Lambda_n, \mathbf{y}_n^*)$ define goals and consist of a time step, a task space mapping, a diagonal weight matrix and the desired state in task space. For example, for point targets, the task space mapping is $\phi(\mathbf{x}) = (x, y, \dot{x}, \dot{y})^T$, i.e., the map from $\mathbf{x}$ to the vector of end point positions and velocities in task space coordinates, and $\mathbf{y}^*$ is the target coordinate.

Variable Distance Reaching Task

In order to evaluate the behaviour of PPI-T we applied it to a reaching task with varying start-target distance. Specifically, for a fixed start point we considered a series of targets lying equally spaced along a line in task space. It should be noted that, although the targets are equally spaced in task space and results are shown with respect to movement distance in task space, the distances in joint space scale non linearly. The state cost (7.17) contained a single term incurred at the final discrete step with $\Lambda = 10^6 \cdot \mathbf{I}$. Figure 7.3(c)&(d) show the movement duration ($= K \cdot \theta$) and standard reaching cost[4] for different temporal-cost parameters α (we used $\alpha_0 = 2 \cdot 10^7$), demonstrating that PPI-T successfully trades-off the movement duration and standard reaching cost for varying movement distances. In Figure 7.3(b), we compare the reaching costs of PPI-T with those obtained with a fixed duration approach, in this case PPI. Note that although with a fixed, long duration (e.g., PPI with duration T=0.41) the control and error costs are reduced for short movements, these movements necessarily have up to 4× longer durations than those obtained with PPI-T. For example for a movement distance of 0.2 application of PPI-T results in a optimised movement duration of 0.07 (cf. Figure 7.3(c)), making the fixed time approach impractical when temporal costs are considered. Choosing a short duration on the other hand (PPI (T=0.07)) leads to significantly worse costs for long movements. We further emphasise that the fixed durations used in this comparison were chosen post hoc by exploiting the

[4]n.b. the *standard reaching cost* is the sum of control costs and cost on the endpoint error, without taking duration into account, i.e., (7.16) without the $\mathcal{T}(\theta)$ term.

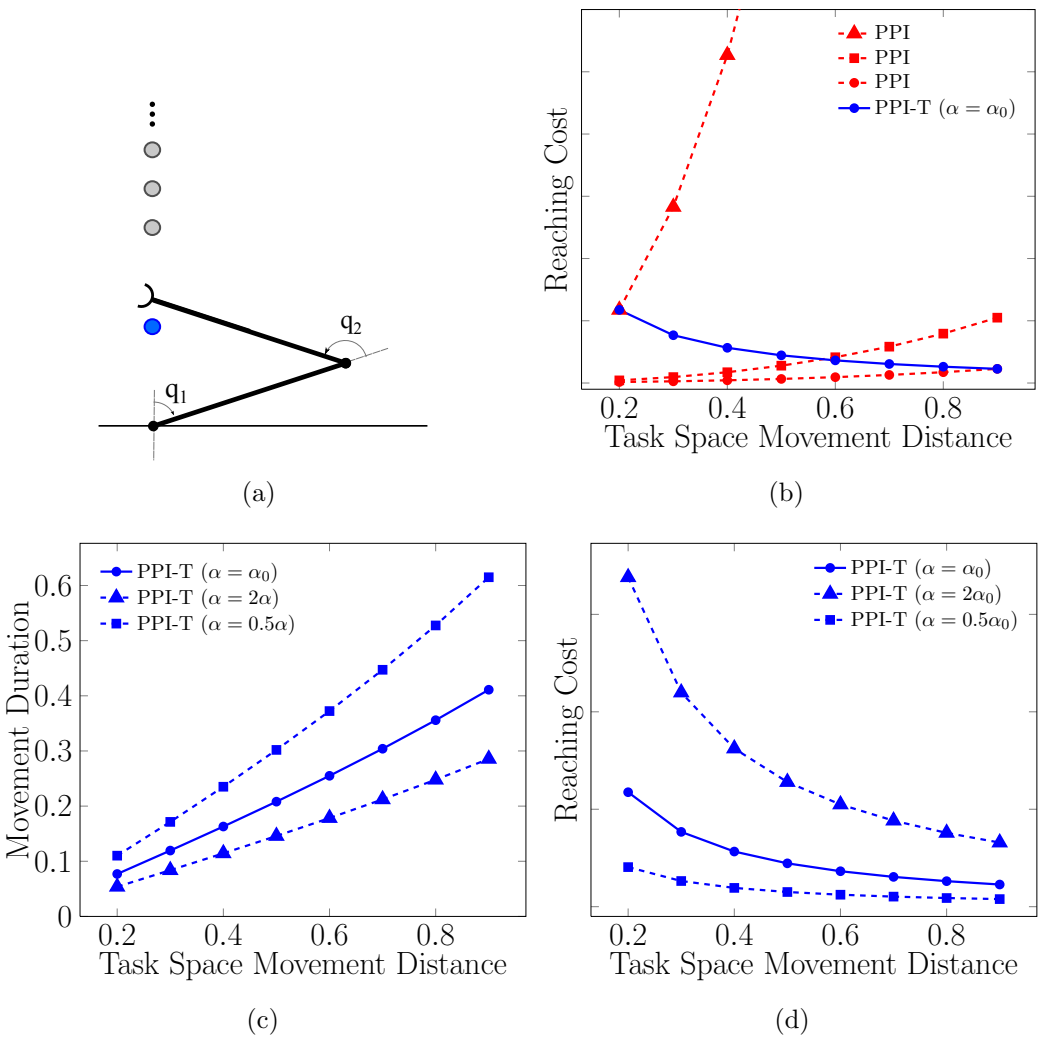


Figure 7.3: Temporal scaling behaviour using PPI-T. **(a)** Schematic of plant together with mean start position and list of targets **(b)** Comparison of reaching costs (control + error cost) for PPI-T and a fixed duration approach, i.e. PPI. **(c)&(d)** Effect of changing time-cost weight α, (effectively the ratio between reaching cost and duration cost) on duration and reaching cost (control + state cost).

durations suggested by PPI-T. In it's absence there would have been no practical way of choosing them apart from experimentation. Furthermore, we would like to highlight that, although the results suggests a simple scaling of duration with movement distance, in cluttered environments and plants with more complex forward kinematics, an efficient decision on the movement duration cannot be

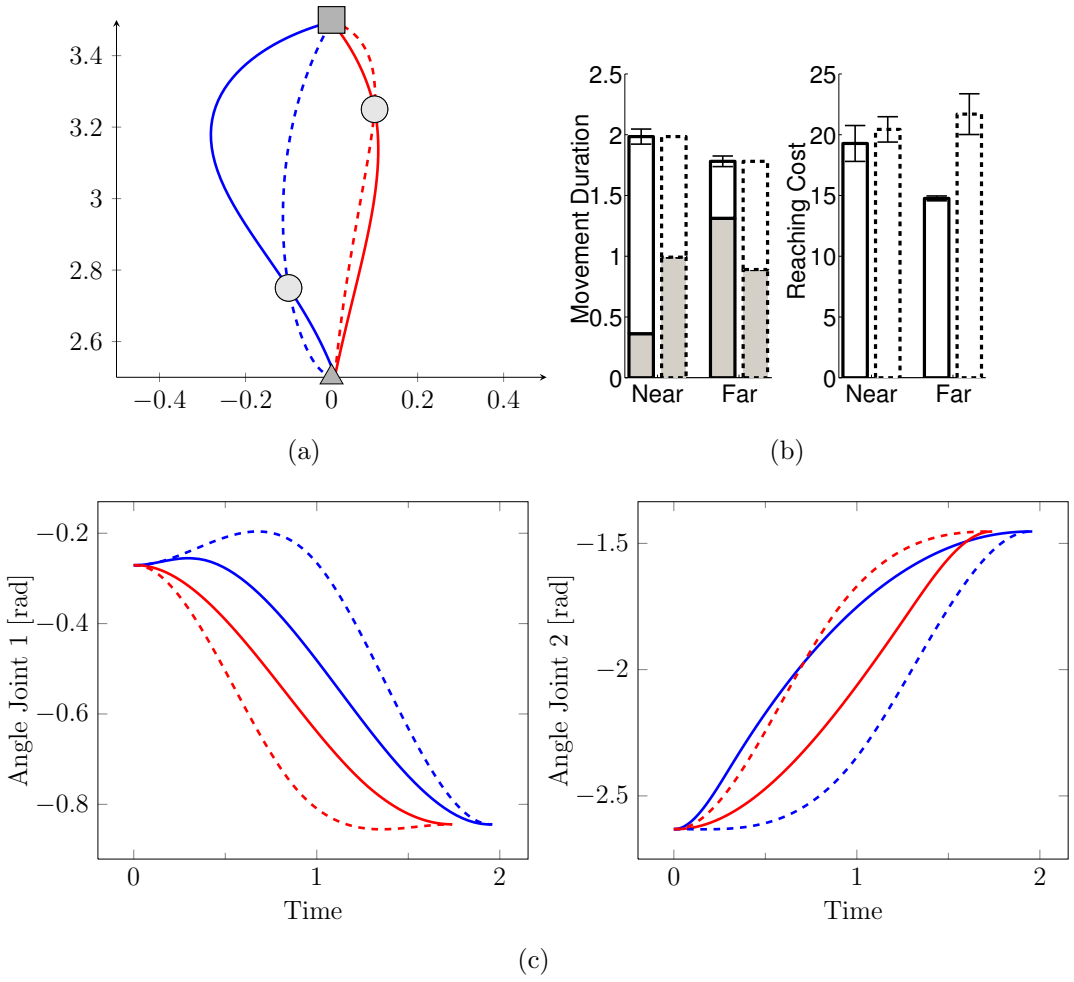


Figure 7.4: Comparision of PPI-T (—) to PPI with the common modelling approach (- -) with fixed times on a via point task. **(a)** End point task space trajectories for two different via points ○ obtained for a fixed start point △. **(c)** The corresponding joint space trajectories. **(b)** Movement durations and reaching costs (control + error costs) from 10 random start points. The proportion of the movement duration spend before the via point is shown in light gray (mean in the PPI-T case).

based only on task space distance.

Via Point Reaching Tasks

We also evaluated the proposed algorithm in a more complex via point task. The task requires the end-effector to reach to a target, having passed at some point through a given second target, the via point. This task is of interest as it can be

seen as an abstraction of a diverse range of complex sequential tasks that require one to achieve a series of sub-tasks in order to reach a final goal. This task has also seen some interest in the literature on modelling of human movement using the optimal control framework (e.g., Todorov and Jordan, 2002). Here the common approach is to choose the time point at which one passes the via point such as to divide the movement duration in the same ratio as the distances between the start point, via point and end target. This requires on the one hand prior knowledge of these movement distances and on the other, makes the implicit assumption that the two movements are in some sense independent.

Here, we demonstrate the ability of our approach to solve such sequential problems, adjusting movement durations between sub goals in a principled manner, and show that it improves upon the standard modelling approach. Specifically, we apply PPI-T to the two via point problems illustrated in Figure 7.4(a) with randomised start states[5]. For comparison, we follow the standard modelling approach and apply PPI to compute the controller. We again choose the movement duration for the standard case post hoc to coincide with the mean movement duration obtained with PPI-T for each of the individual via point tasks. Each task is expressed using a cost function consisting of two point target cost terms. Specifically, (7.17) takes the form

$$\mathcal{C}(\bar{\mathbf{x}}) = (\phi(\mathbf{x}_{\frac{K}{2}}) - \mathbf{y}_v^*)^T \Lambda_v (\phi(\mathbf{x}_{\frac{K}{2}}) - \mathbf{y}_v^*) + (\phi(\mathbf{x}_K) - \mathbf{y}_e^*)^T \Lambda_e (\phi(\mathbf{x}_K) - \mathbf{y}_e^*) \ ,$$

with diagonal matrices

$$\begin{aligned} \Lambda_v &= \mathrm{diag}(\lambda_{pos}, \lambda_{pos}, 0, 0) \\ \Lambda_e &= \mathrm{diag}(\lambda_{pos}, \lambda_{pos}, \lambda_{vel}, \lambda_{vel}) \ , \end{aligned}$$

where $\lambda_{pos} = 10^5$ & $\lambda_{vel} = 10^7$ and vectors $\mathbf{y}_v^* = (\cdot, \cdot, 0, 0)^T$, $\mathbf{y}_e^* = (\cdot, \cdot, 0, 0)^T$ for via point and target respectively. Note that the cost function does not penalise velocity at the via point but encourages the stopping at the target. While admittedly the choice of incurring the via point cost at the middle of the movement ($\frac{K}{2}$) is likely to be a sub-optimal choice for the standard approach, one has to consider that in more complex task spaces, the relative ratio of movement distances may

[5]For the sake of clarity, Figure 7.4(a)&(c) show mean trajectories of controllers computed for the mean start state.

Method	Simple Obstacles	Complex Obstacle
PPI	1	1
PPI-T (end cost)	0.585 (± 0.337)	0.635 (± 0.085)
PPI-T (full)	0.549 (± 0.311)	0.123 (± 0.047)

Table 7.1: Results for application of PPI-T to the robotic manipulation with obstacles as described in Section 5.4.2. Shown are the mean ratio of expected cost relative to PPI and it's standard deviation.

not be easily accessible and one may have to resort to the most intuitive choice for the uninformed case as we have done here. Note that, although for PPI-T this cost is incurred at the same discrete step, we allow θ before and after the via point to differ, but constrain them to be constant throughout each part of the movement, hence, allowing the cost to be incurred at an arbitrary point in real time. We sampled the initial position of each joint independently from a Gaussian distribution with a variance of 3°. In Figure 7.4(a)&(c) we show mean trajectories in task space and joint space for controllers computed for the mean initial state. Interestingly, although the end point trajectory for the *near* via point produced by PPI-T may look sub-optimal compared to that produced by the standard PPI algorithm, closer examination of the joint space trajectories reveals that our approach results in more efficient actuation trajectories. In Figure 7.4(b), we illustrate the resulting average movement durations and costs of the mean trajectories. As can be seen, PPI-T results in the expected passing times for the two via points, i.e., early vs. late in the movement for the near and far via point, respectively. This directly leads to a lower incurred cost compared to un-optimised movement durations.

7.3.2 Gradient Based Method

We now turn to the gradient based updates introduced in Section 7.2.1 and their application to planning with the 7-DOF Kuka lightweight robot. Specifically, we use the same tasks as used in Chapter 5 for evaluation of PPI. Our aim is two fold, on the one hand to demonstrate scalability to practical applications and,

on the other hand, to demonstrate that temporal optimisation can significantly improve the results compared to naive selection of the movement durations.

With the task setup for simple and complex obstacles as discussed in Section 5.4.2, we compare three methods.

- **PPI-T(full)** is the complete algorithm as described in Section 7.2.1.

- **PPI-T(end cost)** is the algorithm as described in Section 7.2.1. However, the gradient is calculated taking only the reaching cost into account, i.e., ignoring joint limit and obstacle costs. The intention is to illustrate that selection of duration needs to take into account the entire problem and can not be simply based on a target-distance law as could be derived from, e.g., Figure 7.3.

- **PPI** is the algorithm with fixed duration. This is to provide a comparison to the naive approach prevalent in the literature. Note however that we set the duration to the mean duration obtained by *PPI-T(end cost)*. Hence it was in some sense adapted to the task distribution. Without PPI-T selection would have, at best, relied on manual selection based on an individual task instance or, at worst, a random guess. Both approaches lead to substantially worse results.

As in the evaluation of PPI, we sample 50 task instances and compare the relative expected costs. However, unlike with the variation of η, we did not experience the same problems with outliers, all methods converging to conceptually equivalent local minima. We therefore report both the mean and standard deviation in Table 7.1, rather then the median. As can be seen, temporal optimisation improves upon the naive application of PPI. In particular, note that instance specific durations as given by PPI-T(end cost) improve significantly on selecting an informed constant duration (the mean duration over task instances). Furthermore, taking the entire problem into account leads to increasing gains as the problem complexity increases.

7.4 Discussion

The contribution of this chapter is a novel method for jointly optimizing a trajectory and its time evolution (temporal scale and duration) in the stochastic optimal control framework. In particular, two extension of the PPI method of Chapter 5 with complementary strength and weaknesses are presented. The gradient based approach, on the one hand, is widely applicable but can become computationally demanding. Meanwhile, the EM method provides an algorithm with lower computational cost, is however only applicable for certain classes of problems.

The experiments have concentrated on demonstrating the benefit of temporal optimisation in manipulation tasks. However, arguably it is dynamic movements which can benefit most from temporal adjustment. An example of this was seen in the brachiation task of Example 7.2. Subsequent work by Nakanishi and Vijayakumar (2012) extended the application of our framework to brachiation with variable stiffness actuation, showing that an coordinated interplay of stiffness and temporal adjustment gives rise to gains in performance. We anticipate that, with the general rise of interest in variable impedance, e.g., in throwing (Braun et al., 2012) or locomotion (Enoch et al., 2012), temporal optimisation will become a necessity if the capabilities of the dynamical system are to be fully exploited. Our framework provides a principled step in this direction.

Chapter 8

Conclusion

In this thesis, we have studied the relations that exist between Stochastic Optimal Control and Probabilistic Inference, concentrating in particular on control problems from the robotics domain. Our aim was to arrive at an understanding of the relation between these two problems, sufficient to allow for transfer of ideas from approximate inference to the SOC setting. Our efforts in this direction led us to novel insights into the connection between the two problems. Utilising these we proposed a series of novel algorithms, which were shown to have distinct benefits over current approaches.

In Chapter 3, we proposed a specific dual interpretation of general discrete time SOC problems in terms of minimisation of a KL divergence. Specifically, we observe that SOC can be understood as the problem of attempting to match an uncontrolled process conditioned on task fulfilment with a controlled process. This was formalised in the general duality of Corollary 3.1. Contrasting our result with previous approaches in this area we found that, under additional assumptions, the latter often arise as special cases within our formulation. While our result is more general, it does however not directly yield analytically tractable solutions. Despite this, we were able to demonstrate in the remainder of this thesis that it does provide a basis for practical iterative approaches to SOC.

In Chapter 4, we demonstrated that relaxation of Corollary 3.1 gives rise to a simple condition for iterative policy improvement (Proposition 4.1). Based on this, we derived novel iterations in policy space for both, finite and discounted infinite horizon problems. Importantly, we were able to show these to converge

to a globally optimal policy. Following on from these analytical results, we formulated the practical sample based Ψ-Learning algorithms – novel model free RL algorithms for both discrete and continuous state and control spaces. While the iterations obtained in this chapter posses pleasing analytical properties and provide a unifying view on several previously proposed methods, they did not bring us closer to our fundamental aim, the application of inference methods.

In Chapter 5, we addressed this shortcoming by proposing an alternative relaxation of our duality, leading to the PPI procedure. We highlighted the relation of PPI to, on the one hand, risk sensitive SOC and, on the other hand, to certain previously proposed formulations of inference based control which had lacked a formal understanding in the context of SOC. Based on these insights, we identified these methods as risk seeking control procedures. We then proceeded to adapt the inference algorithm of one of them (Approximate Inference Control) so as to reduce it's risk seeking tendencies and demonstrated the benefit of such an adjustment on a series of manipulation tasks with a modern hand-arm system.

In Chapter 6, we presented a novel algorithmic approach to SOC, exploiting the form the problem assumes under specific inference based formulations (including PPI). Specifically, we expressed the problem in terms of linear operators in a Reproducing Kernel Hilbert Space. Importantly, these operators can be consistently estimated from a set of transition samples. Building on the basic algorithm, we presented a series of improved estimators which utilise the structure of the SOC problem to allow the sample data to be used more efficiently. The advantage of these approaches over naive Monte Carlo based inference was demonstrated in experiments which highlighted the ability of our estimators to transfer information across tasks.

Finally, in Chapter 7, we addressed the lack of practical approaches which allow for optimisation of temporal task parameters like, e.g., movement duration. We proposed both a general formulation of the problem in terms of a principled extension of the standard finite horizon formulation and a practical algorithmic approach based on PPI. Experiments highlighted the benefits of such temporal optimisation.

Outlook & Future Work

We have already discussed several specific potential extensions to our work in the discussions of individual chapters. However, there also exist various more general directions in which the work presented in this thesis may be extended in the future.

A General Perspective on Policy Search Methods

We have already shown Ψ-Learning based algorithms, presented in Chapter 4, to be closely related to a number of alternative RL methods. In general we may observe that, Ψ-Iterations are based on the projection of $p_{\pi^n}(\bar{\mathbf{x}}, \bar{\mathbf{u}})$ onto $\{q_\pi; \pi \in \mathcal{F}_\pi\}$, the family of possible trajectory distributions under a class $\mathcal{F}_\pi$ of policies. This projection is accomplished by means of minimisation of the KL divergence $\mathrm{KL}\left(\cdot \| p_{\pi^n}\right)$. Interestingly, choosing the alternative projection given by minimisation of $\mathrm{KL}\left(p_{\pi^n} \| \cdot\right)$ gives rise to the EM based algorithms discussed in Section 3.2.1. Consequently, the analysis of approximate updates in Section 4.1.2 provides a framework in which the behaviour of these algorithms can be jointly studied, potentially providing principled insights into their relative merits.

Furthermore, we may take the unification of message passing based inference algorithms based on α-Divergences by Minka (2005) as an inspiration. It suggests an analogous family of RL algorithms indexed by the divergence used for projection and containing the above two KL divergences as special cases. The analytical tools developed for the analysis of the asymptotic behaviour of Ψ-Iterations generalise across this entire family and may highlight benefits of intermediate algorithms which sit between Ψ-Learning and the EM methods.

Improved Approximations

The implementation of PPI in Chapter 5 was based on Gaussian approximations of the time slice marginals. As already discussed in Section 5.3.2, computation of these approximations could be improved by using statistical linearisation techniques from the filtering and smoothing literature (e.g., Hartikainen et al., 2011). However, a more interesting extension would be the use of structured models. It has been observed that the movements of biological systems exhibit significant

structure and regularities even across varying tasks. This has given rise to the hypothesis that control is performed on a manifold spanned by so called synergies, that is, a coupling amongst sets of actuators (e.g., d'Avella et al., 2003). Chhabra and Jacobs (2006) furthermore demonstrate that such structure also naturally arises by application of SOC. These observations suggest the use of a structured approximating distribution based on such synergies, rather then the fully factorised Gaussian. This could significantly simplify the inference problem, as such a synergy based distribution would have fewer parameters. Also, by providing a better model of optimal movements, such structured approximations could significantly improve the quality of the eventual result. Eventually, such an approach could incorporate Bayesian model selection (MacKay, 2003, Chap. 28) as a principled tool for picking, or even learning, an appropriate model.

Hierarchical Control Algorithms

We already alluded to two possible extensions of the RKHS based approach of Chapter 6 during it's discussion (cf. Section 6.6). Specifically, on the one hand, we observed that the algorithm requires solutions of local point-to-point SOC problems and that these may be solved using alternative approaches like PPI. On the other hand, we commented on the possibility of using more elaborate kernels to incorporate state space abstractions directly into the computations. The combination of these ideas directly suggests a hierarchical control algorithm which exploits the strength of both, local methods and the global properties of the RKHS approach. Concretely, we suggest to apply the RKHS based methodology on a coarse temporal scale, with a kernel which implies a high level abstraction on the state. For example, we may use kernels on a relational abstraction of the problem (Gärtner et al., 2003), thus effectively directly linking inference based relational planning (Lang, 2011) with actuator level control of the plant. The arising local problems can be solved using either PPI or any other local approximation method. Such an architecture would exploit the capabilities of the RKHS approach to provide a general global roadmap, guiding the local method, which in turn can provide the fine detail.

Dynamic Temporal Optimisation

In Chapter 7, we performed temporal optimisation in static tasks. That is to say, the proposed approach computes fixed temporal parameters based on a known deterministic configuration of the environment. However, when interacting with an dynamic uncertain environment such an approach is unsuitable. For example, when obstacles in the environment move in an unpredictable pattern, e.g., when interacting with other agents, the assumptions made during optimisation may quickly become invalid. As such, an approach which extends to these scenarios is desirable. In principle the problem can be addressed by modelling the stochastic dynamics of the entire system, i.e., the plant and environment, and computing feedback policies for both ordinary and temporal controls. In practise such an approach is however not feasible, due to the complexity of the resulting problem. An alternative would be to selectively re-plan, if during the execution of the movement observations indicate sufficient divergence from the conditions assumed during initial planning.

Appendices

Appendix A

Kullback-Leibler Divergence

The Kullback-Leibler divergence – also known as the Relative Entropy, Information divergence or Information gain – is a measure of difference between two probability distributions. For discrete probability distributions p & q it is defined by

$$\mathrm{KL}\left(p\|q\right) = \sum_{x} p(x) \log \frac{p(x)}{q(x)} \ ,$$

where by a continuity argument we take $0 = 0 \log 0$. Note that the KL divergence requires, p be absolutely continuous with respect to q, that is for any x, $q(x) = 0 \Rightarrow p(x) = 0$. In the continuous case the KL divergence takes the corresponding form,

$$\mathrm{KL}\left(p\|q\right) = \int p(x) \log \frac{p(x)}{q(x)} \mathrm{d}x \ .$$

Although the KL divergence is often interpreted as a distance measure between probability distributions it does not satisfy the conditions to be a true metric. In this thesis we make use of the following properties of the KL divergence, proofs of which are given by Cover and Thomas (1991),

- $\mathrm{KL}\left(p\|q\right) \geq 0$ for all p, q
- $\mathrm{KL}\left(p\|q\right) = 0$, if and only if $p = q$
- $\mathrm{KL}\left(p\|q\right) \neq \mathrm{KL}\left(q\|p\right)$, if $p \neq q$

Appendix B

Supplamentary Results to Chapter 3

We briefly outline the necessary steps in obtaining (3.11). Recall that we begin with the Duncan-Mortensen-Zakai equation given by

$$-\mathrm{d}\tilde{p} = f^T \nabla \tilde{p} \mathrm{d}t + \frac{1}{2} \operatorname{Tr}\left(\mathbf{S}_{\eta_x} \nabla^2 \tilde{p}\right) \mathrm{d}t + \tilde{p} g^T \mathrm{d}y \ .$$

We now definine the function

$$\Lambda(\mathbf{x}, t) = \exp\{ \int_0^t g(\mathbf{x}, s)^T \mathrm{d}\mathbf{y}(s) - \frac{1}{2} \int_0^t \|g(\mathbf{x}, s)\|^2_{\mathbf{S}_{\eta_y}} \mathrm{d}s \} \ .$$

so that, assuming a differentiable observation path,

$$\frac{\partial \Lambda(\mathbf{x}, t)}{\partial t}\Big|_{t=s} = \left(g(\mathbf{x}, s)^T \nabla_t \mathbf{y}\Big|_{t=s} - \frac{1}{2} \|g(\mathbf{x}, s)\|^2_{\mathbf{S}_{\eta_y}} \right) \Lambda(\mathbf{x}, s)$$

Krishnamurthy and Elliott (2002) show that

$$-\frac{\partial}{\partial t}(\Lambda \cdot \tilde{p}) = \left((\nabla \tilde{p})^T f + \frac{1}{2} \operatorname{Tr}\left(\mathbf{S}_{\eta_x} \nabla^2 \tilde{p}\right) \right) \Lambda \ .$$

Thus using the product rule of differentiation in conjecture with the definition of Λ we obtain

$$\left((\nabla \tilde{p})^T f + \frac{1}{2} \operatorname{Tr}\left(\mathbf{S}_{\eta_x} \nabla^2 \tilde{p}\right) \right) \Lambda = - \left(g^T \nabla_t \mathbf{y} - \frac{1}{2} \|g\|^2_{\mathbf{S}_{\eta_y}} \right) \Lambda \tilde{p} - \Lambda \frac{\partial \tilde{p}}{\partial t} \ .$$

By dividing through Λ and rearranging, we may now recover the generalised form of (3.11)

$$-\partial_t \tilde{p} = g^T \nabla_t \mathbf{y} \tilde{p} - \frac{1}{2} \|g\|^2_{\mathbf{S}_{\eta_y}} \tilde{p} + (\nabla \tilde{p})^T f + \frac{1}{2} \operatorname{Tr}\left(\mathbf{S}_{\eta_x} \nabla^2 \tilde{p}\right) \ .$$

Assuming, as done in the main text, zero observations, i.e., $y(\cdot) = 0$, the first term drops out and we recover (3.11).

Appendix C

Supplamentary Results to Chapter 4

In the following an alternative proof of convergence for the updates in the discounted infinite horizon case (cf. Section 4.1.3) is given. As in the main text we assume the cost is bounded. Hence, $\exists \bar{\mathcal{C}}$ s.t. $\forall \pi \ \bar{\mathcal{C}} \geq \left\langle \sum_k \gamma^k \mathcal{C}_\bullet(\mathbf{x}_k, \mathbf{u}_k) \right\rangle_{q_\pi}$. For notational convenience we shall also assume $\eta = 1$. We begin by showing that

Proposition C.1. *Let $\{\pi^i\}$ be a sequence of policies generated by (4.11) and let $\hat{\pi}$ be an arbitrary (stochastic) policy, then*

$$\lim_{N\to\infty} -\frac{1}{N}\sum_{i=0}^{N} \bar{\Psi}^{i+1}(x) \leq \mathbb{E}_{q_{\hat{\pi}}}\left[\sum_{t=0}^{\infty} \gamma^t \mathcal{C}_\bullet(\mathbf{x}_k, \mathbf{u}_k)\right]$$

The proof is by induction on the time horizon using the following two lemmas. The base case is given by

Proposition C.2. *For any $\epsilon > 0$ there exists N_ϵ s.t. for all $N > N_\epsilon$*

$$-\frac{1}{N}\sum_{i=0}^{N} \Psi^i(x_0) \leq \mathbb{E}_{q_{\hat{\pi}}}\left[\mathcal{C}_\bullet(x_0, u_0)\right] + \gamma\bar{\mathcal{C}} + \epsilon$$

Proof. Consider

$$\begin{aligned}
&\mathrm{KL}\left(\hat{\pi}\|\pi^{n+1}\right) - \mathrm{KL}\left(\hat{\pi}\|\pi^{n}\right) \\
&\qquad = \int_u \hat{\pi}(u|x) \log \frac{\pi^n}{\pi^{n+1}} \\
&\qquad = \int_u \hat{\pi}(u|x) \log \exp\{\mathcal{C}(x,u) - \gamma \int_y P(y|x,u)\bar{\Psi}^i(y) + \bar{\Psi}^{i+1}(x)\} \\
&\qquad = \int_u \hat{\pi}(u|x) \left[\mathcal{C}_\bullet(x,u) - \gamma \int_y P(y|x,u)\bar{\Psi}^i(y)\right] + \bar{\Psi}^{i+1}(x)
\end{aligned}$$

$$\leq \int_u \hat{\pi}(u|x) \left[\mathcal{C}_\bullet(x,u) + \gamma\bar{\mathcal{C}}\right] + \bar{\Psi}^{i+1}(x)$$

Summing the bound over $i = 1..N$ we have

$$\mathrm{KL}\left(\hat{\pi}\|\pi^N\right) - \mathrm{KL}\left(\hat{\pi}\|\pi^0\right) \leq N \int_u \hat{\pi}(u|x) \left[\mathcal{C}_\bullet(x,u) + \gamma\bar{\mathcal{C}}\right] + \sum_{i=0}^{N} \bar{\Psi}^{i+1}(x)$$

and hence

$$\frac{1}{N}\sum_{i=0}^{N} \bar{\Psi}^{i+1}(x) \leq \int_u \hat{\pi}(u|x) \left[\mathcal{C}_\bullet(x,u) + \gamma\bar{\mathcal{C}}\right] + \frac{1}{N}\mathrm{KL}\left(\hat{\pi}\|\pi^0\right) \ .$$

■

The following inductive step completes the proof of Proposition C.1.

Proposition C.3. *Assume for a given K and any $\epsilon > 0$ there exists N_ϵ s.t. for all $N > N_\epsilon$*

$$-\frac{1}{N}\sum_{n=0}^{N} \bar{\Psi}^n(x) \leq \mathbb{E}_{q_{\hat{\pi}}}\left[\sum_{k=0}^{K} \gamma^k C(\mathbf{x}_k, \mathbf{u}_k) + \gamma^K \bar{C}\right] + \epsilon$$

then for any $\delta > 0$ there exists N_δ s.t. for all $N > N_\delta$

$$-\frac{1}{N}\sum_{n=0}^{N} \bar{\Psi}^n(x) \leq \mathbb{E}_{q_{\hat{\pi}}}\left[\sum_{k=0}^{K+1} \gamma^k \mathcal{C}_\bullet(\mathbf{x}_k, \mathbf{u}_k) + \gamma^{K+1} \bar{C}\right] + \delta$$

Proof. Consider

$$\mathrm{KL}\left(\hat{\pi}\|\pi^{n+1}\right) - \mathrm{KL}\left(\hat{\pi}\|\pi^n\right)$$
$$= \int_u \hat{\pi}(u|x) \left[\mathcal{C}_\bullet(x,u) - \gamma \int_y P(y|x,u)\bar{\Psi}^n(y)\right] + \bar{\Psi}^{n+1}(x) \ .$$

Summing the bound over $i = 1..N$ we have

$$\mathrm{KL}\left(\hat{\pi}\|\pi^N\right) - \mathrm{KL}\left(\hat{\pi}\|\pi^0\right)$$
$$\leq \sum_{n=0}^{N} \int_u \hat{\pi}(u|x) \left[\mathcal{C}_\bullet(x,u) - \gamma \int_y P(y|x,u)\bar{\Psi}^n(y)\right] + \sum_{n=0}^{N} \bar{\Psi}^{n+1}(x)$$

and therefore

$$
\begin{aligned}
&-\frac{1}{N}\sum_{n=0}^{N}\bar{\Psi}^{n+1}(x)\\
&\leq \int_u \hat{\pi}(u|x)\left[\mathcal{C}_\bullet(x,u)-\gamma\int_y P(y|x,u)\frac{1}{N}\sum_{n=0}^{N}\bar{\Psi}^{n}(y)\right]+\frac{1}{N}\,\mathrm{KL}\left(\hat{\pi}\|\pi^0\right)\\
&\leq \int_u \hat{\pi}(u|x)\left[\mathcal{C}_\bullet(x,u)-\gamma\int_y P(y|x,u)\frac{1}{N}\mathbb{E}_{q_{\hat{\pi}}}\left[\sum_{k=0}^{K}\gamma^k\mathcal{C}_\bullet(x,u)+\gamma^K\bar{\mathcal{C}}\right]+\epsilon\right]\\
&\qquad+\frac{1}{K}\,\mathrm{KL}\left(\hat{\pi}\|\pi^0\right)\\
&=\mathbb{E}_{q_{\hat{\pi}}}\left[\sum_{k=0}^{K+1}\gamma^k\mathcal{C}_\bullet(x,u)+\gamma^{K+1}\bar{\mathcal{C}}\right]+\frac{1}{K}\,\mathrm{KL}\left(\hat{\pi}\|\pi^0\right)
\end{aligned}
$$

and the result follows. ■

Using the above, we may now show the desired result analogous to Proposition 4.9 in the main text.

Proposition C.4. *Let the cost be bounded and let π^n be a sequence of policies generated by (4.11) with π^0 s.t. $\forall x\ \mathrm{KL}\left(\pi^*(\cdot|x)\|\pi^0(\cdot|x)\right)<\infty$ then*

$$
\lim_{n\to\infty}\mathbb{E}_{q_{\pi^n}}\left[\sum_{k=0}^{\infty}\gamma^k\mathcal{C}_\bullet(\mathbf{x}_k,\mathbf{u}_k)\right]=\min_{\pi}\mathbb{E}_{q_\pi}\left[\sum_{k=0}^{\infty}\gamma^k\mathcal{C}_\bullet(\mathbf{x}_k,\mathbf{u}_k)\right]
$$

Proof. As $\hat{\pi}$ in Proposition C.1 is arbitrary we may choose the tightest bound given by[1]

$$
\hat{\pi}=\pi^*=\operatorname*{argmin}_{\pi}\mathbb{E}_{q_\pi}\left[\mathcal{C}(\bar{\mathbf{x}},\bar{\mathbf{u}})\right]\ ,
$$

where we use the notation $\mathcal{C}(\bar{\mathbf{x}},\bar{\mathbf{u}})=\sum_{k=0}^{\infty}\gamma^k\mathcal{C}_\bullet(\mathbf{x}_k,\mathbf{u}_k)$. Now as for a given x_0

$$
\mathbb{E}_{q_{\pi^n}}\left[\mathcal{C}(\bar{\mathbf{x}},\bar{\mathbf{u}})\right]\leq-\bar{\Psi}^n(x_0)\ ,
$$

we have

$$
\lim_{n\to\infty}\frac{1}{N}\mathbb{E}_{q_{\pi^n}}\left[\mathcal{C}(\bar{\mathbf{x}},\bar{\mathbf{u}})\right]\leq\lim_{n\to\infty}-\frac{1}{N}\sum_{i=0}^{N}\bar{\Psi}^{n+1}(x_0)\leq\mathbb{E}_{q_{\pi^*}}\left[\mathcal{C}(\bar{\mathbf{x}},\bar{\mathbf{u}})\right]\ .
$$

[1] n.b., we assume the π^* is not a limit point, if this is the case we may proceed along the lines of Proposition 4.6 by picking an ϵ-good policy.

As the lhs is the average expected cost over $\pi^1 \dots \pi^N$ there exists $n \in 1 \dots N$ s.t.

$$\mathbb{E}_{q_{\pi^{N+1}}}\left[\mathcal{C}(\bar{\mathbf{x}}, \bar{\mathbf{u}})\right] \leq \mathbb{E}_{q_{\pi^n}}\left[\mathcal{C}(\bar{\mathbf{x}}, \bar{\mathbf{u}})\right] \leq \mathbb{E}_{q_{\pi^*}}\left[\mathcal{C}(\bar{\mathbf{x}}, \bar{\mathbf{u}})\right] \ ,$$

with the first inequality following from Proposition 4.1. Noting that by the definition of π^* the rhs is also a lower bound gives the required result. ■

Appendix D

Supplamentary Results to Chapter 5

The following proposition establishes the result concerning global optimality of PPI for MDPs. The result can be extended to LQG problems, as in this case the iteration stays within a class of policies with finite dimensional parametrisation.

Proposition D.1. *Let $\mathcal{X}$ and $\mathcal{U}$ be finite and $\{\pi^n\}$ be a sequence of policies generated by (5.2), then*

$$\pi^n \to \operatorname*{argmin}_{\pi} -\frac{1}{\eta} \log \mathbb{E}_{q_\pi} \left[\exp\{-\eta \mathcal{C}(\bar{\mathbf{x}}, \bar{\mathbf{u}})\}\right]$$

Proof. We may write the policy in terms of a suitable distribution over deterministic policies τ and in particular $\pi^n \propto \int P(\mathbf{u}_k|\mathbf{x}_k, \tau(\cdot))P^n(\tau(\cdot))$ where $P(\mathbf{u}_k|\mathbf{x}_k, \tau(\cdot)) = \delta_{\mathbf{u}_k=\tau(\mathbf{x}_k)}$. Under this notation the iteration becomes

$$P^{n+1}(\tau(\cdot)) = \frac{1}{Z} P(\bar{r} = 1|\tau(\cdot))P^n(\tau(\cdot)) \ ,$$

with Z a normalisation constant. Expanding from P^0 for n iterations we therefore have

$$P^n(\tau(\cdot)) \propto \left[P(\bar{r} = 1|\tau(\cdot))\right]^n P^0(\tau(\cdot))$$

and hence for $n \to \infty$, $P(\bar{r} = 1|\tau(\cdot))$ dominates and $P^n(\tau)$ converges to the delta at the maximum of

$$P(\bar{r} = 1|\tau(\cdot)) = \mathbb{E}_{q_\tau} \left[\exp\{-\eta \mathcal{C}(\bar{\mathbf{x}}, \bar{\mathbf{u}})\}\right] \ .$$

As log is strictly monotonic and $\eta > 0$ this establishes the result. ■

Appendix E

Supplementary Results to Chapter 6

We present two supplemetary results ommited from the main text. The first demonstrates convergence of $\bar{\Psi}$ to Ψ in the infinity norm, based on convergence in Hilbert space norm demonstrate in Proposition 6.2. The second result utilises the first to demonstrate convergence of $\hat{\mathcal{V}}$, the approximation of the value function associated with $\bar{\Psi}$.

Proposition E.1. *Under the assumptions of Proposition 6.2 in the main text,*

$$\|\bar{\Psi}_i(\mathbf{x}) - \Psi_i(\mathbf{x})\|_\infty$$

converges to zero in probability.

Proof. From Proposition 6.2 we have

$$\|\bar{\Psi}_{i+1} - \Psi_{i+1}\| \to 0 \tag{E.1}$$

Let $\hat{\mathcal{E}}^k_{\mathcal{D}}[\cdot]$ be the empirical estimate of $\mathcal{E}^k[\cdot]$, then

$$\begin{aligned}
\bar{\Psi}_i(\mathbf{x}) &= \left\langle \Phi \otimes \bar{\Psi}_{i+1}, \hat{\mathcal{E}}^k_{\mathcal{D}}[X_{i+1}|\mathbf{x}] \right\rangle \\
&= \left\langle \Phi \otimes (\bar{\Psi}_{i+1} + \Psi_{i+1} - \Psi_{i+1}), \hat{\mathcal{E}}^k_{\mathcal{D}}[X_{i+1}|\mathbf{x}] \right\rangle \\
&= \left\langle \Phi \otimes \Psi_{i+1}, \hat{\mathcal{E}}^k_{\mathcal{D}}[X_{i+1}|\mathbf{x}] \right\rangle + \left\langle \Phi \otimes (\bar{\Psi}_{i+1} - \Psi_{i+1}), \hat{\mathcal{E}}^k_{\mathcal{D}}[X_{i+1}|\mathbf{x}] \right\rangle \ .
\end{aligned}$$

Hence

$$\begin{aligned}
\|\bar{\Psi}_i(\mathbf{x}) - \Psi_i(\mathbf{x})\|_\infty =& \sup_{\mathbf{x}} |\bar{\Psi}_i(\mathbf{x}) - \Psi_i(\mathbf{x})| \\
=& \sup_{\mathbf{x}} |\bar{\Psi}_i - \mathbb{E}_{X_{i+1}|\mathbf{x}} [\Phi \Psi_{i+1}] | \\
\leq& \sup_{\mathbf{x}} | \left\langle \Phi \otimes (\bar{\Psi}_{i+1} - \Psi_{i+1}), \hat{\mathcal{E}}^k_{\mathcal{D}} [X_{i+1}|\mathbf{x}] \right\rangle | \\
&+ \sup_{\mathbf{x}} | \left\langle \Phi \otimes \Psi_{i+1}, \hat{\mathcal{E}}^k_{\mathcal{D}} [X_{i+1}|\mathbf{x}] - \mathcal{E}^k [X_{i+1}|\mathbf{x}] \right\rangle | \\
\leq& \|\Phi \otimes (\bar{\Psi}_{i+1} - \Psi_{i+1})\| \sup_{\mathbf{x}} \|\hat{\mathcal{E}}^k_{\mathcal{D}} [X_{i+1}|\mathbf{x}] \| \\
&+ \|\Phi \otimes \Psi_{i+1}\| \sup_{\mathbf{x}} \|\hat{\mathcal{E}}^k_{\mathcal{D}} [X_{i+1}|\mathbf{x}] - \mathcal{E}^k [X_{i+1}|\mathbf{x}] \| \\
=& \|\Phi\| \sup_{\mathbf{x}} \|\hat{\mathcal{E}}^k_{\mathcal{D}} [X_{i+1}|\mathbf{x}] \| \|\bar{\Psi}_{i+1} - \Psi_{i+1}\| \\
&+ \|\Phi \otimes \Psi_{i+1}\| \sup_{\mathbf{x}} \|\hat{\mathcal{E}}^k_{\mathcal{D}} [X_{i+1}|\mathbf{x}] - \mathcal{E}^k [X_{i+1}|\mathbf{x}] \|
\end{aligned}$$

and the result follows by (E.1) and the consistency of $\hat{\mathcal{E}}^k_{\mathcal{D}} [\cdot]$. ■

Proposition E.2. *Let the optimal value function be bounded, say* $\mathcal{V}(\cdot, t) < c$ *then,*

$$\|\bar{\Psi}(\cdot, t) - \Psi(\cdot, t)\|_\infty \to 0 \Longrightarrow \|\hat{\mathcal{V}}(\cdot, t) - \mathcal{V}(\cdot, t)\|_\infty \to 0 \ .$$

Proof. As we have $\Psi(\cdot, \cdot) = \exp\{-\lambda^{-1}\mathcal{V}(\cdot, \cdot)\}$

$$0 < \mathcal{V}(\cdot, t) < c \Rightarrow \exists c' \text{ s.t. } \Psi(\cdot, t) > c' > 0 \ .$$

Now

$$\begin{aligned}
\|\hat{\mathcal{V}}(\cdot, t) - \mathcal{V}(\cdot, t)\|_\infty =& \sup_{\mathbf{x}} |\hat{\mathcal{V}}(\mathbf{x}, t) - \mathcal{V}(\mathbf{x}, t)| \\
=& \lambda \sup_{\mathbf{x}} | \log \frac{\bar{\Psi}(\mathbf{x}, t)}{\Psi(\mathbf{x}, t)} | \\
=& \lambda \sup_{\mathbf{x}} | \log (\frac{\bar{\Psi}(\mathbf{x}, t) - \Psi(\mathbf{x}, t)}{\Psi(\mathbf{x}, t)} + 1) | \\
\leq& \lambda \sup_{\mathbf{x}} | \log (\frac{\bar{\Psi}(\mathbf{x}, t) - \Psi(\mathbf{x}, t)}{c'} + 1) | \ ,
\end{aligned}$$

and thus

$$\begin{aligned}
\|\bar{\Psi}(\cdot, t) - \Psi(\cdot, t)\|_\infty \to 0 \Longrightarrow & \frac{\bar{\Psi}(\mathbf{x}, t) - \Psi(\mathbf{x}, t)}{c'} + 1 \to 1 \\
\Longrightarrow & \|\hat{\mathcal{V}}(\cdot, t) - \mathcal{V}(\cdot, t)\|_\infty \to 0 \ .
\end{aligned}$$

■

Bibliography

An, C., Atkeson, C., and Hollerbach, J. (1988). *Model-based control of a robot manipulator*. MIT press Cambridge, MA.

Archambeau, C., Cornford, D., Opper, M., and Shawe-Taylor, J. (2007). Gaussian process approximations of stochastic differential equations. *Journal of machine learning research*, 1:1–16.

Attias, H. (2003). Planning by probabilistic inference. In *Proc. of the 9th Int. Workshop on Artificial Intelligence and Statistics*.

Azar, M. G., Gomez, V., and Kappen, H. J. (2011). Dynamic policy programming with function approximation. In *Proc. of 14th Int. Conf. on Artificial Intelligence and Statistics*.

Barber, D. and Furmston, T. (2009). Solving deterministic policy (PO)MDPs using EM and antifreeze. In *Proc. of the 1st Int. Workshop on Learning and data Mining for Robots*.

Beal, M. (2003). *Variational algorithms for approximate Bayesian inference*. PhD thesis, University of London.

Bellman, R. (1957). *Dynamic Programming*. Princeton University Press.

Bensoussan, A. (1992). *Stochastic control of partially observable systems*. Cambridge University Press Cambridge.

Berlinet, A. and Thomas-Agnan, C. (2004). *Reproducing kernel Hilbert spaces in probability and statistics*. Kluwer Academic.

Berret, B., Ivaldi, S., Nori, F., and Sandini, G. (2011). Stochastic optimal control with variable impedance manipulators in presence of uncertainties and delayed feedback. In *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4354–4359. IEEE.

Bertsekas, D. (1995). *Dynamic programming and optimal control*. Athena Scientific Belmont.

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.

Boltyanskiy, V., Gamkrelidze, R., Mishchenko, Y., and Pontryagin, L. (1962). Mathematical theory of optimal processes.

Braun, D., Howard, M., and Vijayakumar, S. (2012). Optimal variable stiffness control: formulation and application to explosive movement tasks. *Autonomous Robots*, 33(3):237–253.

Broek, J. v. d., Wiegerinck, W., and Kappen, H. (2010). Risk sensitive path integral control. In *Proc. of the 26th Conf. on Uncertainty in Artificial Intelligence*.

Broek, J. v. d., Wiegerinck, W., and Kappen, H. (2011). Stochastic optimal control of state constrained systems. *International Journal of Control*, 84(3):597–615.

Bryson, A. E. and Ho, Y. C. (1975). *Applied optimal control*. Hemisphere/Wiley.

Chhabra, M. and Jacobs, R. (2006). Properties of synergies arising from a theory of optimal motor behavior. *Neural computation*, 18(10):2320–2342.

Cover, T. M. and Thomas, J. A. (1991). *Elements of Information Theory*. Wiley.

Daniel, C., Neumann, G., and Peters, J. (2012). Hierarchical relative entropy policy search. In *Proc. of the 15th Int. Conf. on Artificial Intelligence and Statistics*.

d'Avella, A., Saltiel, P., and Bizzi, E. (2003). Combinations of muscle synergies in the construction of a natural motor behavior. *Nature neuroscience*, 6(3):300–308.

Dayan, P. and Hinton, G. E. (1997). Using Expectation Maximization for Reinforcement Learning. *Neural Computation*, 9:271–278.

Deisenroth, M. and Ohlsson, H. (2010). A probabilistic perspective on gaussian filtering and smoothing. Technical Report arXiv:1006.2165.

Deisenroth, M. and Rasmussen, C. (2011). Pilco: A model-based and data-efficient approach to policy search. In *Proc. of the 25th Int. Conf. on Machine Learning*.

Deisenroth, M., Rasmussen, C., and Peters, J. (2009). Gaussian process dynamic programming. *Neurocomputing*, 72:1508–1524.

Deisenroth, M., Turner, R., Huber, M., Hanebeck, U., and Rasmussen, C. (2012). Robust filtering and smoothing with Gaussian processes. *IEEE Transactions on Automatic Control*, 57(7):1865–1871.

Enoch, A., Sutas, A., Nakaoka, S., and Vijayakumar, S. (2012). BLUE: A bipedal robot with variable stiffness and damping. In *Proc. of. IEEE/RAS International Conference on Humanoid Robots*.

Fu, Y.-Y., Wu, C.-J., Su, K.-L., and Ko, C.-N. (2008). A time-scaling method for near-time-optimal control of an omni-directional robot along specified paths. *Artificial Life and Robotics*, 13(1):350–354.

Furmston, T. and Barber, D. (2010). Variational methods for reinforcement learning. In *Proc. of the 13th Int. Conf. on Artificial Intelligence and Statistics*.

Gärtner, T., Driessens, K., and Ramon, J. (2003). Graph kernels and gaussian processes for relational reinforcement learning. *Inductive Logic Programming*, 2835/2003:146–163.

Ghavamzadeh, M. and Engel, Y. (2007). Bayesian policy gradient algorithms. In *Proc. of Advances in Neural Information Processing Systems 19*.

Grebenstein, M., Albu-Schaffer, A., Bahls, T., Chalon, M., Eiberger, O., Friedl, W., Gruber, R., Haddadin, S., Hagn, U., Haslinger, R., et al. (2011). The DLR

hand arm system. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pages 3175–3182. IEEE.

Grünewälder, S., Lever, G., Baldassarre, L., Patterson, S., Gretton, A., and Pontil, M. (2012a). Conditional mean embeddings as regressors. In Langford, J. and Pineau, J., editors, *Proc. of the 29th Int. Conf. on Machine Learning*, ICML '12, pages 1823–1830, New York, NY, USA. Omnipress.

Grünewälder, S., Lever, G., Baldassarre, L., Pontil, M., and Gretton, A. (2012b). Modelling transition dynamics in MDPs with RKHS embeddings. In *Proc. of the 29th Int. Conf. on Machine Learning*.

Guestrin, C., Koller, D., Parr, R., and Venkataraman, S. (2003). Efficient solution algorithms for factored MDPs. *J. Artificial Intelligence Research*, 19:399–468.

Guigon, E., Baraduc, P., and Desmurget, M. (2008). Optimality, stochasticity, and variability in motor behavior. *Journal of computational neuroscience*, 24(1):57–68.

Hartikainen, J., Solin, A., and Särkkä, S. (2011). *Optimal Filtering with Kalman Filters and Smoothers*.

Havoutis, I. and Ramamoorthy, S. (2010). Geodesic trajectory generation on learnt skill manifolds. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*.

Hoffman, M., de Freitas, N., Doucet, A., and Peters, J. (2009a). An Expectation Maximization algorithm for continuous Markov Decision Processes with arbitrary rewards. In *Proc. of the 12th Int. Conf. on Artificial Intelligence and Statistics*.

Hoffman, M., Kueck, H., Doucet, A., and de Freitas, N. (2009b). New inference strategies for solving markov decision processes using reversible jump mcmc. In *Proc. of the 25th Conf. on Uncertainty in Artificial Intelligence*.

Hofmann, T., Schlkopf, B., and Smola, A. (2008). Kernel methods in machine learning. *The Annals of Statistics*, 36(3):1171–1220.

Honkela, A., Raiko, T., Kuusela, M., Tornio, M., and Karhunen, J. (2010). Approximate riemannian conjugate gradient learning for fixed-form variational bayes. *The Journal of Machine Learning Research*, 11:3235–3268.

Huh, D. and Sejnowski, T. (2011). Exact analytic solutions for optimal control problems under multiplicative noise. In *Proc. of the 18th World Congress of the Int. Federation of Automatic Control*.

Ivan, V., Zarubin, D., Toussaint, M., and Vijayakumar, S. (2013). Topology-based representations for motion planning and generalisation in dynamic environments with interactions. *International Journal of Robotics Research*, (in press).

Jacobson, D. (1967). *Differential Dynamic Programming Methods for Determining Optimal Control of Nonlinear Systems*. PhD thesis, University of London.

Kakade, S. (2001). A natural policy gradient. In *Proc. of Advances in Neural Information Processing Systems 14*.

Kalakrishnan, M., Righetti, L., Pastor, P., and Schaal, S. (2012). Learning force control policies for compliant robotic manipulation. In *Proc. of the 29th Int. Conf. on Machine Learning*.

Kalman, R. (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45.

Kappen, H. (2005). Path integrals and symmetry breaking for optimal control theory. *Journal of Statistical Mechanics: Theory and Experiment*, (11):P11011.

Kappen, H. (2011). Optimal control theory and the linear bellman equation. *Inference and Learning in Dynamic Models*, pages 363–387.

Kappen, H., Gomez, V., and Opper, M. (2009). Optimal control as a graphical model inference problem. Technical Report arXiv:0901.0633v2.

Kober, J. and Peters, J. (2009). Policy search for motor primitives in robotics. In Koller, D., Schuurmans, D., and Bengio, Y., editors, *Proc. of Advances in Neural Information Processing Systems 21*, Cambridge, MA. MIT Press.

Kober, J. and Peters, J. (2012). Reinforcement learning in robotics: a survey. *Reinforcement Learning*, 12:579–610.

Kohl, N. and Stone, P. (2004). Policy gradient reinforcement learning for fast quadrupedal locomotion. In *Proc. of IEEE Int. Conf. on Robotics and Automation*, volume 3, pages 2619–2624. IEEE.

Krishnamurthy, V. and Elliott, R. (2002). Robust continous-time smoothers without two-sided stochastic integrals. *IEEE Transactions on Automatic Control*, 2:386–394.

Kulchenko, P. and Todorov, E. (2011). First-exit model predictive control of fast discontinuous dynamics: Application to ball bouncing. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*.

Lagoudakis, M. and Parr, R. (2003). Least-squares policy iteration. *The Journal of Machine Learning Research*, 4:1107–1149.

Lang, T. (2011). *Planning and Exploration in Stochastic Relational Worlds*. PhD thesis, Freie Universität Berlin.

Li, W. (2006). *Optimal Control for Biological Movement Systems*. PhD thesis, University of California San Diego.

MacKay, D. (2003). *Information theory, inference and learning algorithms*. Cambridge university press.

Marcus, S., Fernández-Gaucherand, E., Hernández-Hernandez, D., Coraluppi, S., and Fard, P. (1997). Risk sensitive markov decision processes. *Progress in Systems and Control Theory*, 22:263–280.

Mensink, T., Verbeek, J., and Kappen, H. (2010). EP for efficient stochastic control with obstacles. In *Proc. of the 19th European Conference on Artificial Intelligence*.

Minka, T. (2001). *A family of algorithms for approximate Bayesian inference*. PhD thesis, Massachusetts Institute of Technology.

Minka, T. (2005). Divergence measures and message passing. Technical Report MSR-TR-2005-173, Microsoft Research Technical Report.

Mitrovic, D., Klanke, S., and Vijayakumar, S. (2010a). Adaptive optimal feedback control with learned internal dynamics models. *From Motor Learning to Interaction Learning in Robots*, 264:65–84.

Mitrovic, D., Nagashima, S., Klanke, S., Matsubara, T., and Vijayakumar, S. (2010b). Optimal feedback control for anthropomorphic manipulators. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*.

Mitter, S. and Newton, N. (2000). The duality between estimation and control. Published in Festschrift for A. Bennoussan.

Morimoto, J., Zeglin, G., and Atkeson, C. (2003). Minimax differential dynamic programming: Application to a biped walking robot. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*.

Nakanishi, J., Fukuda, T., and Koditschek, D. (2000). A brachiating robot controller. *IEEE Transactions on Robotics and Automation*, 16(2):109–123.

Nakanishi, J., Rawlik, K., and Vijayakumar, S. (2011). Stiffness and temporal optimization in periodic movements: An optimal control approach. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*.

Nakanishi, J. and Vijayakumar, S. (2012). Exploiting passive dynamics with variable stiffness actuation in robot brachiation. In *Robotics: Science and Systems VIII*.

Øksendal, B. (2010). *Stochastic differential equations: an introduction with applications*. Springer.

Peters, J., Mulling, K., and Altun, Y. (2010). Relative entropy policy search. In *Proc. of. 24th AAAI Conference on Artificial Intelligence*.

Peters, J. and Schaal, S. (2007). Reinforcement Learning by reward-weighted regression for operational space control. In *Proc. of the 24th Int. Conf. on Machine Learning*.

Peters, J. and Schaal, S. (2008). Reinforcement learning of motor skills with policy gradients. *Neural Networks*, 21(4):682–697.

Petersen, K. and Pedersen, M. (2008). The matrix cookbook. Technical report, Technical University of Denmark.

Raiko, T. and Tornio, M. (2005). Learning nonlinear state-space models for control. In *Proc. of Int. Joint Conf. on Neural Networks*.

Rawlik, K., Toussaint, M., and Vijayakumar, S. (2010). An approximate inference approach to temporal optimization in optimal control. In *Proc. of Advances in Neural Information Processing Systems 22*.

Rawlik, K., Toussaint, M., and Vijayakumar, S. (2012). On Stochastic Optimal Control and Reinforcement Learning by approximate inference. In *Proc. Robotics: Science and Systems VIII*.

Riedmiller, M., Peters, J., and Schaal, S. (2007). Evaluation of policy gradient methods and variants on the cart-pole benchmark. In *Proc. of IEEE Int. Symp. on Approximate Dynamic Programming and Reinforcement Learning (ADPRL 2007)*.

Sabes, P. N. and Jordan, M. I. (1996). Reinforcement Learning by probability matching. In *Proc. of Advances in Neural Information Processing Systems 9*.

Sahar, G. and Hollerbach, J. (1986). Planning of minimum- time trajectories for robot arms. *The International Journal of Robotics Research*, 5(3):90–100.

Smola, A., Gretton, A., Song, L., and Schlkopf, B. (2007). A hilbert space embedding for distributions. In *Algorithmic Learning Theory: 18th International Conference*.

Song, L., Gretton, A., Bickson, D., Low, Y., and Guestrin, C. (2011). Kernel belief propagation. In *Proc. of the 14th Int. Conf. on Artificial Intelligence and Statistics*.

Song, L., Huang, J., Smola, A., and Fukumizu, K. (2009). Hilbert space embeddings of conditional distributions. In *Proc. of the Int. Conf. on Machine Learning*.

Spong, M. (1995). The swing up control problem for the acrobot. *IEEE Control Systems*, 15(1):49–55.

Stengel, R. (1986). *Optimal Control and Estimation*. Dover Publications.

Sutton, R. and Barto, A. (1998). *Reinforcement Learning*. MIT Press, Cambridge.

Szepesvári, C. (2010). Algorithms for reinforcement learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 4(1):1–103.

Tang, J. and Abbeel, P. (2010). On the connection between importance sampling and the likelihood ratio policy gradient. In *Proc. of Advances in Neural Information Processing Systems 23*.

Tassa, Y. and Todorov, E. (2011). High-order local dynamic programming. In *IEEE Symposium on Adaptive Dynamic Programming And Reinforcement Learning*, pages 70–75. IEEE.

Theodorou, E., Buchli, J., and Schaal, S. (2009). Path integral-based stochastic optimal control for rigid body dynamics. In *IEEE Symposium on Adaptive Dynamic Programming And Reinforcement Learning*.

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

Theodorou, E., Tassa, Y., and Todorov, E. (2010b). Stochastic differential dynamic programming. In *Proc. of the American Control Conference*, pages 1125–1132. IEEE.

Theodorou, E. A. (2011). *Iterative Path Integral Stochastic Optimal Control: Theory and Applications to Motor Control*. PhD thesis, University of Southern California.

Todorov, E. (2006a). Linearly-solvable Markov Decision problems. In *Proc. of Advances in Neural Information Processing Systems 19*.

Todorov, E. (2006b). Optimal control theory. In Doya, K., editor, *Bayesian Brain: Probabilistic Approaches to Neural Coding*, pages 269–298. MIT Press.

Todorov, E. (2009a). Compositionality of optimal control laws. In *Proc. of Advances in Neural Information Processing Systems 22*.

Todorov, E. (2009b). Efficient computation of optimal actions. *Proc. of the National Academy of Sciences*, 106:11478–11483.

Todorov, E., Hu, C., Simpkins, A., and Movellan, J. (2010). Identification and control of a pneumatic robot. In *Proc. of the IEEE RAS and EMBS Int. Conf. on Biomedical Robotics and Biomechatronics*, pages 373–380. IEEE.

Todorov, E. and Jordan, M. (2002). Optimal feedback control as a theory of motor coordination. *Nature Neuroscience*, 5(11):1226–1235.

Togelius, J., Schaul, T., Wierstra, D., Igel, C., Gomez, F., and Schmidhuber, J. (2009). Ontogenetic and phylogenetic reinforcement learning. *Künstliche Intelligenz*, 23(3):30–33.

Toussaint, M. (2009a). Pros and cons of truncated gaussian ep in the context of approximate inference control. NIPS Workshop on Probabilistic Approaches for Robotics and Control.

Toussaint, M. (2009b). Robot trajectory optimization using approximate inference. In *Proc. of the 26th Int. Conf. on Machine Learning*, pages 1049–1056. ACM.

Toussaint, M., Plath, N., Lang, T., and Jetchev, N. (2010a). Integrated motor control, planning, grasping and high-level reasoning in a blocks world using probabilistic inference. In *Proc. of the Int. Conf. on Robotics and Automation*.

Toussaint, M. and Storkey, A. (2006). Probabilistic inference for solving discrete and continuous state Markov Decision Processes. In *Proc. of the 23rd Int. Conf. on Machine Learning*, pages 945–952.

Toussaint, M., Storkey, A., and Harmeling, S. (2010b). Expectation-maximization methods for solving (PO)MDPs. In Barber, D., editor, *Inference and Learning in Dynamic Models*. Cambridge University Press. In print.

Van Der Merwe, R. (2004). *Sigma-point Kalman filters for probabilistic inference in dynamic state-space models*. PhD thesis, University of Stellenbosch.

Vlassis, N. and Toussaint, M. (2009). Model-free Reinforcement Learning as mixture learning. In *Proc. of the 26th Int. Conf. on Machine Learning*, pages 1081–1088. ACM.

Vlassis, N., Toussaint, M., Kontes, G., and Piperidis, S. (2009). Learning model-free robot control by a Monte Carlo EM algorithm. *Autonomous Robots (Special issue on Robot Learning)*, 27:123–130.

Yedidia, J., Freeman, W., and Weiss, Y. (2003). Understanding belief propagation and its generalizations. *Exploring artificial intelligence in the new millennium*, 8:236–239.

Yedidia, J. S., Freeman, W. T., and Weiss, Y. (2005). Constructing free-energy approximations and generalized belief propagation algorithms. *IEEE Transactions on Information Theory*, 51(7):2282–2312.

Yong, J. and Zhou, X. (1999). *Stochastic controls: Hamiltonian systems and HJB equations*. Springer Verlag.

Zarubin, D., Ivan, V., Toussaint, M., Komura, T., and Vijayakumar, S. (2012). Hierarchical motion planning in topological representations. In *Proc. of Robotics: Science and Systems VIII*.

Zhong, M. and Todorov, E. (2011a). Aggregation methods for linearly-solvable MDPs. In *World Congress of the Int. Federation of Automatic Control*.

Zhong, M. and Todorov, E. (2011b). Moving least-squares approximations for linearly-solvable stochastic optimal control problems. *Journal of Control Theory and Applications*, 9:451–463.