Adaptive Control of Nonlinear Stochastic Systems by Particle Filtering

A. Greenfield, A. Brockwell
Carnegie Mellon University
Pittsburgh, PA

Abstract

We introduce an adaptive moving horizon control scheme for nonlinear stochastic systems. The scheme uses the recently developed particle filter to track the hidden state, as well as to estimate unknown parameters. In addition, expected costs are approximated by Monte Carlo integration where necessary. Although computationally intensive, the scheme has wide applicability, and we demonstrate its robustness in simulations.

Keywords: adaptive control, Bayesian control, moving horizon control, particle filtering, nonlinear systems

1 Introduction

A number of adaptive control schemes have been proposed to regulate unknown stochastic systems. Among the most well-known of these is the self-tuning regulator of [1], which, along with various generalizations, has been studied extensively (see, e.g., [2]). Although [6] established stability of the self-tuning regulator when applied to time-varying systems, such results have proved difficult to extend to more complex nonlinear systems. Perhaps the most significant recent development in this direction has been the control-Lyapunov function based work of [5] and [4], but these methods require a substantial amount of effort to be allocated to construction of an appropriate Lyapunov function.

In this paper we propose the use of a Bayesian approach to adaptive control, in conjunction with a standard moving horizon control scheme, and recently developed nonlinear filtering methods known as "particle filters". (See, e.g., [3], for a nice description of these techniques.) The approach is quite similar to that proposed in [7], but we extend it to an adaptive control setting, where the system involves one or more unknown parameters which must be estimated. The Bayesian approach provides a formal framework within which we can specify costs, which may take into account parameter estimation error as well as state and control trajectory costs. The particle filtering methods allow us to carry out Bayesian parameter and state estimation for a quite general class of nonlinear stochastic systems.

We consider nonlinear stochastic discrete-time systems, in which one or more parameters are unknown. To identify the system in a Bayesian framework, unknown parameters are assigned prior distributions, and at every point in time, current information about system parameters is updated by computing a new posterior distribution on those parameters. This update procedure is simple in principle, but in practice, can be difficult to implement.

The following example illustrates the kind of difficulties which arise. The well-known Kalman filter, when applied to the linear Gaussian system

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t + \epsilon_{t+1} \\ y_t &= Cx_t + \nu_t, \end{aligned} \tag{1}$$

can be thought of as implementing exactly these recursions when x_0 is assigned a Gaussian prior. In this case, the posterior distribution of x_t, given $y_0, y_1, \ldots, y_t$ is itself Gaussian, and its mean vector and covariance matrix come directly out of the Kalman filter. When one of the matrices A,B or C contains unknown parameters, which we will denote by θ, it is tempting to reformulate the system in the form

$$\begin{aligned} x^*_{t+1} &= A^*x^*_t + B^*u_t + \epsilon^*_{t+1} \\ y_t &= C^*x^*_t + \nu^*_t, \end{aligned} \tag{2}$$

where $x^*_t = (x_t, \theta)^T$,

$$A^* = \begin{bmatrix} A & 0 \\ 0 & I \end{bmatrix}, B^* = \begin{bmatrix} B \\ 0 \end{bmatrix}, C^* = [C, 0],$$

with $\nu^*_t = (\nu_t, 0)^T$ and $\epsilon^*_t = (\epsilon_t, 0)^T$. Doing this ensures that the posterior (or "filtering") distribution $p(x^*_t|y_t, y_{t-1}, \ldots, y_1)$ includes as a marginal distribution the posterior distribution $p(\theta|y_t, \ldots, y_1)$ of the unknown parameter vector θ at time t. Unfortunately, the system of equations (2) is no longer linear since A^*, B^* and C^* depend on θ, and hence the Kalman filter cannot be used to find these posteriors. This problem is only exacerbated when the original system is nonlinear.

Particle filtering methods, however, are ideally suited to finding the desired posterior distributions. Loosely speaking, these methods approximate filtering and predictive distributions with samples rather than specifying them in a parametric form. A full description of the standard particle filter is beyond the scope of this paper, but the interested reader is referred to [3] for further details. For the sake of understanding the algorithm in this paper, it is sufficient to know that each posterior distribution $p(x_t|y_1,\ldots,y_t)$ is represented by a collection of M particles $\{\xi_t^{(i)},\ i=1,2,\ldots,M\}$. The collection of particles can be regarded as an approximate sample of size M from the posterior distribution, and it follows that properties of the posterior distribution can be estimated directly from the particles. Specifically, for functions $g(\cdot)$ which are integrable with respect to the posterior distribution, one can estimate $\int g(x_t)p(x_t|y_1,\ldots,y_t)dx_t$ by $\hat{g} = M^{-1}\sum_{i=1}^{M} g(\xi_t^{(i)})$.

2 The Control Algorithm

We are interested in controlling a nonlinear stochastic system of the form

$$\begin{aligned} x_{t+1} &= f(x_t, u_t, \epsilon_t; \theta) \\ y_t &= g(x_t, \nu_t; \theta), \end{aligned} \tag{3}$$

where $x_t \in \mathbb{R}^p$, $u_t \in \mathbb{R}^q$, $y_t \in \mathbb{R}^s$, $\{\epsilon_t\}$ and $\{\nu_t\}$ are independent and identically distributed sequences of random variables, $f(\cdot)$ and $g(\cdot)$ are some specified functions, and θ is a vector of unknown parameters. As usual, we assume that the process $\{y_t\}$ is observed, while $\{x_t\}$ remains hidden. We will use I_t to denote observations up to time t, that is, $I_t = \{y_s,\ s \le t\}$,, with I_{-1} equal to the empty set. Again, we adopt the approach of incorporating the unknown parameter vector into a new state vector, defining $x_t^* = (x_t, \theta)$. This ensures that the posterior (or filtering) distribution $p(x_t^*|I_t)$ contains information about *both* the hidden state x_t and the parameter vector θ.

We propose the following scheme to carry out approximate optimal control of the unknown system (3), in the face of parameter uncertainty.

Adaptive Bayesian Moving Horizon Control Algorithm

1. **Initialization:** Specify a prior distribution for θ. Set $t = 0$ and choose some positive integer "horizon" h. Initialize the particle filter.
2. **Parameter update step:** Observe y_t. Carry out one iteration of the particle filter, to compute the new posterior distribution for the hidden state and unknown parameter vector, that is, $p(x_t^*|I_t)$.
3. **Moving horizon control step:** Choose $u_t,\ldots,u_{t+h-1}$ so as to minimize the expected value of the cost $J(x_{t+1},\ldots,x_{t+h},u_t,\ldots,u_{t+h-1})$, given I_t. The expectation is taken over the distribution of the unobserved state x_t, and the random variation in the system itself, as well as over the distribution $p(\theta|I_t)$.
4. **Control:** Apply the control signal u_t to the system.
5. **Iteration:** Replace t by $t+1$ and go back to step 2.

One approach to carrying out Step 3 is to perform direct numerical minimization of $E[J(\ldots)]$, evaluating the expectation by Monte-Carlo integration for each possible choice of $\{u_t,\ldots,u_{t+h-1}.\}$ Thus $E[J(\ldots)]$ is approximated by

$$\begin{aligned} &\hat{J}(x_{t+1},\ldots,x_{t+h},u_t,\ldots,u_{t+h-1}) \\ &\quad = \sum_{m=1}^{M}\sum_{i=1}^{N} J(x_{t+1}^{(i,m)},\ldots,x_{t+h}^{(i,m)},u_t,\ldots,u_{t+h-1}), \end{aligned}$$

where $\{x_{t+1}^{(i,m)},\ldots,x_{t+h}^{(i,m)}\}$ represents the ith of N simulations of the system, using the control sequence $u_t,\ldots,u_{t+h-1}$, starting with x_t and θ taken directly from the particle $\xi_t^{(m)}$.

Note that most optimization algorithms do not perform particularly well when evaluations of the target function are perturbed by random noise. Hence it is necessary to use a large value of N in the expression for $\hat{J}$ given above. There are a number of stochastic programming techniques, however (see, in particular, [**?**]), which could potentially lead to more efficient implementation of Step 3.

3 Simulation Studies

We study the performance of the algorithm by simulation studies. We compare the perform of the algorithm to that of the LQG controller applied to the same problem.

3.1 A Simple Second-Order System

We first consider the second-order system

$$\begin{aligned} x_{t+1} &= \begin{bmatrix} 1 & 1 \\ 0 & \phi \end{bmatrix} x_t + \begin{bmatrix} 0 \\ g \end{bmatrix} u(t) + \begin{bmatrix} 0 \\ \epsilon_{t+1} \end{bmatrix}, \\ y_t &= \begin{pmatrix} 1, & 0 \end{pmatrix} x_t + \nu_t, \end{aligned} \tag{4}$$

where ϵ_t and ν_t are independent Gaussian noise sequences with means zero and variances equal to one.

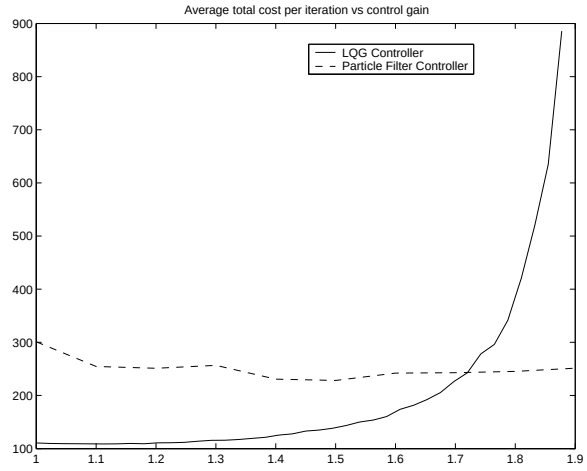

Figure 1: Average costs over multiple simulations of the system (4) with the standard LQG controller and the particle-filter controller.

This system represents the motion of a particle subject to a force, in the presence of friction and random disturbances.

We consider two controllers. The first is a moving horizon version of an LQG regulator, which, at each point t in time, chooses the control so as to minimize the expected value of the cost

$$J = \sum_{s=t+1}^{t+5} x_t^T Q x_t + u_{t-1}^2,$$

when $\phi = 0.8$ and $g = 1$. The second is the particle filtering-based controller described above, using the same cost J, but restricting the selection of the control at each point in time to the case where $u_{t+1}, \ldots, u_{t+5}$ are all equal. (This restriction on the control makes the optimization in Step 2 of the algorithm simpler.)

Figure 1 shows average costs for simulations of the test system 4 for each of the two controllers, as the actual gain g of the system varies from 1 to 2. Clearly, the optimal LQG controller performs well when the gain g is close to the value 1 the controller is designed for, but the particle filter outperforms the LQG controller once the gain increases beyond about 1.7, and gives relatively good performance over the entire range of gains. Note that the moving-horizon LQG controller has an advantage here because of the restriction we impose on the choice of control signal for the particle-filter controller. This advantage would be reduced if we were to remove this restriction (which in general would incur higher computational costs).

3.2 A System of Unknown Order

We consider, as another example, a system of two masses moving in the plane with damping, and connected by a spring. The system is shown in Figure 2.

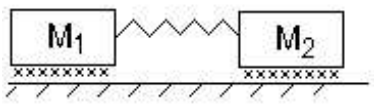

Figure 2: A system of unknown order.

The damping of the system is accurately known, as is the values of the two masses. The spring constant is the unknown parameter in the system and we consider its value restricted to two possibilities, either ∞ or a particular finite value. In the case where the spring is infinitely rigid, the system of two masses operates as a single mass and the system is 2nd-order. If the spring has finite rigidity, the system is 4th-order.

The 2nd and 4th-order models are discrete-time approximations to the true continuous-time dynamical systems, driven by Gaussian noise sequences. To be precise, the 2nd-order model is given by

$$\begin{aligned} x_{t+1} &= A_2 x_t + b_2 u_t + \epsilon_{t+1}^{(2)} \\ y_t &= \begin{pmatrix} 1, & 0 \end{pmatrix} x_t + \nu_t, \end{aligned} \tag{5}$$

where

$$A_2 = \begin{bmatrix} 1 & 1 \\ 0 & 1 - \frac{B}{m_1+m_2} \end{bmatrix}, \quad b_2 = \begin{bmatrix} 0 \\ \frac{1}{m_1+m_2} \end{bmatrix},$$

and $\epsilon_{t+1}^{(2)} = \begin{bmatrix} 0 \\ \epsilon_{t+1}^a \end{bmatrix}$,

and the 4th-order model is given by

$$\begin{aligned} x_{t+1} &= A_4 x_t + b_4 u_t + \epsilon_{t+1}^{(4)} \\ y_t &= \begin{pmatrix} 1, & 0, & 0, & 0 \end{pmatrix} x_t + \nu_t, \end{aligned} \tag{6}$$

where

$$A_4 = \begin{pmatrix} 1 & 1 & 0 & 0 \\ \frac{-K}{m_1} & 1 - \frac{B}{m_1} & \frac{K}{m_1} & 0 \\ 0 & 0 & 1 & 1 \\ \frac{K}{m_2} & 0 & \frac{-K}{m_2} & 1 - \frac{B}{m_2} \end{pmatrix}$$

$$b_4 = \begin{pmatrix} 0 \\ 0 \\ \frac{1}{m_1} \\ 0 \end{pmatrix}, \quad \text{and } \epsilon_{t+1}^{(4)} = \begin{pmatrix} 0 \\ \epsilon_{t+1}^a \\ 0 \\ \epsilon_{t+1}^b \end{pmatrix}.$$

We consider three controllers used to control these two models of the system. The first is a moving horizon LQG controller designed to control the 2nd-order (rigid spring) model. The second is a moving horizon LQG controller designed for the 4th-order model. The third is the proposed adaptive particle filter controller. The parameter θ is the unknown order of the system. Hence the particle filter adapts to the order of the system in

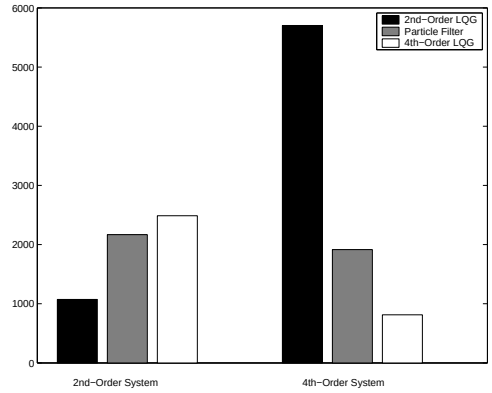

Figure 3: Average cost per iteration for the 2nd and 4th order LQG controllers and the particle filter controller.

order to generate an effective control. The cost function takes the same functional form as for the example in the previous subsection.

Figure 3 shows the average cost for each controller on both the 2nd and 4th-order systems above. As one would expect, for both the 2nd and 4th-order systems, the LQG controller designed for the correct system performed the best. The particle filter controller, however, is clearly far more robust, giving reasonable costs regardless of the order of the system.

4 Discussion

We have demonstrated a general-purpose moving horizon control algorithm which can be used for nonlinear stochastic systems with unknown parameters. The algorithm gathers information about the hidden state and the unknown parameters by use of the particle filter, rather than traditional variants of the Kalman filter. This means that we could expect it to perform reasonably well even for highly complex systems. Furthermore, the algorithm itself is relatively simple to implement, relying heavily on simulation rather than analysis.

There are, however, limitations. General forms of particle filtering are not practical for cases where either the state-dimension or the dimension of θ is very large, since prohibitively large numbers of particles would be required to get good approximations to posterior distributions. There are also a number of parameters associated with the particle filter itself (certain sampling distributions must be chosen), adding a degree of complexity to the implementation of the control law in this paper. One further issue arises in implementation of the particle filter. Incorporation of the parameter vector θ into the expanded state $x_t^* = (x_t, \theta)$ leads to a state-space model for which the second component of the state does not change. In general, this can lead to a problem known as "particle depletion" (see [3] for more details), and to get around it, we use the standard technique of allowing θ to vary slowly over time. While this solves the particle depletion problem, it means that the system is not exactly the same as originally specified. On the other hand, for many real-life systems, parameters do in fact vary slowly over time, so this is arguably not problematic. Furthermore, particle filtering is a currently active area of research, and we expect these problems to be mitigated in the future, as general purpose particle filtering techniques continue to be improved.

References

[1] K.J. Åstrom and B. Wittenmark. On self-tuning regulators. *Automatica*, 9:185–199, 1973.

[2] H.F. Chen and L. Guo. Continuous-time stochastic adaptive tracking - robustness and asymptotic properties. *SIAM Journal of Control and Optimization*, 28(3):513–527, 1990.

[3] A. Doucet, N. de Freitas, and N. Gordon, editors. *Sequential Monte Carlo Methods in Practice*. Springer, New York, 2001.

[4] R. A. Freeman and P. V. Kokotović. *Robust Nonlinear Control Design*. Birkhauser, 1996.

[5] M. Krstić, I. Kanellakopoulos, and P.V. Kokotović. *Nonlinear Adaptive Control Design*. Wiley, New York, 1995.

[6] S.P. Meyn and L. Guo. Stability, convergence, and performance of an adaptive control algorithm applied to a randomly varying system. *IEEE Transactions on Automatic Control*, 37(4), 1992.

[7] D. Salmond and N. Gordon. Particles and mixtures for tracking and guidance. In *Sequential Monte Carlo Methods in Practice*, pages 517–528. Springer, 2001.