

시스템 제어를 위한 지식베이스의 구축

김도성, 이명호

연세대학교 전기공학과

A Construction of the Knowledge Base for System Regulation

Dosung Kim, Myoung-Ho Lee

Dept. of Electrical Engineering, Yonsei University, Seoul, Korea

Abstract

In this paper, an algorithm is implemented to retrieve the control knowledge from the plant being controlled. And this knowledge is stored to the knowledge base and is continuously modified. A control system which realizes this algorithm generates control knowledge automatically and modifies the knowledge base, which was previously generated, in accordance with the experience of input-output relations. And this kind of system can manipulate knowledge by symbolic descriptions.

So this system can be used to implement the heuristic procedure which was difficult to realize through conventional procedural computer languages or numerical techniques.

1. 서론

종래의 digital 제어나 적응제어 algorithm들은 플랜트의 출력과 기준 출력과를 비교함으로써 이 오차를 플랜트 매개 변수 및 제어 매개변수 추정에 사용하고 있다. 이러한 접근법이 갖는 문제는 수학적 입.출력 관계의 모델링의 어려움과 더불어 이들 모델의 복잡성으로 인한 시간지연이다.

또한 PID 제어 algorithm은 간단함과 함께 효율성이 뛰어난 장점을 갖고 있으나 시스템의 비선형성을 극복하는 것과 다양한 제어 목표-정확도나 속도등-를 달성하는 데는 어려움이 따른다. 이러한 문제점을 극복하기 위하여 인간의 결정과정을 제어 시스템에 도입하는 방법이 다양하게 시도되어왔다.

Zadeh[9]가 제시한 Fuzzy set 이론은 인간 의사결정에 있어서의 언어적 개념을 수치화할 수 있는 변환을 제공한다. 언어적인 제어규칙으로써 구성되는 경험적인 지식을 시스템의 제어에 응용하기 위해서는 우선 특정 시스템의 제어 전문가로부터나 data 베이스로부터 대상 플랜트에 대한 정보를 알아야 할 필요가 있다. Heuristics를 사용하는 대부분의 제어기는 이러한 지식에 기초하여 설계된다. 그러나 복잡상황에서 어떤 제어 행동을 취해야할지 기술되어 있지 않은 경우 및 제어지식의 최적성이 불확실할 때에는 수치적 정보를 사용하여 운전자의 제어행동을 모델링 하여야 할 필요가 있다. Procyk와 Mamdani[10]는 여러가지 플랜트에 fuzzy set 이론을 적용하여 제어규칙을 생성하는 예를 보였다. 또한 Li와 Lau[6]는 servo motor의 위치제어에 fuzzy algorithm의 적용가능성을 보였다. 본 논문에서는 제어규칙의 on-line 생성 algorithm을 제시하고 결구된 규칙 베이스의 신뢰성에 관하여 검증하고자한다.

2. 지식 기반 제어기의 구성

2.1 fuzzy set과 제어 규칙

지식 베이스의 각 제어규칙은 플랜트의 오차 E (error = output-set)와 출력의 변화분 C(change = y(n)-y(n-1)) 및 이때 플랜트에 적용할 입력증분 U를 기술

한다. 여기서 E, C, U는 유한이산요소(finite discrete element)의 집합으로 구성된다고 가정한다.

즉 $E = \{e\}, C = \{c\}, U = \{u\}$

각 유한이산집합 E, C, U는 또한 언어적인 묘사가 가능한 subset으로 분할된다.

$$\begin{aligned} E &= \{E1, E2, \dots, E1\} \\ C &= \{C1, C2, \dots, Cn\} \\ U &= \{U1, U2, \dots, Un\} \end{aligned} \tag{1}$$

또한 각 fuzzy subset은 다음과 같이 정의된다.

$$\begin{aligned} Ek &= \{(e, \mu_{Ek}(e))\} \subset E \\ Ck &= \{(c, \mu_{Ck}(c))\} \subset C \\ Uk &= \{(u, \mu_{Uk}(u))\} \subset U \end{aligned} \tag{2}$$

e, c, u는 이산집합의 단일요소들이며 μ 는 fuzzy membership function으로서 fuzzy subset의 요소들에 대한 membership 값을 결정한다. 예를들어 다음과 같이 모집합들을 정의해보자.

$$\begin{aligned} E &= \{(-5, -3), (-4, -2), (-1, 1), \dots\} \\ C &= \{(-4, 0), (-2, 2), (0, 4), \dots\} \\ U &= \{(-3, -1), (-1, 1), (1, 3), \dots\} \end{aligned} \tag{3}$$

여기서 E1 = (-5, -3)을 "Large Negative", C2 = (-2, 2)를 "Zero", U3 = (1, 3)을 "Small Positive"라고 언어적인 정의를 한다면 제어규칙 E1->C2->U3는 "If error is Large Negative then if change is Zero then input change is Small Positive"로 해석된다. 또한 "Large Negative"의 μ 를 다음 Fig.1과 같이 정의하였다면

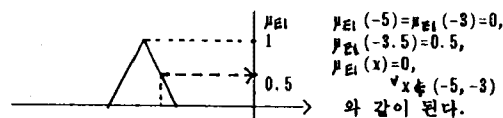


Fig.1 Membership Function

2.2 제어 규칙 생성

plant로부터 측정된 출력 y(n), y(n-1)과 set point y_n 로부터 오차, 출력변화, 입력증분을 해당 fuzzy 변수의 집합 중의 한 요소로 변환하기 위해서 다음과 같은 양자화 과정(quantization)을 사용한다.

$$\begin{aligned} eq &= Q[(y_n - y(n)) * SE] \\ cq &= Q[(y(n) - y(n-1)) * SC] \\ uq &= Q[(u(n) - u(n-1)) * SU] \end{aligned}$$

여기서 SE, SC, SU는 각각 오차(error)와 출력 변화분(output change), 입력증분(input correction)으로부터 fuzzy variable의 영역으로 변환하기 위한 scale factor이다. Q는 다음 table 1과 같이 구성할 수 있다.

여기서 es, cs, us는 실제 data error, change, du 등으로부터 해당 scale factor를 곱하여 얻은 값이며 quantization table의 table 계수 tc를 각 fuzzy subset 요소에 곱하여 quantized level eq, cq, uq를 구하게된다. 이들 eq, cq, uq로부터 fuzzy variable Ek, C1, Um을 형성하기 위해서는 fuzzification 과정이 필요하다.

Table 1 Quantization table

| scaled data es(or cs, us) | quantized level eq(or cq, uq) |
|------------------------------|----------------------------------|
| es <= 1 | sign(es)*0 |
| 1 < es <= 2 | sign(es)*1 |
| 2 < es <= 4 | sign(es)*2 |
| 4 < es <= 6 | sign(es)*3 |
| 6 < es <= 8 | sign(es)*4 |
| es > 8 | sign(es)*5 |

즉 $E_k = F(eq) = \{(eq, \mu_{E_k}(eq))\}$ etc.
 Fuzzification은 다음과 같이 예시할 수 있다.
 양자화된 단일요소 eq에 대해
 $F(eq) = \{(E1, \mu_1), (E2, \mu_2)\}$
 와 같이 fuzzy subset을 구성한다.



Fig. 2 Fuzzification

또한 $C1 = F(cq) = \{(C3, \mu_3), (C4, \mu_4)\}$ 이라 가정하면 $E_k * C1$ 은 $\{(E1 * C3), (E1 * C4), (E2 * C3), (E2 * C4)\}$ 이고 Fuzzy 논리에 따르면 $\mu(E1 * C3) = \min(\mu_1, \mu_3)$ 가 된다. 그러므로 Zadeh의 추론 조합칙(Compositional Rule of Inference) [2]에 의해서 최대 μ 값을 갖는 $E_k * C1$ 의 subset 요소 $E_i * C_j$ 를 key로 하여 규칙 베이스를 탐색한다. 해당 규칙이 발견되면 (예를 들면, $E_i * C_j * \mu$) fuzzy subset U_m 을 결정하고 U_m 으로부터 deterministic한 제어 입력을 결정하기 위해 최대 μ 값을 갖는 u 를 선택한다. 다중의 u 에 대해서는 평균을 취한다.

2.3 성능 평가

앞에서 적용한 입력의 성능을 평가함으로써 새로운 제어 규칙을 생성할 필요가 있다. 2.2에서 결정된 U_m 으로부터 특정한 입력 증분 u 를 결정하기 위해 relation 행렬 [1, 2]을 사용하는 방법이 있으나 행렬의 크기가 방대해지는 결점이 있으므로 eq, cq로부터 요구되는 출력의 고정량을 미리 특정 범위로 정의하는 다음과 같은 성능평가표를 사용한다.

Table 2 Performance Evaluation Table

| | 출력 변화분 (cq) | | | | | | | |
|------------|-------------|----|----|----|----|----|----|----|
| | -3 | -2 | -1 | 0 | 1 | 2 | 3 | |
| ERROR (eq) | -3 | 4 | 4 | 4 | 4 | 3 | 2 | 2 |
| | -2 | 3 | 2 | 2 | 2 | 0 | 0 | 0 |
| | -1 | 2 | 1 | 1 | 1 | 0 | 0 | 0 |
| | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| | +0 | 0 | 0 | 0 | 0 | 0 | 0 | -1 |
| | 1 | 0 | 0 | 0 | -1 | -1 | -1 | -2 |
| | 2 | 0 | 0 | 0 | -2 | -2 | -2 | -3 |
| | 3 | -2 | -2 | -3 | -4 | -4 | -4 | -4 |

이때 cq는 $change=Y(n)-Y(n-1)$, eq는 $error=Y(n)-Y_m$ 의 양자와 같다.

성능평가표로부터 결정된 출력 고정량을 위해서는 이에 소요되는 입력 고정량을 결정해야 한다. SISO의 경우 $DY = M_0 DU$ 와 같이 입력 증분 DU에 대해 출력 변화 DY를 일으키는 증분 model M_0 를 가정할 수 있다. Q table의 eq, cq로부터 P table의 출력 고정량을 결정하므로 원래 model M_0 는 다음과 같이 scale된다.

$$M = M_0 * SC/SU \quad (4)$$

그러므로 pn 으로부터 입력 증분 rn 을 다음 식에서 구한다.

$$rn = M^{-1} * pn = M_0^{-1} * (SU/SC) * pn \quad (5)$$

$$\text{이때 } M_0 = DY/DU \text{ 이므로 } rn = (DU/DY) * (SU/SC) * pn$$

여기서 결정된 rn 으로부터, 기존의 제어규칙 BASE에 U_m 이 없었다면 새로운 규칙을 다음과 같이 구성할 수 있다.

$$vn = uq + rn \quad (6)$$

$$U_{new} = U_{old} + vn / (tc * SU) \quad (7)$$

현재의 출력 data는 1 step 이전의 입력에만 관계된다고 근사적인 가정을 하면 uq 는 이전 step의 양자화된 입력 증분이며 이때 실제 플랜트의 가해지는 입력 U_{new} 는 vn 으로부터 실제 data로 역변환된 값을 이전 입력 U_{old} 에 더하여 얻어진다. 이상에서 새로운 규칙은 다음과 같이 구성된다.

$$\text{즉 } E_k * C1 * V_m \quad \text{여기서 } V_m = F(vn) \quad (8)$$

만약 입력 U_m 을 적용하여 새로운 V_m 을 추론하였다면 지식 베이스 개정과정 [10]이 필요하게 된다.

3. 컴퓨터 시뮬레이션 및 결과 고찰

3.1 SOFTWARE

실험을 위한 program의 흐름도는 다음과 같다.

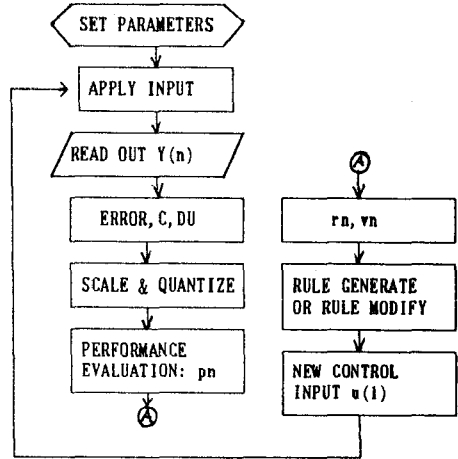


Fig. 3 Flow Chart

eq, cq로부터 pn 을 구하고 rn 을 구하는 program은 C language를 사용하였고 규칙 생성 및 개정은 A.I. 언어인 PROLOG로 program하였다.

3.2 플랜트 선정 및 시뮬레이션

선정한 플랜트의 전달함수는

$$G(s) = 0.05 / (s^2 + 0.6s + 0.05) \quad (9)$$

이산화한 state space system은 다음과 같다.

$$X(k+1) = \begin{bmatrix} \frac{1}{4}(e^{-0.1T} + (5/4)e^{-0.5T} - \frac{1}{2}(e^{-0.1T} - e^{-0.5T})) & \frac{1}{4}(e^{-0.1T} - e^{-0.5T}) \\ \frac{5}{2}(e^{-0.1T} - e^{-0.5T}) & \frac{1}{4}(5e^{-0.1T} - e^{-0.5T}) \end{bmatrix} X(k) + \begin{bmatrix} \frac{1}{4}(e^{-0.1T} - e^{-0.5T}) \\ -10e^{-0.1T} + 2e^{-0.5T} + 8 \end{bmatrix} U(k) \quad (10)$$

$$Y(k) = 0.05 * x_2(k)$$

Sampling time T=1로 하고 $Y_m = 100$ 일때 앞에서 구성한 제어 algorithm을 수행한 결과 제어 입력과 출력을 Fig. 4 - Fig. 8과 같이 구할 수 있었다. Quantization Level은 11단계(-5:5)로 구성하고 이때 $tc=0.5$ 로 선정한다. 성능평가표에서 eq는 (-6:6), cq는 (-6:6) 단계도 구성하였다. 여기서 eq는 출력 zero state를 +0, -0로 구별하여 필요 고정량을 선택한다. 출력 고정량 pn 도 따라서 (-6:6) 단계의 고정량으로 정의된다. 이때 scale factor는 플랜트의 특성에 관계되며 기지의 플랜트 지식으로부터 결정하여야 한다. 실험 1)에서 SE=1, SC=4, SU=10으로 선정하였다. 실험 2)에서는 SE=2, SC=16, SU=8로 하여 rise time의 지연을 관측할 수 있었다. 실험 3)에서는 SE=2, SC=1, SU=8로 하였을때 입력의 큰 진동이 나타난다. 이 현상은 출력변화분 scale factor인 SC를 작게함으로써 출력 변화에 대한 감도의 감소가 원인이 된다. Fig.5와 Fig.7을 비교하면 SU의 증가에 따라 rise time이 증가하고 steady state에서 ripple이 감소함을 관측할 수 있다. 그 이유는 SU의 증가에 따라 입력 증분의 양이 감소하므로 그만큼 정교한 제어가 가능한 반면 rise time은 증가하게 된다. 이상의 관찰에서 transient state에서는 입력 증분을 크게하여 규정출력에 빨리 도달하게 하면서 steady state에서는 증분을 작게하여 정교한 제어를 수행해야 할 필요를 알 수 있다. 여기서 제어관용오차 경계치역(Error Boundary Region for Control Change) CEB를 설정하여 일단 이 영역내에 진입하면 입력 증분을 감소시키는 방법을 택한다. CEB를 규정출력의 0.5%로 하였을때

transient state에서는 SE=2, SC=16, SU=8, 영역 진입 이후에는 SU=16으로 증가시킨 결과는 Fig. 8과 같으며 rise time의 감소 및 ripple의 감소를 관찰할 수 있다.

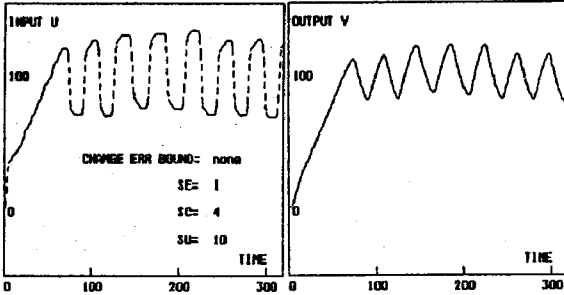


Fig. 4 Control input, plant output in experiment (1)

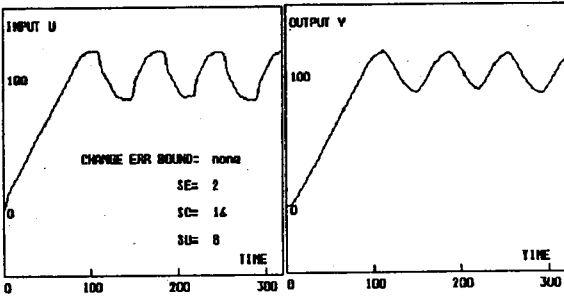


Fig. 5 Control input, plant output in experiment (2)

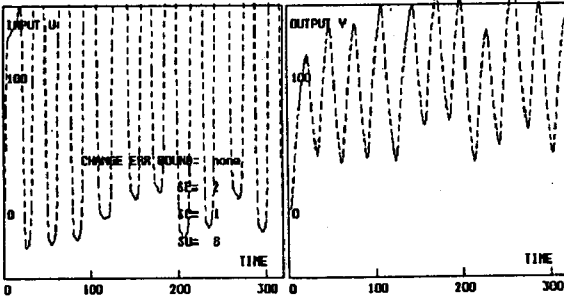


Fig. 6 The effect of decreasing SC in experiment (3)

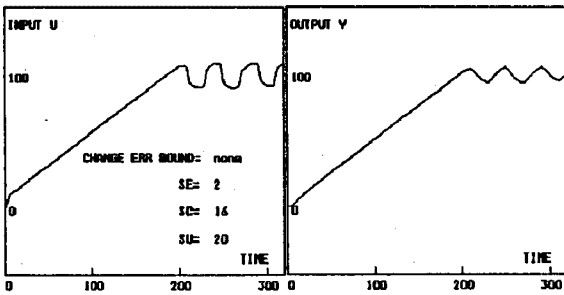


Fig. 7 The effect of increasing SU

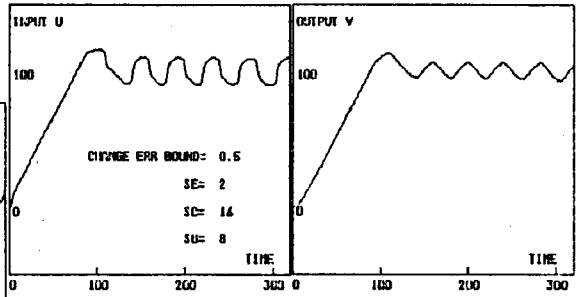


Fig. 8 The effect of using CEB

4. 결론

본 연구에서는 언어적인 제어 규칙으로써 플랜트를 제어할 수 있는 지식기반 제어 시스템의 구성에 관하여 고찰하였다. Fuzzy set 이론을 적용함으로써 heuristic한 언어적인 제어 지식을 시스템 제어에 도입할 수 있다. 본 실험에서는 고정된 기준 출력에 대하여 규칙 생성 및 교정을 행하였으므로 생성된 지식 베이스는 기준 출력에 한정된 것이 된다. 또한 scale factor의 상이한 선정에 따라 응답의 차이가 관찰된다. 이는 제어대상 플랜트의 입력력 변수들의 범위 및 응답 특성에 대한 지식으로부터 scale factor가 결정되어야 함을 나타낸다. 결론적으로 본 연구의 지식 기반 제어기의 장점은 다음과 같이 요약될 수 있다.

- 1) 지식 기반 제어 구조는 새로운 제어 지식의 추가 및 변경, 삭제 등이 용이하다.
- 2) 전문가의 heuristic한 제어 지식으로써 제어기의 초기화가 가능하다.
- 3) 플랜트에 대한 정확한 수학적 model이 불필요하다.

References

- [1] R.M.Tong, "Analysis and Control of Fuzzy systems using finite discrete relations," *Int. J. Control*, vol. 27, No. 3, pp. 431-440, 1978.
- [2] E.H.Mamdani, "Application of Fuzzy Logic to Approximate Reasoning Using Linguistic Synthesis," *IEEE Trans. on Computers*, vol. c-26, No. 12, pp. 1182-1191, December, 1977.
- [3] L.A.Zadeh, "Making Computers Think Like People," *IEEE Spectrum*, pp. 26-32, August 1984.
- [4] R.M.Tong, M.B.Beck, and A.Laten, "Fuzzy control of the Activated Sludge wastewater Treatment Process," *Automatica*, vol. 16., pp. 659-701, 1980.
- [5] J.Walters and Y.S.Sherif, "Applications of Fuzzy Set Theory," *IEEE Trans. on Systems, Man, And Cybernetics*, vol. smc-15, No. 1, pp. 175-189, Jan./Feb., 1985.
- [6] Y.F.Li and C.C.Lau, "Development of Fuzzy Algorithms for Servo systems," *IEEE Control Systems*, vol. 9, No. 3, pp. 65-72, 1989.
- [7] T.Takagi and M.Sugeno, "Fuzzy Identification of Systems and its Application to Modeling and Control," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 15, pp. 116-132, 1985.
- [8] F.Van der Rhee and H.R.van Nauta Lemke, "Applying fuzzy set theory to modeling processes," *IFAC 10th World Congress Automatic Control*, pp. 338-343, 1987.
- [9] L.A.Zadeh, "Fuzzy Sets," *Inf. Control*, 8, pp. 338-353, 1965.
- [10] T.J.Procyk and E.H.Mamdani, "A Linguistic Self-Organizing Process Controller," *Automatica*, vol. 15, pp. 15-30, 1979.