

퍼지 PI 알고리즘을 이용한 산업용 로봇의 위치/힘 제어

서 일홍 *, 오 홍 종혁 **, 오 상록 ***, 김 광배 ***

* 한양대학교 전자공학과 부교수,
** 한양대학교 전자공학과 대학원,
*** 한국과학기술원 제어시스템 연구실

Position / Force Control of Industrial Robots using the Fuzzy PI Algorithm

Il Hong Suh, Jong Hyuck Hong, Sang-Rok Oh, Kwang Bac Kim

Dept. of Electronics Eng., Hanyang Univ
Control System Lab, Kist

Abstract

The hybrid position/force control is required when two or more robots perform a cooperative task in a uncertain environment, or when single robot does a task with a constant force to the environment.

In this paper, a new control algorithm which control simultaneously the position and the force are proposed, however, especially the conventional position controller employed in the present robot control is used. Moreover, in order to improve the output response characteristics of the system, the PI gains which were computed from the PI gain tuning techniques, are varied based on the results of the Fuzzy algorithm.

1. 서 론

현재 주로 사용하는 산업용 로봇은 미리 위치를 알려준 다음 로봇의 End-Effector를 위치제어함으로써 주어진 일을 수행하고 원하는 동작을 수행한다. 그러나 조립작업이나 2대이상의 로봇이 공간상의 한 물체를 놓고 협조제어를 수행하는 작업과 같이 상호간에 힘이 발생하고, 주변환경을 잘 알 수 없는 상황에서는 위치제어만으로 정확히 로봇을 제어할 수 없으므로 주변상황과 로봇의 End-Effector 사이에 접촉이 일어났을 때 발생하는 힘을 측정하여 정확한 로봇 제어를 해야만 한다.

기존의 힘/위치 제어를 검토하면, Raibert와 Craig[1]는 하이브리드(Hybrid) 위치/힘제어 알고리즘을 제시하였다. 여기서는 요소들이 "1" 또는 "0"을 갖는 대각선 행렬인 Compliance Selection Matrix S를 사용하여 직교좌표상에서 힘제어축과 위치제어축으로 나누어 힘과 위치를 동시에 제어하였다. 그러나, 기존의 위치/힘제어 알고리즘은 상용의 로봇제어에 사용되고 있는 위치제어기를 사용할 수 없고 새로운 제어기를 구성해야 된다는 단점을 갖고 있다. 그래서 본 논문에서의 제어 알고리즘은 기존의 위치제어기에 덧붙여서 수행할 수 있는 하이브리드(Hybrid) 위치/힘제어 알고리즘을 제시하였다. 한편, 힘제어용 위하여 PI 제어를 사용할 경우 오버슈트(Overshoot)가 크고 정착시간(Settling time)이 길다는 단점을 갖고 있어 이 점을 개선하기 위해 퍼지 PI제어를 도입하였다.

퍼지논리제어는 80년대에 들어서면서 계통의 자동제어를 위해, 일반적 응용 가능성이 크고 정확히 모델링(Modelling)할 수 없는 계통을 제어할 수 있는 제어이론의 하나로써 전문가(Expert) 지식을 기반으로하여 연구가 활발히 진행되고 있다.

Tzafestas[2]의 퍼지제어와 PID제어와의 합성에 관한 연구중에는 PID 이득 조정의 일반적인 합당성을 갖지 못하는 영역이 있는 단점이 있는데, 본 논문에서는 이러한 단점들을 제거할 수 있는 새로운 방법들을 제안함으로써 PI 제어에 의한 하이브리드(Hybrid) 위치/힘제어 알고리즘의 출력특성을 안정적이고 바람직한 출력특성으로 갖게끔 하였다. 구체적으로, 기존의 조정기술(Tuning Technique)을 이용하여 설정한 이득값들을 기준으로 안정도(Stability)를 만족하는 범위내에서 이득값들의 변화를 이

용하여 오버슈트(Overshoot) 및 정상상태편차(Steady State Error)를 최소한으로 줄일 수 있는 퍼지제어 법칙을 설정하였다.

2. 상용의 위치제어기를 이용할 수 있는 하이브리드(Hybrid) 위치/힘제어 알고리즘

본 논문에서는 상용의 로봇제어에 사용되고 있는 위치제어기에 적용될 수 있는 하이브리드(Hybrid) 위치/힘제어 알고리즘을 제시한다.

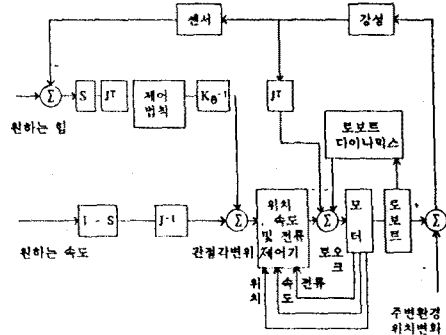


그림. 1 하이브리드(Hybrid) 위치/힘 제어 개념적 블록도

제어법칙을 거쳐서 나온 토크(Torque)값을 기존의 위치제어기의 입력값으로 할 수 없고, 각 회전축의 회전변위값으로써 변환해 주어야 하므로, 본 논문에서는 회전축에 유연성(Compliance)이 있다고 가정하여 회전변위값으로써 변환하였다. 결국, 기존의 위치/힘제어기와 다른점은 힘제어기는 폐루프(Closed Loop)로 제어하고 최종 출력값이 토크(Torque)가 아닌 회전변위(Angle)값이며, 위치제어기는 상용의 로봇제어의 위치제어기를 이용함으로써 개루프(Open Loop)로 제어하였다. 제어모델의 구현을 위한 기본수식은 다음과 같다.

End-Effector가 주변환경과 접촉을 이루고 있을 경우 그 주변환경, End - Effector, 센서(Sensor) 등이 받는 힘과 위치변위값에 비례관계가 성립하는 강성(Stiffness)값을 갖는다.

$$dF = Kx \, dx \quad (1)$$

where, dx = 직교좌표상에서의 위치변위값
 dF = 직교좌표상에서의 End-Effector의 힘변위값
 Kx = 직교좌표상에서의 센서, End-Effector, 주변환경등의 전체 강성(Stiffness)값

이 된다.

$$dX = J \, d\theta \quad (2)$$

where, $d\theta$ = 회전축의 각변위값
 J = 로봇의 자코비언

$$dr = J^T dF \quad (3)$$

where, J^T = 자코비언의 전치행렬

dr = 각 회전축의 토오크 변위값

이고, (1) ---> (2)로 하면

$$dr = J^T Kx dX \quad (4)$$

(2) ---> (4)로 하면

$$dr = J^T Kx J d\theta = K d\theta \quad (5)$$

결국,

$$K = J^T Kx J \quad (6)$$

where, K = 회전축상에서의 강성(Siffness)

$$d\theta = K^{-1} dr \quad (7)$$

where, K^{-1} = 회전축상에서의 유연성(Compliance)

(7)식을 이용하여 힘제어의 출력 토오크(torque)값을 각 회전축의 각변위값으로 변환시킨다. 본 논문에서는 기존의 PI 제어와 퍼지제어를 조합한 퍼지 PI 제어기류 사용하였으며, 다음장에서 기본적인 설명을 하였다.

3. 퍼지(Fuzzy) 제어기

3.1 기본적인 퍼지(Fuzzy) 집합이론

Zadeh[3],[4]에 의해서 처음으로 도입된 퍼지논리를 이용한 퍼지 제어는 선형성이 없는 복잡한 시스템에 효과적으로 적용된다. 또한 퍼지제어는 논리의 애매성을 도입하였고, 수학적인 변수가 아니라 언어적인 변수(Linguistic Variable)를 사용한 전문가(Expert) 시스템이다.

3.2. Tzafestas의 퍼지(Fuzzy) PID제어 행렬

Tzafestas[2]는 다음과 같은 Macvicar-Whelan[5]의 논리를 이용하여 기존의 PID 제어에 응용하였다.

- a) 만약에 출력이 원하는 값이고, 오차변화값이 "0"이면, 제어기의 출력 값을 그대로 유지한다.
- b) 만약에 출력이 원하는 값에서 벗어나면, 오차값과 오차 변화값의 부호와 크기에 따라서 제어기의 출력값을 변화시키는데, 단, 오차가 재빨리 0으로 수렴할 수 있는 상황이면, 제어기의 출력값을 그대로 유지한다.

이러한 논리에 의해서 형성된 퍼지제어 행렬은 기존의 다른 퍼지제어 행렬 보다도 인간사고에 더욱 접근된 논리이다. 이와같이 형성된 행렬을 주 대각선축에 수직인 축에 투영시키면 이 행렬의 policy를 쉽게 관찰할 수가 있다. 행렬의 분해도를 증가시키면 투영된 부분들의 수는 증가하고 제어 policy의 fuzziness는 감소하게 되어있다. Tzafestas[2]는 이러한 현상을 이용하여 제어할 시스템에 적절한 제어 policy를 선택하여 사용하였고, 기존의 이득 조정기술(Gain Tuning Technique)을 이용하여 얻은 PID 제어기의 가장 적절한 이득값을 기준으로하여 시스템이 과도상태(Transient State)동안에는 이득값들을 조금씩 변화시켜 시스템응답특성을 향상시킨다는 생각이다. 인간의 관찰을 중심으로 볼때 PID 제어기의 각각의 특성은 다음과 같이 기술할 수 있다.

- a) 적분항(Integral Term)은 오버슈트(Overshoot)에 영향을 주므로 시스템 응답이 원하는 값을 넘는 순간에 적분 이득값을 천천히 감소시키면 오버슈트(Overshoot)을 상당히 줄일 수 있다. 반면에 상승시간(Rise Time)동안에는 약간 증가시켜 상승시간(Rise Time)을 10~20%정도 줄일 수 있다.
- b) 미분항(Derivative Term)은 단위응답(Step Response)의 평면성(Flattness)에 영향을 주므로 상승시간(Rise Time)동안과 정상상태(Steady State)동안에는 조금 증가시켜 보통 발생하는 작은 떨림(Oscillation)을 제거할 수 있다.
- c) 비례항(Proportional Term)은 증가시킬수록 상승시간(Rise Time)은 줄어들지만 떨림(Oscillation)은 증가한다. 그러므로 떨림(Oscillation)의 특성을 피하기 위해서는 감소시켜야만 한다.

Tzafestas[2]는 각 항의 이득변화값을 퍼지제어를 이용하여 다음과 같이 구성하였다.

$$P = P_0 + CV\{E, \Delta E\} \times k_1 \quad (\text{비례이득값})$$

$$I = I_0 + CV\{E, \Delta E\} \times k_2 \quad (\text{적분이득값}) \quad (8)$$

$$D = D_0 + CD\{E, \Delta E\} \times k_3 \quad (\text{미분이득값})$$

Where, $CV\{E, \Delta E\}$ = 퍼지제어기의 출력값

P_0, I_0, D_0 = 기존의 이득 조정기술을 이용하여 얻어진 이득값들

여기서 k_1, k_2, k_3 의 값은 각 항들의 변화정도를 표시하고 안정도(Stability)분석과 폐루프 응답(Closed Loop Response)의 특성을 고려하여 적절하게 결정한다. 그러나, 여기에서의 문제점은 행렬을 형성하는데 있어서 그림 2에서와 같이 II영역에서도 PID 각각의 이득값변화의 방향을 같게 놓았기 때문에 앞서 설명한 PID 각각의 특성을 위배하는 행렬상의 요소들이 존재하게 된다. 따라서 오버슈트(Overshoot)가 발생할 경우 재빨리 수렴할 수 없는 단점을 갖고 있다. Tzafestas[2]가 PID 제어를 위하여 구성한 행렬은 다음 표 1과 같다.

표. 1 Tzafestas의 퍼지 PID제어 행렬 Δ Error

	-6	-5	-4	-3	-2	-1	0	+1	+2	+3	+4	+5	+6
-6	0	0	-1	-1	-2	-2	-3	-3	-4	-4	-5	-5	-6
-5	0	0	0	0	-1	-1	-2	-2	-3	-3	-4	-4	-5
-4	+1	0	0	0	0	0	-1	-1	-2	-2	-3	-3	-4
-3	+2	+1	0	0	0	0	-1	-1	-2	-2	-3	-3	-4
-2	+3	+2	+1	0	0	0	0	0	-1	-1	-2	-2	-3
-1	+4	+3	+2	+1	0	0	0	0	-1	-1	-2	-2	-3
0	+5	+4	+3	+2	+1	0	0	0	0	0	-1	-1	-2
+1	+6	+5	+4	+3	+2	+1	0	0	0	0	0	0	-1
+2	+6	+5	+4	+3	+2	+1	0	0	0	0	0	0	0
+3	+6	+5	+4	+3	+2	+1	0	0	0	0	0	0	0
+4	+6	+5	+4	+3	+2	+1	0	0	0	0	0	0	0
+5	+6	+5	+4	+3	+2	+1	0	0	0	0	0	0	0
+6	+6	+5	+4	+3	+2	+1	0	0	0	0	0	0	0

3.3 새로운 퍼지(Fuzzy) PI제어 행렬

본 논문에서는 새로운 퍼지(Fuzzy) PI제어 행렬의 구조도와 분해도(Resolution)의 결정방법을 제시하였다.

3.3.1 퍼지(Fuzzy) PI제어 행렬의 구성방법

기존의 PI 특성은 다음과 같다.

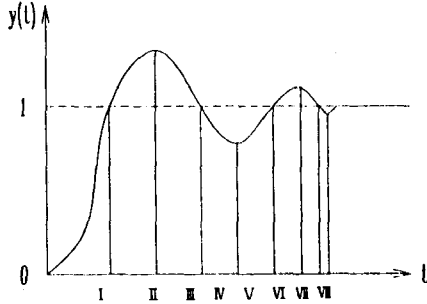


그림. 2 단위응답(Step Response) 특성

각 영역의 특성은 다음과 같다.

I, V 영역)

:P와 I의 이득값은 각 순간의 오차값에 해당하는 기울기, 즉, 오차변화(Error Changes)값을 갖도록 같은 방향으로 증가 또는 감소시킨다. 이렇게함으로써 상승시간(Rise Time)을 줄일 수 있고 오버슈트(Overshoot)를 줄일 수 있다.

II, VI 영역)

:오버슈트(Overshoot)가 발생한 경우이므로 P이득값은 증가시키고, I이득값은 감소시킨다. 결국은 원하는 입력값(Desired Input Value)쪽으로 향하게 한다.

III, VII 영역)

:부호만 반대이고 I,II)영역과 같은 작용을 한다.

IV, VIII 영역)

:II,V)영역과 같은 작용을 한다.

앞 도표에서 보는 바와 같이 각각의 영역에 따라서 P 이득과 I 이득의 증가 또는 감소하여야 할 영역은 상이하르

로 본 논문에서는 이러한 특성들을 모두 담기 위해 다음과 같이 제어행렬의 영역을 여섯개로 나누었다.

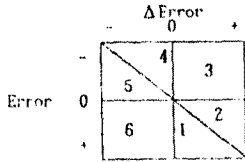


그림. 3 제어행렬 칸막이

- 각 영역의 이득조정방법은 다음과 같다.
- 1 영역) 그림 3의 I, V영역중 기울기, 즉, 오차변화(Error Changes)값이 너무 작은 경우이다.
P 이득: 증가시킨다, I 이득: 증가시킨다
 - 2 영역) 그림 3의 I, V영역중 기울기, 즉, 오차변화(Error Changes)값이 너무 큰 경우이다.
P 이득: 감소시킨다, I 이득: 감소시킨다
 - 3 영역) 그림 3의 오버슈트(Overshoot)가 발생한 영역인 II, VI영역이다.
P 이득: 증가시킨다, I 이득: 감소시킨다
 - 4 영역) I영역과 같다.
P 이득: 증가시킨다, I 이득: 증가시킨다
 - 5 영역) 2영역과 같다.
P 이득: 감소시킨다, I 이득: 감소시킨다
 - 6 영역) 3영역과 같다.
P 이득: 증가시킨다, I 이득: 감소시킨다

각각의 영역은 시스템 특성에 미치는 영향이 상이하므로 각 영역의 출력값을 시스템의 특성에 따라서 주 대역선출을 중심으로 대칭적으로 나누기 않고 각각의 영역을 독립적으로 나누고 각영역의 분해도(Resolution)를 시스템의 특성에 따라 영역별로 결정한다. 그리고 P 이득과 I 이득의 분해도(Resolution)를 깊게 높여도 시스템의 특성을 상당히 향상시킬 수 있고, 1개의 제어행렬을 구성하고 각각의 영역에 부호를 부여함으로써 기억용량을 절약할 수 있다. 결론적으로 제어행렬은 가장 중심된 값을 중심으로 각각의 영역이 대칭적인 관계를 갖게 되므로 여섯개의 영역중 세개의 영역만을 결정하여도 제어행렬을 구성할 수 있게 된다.

3.3.2 제어규칙의 설계

본 논문에서는 P, I 이득의 변화량을 결정하므로 매 샘플링(Sampling)마다 얻어지는 오차(Error)와 오차의 변화량(Error Changes)을 퍼지화(Fuzzification)하지 않고 기존 입력과의 상대적인 크기를 퍼지화(Fuzzification)함으로써 각각의 원하는 값에 상대적인 입력변수에 의해서 이득값을 변화시켰다.

$$E = (R - Y_n) / R_n = 1 - Y_n / R \quad (9)$$

$$\Delta E = E_{n-1} - E_n = (1 - Y_{n-1} / R) - (1 - Y_n / R) = (Y_n - Y_{n-1}) / R \quad (10)$$

Where, R = 기준입력
Y_n = 시스템의 현재출력
Y_{n-1} = 시스템의 과거출력

그리고 If - then 형태의 언어적인 법칙(linguistic rule)을 사용하여 퍼지제어기의 출력을 결정한다. 본 논문에서는 각각의 입력변수에 대한 언어적인 법칙(linguistic rule)을 다음과 같이 결정했다.

- PL : positive large
- PM : positive medium
- PS : positive small
- PZ : positive zero
- NZ : negative zero
- NS : negative small
- NM : negative medium
- NL : negative large

위 언어적인 값(linguistic value)을 사용하여 본 연구대상인 폐루프 시스템(Closed Loop System)에 오버슈트(Overshoot)가 줄어들고 상승시간(Rise Time)을 짧게 하는 제어규칙과 분해도(Resolution)를 앞서 설명한것과 같이 구성하였다.

본 연구에서는 제어규칙을 Look-Up Table화 하기위하여 이산적형태의 멤버십 함수(Membership Function)를 정의하면 다음 표 2 - 5와 같다.

표. 2 제어규칙

		ΔError							
		NL	NM	NS	NZ	PZ	PS	PM	PL
Error	NL	PZ	PZ	PS	PS	PM	PM	PL	PL
	NM	NS	NZ	PZ	PS	PS	PM	PM	PL
	NS	NM	NS	PZ	PZ	PS	PS	PM	PM
	NZ	NL	NM	NS	NZ	PZ	PS	PS	PM
	PZ	NZ	NS	NS	NZ	PZ	NS	NS	NL
	PS	NM	NM	NS	NS	PZ	NZ	NS	NM
	PM	NL	NM	NM	NS	PS	PZ	PZ	NS
	PL	NL	NL	NM	NM	PS	PS	PZ	NZ

(위 요소들은 P 이득변위값이고, I 이득변위변환값은 1-4분변과 3-4분변의 부호가 반대이다)

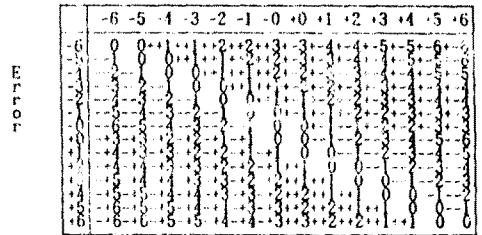
표. 3 양자화된 변수

Error (%)	ΔError (%)	ΔU	Quantized Level
-5 이하	-25 이하	-6	-6
-4	-20	-5	-5
-3	-15	-4	-4
-2	-10	-3	-3
-1	-5	-2	-2
0	0	-1	-1
1	5	0	0
2	10	1	1
3	15	2	2
4	20	3	3
5 이상	25 이상	4	4

표. 4 이산적인 멤버십 함수

	-6	-5	-4	-3	-2	-1	0	0	1	2	3	4	5	6
NL	1	0	0	0	0	0	0	0	0	0	0	0	0	0
NM	0	1	0	0	0	0	0	0	0	0	0	0	0	0
NS	0	0	1	0	0	0	0	0	0	0	0	0	0	0
NZ	0	0	0	1	0	0	0	0	0	0	0	0	0	0
PZ	0	0	0	0	1	0	0	0	0	0	0	0	0	0
PS	0	0	0	0	0	1	0	0	0	0	0	0	0	0
PM	0	0	0	0	0	0	1	0	0	0	0	0	0	0
PL	0	0	0	0	0	0	0	1	0	0	0	0	0	0

표. 5 Look-Up Table



(각 요소의 부호는 P, I gain의 부호순)

4. 시뮬레이션 및 결과

본 논문의 퍼지 PI제어를 사용한 하이브리드(Hybrid) 위치/힘제어기의 전체 블록도는 그림4와 같다.

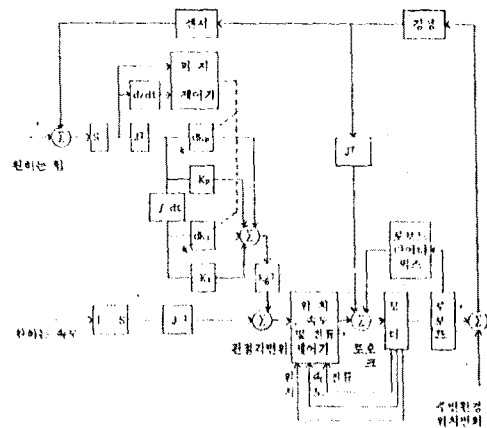


그림. 4 퍼지 PI제어를 사용한 하이브리드 위치/힘 제어기의 전체 블록도

본 시뮬레이션은 로봇 끝이 주변환경과 접촉을 이루고 있는 두축 로봇을 대상으로 하고 있으며, 로봇 끝에 발생하는 힘을 측정하기 위하여 힘 센서를 사용하고, 각 관절의 모터의 사양은 표 7과 같다. 그리고 로봇 끝과 주변환경사이의 마찰력은 무시하였다.

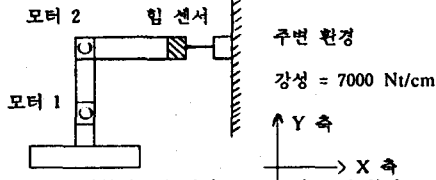


그림. 5 시뮬레이션 대상 로봇 및 주변 환경
표. 6 로봇의 각 링크의 사양

링크	길이	무게	기어감쇠비
링크1	25 cm	2 Kg	1/100
링크2	20 cm	2 Kg	1/80

표. 7 모터1, 모터2의 사양

출력	750 [Watt]
정격 속도	175 [rad/sec]
정격 전압	143 [Volt]
정격 전류	6.4 [Ampere]
Jm	36 [KgCm ²]
Bm	7.2 [KgCm ² /(rad/sec)]
Ra	2.16 [ohm]
La	0.0013 [Henry]
Kb	0.7 [Volt/(rad/sec)]
Kt	6664 [KgCm ² /sec ² /A]

시뮬레이션의 정확성을 위하여 로봇의 다이내믹스와 로봇 끝에 작용하는 반작용힘을 고려하였다.

샘플링시간을 20msec로한 시뮬레이션 결과는 그림 6에서와 같이 기존의 PI제어를 이용한 시스템의 출력특성은 상승시간(Rise Time)이 크고, 오버슈트(Overshoot)가 크게 발생하고 있다. 그리고 Tzafestas의 퍼지 PI제어행렬을 이용한 시스템의 출력특성은 상승시간(Rise Time)이 상당히 줄어드는 효과를 보이지만, 본 논문에서 제시한 문제점과 같이 오버슈트(Overshoot)가 발생하였을 경우 재빨리 대처할 수 없으므로 오버슈트가 크게 발생하였다. 그리고 본 논문의 퍼지 PI제어를 이용한 시스템의 출력특성은 상승시간(Rise Time)이 상당히 줄어들었고, 오버슈트(Overshoot)도 상당히 줄어들었다. 그림 7은 힘 출력특성이 정상상태(Steady State)일때 위치 제어기에 0.5cm/sec로 위치변화값을 입력시킨 경우이다. 여기에서는 위치변화에 의한 힘출력특성이 큰 변화없이 거의 일정하게 유지하고 있음을 볼 수 있다. 이와같이 힘 제어기와 위치 제어기사이에는 Coupling이 일어나지 않음을 볼 수 있다.

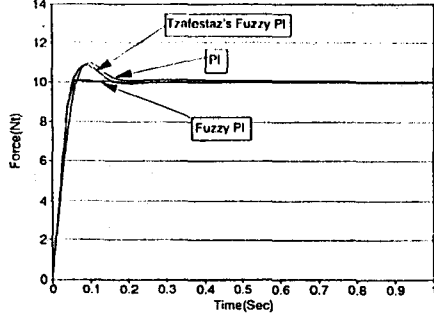
5. 결론

본 논문에서 제시한 퍼지 PI 제어를 이용하므로써 새로운 제어기를 구성하지 않고 상용의 로봇제어에 이용되고 있는 위치제어를 사용할 수 있음을 보여주었고, 또한 출력특성을 향상시키기 위하여 기존의 PI 제어에 퍼지(Fuzzy) 제어를 도입하므로써 시뮬레이션의 결과에서 보여주듯이 오버슈트(Overshoot)가 상당히 줄어들면서 상승시간(Rise Time) 또한 상당히 줄어드는 효과를 보였으며, 이는 기존의 PID제어를 이용하는 시스템에 새로운 하드웨어의 구성없이 본 논문의 퍼지이론만을 적용하여 성능을 상당히 향상시킬 수 있다. 그러나 본 논문에서의 퍼지 PI제어를 수행하기 위한 각각의 이득값의 변화정도를 나타내는 k1, k2값을 결정하는데 있어서 어려움을 겪게되므로, 앞으로 이러한 값들을 시스템 동작중에 스스로 최적의 값을 결정할 수 있는 연구가 요구된다.

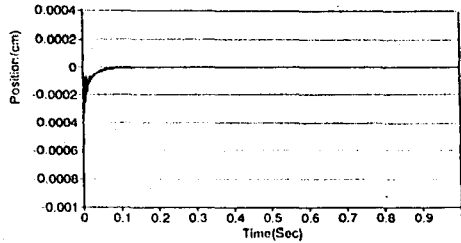
Reference

1) Raibert, M. H., Craig, J. J., "Hybrid Position/Force Control of Manipulators," Transactions of the ASME, Journal of Dynamics Systems, Measurement, and Control, The American Society of Mechanical Engineers, June 1981, vol. 102, pp.126-133.

2) S. Tzafestas, "Incremental Fuzzy Expert PID Control," IEEE Transactions on Industrial Electronics, vol. 37, No. 5, Oct. 1990.
3) L. A. Zadeh, "Fuzzy Sets," Information of the Control, vol. 8, pp. 338-353, 1965.
4) L. A. Zadeh, "Outline of a New Approach to the Analysis of Complex Systems and Decision Processes," IEEE Transaction on Systems, Man, and Cybernetics, vol. SMC-3, no. 1, pp. 28-44, 1973.
5) P. J. Macvicar-Whelan, "Fuzzy Sets for the Man - Machine Interaction", 1976, Int. J. Man-Mach. studies, vol. 8, pp.687 - 697.

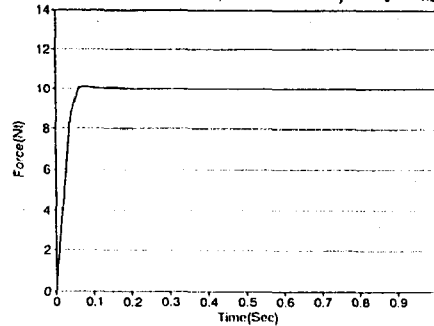


나) 힘 출력특성

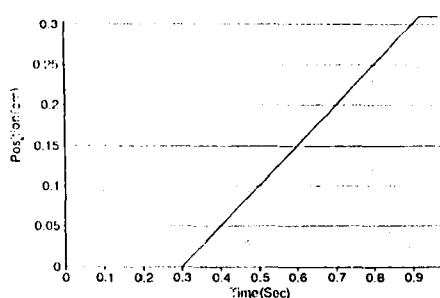


비) 위치 출력특성

그림. 6 힘변위명령은 10N이고, 위치변위는 현재의 위치를 그대로 유지하는 경우의 PI 제어, Tzafestas의 퍼지 PI 제어, 본 논문의 퍼지 PI제어의 시스템 출력특성의 비교. (초기자세=(135, -90), $K_p=0.38, k_1=0.5, k_2=0.07, k_2=0.2$)



나) 힘 출력특성



비) 위치 출력특성

그림. 7 힘 명령은 10N을 유지하고, 위치명령은 0.5cm/sec로 수행한 경우. (초기자세 = (135, -90), $K_p = 0.38, k_1 = 0.5, k_2 = 0.07, k_2 = 0.2$)