The Thirteenth International Symposium on Artificial Life and Robotics 2008(AROB 13th '08),
B-Con Plaza, Beppu, Oita, Japan, January 31-February 2, 2008

A Robust Reinforcement Learning Using Concept of Sliding Mode Control

M.Obayashi*, N.Nakahara*, T.Kuremoto*, and K.Kobayashi*

*Graduate School of Science and Engineering, Yamaguchi Univ.
,2-16-1 Tokiwadai, Ube, Yamaguchi,755-8611, Japan
(Tel : 81-836-85-9518, Fax : 81-836-85-9501)
(Email: {m.obayas,wu,koba}@yamaguchi-u.ac.jp, nakahara@nn.csse.yamaguchi-u.ac.jp)

Abstract: In this paper, we propose a new control method using a reinforcement learning (R.L.) with concept of sliding mode control (S.M.C.). Remarkable characteristics of the S.M.C. method are good robustness and stability for deviation of control conditions. On the other hand, R.L. may have applicability to complex systems that is difficult to model. However, applying reinforcement learning to a real system has serious problem, that is, many trials are required for learning. We employ actor-critic method, a kind of R.L., to unite with S.M.C. It is verified that the effectiveness of the proposed control method through the computer simulation for an inverted pendulum control without use of the inverted pendulum dynamics. Particularly, it is shown that the proposed method enables R.L. to learn in less trial than the only reinforcement learning method.

Keywords: robust, reinforcement learning, actor-critic, sliding mode control, inverted pendulum

I. INTRODUCTION

Recently, as to nonlinear system control method, research which unites conventional model-based control theory (sliding mode control, H_infinity control, etc) and model-free control theory (neuro control, fuzzy control, reinforcement learning control, etc) is very vigorous. But there are not so many researches combining reinforcement learning with model-based control theory.

One of their few reaches is robust reinforcement learning control [1] which unites reinforcement learning and H_infinity control. Based on the theory of H_infinity, Morimoto and Doya considered a differential game in which a disturbance agent tries to make the worst possible disturbance while a control agent tries to make the best control input. And they formulated the problem as finding a min-max of a value function that takes into account the amount of the reward and the norm of the disturbance and got the robust reinforcement learning method for both model-based and model-free. They applied the method to the cart-pole swing-up task, got a robust swing-up policy.

By the way, reinforcement learning is the method that acquires the control input sequences achieving control object based on the states of the system and reward through inputs of the try and error to the system. Unnecessariness of information of the system is very advantageous. On the other hand, too many trials and errors are needed to find the appropriate control input sequences to achieve the control object. However it comes to possibility to destroy the controlled system, therefore it should be able to achieve the object with less try and error as possible.

And sliding mode control is the control method to achieve the desirable dynamics of the system by restricting states of the system on the switching surface of the state space of the system. It is known as the method with superior robustness.

In this paper we consider the case of model-free. Under such conditions, we propose the model-free control method combining the concept of sliding mode control and reinforcement learning. It is verified that the proposed method makes possible to achieve the control object less try and error than the conventional reinforcement learning, and also has robust performance for the change of parameters of the system through the computer simulation of an inverted pendulum control problem.

II. REINFORCEMENT LEARNING

1. Actor-critic reinforcement learning

Actor-critic is one of the representative reinforcement learning methods. Actor-critic is consists of actor generating control signal and critic evaluating it. Actor-critic model is shown in Fig.1. Critic calculates predictive state value function $\hat{V}$. As a result TD-error δ is determined by reward received from environment. Here $r_t + \gamma \cdot \hat{V}_{t+1}$ in Eq. (7) is target of $\hat{V}_t$, and then δ is called Td-error. Output$\hat{V}$of critic, is adjusted in order that δ converges to zero. Actor generates control signal based on δ, that is, results of previous action, if TD-error is positive, choice of previous action is desirable, and in similar situation probability of selecting its action is enhanced. The opposite is also true [2].

III. SLIDING MODE CONTROL

Sliding control is as follows. First it restricts state of the system to switching surface set up in the state space. Then it generates sliding mode s (see in Eq. (4)) on the switching surface, and then stabilizes the state of the system to a specified point in the state space. The feature of sliding mode control is its good robustness.

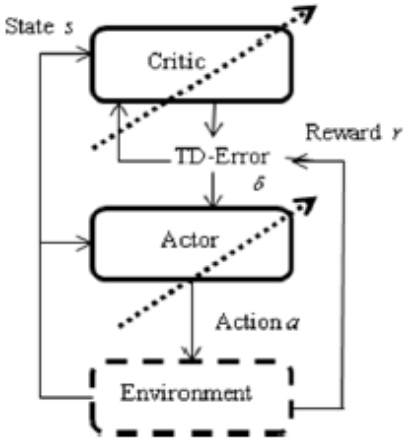
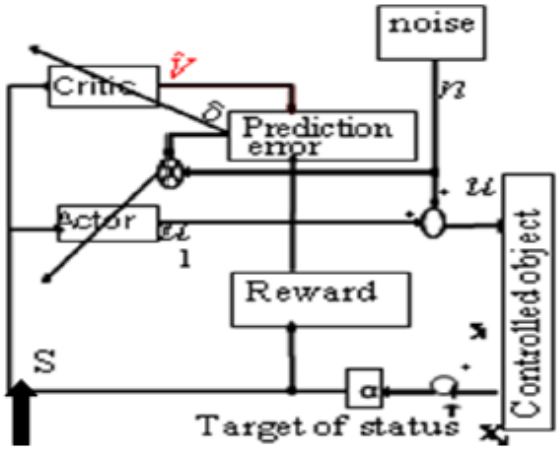


Fig.1 Actor-critic model



S: sliding variable (not equal to state of the system)

Fig.2 Structure of the propose d RRL system with SMC such that control is insusceptible to the model uncertainty and disturbance.

IV. REINFOREMENT LEARNING WITH C ONCEPT OF SLIDING MODE CONTROL

In this section, reinforcement learning method with concept of sliding mode control is explained. Target of this method is enhancing robustness which can not be obtained by conventional reinforcement.

1. Controlled system

This paper considers next nth order non-linear differential equation.

$$x^{(n)} = f(\mathbf{x}) + b(\mathbf{x})\,u\,, \qquad (1)$$

where $\mathbf{x} = [x, \dot{x}, \cdots, x^{(n-1)}]^T$ is the state vector of the system. In this paper, it is assumed that all states are observable. u is the control input, $f(\mathbf{x}), b(\mathbf{x})$ is unknown continuous function, but it is known that is inputted to the system as formulation of Eq. (1).

Object of the system : To decide control input u which leads states of the system to the target of the states of system $\mathbf{x}_d$. We define the error vector $\mathbf{e}$ as follows;

$$\mathbf{e} = [e, \dot{e}, \cdots, e^{(n-1)}]^T,$$
$$= [x - x_d, \dot{x} - \dot{x}_d, \cdots, x^{(n-1)} - x_d{}^{(n-1)}]^T, \qquad (2)$$

Sliding surface H is defined as follows;

$$H : \{\mathbf{e} \mid S(\mathbf{e}) = 0\}, \qquad (3)$$
$$S(\mathbf{e}) = \boldsymbol{\alpha}^T \mathbf{e}, \qquad (4)$$

where $\boldsymbol{\alpha} = [\alpha_0, \alpha_1, \cdots, \alpha_{n-1}]^T$, $\alpha_{n-1}p^{n-1} +$,

$\alpha_{n-2}p^{n-2} + \cdots + \alpha_0$ is strictly stable in Hurwitz,

p is Laplace transformation variable.

2. Constitution of actor-critic

A. Constitution of critic

Critic is constituted of Radial Basis Function (RBF) Network (see Fig.3). The input to critic is sliding variable s in sliding mode control and its output is predictive state value function, as follows,

$$\hat{V} = \sum_{i=1}^{J} \omega_{C_i} \cdot \exp\left\{\frac{-(s - c_{C_i})^2}{\sigma_{C_i}^2}\right\}, \qquad (5)$$

ω_{C_i} is the learning parameter. c_{C_i}, σ_{C_i} is ith average and standard deviation of RBF, respectively. Reward r is defined as Eq, (6), and given in order to make variable s restrict zero. Prediction error δ is defined as Eq.(7). γ is the damping coefficient. r_d is a positive constant.

$$r_t = \begin{cases} +r_d & (s_{t-1}^2 - s_t^2 > 0) \\ -r_d & (s_{t-1}^2 - s_t^2 < 0) \end{cases} \quad (r_d > 0)\,, \qquad (6)$$

$$\delta_t = r_t + \gamma \cdot \hat{V}_{t+1} - \hat{V}_t \quad (0 < \gamma \le 1)\,. \qquad (7)$$

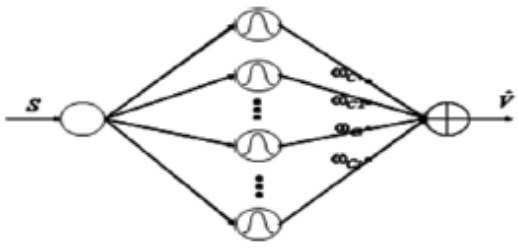


Fig.3 Structure of critic

B. Constitution of actor

Actor is also constituted of RBFN as same as critic. The input to actor is sliding variable s in sliding mode control and its output u_1 is a part of control signal. ω_{C_i} is a learning parameter, as follows,

$$u_1 = \sum_{i=1}^{N} \omega_{Ai} \cdot \exp\left\{\frac{-(s-c_{Ai})^2}{\sigma_{Ai}^2}\right\}, \quad (8)$$

3. Learning

A. Learning of critic's parameters

Learning of Criticis done by using common used Back Propagation method which makes prediction error goes to zero. Learning equation is as follows.

$$\Delta\omega_i^c = \eta_c \cdot \frac{\partial \delta_t^2}{\partial \omega_i^a}, \quad (i=1,\cdots,J). \quad (9)$$

Parameter $\omega_i^c, (i=1,\cdots,J)$ of critic RBFN is adjusted in order to $\delta = 0$.

B. Learning of actor's parameters

Parameter ω_i^a, $(i=1,\cdots,N)$ of actor of RBFN is adjusted by using output u_1 of actor and noise n_t.

$$\Delta\omega_i^a = \eta_a \cdot n_t \cdot \delta_t \cdot \frac{\partial u_1}{\partial \omega_i^a}, \quad (i=1,\cdots,N), \quad (10)$$

$\eta_a (>0)$ is learning coefficient. Eq. (1) means that $(-n_t \cdot \delta_t)$ is considered as error, ω_i^a is adjusted opposite to sign of $(-n_t \cdot \delta_t)$.

4. Noise

Noise n_t is to maintain diversity of search. It is bigger, absolute value of sliding variable s is bigger, and it is smaller, that of s is smaller. Noise is uniform random number, and is generated not to over the upper limit ($\overline{n}$).

$$n = z \cdot \overline{n} \cdot \exp\left(-\beta \cdot \frac{1}{s^2}\right) \quad (11)$$

z is uniform random number of range $[-1,1]$. $\overline{n}$ is upper limit of the perturbation signal for searching. β is a parameter for adjusting.

V. COMPUTER SIMULATION

1. Controlled system

To verify effectiveness of the proposed method, we carried out the control simulation using the inverted pendulum with dynamics described to Eq. (12).(see Fig.4)

$$mg\ddot{\theta} = mgl\sin\theta - \mu v\dot{\theta} + T \quad (12)$$

Parameters in Eq. (12) are described to Table 1. Simulation is carried out using MatX [3].

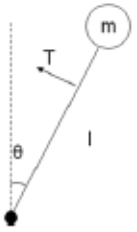


Fig.4 An inverted pendulum

Table.1 Parameters of the system

θ : **joint angle**
$m = 1.0\,[\mathrm{kg}]$
$l = 1.0\,[\mathrm{m}]$: **lengthof thependulum**
$g = 9.8\,[\mathrm{m/s^2}]$
$\mu_V = 0.01$: **coefficient of friction**
T : **input torque**
Observation vector : $\mathbf{x} = [\theta, \dot{\theta}]^{\mathrm{T}}$

2. Simulation condition

One trial means that control starts at $(\theta_0, \dot{\theta}_0) = (\pi/18[rad], 0[rad/\sec])$ and continues for 20[sec], and sampling time is 0.02[sec].The trial ends if $|\theta| \geq \pi/6$ or controlling time is over 20[sec]. We set upper limit for output u_1 of actor. Trial success means that θ is in range $[-\pi/360, \pi/360]$ for last 10[sec].

3. Parameters of the proposed method

Number of RBFN, function approximator consisting of actor-critic, is 15, and average C_C and standard deviation σ_A of actor are set as follows,

$C_C = [-15,-12,-9,-6,-3,-2,-1,0,1,2,3,6,9,12,15]$,

$C_A = [-15,-12,-9,-6,-3,-2,-1,0,1,2,3,6,9,12,15]$,

$\sigma_C = [3, 3, 3, 3, 2, 1, 0.5, 0.3, 0.5, 1, 2, 3, 3, 3, 3]$,

$\sigma_A = [3, 3, 3, 3, 2, 1, 0.5, 0.3, 0.5, 1, 2, 3, 3, 3, 3]$,

$\boldsymbol{\alpha} = [15, 1]^T$ $\omega_{C_i}, \omega_{A_i}$ is initially set to small rando m number and not equal to zero. Upper limit of t he control signal u_1 is set to 10[N・m] other para meters are set to $\gamma = 0.9$, $\overline{n} = 5.0$, $r_d = 10.0$, $\eta_a = 0.1$, $\eta_c = 0.1$, $\beta = 10.0$.

These are decided by trial and error.

4. Parameters of (the conventional) actor-critic m ethod

The predictive state value function, a part of control signal, and the reward function using in the

conventional actor-critic method are as follows, respectively,

$$\hat{V} = \sum_{i=1}^{J} \omega_{Ci} \exp\left(-\frac{(\theta - c_{Ci_\theta})^2}{\sigma_{Ci_\theta}^2} - \frac{(\dot{\theta} - c_{Ci_{\dot{\theta}}})^2}{\sigma_{Ci_{\dot{\theta}}}^2} \right), \quad (13)$$

$$u_1 = \sum_{i=1}^{J} \omega_{Ai} \exp\left(-\frac{(\theta - c_{Ai_\theta})^2}{\sigma_{Ai_\theta}^2} - \frac{(\dot{\theta} - c_{Ai_{\dot{\theta}}})^2}{\sigma_{Ai_{\dot{\theta}}}^2} \right), \quad (14)$$

$$r_t = 10 \exp\left(-\frac{(\theta_t)^2}{2(\alpha_{R1})^2} - \frac{(\dot{\theta}_t)^2}{2(\alpha_{R2})^2} \right) - 5.0 . \quad (15)$$

Where,

$C_{Ai_\theta}, C_{Ai_{\dot{\theta}}}, \sigma_{Cii_\theta}, \sigma_{Cii_{\dot{\theta}}}$ in critic are same as that of the proposed, average and standard deviation are 1/5 of that of actor of proposed. Others are set to $\alpha_{R1} = 0.2, \alpha_{R2} = 20$, $\eta_a = 0.1$, $\eta_c = 0.1$, $\beta = 0.1$ by trial and error.

5. Parameters of PID control method

Control signal u in PID is defined as follows,

$$u(t) = -K_p e(t) - K_d \dot{e}(t) - K_i \int^t e(t)\, dt , \quad (16)$$

where parameters decided by trial and error are $K_p = 50, K_d = 30, K_i = 0.5$.

6. Simulation results

Table 2 shows success rate of learning. In Table 2 A/B means number of success A per number of trial B.

Table says that learning of the proposed method is done stably than that of actor-critic. But both rates are not good. That may be 1) without learning average and standard deviation of RBF, 2) without enough adjusting parameters, 3) without enough number of RBF.

Control results of θ shown in Fig.5. Results of the proposed method is oscillatory on behalf of first 5 [sec], but steady state error is smallest of the 3.

Result of robust performance for change of m is shown in Table 3. The proposed method is best for changing m smaller, but for changing m bigger the proposed

Table 2 Success rate of learning

	Proposed	actor-critic	PID
A / B	170 / 300	2 / 3000	—

Table 3 Robust performance for change of m

	Proposed	Actor-critic	PID
m-max [kg]	2.202	1.668	3.439
m-min [kg]	0.008	0.021	0.022

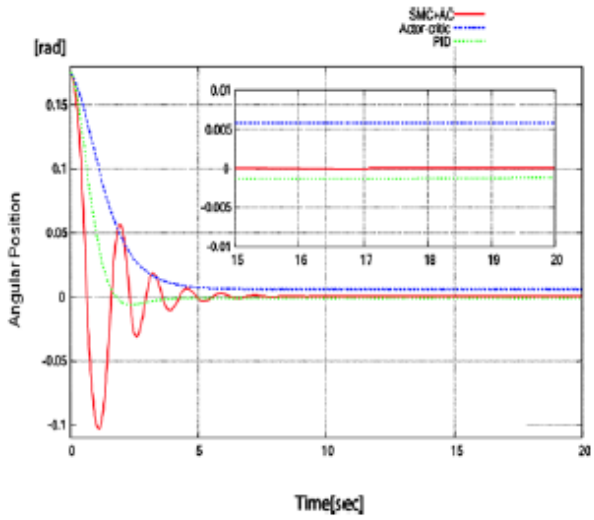


Fig. 5 Result of learning control for θ

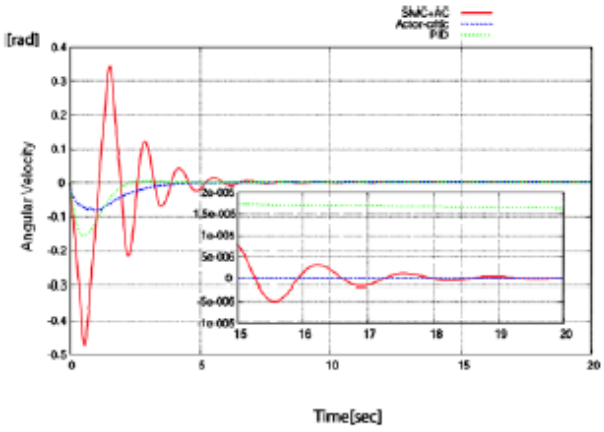


Fig. 6 Result of learning control for $\dot{\theta}$

method is better than actor-critic, but worse than PID.

VI. CONCLUSION

A robust reinforcement learning method using concept of sliding mode control was proposed. Through the inverted pendulum control simulation, it was verified that the proposed method has good robustness, but not good control performance especially because of its oscillatory.

In future work, it is necessary to improve control performance and to clarify limit of robust stability of the proposed method.

REFERENCES

[1] J. Morimoto, K. Doya (2005),Robust Reinforcement Learning. *Neural Computation* 17, 335-359

[2] R.S. Sutton, A.G. Barto (1998), Reinforcement Learning An Introduction. *The MIT Press.*

[3] M. Koga (2000), Numerical computation by MATX, *Tokyo Denki University Press.*