

Profibus-DP를 이용한 네트워크 기반 제어 시스템의 원격 퍼지 제어

Remote Fuzzy Logic Control of Networked Control System Via Profibus-DP

이 경 창, 이 석
(Kyung Chang Lee and Suk Lee)

Abstract : This paper investigates on the feasibility of fuzzy logic control for networked control systems. In order to evaluate its feasibility, a networked control system for motor speed control is implemented on a Profibus-DP network. The NCS consists of several independent, but interacting processes running on two separate stations. By using this NCS, the network-induced delay is analyzed to find the cause and effect of the delay. Furthermore, in order to prove the feasibility, the fuzzy logic controller's performance is compared with those of conventional PID controllers. Based on the experimental results, the fuzzy logic controller can be a viable choice for NCS due to its robustness against parameter uncertainty.

Keywords : networked control system, remote fuzzy logic controller, network-induced delay, fieldbus, profibus-DP

I. 서론

최근 고성능의 마이크로 프로세서가 발전함에 따라 대다수의 제어 시스템은 디지털 제어 시스템으로 대체되고 있다. 이러한 디지털 제어 시스템은 뛰어난 유연성(flexibility)과 적응성(adaptability)으로 인하여, 전통적인 공정 자동화(process automation) 분야뿐만 아니라, 조립 자동화(assembly automation)나 빌딩 자동화(building automation) 등과 같은 분야에서도 활발하게 적용되고 있다. 특히, 보다 향상된 성능을 얻기 위하여 더 많은 수의 센서(sensor)나 구동기(actuator), PLC(Programmable Logic Controller), 산업용 컴퓨터 등이 사용됨에 따라, 디지털 제어 시스템에서 처리되어야 할 데이터량이 급속하게 증가되었다.

그러나, 디지털 제어 시스템에서 중앙 제어기와 센서나 구동기와 같은 필드기기를 직접 연결하는 점대점(point-to-point) 연결 방식은 너무 많은 배선을 필요로 하며, 시스템의 유연성과 신뢰성 확보가 어려울 뿐만 아니라, 시스템의 유지 및 보수가 어렵다. 따라서, 이러한 문제점을 해결하기 위하여, 점대점 연결 방식을 공유된 전송매체로 대체하는 산업용 네트워크(industrial network)에 대한 연구들이 활발하게 진행되고 있다[1].

일반적으로, 산업용 네트워크에서 교환되는 데이터는 실시간 데이터와 비실시간 데이터로 구분되며, 단일한 전송 매체를 통하여 동시에 전송될 수 있어야 한다. 만약, 이러한 데이터들이 제대로 관리되지 못하면, 즉 실시간 데이터가 정해진 시간 한계치 내에 전송되어야 한다는 실시간 요구 조건(real-time requirement)이 만족되지 못하면, 시스템의 성능이 저하될 뿐만 아니라, 치명적인 고장이 발생할 수 있다[2]-[4].

산업 현장에서의 실시간 요구 조건을 만족시키기 위하여, 여러 표준기관들에 의하여 다양한 프로토콜들이 개발되어 왔으며, 1999년 말 널리 사용되던 Profibus와 P-Net, Fieldbus

Foundation, SwiftNet, WorldFIP, Interbus를 포함한 IEC 61158 필드버스가 IEC TC65C WG6에 의하여 국제 표준으로 발표되었다[5].

최근, 필드버스의 보다 폭 넓은 응용을 위하여, Walsh[6], Zhang[7], Halevi[8], Nilsson[9] 등에 의하여 그림 1과 같은 필드버스에 기반한 네트워크 기반 제어 시스템(Networked Control System, NCS)에 대한 연구가 활발하게 진행되고 있다. 그림에서, 구동기나 센서는 필드버스에 연결되어 있으며, 원격지에 위치한 제어기에 의하여 원격 제어가 이루어진다. 또한, 기준 입력(reference input)이나 플랜트 출력(plant output), 제어 입력(control input) 등과 같은 제어 정보들이 필드버스를 통하여 전송된다. 여기에서, 플랜트 출력이나 제어 입력이 주어진 샘플링 시간 내에 전송되지 못하게 되면, 제어 시스템의 성능이 나빠질 뿐만 아니라, 최악의 경우에는 플랜트의 고장이 발생할 수도 있다. 따라서, NCS에서 제어기를 설계하는 경우, 네트워크의 도입으로 인하여 발생하게 되는 네트워크 지연(network-induced delay)에 대한 보상이 이루어져야 한다.

본 논문에서는 IEC 필드버스의 일부인 Profibus-DP를 이용한 NCS의 구축 방법에 대하여 제안하며, 네트워크 지연을 보상하기 위한 방법으로서, 1입력 1출력 플랜트를 위한 원격 퍼지 제어기(Remote Fuzzy Logic Controller, RFLC)의 설계 방법에 대하여 제안한다. 또한, 실험 모델을 구축하여 RFLC

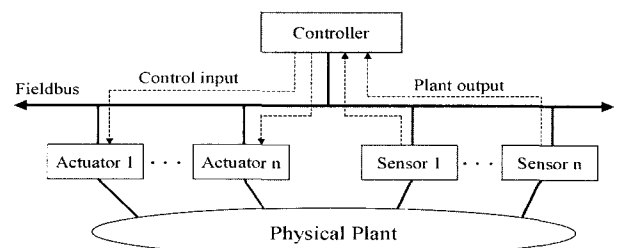


그림 1. 네트워크 기반 제어 시스템의 일반적인 구조.
Fig. 1. A typical structure of NCS.

접수일자 : 2001. 9. 20., 수정완료 : 2002. 1. 15.

이경창, 이 석 : 부산대학교 기계공학부 및 기계기술연구소(gclee@pnu.edu/slee@pnu.edu)

의 성능을 평가한다.

본 논문은 서론을 포함하여 5장으로 구성되어 있다. II장에서는 Profibus-DP를 이용한 NCS의 설계 방법에 대하여 제안하였으며, III장에서는 RFLC의 설계 방법에 대하여 제안하였다. IV장에서는 실험 모델을 이용하여, RFLC의 성능을 평가하였으며, 마지막으로, V장에서는 결론을 서술하였다.

II. Profibus-DP를 이용한 NCS

1. Profibus-DP 프로토콜의 구조

Profibus-DP는 IEC 필드버스의 일부로서, 공정 시스템이나 제어 시스템에서 적용하기 위하여 개발되었다[10]. 특히, Profibus-DP는 제어 시스템에서 요구하는 실시간 요구 조건을 만족시키기 위하여, 그림 2와 같이 OSI(Open Systems Interconnection) 참조 모델 7개 계층 중 1, 2계층인 PHY(PHYsical layer)와 FDL(Fieldbus Data Link layer)만을 사용한다. 또한, 매체 접속 제어(Medium Access Control) 방식으로는 일정한 전송 지연을 보장해주는 폴링(polling) 방식을 사용하고 있다.

Profibus-DP의 전송 매체로는 RS485가 사용되며, 전송속도는 9.6Kbps에서 12Mbps까지 지원된다. Profibus-DP에 접속되는 스테이션들은 마스터 스테이션(master station)과 슬레이브 스테이션(slave station)으로 구분되며, 마스터는 미리 정해져 있는 폴링 주기에 맞추어 슬레이브에게 순차적으로 전송 기회를 부여한다. 반면, 슬레이브는 마스터의 폴링에 따라 마스터로부터 필드 기기의 제어 신호를 수신하거나 필드 기기의 상태 신호를 마스터로 전송하는 역할을 수행한다.

그림 3은 Profibus-DP스테이션의 통신 모델을 나타내고 있다. 통신 프로세스를 가진 사용자 인터페이스는 FDL과의 원활한 통신을 위하여 DPRAM(Dual Port RAM)을 이용한다. 사용자 인터페이스에서 동작하는 모듈들은 폴링 주기마다 DPRAM의 입력 버퍼(input buffer)에 수신되어 있는 데이터를 읽거나, 전송할 데이터를 해당 모듈의 출력 버퍼(output buffer)에 기록한다. 또한, FDL은 DPRAM의 출력 버퍼로부터 데이터를 읽어 해당 스테이션으로 송신하거나, 각 슬레이브로부터 응답 프레임을 수신하여 해당 DPRAM의 입력 버퍼에 기록한다. 이러한 통신 모델의 특성, 즉 응용 프로세스와 통신 기능이 분리되어 있다는 특성으로 인하여, DPRAM에 저장된 데이터들은 일정한 지연 내에 전송이 가능하며, 엄격한 실시간이 요구되는 시스템에 적합하다고 알려져 있다[11].

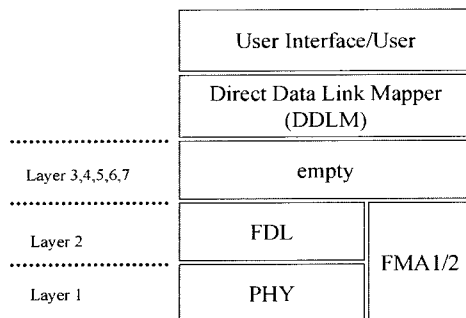


그림 2. Profibus-DP의 프로토콜 구조.
Fig. 2. Protocol architecture of Profibus-DP.

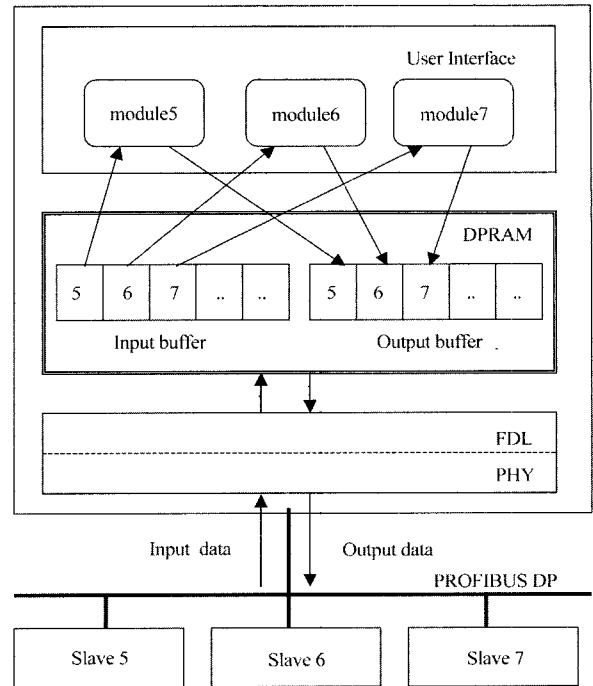


그림 3. Profibus-DP의 통신 모델.
Fig. 3. Communication model of Profibus-DP.

2. Profibus-DP를 이용한 NCS의 구조

그림 4에는 Profibus-DP를 이용한 NCS의 구조를 나타내었다. 그림에서 마스터는 입/출력 데이터 프로세스(input/output data process)와 통신 프로세스(communication process)로, 슬레이브 스테이션은 입/출력, 통신 및 I/O 계층 프로세스(I/O instrumentation process)로 구성된다. 이러한 프로세스들은 통신 사이클(communication cycle)과 계측 사이클(instrumentation cycle)을 따라 정보를 교환한다.

마스터의 동작 순서는 다음과 같다. 먼저, 통신이 시작되면, 슬레이브와의 정보 교환을 위하여 DP 마스터 함수 기능을 사용하는 입/출력 데이터 프로세스가 동작한다. 그리고 나서, 입력 데이터 프로세스는 DPRAM 입력 버퍼로부터 출력 정보를 읽어 출력 데이터 프로세스로 전달한다. 출력 데이터 프로세스 내에 구현되어 있는 플랜트의 제어기 모듈은 출력 정보를 이용하여 플랜트의 제어 정보를 생성하며, 이를 슬레이브로 전송하기 위하여 DPRAM 출력 버퍼에 저장한다.

슬레이브의 경우, 통신이 시작되면 마스터와의 정보 교환을 위하여 DP 슬레이브 함수 기능을 사용하는 입/출력 데이터 프로세스가 동작된다. 입력 데이터 프로세스는 DPRAM 입력 버퍼로부터 제어 정보를 읽어 출력 데이터 프로세스로 전달한다. 출력 데이터 프로세스 내에 있는 플랜트 구성 모듈은 제어 정보를 이용하여 (계측 사이클을 따라) 플랜트를 동작시키고, 플랜트로부터 읽은 피드백 정보를 읽어 DPRAM 출력 버퍼에 저장한다.

이상과 같은 Profibus-DP의 폴링 방식과 DPRAM을 이용한 송수신 방식으로 인하여, Profibus-DP를 이용한 NCS에서는 네트워크의 상태에 따라 확률적으로 변하는 네트워크 지연이 발생한다[12]. 특히, 이러한 네트워크 지연은 제어 시스템의 성능을 악화시키고 불안정하게 만든다. 그러나, 네트워

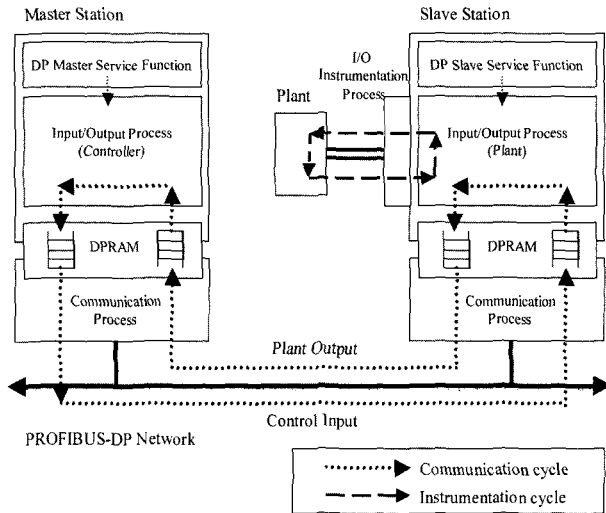


그림 4. Profibus-DP에서의 NCS의 구조.
Fig. 4. Structure of NCS via Profibus-DP.

크 지연이 발생하지 않도록 NCS를 구성하는 것이 불가능하기 때문에, 네트워크 지연을 고려한 원격 제어기(remote controller) 설계가 이루어져야 한다[6][7].

III. NCS에서의 원격 퍼지 제어기의 구조

NCS에서의 제어기 설계 방법은 제어 시스템 내부의 시간 지연 요소를 고려한 설계 방법으로부터 유도될 수 있다. 일반적으로, 시간 지연 요소를 고려한 제어기 설계 방법은 모델 기반 방법과 무모델 기반 방법으로 나눌 수 있다. 대표적인 모델 기반 방법으로는 Smith 예측기[13]나 IMC (Internal Model Control)[14] 등이 있으며, 무모델 기반 방법으로는 Ziegler-Nichols 계수조정법[15]이나 퍼지 제어[16] 등이 있다. 그러나, 모델 기반 방법은 모델의 정확도에 따라 제어기의 보상 능력이 크게 달라지기 때문에, 실용적이지 못하다고 알려져 있다[17]. 반면, 퍼지 제어와 같은 무모델 기반 방법은 모델링이 어려운 플랜트나 기존의 선형적 제어 기법으로는 성능 향상에 한계가 있는 플랜트에 주로 사용되어 왔으며, 많은 성공 사례들이 보고되고 있다[18].

특히, NCS의 경우, 네트워크의 설계 방법이나 네트워크에서 발생하는 트래픽의 양, 네트워크에 참여하는 스테이션의 수 등에 따라 네트워크 지연이 확률적으로 변하기 때문에, 네트워크 지연을 측정하기가 매우 어려우며, 수학적 모델을 구하는 것도 쉽지 않다. 이러한 이유로 인하여, 퍼지 제어는 NCS의 제어기 설계 방법으로 효율적이라 할 수 있다.

그림 5는 NCS에서 1입력 1출력 플랜트를 위한 원격 퍼지 제어기의 구조를 나타내고 있다. 그림에서, 제어기와 플랜트는 필드버스에 의해 연결되어 있으며, 네트워크를 통하여 제어 입력과 플랜트 출력이 전송된다. 이러한 필드버스를 사용함에 따라, 제어 신호와 출력 신호에는 네트워크 지연이 포함되게 된다. 여기에서, $y(t)$ 는 플랜트 출력, $r(t)$ 는 플랜트의 기준 입력을 나타낸다. 또한, $e(t)$ 는 플랜트 출력과 기준 입력 간의 에러 값(error value)을, $\Delta e(t)$ 는 에러 $e(t)$ 의 변화량을, $u(t)$ 는 플랜트의 제어 입력을 나타낸다.

RFLC는 플랜트의 에러 값을 퍼지 언어 변수(linguistic var-

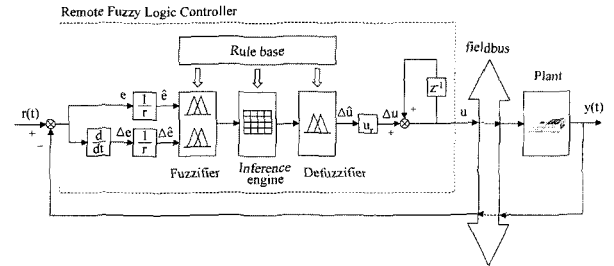


그림 5. NCS를 위한 RFLC의 구조.
Fig. 5. Structure of RFLC for NCS.

iable)와 언어 값(linguistic value)으로 변환하는 퍼지화부와 전문가의 경험을 바탕으로 생성된 퍼지 제어 규칙(fuzzy control rule)을 이용하여 퍼지 출력을 추론하는 추론 엔진(inference engine), 추론된 결과를 제어 입력으로 변환하는 비퍼지화부로 구성되어 있다. RFLC의 퍼지 입력으로는 (1)과 같이 $\hat{e}(t)$ 와 $\Delta \hat{e}(t)$ 가, 퍼지 출력으로는 $\Delta \hat{u}(t)$ 가 사용되었다. 여기에서, $\hat{e}(t)$ 와 $\Delta \hat{e}(t)$, $\Delta \hat{u}(t)$ 는 정규화된 값으로서, 기준 입력이나 제어 입력이 변화하는 경우에도 동일한 제어 효과를 얻기 위하여, $e(t)$ 와 $y(t)$ 를 스케일 계수(scale factor)인 기준 입력 r 과 스텝 응답이 안정화되었을 때 제어 시스템에 입력되는 제어 입력으로 정의되는 기준 제어 입력 u_r 로 나눈 값으로 정의하였다.

$$\begin{aligned} \hat{e}(t) &= \frac{r - y(t)}{r} \\ \Delta \hat{e}(t) &= \frac{e(t) - e(t-1)}{r} \\ \Delta \hat{u}(t) &\approx \frac{\Delta u(t)}{u_r} \end{aligned} \tag{1}$$

1. 퍼지화, 퍼지추론 및 비퍼지화

그림 6은 RFLC의 퍼지 입력력에 대한 소속 함수(member-ship function)와 언어 값을 나타낸다. 본 논문에서는, 퍼지 제어기의 연산 속도를 위하여, 소속 함수로 0값에서 대칭인 삼각 퍼지 수(triangular fuzzy number)를 사용하였으며, 플랜트 스텝 응답에 대한 경험을 바탕으로 하여 퍼지 제어기의 퍼지 언어 변수를 Negative Big(NB), Negative Medium(NM), Negative Small(NS), Zero(ZE), Positive Small(PS), Positive Medium(PM), Positive Big(PB)으로 선정하였다. 특히, 언어 값은 플랜트의 설계 기준이 변하더라도 동일한 제어 성능을 얻기 위하여, 시스템 설계자의 요구에 따른 설계 사양에 관한 계수로서, R_{TS} 와 R_{MP} 을 이용하여 표시하였다. 여기에서, R_{TS} 는 정착시간에 대한 계수로서, 만약, 플랜트의 설계 사양으로 2% 정착시간이 사용되는 경우에는 R_{TS} 는 0.02로, 5% 정착시간이 사용되는 경우에는 R_{TS} 는 0.05로 설정된다. 또한, R_{MP} 는 최대 오버슈트에 대한 계수로서, 플랜트의 최대 오버슈트에 대한 설계 사양으로 10%가 사용되는 경우에는 R_{MP} 는 0.1로 설정된다.

퍼지 추론법과 비퍼지화 과정으로, 퍼지 제어기의 계산 부담을 줄이기 위하여, 그림 7과 같이 계산 과정이 비교적 단순한 Mamdani의 Minimum operation rule과 SCG(Simplified

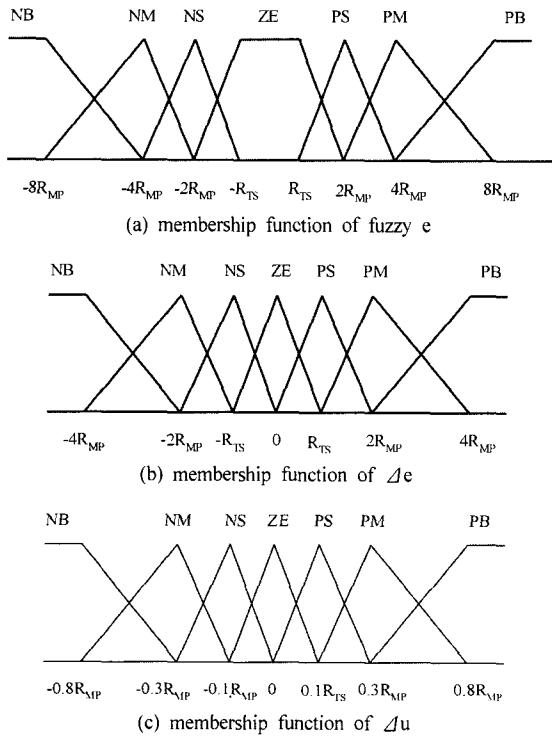


그림 6. RFLC의 소속함수.
Fig. 6. Membership functions of RFLC.

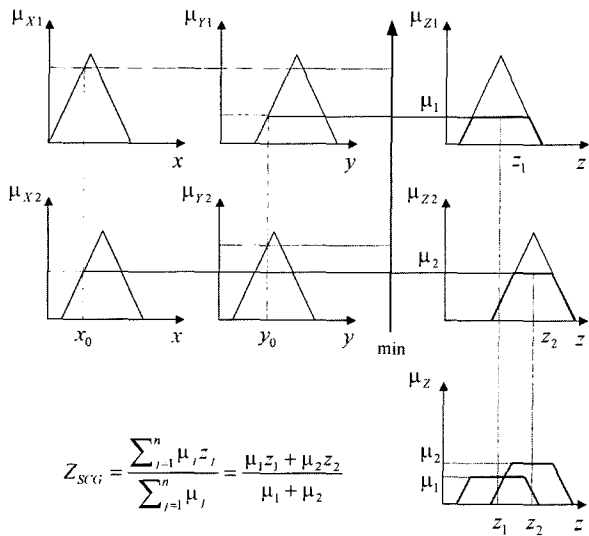


그림 7. RFLC의 추론 과정.
Fig. 7. Inference Procedure of RFLC.

Center of Gravity)를 사용하였다. 특히, SCG는 COG(Center Of Gravity)와는 달리, 추론 결과로 얻어진 각 퍼지 출력의 소속 함수의 꼭지점을 이용하여 퍼지 출력을 계산한다[19].

2. 퍼지 제어 규칙

일반적으로, 1입력 1출력 퍼지 제어기에서 제어 규칙은 그림 8(a)와 같은 폐루프 제어 시스템의 스텝 응답과 그림 8 (b)와 같은 스텝 응답으로부터 계산된 \hat{e} - $\Delta\hat{e}$ 위상 공간(phase plane)으로부터 구할 수 있다[19][20].

특히, 그림 8(b)의 위상 공간에서 보는 바와 같이 특징점 들은 4개의 그룹, 즉 a_i, b_i, c_i, d_i 로 분류할 수 있으며, 폐루프

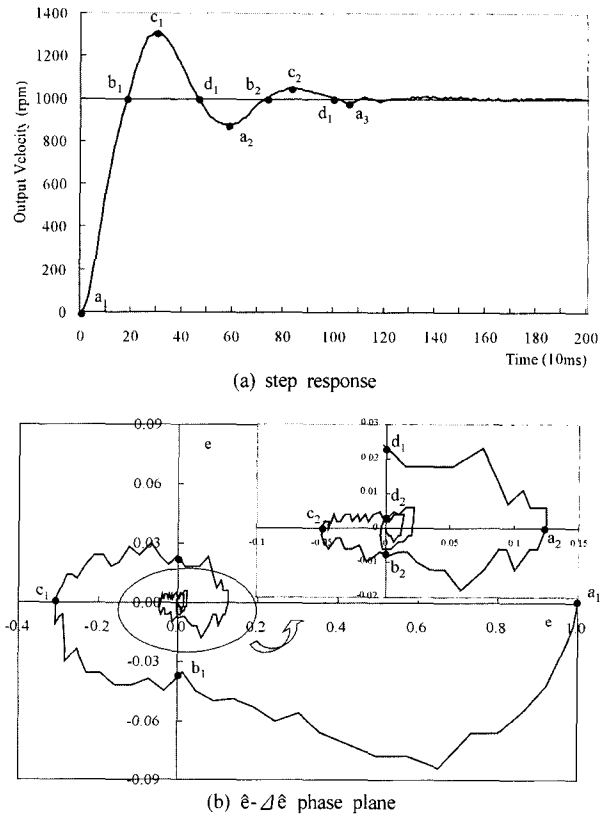


그림 8. 폐루프제어시스템의 스텝 응답 및 위상 공간.
Fig. 8. Typical step response and phase plane of a closed loop system.

제어 시스템의 퍼지 제어 규칙은 이러한 특징점들로부터 구할 수 있다. 예로, 첫 번째 특징점 a_1 에서, \hat{e} 는 양(positive)이고, $\Delta\hat{e}$ 는 0이므로, 기준 응답과의 오차를 줄이기 위해서는 가속이 요구되므로, $\Delta\hat{u}$ 를 양(positive)으로 하여야 한다. 또한, \hat{e} 의 크기에 따라, $\Delta\hat{u}$ 의 크기를 조절하여야 한다. 따라서, a_i 점 부근에서 필요한 제어 규칙은 \hat{e} 값에 따라 다음과 같이 나타낼 수 있다.

- If \hat{e} is PB and $\Delta\hat{e}$ is ZE then $\Delta\hat{u}$ is PB
- If \hat{e} is PM and $\Delta\hat{e}$ is ZE then $\Delta\hat{u}$ is PM
- If \hat{e} is PS and $\Delta\hat{e}$ is ZE then $\Delta\hat{u}$ is PS
- If \hat{e} is ZE and $\Delta\hat{e}$ is ZE then $\Delta\hat{u}$ is ZE

위와 같은 방법으로 b_i, c_i, d_i 점에서 제어 규칙을 구하면, 표 1(a)에서 보는 바와 같이 13개의 제어 규칙으로 구성되는 기본 퍼지 제어 규칙(primary fuzzy control rule)을 얻을 수 있다.

또한, 보다 더 향상된 스텝 응답을 얻기 위하여, \hat{e} - $\Delta\hat{e}$ 위상 공간의 모든 영역에서 필요한 제어 규칙을 구하면, 표 1(b)에서 보는 바와 같이 49개의 제어 규칙으로 구성되는 개선된 제어 규칙(advanced fuzzy control rule)을 구할 수 있다.

IV. NCS에서 RFLC의 성능 평가

본 논문에서는 제안된 NCS를 위한 RFLC의 성능을 평가하기 위하여, 그림 9와 같이 DC 모터를 이용하여 실험 모델을 구성하였다. 실험 모델에는, 1대의 제어기와 1대의 DC 모터가 접속되어 있으며, 네트워크에서의 트래픽 생성을 위하여 트래픽 생성기(traffic generator)가 접속되어 있다. 또한,

표 1. 퍼지 제어 규칙.

Table 1. Fuzzy control rule.

(a) primary fuzzy control rule

$\Delta \hat{e}$ \ \hat{e}	NB	NM	NS	ZE	PS	PM	PB
NB				NB			
NM				NM			
NS				NS			
ZE	NB	NM	NS	ZE	PS	PM	PB
PS				PS			
PM				PM			
PB				PB			

(b) advanced fuzzy control rule

$\Delta \hat{e}$ \ \hat{e}	NB	NM	NS	ZE	PS	PM	PB
NB	NB	NB	NB	NB	NM	NS	ZE
NM	NB	NB	NB	NM	NS	ZE	PS
NS	NB	NB	NM	NS	ZE	PS	PM
ZE	NB	NM	NS	ZE	PS	PM	PB
PS	NM	NS	ZE	PS	PM	PB	PB
PM	NS	ZE	PS	PM	PB	PB	PB
PB	ZE	PS	PM	PB	PB	PB	PB

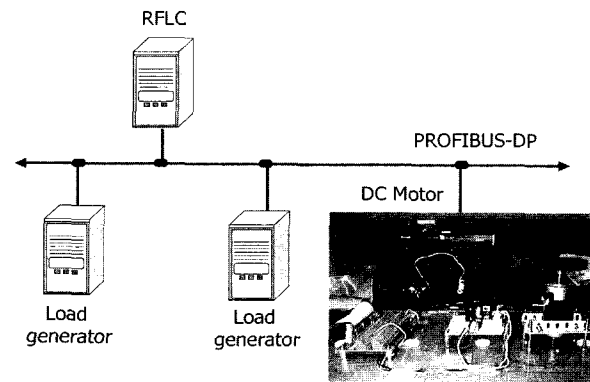


그림 9. NCS 실험 모델.

Fig. 9. Experimental model of NCS.

플랜트의 제어 입력을 위하여 12비트 D/A 컨버터가, 엔코더 출력을 위하여 16비트 카운터가 사용되었으며, DC 모터의 샘플링 시간은 10ms로, 기준입력은 700rpm으로 설정하여 속도제어를 수행하였다.

각 스테이션들의 네트워크 인터페이스로는 Siemens의 SAB-C165 마이크로컨트롤러와 마스터 및 슬레이브 전용 프로토콜 소자인 ASPC2와 SPC3을 장착한 Softing사의 PROFI-board가 사용되었다. Profibus-DP의 전송속도는 1.5Mbps로, 데이터 전송량은 2바이트(byte)로 설정하였다. 여기에서, 데이터 전송량이 2바이트인 이유는 DC 모터의 엔코더 카운트 신호가 16비트이므로 2바이트로 처리가 가능하기 때문이다.

1. FLC와 RFLC간의 성능 비교

본 절에서는 직접 연결된 퍼지 제어기와 네트워크에 의하여 연결된 원격 퍼지 제어기의 성능을 비교함으로써, 네트워

크 지연이 퍼지 제어기의 성능에 미치는 영향을 분석하였다. 또한, 퍼지 제어기 규칙의 수가 제어 시스템의 성능에 미치는 영향을 분석하였다.

먼저, FLC의 설계사양으로 최대 오버슈트는 10% 이하, 5% 정착시간은 0.8sec 이하로 선정하여, 퍼지 입출력 소속함수의 언어 값을 결정하였다. 또한, 기준 입력 r 은 700rpm으로, 기준 제어 입력 u_r 는 2840, R_{Ts} 는 0.05, RMP는 0.1로 설정하였다.

그림 10(a)는 직접 연결된 FLC를 사용하는 경우, 13개 및 49개의 퍼지 제어 규칙을 가진 FLC의 응답 성능을 나타낸다. 그림에서, 49개의 퍼지 제어 규칙을 가진 FLC의 성능이 13개의 제어 규칙을 가진 FLC보다 약간 우수한 성능을 보이고 있다. 그러나, 전체적으로는 크게 차이를 보이지 않는다. 오히려, 13개의 퍼지 제어 규칙을 가진 FLC의 실행 속도가 49개보다 빠른 점을 고려한다면, 13개의 퍼지 제어 규칙을 가진 FLC가 더 효과적이라고 볼 수 있다.

그림 10(b)는 직접 연결된 FLC를 수정없이 NCS에 적용한 경우, 13개 및 49개의 퍼지 제어 규칙을 가진 RFLC의 응답 성능을 나타낸다. 그림에서 직접 연결된 FLC를 수정없이 NCS에서 적용하더라도, 제어 시스템의 응답 성능은 큰 변화를 보이지 않는다. 이는 제어 시스템에 불확실성이 존재하는 경우에도 효과적으로 제어를 수행할 수 있는 퍼지 제어의 강인성(robustness)에 의한 것이다. 따라서, 이러한 결과로부터, 직접 연결된 시스템을 위하여 설계된 퍼지 제어기가 확률적인 네트워크 지연이 발생하는 NCS를 위한 효율적인 제어 방법임을 알 수 있다.

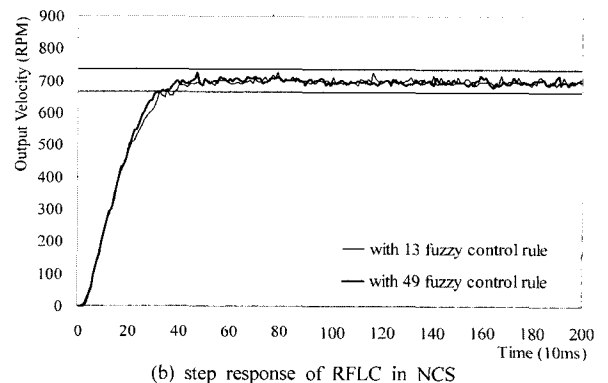
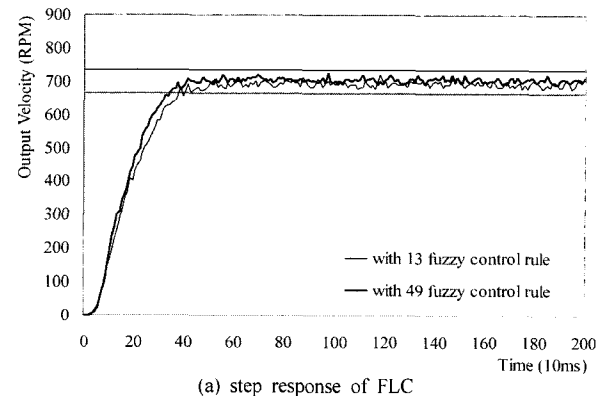


그림 10. FLC와 RFLC간의 성능 비교.

Fig. 10. Comparison between FLC and RFLC.

2. 고전적인 PID, Ziegler-Nichols 계수조정법 및 RFLC간의 성능 비교

본 절에서는 고전적인 PID 제어기 설계 방법(conventional PID) [21]과 대표적인 무모델 기반 방법 중의 하나인 Ziegler-Nichols 계수조정법(ZN tuned PID)[15] 및 RFLC의 응답 성능을 비교하였다.

먼저, 고전적인 제어기 설계 방법으로 PID 제어기를 구하기 위해서는 플랜트 모델이 필요하다. 실험 모델에서, DC 모터를 1차 함수로 가정한다면, 신호 압축법에 의하여 (2)와 같이 플랜트의 전달 함수를 구할 수 있다.

$$G(s) = \frac{y(s)}{u(s)} = \frac{543.47}{0.12s+1} \quad (2)$$

여기에서, 제어기의 설계 사양을 최대오버슈트는 10% 이하, 5% 정착시간은 0.8sec 이하, 감쇠비는 0.5 이하로 설정하고, PID 제어기를 구하면, $K_p=0.011$, $K_i=0.2$, $K_d=0.0001$ 와 같다. 이를 NCS에 적용하였을 경우, 응답 성능은 그림 11에서 보는 바와 같이 최대 오버슈트는 16%, 정착시간은 1.12초로 나빠짐을 알 수 있다. 이러한 결과는 플랜트의 출력 신호가 네트워크 지연으로 인하여, 3~5 샘플링 주기 정도 늦게 제어기에 도착하기 때문에 발생한다. 따라서, 플랜트의 설계 사양을 만족시키기 위해서는 네트워크 지연을 고려한 제어기 설계가 필요함을 알 수 있다.

다음으로, 플랜트 내부에서의 시간 지연을 고려하는 Ziegler-Nichols 계수조정법에서는 시간 지연에 대한 정확한 정보가 필요하다. 본 절에서는 NCS에서 발생하는 네트워크 지연을 시간 지연으로 가정하고, 네트워크 지연이 약 3 샘플링 시간이라고 하는 경우, Ziegler-Nichols 계수조정법에 의하여 PID 제어기를 구하면, $K_p=0.008832$, $K_i=0.147$, $K_d=0.000132$ 와 같다. 이를 NCS에 적용하였을 경우, 응답 성능은 그림 11에서 보는 바와 같이 최대 오버슈트는 8%, 정착시간은 0.85초로 나타났다. 이러한 결과는 플랜트의 설계 사양을 만족하고 있다.

마지막으로, RFLC에 의한 응답 성능은 그림 11에서 보는 바와 같이 최대 오버슈트는 4.5%, 정착시간은 0.53초로 나타났다. 이러한 결과는 다른 두 제어기에 비하여 상승시간이

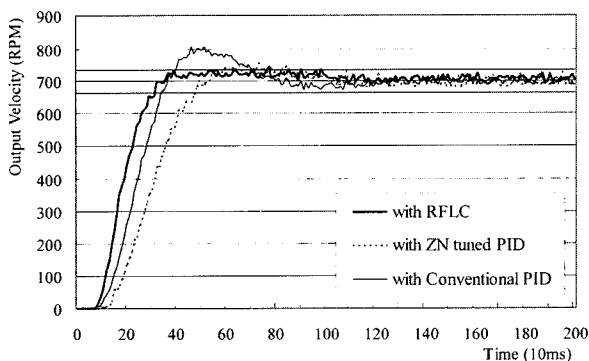


그림 11. 고전적인 PID, ZN 계수조정법 및 RFLC의 실험 결과.

Fig. 11. Experimental result of conventional PID, ZN tuned PID and RFLC.

매우 짧고, 최대 오버슈트가 상당히 작은 편이다.

이상과 같은 실험 결과로 볼 때, NCS에서 퍼지 제어를 이용하는 경우, Ziegler-Nichols 계수조정법에 비하여 상대적으로 성능이 우수함을 알 수 있다. 특히, Ziegler-Nichols 계수조정법은 네트워크 지연을 직접 측정하여 그 크기를 정확히 알아야 하지만, 직접 연결된 상태에서 튜닝된 퍼지 제어기는 네트워크 지연에 대한 정보를 알지 못하는 경우에도 효과적인 제어가 가능함을 알 수 있다. 일반적으로, 네트워크 지연은 네트워크에서의 여러 가지 요인들에 의하여 확률적으로 변하기 때문에, NCS에 대한 충분한 성능 평가를 수행하기 전에는 정확한 값을 알기가 매우 어렵다. 따라서, 네트워크 지연에 대한 정확한 정보 없이도 효과적인 응답 성능을 보이는 RFLC가 다른 제어 방법보다 우수하다고 볼 수 있다.

V. 결론

본 논문은 Profibus-DP를 이용한 NCS의 구축 방법을 다루었다. 특히, NCS에서 발생하는 네트워크 지연은 제어 시스템의 성능을 저하시키기 때문에, 네트워크 지연을 고려한 제어기 설계가 필요하다. 본 논문에서는 네트워크 지연을 효과적으로 보상할 수 있는 제어기 설계 방법으로서, 원격 퍼지 제어기에 대하여 제안하였으며, 설계된 원격 퍼지 제어기를 고전적인 PID 제어기 및 Ziegler-Nichols 계수조정법과 비교하여 다음과 같은 결론을 얻을 수 있었다.

첫째, 플랜트와 직접 연결된 상태에서 튜닝된 퍼지 제어기를 NCS에서 사용하는 경우, 플랜트의 설계 사양이 만족됨을 확인하였다.

둘째, 원격 퍼지 제어기는 NCS에서 발생하는 네트워크 지연에 대한 정확한 정보가 없는 상태에서도 효과적인 제어가 가능함을 확인하였다. 특히, 네트워크 지연은 네트워크에서의 여러 가지 요인들에 의하여 확률적으로 변하기 때문에, 정확한 값을 추측하기가 매우 어렵다. 따라서, 원격 퍼지 제어는 정확한 네트워크 지연을 요구하는 Ziegler-Nichols 계수조정법과 같은 제어 시스템에서의 시간 지연을 고려한 PID 튜닝 방법보다 NCS에 더 적합함을 확인하였다.

참고문헌

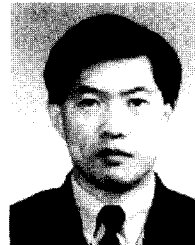
- [1] A. Ray, "Networking for computer-integrated manufacturing," *IEEE Network*, vol. 2, no. 3, pp. 40-47, 1988.
- [2] 홍승호, "필드버스 기술 동향," 제어·자동화·시스템공학회지, 제4권, 제6호, pp. 13-18, 1998.
- [3] 권옥현, 김영신, "분산 실시간 시스템에서의 네트워크 프로토콜," 제어·자동화·시스템공학회지, 제4권, 제6호, pp. 27-34, 1998.
- [4] 이경창, 김기용, 김희현, 이석, "실시간 페루프 제어 시스템을 위한 Profibus-FMS 네트워크의 구현," 제어·자동화·시스템공학회 논문지, 제6권, 제5호, pp. 92-98, 2000.
- [5] IEC 61158-4, *Digital Data Communications for Measurement and Control - Fieldbus for Use in Industrial Control Systems - Part 4: Data Link Protocol Specification*, IEC, 1999.
- [6] G. C. Walsh, and Y. Hong, "Scheduling of networked control systems," *IEEE Control Systems Magazine*, vol. 21, no. 1, pp. 57-65, 2001.

- [7] W. Zhang, M. S. Branicky, and S. M. Phillips, "Stability of networked control systems," *IEEE Control Systems Magazine*, vol. 21, no. 1, pp. 84-99, 2001.
- [8] Y. Halevi, and A. Ray, "Integrated communication and control systems: Part I. analysis," *Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME*, vol. 110, pp. 367-373, 1988.
- [9] J. Nilsson, *Real-Time Control Systems with Delays*, Ph. D. dissertation, Dept. Automatic Control, Lund Institute of Technology, Lund, Sweden, 1998.
- [10] EN 50170, *Profibus Specification - Normative Parts of Profibus-FMS, -DP, -PA According to the European Standard*, vol. 2, 1998.
- [11] K. Bender, *Profibus-The Fieldbus for Industrial Automation*, Prentice Hall, 1993.
- [12] 이경창, 이석, "유전자 알고리즘을 이용한 네트워크 기반 제어 시스템의 원격 제어기 설계," 제어·자동화·시스템공학 논문지, 제8권, 제1호, 2002.
- [13] O. J. Smith, "Closer control of loops with dead time," *Chemical Engineering Process*, vol. 53, no. 5, pp. 217-219, 1959.
- [14] M. Morari and E. Zafiriou, *Robust process control*, Prentice-Hall, 1989.
- [15] J. G. Ziegler, and N. B. Nichols, "Optimum settings for automatic controllers," *Transactions on ASME*, vol. 64, 1942.
- [16] M. J. Jang and C. L. Chen, "Fuzzy successive modeling and control for time-delay system," *International Journal on Systems Science*, vol. 27, no. 12, pp. 1483-1490, 1996.
- [17] F. Zhao, J. Ou, and W. Du, "Pattern-based fuzzy predictive control for a chemical process with dead time," *Engineering Applications of Artificial Intelligence*, vol. 13, pp. 37-45, 2000.
- [18] F. Betin, D. Pinchon, and G. A. Capolino, "Fuzzy logic applied to speed control of a stepping motor drive," *IEEE Transactions on Industrial Electronics*, vol. 47, no. 3, pp. 610-622, 2000.
- [19] C. C. Lee, "Fuzzy logic in control systems: Fuzzy logic controller - Part I and II," *IEEE Transactions on systems, Man, and Cybernetics*, vol. SMC-20, no. 2, pp. 404-435, 1990.
- [20] G. Li, K. M. Tsang, and S. L. Ho, "Fuzzy based variable step approaching digital control for plants with time delay," *ISA Transactions*, vol. 37, pp. 167-176, 1998.
- [22] K. Ogata, *Modern Control Engineering*, Prentice-Hall, pp. 257-270, 1990.



이 경 창

1971년 5월 1일생. 1996년 부산대 생산기계공학과 졸업. 동대학원 석사(1998). 동대학원 박사수료(2001). 1998년~현재 기계공학연구정보센터 전임연구원. 관심분야는 필드버스(Profibus, Fieldbus Foundation, DeviceNet, Lonworks), 산업용 이더넷, 지능 제어, 인터넷 기반 생산 시스템.



이 석

1961년 12월 11일생. 1984년 서울대 기계공학과 졸업. 펜실바니아 주립대 석사(1985). 동대학원 박사(1990). 1990년~1993년 신시내티 대학교 기계공학과 조교수. 1993년~현재 부산대학교 기계공학부 부교수. 관심분야는 필드버스(Profibus, Fieldbus Foundation, DeviceNet, Lonworks), 차량용 네트워크(CAN, LIN), DES. RP.