

マイクロマウスの走行位置の制御

2016SC091 鳥井量平

指導教員：大石泰章

1 はじめに

近年、小型で動力源を持つロボットが、自律的に迷路を探索し、ゴールまで達する時間を競うマイクロマウス競技が注目されている [1]。この競技は、機械工学、センシング、制御など様々な技術を必要とするので、その知見は、自動車の自動運転や衝突回避の研究に役立つと考えられる。

本研究では、マイクロマウスが迷路内を壁に衝突せずに、スムーズに走行できるようにすることを目標とする。具体的には、マイクロマウスに取り付けられた左右の光距離センサの特性を解析し、これにもとづいて、左右の壁の有無に関わらず、迷路の通路の中心を通るように走行させることを試みる。

2 マイクロマウスとは

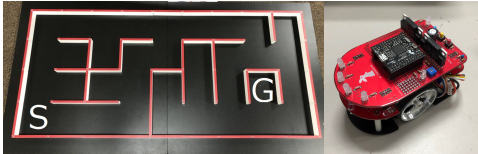


図1 実験で用いた迷路

図2 実験で用いたマイクロマウス

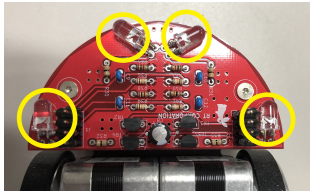


図3 マイクロマウスの光距離センサ

現在マイクロマウス競技は、迷路の大きさによって分類されており、 $18 \times 18\text{cm}$ を1区画として 16×16 区画の大きさの迷路を用いるクラシックマウス競技と、 $9 \times 9\text{cm}$ を1区画として最大 32×32 区画の迷路 (通常は 16×16 区画で全国大会のファイナルのみ 32×32 区画) を用いるマイクロマウス競技がある [1]。区画の壁の高さは 5cm 、厚さは 1.2cm である [2]。

競技内容を説明する。スタート地点は、迷路の四隅のいずれかが決められた場所とし、ゴールは迷路中心の4区画のいずれかとする。制限時間10分以内に5回走行でき、1回ごとにスタート地点からゴール地点までのタイムを計測する。また、途中で止まるなどによりゴールできなかった場合は、リタイアになる。5回走行するか、制限時間が尽きたときに、走行した中の最短のタイムが記録され、採用される [1][2]。

本研究では、迷路として、 4×4 区画のクラシックマウス競技サイズのベースを2つ並べたものを使う (図1)。図1中のSはスタートを、Gはゴールを表している。また、マイクロマウスとして、株式会社アールティのマイクロマウス学習キット Pi:co Classic3 を使用する (図2)。このマイクロマウスは、CPU (Central Processing Unit, 中央処理装置) を含むマイコンユニット、センサユニット、モータユニット、電源ユニットなどで構成されている。

マイクロマウスのセンサユニットには、壁の有無や壁との距離を読み取るために反射型光距離センサが用いられている。特に、前壁を見るセンサが2つ、横壁を見るセンサが左右1つずつで合計4つのセンサがついている (図3)。センサの角度が重要であり、横壁を見るセンサは真横ではなく、 45° 傾けた。これは、マイクロマウスが壁に対して平行でない状態で走行しても対応できるようにするためである。前壁を見るセンサは、前壁の認識が安定するように外に少し傾けた [1]。

3 走行位置の制御

サンプルプログラムのままでは、直進走行中に右壁がなくなると、マウスが通路の中心から右にずれて走行することが多い。このように、通路の中心からずれて走行すると壁と衝突して現在位置を見失うことがあるので、通路の中心を走らせる制御が必要である。特に、通路の両側に壁があることもあれば、右側または左側だけに壁があることもあるので、それぞれの場合に対応しなくてはならない。

通路の中心を走らせるための制御は、文献 [1] に従って次のようにする。すなわち右モータへの入力を u_r 、左モータへの入力を u_l 、正の比例定数を K とかくとき、

$$u_r = u_0 + K\Delta, \quad (1)$$

$$u_l = u_0 - K\Delta. \quad (2)$$

ただし、偏差 Δ は左右の光距離センサの出力値から計算される量で、マウスが通路の中心にあるときは、 $\Delta = 0$ 、中心よりも右側にあるときは $\Delta > 0$ 、左側にあるときは $\Delta < 0$ になるようにする。

偏差 Δ の計算法を考えるため実験を行う。マウスを迷路中の両壁のある通路に置き、通路の中心にあるときその位置を $x = 0$ 、右寄りにあるとき $x > 0$ 、左寄りにあるとき $x < 0$ で表す。様々な位置 x の値に対して、マウスの左右の光距離センサの出力値を測定したものを図4に示す。青線が右センサの出力値 y_r 、赤線が左センサの出力値 y_l である。

図4から、マウスの位置が $x = 0$ のときの右センサの値 y_{r0} と左センサの値 y_{l0} は等しくないことがわかる。そこ

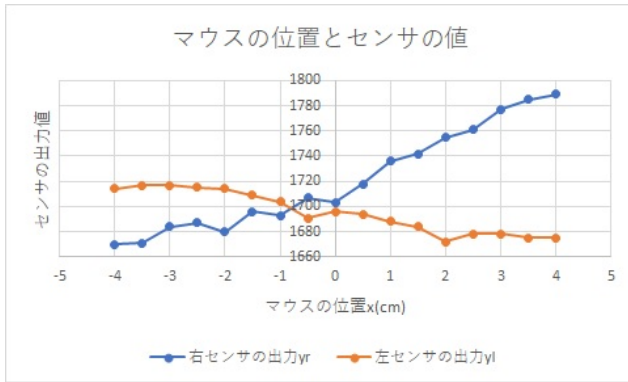


図4 マウスの位置とセンサの変化

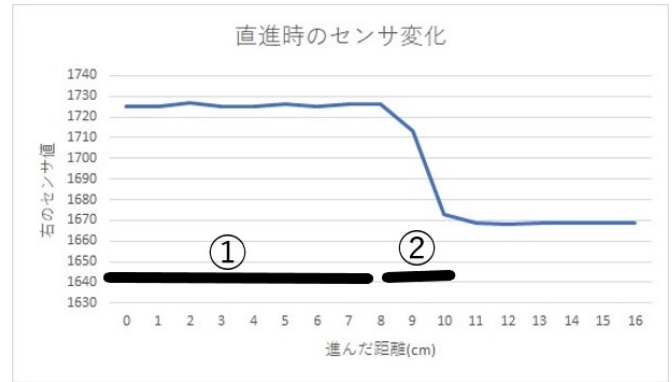


図5 直進時の右センサの出力値の変化

で、偏差 Δ を計算するときは、右センサについては値 y_{r0} との差、左センサについては y_{l0} との差を使うことにした。すなわち両壁がある場合は

$$\Delta = (y_{l0} - y_l) + (y_r - y_{r0}), \quad (3)$$

左壁があり、右壁がない場合は

$$\Delta = 2(y_{l0} - y_l), \quad (4)$$

左壁がなく、右壁がある場合は

$$\Delta = 2(y_r - y_{r0}) \quad (5)$$

とする。

4 壁の有無の判定

4.1 基本の判定法とその問題

壁の有無を判定するためには、マウスを両壁のある迷路内で左端に寄せた時の右センサの値、右端に寄せた時の左センサの値を、壁があると判断する限界値として設定する。この設定した値よりも、左右のセンサの出力値が大きければ壁があると判定し、小さければ壁がないと判定する。しかしこの判定法を用いて実験すると、右壁があるときは通路中央を走っていても、右壁がなくなると右に寄ってしまうことが多い。この現象の原因を調べるため、次節で実験を行う。

4.2 壁がなくなるときのセンサ出力値の変化

壁の切れ目における右センサの出力値の変化を調べるため、両壁がある状態から右壁がない状態へと1cmずつ移動させた際の右センサの出力値を記録した(図5)。

図5の横軸は基準の位置からマウスの進んだ距離を、縦軸は右センサの値を示している。距離が0~8cmの間では右壁があるが、8~16cmでは右壁はない。このグラフから、右センサの値は右壁がなくなっても直ちに限界値を下回るわけではないことがわかる。したがって図の②の区間では、まだ右壁があると判定し、例えば迷路の中心を走行したとしても、右センサの値 y_r が右センサの目標値 y_{r0} よりも小さくなることから偏差 Δ は負となり、マウスは迷

路の中心よりも左側にあると判断してしまう。これによりマウスは右側へ動こうとするため、迷路の中心よりも右にずれてしまうと考えられる。

4.3 問題の解決策

4.1節の問題を解決するために、右センサの値の変化量の絶対値が、ある一定値よりも大きくなった場合に、右壁があると判定する限界値を、右センサの目標値 y_{r0} へ引き上げることとした。これにより直ちに右壁がなく左壁がある場合に切り替えることになる。

以上の変更のもとで、3区画右壁がない状態を直進走行する迷路を使用し、走行実験を行ったところ、変更前は右へ大きくずれたのに対して、変更後は迷路の中心近くを走行した。

5 まとめ

本研究では、左右の壁の有無に関わらず迷路の通路の中心を走行できるように制御方法を変更した。またセンサの出力値の変化量に着目することで、壁の有無の判定法を改良した。今後の課題は、左右の光距離センサの特性に留意し、距離と光距離センサの関係式を求めることで、より精度を高めることができると考えられる。また、走行中に角度の修正をできるようにし、誤差をより少なくできるようにすることである。

参考文献

- [1] 株式会社アールティ「ロボコンマガジン」編集部：『マイクロマウスではじめようロボットプログラミング入門』。株式会社 オーム社、東京、2018年。
- [2] 公益財団法人ニューテクノロジー振興財団：「マイクロマウスクラシック競技規定」。2013年8月1日。www.ntf.or.jp/mouse/micromouse2013/kitei-classic-since2013.html。