CHAPTER 1
INTRODUCTION

In a broad sense, this dissertation explores the development of reinforcement learning based techniques suited to applications in nonlinear filtering (or nonlinear state estimation) and Markov chain Monte Carlo (MCMC) algorithms. During the initial development of this dissertation, a special class of approximate nonlinear filters called the feedback particle filter (FPF) was the primary focus. Later, by lucky coincidence, it was observed that similar techniques could be applied to obtain interesting results in MCMC algorithms as well. Hence, the chapter on MCMC forms a smaller portion of the dissertation.

The goal of this chapter is a cursory introduction of the elements that are key to the problems considered in this dissertation. In Section 1.1, we describe the two major goals undertaken in the areas of nonlinear filtering and MCMC. Subsequently, in Section 1.2, we introduce the various tools used, motivate the solution approaches adopted, without delving much into the technical details.

1.1 Goals of the Dissertation

1.1.1 Nonlinear Filtering/State Estimation

Consider a dynamic system evolving in time according to a given mathematical model. A complete characterization of the system is given by its states. Uncertainties in the system model or external disturbances that affect the state are modeled as process noise, and indirect observations of the state, corrupted by measurement noise are available. The observation model and the noise statistics are assumed to be known. The state dynamics and observations may be in either continuous or discrete time depending on the system properties. Additionally, the state dynamics are assumed to be Markovian, i.e., in rough terms, the probability of the current state just depends on the previous state, and not on the entire history. These assumptions will be made more precise in Chapter 4.

A generic block diagram of a state estimator is depicted in Fig. 1-1. The goal of any filtering/state estimation problem is to recursively estimate the states of the system based on

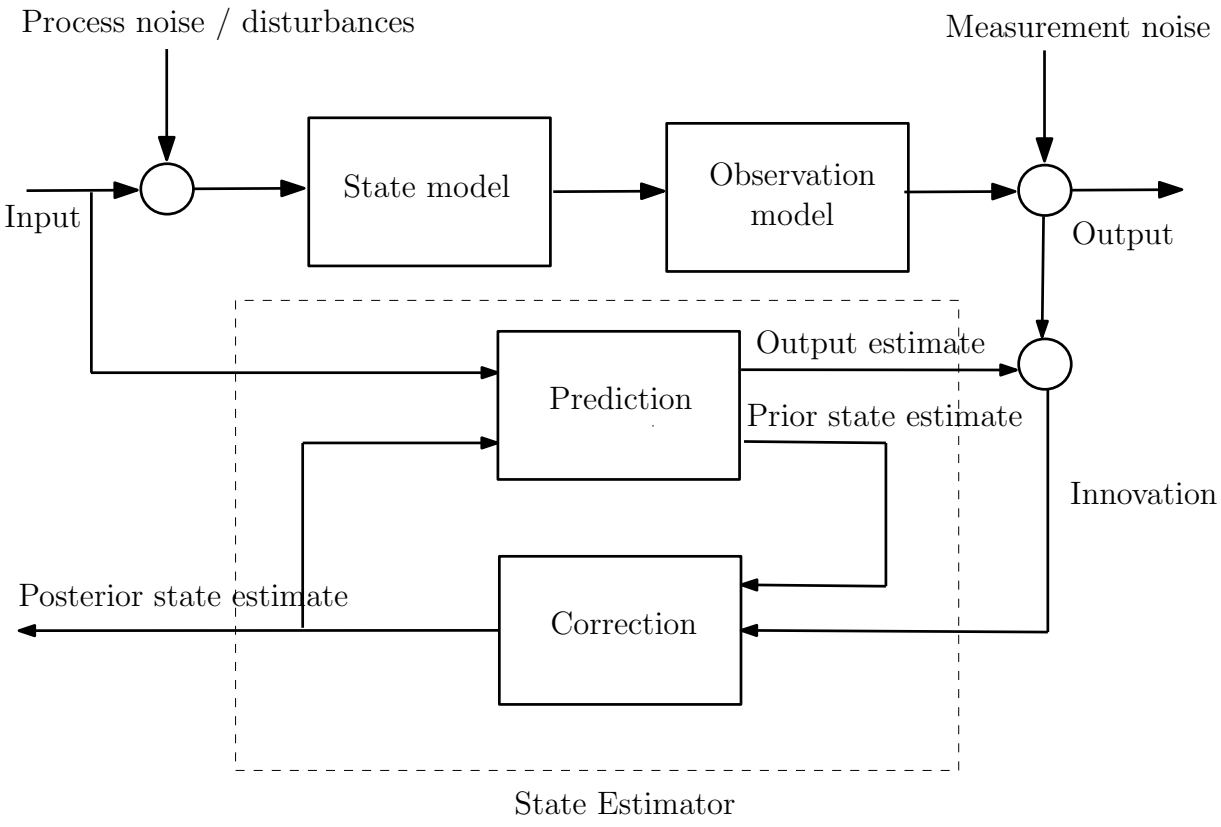

Figure 1-1. Block diagram of a state estimator

noisy partial observations. The top row of the block diagram, with the state and observation models, indicates the actual evolution of the system. The prediction and correction blocks form part of the state estimator. Given the current estimate of the state, the prediction block gives the best state estimate, called the prior estimate for the next time instant using the state model. The correction block updates the prior after receiving the most recent observation, and outputs the posterior estimate. All filtering approaches share this basic structure, although in some cases, the prediction and correction steps may be combined.

Initial applications of filtering included satellite orbit determination, aircraft navigation and tracking [3]. More recently, filtering has found applications in diverse areas such as machine learning [4], queueing networks, mathematical finance [5] and data assimilation problems for weather forecasting [6].

Question:Need to verify if the block diagram is correct

I believe it is ok

Question:A simple example may be good here.

Maybe. Not necessary here

When the system dynamics are linear and the noise quantities are Gaussian, the problem is simpler and the well-known Kalman filter is the optimal solution. The Kalman filter gives a linear SDE (stochastic differential equation) for the conditional mean of the state and a Riccati type ODE for the state covariance. These two quantities completely characterize the posterior estimate. However, in practice, the linear state dynamics with Gaussian disturbances assumption is often violated. For example, in weather forecasting the states evolve via a complex system of fluid mechanics equations that are nonlinear. The optimal solution is given by a set of SDEs, like the Zakai's equation or the Kushner-Stratonovich equation. The posterior estimates are in the form of conditional distributions of the state, given the entire history of observations. A detailed discussion of the nonlinear filtering theory can be found in [7].

Linear approximations like the extended Kalman filter (EKF) were studied for application to nonlinear systems. They perform well as long as the state or observation dynamics do not deviate significantly from linearity. Later with the advent of modern computing, Monte Carlo based methods like the conventional bootstrap particle filter gained popularity. The underlying principle here is to approximate the posterior distribution using empirical samples called particles. Budhiraja et al. [8] provide a comprehensive survey of numerical methods for nonlinear filtering problems.

The main focus of this dissertation is a class of controlled particle system algorithms called the feedback particle filter (FPF). Feedback particle filter was originally formulated for the continuous-time nonlinear filtering problem in the Euclidean setting [1]. They have since been extended to Riemannian manifolds and matrix Lie groups [9]. The FPF is similar in its feedback structure to the Kalman filter and in its empirical approximation approach to the standard particle filter. In other respects, they are significantly different. A crucial component of the FPF is the gain function, which is analogous to the Kalman gain in the Kalman filter. The optimal gain function in the FPF is obtained as the gradient of the solution to a particular version of Poisson's equation [1, 10]. Obtaining an analytical solution to the Poisson's equation

is often difficult and hence, approximation is required. In this dissertation, our main focus is on developing algorithms to approximate the gain function. A detailed discussion of the FPF theory and gain approximation algorithms is ~~reserved for~~ Chapter 4.

1.1.2 Markov Chain Monte Carlo (MCMC) Algorithms

The second application of interest is Markov chain Monte Carlo (MCMC) algorithms. MCMC algorithms have a long history of being applied to problems in Bayesian statistics [].

In standard Monte Carlo methods, expectation of a function f of a random variable X distributed according to a density ρ is approximated ~~empirically as~~,

$$\mathsf{E}_{X\sim\rho}[f(X)] := \int_{\mathsf{X}} f(x)\rho(x)dx \approx \frac{1}{N}\sum_{i=1}^{N} f(X_i) \tag{1-1}$$

where each X_i is distributed according to ρ and N is sufficiently large. As is often the case, it may be difficult to generate a sequence of samples $\{X_i\}_1^N$ according to the desired target distribution ρ. Methodologies such as rejection sampling and importance sampling make use of an easy-to-sample surrogate distribution to sample from the original target distribution. But, if ~~X~~ is high-dimensional, it is difficult to find a closely matching simple surrogate distribution.

MCMC algorithms provide an alternative solution in this situation. They are a special class of Monte Carlo methods in which the samples X_i are the states of an ergodic Markov chain. Given the target density ρ, the problem reduces to designing an appropriate transition kernel for a Markov chain that has ρ as its invariant density. The Langevin diffusion is a continuous-time Markov process, which can be thought of as a perturbed gradient flow with respect to a potential function. It ~~forms the basis of~~ many MCMC algorithms. The Gibbs algorithm [11] and Metropolis-Hastings (M-H) algorithm [12] are ~~other~~ popular discrete-time MCMC techniques. They have been widely applied for problems in Bayesian inference, statistical physics, computation biology etc.

The ~~asymptotic~~ convergence of the empirical averages of the form (1-1) to the true expected value is guaranteed under general conditions by law of large numbers. The main drawback of these techniques as compared to standard Monte Carlo sampling which provides

independent and identically distributed (i.i.d.) samples, is that the successive samples of the Markov chain are correlated to each other. This results in slower convergence of the algorithms to the target density. The central limit theorem states that,

$$\sqrt{N}\Big(\frac{1}{N}\sum_{i=1}^{N} f(X_i) - \mathsf{E}_{X\sim\rho}[f(X)]\Big) \xrightarrow{d} \mathcal{N}(0,\sigma_\infty^2), \qquad \text{as } N\to\infty, \tag{1-2}$$

where $\mathcal{N}(0,\sigma_\infty^2)$ refers to the Gaussian distribution with zero mean and variance σ_∞^2. The asymptotic variance σ_∞^2 is a measure quantifying the rate of convergence; lower its value, faster is the convergence of the Markov chain to its invariant distribution and hence, the goal is to minimize it. Asymptotic variance can be expressed in terms of the solution to Poisson's equation [13].

Control variates, which are zero-mean terms added to the function f, have been used to reduce the asymptotic variance of the estimates without adding any bias. Henderson, in his dissertation [14], notes that the best choice of control variates can be constructed using the solution to Poisson's equation. They also feature prominently in Chapter 11 of the book [13] with the objective of constructing reduced-variance estimators for network models. Control variates constructed using the fluid value function have been shown to produce a 100-fold reduction in variance over the standard estimator for the KSRS queueing model in the examples considered in the book chapter.

In this dissertation, we demonstrate that the same algorithms we propose for approximating the FPF gain function, find additional application in improving the performance of popular MCMC algorithms. A detailed discussion on MCMC, including numerical examples demonstrating asymptotic variance reduction is provided in Chapter 5.

1.2 Tools Used in the Dissertation

Now, that the main application areas of the dissertation have been described briefly, we introduce the various tools that help us achieve our goal. In Section 1.2.1, a preliminary description of the Poisson's equation, that is crucial to both our applications of interest is given.

1.2.1 Poisson's Equation

In its most general form, Poisson's equation is a second-order differential equation of the form,

$$\mathcal{D}h := -f,$$

where $\mathcal{D}$ is a second-order differential operator. Usually, $f \in C^2$ is given and is "centered" by subtracting its mean. The function h is unknown and is called the solution to the Poisson's equation. In physics, the operator $\mathcal{D}$ is often taken to be the Laplacian. Poisson's equation appears widely in the context of Markov chains and stochastic optimal control. In the context of a continuous-time diffusion process, the operator $\mathcal{D}$ refers to the infinitesimal generator, also called the differential generator. .

Poisson's equation is central to average-cost optimal control theory. In this case, f is a one-step cost function and h is called a relative value function. Relative value function gives the infinite-horizon expected cost when starting from a given state under this/a stationary policy. Approximate solutions to the equation lead to direct performance bounds of the control algorithm [13]. Explicit bounds on the solution h have been obtained under general conditions of the chain in [].

Our interest lies in a particular version of Poisson's equation associated to the Langevin diffusion process. Langevin diffusion is discussed in greater detail in Sections section 2.1 and **??**. Gradient of the solution to this equation is the optimal choice of the gain function associated with the FPF [1]. In MCMC algorithms, as noted by Henderson [14] and later by Dellaportas et al. [16], the optimal control variates can be constructed from this solution. Thus, Poisson's equation and its solution are central to the goals of this dissertation.

Obtaining a closed form solution is difficult outside of special cases and this motivates the study of approximation algorithms. Finding an approximate solution falls within the framework of reinforcement learning and in particular, temporal difference (TD) learning. In this dissertation, we develop variants of the TD learning algorithm that can approximate the

Anand: If $f \equiv 0$, then h is precisely a harmonic function [15]

Question:The states evolve in the form of a controlled Markov chain based on a given policy.

Question:citation needed

gradient of the solution to Poisson's equation directly. Other approaches include the Markov semigroup approximation by Taghvaei et al. [17].

1.2.2 Reinforcement Learning and TD Learning

In this section, a beginner level introduction to reinforcement learning algorithms is provided. Reinforcement learning algorithms have gained popularity over the last decade having achieved major successes in a wide variety of applications like AlphaGo, backgammon etc. In a general setting, such algorithms involve learning what actions to take in a given situation, so as to maximize a numerical reward (or equivalently minimize a numerical cost) over a (possibly infinite) time-horizon. The learned set of actions, called a policy is a mapping from the state space to the action space. In a stochastic setting, this mapping is expressed in terms of probability of taking a particular action in a given state. The learning is performed purely based on interactions with the environment without any prior knowledge of the system model. A whole variety of algorithms including Q-learning [18] and temporal difference (TD) learning [19] belong to this category. A slightly different class that makes use of model information is called approximate dynamic programming. Although, the end objective is to obtain optimal policies, a central theme in all these algorithms is value function approximation. This aspect is what makes these algorithms an attractive choice for our objective.

Sutton and Barto write in their monograph [20], "If one had to identify one idea as central and novel to reinforcement learning, it would undoubtedly be temporal difference learning". Originally introduced by Sutton in [19], TD learning algorithms address the problem of policy evaluation associated with discrete-time stochastic optimal control problems called Markov decision processes (MDPs). In other words, for a given fixed policy, the algorithm computes estimates of the value function through an iterative procedure. A large body of prior research is available that studies the asymptotic convergence properties of these algorithms. Most of them, however are restricted to either the discounted-cost case or an undiscounted-cost (average-cost) setting for a finite state space Markov chain with an absorbing state. Both