2011 American Control Conference
on O'Farrell Street, San Francisco, CA, USA
June 29 - July 01, 2011

Robust Adaptive Optimal Control for Unknown Dynamical Systems

Tomonori Sadamoto and Masaki Yamakita

Abstract— **In this paper, we propose an algorithm of adaptive optimal control scheme for systems whose dynamics are unknown and the states are contaminated by noises. The basic control law is Policy Iteration which can solve HJB equation recursively. In the proposed method, the value function is estimated using a nonlinear filtering but the state of the system is not estimated since the system model is not available. Since the proposed method can reduce the effects of the noises without using the system model, we can apply this method to many practical systems without model and parameters.**

I. Introduction

In control problems for real plants, optimal control is popular and effective method. The optimal control has an advantage of minimizing cost functions which we define, however, we can not obtain good responses in cases where the system model includes modeling uncertainties. Since it is difficult to identify the accurate system model of nonlinear systems, the optimal control is not effective in those cases. For linear systems with a quadratic cost function the optimal control law can be derived by algebraic Riccati equation(ARE), in the nonlinear systems, the optimal control law can be derived by Hamilton Jacobi Bellman(HJB) equation. However, it is hard to be solved, and there are only offline techniques for obtaining an approximate optimal solution.

In a recent research, a method which can solve HJB equation for the nonlinear affine systems have been developed in [1]. Since the online technique called “Policy Iteration” is used in this method, it doesn't need the knowledge of the drift term of the system. It can solve HJB equation by updating control law recursively using responses of the system. Furthermore, an extended method which can be applied to systems whose dynamics are unknown has been developed in [2]. However, these methods require exact state observations, and it is pointed out that they don't work well when the states are contaminated by noises. Therefore, in this paper, we propose a method which can derive an approximate optimal control law even when state observations are contaminated by noises. We call the proposed method 'Robust Extended Policy Iteration (REPI)'. REPI can estimate the true cost function using a gradient method and a method of estimation of parameters. The effectiveness of the proposed method is shown by a numerical simulation.

This paper is organized as follows. In section II we state the problem formulation and HJB equation. EPI algorithm is explained in section III. We propose a method in section IV and V while the algorithm is summarized in section VI. A simulation result is given in section VII. Finally we conclude the paper in section VIII.

II. Problem formulation and HJB equation

The problem formulation is described in the former subsection. In the latter subsection, we explain HJB equation. The solution of the problem is equivalent to the solution of HJB equation.

A. Problem formulation

In this paper, we consider nonlinear affine systems described by

$$\begin{aligned} \dot{x}(t) &= f(x(t)) + g(x(t))u(t)\ ,\ x_0 = x(0) && (1) \\ y(t) &= x(t) + w(t)\ ,\ w \sim \mathcal{N}(0, \Sigma) && (2) \end{aligned}$$

where $x(t) \in \mathbb{R}^n$ is a state, $u(t) \in \mathcal{U} \subseteq \mathbb{R}^m$ is a control input, $y(t) \in \mathbb{R}^n$ is an output and $w(t)$ is an observation noise normally distributed with 0 mean and a covariance matrix Σ. $f(x) \in \mathbb{R}^n,\ g(x) \in \mathbb{R}^{n\times m}$ are also unknown and $f(x) + g(x)u$ is assumed to be Lipschitz continuous. Furthermore, the covariance matrix $\Sigma := [\sigma_{ij}]$ is also unknown, but the range of each element is known. Namely, for all elements $\sigma_{ij}, 1 \leq i, j \leq n$,

$$\sigma_{ij}^{\min} \leq \sigma_{ij} \leq \sigma_{ij}^{\max}$$

are satisfied while $\sigma_{ij}^{\min}$ and $\sigma_{ij}^{\max}$ are known.

We consider a cost function described by

$$J(x_0, u(\cdot)) = \int_0^\infty r(x(\tau), u(\tau))d\tau \qquad (3)$$

where $r(x, u) := x^T Q x + u^T R u$ is a stage cost while Q and R are positive definite matrices.

The optimal control problem is to find a control input $u(t)$ such that it stabilizes the system (1) while minimizes the cost function $J(x_0, u(\cdot))$.

B. HJB equation

In this subsection and next section, we assume there are no observation noises for illustrative purposes. We introduce a Hamiltonian described by

$$H(x, u, V_x) = r(x, u) + V_x\left(f(x) + g(x)u\right) \qquad (4)$$

where $V_x := \frac{\partial V}{\partial x}$. In the following sections, we call a function $V(x)$ *value function*. According to [3], $V^*(x)$ and

This work was not supported by any organizations.

T. Sadamoto and M. Yamakita are with the Department of Mechanical and Control Engineering, Tokyo Institute of Technology, Japan, yamakita@ac.ctrl.titech.ac.jp

978-1-4577-0079-8/11/$26.00 ©2011 AACC

$u^*(x)$ are an optimal function and an optimal input if and only if they satisfy the following equations:

$$\begin{aligned} 0 &= \min_{u^*}\left[H\left(x, u^*(x), (V_x^*)^T\right)\right], && (5)\\ u^*(x) &= -\frac{1}{2}R^{-1}g^T(x)\left(V_x^*(x)\right)^T. && (6)\end{aligned}$$

III. Extended Policy Iteration

The optimal input can be obtained by solving HJB equation, however it is hard to be solved. To overcome this problem, EPI algorithm is proposed in [2]. Before exmlaining EPI, we summarize PI algorithm briefly(See the details in [1]).

PI algorithm

1. Policy Evaluation
   Determine a value function using responses of the system from some initial states. Namely, compute a value at time t described by
   $$V^{(i)}(x) = \int_t^{T_e} r(x(\tau), \mu^{(i)}(x(\tau)))d\tau. \quad (7)$$
   where T_e is a time such that the system is regarded as converged. Furthermore, approximate a value function as $V^{(i)}(x) := \theta_i^T\phi(x)$ and determine the parameter by a least-square method as
   $$\theta_{i+1} = \text{argmin}_\theta \sum_{j=1}^{L} |V^{(i)}(x_j) - \theta^T\phi(x_j)|^2, \quad (8)$$
   where L is a number of data, $\phi(x) \in \mathbb{R}^N$ is a basis function and $V^{(i)}(x_j)$ in (8) is computed by (7).
2. Policy Improvement
   $$\mu^{(i+1)}(x) = -\frac{1}{2}R^{-1}g^T(x)\left(V_x^{(i+1)}(x)\right)^T \quad (9)$$
3. $i \leftarrow i+1$, go to step 1.

In the algorithm, $\mu^{(i)}(x)$ and $V^{(i)}(x)$ indicate a state feedback law and a value function after i times updated. Furthermore, let an initial input $\mu^{(0)}(x)$ stabilize (1) asymptotically. If the function $V^{(i)}(x)$ is accurately approximated then $\mu^{(i)}(x) \to \mu^*(x)$ and $V^{(i)}(x) \to V^*(x)$ as $i \to \infty$.

Since it requires input dynamics $g(x)$ in (9), we cannot obtain $\mu^{(i+1)}(x)$ because $g(x)$ is unknown. EPI introduces integrators in front of unknown systems, in order to transform the systems into augmented ones whose input dynamics is known, i.e.,

$$\begin{aligned}\dot{x}_a &= \begin{bmatrix}\dot{x}\\ \dot{u}\end{bmatrix} = \begin{bmatrix} f(x)+g(x)u \\ 0\end{bmatrix} + \begin{bmatrix}0\\ I\end{bmatrix}v\\ &:= f_a(x_a) + g_a(x_a)v && (10)\end{aligned}$$

where $x_a := \left[x^T, u^T\right]^T$ and a new feedback law is $v = \mu_a(x_a)$. We define a new cost function corresponding to (3) as follows.

$$\begin{aligned} J_a(x_a(t), v(\cdot)) &:= \int_t^\infty r_a(x_a(\tau), v(\tau))d\tau && (11)\\ r_a(x_a, v) &:= x_a^T Q_a x_a + v^T R_a v && (12)\end{aligned}$$

where $Q_a := \text{diag}(Q, R)$ and R_a are positive definite matrices. Note that right hand side of (11) is equivalent to the original cost function (3) when $R_a \to 0$ [2].

This algorithm can approximately solve HJB equation for the nonlinear affine systems without the knowledge of $f(x)$ and $g(x)$.

IV. Reduction noise effect for cost

The EPI algorithm assumes that a true state is available. In the following sections, we consider a situation where there are observation noises and an output (2) is available instead of a true state. The proposed method is mainly described in this and next section. The method is constructed by 3 steps, namely, (a) estimation of true values (b) averaging the estimated values and (c) reduction the effects of noises for a basis function. In this section, (a) and (b) steps are stated.

The cost while $t \in [kT, (k+1)T)$ in the iteration $i = 0, 1, \cdots$ is obtained by

$$C_k^{\mu^{(i)}} = \int_{kT}^{(k+1)T} r(y(\tau), \mu^{(i)}(y(\tau)))d\tau. \quad (13)$$

For simple notation, we consider a sample-path from an initial state x_0 to the origin under the policy $\mu^{(i)}(x)$. Then, $C_k^{\mu^{(i)}}$ can be written as C_k in short. Although the output and the value are dependent on the iteration number, it is dropped unless otherwise stated.

Since the cost which we can observe is obtained by $y(t)$ instead of $x(t)$, the value is given by

$$\begin{aligned}\hat{v}_k = &\int_{kT}^{MT} y(\tau)^T Q y(\tau) + \mu(y(\tau))^T R\mu(y(\tau))d\tau \\ &+\hat{v}_M\ ,\ k = 0, 1, \cdots, M-1.\end{aligned} \quad (14)$$

where M is a sufficiently large number and satisfies $T_e = MT$. Hereafter, we call this value *observed value* and the value expressed by

$$\begin{aligned}v_k = &\int_{kT}^{MT} x(\tau)^T Q x(\tau) + \mu(x(\tau))^T R\mu(x(\tau))d\tau \\ &+v_M\ ,\ k = 0, 1, \cdots, M-1.\end{aligned} \quad (15)$$

true value. If there are no observation noises, costs are nearly equal to zero near the origin, hence the values are also nearly zero near the origin. However, integrated term in (14) contains square of the noise w, hence $\mathbb{E}[\hat{v}_k - v_k] \neq 0$. Therefore, the observed values are not nearly zero near the origin. Namely, the observed values become much larger than that in the case of no noises. Therefore, it is necessary to suppress the effects of noises in (14). To achieve this purpose, REPI estimates true values from the observed values.

A. Derivation of discrete time model

In this subsection, we introduce a discrete time model to estimate the true values. Since a true value at time MT is nearly zero, it is convenient to compute the value function in a reverse time. Namely, we regard a time MT as an initial time and the value is computed backward. The observed value and the true value are described by

$$\begin{aligned}\hat{v}_k &= \int_0^{kT} y(\tau)^T Q y(\tau) + \mu(y(\tau))^T R \mu(y(\tau)) d\tau \\ &+ \hat{v}_0\ ,\ k = 1, 2, \cdots, M\end{aligned} \tag{16}$$

$$\begin{aligned}v_k &= \int_0^{kT} x(\tau)^T Q x(\tau) + \mu(x(\tau))^T R \mu(x(\tau)) d\tau \\ &+ v_0\ ,\ k = 1, 2, \cdots, M\end{aligned} \tag{17}$$

where v_0 is the value of $\int_{MT}^{\infty} r(x,u)d\tau$ and $\hat{v}_0$ is its observed value respectively. Approximate (16) as

$$\begin{aligned}\hat{v}_k &\simeq \int_0^{kT} x(\tau)^T Q x(\tau) + \mu(x(\tau))^T R \mu(x(\tau)) d\tau \\ &+ \int_0^{kT} w^T Q w + \left(\left.\frac{\partial \mu}{\partial x}\right|_{x=y} w \right)^T R \left(\left.\frac{\partial \mu}{\partial x}\right|_{x=y} w \right) d\tau \\ &+ \hat{v}_0,\end{aligned} \tag{18}$$

$$\begin{aligned}\hat{v}_0 &= \theta^T \phi(x+w) \\ &\simeq \theta^T \phi(x) + \theta^T \left.\frac{\partial \phi}{\partial x}\right|_{x=y} w = v_0 + W_0.\end{aligned} \tag{19}$$

Please note here that a linear term with respect to w can be considered zero for a fixed state x since the mean value of w is zero. From (17), recursive relation of v_k implies

$$\begin{aligned}v_k &= \int_0^{(k-1)T} x(\tau)^T Q x(\tau) + \mu(x(\tau))^T R \mu(x(\tau)) d\tau \\ &+ \int_{(k-1)}^{kT} x(\tau)^T Q x(\tau) + \mu(x(\tau))^T R \mu(x(\tau)) d\tau + v_0 \\ &= \int_{(k-1)}^{kT} x(\tau)^T Q x(\tau) + \mu(x(\tau))^T R \mu(x(\tau)) d\tau \\ &+ v_{k-1}\ ,\ k = 1, 2, \cdots, M.\end{aligned} \tag{20}$$

Therefore, a discrete time model of the value function is derived from (18) and (20) as follows:

$$\begin{cases} v_k = v_{k-1} + \int_{(k-1)}^{kT} x^T Q x + \mu(x)^T R \mu(x) d\tau \\ \eta_k := \int_0^{kT} \left(w^T Q w + w^T \gamma(y) w \right) d\tau + W_0 \\ \hat{v}_k = v_k + \eta_k. \end{cases} \tag{21}$$

where $\gamma(y) := \left(\left.\frac{\partial \mu}{\partial x}\right|_{x=y} \right)^T R \left(\left.\frac{\partial \mu}{\partial x}\right|_{x=y} \right)$.

However, we cannot estimate v_k with this model because

1) $\int_{(k-1)}^{kT} x(\tau)^T Q x(\tau) + \mu(x(\tau))^T R \mu(x(\tau)) d\tau$ is unknown
2) A covariance matrix Σ is unknown

First, we consider the problem 1). We consider moving mean random walk models defined by

$$\begin{aligned}\alpha_k &= \alpha_{k-1} + \xi\ ,\ \xi \sim \mathcal{N}(b_{k-1}, \sigma_1^2) \\ b_k &= b_{k-1} + \zeta\ ,\ \zeta \sim \mathcal{N}(0, \sigma_2^2).\end{aligned} \tag{22}$$

and approximate $\alpha_{k-1} \simeq \int_{(k-1)T}^{kT} x(\tau)^T Q x(\tau) + \mu(x(\tau))^T R \mu(x(\tau)) d\tau$. Then, we have a new discrete time model described by

$$\begin{cases} \mathbf{x}_k = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{x}_{k-1} + \begin{bmatrix} 0 \\ \xi \\ \zeta \end{bmatrix} \\ \hat{v}_k = [1\ 0\ 0]\, \mathbf{x}_k + \eta_k \\ \mathbf{x}_k := [v_k, \alpha_k, b_k]^T \\ \eta_k := \int_0^{kT} \left(w^T Q w + w^T \gamma(y) w \right) d\tau + W_0 \end{cases} \tag{23}$$

where $\xi \sim \mathcal{N}(0, \sigma_1^2)$ and $\zeta \sim \mathcal{N}(0, \sigma_2^2)$. Note that this system is observable.

Next, we consider the problem 2). Although Σ is time-invariant, we introduce a slowly time-varying random walk model $s_k = s_{k-1} + \nu, \nu \sim \mathcal{N}(0, U)$ in order to robustfy the algorithm, where U is a design parameter and $s_k := [\sigma_{k,11}, \sigma_{k,12}, \cdots, \sigma_{k,1n}, \sigma_{k,21}, \cdots, \sigma_{k,nn}]^T \in \mathbb{R}^{n^2}$ and $\Sigma_k := [\sigma_{k,ij}]$. Then, an observation noise $w(t)$ satisfies following equations:

$$\begin{aligned}\mathbb{E}[w(t)^T Q w(t)] &= \mathbb{E}\left[\mathrm{tr}\left(Q w(t) w^T(t)\right)\right] = \mathrm{tr}\left(Q\Sigma\right), \\ \mathbb{E}\left[w(t)^T \gamma(y) w(t)\right] &= \mathrm{tr}\left(\gamma(y)\Sigma\right).\end{aligned} \tag{24}$$

Furthermore, we define $\Gamma(y_{k-1})$ and $\Lambda(y_{k-1})$ by

$$\begin{aligned}\Gamma(y_{k-1}) &:= [\Gamma_{ij}(y_{k-1})] = (Q + \gamma(y_{k-1})), \\ \Lambda(y_{k-1}) &:= [\Gamma_{11}(y_{k-1}), \cdots, \Gamma_{nn}(y_{k-1})].\end{aligned}$$

and assume that this system is ergodic and $\frac{\partial \mu}{\partial x}$ is constant over $t \in [(k-1)T, kT)$. Then we have

$$\eta_k \simeq \eta_{k-1} + \mathrm{tr}\left((Q + \gamma(y_{k-1}))\Sigma_{k-1}\right) T \tag{25}$$

$$= \eta_{k-1} + T\Lambda(y_{k-1}) s_{k-1}. \tag{26}$$

We estimate Σ in an interval such that the system has converged almost to zero. Then $v_k \simeq 0$ and $\hat{v}_k = v_k + \eta_k \simeq \eta_k$. Therefore, another discrete time model is given by

$$\begin{cases} \begin{bmatrix} \eta_k \\ s_k \end{bmatrix} = \begin{bmatrix} 1 & T\Lambda(y_{k-1}) \\ 0 & I \end{bmatrix} \begin{bmatrix} \eta_{k-1} \\ s_{k-1} \end{bmatrix} + \begin{bmatrix} 0 \\ \nu \end{bmatrix} \\ \hat{v}_k = \eta_k. \end{cases} \tag{27}$$

Note that $\sigma_{ij} = \sigma_{ji}$ and $\Gamma_{ij} = \Gamma_{ji}$, hence we can use a reduced model instead of (27). Furthermore, if we know the structures of Σ e.g., x_1 and x_2 are uncorrelated, then we can further reduce the dimension of (27) using the information explicitly.

The estimation procedure is summarized as follows. First, obtain the data along a sample-path. Second, estimate Σ using (27). Finally, estimate the true values using (23) where $w(t) \sim \mathcal{N}(0, \hat{\Sigma})$ while $\hat{\Sigma}$ is an estimated value of Σ.

B. Estimation of true values using E-CEnKF

We derived discrete time models for the estimation in previous subsection. Since there are some state constraints e.g., v_k is always non-negative and we know each bounds of σ_{ij}, we apply E-CEnKF(Efficient - Constraind Ensemble Kalman Filter) which can explicitly deal with nonlinear and non-gaussian noise and state constraints. For details, see

[4],[5]. The estimated value of $v^{(i)}(y_k)$ is denoted $v_{est}^{(i)}(y_k)$ in the following.

C. Averaging estimated values

An averaged value $\mathbb{E}\left[V(y_k)\right]$ may be more accurate approximation for a true value $V(x_k)$ rather than the estimated value $v_{est}^{(i)}(y_k)$. For an arbitraly output y_k, a new value is defined by

$$\bar{v}^{(i)}(y_k) := \frac{1}{i+1}\left(\sum_{j=0}^{i} V^{(j)}(y_k) + v_{est}^{(i)}(y_k)\right) \tag{28}$$

where i is a current iteration number, and $V^{(j)}$ is an approximated value function at j-th step. Since (28) requires full memories of parameters θ_j at $j = 0, 1, \cdots, i$, we use the following equivalent equation:

$$\bar{v}^{(i)}(y_k) = (1-\alpha_i)V^{(i)}(y_k) + \alpha_i v_{est}^{(i)}(y_k) \tag{29}$$

where $\alpha_i = \frac{1}{i+1}$. We determine the optimal parameter using $\bar{v}_k^{(i)}$ for policy improvement.

V. Reduction of noise effect for basis function

The value function is approximated by weighted sum of input data $\phi(y)$ and output data $V^{(i)}(y)$, and the parameter is determined by a least-square method. So far we have considered the influence of noises only to the output. In this section, we consider to suppress the effects of the noises to the basis function $\phi(y)$.

A. Policy Improvement

In a least-square method, an optimal parameter θ_* is derived by

$$\theta_* = \left(\Phi\Phi^T\right)^{-1}\Phi Y \tag{30}$$

where

$$\Phi := \left[\cdots|\phi(y_k)|\cdots\right] \ , \ Y := \begin{bmatrix} \vdots \\ \bar{v}^{(i)}(y_k) \\ \vdots \end{bmatrix}.$$

Since the noises are squared due to square operation $\Phi\Phi^T$ in (30), the parameter is always updated as $\theta_i \neq \theta_{i+1}$. Hence, we employ an update law by

$$\theta_{i+1} = \theta_* - \beta_i\left(\theta_* - \theta_i\right). \tag{31}$$

The second term in (31) is a feedback to keep the parameter 'small' and β_i is a gain.

Furthermore, not to continue to update, we also employ a dead zone method as

$$|\theta_* - \theta_i| < \epsilon_\theta \Rightarrow \beta_i = 1, \tag{32}$$

where ϵ_θ is a design parameter.

VI. REPI Algorithm

In sec.IV,V, we proposed a method to solve the optimal control problem given in Sec. II-A approximately. We call the algorithm 'Robust Extended Policy Iteration' in the sense of robustness against an observation noise and the pseudo-code is summarized as follows.

```
Step 0: Initialize
    Step 0a: Set $k = 0, i = 0, m = 0, x(0) = x_0, \delta_1, \delta_2, \Omega$ and $\theta_0$
    Step 0b: Set the iteration number for Policy Iteration $I$, a number of initial states $L$ and a sampling interval $T$
    Step 0c: Determine the first policy satisfied such that $\mu^{(0)}(0) = 0$ and $\mu^{(0)}(x)$ is continuous and stabilizes the system.
Step 1: Do for $i = 0, 1, \cdots, I$
Step 2: Do for $m = 1, 2, \cdots, L$
Step 3: Do for $k = 0, 1, \cdots, M-1$
    Step 3a: $y_k^m = x(kT) + w(kT)$
    Step 3b: Transit states following system dynamics while $t \in [kT, (k+1)T)$ using a feedback law $u(y) = \mu^{(i)}(y(t)), y(t) = x(t) + w(kT)$
    Step 3c: $y_{k+1}^m = x((k+1)T) + w((k+1)T)$
    Step 3d: If $|y_{k+1}^m| > \delta_2$ then reject data, set $x_0$ on $\Omega$ randomly, go back to Step 3.
    Step 3e:
        $C_k^{\mu^{(i)},m} = \int_{kT}^{(k+1)T} r(y, \mu^{(i)}(y))d\tau$
    Step 3f: $k \leftarrow k+1$
    Step 3g: If $|C_k^{\mu^{(i)},m}| < \delta_1$ then go to Step 4.
Step 4: $\hat{v}_M^{m,(i)} = \theta_i^T\phi(y_M)$
Step 5: Do for $j = M-1, M-2, \cdots, 1$
    Step 5a: $\hat{v}_j^{m,(i)} = C_k^{\mu^{(i)},m} + \hat{v}_{j+1}^{m,(i)}$
    Step 5b: $j \leftarrow j-1$
Step 6: As sec.IV-A ,IV-B, obtain estimated values $v_{est}^{m,(i)}(y_j)$ , $j = 1, 2, \cdots, M$ using the sequential data $\hat{v}_j^m, j = 1, \cdots, M-1$
Step 7: Do for $j = M-1, M-2, \cdots, 1$
    Step 7a: $\bar{v}^{m,(i)}(y_j) = (1-\alpha_i)V^{(i)}(y_j^m) + \alpha_i v_{est}^{m,(i)}(y_j)$
Step 8: Set $x_0$ on $\Omega$ randomly.
Step 9: $m \leftarrow m+1$, if $m \leq L$ then go to Step 2.
Step 10: Compute the parameter $\theta_*$ as minimizing a sum of squared approximation error $\hat{J}^{(i)}$.
        $\hat{J}^{(i)} := \sum_{m=1}^{L}\sum_{j=0}^{M-1}\left(\hat{\epsilon}_j^{m,(i)}\right)^2$
        $\hat{\epsilon}_j^{m,(i)} := \bar{v}_j^{m,(i)} - \theta_*^T\phi(y_j^m)$
Step 11: Update the parameter and the value function as
        $\theta_{i+1} = \theta_* - \beta_i(\theta_* - \theta_i)$ , $V^{(i+1)}(x) = \theta_{i+1}^T\phi(x)$.
```

Step 12: Update policy as

$$\mu^{(i+1)}(x) = -\frac{1}{2}R^{-1}g^T(x)\left(\frac{\partial V^{(i+1)}(x)}{\partial x}\right)^T$$

Step 13: $i \leftarrow i+1$, if $i \le I$ then go to Step 1.

Remarks:

- Step 3d rejects the data if the system is destabilized and it is better that not to use the data. The reason is as follows. The destabilized state causes a very large observed value. In a least-square method, an optimal parameter is strongly affected by the value. Therefore, the data in the case where a state is destabilized should be rejected and δ_2 indicates the threshold value.
- Step 3g judges whether a state converges to zero and δ_1 is the threshold value. It can be replaced by $t > T_e$ where T_e is a sufficient large time.

VII. Simulation

For simplisity, we assume that the covariance matrix is defined by

$$\Sigma = \begin{bmatrix} \sigma & 0 \\ 0 & \sigma \end{bmatrix}$$

and the range is described by $\sigma_{\min} \le \sigma \le \sigma_{\max}$.

We consider the following nonlinear system:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ x_1^3 - 3x_1^2x_2 - x_1 + x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ -1 \end{bmatrix} u \quad (33)$$

and the following cost function as $Q = I$, $R = 1$. Namely,

$$J(x_0, u(\cdot)) = \int_0^\infty \left(x^Tx + u^Tu\right) d\tau. \quad (34)$$

As Sec. III, REPI is applied to the system in the case that $g(x)$ is unknown. A standard deviation is $\sigma = 0.04$, weights of an augmented cost function are $Q_a = \mathrm{diag}(I_{2\times 2}, 1)$ and $R_a = 0.05$, a basis function is defined as follows, and other settings are shown in Tab.I:

$$\begin{aligned}\phi(x) = [&x_1^2, x_1x_2, x_2^2, x_1u, u^2, ux_2, x_1^4, x_1^3x_2, x_1^2x_2^2,\\ &x_1x_2^3, x_2^4, x_1^3u, x_2^3u, x_1^2u^2, x_2^2u^2, x_1u^3,\\ &x_2u^3, u^4, x_1x_2u^2, x_1x_2^2u, x_2x_1^2u]^T\end{aligned}$$

Accuracy of estimation

The estimated value v_k with respect to time is shown in Fig.1. The blue line indicates observed values, red and green ones indicate true values and estimated values respectively. This figure shows that REPI suppress the effects of the noises.

Convergence of a parameter

Transition of a parameter θ with respect to iteration is shown in Fig.2. Altough in the case of using original EPI, a parameter becomes so large and unstable, in the case of using REPI, it is converged.

Furthermore, we apply REPI to an augmented system in the cases of $\sigma = 0$ and $\sigma = 0.04$ are shown in Fig.3. The number of augments of a value function is 3, namely $[x_1, x_2, u]^T$. For illustration, Fig.3 shows projected value function to a hyperplane of $u = 0$. This figure shows REPI can efficiently suppress the effect of the noises to the value function.

Although we can consider a method which uses EPI with a filtered output, it is not practical. In the case that using one dimensional discrete time LPF with a cutoff frequency 0.5[Hz], the delay of signal destabilizes the system. On the other hand, in the case of a cutoff frequency 1.0[Hz], it cannot suppress the effects of noises to values well, as a result, the parameter is destabilized. These result indicate superiority of REPI.

Computation time

The mean computation time under a condition as in Tab.II is as follows:

A mean length of a sample-path is $\bar{M} = 241$, a mean computation time to obtain observed values is $\bar{T}_{compute} =$ 14.6[ms] and a mean time for estimation is $\bar{T}_{estimate} =$ 7.9[sec]. Hence, a mean time to update a parameter is

$$\left(\bar{M} \times T + \bar{T}_{compute} + \bar{T}_{estimate}\right) \times L = 2549[\text{sec}].$$

Systems with more dimension require the larger state-space to explore. Since REPI requires enough data in Ω, the number L is increasing as dimension n. Furthermore, the number of σ_{ij} is increasing as a square of n. Since accurate estimation requires to use some estimated standard deviations, the number of particles N is increasing as n. Hence, the more number n forces the more computation time $\bar{T}_{estimate}$.

But we can avoid this curse of dimensionality by using fixed Σ after it is estimated accurately. Then we only have to estimate $\mathbf{x}_k$, hence the dimension of the estimator is 3 independent on the dimension of the system (1). Furthermore, the length of a sample path $\bar{M}$ implies a time of convergence, hence it is independent on the dimension. $\bar{T}_{compute}$ is independent on the dimension of the original system and this computation time is only proportional to the length.

Accuracy of an approximated value function

The value function $V_{unknown}(x)$ in the case where $\sigma =$ 0.04 and $g(x)$ is unknown and the value function $V_{known}(x)$ in the case where $\sigma = 0$, and $g(x)$ is known are shown in Fig.4. It shows that $V_{unknown}(x)$ can approximate true value function $V_{known}(x)$. Furthermore, the cost at initial state $x_a(0) = [0.4, 0.4, 0.2]^T$ is $J_a(x_a(0), \mu_a^{(30)}(x)) = 0.840$, on the other hand, the true cost is $J(x_0, \mu^*(x)) = 0.833$. This figure and the result show that REPI can approximately solve the optimal control problem given in Sec. II-A.

VIII. Conclusion

We proposed an algorithm which can derive an approximate optimal control law for unknown dynamical systems even when the states are contaminated by an observation noise. The validity of this method was shown by a numerical simulation and we confirmed robustness and superiority of this method. The other discussion are as follows:

1) Although the observation noise was assumed to be additive as in (2), it can be also replaced that it is nonlinear affine if it is additive like

$$y = x + G(w), \quad G \text{ is nonlinear function}$$

where $G(w)$ and the distribution of w are known and mean value of $G(w)$ is equal to zero. The above simulation is a special case of $G(w) = w, w \sim \mathcal{N}(0, \Sigma)$.

2) $\phi(y)$ may be estimated by a filtered y, however, REPI has a feature to derive an approximate optimal control without state estimation.

3) Step 5a is generally described by

$$\hat{v}_j^{m,(i)} = C_k^{\mu^{(i)},m} + \gamma \hat{v}_{j+1}^{m,(i)}$$

where γ is a discount rate and our case means $\gamma = 1$. Although γ is usually determined by $0 < \gamma < 1$, note that the case of $\gamma = 1$ has no problem because the system is stabilized.

Although REPI can derive the approximate optimal parameter for a fixed cost function, if the cost function is changed, the same process should be repeated. Therefore, development of the algorithm which can directly learn optimal control laws without re-learning is a future work.

TABLE I

SETTINGS FOR THE SIMULATION

Notation	Numerical value
θ_0	$[0.1, 0, 0.1, 0, 1, 0_{1\times16}]^T$
I	30
L	20
M	2500
T	20[ms]
δ_1	5×10^{-5}
δ_2	5
Ω	$[-1.5, -1.5, -0.2]^T < x_0 < [1.5, 1.5, 0.2]^T$
$[\sigma_{\min}, \sigma_{\max}]$	$[0.035, 0.045]$

TABLE II

SETTINGS TO MEASURE TIME

CPU	Athron 2600+, 2.0[GHz]
Language	MATLAB ver 6.1.0.450(not compiled)
OS	Linux
Memory	1[GB]

REFERENCES

[1] D. Vrabie and F. L. Lewis, "Adaptive optimal control algorithm for continuous-time nonlinear systems based on policy iteration," in *2008 47th IEEE Conference on Decision and Control*. IEEE, December 2008, pp. 73–79.

[2] S. Ohtake and M. Yamakita, "Adaptive output optimal control algorithm for unknown system dynamics based on policy iteration," in *American Control Conference 2010*. IEEE, July 2010, pp. 1671–1676.

[3] W. M. Haddad and V. Chellaboina, *Nonlinear Dynamical Systems and Control: A Lyapunov-Based Approach*, illustrated edition ed. Princeton University Press, January 2008.

[4] S. Ishihara and M. Yamakita, "Constrained state estimation for nonlinear systems with non-gaussian noise," in *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*. IEEE, December 2009, pp. 1279–1284.

[5] N. Gupta and R. Hauser, "Kalman filtering with equality and inequality state constraints," *ArXiv e-prints*, September 2007.

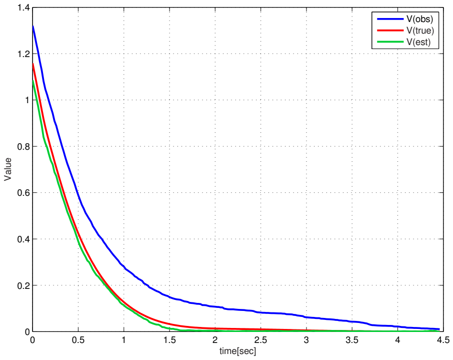

Fig. 1. Typical plots of observed, true and estimated values

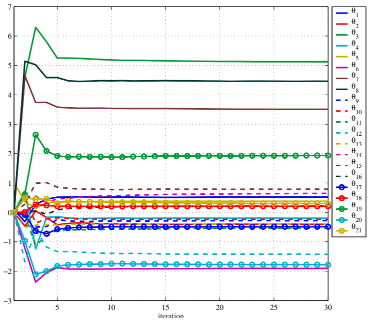

Fig. 2. Parameter θ converging (in the case of $\sigma = 0.04$)

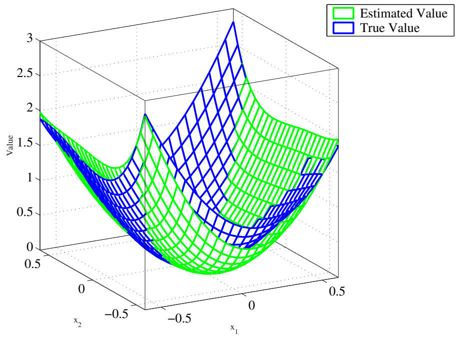

Fig. 3. Estimated value and true value function for an augmented system

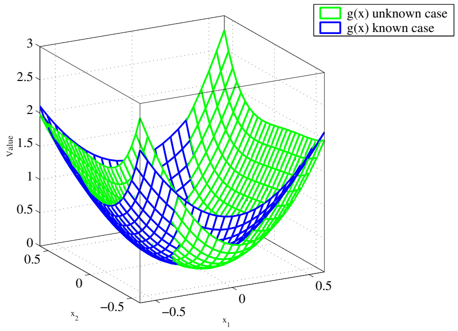

Fig. 4. The value function in the case of $g(x)$ unknown and $g(x)$ known