
지연 예측신경망을 이용한 적응 GPC

정희태

이 논문은 2002년 부산외국어대학교 학술연구조성비에 의하여 연구되었음

요약

기존의 GPC방법으로 제어하기 힘든 비선형성과 플랜트의 변수변화를 포함하는 비선형 플랜트를 지연 예측신경망을 사용하여 효과적으로 제어하는 적응 GPC방법을 제안한다. 제안한 방법에서는 플랜트의 선형 변수 추정이나 근사적인 모델로부터 선형 매개변수를 구해서 선형 모델을 만들고 실제 시스템의 출력과 선형 모델의 오차를 신경망의 출력으로 표현한 다음, 이 식으로부터 적응 GPC 알고리즘을 유도한다. 여기서 지연 예측신경망은 적응 GPC에 이용될 플랜트의 출력을 예측하도록 학습된다. 이와 같은 제어를 구성함으로써 선형 변수만으로 적응 GPC 제어가 구성될 경우 생기는 비선형 변수의 추정과 출력 예측 값을 계산하는 번거로움을 해결하였다.

1. 서론

일반적으로 플랜트의 입출력 관계가 수식으로 명쾌히 표현되는 경우에는 수식을 바탕으로 설계 목적에 따라 제어를 구성할 수 있다. 그러나 제어 대상 플랜트의 수학적 모델이 잘 알려져 있지 않거나, 플랜트를 구성하는 변수가 동작 조건에 따라서 변할 경우 제어를 설계하는 것은 쉽지 않다. 적응제어, 자기동조 제어, 적응 GPC(Generalized Predictive Control)방법 등[1-4]은 이와 같은 복잡한 제어 문제를 해결하는데 성공적으로 이용되어 왔다. 특히 GPC방법은 선형 모델변수 추정과 플랜트 출력 예측을 통하여 공정제어 등에 효과적으로 이용되고 있다. 하지만 GPC방법은 선형 모델에 기초를 둔 제어 알고리즘이기 때문에 플랜트의 비선형 특성으로 인한 모델링 오차나, 플랜트의 구조적 불확실성에 의한 제어성능의 저하를 항상 수반한다. 최근에는 이 문제를 인공 신경망을 도입하여 해결하려는 많은 연구가 진행되어 왔다.[5-6] 신경망은 입출력 변수간의 비선형 관계를 표현할 수 있을 뿐만 아니라, 학습을 통하여 신경망을 구성하는 변수를 조정함으로써 환경

이나 플랜트의 변화에 적용할 수 있기 때문에 효과적으로 복잡하고 비선형 특성을 가지는 플랜트의 제어에 이용할 수 있다[7-13]. 신경망을 이용한 제어방법에는 여러 가지가 있지만 대체로 플랜트를 인식하고 제어하는데 신경망만을[14] 사용하는 방법과 신경망과 선형 모델을 함께 사용하는 방법이 있다. 신경망만을 사용하는 방법은 신경망의 학습범위가 작을수록 효과적이기 때문에 주로 플랜트의 동작점이 작은 영역에 제한되어 있는 플랜트에 적용되고, 신경망을 제어로 사용하기 위해서는 사전에 충분한 학습이 이루어져야 한다. 한편 신경망과 선형 모델을 같이 사용하는 경우에는 신경망이 선형 플랜트에 존재하는 비선형 모델 오차나 플랜트의 비선형 불확실성을 학습하여 이를 신경망 제어기에서 보상하여 플랜트의 제어 성능을 개선하는 보조 제어기 역할을 한다. 이 경우에는 플랜트가 선형 제어기에 의해서 어느 정도 제어가 되기 때문에 신경망의 학습은 좁은 영역에 국한되고, 제한된 입출력 정보를 이용하므로 온라인 학습이 가능하게 되며 충분한 초기학습을 필요로 하지 않는다.

*부산외국어대학교 정보과학대학 디지털정보공학부 ** 부산외국어대학교 정보과학대학 디지털정보공학부
접수일자 : 2003. 7. 3

본 논문에서는 기존의 GPC방법으로 제어하기 힘든 비선형성과 플랜트의 변수변화를 포함하는 비선형 플랜트를 지연 예측 신경망을 사용하여 효과적으로 제어하는 적응 GPC방법을 제안한다. 제안한 방법에서는 플랜트의 선형 변수 추정이나 근사적인 모델로부터 선형 매개변수를 구해서 선형 모델을 만들고 실제 시스템의 출력과 선형 모델의 오차를 신경망의 출력으로 표현한 다음, 이 식으로부터 적응 GPC 알고리즘을 유도한다. 여기서 지연 예측신경망은 적응 GPC에 이용될 플랜트의 출력을 예측하도록 학습된다. 이와 같은 제어기를 구성함으로써 선형 변수만으로 적응 GPC 제어가 구성되어질 경우 생기는 비선형 변수의 추정과 출력 예측 값을 계산하는 번거로움을 해결하였다. 제안한 방법을 제한적인 비선형 요소를 포함하고 플랜트의 변수가 환경 변화에 대해서 변하는 플랜트에 적용하여 제안한 방법의 타당성을 확인한다.

II. 적응 GPC 제어 알고리즘

선형 플랜트에 비선형 요소에 의한 모델 오차가 포함된 일반적인 플랜트는 ARIMAX(Auto-Regressive Integrated Moving Average exogenous input)모델로[1] 아래와 같이 표현된다.

$$A(q^{-1})y_t = B(q^{-1})u_{t-1} + e_{t-1} + \frac{C(q^{-1})}{\Delta} \xi_t \quad (1)$$

여기서 u_t, y_t 는 플랜트의 입력과 출력 변수이고, ξ_t 는 플랜트의 외란이며 $A(q^{-1}), B(q^{-1})$ 그리고 $C(q^{-1})$ 는 각각

$$A(q^{-1}) = 1 + a_1q^{-1} + \dots + a_nq^{-n},$$

$$B(q^{-1}) = b_0 + b_1q^{-1} + \dots + b_mq^{-m},$$

$C(q^{-1}) = 1 + c_1q^{-1} + \dots + c_nq^{-n}$ 를 만족하는 지연 연산자(q^{-1})의 다항식이다. 또한 m, n 는 플랜트의 입력 및 출력변수의 지연 차수로서 $n \geq m$ 을 만족한다. Δ 는 미분연산자($\Delta = 1 - q^{-1}$)이고, e_{t-1} 는 플

랜트의 모델 오차로서

$$e_{t-1} = f_{t-1}(y_{t-1}, y_{t-2}, \dots, y_{t-n}, u_{t-1}, u_{t-2}, \dots, u_{t-m}) \quad (2)$$

과 같이 정의하였다. 식 (2)는 지연 예측신경망으로 구성이 되어 플랜트의 모델 오차 출력을 예측하게 되며 적응 GPC 제어 구조에서 제어변수로 사용된다. 식 (1)의 플랜트에서 제한된 입력으로 목표신호를 최적으로 추종하기 위한 목적함수들

$$J(u, t) = E \left\{ \sum_{j=N_1}^{N_2} (y_{t+j} - r_{t+j})^2 + \lambda \sum_{j=1}^{N_2} (\Delta u_{t+j-1})^2 \right\} \quad (3)$$

subject to $\Delta u_{t+j} = 0, \quad j = N_u, \dots, N_2$

과 같이 정의한다. 여기서 $E(\cdot)$ 는 확률적 기대 값을 나타내며, N_1, N_2 는 각각 플랜트가 추종해야 할 목표신호의 최소 및 최대 시간지표이다. 또한 N_u 는 목적함수를 최소화 하기 위한 제어 입력을 발생시키는 시간지표이고, λ 는 입력 제한하기 위한 상수이다. t 시간에서 식 (3)을 최소화하는 제어 입력은 $\Delta u_t, u_{t+1}, \dots, \Delta u_{t-N_u-1}$ 로 구해지지만 실제 제어 입력은 Δu_t 만 이용하고 $t+1$ 시간에서 같은 방법으로 Δu_{t+1} 을 사용하게 된다.

목적함수를 최소화 하는 제어 입력을 구하기 위해 식 (1)을 j -단계 예측식으로 나타내면

$$y_{t+j} = \frac{B}{A} u_{t+j-1} + \frac{1}{A} e_{t+j-1} + E_j \xi_{t+j} + \frac{F_j}{A\Delta} \xi_t \quad (4)$$

와 같다. 여기서 E_j, F_j 는 Diophantine 방정식 $C = E_j(q^{-1})A\Delta + q^{-j}F_j(q^{-1})$ 을 만족한다. 식 (1)을 ξ_t 에 대해서 풀어서 식 (4)에 대입하여 정리하면

$$\begin{aligned}
 y_{t+j} &= \frac{F_j}{C} y_t + \frac{B}{A} u_{t+j-1} \\
 &\quad - \frac{F_j B}{C} u_{t-1} + \frac{1}{A} e_{t+j-1} \\
 &\quad + \frac{A}{C} e_{t-1} + E_j \xi_{t+j} \\
 &= \frac{F_j}{C} y_t + \frac{E_j B}{C} \Delta u_{t+j-1} \\
 &\quad + E_j \xi_{t+j} + \frac{E_j}{C} \Delta e_{t+j-1}
 \end{aligned} \tag{5}$$

이 되고, 최소 자승문제로 취급하여 식 (5)의 ξ_{t+j} 을 영으로 두면 식 (5)는 미래의 입력과 플랜트의 지연 출력으로

$$\hat{y}_{t+j} = \frac{F_j}{C} y_t + \frac{E_j B}{C} \Delta u_{t+j-1} + \frac{E_j}{C} \Delta e_{t+j-1} \tag{6}$$

과 같이 확정적 플랜트 예측 방정식으로 나타낼 수 있다. 다시 식 (6)을 과거와 미래의 제어 입력으로 표현하면 식 (7)과 같다. 여기서 식 (7)의 G_j, Γ_j 는 Diophantine 방정식 $E_j B = G_j C + q^{-j} \Gamma_j$ 을 만족한다.

$$\begin{aligned}
 \hat{y}_{t+j} &= G_j \Delta u_{t+j-1} \\
 &\quad + \frac{\Gamma_j}{C} \Delta u_{t-1} + \frac{F_j}{C} y_t + \frac{E_j}{C} \Delta e_{t+j-1}
 \end{aligned} \tag{7}$$

식 (7)에서 모델 오차에 관한 다항식은 식 (2)로부터 신경망 함수로

$$\begin{aligned}
 C^{-1} E_j \Delta e_{t+j-1} &\approx \hat{e}_{t+j-1} \\
 &= NN(y_{t+j}, y_{t+j-1}, \dots, y_{t+j-n}, \\
 &\quad u_{t+j-1}, u_{t+j-2}, \dots, u_{t+j-m}, W_j)
 \end{aligned} \tag{8}$$

와 같이 정의 할 수 있다. 여기서 w_j 는 지연 예측신경망의 가중치이다. 식 (8)을 식 (7)에 대입하여 정리하면

$$\begin{aligned}
 \hat{y}_{t+j} &\approx G_j \Delta u_{t+j-1} + \frac{\Gamma_j}{C} \Delta u_{t-1} \\
 &\quad + \frac{F_j}{C} y_t + \hat{e}_{t+j-1} \\
 &= G_j \Delta u_{t+j-1} + \hat{y}_{t+j|t}
 \end{aligned} \tag{9}$$

와 같이 쓸 수 있다. 식 (9)의 두 번째 항은 선형 모델 출력과 지연 예측신경망의 출력으로

$$\hat{y}_{t+j|t} = \Gamma_j(q^{-1})u'_{t-1} + F_j(q^{-1})y'_t + \hat{e}_{t+j-1} \tag{10}$$

와 같다. 여기서 $u'_t = C^{-1}(q^{-1})\Delta u_t$, $y'_t = C^{-1}(q^{-1})y_t$ 이다. 플랜트의 모델이 식 (9)와 같이 선형 모델식과 지연 예측신경망으로 표현되면, 식 (3)으로부터 최적 제어입력을 구하기 위해

$$\begin{aligned}
 \hat{\mathbf{y}} &= [\hat{y}_{t+1} \hat{y}_{t+2} \dots \hat{y}_{t+N_t}]^T, \\
 \bar{\mathbf{u}} &= [\Delta u_t \Delta u_{t+1} \dots \Delta u_{t+N_t-1}]^T, \\
 \bar{\mathbf{d}} &= [\hat{y}_{t+1|t} \hat{y}_{t+2|t} \dots \hat{y}_{t+N_t|t}]^T
 \end{aligned}$$

및 $\mathbf{r} = [r_{t+1} r_{t+2} \dots r_{t+N_t}]^T$ 로 두면 식 (9)는

$$\hat{\mathbf{y}} = \mathbf{G} \bar{\mathbf{u}} + \bar{\mathbf{d}} \tag{11}$$

와 같이 벡터함수로 간단히 된다. 또 식 (11)을 식 (3)에 대입하면 식 (3)은

$$\begin{aligned}
 J(\mathbf{u}, t) &\approx E \left\{ \sum_{j=1}^{N_t} (\hat{y}_{t+j} - r_{t+j})^2 \right. \\
 &\quad \left. + \lambda \sum_{j=1}^{N_t} (\Delta u_{t+j-1})^2 \right\} \\
 &= (\hat{\mathbf{y}} - \mathbf{r})^T (\hat{\mathbf{y}} - \mathbf{r}) + \lambda \bar{\mathbf{u}}^T \bar{\mathbf{u}}
 \end{aligned} \tag{12}$$

와 같이 정리된다. 식 (11)과 (12)로부터 최적 제어 입력을 구하면

$$\bar{\mathbf{u}} = (\mathbf{G}^T \mathbf{G} + \lambda \mathbf{I})^{-1} \mathbf{G}^T (\mathbf{r} - \bar{\mathbf{d}}) \tag{13}$$

와 같이 적응 GPC 제어법칙으로 유도된다.

III. 지연 예측신경망의 학습 및 적응 GPC 제어구조

3.1 지연 예측신경망의 학습

지연 예측신경망을 학습하기 위하여 선형 모델의 출력과 실제 플랜트의 출력을 표시하면

$$e_{t+j-1} = y_{t+j-1} - y_{t+j-1}^L \quad (14)$$

와 같다. 여기서 y_{t+j-1}^L 는 선형 모델의 출력이다. 여기서 지연 예측신경망은 t 시간 이전의 플랜트 입력 및 출력 값으로 $t+N_1$ 에서 $t+N_2$ 사이의 모델오차를 예측하여 적응 GPC 제어 구조의 제어 변수로 사용한다. 학습은 $t-N_2$ 이전의 플랜트 입출력 값으로 t 시간 사이의 지연모델 오차를 학습한다. 따라서 지연 예측신경망은 과거 $t-N_2$ 시간의 모델오차를 지연으로 학습하고 미래의 $t+N_2$ 시간까지의 모델 오차를 예측하는 기능을 가진다. 제안한 지연 예측신경망의 두 가지 기능을 그림으로 나타내면 그림 1 및 2와 같다.

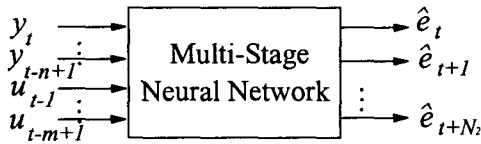


그림 5. $t+N_2$ 시간까지의 모델 오차를 예측.

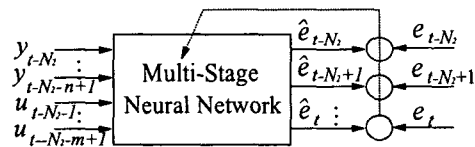


그림 6. $t-N_2$ 이전의 입출력 값으로 모델 오차를 학습.

3.2 적응 GPC 제어구조

지연 예측신경망을 이용한 제어 구조는 플랜트의 선형 모델과 모델 오차를 지연으로 예측 및 학습하는 지연 예측신경망 그리고 지연 예측신경망의 출력과 플랜트의 목표치로 구성되는 적응 GPC 제어기로 구성이 된다. 특히 적응 GPC 제어기에서는 선형 모델에 의한 선형 제어 입력과 지연 예측신경망에 의한 비선형 제어 입력을 만들어 낸다. 제안한 방법의 제어 구조는 그림 3과 같다.

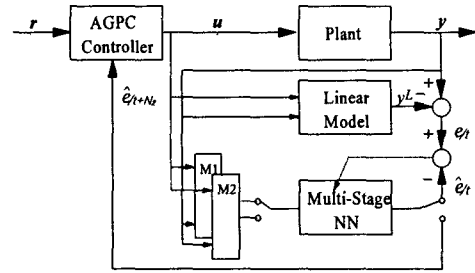


그림 3. 지연 예측신경망에 의한 적응 GPC 제어구조.

IV. 시뮬레이션 및 고찰

제안한 방법을 아래와 같은 이산시간 플랜트에 [1] 적용하여 기존의 신경망을 이용한 간접제어 방법과 비교하였다.

$$y_{t+1} = 1.7y_t - 0.7y_{t-1} + 0.9\Delta u_t - 0.6\Delta u_{t-1} + e_t + \xi_{t+1}$$

여기서 $N_1=1$, $N_2=3$, $N_u=3$ 그리고 $\lambda=0.1$ 로 두고 제안한 방법에서는 오프라인 학습 없는 신경망을 사용하였고, 간접 제어 방법에서 1.0에서 -1.0의 랜덤신호를 발생시켜 제어 및 인식 신경망을 12000번 학습시킨 후에 연속적인 사인파와 구형파를 추종하게 하였다. 그리고 목표신호 추종과정에서 $t=125$ 에서 플랜트의 모델 변수 변화와 비선형 외란을 첨가하여 각 제어 특성을 살펴 보았다.

$$e_t = \begin{cases} 0.0 & 0 \leq t < 125 \\ 0.3 \sin(y_t) - 0.1 \cos(y_t + y_{t-1}) + 0.2u_t + 0.2y_{t-1} & t \geq 125 \end{cases}$$

그림 3-(a)는 목표신호와 인식 및 제어 신경망을 이용한 플랜트의 출력과 제안한 방법에 의한 플랜트 출력이다. 이 결과로부터 신경망만을 사용한 제어에서는 플랜트에 외란이 인가되고 모델 변수가 변했을 때, 플랜트의 순간 적응성은 좋으나, 다른 값의 목표신호에 대해서는 신경망의 가중치들이 플랜트의 변화에 대해서 대부분 재학습됨으로 큰 오버슈트가 발생하고 안정되는 시간이 길어짐을 볼 수 있다. 제안한 방법에서는 플랜트의 변

화에 대해서 순간 천이 특성은 떨어지나 큰 진동 없이 빨리 안정되게 목표신호를 추종하고 있음을 볼 수 있다.

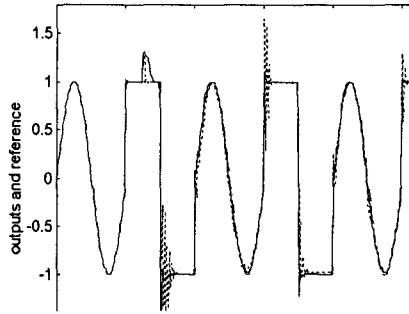


그림 4. 목표신호 및 출력신호.

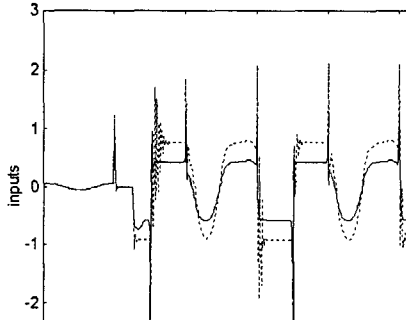


그림 5. 플랜트의 입력신호.

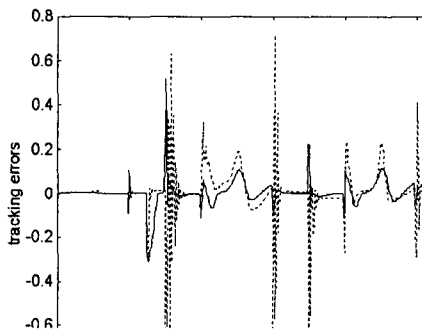


그림 6. 목표치 추종 오차신호.

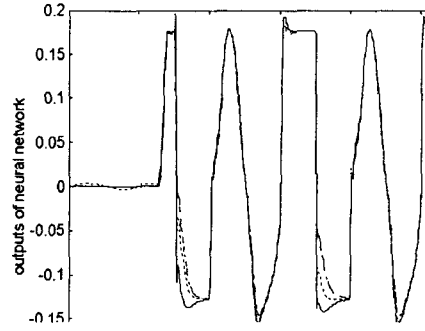


그림 7. 지연 예측신경망의 출력신호.

V. 결론

본 논문에서는 적응 GPC 제어방법과 지연 예측신경망을 이용하여 비선형 외란 및 선형 모델 변수 변화에 적응하는 제어방법을 제안하였다. 또 지연 예측신경망은 모델 오차의 학습 기능과 모델 오차 예측 기능을 수행하도록 두 가지의 입출력 지연 정보를 이용하였다. 이렇게 함으로써 기존의 GPC 방법을 좀더 다양한 플랜트의 환경 변화에 효과적으로 대응하게 하였고, 지연 예측신경망의 학습 또한 기존의 간접 제어방법과 비교하여 간단히 하였다. 앞으로의 연구과제는 신경망 변수를 포함하는 전체 플랜트의 안정성을 분석하는 것이다.

참고 문헌

- [1] R. R. Bitmead, M. Gevers and V. Wertz, "Adaptive optimal control," Prentice Hall, 1990.
- [2] D. S. Shook, C. Mohtadi and S. L. Shah, "Identification for long-range predictive control," IEE Proceedings-D, vol. 138, no. 1, Jan. 1991.
- [3] J. J. E. Slotine and W. Li, Applied Nonlinear Control. Englewood Cliffs, NJ: Prentice-Hall, 1991.
- [4] S. S. Sastry and A. Isidori, "Adaptive control of linearizable systems," IEEE Trans. Automat. Contr., vol. 34, pp. 1123

-1131, 1989.

- [5] Moonyong Lee and Sunwon Park, "A new combining neural feedforward control with model-predictive control," *AIChE Journal*, vol. 38, no. 2, Feb. 1992.
- [6] A. Draeger, S. Engell and H. Ranke, "Model predictive control using neural networks," *IEEE Control Systems*, pp. 61-66, oct. 1995.
- [7] D. E. Rumelhart and J. L. McClelland, *Parallel Distributed Processing*. Cambridge, MA: M.I.T. Press, 1986.
- [8] N. V. Bhat, P. A. Mindelman, T. McAvoy, and N. S. Wang, "Modeling chemical process system via neural computation," *IEEE Control System Mag.*, vol. 10, pp. 24-31, 1990.
- [9] D. Psaltis, A. Sideris, and A. A. Yamamura, "A multilayered neural network controller," *IEEE Control System Mag.*, pp. 17-21, Apr., 1988.
- [10] L. Jin, P. N. Nikiforuk, and M. M. Gupta, "Adaptive control of discrete-time nonlinear systems using recurrent neural networks," *IEE proc. D*, vol. 141, No.3, May, 1994.
- [11] R. M. Sanner and J. E. Slotine, "gaussian networks for direct adaptive control," *IEEE Trans. Neural Networks*, vol. 3, No. 6, pp. 837-863, 1992.
- [12] J. J. Hofield and D. W. Tank, "Neural computation of decisions in optimization problems," *Bio. Cybern.*, vol. 52, pp. 141-152, 1985.
- [13] K. S. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural network," *IEEE Trans. Neural Networks*, vol. 1, pp. 4-27, 1990.
- [14] D. H. Nguyen and B. Widrow, "Neural networks for self-learning control system," *IEEE Control System Mag.*, vol., 56, aug., 1992.

저자 소개



정희태

1964년 2월 1일생. 1986년 경북대학교 전자공학과 졸업.

1988년 동 대학원 전자공학과 졸업(공학석사).

1996년 동 대학원 공학박사.

1996년-1997년 특허청 심사관.

1997년-현재 부산외국어대학교 디지털정보공학부 부교수.

※ 관심분야 : Web 기반의 원격제어, 퍼지 시스템, 인공 신경회로망을 이용한 제어, 지능형 네트워크 제어등