

フレキシブルアームのファジィ制御

2003MM061水野 祐樹

指導教員:高見 勲

1 はじめに

エキスパートシステムとしての認識が高いファジィ制御は、ルールベースの制御と呼ばれ高い適応性を持ち、非線形の制御対象でも制御できることから注目されている。本研究では、フレキシブルアームをファジィ制御により安定させることを考える。そこで目標値付近の振動を抑制するために、積分補償を用いたファジィコントローラを用いる。最終的にシミュレーションと実験結果より理論の有効性を示すことを目標とする。

2 制御対象

本研究では図1に示す制御対象を用いる。このフレキシブルアームはモーターによって、アームの先端角度を制御するものである。

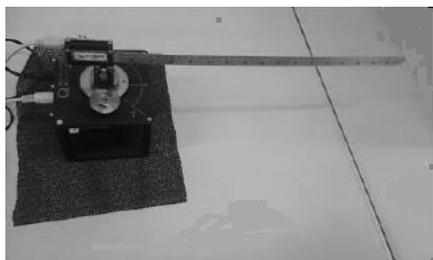


図1: フレキシブルアーム

線形化した数学的モデルを次に示す。

$$\dot{x}(t) = Ax(t) + Bu(t), \quad y(t) = Cx(t)$$

$$x = \begin{bmatrix} \theta & \alpha & \dot{\theta} & \dot{\alpha} \end{bmatrix}^T$$
$$A = \begin{bmatrix} 0 & 0 & 1.0000 & 0 \\ 0 & 0 & 0 & 1.0000 \\ 0 & 708.9 & -37.02 & 0 \\ 0 & -1154 & 37.02 & 0 \end{bmatrix}$$
$$B = \begin{bmatrix} 0 \\ 0 \\ 65.1134 \\ -65.1134 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 1 & 0 & 0 \end{bmatrix}$$

3 ファジィコントローラの設計

3.1 ファジィ制御

ファジィとは言葉の意味や概念の定義など人間の主観性に非常に深く関わるあいまいさを指している。ファジィ制御は、「 θ のとき α する」という人間が経験的に蓄積するものを基に、複数のルールを利用し、コントローラを作成する手法である。特徴として、ルールを基にコントローラを考えるので、モデルが不完全であっても制御できることが挙げられる。また、膨大なルールを処理するために計算量が非常に多くなるという欠点も持つ。

3.2 ファジィ推論

本研究の推論では、まず2入力1出力型のコントローラを設計することから、2つの入力メンバーシップ関数と1つの出力メンバーシップ関数を作成した。入力メンバーシップ関数は、角度 $\theta + \alpha$ についてであり、もう1つは角速度 $\dot{\theta} + \dot{\alpha}$ についてである。また出力メンバーシップ関数はモーターへの電圧についてである。これらのメンバーシップ関数を「IF-THEN～」形式で表現されたファジィルールによって結びつける。ファジィ制御ではこのルールの組み合わせによって制御を行う。前件部と呼ばれるIFの部分で入力である角度と角速度の適合度を計算し、THEN以降の後件部によって出力である電圧の操作量を決定する。またシミュレーションの演算方法は、min-max法という一般的なものを使用する。

4 積分補償ファジィコントローラ

第5章の5.1のシミュレーション結果より、第3章で作成した標準ファジィコントローラでは理想的な波形を得ることができたが、第6章の6.2の実験結果より、実験では目標値とずれ、振動が起きてしまった。これはメンバーシップ関数やファジィルールの試行錯誤によるチューニングによって、ある程度は減らすことができるが、完全に0にすることは難しいことが分かった。

そこで角度が目標値付近に近づいたときに、角度 $\theta + \alpha$ の目標値($\pi/4$ [rad])と観測値の偏差の積分をする入力メンバーシップ関数を追加した。また、積分をする条件を角度 $\theta + \alpha$ の偏差が ± 0.04 [rad]の範囲の時のみ考慮することにした。さらに、このメンバーシップ関数を考慮する時は目標値付近(角度がZOの場合)であることから、大きな出力は必要ないため、3つのファジィ集合によって作成した。ファジィコントローラのルールを次のように変更し、3入力1出力型のコントローラにした。

- 角度 θ がNB, NS, PS, PBの時
角度 θ と角速度 $\dot{\theta}$ の2つのメンバーシップ関数から出力メンバーシップ関数を決定する。
- 角度 θ がZOの時
角度 θ と角度 $\theta + \alpha$ の積分の2つのメンバーシップ関数から出力メンバーシップ関数を決定する。

これにより、通常3つの入力メンバーシップ関数からなるファジィルールは、 $5 \times 5 \times 3 = 75$ 個のファジィルールが必要となるが、目標値付近でのみ考慮することによって、ルール数は23個にすることができる。よって、計算量の多さに影響を与えずにすむ。この2つの条件から作成したファジィルールを表1に示す。

		角度				角度 ZO		
		NB	NS	PS	PB			
角 速 度	NB	PB	PB	PS	ZO	角 度 の 積 分	NB	PS
	NS	PB	PS	ZO	NS		ZO	ZO
	ZO	PB	PS	NS	NB			
	PS	PS	ZO	NS	NB			
	PB	ZO	NS	NB	NB			

表 1: ファジィルール

5 シミュレーション

5.1 標準ファジィ制御

一般的なメンバーシップ関数とファジィルールで設計した2入力1出力型のファジィコントローラでシミュレーションを行った。角度 $\theta + \alpha$ の目標値を45°とした時のステップ応答を図2に示す。オーバーシュートやオフセットをすることなく目標値に追従することができた。

5.2 積分補償ファジィ制御

第4章で作成した、積分の入力メンバーシップ関数を追加した3入力1出力型のファジィコントローラでシミュレーションを行った。同様に角度 $\theta + \alpha$ の目標値を45°とした時のステップ応答を図3に示す。このコントローラでも、オーバーシュートやオフセットをすることなく目標値に追従できた。

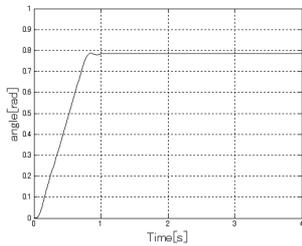


図 2: 標準ファジィ制御のシミュレーション結果

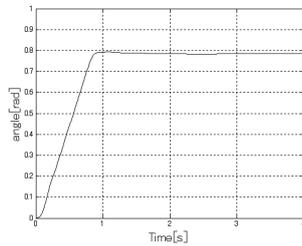


図 3: 積分補償ファジィ制御のシミュレーション結果

6 実験

6.1 実験への適用

シミュレーションにおいて理想的な出力を得ることができたことから、実験器具への適用を試みる。シミュレーションでは、MATLAB/SimulinkのFuzzy Logic Toolbox¹を使用した¹が、実験では使用できないので、同様のアルゴリズムとなるコントローラを作成した。

6.2 標準ファジィ制御

まず第3章で作成したコントローラで実験を行う。この2入力1出力型のファジィコントローラでは、シミュレーション結果と比べ、立ち上がりまでの時間が1.5

倍近くかかり、大きく振動してしまった。そこで、まず立ち上がり時間を早くするため、出力メンバーシップ関数の間隔を2倍にした。また、オーバーシュートを抑止するために、入力メンバーシップ関数1(角度)の幅を広くしてブレーキがかかるように変更した。その時の角度 $\theta + \alpha$ の実験結果を図5に示す。実験結果を見ると、立ち上がりの時間が早くなり、オーバーシュートも無いが、目標値付近での振動は残ってしまった。

6.3 積分補償ファジィ制御

次に第4章で作成したコントローラで実験を行う。実験で行ったメンバーシップ関数を図4に示す。

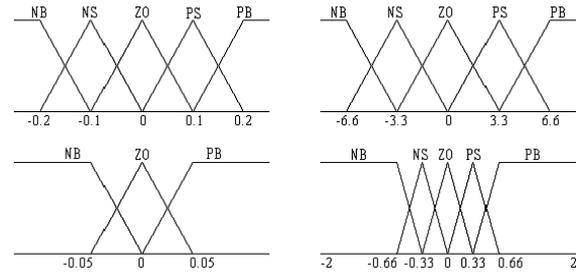


図 4: 入力(左上:角度 θ [rad], 右上:角速度 $\dot{\theta}$ [m/s], 左下:積分[rad*s]), 出力(右下:電圧[V])

この3入力1出力型のファジィコントローラでの、角度 $\theta + \alpha$ の実験結果を図6に示す。実験結果を見ると、オーバーシュートもせず目標値に追従させることができた。

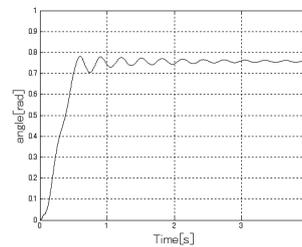


図 5: 標準ファジィ制御の実験結果

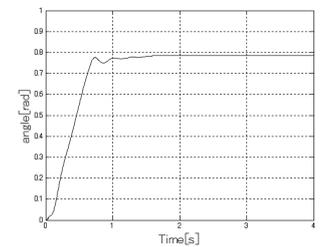


図 6: 積分補償ファジィ制御の実験結果

7 おわりに

シミュレーション結果では見られなかった振動が、実験結果では見られたが、部分的に積分補償を追加することで、振動を抑止することができた。また積分についてのメンバーシップ関数のファジィ集合を3つにして、ファジィルールの数を減らしたが、制御性を損ねることなく安定化することができ、理論の有効性を実証できた。

参考文献

- [1] 田中英夫：ファジィモデリングとその応用, 朝倉書店(1990).
- [2] 宮地貴広：積分補償型ファジィコントローラによる制御設計, 南山大学数理情報学部数理科学科(2005).

¹MATLAB/Simulinkは米国 the Math Works 社の登録商標