입출력 데이터 기반 Q-학습과 LMI를 이용한 선형 이산 시간 시스템의 모델-프리 H_∞ 제어기 설계

논 문
58-7-23

Model-free H_∞ Control of Linear Discrete-time Systems using Q-learning and LMI Based on I/O Data

김 진 훈† · F. L. Lewis*
(Jin-Hoon Kim · Frank. L. Lewis)

Abstract - In this paper, we consider the design of H_∞ control of linear discrete-time systems having no mathematical model. The basic approach is to use Q-learning which is a reinforcement learning method based on actor-critic structure. The model-free control design is to use not the mathematical model of the system but the informations on states and inputs. As a result, the derived iterative algorithm is expressed as linear matrix inequalities(LMI) of measured data from system states and inputs. It is shown that, for a sufficiently rich enough disturbance, this algorithm converges to the standard H_∞ control solution obtained using the exact system model. A simple numerical example is given to show the usefulness of our result on practical application.

Key Words : Model-free H_∞ control, Linear discrete-time system, I/O data, Q-learning, LMI.

1. 서 론

제어 시스템에 외란이 가하여질 때, 외란이 시스템에 미치는 영향을 최대한 감쇠하도록 제어기를 설계하는 것이 바람직하다. 그리고 이러한 외란 감쇠(disturbance attenuation)를 효과적으로 할 수 있는 대표적인 제어 방법이 H_∞ 제어이다[9][11]. H_∞ 제어는 원래 외란이 존재하는 시스템의 안정도를 확보하기위한 기법으로 주파수 영역에서 시작되었지만[11], 등가적 변환에 의해 지금은 시간영역에서 외란으로부터 측정출력까지의 L_2 이득을 최소화하는 제어기 설계 영역에서 매우 활발히 연구되어지고 있다. 시간영역에서의 H_∞ 제어기 설계는 선형행렬부등식(linear matrix inequality) 형태를 가지며 이는 잘 알려진 소프트웨어(LMItoolbox in Matlab)를 이용하면 쉽게 이를 구할 수 있다. 그러나 일반적으로 H_∞ 제어기를 설계하고자하면 수학적으로 기술된 시스템 모델이 필요하고[9,. 또한 시스템의 수학적 모델을 얻는 것은 매우 어렵거나 경제적으로 많은 부담을 주기에 수학적 모델이 없는 시스템에 대한 제어기 설계 기법이 절실히 요구되어지고 있다[2].

수학적 모델이 없는 시스템의 제어기 설계 방법은 먼저 시스템의 입출력을 이용한 시스템 모델링과정을 거친 후 얻어진 모델을 바탕으로 제어기를 설계하는 것도 하나의 방법이다. 그러나 이는 모델링 과정을 거치므로 이는 시스템 모델링 분야의 문제이기에 이를 제외하고 모델링 과정을 거치지 않고 곧 바로 제어기를 설계하는 것이 본 연구의 목적이다. 즉 시스템 모델에 주어지는 행렬들을 이용하지 않고 시스템으로부터 측정되어지는 상태변수나 입력변수를 토대로 시스템 모델링 과정을 거치지 않는 외란으로부터 측정출력까지의 L_2이득을 최소화하는 H_∞ 제어기를 설계하는 것이다. 이를 위하여 기본적으로 채택한 것이 적응동적계획법(Adpative dynamic programming)이다.

적응동적계획법은 초기에 실시간으로 최적제어 문제를 풀기위한 수단으로 소개되었고[5][6][7], 이는 적응감정구조(adaptive critic structure)를 이용한 강화학습(reinforcement learning)과 동적계획(dynamic programming)의 결합체로써 최근에는 적응동적계획을 이용한 피드백 제어에 많은 결과가 있다[10][12][13]. 또한 이산시간시스템에 대하여 적응동적계획의 일종이면서 시스템의 동특성(dynamics)을 필요로 하지 않는 실시간으로 최적제어문제를 해결하기위한 Q-학습(Q-learning)이 발표되었다[8]. 이는 최적제어문제를 해결함에 있어 필요한 강화학습(reinforcement learning)을 구현하기위하여 계속적으로 근사화(approximation)과정을 반복하는 정책반복(policy iteration)이다. 여기서 정책반복은 초기에 안정화 제어를 선택하고, 미리 정의된 비용함수(cost function)에 따라 선택된 제어에 따른 비용 값(cost value)를 계산한 후, 이 비용 값을 최소로 하는 새로운 제어를 갱신하는 구조이다. 최근에는 온라인(online)으로 시스템의 동특성을 필요로 하지 않는 이차최적제어(quadratic optimal control) 문제를 해결하는 데 이용되었으며[3], 주어진 L_2이득 값을 만족하는 제어기를 제로썸(zero-sum) 게임 이론을 이용하여 구하는데 이용되었다[2]. 그러나 최근의 결과[2]를 포함한 기존의 대부분의 결과들은 행렬-벡터 방정식의 해를 얻는 과정에 유사역변환(pseudo-inverse)과 최소자승법(least square method)의 근사해(approximate solution)를 이용하기

† 교신저자, 정회원 : 충북대 전자정보대 전자전공 교수
E-mail : jinhkim@chungbuk.ac.kr
* 비 회 원 : 미국 UTA 전기공학과 교수
접수일자 : 2009년 4월 7일
최종완료 : 2009년 5월 18일

때문에 진정한 해를 구하는 데 한계가 있고 수렴성이 매우 떨어지는 단점이 있다[1][2].

본 논문에서는 수학적 모델링이 되어있지 않은 선형이산 시스템에 대하여 입력과 상태변수의 데이터만을 이용한 Q-학습과 LMI를 이용하여 외란으로부터 측정출력까지의 L_2이득을 최소화하는 H_∞ 제어기를 설계한다. 또한 제시된 제어기 설계 방법은 충분한 외란이 주어지는 경우, 정확한 모델에 기초한 표준 H_∞ 제어기에 수렴함을 보여준다. 끝으로 수치예제를 통하여 제시되는 제어기 설계 방법의 유용성을 보여준다. 이 논문에서 별도로 정의되지 않는 정의는 표준적인 것이다. R^n은 $n\times 1$ 실수 벡터, $R^{m\times n}$은 $m\times n$ 실수 행렬을 각각 나타내고, 대칭행렬 X에 대한 $X>0$은 행렬 X가 양확정(positive definite)임을 나타낸다. 또한 행렬속의 원소 (★)는 대칭행렬 요소, 즉 $\begin{bmatrix} X & Y \\ \star & Z \end{bmatrix}=\begin{bmatrix} X & Y \\ Y^T & Z \end{bmatrix}$이다. 끝으로 $\|\cdot\|_2$는 L_2 노옴, 즉 $\|x_k\|_2^2=\sum_{k=1}^{N}(x_k^T x_k)$을 의미한다.

2. H_∞ 제어와 예비결과

여기에서는 선형이산시간 시스템에 대한 H_∞ 제어의 정의와 다음의 주요결과의 유도에 필요한 약간의 예비결과를 제시한다. 먼저 다음의 이산시간 선형시스템을 생각하자.

$$\begin{cases} x_{k+1}=Ax_k+B_1u_k+E_1w_k \\ z_k=Cx_k+B_2u_k+E_2w_k,\ \ x_0=0 \end{cases} \tag{1}$$

여기서 $x\in R^n$은 상태, $u\in R^m$은 제어, $w\in R^d$는 L_2노옴이 제한된 외란, $z\in R^p$는 측정출력이고 행렬 A,B_1,B_2,C,E_1,E_2는 적당한 차원을 갖는 상수 행렬들이다. 전형적인 H_∞제어기 설계 문제는 외란 w_k로부터 측정출력 z_k까지의 L_2이득을 최소화하는 다음의 상태 궤환 제어기를 구하는 것이고

$$u_k=Kx_k \tag{2}$$

이 문제를 하나의 수식으로 표현하면 다음과 같다.

$$Find\ u_k=Kx_k\ \ s.t.\ minimizes\ \gamma=\left\{\frac{\|z_k\|_2}{\|w_k\|_2}\right\} \tag{3}$$

여기서 $\gamma^*=\min_{(u_k=Kx_k)}\{\gamma\}$는 외란 w_k로부터 측정출력 z_k까지 최소화된 L_2이득이고, 이를 만족하는 제어 $u_k=K^*x_k$는 최적 H_∞제어기이다. 그리고 이 문제는 참고문헌[12]에 잘 알려진 바와 같이 다음의 스칼라함수 J_P^v 정의하면

$$J_P^v(x_k,u_k,w_k):=x_{k+1}^TPx_{k+1}-x_k^TPx_k+z_k^Tz_k-\gamma^2w_k^Tw_k,\ P>0 \tag{4}$$

수식 (3)의 문제는 다음과 동치이다.

$$\min_{(K,P)}\{\gamma\},\ \ subject\ to\ \max_{\forall w_k}\{J_P^v(x_k,u_k,w_k)\}\le 0. \tag{5}$$

또한 시스템의 행렬들이 모두 알려진 경우에는 (5)를 시스템 (1)에 적용하면 선형행렬부등식(LMI) 형태의 다음의 결과를 얻는다.

$$\min_{(Y;P>0)}\{\gamma\},\ \text{subject to}\ \begin{bmatrix} -Q & 0 & QA^T+Y^TB_1^T & QC^T+YB_2^T \\ 0 & -\gamma^2 & E_1^T & E_2^T \\ \star & \star & -Q & 0 \\ \star & \star & \star & -I \end{bmatrix}\le 0 \tag{6}$$

여기서 $P=Q^{-1}>0$, $K=YQ^{-1}$이고, 또한 LMI형태인 (6)은 잘 알려진 Matlab의 제어패키지(LMItoolbox)를 이용하면 손쉽게 행렬 P와 제어기 이득 행렬 K를 찾을 수 있다. 그러나 (6)을 이용하여 제어 (2)를 구하려면 시스템 (1)의 행렬들(A,B_1,B_2,C,E_1,E_2)이 알려져 있어야한다. 그러나 본 연구에서는 이들 행렬이 알려지지 않은 경우에 대하여 제어 (2)를 구하여야하는 관계로 시스템 행렬이 포함되지 않은 형태의 새로운 관계식을 유도하고 이로부터 제어기를 구하여야한다. 이를 위하여 다음과 같은 이차함수(quadratic function) Q_H와 $\overline{Q_H}$를 정의하자.

$$\begin{cases} Q_H(x_k,u_k,w_k):=\xi_k^TH\xi_k \\ \overline{Q_H}(x_k,u_k):=\overline{\zeta_k}^T\overline{H}\overline{\zeta_k} \end{cases} \tag{7}$$

여기서 $\zeta_k^T=[x_k^T:u_k^T:w_k^T]$, $\overline{\zeta_k}=[x_k^T:u_k^T]$이고, 행렬 H, $\overline{H}$는 다음으로 정의되며

$$\begin{cases} H=H^T=\begin{bmatrix} H_{xx} & H_{xu} & H_{xw} \\ \star & H_{uu} & H_{uw} \\ \star & \star & H_{ww} \end{bmatrix}\ \text{with}\ \ H_{ww}<0 \\ \overline{H}=\overline{H}^T=\begin{bmatrix} \overline{H_{xx}} & \overline{H_{xu}} \\ \star & \overline{H_{uu}} \end{bmatrix} \end{cases} \tag{8}$$

또한 $\overline{H_{xx}},\overline{H_{xu}},\overline{H_{uu}}$는 다음으로 주어진다.

$$\begin{cases} \overline{H_{xx}}=H_{xx}-H_{xw}H_{ww}^{-1}H_{xw}^T, \\ \overline{H_{xu}}=H_{xu}-H_{xw}H_{ww}^{-1}H_{uw}^T, \\ \overline{H_{uu}}=H_{uu}-H_{uw}H_{ww}^{-1}H_{uw}^T. \end{cases}$$

다음의 보조정리는 Q-학습을 이용하여 L_2최적제어를 구하는 과정에 필요한 예비결과이다.

보조정리 1: 위의 (7)과 (8)에 정의된 이차함수 Q_H, $\overline{Q_H}$와 다음의 스칼라 함수를 $J_1=\max_{\forall w_k}Q_H(x_k,u_k,w_k)$ 생각하자. 그러면 다음의 결과를 얻는다.

$$\min_{u_k=Kx_k}\{J_1\}=x_k^T\left[\overline{H_{xx}}-\overline{H_{xu}}\,\overline{H_{uu}}^{-1}\,\overline{H_{xu}}^T\right]x_k$$

여기서 최적의 값 K_o는 $K_o = -\overline{H_{uu}}^{-1}\overline{H_{ux}}$이다.

증명: 먼저 (7)에 정의된 $H_{ww} < 0$의 사실과 다음의 관계식 $a^Tb+b^Ta \le a^TX^{-1}a+b^TXb, \forall X>0$ 를 이용하면 다음의 결과를 얻는다.

$$\begin{aligned} J_1 &= \max_{\forall w_k} Q_H(x_k,u_k,w_k) \\ &= \zeta_k \begin{bmatrix} H_{xx} & H_{xu} \\ \star & H_{uu} \end{bmatrix} \zeta + \max_{\forall w_k} \left\{ 2\zeta_k^T \begin{bmatrix} H_{xw} \\ H_{uw} \end{bmatrix} w_k + w_k^T H_{ww} w_k \right\} \\ &\le \zeta_k \begin{bmatrix} H_{xx} & H_{xu} \\ \star & H_{uu} \end{bmatrix} \zeta_k + \zeta_k^T \begin{bmatrix} H_{xw} \\ H_{uw} \end{bmatrix} \left(-H_{ww}^{-1}\right) \left[H_{xw}^T : H_{uw}^T\right] \zeta_k \\ &= \zeta_k \begin{bmatrix} \overline{H_{xx}} & \overline{H_{xu}} \\ \star & \overline{H_{uu}} \end{bmatrix} \zeta_k := J_1^* \end{aligned}$$

여기서 위의 부등식에서 등식이 성립하는 경우는 $w_k = -H_{ww}^{-1}[H_{xw}^T : H_{uw}^T]\zeta_k$인 경우이다. 다음으로 J_1^*는 2차 함수이므로, 이를 최소로 하는 u_k^*를 쉽게 $\nabla_{u_k}\{J_1^*\}=0$를 이용하여 구할 수 있고, 간단한 계산을 거치면 우리는 $u_k^* = K^*x_k;\ K^* = -\overline{H_{uu}^{-1}}\overline{H_{ux}}$를 얻는다. 다음으로 이의 결과를 J_1에 대입하면 $J_1|_{u_{k=k}^*} = x_k^T\left[\overline{H_{xx}} - \overline{H_{xu}}\ \overline{H_{uu}^{-1}}\ \overline{H_{xu}^T}\right]x_k$을 얻는다. 이로써 증명을 마친다.

3. 입출력 데이터 기반 Q-함수와 LMI에 기초한 H_∞ 제어기 설계

이제 시스템 행렬들을 이용하지 않고 시스템의 입력과 출력만을 이용하여 H_∞ 제어기를 설계하자. 이 새로운 설계 방법은 적응동적계획 기법중의 하나인 Q-학습을 이용하는 것이다. 이의 기본적인 발상은 먼저 시스템의 성능지수 함수를 정한 후, 입력과 출력을 이용하여 이의 함수 값을 계산하고 이 계산된 함수 값을 바탕으로 새로운 제어를 설계하는 반복적인 제어기 설계 기법이다. 먼저, 시스템 (1)의 궤적에 따른 (4)에 정의된 함수 J_P^i를 계산하면 다음을 얻고

$$\begin{aligned} J_P^i &:= x_{k+1}^T P x_{k+1} - x_k^T P x_k + z_k^T z_k - \gamma^2 w_k^T w_k \\ &= \xi_k^T \begin{bmatrix} A^TPA + C^TC - PA^TPB_1 + C^TB_2 & A^TPE_1 + C^TE_2 \\ \star & B_1^TPB_1 + B_2^TB_2 & B_1^TPE_1 + B_2E_2 \\ \star & \star & E_1^TPE_1 + E_2^TE_2 - \gamma^2 I_d \end{bmatrix} x_k \end{aligned}$$

여기서 $\xi_k^T = [x_k^T : u_k^T : w_k^T]$이다. 따라서 $J_P^i(x_k,u_k,w_k) \le 0,\ \forall w_k$을 만족하는 최소 γ를 구하기 위해서는 시스템 행렬 $(A, B_1, B_2, C, E_1, E_2)$를 모두 알아야 한다. 우리의 문제는 이들 행렬이 알려져 있지 않으므로 이들을 사용하지 않도록 하기 위하여 (7)에 정의된 다음과 같은 Q-함수를 정의하자.

$$Q_H(x_k,u_k,w_k) := \xi_k^T H \xi_k;\ H = H^T = \begin{bmatrix} H_{xx} & H_{xu} & H_{xw} \\ \star & H_{uu} & H_{uw} \\ \star & \star & H_{ww} \end{bmatrix}. \quad (9)$$

그러면 시스템을 안정화 시키는 임의의 입력 $u_k = Kx_k$를 적용하였을 때, 시스템의 입출력 데이터 $\{x_k\},\{u_k\},\{w_k\}$를 이용하여 다음을 만족하는 최소 스칼라 값 γ와 행렬 P, H를 시스템의 행렬들이 알려져 있지 않아도 구할 수 있다.

$$J_P^i(x_k,u_k,w_k) = Q_H(x_k,u_k,w_k) \le 0 \quad (10)$$

여기서 J_P^i는 수식 (4)에 정의되어있는 함수이다. Q-학습의 기본 개념은 다음의 두 단계를 반복 수행하여 최적의 L_2이득 γ와 이를 실현하기위한 L_2최적제어 $u_k = K^*x_k$를 구하는 것이다.

$$\min_{P,H}\{\gamma\},\ subject\ to\ \text{eqn. (10)} \quad (11)$$

$$Find\ u_k = Kx_k\ \ by\ \nabla_{u_k}\{\max_{\forall w_k} Q_H(x_k,u_k,w_k)\} = 0. \quad (12)$$

관계식 (10)을 만족하는 행렬 H를 구하기 위하여 기존에 일반적으로 사용된 방법은 유사역변환(pseudo-inverse)을 이용하는 것이다[1][2]. 그러나 이는 많은 제약 조건과 수렴성이 떨어지므로, 이를 해결하기 위하여 우리는 선형행렬부등식(LMI)를 이용하도록 한다. 그러나 유감스럽게 (10)은 선형행렬부등식의 형태가 아니어서 곧바로 이용하기가 어려우므로, 우리는 (10)을 선형행렬부등식형태로 변환하여 이용하고자 한다. 즉, (10)를 근사화하면 다음이 된다.

$$\begin{cases} |J_P^i(x_k,u_k,w_k) - Q_H(x_k,u_k,w_k)| \le \varepsilon, \\ Q_H(x_k,u_k,w_k) \le 0, \quad \forall k \end{cases} \quad (13)$$

여기서 $\varepsilon \to 0$인 경우에는 (10)과 (13)이 동치 관계이다. 다음으로 관계식 (13)을 선형행렬부등식 형태로 등가 변환하면 다음을 얻는다.

$$\begin{cases} \begin{bmatrix} \varepsilon & J_P^i(x_k,u_k,w_k) - Q_H(x_k,u_k,w_k) \\ \star & \varepsilon \end{bmatrix} \ge 0, \\ J_P^i(x_k,u_k,w_k) + \varepsilon \le 0, \forall k. \end{cases} \quad (14)$$

따라서 우리는 (11)와 (12)를 수행하는 대신, $\varepsilon \to 0$ 하에서 다음을

$$\min_{P,H}\{\gamma\},\ subject\ to\ \text{LMI (14)}$$

$$Find\ u_k = Kx_k\ \ by\ \nabla_{u_k}\{\max_{\forall w_k} Q_H(x_k,u_k,w_k)\} = 0$$

반복 수행함으로써 수렴 된 L_2최적제어 $u_k = Kx_k$를 구할 수 있다. 또한 외란 w_k의 주파수 성분이 충분한 경우, 반복 수행에 의하여 구하여진 제어기는 시스템의 행렬이 알려진 경우의 표준 L_2최적제어 $u_k = K^*x_k$에 수렴한다. 이의 수렴성에 대한 증명은 다음 장의 알고리즘 뒤에 한다.

4. H_∞ 제어기 설계 알고리즘과 수렴성

4.1 H_∞ 제어기 설계 알고리즘

다음은 위에서 기술한 입출력 데이터를 사용한 Q-함수와 LMI에 기초한 H_∞ 제어기를 설계하는 알고리즘을 제시하고, 이의 알고리즘이 시스템의 행렬이 알려진 경우의 표준 L_2최적제어 수렴함을 보인다.

step 1: $i=0$ 이라 하고, 사용할 적당한 I/O 데이터 개수 N과 폐루프 시스템의 안정성을 보장하는 초기제어기 $K^{(0)}$를 선택한다.

step 2: $K=K^{(i)}$라하고, 제어 $u_k=Kx_k+v_k$과 외란 w_k를 시스템에 적용하여 데이터 $\{x_k\}_{k=0}^{N}$, $\{u_k\}_{k=0}^{N}$, $\{w_k\}_{k=0}^{N}$를 얻는다. 여기서 v_k,w_k는 주파수 성분이 충분한 외란이며, 특히 v_k는 시스템의 충분한 활성화를 위하여 가미된 매우 적은 크기의 외란이다.

step 3: 다음의 LMI로 기술된 최소화 문제를 만족하는 행렬 $P^{(i)},H^{(i)}$와 스칼라 $\gamma^{(i)}$를 구한다.

$$\min\{\gamma^{(i)}\},\ \text{subject to}$$

$$\text{(i)}\quad \begin{bmatrix} \varepsilon e^{-\alpha i} Q_H - J_P^{(i)} & \\ \star & \varepsilon e^{-\alpha i} \end{bmatrix} \geq 0$$

$$\text{(ii)}\quad J_P^{(i)} + \varepsilon e^{-\alpha i} \leq 0,\ \forall k \in [0, N]$$

여기서 $\varepsilon>0$은 충분히 작은 상수이며, $\alpha>0$은 적당한 감쇠상수이다.

step 4: 만약 충분히 작은 $\varepsilon_1>0$에 대하여 $|\Delta\gamma^{(i)}| \leq \varepsilon_1$이면 다음의 **step 6**으로 간다.

step 5: 새로운 제어기 $K^{(i+1)}$를 다음 식을 이용하여 구한 후, **step 2**로 간다.

$$K^{(i+1)} = -\left[H_{uu}^{(i)} - H_{uw}^{(i)} (H_{ww}^{(i)})^{-1} (H_{uw}^{(i)})^T\right]^{-1} \cdot \left[H_{xu}^{(i)} - H_{xw}^{(i)} (H_{ww}^{(i)})^{-1} (H_{uw}^{(i)})^T\right]^T.$$

step 6: 설계된 H_∞제어기는 $u_k=K^{(i)}x_k$이고, 이는 폐루프 시스템의 L_2이득이 $\gamma^{(i)}$보다 크지 않음을 보장한다.

Remark 1: i 번째의 제어기 $u_k=K^{(i)}x_k$를 가하여 얻은 L_2이득을 $\gamma^{(i)}$라하고, $i+1$ 번째의 제어기 $u_k=K^{(i+1)}x_k$를 가하여 얻은 L_2이득을 $\gamma^{(i+1)}$라 하면, $\gamma^{(i+1)} \leq \gamma^{(i)}$이다. 이는 위의 반복 알고리즘 특성상 다음의 관계식이 얻어지므로

$$\begin{aligned} 0 &\geq J_{P^{(i)}}^{(i)} \\ &= (x_{k+1}^{(i)})^T P^{(i)} x_{k+1}^{(i)} - (x_k^{(i)})^T P^{(i)} x_k^{(i)} + (z_k^{(i)})^T z_k^{(i)} - (\gamma^{(i)})^2 w_k^T w_k \\ &\geq (x_{k+1}^{(i+1)})^T P^{(i)} x_{k+1}^{(i+1)} - (x_k^{(i+1)})^T P^{(i)} x_k^{(i+1)} + (z_k^{(i+1)})^T z_k^{(i+1)} \\ &\quad - (\gamma^{(i)})^2 w_k^T w_k \\ &\geq (x_{k+1}^{(i+1)})^T P^{(i+1)} x_{k+1}^{(i+1)} - (x_k^{(i+1)})^T P^{(i+1)} x_k^{(i+1)} + (z_k^{(i+1)})^T z_k^{(i+1)} \\ &\quad - (\gamma^{(i)})^2 w_k^T w_k \end{aligned}$$

반드시 $J_{P^{(i-1)}}^{(i+1)} \leq 0$ 이 보장되는 $\gamma^{(i+1)} \leq \gamma^{(i)}$가 반드시 존재한다.

4.2 H_∞ 제어기 설계의 수렴성

다음은 제시된 H_∞제어기 설계 알고리즘이 외란w_k와 v_k는 주파수 성분이 충분히 풍부한 경우 모든 시스템 행렬이 알려진 최적의 경우에 수렴함을 보인다.

정리 1: 위에 제시된 알고리즘이 적용된 시스템 (1)을 생각하자. 그리고 외란w_k와 v_k는 주파수 성분이 충분히 풍부하다고 하고, $P^{(i)},K^{(i)},\gamma^{(i)}$는 반복지수 i에서 얻어진 값들이라 하자. 그러면 $i\to\infty$이면 $P^{(i)}\to P^*,K^{(i)}\to K^*,\gamma^{(i)}\to\gamma^*$이다. 여기서 P^*,K^*,γ^*는 시스템의 모든 행렬이 알려진 경우의 L_2 최적해이다.

증명: 다음 값 P^*,K^*,γ^*를 L_2최적해라 하고 $u_k=K^*x_k$를 L_2최적제어라 하자. 그러면 다음이 성립하고

$$\text{(i)}\quad J_{P^*}(x_k,u_k^*,w_k) \leq 0,$$

$$\text{(ii)}\quad J_P(x_k,u_k,w_k)>0,\ \forall P\neq P^*,\forall K\neq K^*,\forall\gamma\leq\gamma^*$$

또한 위에서 제시된 제어기 설계 알고리즘과 remark 1에 의하여 다음이 성립한다.

$$J_{P^{(i)}}^{(i)}(x_k,u_k^*,w_k) \leq 0,\forall k \text{ and } \gamma^{(i+1)} \leq \gamma^{(i)},\forall i\ .$$

그리고 γ^*는 최적해기에 $\gamma^* \leq \gamma^{(i)},\forall i$가 성립한다. 증명은 $\lim_{i\to\infty} K^{(i)} = K^*$임을 보이면 완성되며, 이는 대우법(by contradiction)으로 한다. 이를 위하여 충분히 큰 N_0반복 후에 $K^{(i)}$가 수렴하여 $K^{(N_0)}$가 되었는데에도 불구하고 수렴값이 최적해 K^*와 다르다고 가정하자. 즉, $K^{(i)}=K^{(N_0)}\neq K^*,\forall i\geq N_0$. 그러면 이것은 K^*가 정상상태에서 가질 수 있는 유일한 값이라는데 위배된다. 이것은 만약 $K^{(N_0)}\neq K^*$이면 반드시 $K^{(N_0+1)}\neq K^{(N_0)}$이 되어야하고 이는 $K^{(N_0+1)}=K^{(N_0)}\neq K^*$에 위배된다. 따라서 제어기 $K^{(i)}$가 수렴

하면 반드시 K^*가 되어야하고, $K^{(i)}$가 수렴하면 $P^{(i)},\gamma^{(i)}$도 반드시 수렴하여야하고, 수렴 값은 반드시 P^*,γ^*가 되어야한다. 이로써 증명을 마친다.

Remark 2: 시스템을 충분히 여기(excitation)시키기 위해서는 시스템의 입력에 여러 가지 주파수 성분이 고루 분포되어 있어야하고, 이런 경우 주파수 성분이 충분히 풍부하다고 한다. 이런 의미에서, 위의 정리1에서 사용되는 외란 w_k와 v_k는 주파수 성분이 충분히 풍부하여야 시스템을 충분히 여기 할 수 있다.

5. 수치 예제

위에서 새로이 제시된 H_∞제어기 설계의 유용성을 보이기 위한 다음의 간단한 선형이산시간 시스템을 생각하자.

$$\begin{cases} x_{k+1} = \begin{bmatrix} 0.1 & 1.0 \\ 0.5 & -0.3 \end{bmatrix} x_k + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_k + \begin{bmatrix} 1 \\ 0 \end{bmatrix} w_k \\ z_k = [1 \quad 0] x_k \end{cases} \tag{15}$$

여기서 시스템 행렬의 고유치는 $\lambda(A)=\{0.6348,\ -0.8348\}$ 이므로, 개루프(open-loop) 시스템은 안정하다. 위 시스템 (15에서 시스템의 행렬들이 모두 알려진 경우의 표준 H_∞제어를 잘 알려진 LMI (6)에 의하여 구하면 다음을 얻는다,

$$\gamma_{\text{open}} = 3.7037, \quad \gamma^* = 1.0522, \quad K^* = -[0.6030 \quad 0.7319] \tag{16}$$

여기서 γ_{open}은 개루프(open-loop) 상태의 L_2이득이고, γ^*, K^*는 모든 시스템 행렬이 알려진 경우의 최적 L_2이득과 이 때의 제어기 행렬이다.

다음은 위에서 새로이 제시된 Q-학습을 이용한 제어기를 설계한다. 이를 위하여 시스템을 충분히 자극시킬 수 있는 주파수 성분이 풍부한 외란 w_k,v_k가 필요하다. 이를 위하여 여기서는 각각 평균편차(variance) $1, 10^{-6}$를 갖는 각각 영평균 백색 무작위(zero mean white random) 신호를 선택하였다. 다음의 그림 1은 시뮬레이션에 사용된 $w_k = w_k^0$를 나타낸다.

또한 제시된 H_∞제어기 설계 기법이 초기제어기의 선택에 대하여 강인함을 보이기 위하여, 폐루프 시스템의 안정도를 보장하는 다음의 서로 다른 두 개의 초기 제어기를 사용하여 시뮬레이션을 수행한다.

$$\begin{cases} \text{Case(i)}: K^{(0)} = [0.0,\ 0.0] \\ \text{Case(ii)}: K^{(0)} = [0.1, -0.1]. \end{cases} \tag{17}$$

시뮬레이션에서 LMI로 표시된 문제의 해를 구하기 위하여 Matlab의 `mincx.m`을 사용하였고, 시간에 따른 궤적을 얻기 위하여 simulink를 이용하였다.

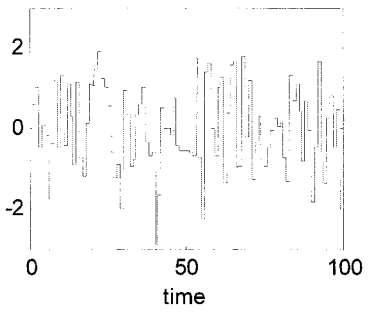

그림 1 시뮬레이션에 사용된 외란 w_k^0

Fig. 1 The used disturbance w_k^0 in simulations

그리고, 시뮬레이션에서는 $N=100, \varepsilon=10^{-9}, \alpha=0.1$의 변수값들이 이용되었다. 위에서 새로이 제시된 H_∞제어기 설계 알고리즘에 의하여 위의 두 가지 경우의 초기치에 대한 각 반복(iteration)에 대한 L_2 최소값 $\gamma^{(i)}$와 이에 상응한 H_∞ 제어기 $K^{(i)}=[K_1^{(i)},\ K_2^{(i)}]$를 얻었다. 이를 그림으로 표시한 것이 다음의 그림 2와 그림 3이다. 먼저 그림 2는 Case (i)에 대한 결과이고

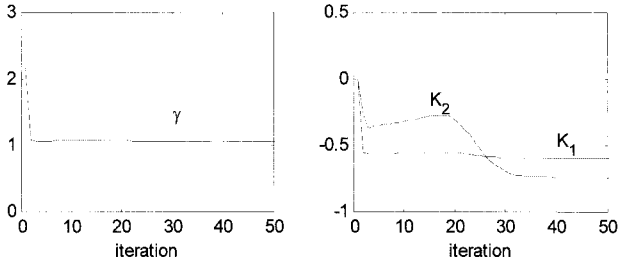

그림 2 초기치 case (i) 에 대한 $\gamma^{(i)}$와 $K^{(i)}$

Fig. 2 $\gamma^{(i)}$ and $K^{(i)}$ for initial value of case (i)

다음 그림 3은 초기치 case (ii)에 대한 결과이다.

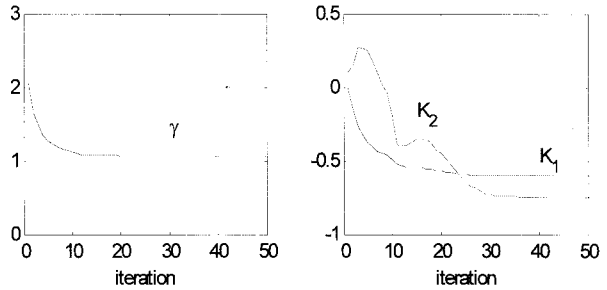

그림 3 초기치 case (ii) 에 대한 $\gamma^{(i)}$와 $K^{(i)}$

Fig. 3 $\gamma^{(i)}$ and $K^{(i)}$ for initial value of case (ii)

위의 결과들은 우리가 예측한 바대로 유한한 반복지수 ($i \simeq 40$)에서 수렴하며 이의 수렴한 값과 모든 시스템의 행렬이 알려진 경우와의 비교는 다음의 표 1에 제시되었다.

표 1 최적 H_∞과 Q-학습 H_∞과의 결과 비교

Table 1 Comparison of optimal and Q-learning

	최적 H_∞	Q-학습 H_∞	
		case (i)	case (ii)
γ	1.0522	1.0512	1.0512
K^T	$-\begin{bmatrix}0.6030\\0.7319\end{bmatrix}$	$-\begin{bmatrix}0.6051\\0.7512\end{bmatrix}$	$-\begin{bmatrix}0.6051\\0.7512\end{bmatrix}$

이 표에서 보듯이 새로이 제시된 Q-학습에 의한 H_∞ 제어기 설계 결과는 모든 시스템 행렬이 알려진 경우에 매우 유사하고 최적제어의 L_2이득 보다 약간 작다. 이는 최적제어의 경우는 모든 w_k에 대한 설계이나, Q-학습에 이용된 외란은 특별한 경우의 w_k^0이기 때문이다. 만약 여러 가지 경우의 외란 w_k를 이용하여 Q-학습에 의한 제어기를 설계하였다면 두 경우의 L_2이득과 제어기 행렬은 동일할 것이다.

다음은 외란 $w_k = w_k^0$에 대하여 설계된 위의 제어기들에 대하여 다음의 서로 다른 세 가지 외란이 각각 가하여진 경우에 대하여

$$w_k = \begin{cases} w_{k1} = w_k^0 \\ w_{k2} = \sin(0.1k) \\ w_{k3} = \sin(k) + \cos(0.1k) \end{cases}$$

각각의 시스템의 시간궤적을 구하고 이들의 I/O 데이터로부터 계산된 L_2이득을 다음 표 2에서 보여준다.

표 2 서로 다른 외란 w_k에 대한 L_2 이득 비교

Table 2 Comparison of L_2 gain for various w_k

	최적 H_∞	Q-학습 H_∞	
		case (i)	case (ii)
w_{k1}	1.0398	1.0420	1.0420
w_{k2}	1.0485	1.0479	1.0479
w_{k3}	1.0419	1.0413	1.0413

위의 표 2는 새로이 제시된 Q-학습을 이용한 H_∞ 제어기 설계 결과는 모든 시스템 행렬이 알려진 경우의 최적 H_∞에 비해 그 차이가 매우 미미함을 보이 보이고, 만약 설계 시 w_k가 좀 더 충분한 주파수 특성을 가져 모든 w_k에 해당된다면 이의 결과는 동일하게 될 것이다.

5. 결 론

모델링이 되어 있지 않은 선형 이산 시간 시스템에 대하여 Q-학습기법을 이용한 H_∞ 제어기 설계방법을 제시하였다. 제시된 H_∞ 제어기 설계 방법은 다음 세 단계를 계속적으로 반복함으로써 얻어진다.

(i) 제어기를 포함한 시스템의 I/O 데이터를 얻는다.

(ii) I/O 데이터를 이용하여 LMI로 표시된 Q-함수를 L_2 이득을 최소화함으로써 얻는다.

(iii) Q-함수를 최소화 하는 다음 단계의 H_∞ 제어기를 얻는다.

또한 새로이 제시된 알고리즘은 입력 외란이 충분한 주파수 성분을 가지는 경우 모든 시스템 행렬이 알려진 경우에 수렴함을 보였다. 끝으로 하나의 예제를 통하여 제시된 Q-학습을 이용한 H_∞ 제어기 설계 방법의 유용성을 보였다.

감사의 글

이 논문은 2008년도 충북대학교 학술연구지원사업의 연구비지원에 의하여 연구되었음.

참 고 문 헌

[1] M. Abu-Khalaf and F.L. Lewis, "Nearly optimal controls laws for nonlinear systems with saturating actuators using a neural network HJB approach," *Automatica* ,vol. 41, no.5, pp.779-791, 2005.

[2] A. Al-Tamimi, M. Abu-Khalaf and F.L. Lewis, "Model-Free Q-Learning Designs for Discrete-Time Zero-Sum Games with Application to H-Infinity Control," *Automatica*, vol.43, no.3, pp.473-482, 2007.

[3] S.J. Bradtke, B.E. Ydstie and A.G. Barto, "Adaptive Linear Quadratic Control Using Policy Iteration, *Proc. of ACC*, pp.3475-3476, 1994.

[4] G. Saridis and C.S. Lee, "An Approximation Theory of optimal Control for Trainable Manipulators," *IEEE Trans. Systems, Man, Cybernetics*, vol.9, no.3, pp.152-159, 1979.

[5] P.J. Werbos, "Approximate dynamic programming for real-time control and neural modeling," *Handbook of Intelligent Control*, edited by D.A. White and D.A. Sofge, New York: Van Nostrand Reinhold, 1992.

[6] R.A. Howard, *Dynamic Programming and Markov Processes*, MITPress, Cambridge, 1960.

[7] P. Werbos, "Neural networks for control and system identification", *Proc. of CDC*, 1989.

[8] C.J. Watkins. *Learning from delayed rewards*, Ph.D. Thesis, University of Cambridge, England, 1989.

[9] S. Boyd, L. El Ghaoui, E. Feron and V. Balakrishnan, *Linear matrix inequalities in systems and control theory*, Philadelphia, PA: SIAM, 1994.

[10] D.P. Bertsekas and J.N. Tsitsiklis. *Neuro-Dynamic*

Programming, Athena Scientific, MA.1996.
[11] K. Zhou and J.C. Doyle. *Essentials of robust control*, Prentice-Hall, 1997.
[12] R.S. Sutton and A.G. Barto. *Reinforcement Learning -An introduction*, MIT Press, Cambridge, 1998.
[13] J. Si, A. Barto, W. Powel and D. Wunch, *Handbook of Learning and Approximate Dynamic Programming*, John Wiley, New Jersey, 2004.

저 자 소 개

김 진 훈 (金 鎭 勳)

1961년 10월 18일생. 1985년 서울대 전기공학과 졸업. 1985년-1986년 신영전기(주) 연구원. 1989년 한국과학기술원 전기 및 전자공학과졸업(석사). 1993년 동 전기 및 전자공학과 졸업(공박). 1993년-1994년 경상대 제어계측공학과 전임강사. 1998년 미국 UCI 방문교수. 2008년 미국 UTA 방문교수. 1995년 - 현재 충북대학교 전자정보대학 교수.
Tel : 043-261-2387
Fax : 043-268-2386
E-mail : jinhkim@chungbuk.ac.kr

Frank L. Lewis

1971년 미국 Rice대학 (물리,전기,기계)학과 졸업. 1977년 미국 West Fla. 대학 항공공학과 졸업(석사). 1981년 Georgia Tech. 전기공학과 졸업(공박). 1981년-1990년 Georgia Tech. 전기공학과 교수, 1990년-현재 UTA 전기공학과 교수. IEEE Fellow, Fellow U.K. Inst. Measurement & Control.
E-mail : lewis@arri.uta.edu