Kernel Least Mean Square Algorithm in Control of Nonlinear Systems

Zinat Mazloomi*, Heydar Tusian Shandiz**, and Hossein Faramarzpour***

*Faculty of Electerical and Robotic Engineering, Shahrood University of Technology, Shahrood 36199-95161
z_mazloomi@yahoo.com

**Faculty of Electerical and Robotic Engineering, Shahrood University of Technology, Shahrood 36199-95161
htshandiz@shahroodut.ac.ir

***Department of Mechanical Engineering, Faculty of Engineering, Tarbiat Modares University, Tehran 14115-143,
air_dr2010@yahoo.com

***Abstract**: Although some research has been presented about the application of Kernel Least Mean Square (KLMS) algorithm in the estimation and approximation of functions, this algorithm wasn't applied to the control of nonlinear systems. In this paper, an efficient and novel adaptive Control strategy based on Kernel Least Mean Square is introduced to realize the control of a nonlinear aircraft system. Actually the KLMS algorithm is a growing radial basis function (GRBF) network, when Kernel function is a Gaussian function. In this research, based on Lyapunov theory, KLMS is used as an online method for tuning the kernel size to control nonlinear systems. This technique certifies the stability and provides an acceptable accuracy. Finally, we utilize this algorithm to control a nonlinear fighter aircraft by using a dynamic model of the F-18 aircraft.*

Keywords: Kernel Least Mean Square, Lyapunov theory, Online learning;Tracking a maneuver.

1. Introduction

When an aircraft is moving in rectilinear without any change in velocity or direction of movement, it is impossible to make a maneuver just by tuning control surfaces. On the other hand, because of complexities in the aircraft dynamic and nonlinear properties due to the high aerobatic maneuvers, the control of aircraft maneuver is difficult. For example in Herbest maneuver although a conventional controller may certify stability of aircraft maneuver, it can't provide a good tracking performance. So it is needed to design an appropriate controller to track the desired maneuver and ensure the stability of the system. Since neural networks are powerful tools to approximate mapping between input and output of nonlinear systems, they can be widely used in the area of modeling, identification and control of nonlinear systems. Recently, various neural networks with different structures have been implemented in the control of nonlinear aircraft. These researches show that Radial Basis Function Networks (RBFN) have an acceptable performance in the control of nonlinear systems. The simplicity of implementing is an advantage of RBFN. In one research, RBFN was used to identify the dynamic of F-16 aircraft and control it [1]. In another research an online controller based on fully tuned RBFN was proposed to control aircraft maneuver [2]. In this paper, for the first time famed KLMS algorithm [3, 4] is used as an online method to control nonlinear systems like fighter aircraft. KLMS algorithm has improved the basic idea of least mean square in the viewpoint of machine learning. If the kernel function is in the Gaussian form, this algorithm would be a growing radial basis function network which doesn't need to tune the weight vector and the centers of the Gaussian function. Because of the simplicity in the structure of this algorithm, it can be used in control applications. Current paper uses an adaptive method based on the Lyapunov stability theory for tuning kernel size to control an aircraft maneuver with nonlinear dynamic.

2. The kernel least mean square algorithm

2.1 Least Mean Square

The least mean square technique, firstly, was proposed by Widrow and Hoff and, due to its simplicity, is widely used in the signal processing [5]. This algorithm is the most famous adaptive technique based on the method of steepest descent.

If the actual, estimated output, and input of given system whose dynamic is under estimation are considered $y(n)$, $\hat{y}(n)$, and $x(n)$, respectively, the cost function and weight update equation can be written as:

$$\hat{y}(n) = w^T x(n). \tag{1}$$

$$f_c = (y(n) - \hat{y}(n))^2 = e^2(n). \tag{2}$$

$$w(n+1) = w(n) - \eta \frac{\partial f_c}{\partial w} = w(n) + 2\eta\, e(n)\, x(n). \tag{3}$$

Where w is the weight vector; f_c is the cost function and η is the convergence rate. Normalized least mean square algorithm can be used to certify stability and improve convergence speed [6].

2.2 Kernel Forms(Tricks)

Already some research has been reported about kernel forms like kernel support vector machine [7], kernel principal component analysis [8], regularization network [9], KLMS [3, 4] in application of machine learning and signal processing. Among these methods, the KLMS is a proper and simple approach for use in online applications. The fundamental viewpoint of kernel forms is to map input data x_i to a high dimensional feature space of vectors $\varphi(x_i)$. In this space inner products can be defined by using a kernel function.

$$\kappa(x_i, x_j) = \langle \varphi(x_i), \varphi(x_j) \rangle. \quad (4)$$

The kernel function can be in a Gaussian or polynomial form. In this paper the Gaussian form is used.

$$\kappa(x, y) = \exp(-\frac{\|x - y\|^2}{2\sigma^2}). \quad (5)$$

σ is the kernel size which has an effective role in the performance of the algorithm. The polynomial kernel function is defined as:

$$\kappa(x, y) = (1 + xy)^n. \quad (6)$$

n represents the order of kernel function.

3. Nonlinear Controller Design Using Kernel Least Mean Square Algorithm

3.1 Control Approach Based Kernel Least Mean Square

To use KLMS in control, based on least mean square algorithm, weight update equation would be:

$$\begin{aligned}\Omega(n+1) &= \Omega(n) + 2\eta e(n)\frac{\partial \hat{y}}{\partial \Omega} \\ &= \Omega(n) + 2\eta e(n)\frac{\partial \hat{y}}{\partial u}\frac{\partial u}{\partial \Omega}.\end{aligned} \quad (7)$$

Where $\hat{y}$ is the actual output of system and supposed to be controlled. η is the learning rate and $e(n)$ is tracking error between actual output and desired output y_d . Since the control input or output of KLMS is in the form of $u(n) = \langle \Omega(n), \phi(y_d(n)) \rangle$, the weight update equation in a high dimensional space is written as:

$$\Omega(n+1) = \Omega(n) + 2\eta e(n)\frac{\partial \hat{y}}{\partial u}\phi(y_d(n)). \quad (8)$$

Computation of current equation is impractical. The non-recursive type of this weight vector when initial condition, $\Omega(0)$, is considered zero is obtained in the form :

$$\Omega(n) = 2\eta\sum_{i=0}^{n-1} e(i)\frac{\partial \hat{y}(i)}{\partial u(i)}\phi(y_d(i)). \quad (9)$$

while $\Omega(n)$ is in a high dimensional space, the output of network is defined in the following form.

$$\begin{aligned}u(n) &= \langle \Omega(n), \phi(y_d(n)) \rangle \\ &= \eta\sum_{i=0}^{n-1} e(i)\frac{\partial \hat{y}}{\partial u}\langle \Phi(y_d(i)), \Phi(y_d(n)) \rangle \\ &= \eta\sum_{i=0}^{n-1} e(i)\frac{\partial \hat{y}}{\partial u}\kappa(y_d(i), y_d(n)).\end{aligned} \quad (10)$$

As mentioned, $u(n)$ is the input control of system and η is the step size of the algorithm. While Gaussian form of kernel is used, the output of controller is rewritten in the form :

$$u(n) = \eta\sum_{i=0}^{n-1} e(i)\frac{\partial \hat{y}}{\partial u}\exp(-\frac{\|y_d(i) - y_d(n)\|^2}{2\sigma^2}). \quad (11)$$

It's necessary to know that σ is usually considered as a constant value. It can be updated to gain a better performance.

3.2 Adaptive Rule to Update Kernel Size

The kernel size can be updated instead of a fix value to improve tracking accuracy. The proposed technique is based on online feedback error learning.

The system dynamic to be controlled is defined by a nonlinear continuous equation.

$$\dot{x} = f(x, u). \quad (12)$$

Without loss of generality, some assumptions can be considered as:

1. $f(x, u)$, is smooth and $f(0,0) = 0$.
2. The total number of state is k. The number of control inputs is r, which is equal to the number of states controlled. Simultaneously, the rest of states converge to the equilibrium points.

Under these assumtions, desired control inputs can be determined as:

$$\begin{aligned}u_d(t) &= \eta\sum_{i=0}^{n-1} e(i)\frac{\partial \hat{x}}{\partial u}\exp(-\frac{\|x_d(i) - x_d(n)\|^2}{2\sigma_*^2}) \\ &= \eta\sum_{i=0}^{n-1} e(i)\frac{\partial \hat{x}}{\partial u}\Phi_*.\end{aligned} \quad (13)$$

Where the optimal kernel size is $\sigma_*, e(i) = x(i) - x_d(i)$ is the tracking error and Φ_* is the optimal kernel function. The tracking error dynamic of system is expressed as:

$$\dot{e} = f(x,u) - f(x_d, u_d). \tag{14}$$

using Taylor series expansion and ignoring higher order terms, we have:

$$\dot{e} = A(t)e + B(t)(u - u_d) \tag{15}$$

Where *A(t)* and *B(t)* are in the form of:

$$A(t) = \frac{\partial f(x,u)}{\partial x^T}|_{x_d, u_d}, B(t) = \frac{\partial f(x,u)}{\partial u^T}|_{x_d, u_d}. \tag{16}$$

In the proposed approach a conventional controller is utilized to certify the stability of the overall system along tracking desired maneuver. The conventional controller can be a PID, LQR which is parallel to the main controller. In this research a proportional controller, which provides a stable closed loop system is applied.

The total control input is given by:

$$u = u_{klms} + K_p(t)e. \tag{17}$$

Assume that the output of KLMS (u_{klms}) and weight vector can be written as :

$$\hat{u}_K = w^T \hat{\Phi}. \tag{18}$$

$$W = [e(1) \;\; e(2) ... e(n-1)]^T. \tag{19}$$

Here $\hat{\Phi}$ is the estimated kernel function. So, estimated error would be :

$$\dot{\tilde{\Phi}} = \dot{\Phi}^* - \dot{\hat{\Phi}} \tag{20}$$

Finally, the error dynamics of system is obtained as :

$$C(t) = A(t) + B(t)K_p(t). \tag{21}$$

$$\dot{e} = C(t)e - B(t)w^T \hat{\Phi}. \tag{22}$$

The main condition to make *C(t)* stable is depended on designed $K_p(t)$.

An option for Lyapunov function is proposed as :

$$V = 1/2 e^T M(t) e + 1/2 \tilde{\Phi}^T \Gamma \tilde{\Phi} \tag{23}$$

Where $M(t)$, and Γ are symmetric, positive definite matrix and fixed, positive definite matrix, respectively. By the equations mentioned above, the derivative of Lyapunov function is obtained as:

$$\dot{V} = -e^T \mathrm{H}(t) e - \tilde{\Phi}^T w B(t)^T M(t) e + \tilde{\Phi}^T \Gamma \dot{\tilde{\Phi}} \tag{24}$$

Where $\mathrm{H}(t)$ represents:

$$H(t) = -1/2(C(t)^T M(t) + M(t)C(t) + \dot{M}(t)) \tag{25}$$

Finally, the derivative of Lyapunov function will be achieved as:

$$\dot{V} = -e^T \mathrm{H}(t) e + \tilde{\Phi}^T(-w B(t)^T M(t) e + \Gamma \dot{\tilde{\Phi}}) \tag{26}$$

It's obvious that $\dot{V}$ is negative when

$$\dot{\tilde{\Phi}} = \Gamma^{-1} w \; B(t)^T \; M(t) e \tag{27}$$

While $\dot{\Phi}^* = 0$, the tuning rule for kernel function is :

$$\dot{\hat{\Phi}} = -\Gamma^{-1} w \; B(t)^T \; M(t) e \tag{28}$$

The discrete form of this equation is obtained as follows:

$$\begin{aligned} \nabla\hat{\Phi}(n) &= \hat{\Phi}(n+1) - \hat{\Phi}(n) \\ &= -\gamma_1 \Gamma^{-1} w \; B(n)^T \; M(n) e(n). \end{aligned} \tag{29}$$

Here γ_1 is a constant value which represents learning rate.

Since $\hat{\Phi}$, is a function of $\hat{\sigma}$, the tuning rule for updating kernel size is achieved as :

$$\begin{aligned} \hat{\sigma}(n+1) &= \hat{\sigma}(n) + \gamma_2 \frac{\partial \hat{\Phi}}{\partial \hat{\sigma}^T} \nabla\hat{\Phi}(n) \\ &= \hat{\sigma}(n) - \gamma_1 \gamma_2 \frac{\partial u_k}{\partial \hat{\sigma}^T} B(n)^T \; M(n) e(n) \end{aligned} \tag{30}$$

4. Illusrative application

In this section the performance of the proposed technique is investigated. The 6DOF nonlinear model of an F-18 as a fighter aircraft is used to demonstrate the performance of designed controller. Simulation is implemented under MATLAB software.

4.1 Aircraft Model

Before illustrating tracking performance, a short description of nonlinear model of aircraft motions is represented. The detail of model is available in [10, 11].

Force equations:

$$\dot{u} = rv - qw - g\sin(\theta) + (\bar{q}SC_{x,t} + T)/m \qquad (31)$$

$$\dot{v} = pw - ru + g\cos(\theta)\sin(\varphi) + \bar{q}SC_{y,t}/m \qquad (32)$$

$$\dot{w} = q\ u - p\ v + g\cos(\theta)\cos(\varphi) + \bar{q}SC_{z,t}/m \qquad (33)$$

Moment equations:

$$\dot{p} = (c_1 p + c_1 r + c_4 he)q + \bar{q}Sb\,(c_3 C_{l,t} + c_4 C_{n,t}). \qquad (34)$$

$$\dot{q} = (c_5 p - c_7 he)r + c_6(r^2 - p^2) + \bar{q}S\bar{c}\,c_7 C_{m,t} \qquad (35)$$

$$\dot{r} = (c_8 p - c_2 r + c_9 he)q + \bar{q}Sb\,(c_4 C_{l,t} + c_9 C_{n,t}) \qquad (36)$$

In the above equations, $c_1 - c_9$ are defined as

$$\begin{aligned}
c_1 &= (I_z(I_y - I_z) - I^2_{xz})/(I_x I_z - I^2_{xz}) \\
c_2 &= (I_{xz}(I_x - I_y + I_z))/(I_x I_z - I^2_{xz}) \\
c_3 &= I_z/(I_x I_z - I^2_{xz}) \\
c_4 &= I_{xz}/(I_x I_z - I^2_{xz}) \\
c_5 &= (I_z - I_x)/I_y \\
c_6 &= I_{xz}/I_y \\
c_7 &= 1/I_y \\
c_8 &= (I_x(I_x - I_y) - I^2_{xz})/(I_x I_z - I^2_{xz}) \\
c_9 &= I_x/(I_x I_z - I^2_{xz})
\end{aligned} \qquad (37)$$

4.2 Simulation Results

As mentioned, the presented technique is utilized to control a maneuvering fighter aircraft with 6DOF nonlinear dynamic model named F-18. In this research the variables affect the maneuver style are α (angle of attack) and $\dot{\mu}$ (stability axis roll rate). This maneuver is similar to the maneuver studied in[2]. The equations of these variables are:

$$\alpha = \tan^{-1}(w/u). \qquad (38)$$

$$\dot{\mu} = p\cos(\alpha) + r\sin(\alpha) \qquad (39)$$

The maneuver starts at the initial condition of v_t=400ft/sec, α= 2.837deg, h=1000m/sec. A pitch command is implemented to increase α from it's initial value to 15 deg. $\dot{\mu}$ is controlled to maintain α in desired value and repels crash of aircraft when tries to turn 180 deg about stability axis in executing maneuver. So, the aircraft stays stable in this situation. On the other hand, the side slip angle (β) has to lie in less than 1 deg. The control surfaces are δ_e and δ_a whose ranges are:

$$\delta_e = [-25\deg, +25\deg], \delta_a = [-21.5\deg, 21.5\deg].$$

As the results demonstrate, α and $\dot{\mu}$ track the desired inputs and β doesn't deviate from admissible range.

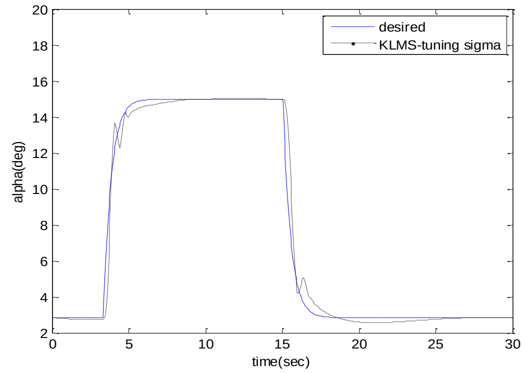

Fig. 1: Angle of attack(deg)tracking

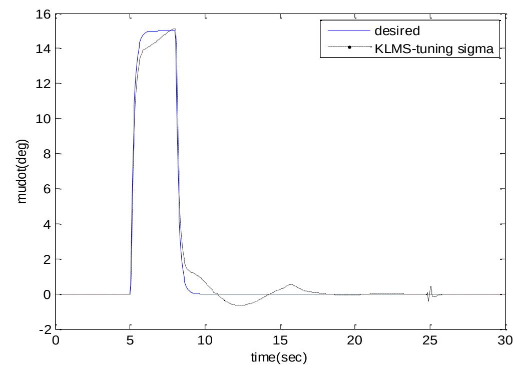

Fig. 2: Stability Axis roll rate(deg)

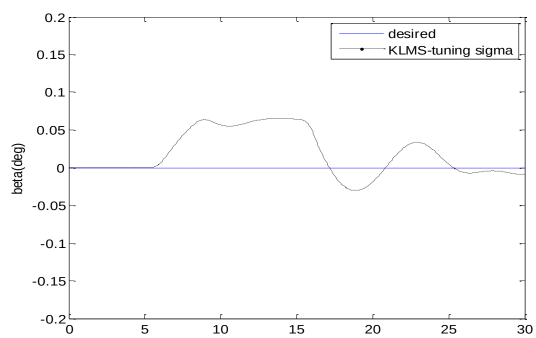

Fig. 3: Sideslip angle(deg)

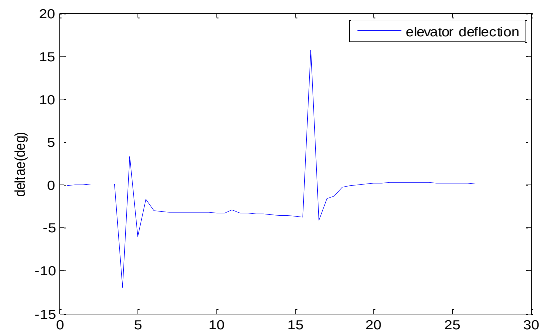

Fig. 4: Elevator deflection(deg/s)

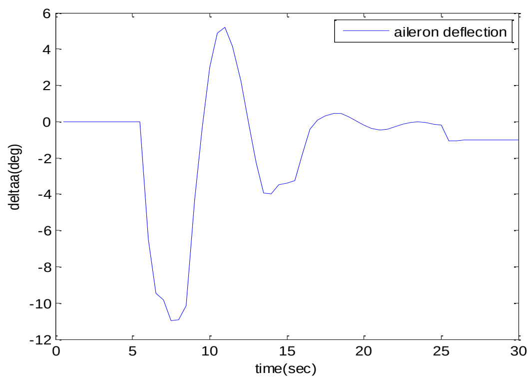

Fig. 5: Aileron deflection(deg/s)

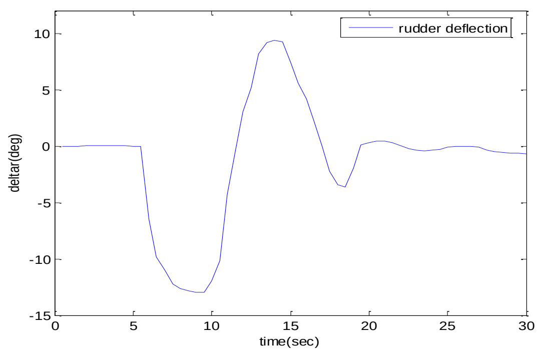

Fig. 6: Rudder deflection(deg/s)

5. Conclusion

In this paper the KLMS algorithm was applied for the first time to control a nonlinear system like a 6DOF model of an aircraft. In the proposed technique, based on adaptive rule, the kernel size was updated to reduce tracking error and insure stability of the system. The simulation results represent the capability of the proposed control method.

6. Reference

[1] F. Ahmed Ziad, P. A. Ioannou and M. M. Polycarpou, "Identification and Control of Aircraft Dynamic Using Radial Basis Function Networks," Second IEEE Conference on Control Application, September,pp. 13-16,1993.

[2] Y. Li, N. Sundararajan, and P. Saratchandran, "Neuro-controller design for nonlinear aircraft maneuver using fully tuned RBF networks,"Automatica, vol. 37, 2001.

[3] P. P.Pokharel , W.Liu, and J.C.Principe, "Kernel least mean square algorithm with constrained growth,".signal processing, pp. 257-258,2009.

[4] W. Liu., P. P. Pokharel, andJ. C. Principe,"The Kernel Least-Mean-Square Algorithm,"IEEE Transaction On Signal Processing, 2008.

[5] B.Widrow, S.D. Stearns, "Adaptive Signal Processing," Prentice-Hall, Englewood Cliffs, NJ, 1985.

[6] H.Modaghegh, H.Khosravi R and S.Ahoon Manesh, and H.Sadoghi Yazdi,"A new modeling algorithm-Normalized Kernel Least Mean Square". IEEE,2009.

[7] V. Vapnik,"The Nature of Statistical Learning Theory,"Springer, New York, 1995.

[8] B.Scholkopf, A.Smola and K. Muller,"Nonlinear component analysis as a kernel eigenvalue problem,"Neural Comput, pp. 1299-1319,1998.

[9] F. Girosi, M. Jones, and T. Poggio, "Regularization theory and neural networks architectures," Neural Comput. pp. 219-269, 1995.

[10] Iliff, Kenneh W., Kon-sheng charles Wag, "Flight-determined subsonic longitudinal stability and control derivatives of the F-18 HARV with thrust vectoring," NASA/TP-97-206539, 1997.

[11] B. l.stevens, and F. l.lewis,"Aircarft control andsimulation,wiley,", 2003.