

입출력의 증감 정보를 이용한 LQR 제어기 학습법

정병목[#]

A Learning Method of LQR Controller using Increasing or Decreasing Information in Input-Output Relationship

Byeong Mook Chung[#]

ABSTRACT

The synthesis of optimal controllers for multivariable systems usually requires an accurate linear model of the plant dynamics. Real systems, however, contain nonlinearities and high-order dynamics that may be difficult to model using conventional techniques. This paper presents a novel learning method for the synthesis of LQR controllers that doesn't require explicit modeling of the plant dynamics. This method utilizes the sign of Jacobian and gradient descent techniques to iteratively reduce the LQR objective function. It becomes easier and more convenient because it is relatively very easy to get the sign of Jacobian instead of its Jacobian. Simulations involving an overhead crane and a hydrofoil catamaran show that the proposed LQR-LC algorithm improves controller performance, even when the Jacobian information is estimated from input-output data.

Key Words : LQR(Linear Quadratic Regulator), LC(Learning Control: 학습제어), Overhead Crane (크레인), Hydrofoil Catamaran (수중익 쌍동선)

1. 서론

다변수 시스템에서 모델식을 구할 수 있다면, 최적의 제어기로 LQR(Linear Quadratic Regulator) 제어기가 널리 알려져 있다.¹⁻³ 그러나 산업 현장과 같이 시스템의 모델식을 구하기 어려운 경우에는 LQR 제어기를 설계하기가 매우 힘들다. 이러한 경우에는 신경회로망을 이용하여 시스템을 모사하는 모델을 구한 다음에 이 모델을 토대로 적응제어기를 설계하는 방법이 알려져 있지만 시스템을 모사하는 모델과 실제 모델 사이의 에러를 줄이지 못하면 설계된 제어기의 성능이

기대치에 못 미치는 경우가 많다.⁴⁻⁸ 따라서 시스템의 모델식을 구하기 어려운 경우, 모델식이나 모사 모델을 대신에 자코비안을 이용하여 LQR 제어기를 설계하는 방법이 제안된 바 있다.⁹ 이 논문은 LQR 제어기와 같이 상태변수의 목표값과 결과값 간의 에러와 입력을 목적함수로 정의하고, 이 목적함수를 최소화 되도록 학습을 통하여 제어기를 수정한다. 이때 목적함수를 최소화하기 위해서는 시스템의 입력 변화에 대한 출력 변화의 관계를 나타내는 자코비안이 사용되었다.⁹⁻¹³ 그러나 시스템의 모델식을 모르는 다중 입출력을 갖는 동적

접수일: 2006년 4월 20일; 게재승인일: 2006년 7월 11일

영남대학교 공과대학 기계공학부

E-mail bmchung@yu.ac.kr Tel. (053) 810-2569

시스템에서 실험적으로 자코비안을 구하는 일이 간단한 일은 아니다.⁹

본 논문에서는 시스템의 자코비안 대신에 입력에 대한 출력의 증감 정보(자코비안의 부호)만을 이용하여 LQR 제어를 실시간으로 학습할 수 있음을 보이고자 한다. 자코비안과 달리 자코비안의 부호는 입력에 대해 출력변화를 정량적이 아니라 정성적으로 구할 수 있다. 즉, 임의의 입력에 대해 각각의 출력이 증가하는지 또는 감소하는지 만을 판단하면 되므로 비록 외란이 많은 동적 시스템에서 조차도 비교적 구하기 쉽다. 시뮬레이션에서는 크레인과 수중익 쌍동선의 제어 문제를 통해 모델식에 근거한 LQR의 결과와 자코비안의 부호 정보만을 이용한 LQR의 학습 결과를 서로 비교하였다.

2. LQR 제어기의 학습법

LQR 제어기는 시스템의 상태방정식을 구속조건으로 하여 주어진 목적함수(cost function)를 최소화하는 최적제어 이론에 기반을 둔 상태 피드백 기법이다. 선형 시스템의 상태 방정식은 식(1)과 같이 표현한다.

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \quad (1)$$

여기서 초기 상태 $\mathbf{x}(0)$ 는 주어지고 있고 $\mathbf{x}(t) \in R^p$, $\mathbf{u}(t) \in R^m$ 이며 시스템은 안정 가능하다고 가정한다. 그리고 LQR의 목적함수 C 를 다음과 같이 정의한다.

$$C = \int_0^{\infty} \left[\mathbf{x}(t)^T \mathbf{Q}\mathbf{x}(t) + \mathbf{u}(t)^T \mathbf{R}\mathbf{u}(t) \right] dt \quad (2)$$

여기서 \mathbf{Q} 는 $p \times p$ 인 양의 반한정 대칭행렬인 상태가중행렬이고, \mathbf{R} 은 $m \times m$ 인 양의 한정 대칭행렬인 제어가중행렬이다. 최적제어 문제는 상태방정식의 구속조건에 대해 목적함수를 최소화하는 제어법칙을 찾는 문제로서 최적제어가 유일하게 존재하고, 실시간에서 모든 상태변수를 측정할 수 있다면 최적 제어 법칙을 다음과 같이 피드백 형태로 나타낼 수 있다.

$$\mathbf{u}(t) = -\mathbf{K}\mathbf{x}(t) \quad (3)$$

여기서 제어게인행렬 \mathbf{K} 는 다음과 같이 정의되며 \mathbf{S} 는 양의 반한정 대칭행렬로서 다음과 같은 제어대수 Riccati 식의 유일해이다.

$$\mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{S} \quad (4)$$

$$\mathbf{S}\mathbf{A} + \mathbf{A}^T\mathbf{S} + \mathbf{Q} - \mathbf{S}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S} = 0 \quad (5)$$

다변수 시스템의 LQR 제어를 설계하기 위해서는 식(5)의 Riccati 식으로부터 구할 수 있지만 시스템의 모델식을 모르는 경우에는 불가능하다. 따라서 제안하는 학습법은 식(2)의 LQR 제어기법과 같은 목적함수를 정의하고 이를 최소화하도록 LQR 제어기의 게인을 학습한다.

추적 문제(Tracking control)를 고려하여 LQR 제어를 상태변수 대신에 상태 변수의 에러를 이용하여 나타내면

$$\mathbf{u}(k) = \mathbf{K}\mathbf{e}(k) \quad (6)$$

이고 $\mathbf{e}(k) = \mathbf{d}(k) - \mathbf{x}(k)$ 상태 변수 에러이다. 그리고 LQR의 목적함수를 입력과 상태변수 에러의 합으로 나타내면 다음과 같다.

$$C = \frac{1}{2} \sum_{k=0}^{\infty} \left[\mathbf{e}^T(k)\mathbf{Q}\mathbf{e}(k) + \mathbf{u}^T(k)\mathbf{R}\mathbf{u}(k) \right] \quad (7)$$

실시간 학습을 위해서는 현재의 입력과 그에 따른 출력만을 고려하는 것이 바람직하므로 k 번째 스텝에서 작용한 입력이 $k+n$ 번째 출력에 가장 큰 영향을 끼쳤다고 가정하면 k 번째 스텝에서 줄여야 할 비용함수는

$$c(k) = \frac{1}{2} \left[\mathbf{e}^T(k+n)\mathbf{Q}\mathbf{e}(k+n) + \mathbf{u}^T(k)\mathbf{R}\mathbf{u}(k) \right] \quad (8)$$

이며 이때, k 번째 스텝에서 LQR 제어게인은 다음과 같이 수정될 수 있다.

$$\Delta\mathbf{K}(k) \propto -\frac{\partial c(k)}{\partial \mathbf{K}} \quad (9)$$

비용함수는 상태변수 에러와 입력의 함수이므로 다음과 같이 정리할 수 있다.⁹

$$\Delta K(k) \propto - \left[\mathbf{e}^T(k+n) \mathbf{Q} \frac{\partial \mathbf{x}(k+n)}{\partial \mathbf{u}(k)} + \frac{\partial c(k)}{\partial \mathbf{u}(k)} \right] \frac{\partial \mathbf{u}(k)}{\partial \mathbf{K}} \quad (10)$$

이 된다. 이 식을 자코비안을 이용해 정리하면

$$\Delta K(k) = \eta \left[\mathbf{J}^T \mathbf{Q} \mathbf{e}(k+n) - \mathbf{R} \mathbf{u}(k) \right] \mathbf{e}^T(k), \quad (11)$$

$$\mathbf{J} \equiv \frac{\partial \mathbf{x}(k+n)}{\partial \mathbf{u}(k)}. \quad (12)$$

여기서 η 는 학습율이다. 그런데 일반적으로 LQR 제어기에서는 입력의 범위를 제한하기 위한 방법으로 입력의 가중치를 이용하지만 학습에 의해 LQR 제어기를 구하는 경우에는 학습 시에 이미 입력이 최대값으로부터 제한을 받기 때문에 \mathbf{R} 에 대해서는 별도로 고려할 필요가 없다. 따라서 다음과 같이 수정된 피드백 게인의 학습식을 사용하는 것이 가능하다.

$$\Delta K(k) = \eta \mathbf{J}^T \mathbf{Q} \mathbf{e}(k+n) \mathbf{e}^T(k) \quad (13)$$

여기서 학습의 안정성을 해석하기 위해서는 수정된 제어제인에 의한 출력에러가 학습의 진행에 따라 수렴하는지 알아보는 것으로 충분하다. 식 (6)에 의해 다음 번의 입력은 다음과 같이 차이가 난다.

$$\Delta u(k) = \eta \mathbf{J}^T \mathbf{Q} \mathbf{e}(k+n) \mathbf{e}^T(k) \mathbf{e}(k) \quad (14)$$

여기서 $\mathbf{e}^T(k) \mathbf{e}(k)$ 는 양의 스칼라 값이므로 η 에 포함시킬 수 있다. 따라서 학습이 i 번 진행되고 난 후에 $k+n$ 번째 스텝에서 에러의 개선은 다음과 같이 된다.⁹

$$\mathbf{e}^i(k+n) = [\mathbf{I} - \eta \mathbf{J} \mathbf{J}^T \mathbf{Q}]^i \mathbf{e}^0(k+n) \quad (15)$$

여기서 \mathbf{I} 는 고유행렬(identity matrix)이고 만일 학습이 무한히 반복되고 나서 $\mathbf{e}^i(k+n)$ 가 영이

되기 위해서는 $\mathbf{J} \mathbf{J}^T \mathbf{Q}$ 가 양의 한정(positive definite) 대칭행렬이므로 충분히 작은 학습율(η)를 사용하면 $\|\mathbf{I} - \eta \mathbf{J} \mathbf{J}^T \mathbf{Q}\| < 1$ 에 의해 학습의 수렴성은 보장된다.¹⁴ Fig. 1는 식 (13)의 LQR 제어기 학습을 블록선도로 나타낸 것이다. LQR 제어제인은 시스템의 출력지연을 n 스텝이라고 둘 때, n 스텝 후의 에러와 자코비안을 이용하여 수정한다.

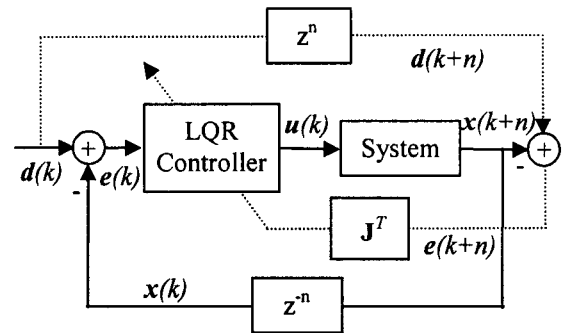


Fig. 1 Block Diagram of LQR learning controller using Jacobian

3. 학습의 수렴

시스템의 모델식을 모르는 상황에서 얼마나 정확한 자코비안을 구해야 학습이 수렴할 것인가 하는 문제는 중요하다. 먼저, 간단히 입력에 대한 출력의 증감 정보만으로 자코비안을 추정해 볼 수 있고 이렇게 구한 추정 자코비안을 사용하여 학습이 수렴한다면 LQR 제어기의 학습은 산업 현장에서도 쉽게 구현할 수 있는 편리한 방법이 되리라고 본다.

정리 1:

입력의 2×2 비특이(non-singular) 행렬을 \mathbf{A} 라고 하고 행렬 \mathbf{A} 의 각 요소 값의 부호만을 취한 행렬을 $\hat{\mathbf{A}}$ 라고 할 때, 행렬 $\mathbf{A} \hat{\mathbf{A}}^T$ 는 양의 한정(positive semi-definite)이다.

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \quad (16)$$

$$\hat{\mathbf{A}} = \begin{bmatrix} \text{sign}(a_{11}) & \text{sign}(a_{12}) \\ \text{sign}(a_{21}) & \text{sign}(a_{22}) \end{bmatrix} = \begin{bmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{bmatrix} \quad (17)$$

이라고 두면

$$\begin{aligned} \widehat{\mathbf{A}}\mathbf{A}^T &= \begin{bmatrix} a_{11}s_{11} + a_{12}s_{12} & a_{11}s_{21} + a_{12}s_{22} \\ a_{21}s_{11} + a_{22}s_{12} & a_{21}s_{21} + a_{22}s_{22} \end{bmatrix} \\ &= \begin{bmatrix} |a_{11}| + |a_{12}| & a_{11}s_{21} + a_{12}s_{22} \\ a_{21}s_{11} + a_{22}s_{12} & |a_{21}| + |a_{22}| \end{bmatrix} \end{aligned} \quad (18)$$

이므로 $\widehat{\mathbf{A}}\mathbf{A}^T$ 의 특이값(eigen value)은 다음과 같이 구할 수 있다.

$$\begin{aligned} |\lambda\mathbf{I} - \widehat{\mathbf{A}}\mathbf{A}^T| &= 0 \text{ or} \\ \begin{vmatrix} \lambda - (|a_{11}| + |a_{12}|) & -(a_{11}s_{21} + a_{12}s_{22}) \\ -(a_{21}s_{11} + a_{22}s_{12}) & \lambda - (|a_{21}| + |a_{22}|) \end{vmatrix} &= 0 \end{aligned} \quad (19)$$

이식을 전개하면

$$\begin{aligned} \lambda^2 - (|a_{11}| + |a_{12}| + |a_{21}| + |a_{22}|)\lambda + \\ (|a_{11}| + |a_{12}|)(|a_{21}| + |a_{22}|) + \\ (a_{11}s_{21} + a_{12}s_{22})(a_{21}s_{11} + a_{22}s_{12}) = 0 \end{aligned} \quad (20)$$

2 차 방정식이 항상 양의 근을 갖기 위해서는 1 차항이 음수고 상수항이 양수이면 된다. 위의 식에서 1 차항은 음수이므로 상수항만 양수인지를 조사하면 된다. 상수항을 정리하면 다음과 같고 항상 영보다 같거나 크다는 것을 알 수 있다.

$$(a_{11}a_{22} - a_{12}a_{21})(s_{11}s_{22} - s_{12}s_{21}) \geq 0 \quad (21)$$

즉, $a_{11}a_{22}$ 과 $a_{12}a_{21}$ 이 부호가 다르면 $(a_{11}a_{22} - a_{12}a_{21})$ 과 $(s_{11}s_{22} - s_{12}s_{21})$ 은 같은 부호를 가지므로 곱은 양의 값이 되고 같은 부호이면 $(s_{11}s_{22} - s_{12}s_{21})$ 에 의해 영이 된다. 따라서 행렬 $\widehat{\mathbf{A}}\mathbf{A}^T$ 는 양의 반한정(positive semi-definite)이다.

Q.E.D.

정리 1 에 의해 2 차원 입출력 시스템의 경우에는, 식 (13)의 학습식에서 입력에 대한 출력의 증감정보만 갖는 자코비안을 사용할

경우에도 LQR 계인의 학습이 적어도 발산하지 않는다는 것을 알 수 있다.

4. 시뮬레이션

4.1 크레인 시스템

Fig. 2 에서 보는 바와 같이 크레인 시스템은 1 개의 입력과 2 개의 출력이 있는 시스템이다. 크레인 시스템은 운반물을 크레인으로 잡아서 트레일러를 일정한 거리로 이동시키는 장치로써 트레일러의 이동 위치를 제어함과 동시에 운반물을 잡은 크레인의 진동도 역시 제어해야 한다. 따라서 트레일러를 재빨리 1m 이동시키면서 이때 발생하는 크레인의 각도를 제어하는 문제에 대해 LQR 제어를 구해 보고자 한다.

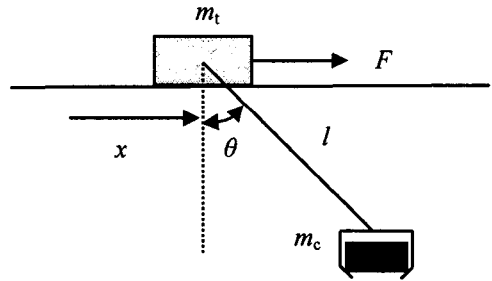


Fig. 2 Schematic diagram of Overhead Crane

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \\ \dot{x} \\ \ddot{x} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{(m_c + m_t)g}{m_t l} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{m_c g}{m_t} & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ x \\ \dot{x} \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{1}{m_t l} \\ 0 \\ \frac{1}{m_t} \end{bmatrix} F \quad (22)$$

식 (22)의 운동방정식을 크레인의 시뮬레이션에서 사용하였다. 여기서 트레일러(m_t)와 크레인(m_c)의 질량은 각각 100kg 과 200kg 이고 줄의 길이(l)는 2m 이며 g 는 중력가속도 이다. 그리고 입력의 최대치는 $\pm 300N$ 으로 제한했다. LQR 제어기의 학습법을 적용하기 위해서는 자코비안을 추정해야 하는데 이 경우에는 입력 F 의 증가에 대해 거리 x 는 증가하지만 줄의 각도 θ 는 감소한다는 것을 직관적으로 간단히 알 수 있다. 따라서 \dot{x} 와 $\dot{\theta}$ 를

포함한 자코비안의 부호는 다음과 같다.

$$\hat{J} \approx \text{sign}(\mathbf{J}) = [-1 \ -1 \ 1 \ 1]^T \quad (23)$$

LQR 학습 제어기(LQR-LC : LQR Learning Controller)를 구하기 위해 \mathbf{Q} 가중치 행렬을 정해야 하는데 이것은 LQR 에서 \mathbf{Q} 를 정하는 방법과 같다. 먼저, 간단히 $\mathbf{Q} = \text{diag}[1 \ 1 \ 1 \ 1]$ 로 두었을 때에는 학습이 완료된 후, 목표점으로부터 0.2m(20%)의 오버슈트 나타났기 때문에 다음과 같이 각도제어보다는 거리제어에 더 큰 비중을 둔 \mathbf{Q} 를 선정하였고 학습율은 $\eta=0.01$ 로 하였다.

$$\mathbf{Q} = \text{diag}[1 \ 1 \ 8 \ 10] \quad (24)$$

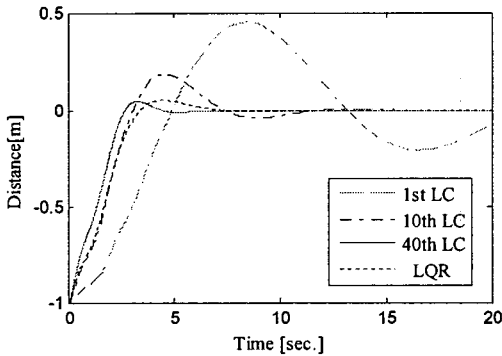


Fig. 3 Comparison of LQR-LC and LQR in Distance

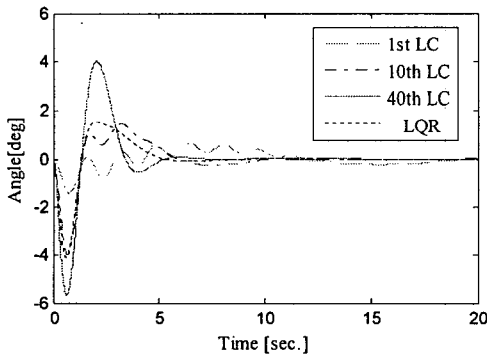


Fig. 4 Comparison of LQR-LC and LQR in Angle

Fig. 3 과 4 에서 파선은 첫 번째 학습이 완료된 후의 제어결과를, 일점 쇄선은 10 번째 학습이 완료된 후, 그리고 실선은 식 (7)의 목적함수가 더

이상 감소하지 않는 40 번째 학습의 결과를 보여준다. LQR-LC 를 통해 최종적으로 구한 제어게인은 다음과 같았다.

$$\mathbf{K} = [-65 \ -65 \ 520 \ 650] \quad (25)$$

다음에는 이 결과를 LQR 제어기와 비교하기 위해 크레인의 위치제어에서 5%의 오버슈트 가지며 입력이 구동기의 제한범위($\pm 300\text{N}$)를 넘지 않도록 MatLab 을 이용하여 설계했다. 따라서 식 (26)와 같은 \mathbf{Q} , \mathbf{R} 가중치에 대해 식 (27)와 같은 제어기를 얻을 수 있었다.

$$\mathbf{Q} = \text{diag}[10 \ 50 \ 10 \ 1], \mathbf{R} = 0.0001 \quad (26)$$

$$\mathbf{K} = [-566 \ -37 \ 316 \ 486] \quad (27)$$

LQR 제어의 결과는 Fig. 3 과 4 에서 점선으로 나타내었다. 학습이 완료된 LQR-LC 와 LQR 의 결과를 비교했을 때, LQR-LC 가 크레인의 진동은 다소 증가했지만 정착시간이 7 초인 LQR 에 비해 훨씬 빠른 5 초의 제어 결과를 얻었음을 알 수 있다.

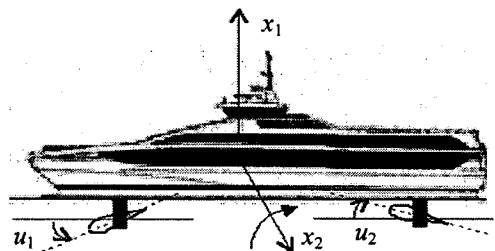
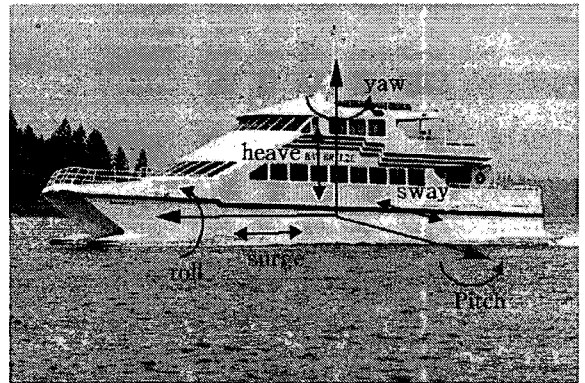


Fig. 5 Hydrofoil catamaran and Schematic diagram

4.2 수중의 쌍동선

수중의 쌍동선은 공기 부양선에 비해 거친 해상 상태에서도 운동 성능이 우수하기 때문에 차세대 고속선으로써 지속적인 연구 개발이 수행되고 있다. 수중의 쌍동선은 2 개의 입력, 2 개의 출력과 각 출력의 속도를 가진 4 개의 상태 변수를 가진 시스템으로 Fig. 5 에서 보는 것처럼 수중의 쌍동선은 선두부의 수중의 입사각 u_1 과 선미부의 수중의 입사각 u_2 를 조정하여 쌍동선의 상하 진동 변위 x_1 와 피치 각도 x_2 를 제어한다.

식(28)은 1m 길이의 모형 수중의 쌍동선의 상태방정식을 나타낸다.^{15,16}

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -109.2 & -4.19 & -0.24 & -5.69 \\ 0 & 0 & 1 & 0 \\ -26.45 & -3.54 & -6.43 & -94.10 \end{bmatrix} \begin{bmatrix} x_1 \\ \dot{x}_1 \\ x_2 \\ \dot{x}_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 1.68 & 1.30 \\ 0 & 0 \\ -5.91 & 7.61 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + \mathbf{D}(\omega) \quad (28)$$

여기서 $\mathbf{D}(\omega)$ 는 바람이나 파도 등의 해상 환경에 따라 발생하는 외란(disturbance)으로 다음과 같이 나타내었다.

$$\mathbf{D}(\omega) = \begin{bmatrix} 0 \\ 0.02 \sin(10t) \\ 0 \\ 0.05 \cos(10t) \end{bmatrix} \quad (29)$$

외란이 있는 시스템에서 자코비안을 구하기는 굉장히 어렵다. 그러나 자코비안의 부호는 외란과 상관없이 출력의 평균값을 이용하여 쉽게 구할 수 있으므로 다음과 같이 추정할 수 있다.

$$\hat{\mathbf{J}} \approx \text{sign}(\mathbf{J}) = \begin{bmatrix} 1 & 1 & -1 & -1 \\ 1 & 1 & 1 & 1 \end{bmatrix}^T \quad (30)$$

만일 입력에 대한 출력의 변화가 작아서

증가했는지 감소했는지 그 부호를 판단하기 어려운 경우에는 그 값은 영으로 두어도 식 (21)의 학습에 대한 수렴성은 보장된다. 이러한 외란에 대해 시스템을 안정화하는 LQR 제어를 학습하고 이를 기존의 LQR 제어기와 비교한다. 그런데 성능을 객관적으로 비교하기 위해 기존의 LQR 제어기는 Lee^{15,16} 가 설계한 제어기를 그대로 사용하였으며 그 값은 다음과 같다.

$$\mathbf{Q} = \text{diag}[500 \ 50 \ 100 \ 10], \mathbf{R} = \begin{bmatrix} 1.0 & 0 \\ 0 & 0.8 \end{bmatrix} \quad (31)$$

$$\mathbf{K} = \begin{bmatrix} 3.06 & 4.76 & -3.16 & -1.68 \\ 4.39 & 3.59 & 3.66 & 2.42 \end{bmatrix} \quad (32)$$

식 (32)의 피드백 게인을 사용하여 제어한 LQR 제어결과는 Fig. 6 과 7 에 나타내었다. 그림에서 보는 바와 같이 점선으로 나타낸 LQR 제어는 배의 중심 위치와 피치각도에서 전반적으로 진동을 줄이긴 했으나 진동이 여전히 나타나고 있음을 알 수 있다. 다음, LQR-LC 에서는 식 (30)의 증감부호를 추정 자코비안으로 사용하고 학습을 $\eta = 0.002$ 과 학습지연 $n=2$ 으로 LQR 에서 사용한 가중치 행렬 \mathbf{Q} 를 그대로 사용하여 학습하였다. 샘플링 시간을 0.01 초로 두고 10 초단위로 학습을 반복한 결과 50 회 만에 학습이 완료되었고 LQR-LC 에서 구한 제어기 게인은 다음과 같은데 LQR 제어기에 비해 그 값이 상당히 크다는 것을 알 수 있다.

$$\mathbf{K} = \begin{bmatrix} 382 & 38.2 & -76.5 & -7.65 \\ 382 & 38.2 & 76.5 & 7.65 \end{bmatrix} \quad (33)$$

이 제어기를 사용하여 제어한 결과는 Fig. 6 과 7 에서 실선으로 나타내었다. 여기서 LQR 제어 결과와는 달리 LQR-LC 에서는 정상상태에서의 진동이 크게 줄었는데 그 이유는 먼저 모델링 식을 이용하는 LQR 에서는 모델링하기 어려운 외란항이 고려되지 못한 반면 실제 시스템을

사용하는 LQR-LC 에서는 고려되었다는 점이고 다음은 LQR-LC 에서는 포화입력(saturated input)의 작용이 가능하다는 것이다. LQR 제어에서는 가중치 행렬 R 을 이용하여 최대입력을 초과하지 않는 제어 계인을 찾아내기 때문에 포화입력이 나타날 수 없으나 LQR-LC 에서는 제어입력이 최대입력 범위를 벗어나도 실제 구동기에서 최대 입력으로 제한하기 때문에 포화입력으로 나타나는데 이러한 경우 제어성능이 더 좋은 것으로 알려져 있다.¹⁷

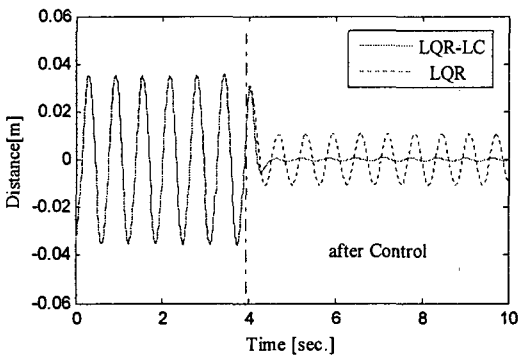


Fig. 6 Vertical distance of LQR-LC and LQR

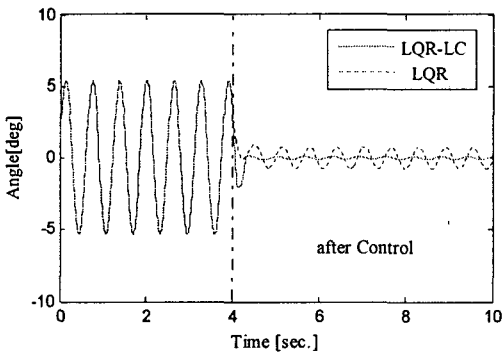


Fig. 7 Pitch angle of LQR-LC and LQR

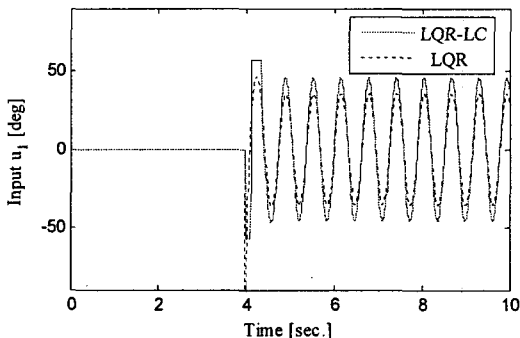


Fig. 8 Input u_1 comparison between LQR-LC and LQR

Fig. 8 과 9 에서는 LQR 과 LQR-LC 에서의 제어 입력들을 서로 비교해 보았다. LQR 에서는 제어 초기에 u_1 에서 순간적으로 피크가 생기는 것을 제외하고는 최대입력을 벗어나지 않지만 LQR-LC 에서는 식 (33)에서 보는 바와 같이 큰 제어 계인으로 인해 u_1 과 u_2 양쪽에서 제어 초기에 상당한 시간 동안 실제로 포화입력이 나타나고 있음을 알 수 있다.

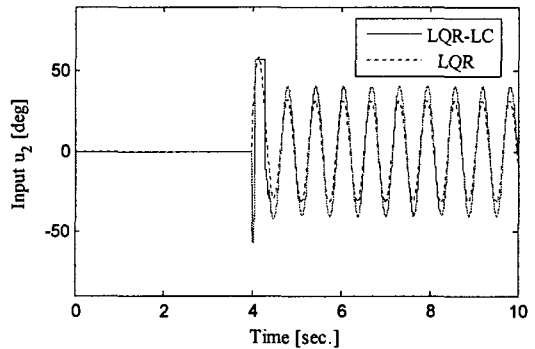


Fig. 9 Input u_2 comparison between LQR-LC and LQR

5. 결론

본 논문에서는 산업 현장과 같이 모델식을 구하기 어려운 경우에 실시간 학습을 통하여 최적의 LQR 제어를 설계하는 학습법을 제안하였다. 다변수 시스템에 대한 LQR 제어를 학습하기 위해서는 시스템의 자코비안 정보를 알아야 하나 실험적으로 시스템의 자코비안을 구하는 것도 간단한 일은 아니므로 자코비안의 부호 정보를 이용하는 방법을 제시하였다. 그리고 이 방법을 크레인과 수중익 쌍동선의 제어에서 적용한 결과 모델식을 알고 구한 LQR 제어기보다 더 좋은 결과를 얻었다. 이것은 LQR-LC 가 포화입력을 이용하였고 LQR 이 고려하지 못한 비선형 성분을 고려한 때문이라고 볼 수 있다. 궁극적으로 LQR-LC 는 모델식 때문에 LQR 제어기를 사용할 수 없었던 복잡한 시스템의 제어에 크게 도움이 되리라고 본다.

참고문헌

1. Maciejowski, J. M., "Multivariable feedback design," Addison-Wesley Inc., 1989.

2. Franklin, G. F., Powell, J. D. and Emami-Naeini, A., "Feedback Control of Dynamic Systems," Addison-Wesley Inc., 2002.
3. Dorf, R. C. and Bishop, R. H., "Modern Control Systems," Addison-Wesley Inc., 1995.
4. Zhang, Y. M. and Kovacevic, R., "Neurofuzzy model-based predictive control of weld fusion zone geometry," IEEE Trans. on Fuzzy Systems, Vol. 6, No. 3, pp. 389-401, 1998.
5. Narendra, K. S. and Parthasarathy, K., "Identification and control of dynamical systems using neural networks," IEEE Trans. Neural Networks, Vol. 1, pp. 4-26, 1990.
6. Sanner, R. M. and Slotine, J. E., "Gaussian networks for direct adaptive control," IEEE Trans. Neural Networks, Vol. 3, pp. 837-863, 1992.
7. Chen, F. C., "Back-propagation neural networks for nonlinear self-tuning adaptive control," IEEE Control System Magazine(Special issue on Neural Networks for Control Systems), Vol. 10, pp. 44-48, 1990.
8. Chen, F. C. and Khalil, H. K., "Adaptive control of nonlinear systems using neural networks," Int. J. of Control, Vol. 55, pp. 1299-1317, 1992.
9. Lim, Y. K. and Chung, B. M., "A learning method of LQR controller using Jacobian," J. of the KSPE, Vol. 22, No. 8, pp. 34-41, 2005.
10. Chung, B. M., "Control of nonlinear multivariable Systems using direct fuzzy learning method," Int. J. of Intelligent & Fuzzy Systems, Vol. 5, pp. 297-310, 1998.
11. Lim, Y. K. and Chung, B. M., "PID learning method using gradient approach for optimal control," J. of the KSPE, Vol. 18, No. 1, pp. 180-186, 2001.
12. Lim, Y. K., Chung, B. M. and Cho, C. S., "Optimal neural network controller design using Jacobian," J. of the KSPE, Vol. 20, No. 2, pp. 85-93, 2002.
13. Lim, Y. K. and Chung, B. M., "A learning method of PID controller by Jacobian in multi variable system," J. of the KSPE, Vol. 20, No. 2, pp. 112-119, 2003.
14. Zhang, F., "Matrix Theory : Basic Results and Techniques," Springer-Verlag New York Inc., 1999.
15. Lee, S. Y., "Theoretical and experimental study on the attitude control system of foil-catamaran," Ph. D. thesis, Seoul National University, Korea, 1999.
16. Lee, S. Y. and Rhee, K. P., "Design of ship-motion regulators for foil catamarans in irregular sea waves," IEEE J. of Oceanic Engineering, Vol. 27, No. 3, pp. 738-752, 2002.
17. Abu-Khalaf, M. and Lewis, F. L., "Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach," Automatica, Vol. 41, pp. 779-791, 2005.