

ニューラルネットによる倒立振子の安定化制御

2002MM028 稲葉 寿久

指導教員 高見 勲

1 はじめに

PID 制御における比例、積分、および微分の各ゲインは経験的に求められることが多く、多大な時間を要している。このため K_p, K_i, K_d の 3 つの PID ゲインを自動的に求めるセルフチューニングによる PID 制御系の設計について研究が行われてきた。本研究では倒立振子モデル 505 を制御対象とし、ニューラルネットによる学習を用いた PD 制御におけるゲインの設定を行った。

2 制御対象

2.1 倒立振子

図 1 が本研究の制御対象である。水平バランシングロッドを操作して垂直振子ロッドを制御する倒立振子である。



図 1 倒立振子

2.2 倒立振子の制御

倒立振子の制御に用いたブロック線図を MATLAB/Simulink^{*1} を使って図 2 に示す。バランソロッドの中心からの位置 x についての制御 (伝達関数 $x(s)/F(s)$ の制御) を内側のループ (副ループ) で閉じる。位置 x から振子の角度 θ についての制御 (伝達関数 $\theta(s)/x(s)$ の制御) を外側のループ (主ループ) で行う [1][2]。

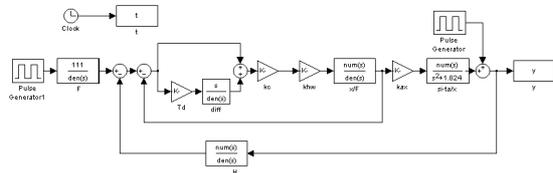


図 2 ブロック線図

3 ニューラルネットの制御への応用

3.1 ニューロ PID

ニューラルネットとは人間の脳の仕組みをまねた学習アルゴリズムの一種である。人間の脳はニューロンと言う細胞が多数繋がったネットワーク構造になっており、こ

のネットワークを単純化、定式化したのがニューラルネットである [3]。図 3 の、閉ループの目標値を $r(n)$ 、偏差を $e(n)$ 、操作量を $u(n)$ 、出力を $y(n)$ とする。ニューラルネットの入力は、操作量 $u(n)$ と出力 $y(n)$ の 2 個である。またニューラルネットの出力は、 K_p, K_i, K_d の PID パラメータであるから、出力層のニューロン数は 3 つである [4]。

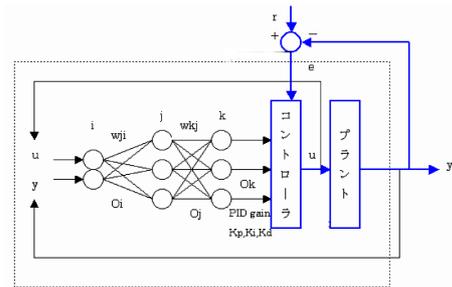


図 3 ニューロ PID

オフラインでの学習の目標は偏差の二乗和である誤差面積を最小にすることとし、オンラインでの学習の目標は偏差を最小にしていこうとする。このとき誤差逆伝播法を用いる。手順は下記の通りである。

手順 1: 初期設定

入力層のニューロン数 $M (= 2)$ 、中間層のニューロン数 N 、結合強度 W_{ji} 、閾値 $\theta_j (= W_{j0})$ 、結合強度更新の重み、慣性項 η, α 、学習終了限界 ϵ 、を設定する。

手順 2: 操作量 $u(n)$ 、出力 $y(n)$ を読み込む。

手順 3: ニューラルネットの出力の計算。

$$O_j = f(\text{net}_j), \text{net}_j = \sum_{i=0}^M W_{ji} O_i - \theta_j$$

$$\theta_j = W_{j0}, f(x) = \frac{1}{1 + \exp(-x)} \quad (1)$$

$$O_k = \text{net}_k, \text{net}_k = \sum_{j=0}^N W_{kj} O_j,$$

$$i = 1, 2 \quad j = 1, 2, 3 \quad k = 1, 2, 3 \quad (2)$$

手順 4: PID 制御の操作量と出力の計算。

$$u(n) = u(n-1) + K_p(e(n) - e(n-1)) + K_i e(n) + K_d(e(n) - 2e(n-1) + e(n-2)) \quad (3)$$

手順 5: 誤差の計算と誤差面積の導出。

(1) オフライン

$$e(n) = r(n) - y(n), E_n = \frac{1}{2} \sum_{k=0}^n e(k)^2 \quad (4)$$

*1 MATLAB/Simulink は theMathWark 社の登録商標

(2) オンライン

$$e(n) = r(n) - y(n), E_n = \frac{1}{2}e(n)^2 \quad (5)$$

$E_n \leq \epsilon$ ならば終了。

手順6: ニューラルネットの誤差への計算。

$$\Delta W_{kj} = W_{kj}(new) - W_{kj}(old) = -\eta \frac{\partial E_n}{\partial net_k} \frac{\partial net_k}{\partial W_{kj}} \quad (6)$$

$$\Delta W_{ji} = W_{ji}(new) - W_{ji}(old) = -\eta \frac{\partial E_n}{\partial net_j} \frac{\partial net_j}{\partial W_{ji}} \quad (7)$$

手順7: 結合強度の更新。

$$W_{ji}(new) = W_{ji}(old) + \Delta W_{ji} + \alpha \Delta W_{ji}(old), \quad (8)$$

$$W_{kj}(new) = W_{kj}(old) + \Delta W_{kj} + \alpha \Delta W_{kj}(old), \quad (9)$$

手順2へ戻る。

3.2 結合強度 (初期値) の決め方

乱数により結合強度の初期値を設定すると、学習開始時のゲインも乱数に近くなる。もし学習が不安定なゲインから始まった場合、誤差面積は収束できず正しく学習できない。そこでまず、事前に制御理論で求めたゲインでシミュレーションを行い、そのときの操作量 $u(n)$ と出力 $y(n)$ をニューラルネットの入力とし、ニューラルネットの出力目標を制御理論で求めた K_p, K_d として学習をさせる。結合強度の初期値を学習によって求めることにより、発散しないゲインから学習できるだけでなく、学習時間も抑えることができる。

4 シミュレーション、実験結果

4.1 副ループでの学習 (オフライン)

副ループでのシミュレーション結果を図4に示す。大きさ1のパルスを2秒間入力した。学習は初期設定 $N = 3$ 、 $\eta = 0.3$ 、 $\alpha = 0.5$ とし、 W_{ji} は3.2節の方法により求めた値で行った。左は学習前 $K_p = 0.2647$ 、 $K_d = 0.0086$ 、右は学習後 $K_p = 1.0581$ 、 $K_d = 0.0070$ のパラメータを用いて、2.2節の制御設計を行った。図4から学習の目的である誤差面積の縮小は達成できたことがわかる。

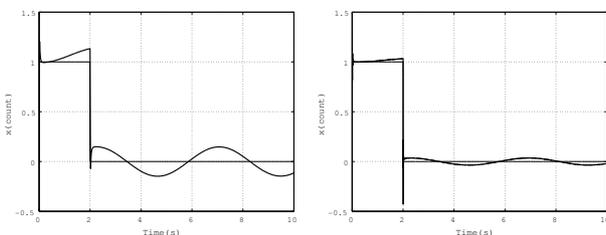


図4 副ループでの学習 左:学習前 右:学習後

4.2 主ループでの学習 (オフライン)

主ループでのシミュレーション結果を図5, 図6に示す。目標応答では大きさ1のパルスを2秒間入力した。外乱応答では入力を0とし、大きさ1のパルスを2秒間外乱

として入れた。学習は初期設定 $N = 3$ 、 $\eta = 0.5$ 、 $\alpha = 0.7$ とし、 W_{ji} は3.2節の方法により求めた値で行った。図5は学習前 $K_p = 0.1445$ 、 $K_d = 0.1196$ 、図6は学習後 $K_p = 0.5006$ 、 $K_d = 0.0052$ のパラメータを用いて2.2節の制御設計を行った。学習により誤差面積は減った。

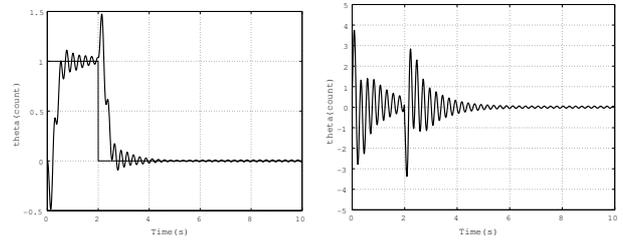


図5 主ループでの学習 (学習前) 左:目標応答 右:外乱応答

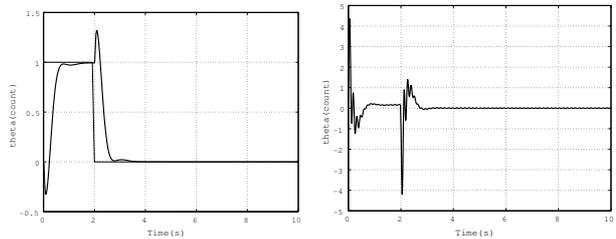


図6 主ループでの学習 (学習後) 左:目標応答 右:外乱応答

4.3 実機での実験結果 (オフライン)

図7に実験結果を示す。制御設計を2.2節の方法で行い、パラメータは4.2節と同じである。500カウントのパルスを2.5秒間入れた。速応性が向上したことがわかる。

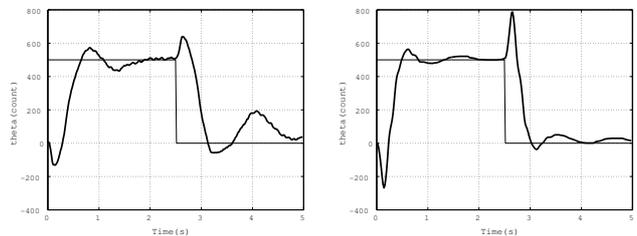


図7 実機での実験 (オフライン) 左:学習前 右:学習後

5 おわりに

本研究では倒立振子のPDゲインをニューラルネットを用いてよりよいパラメータを求めた。オフラインでの学習ではシミュレーションを繰り返しその都度、偏差の二乗和が小さくなるように学習し、オンラインでは偏差の二乗が小さくなるように学習することで求めた。また、発散しない学習開始時の結合強度の初期値を別の学習によって求めた。

参考文献

- [1] モデル505 マニュアル,ECP.
- [2] 川田昌克、西岡勝博: MATLAB/Simulink によるわかりやすい制御工学, 森北出版 (2001).
- [3] 熊沢逸夫: 学習とニューラルネットワーク, 森北出版 (1998).
- [4] 大松繁: ニューラルネットワークによるPIDゲインの自動調節, システム制御情報チュートリアル講座 2005,(2005).