

ファジィ論理とニューラルネットワークを用いた ライトレースカーの制御

2017SC093 吉田裕基

指導教員：陳幹

1 はじめに

今現在ロボットは世界中で使われており、その技術は日々進歩し、構造は複雑化している。ゆえに、これからのロボットの制御にはより柔軟に対応できる AI 技術の導入が有用である。本研究は先行研究 [1] をもとにファジィ制御を付け加えることでより柔軟な制御結果の実現を試みた。

2 ロボットの運動モデル

図 1 のロボットを想定し、使用される変数は

$$v_r = r\dot{\phi}_r, v_l = r\dot{\phi}_l, \quad \omega = \frac{(v_r - v_l)}{2d}, v = \frac{(v_r + v_l)}{2},$$

$$\dot{x}_m = v \cos \theta, \dot{y}_m = v \sin \theta, \quad \dot{\theta} = \omega.$$

とする [2]。表 1 に変数の内容を示す。

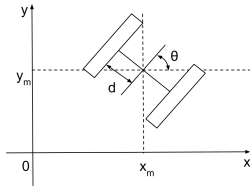


図 1 ライトレースカーの構造

表 1 車両モデルのパラメータ

変数内容	変数	単位
右ホイールと左ホイール速度	v_r, v_l	m/s
ホイール半径	r	m
右ホイールと左ホイール角度	ϕ_r, ϕ_l	rad
マシン速度	v	m/s
マシン進行方向角度	θ	rad
マシン進行方向各速度	ω	rad/s
マシン重心位置	(x_m, y_m)	m
車輪から車体中心までの距離	d	m

モータの運動は

$$\ddot{\phi}_r = -K_1\dot{\phi}_r + K_2u_r, \ddot{\phi}_l = -K_1\dot{\phi}_l + K_2u_l,$$

$$u_r = u_0 + u, u_l = u_0 - u$$

とする。 K_1, K_2 はモータの物理定数であり、 u_0 はバイアス電圧である。また、 u_r, u_l は左右のモータに入力する電圧であり、 u はコントローラから出力されるモータへの入力である。センサは接地面に対し [白, 黒] = [1, 0] を返すものがロボット前面に 3 つついている。3 つのセンサから誤差の値 $e_1(k)$ を表 2 にしたがって決める。

表 2 センサの仕様

誤差 $e_1(k)$	検知結果 $y(k)$
3	$[1, 1, 1] \& 0 < e_1(k-1)$
2	$[0, 1, 1]$
1	$[0, 0, 1]$
0	$[0, 0, 0]$
-1	$[1, 0, 0]$
-2	$[1, 1, 0]$
-3	$[1, 1, 1] \& e_1(k-1) < 0$

3 ニューラルネットワーク

構造を図 2 に示す [3]。ただし、今回は図 2 の中央と下側のラインのみを使用する。シンプルなセンサからの情報

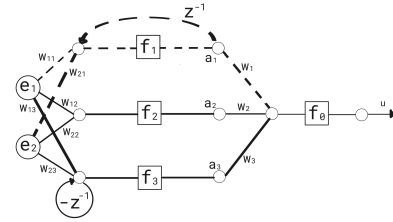


図 2 ニューラルネットワークの構造

を補完するために、ニューラルネットワークへの入力に新しく誤差 $e_1(k)$ の勾配 $e_2(k)$ を定義する。 $e_2(k)$ は

$$e_2(k) = \left[\frac{1}{2} \quad 0 \quad -\frac{1}{2} \right] [e_1(k) \quad e_1(k-1) \quad e_1(k-2)]^T \quad (1)$$

である。活性化関数は全て $f_1(x) = f_2(x) = f_3(x) = f_0(x) = x$ とする。評価関数 $J(k)$ は

$$J(k) = J_1(k) + J_2(k) + \gamma_1 J_3(k) \quad (2)$$

$$J_1(k) = \frac{1}{2} e_1(k)^2, J_2(k) = \frac{1}{2} e_2(k)^2 \quad (3)$$

$$J_3(k) = \frac{1}{2} \frac{u(k-1)^2}{\gamma_2 (e_{1MAX}^2 - e_1(k)^2) + 1} \quad (4)$$

とし、重み $w_{ij}(i: 1, 2)(j: 1, 2, 3)$ の更新は

$$w_{ij}(k) = w_{ij}(k-1) + \Delta w_{ij}(k-1) \quad (5)$$

$$\begin{aligned} \Delta w_{ij}(k-1) = & \eta_{ij} [e_1(k) \operatorname{sgn}\left(\frac{e_1(k) - e_1(k-1)}{u(k-1) - u(k-2)}\right) \\ & + e_2(k) \operatorname{sgn}\left(\frac{e_2(k) - e_2(k-1)}{u(k-1) - u(k-2)}\right) \\ & + \left\{ \frac{\gamma_2 e_1(k) u(k-1)^2}{\{\gamma_2(9 - e_1(k)^2) + 1\}^2} \operatorname{sgn}\left(\frac{e_1(k) - e_1(k-1)}{u(k-1) - u(k-2)}\right) \right. \\ & \left. + \frac{u(k-1)}{\gamma_2(9 - e_1(k)^2) + 1} \right\} e_i(k-1)] \end{aligned}$$

で行う。 η_{ij} は学習率である。

4 ファジィ推論

前件部と後件部のファジィ集合を図3 図4 に示す。ファ

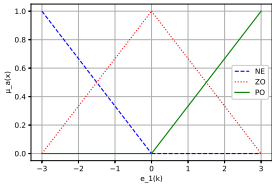


図3 前件部

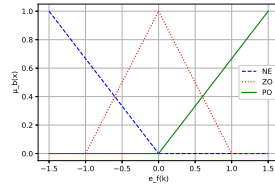


図4 後件部

ジールールは表3 に示す。前件部メンバーシップ関数

表3 ファジィルール

前件部	後件部
NE	NE
ZO	ZO
PO	PO

$\mu_a(x)$ には誤差 $e_1(k)$ を代入する。推論法は min-max 法を使用する [4]。ただし、計算を簡略化するため、図4の後件部ファジィ集合は離散化したものを使用する。よって、非ファジィ化で得られる重心 x_0 は

$$x_0 = \frac{\sum x \mu_b(x)}{\sum \mu_b(x)} \quad (6)$$

となる。

5 シミュレーション

シミュレーションは「Zumo Robot for Arduino,v1.2」のモデルを使用した [5]。ニューラルネットワークコントローラを用いたシミュレーションと、センサにファジィ推論を用いたニューラルネットワークコントローラのシミュレーションを行った。結果は図5 図6 である。

また、評価関数を

$$J_s = \sum e_1(k)^2 \quad (7)$$

と定義し、それぞれのシミュレーションの評価値を表4 に示す。PDNNC のシミュレーションは 2575.0, PDNNC+ファジィのシミュレーションは 1064.0 であった。

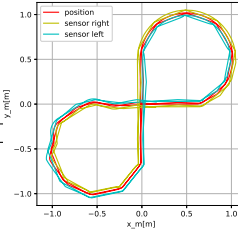


図5 ニューラルネットワークコントローラの結果

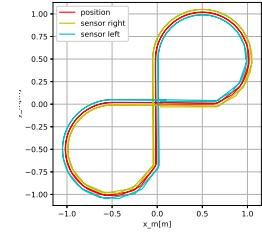


図6 ファジィ+ニューラルネットワークコントローラの結果

表4 シミュレーションの評価値

シミュレーション	評価値
ニューラルネットワークコントローラ	2575.0
ファジィ+ニューラルネットワークコントローラ	1064.0

6 おわりに

センサの出力を定式化した $e_1(k)$ に対し、ファジィ推論を行ったところ制御性能の向上が確認できた。今後はより柔軟なファジィ則及びファジィ集合の発見と、ニューラルネットワークの精密な構築が課題である。

参考文献

- [1] Madoka.Asai, Gan.Chen, Isao.Takami(2018). "Neural network trajectory tracking of tracked mobile robot", *2019 16th International Multi-Conference on Systems, Signals & Devices (SSD'19)*, pp.225-230.
- [2] 山川聡子 (2019). 「移動ロボットを用いて制御理論を学ぶ実験カリキュラム」, 『計測自動制御学会論文集』, vol.55, no.7, pp.457-465.
- [3] Shuang.Cong, Guodong.Li, Beichen.Ji(2005). "A novel PID-like Neural Network Controller", *16th IFAC World Congress*, vol.38, no.1, pp.121-126.
- [4] 水元雅晴 (1989). 「ファジィ制御向きのファジィ推論法」, 『計測と制御』, vol.28, no.11, pp.959-963.
- [5] "Pololu Zumo Shield for Arduino User's Guide", *Pololu Robotics&Electronics*, <https://www.pololu.com/docs/0J57>, 最終閲覧:2022-1-18.