

1. 目的

近年、私たちの身の回りで人工知能が使われている。本実習では、人工知能を用いてロボットにブランコのこぎ方を学習させる方法と仕組みを考えて人工知能の概要について理解する。

2. 予備知識

(1) 人工知能について

人工知能の概念として「人間が行う知的行動や作業をコンピュータで模倣するシステム」を示し、「Artificial Intelligence (人工的な知能)」といい、英語の頭文字をとって A I と呼ぶことが多い。人工知能の開発は、第二次世界大戦後の 1950 年頃始まり、大きく 3 つの方式が開発され、現在に至っている。

● 推論探索システム

最初に開発された人工知能のシステムで、考えられる選択肢を試行しながら結論を探し出していくという方式である。この方法は、選択肢が多くなればなるほど答えを見つけ出すのに時間がかかること、当時のコンピュータの性能が今ほど高くなかったこと、用途が限定されることなどから適用範囲はそれほど広くない。

● エキスパートシステム

文字どおりエキスパート（専門家）のシステムであり、ある特定の分野に関する問題の情報とそれに基づく処理の手順をあらかじめシステムとして組み込んだものであり、利用者との対話によって答えを導きだしていく方式である。

「推論エンジン」と呼ばれる処理を行う部分と、判断基準となる情報である「知識」を分けていたため、「知識」の情報を更新することで精度を向上できるという点で推論探索システムより優れている。しかし、「知識」は私たち人間が更新する必要がある点、専門家が考えることを定型化することが難しい点など課題も多い。

● 機械学習

推論探索やエキスパートシステムでは、人工知能の精度を向上させるために私たち人間が知識を与えなければならなかった点に対して、2006 年にコンピュータ自身が知識を習得できる「ディープラーニング (深層学習)」が発明され、人工知能の精度は飛躍的に向上することになる。「機械学習」はコンピュータ自身が知識を習得する仕組みのことで、ディープラーニングの一つである。

機械学習による人工知能が進化した背景に、機械学習のアルゴリズムが開発されたことである。また、知識のもととなる莫大な情報の流通を可能にしたインターネットの普及や、その情報を処理できる高い計算能力を持ったコンピュータの開発が機械学習の考え方を生み出した一つの要素でもありとされている。

(2) 人工知能と機械学習

機械学習はコンピュータがデータを分析して処理するアルゴリズム（情報の処理手順）を搭載したものである。コンピュータが学習する方法として大きく「教師あり学習」と「教師なし学習」に分けられる。

- 教師あり学習

情報と答えをセットでコンピュータに与える学習方法。認識精度は短時間で向上させられる一方で、答え付きの情報を人間が用意しなければならないために手間がかかる。

- 教師なし学習

答えを与えずに情報だけ与え、多くのコンピュータが分析、分類を繰り返して自分で答えを見つけ出す方法。精度向上に時間はかかるが、データさえあれば私たち人間の手間はほとんどかからないことや、私たち人間が考えなかったことをコンピュータが発見する可能性が考えられる。

- ※ 強化学習

コンピュータに試行錯誤させて、その達成度に対して評価が与えられ、コンピュータはより高い評価を得るために、行動を修正することで目標に近づけるような学習方法。ある環境においてコンピュータが現在の状態を観測し、取るべき行動を決定しながら答えを見つけていくという方法で、人間が答えを与えないという意味で「教師なし学習」である一方で、一定の基準があるという意味で「教師あり学習」とどちらともとらえることができる方法である。

3. 協議

(1) 人工知能と私たちの生活

近年進化をしている人工知能が私たちの生活の中でどのように役立っているか考える。まず、私たちの生活で人工知能が使われている場面を協議する。

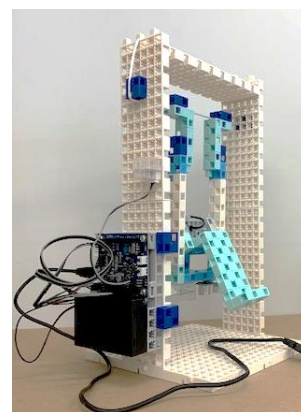
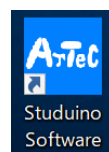
(2) (1)で協議した場面をもとに、人工知能を活用した製品を提供する側、および利用する側においてどのようなメリットがあるかについて協議する。

4. 実習

(1) ブランコを漕ぐプログラム

A I ブランコロボット（右図）を用いる。ロボットがブランコを漕ぐためにどのように動かせばよいかを考える。

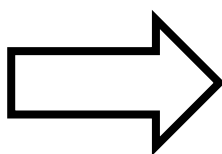
ロボットを動かすためにプログラムを作成する必要がある。プログラムを作成するためのソフトウェアを起動する。パソコンのデスクトップ（画面）上にアイコンがあるため、このアイコンをマウスでダブルクリックする。



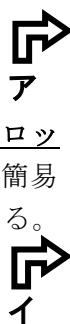
A I ブランコロボット

- 初期設定

① ダブルクリック後、以下のようにプログラミング環境を設定する。



② 右図のような画面が表示されることを確認する。プログラムは Scratch と呼ばれるプログラミング学習環境をベースとした環境で作成を行う。コンピュータへの命令を表すアイコン（ブロック）をマウスのドラッグ&ドロップで簡易的にプログラムを作成することができる。画面の構成は以下のとおりである。



- ア カテゴリ
- イ ブロックパレット
- ウ スクリプトエリア

③ ブランコロボットの制御（基本）

制御の基本として、ロボットの足とつながっている「サーボモータ」を動かす。ブロックパレットから部品を取り出して下図のようなプログラムを作成する。



（手順１）

カテゴリの「制御」を選択する。ブロックパレットから「ずっと」のブロックをスクリプトエリアまでドラッグ&ドロップする。

（手順２）

カテゴリの「動き」を選択する。ブロックパレットから「サーボモータ」のブロックを選択してスクリプトエリアにあるブロック「ずっと」の隙間部分にドラッグ&ドロップする。

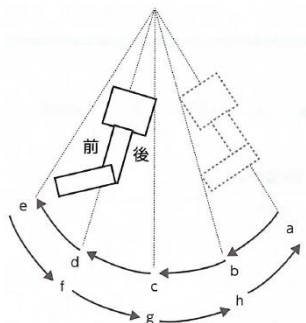


（手順３）

（手順１）と（手順２）を参考にして左図のようにブロックを組んでプログラムの作成を行う。

(2) ブランコロボットの観察

ロボットがすぐにブランコを漕ぐことはできない。ロボットがブランコを漕ぐ動作のイメージを下図に示す。a～hはそれぞれロボットが揺れているときの位置を示しており、 $a \Rightarrow b \Rightarrow c \Rightarrow d \Rightarrow e$ は前向きに揺れる動きを、 $e \Rightarrow f \Rightarrow g \Rightarrow h \Rightarrow a$ は後ろ向きに揺れる動きを示す。



aが後ろに揺れ切って静止している状態、eが前に揺れ切って静止している状態、cとgがもっとも地面に近くなる状態、b、d、h、fは揺れている途中の状態、隣接する各地点の時間間隔を一定になるようにし、それぞれの地点で足を上げ下げすることで、ブランコロボットを揺らすように制御する。足の上げ下げについて以下に例を示す。

<例1> a～hの各地点において足を交互に上げ下げする組み合わせを考える。

h点	g点	f点	e点	d点	c点	b点	a点
下げ	上げ	下げ	上げ	下げ	上げ	下げ	上げ

<例2> 前後もっとも振れ幅の大きい地点だけ足を上げる組み合わせを考える。

h点	g点	f点	e点	d点	c点	b点	a点
下げ	下げ	下げ	上げ	下げ	下げ	下げ	上げ

このように、異なる足の動かし方でロボットを動かした場合、ロボットの動き方も異なる。それぞれのロボットは異なる性質を持つものとなり、足の動かし方がロボットの性質を決定する「遺伝子」と考えることができる。

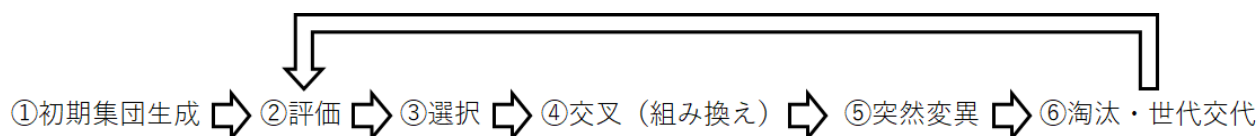
<例2>の遺伝子の情報を数値化するために足を上げる動作を「1」、下ろす動作を「0」として、各遺伝子の情報を数値化すると下表となる。

h点	g点	f点	e点	d点	c点	b点	a点
0	0	0	1	0	0	0	1

この動きをまとめて2進数へ数値化させると $(00010001)_2$ となり、10進数に変換すると $(17)_{10}$ である。このように、足の動かし方を数値化して、複数のパターンを準備することで複数の遺伝子を持つ個体を作ることができる。この個体をもとにして次の手順で遺伝的アルゴリズムの処理を行う。実際の動作を観察し、ロボットがブランコを漕いでいることを検証する。

(3) プログラム処理手順

プログラムは「Scratch(スクラッチ)」をベースとしたブロックプログラムを用いている。

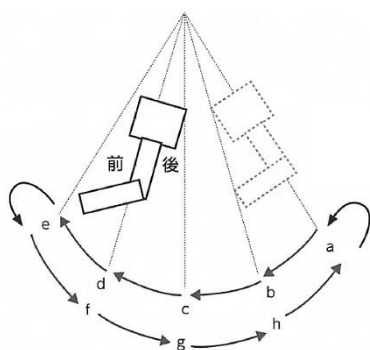


① 初期集団生成

足の動かし方のパターンを 10 種類ランダムに作成する。これによって 10 種類の遺伝子を持つ個体となる。＜例 1＞、＜例 2＞のように足の上げ下げの動きを「0」と「1」の数値化（2 進数）として扱う。個体の数は任意で指定できるが、ブランコロボットでは 10 個の個体を用意している。ここでは以下のように数値を作成したものとする。

数値(10 進)	数値 (2 進) 8 ケタ	h	g	f	e	d	c	b	a
6 6	0 1 0 0 0 0 1 0	下	上	下	下	下	下	上	下
2 1 7									
1 6 4									
1 1									
8 9									
1 3 7									
2 1 4									
2 1 3									
1 6 8									
1 6 1									

② 評価

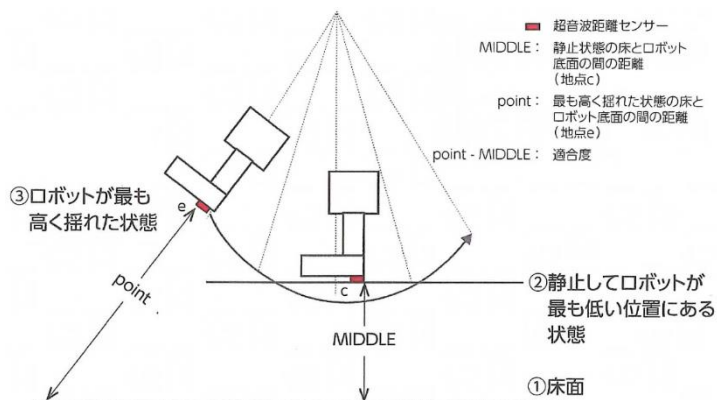


各個体が持つ遺伝子（足の動かし方）を試行し、揺れ幅を数値化して評価する。この数値化は「ジャイロセンサ」を用いて検出してブランコロボットの足を動かすためのタイミングを決定している。左図において、地点 a から揺れはじめ、地点 e で折り返し、再度地点 a に戻るまでがブランコの周期となる。

地点 a では、ブランコの揺れる向きが後ろから前になることを利用して、ジャイロセンサで揺れる向きが変わったことを検出する。検出したタイミングでタイマーをスタートして、次の地点 a を検出した時間の差によってブランコの一周期の時間を計算することができる。

揺れの大きさはロボットの底面につけられた「超音波センサ」によって数値化している。超音波センサは床面に向いているため、床とロボットとの距離を測定することができる。

揺れが大きいほどロボットと床の距離は離れるため、超音波センサが示した距離が離れているほど、大きく揺れていると判断することができる。

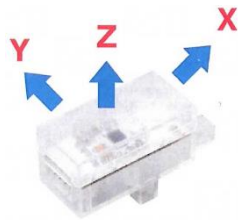


静止状態のロボットと床の距離と、最も高く揺れた状態で測定した距離の差を適合度としている。

- ・適合度が低い場合 ⇒ 揺れ幅が小さい。
- ・適合度が高い場合 ⇒ 揺れ幅が大きい。

※ ジャイロセンサと超音波センサ（下図）

ジャイロセンサは物体の回転や向きの変化を角速度（回転角度）として検出し、電気的信号に変換させる素子である。超音波センサは基板に超音波の送信部と受信部を設け、超音波が物体で反射して戻ってくる時間を利用して距離を測定する素子である。



ジャイロセンサ

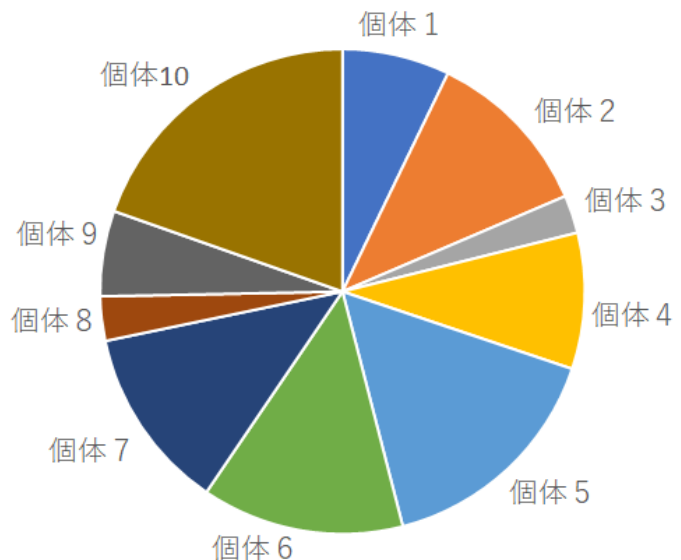


超音波センサ

③ 選択

10種類の個体の中から、2つの個体をランダムに選び出す。この時、適合度の高い個体が選ばれやすいように選択される確率を高くし、適合度の低い個体が選ばれにくいように選択される確率を低くして適合度に応じた選択を行う。

選択方法として「ルーレット方式」で「重み付きランダム抽出」という方法を用いる。右図のように適合度の高い個体については的を広く、低い個体は的を狭くしてルーレットの的（円グラフ?）のように見立てる。



適合度によって選択される個体の面積比が異なる状態をつくり、このルーレットに対して乱数で2つの個体を選択する。ここで、適合度の高い個体は確率的に選ばれやすいという意味で、必ずしも選ばれるわけではない。各個体が選択される確率は以下の式である。

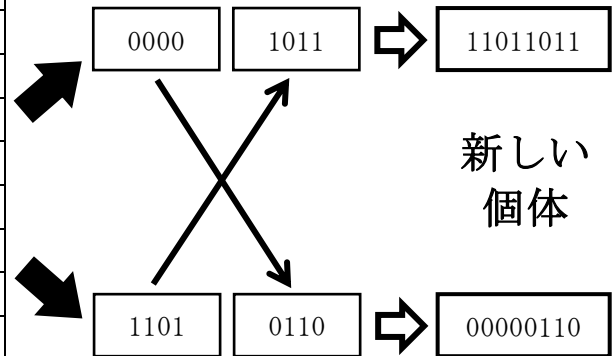
$$\text{各個体が「選択」される確率} = \frac{\text{各個体の適合度}}{\text{各個体の適合度の和}}$$

④ 交叉（組み換え）

選択された2つの個体の遺伝子を交叉させて、新たな遺伝子をもつ個体を作成する。選択された個体の数値を2進数で示し、その数値を任意の場所で2つに分け、2つの個体の上半分とした半分をつなぎ合わせて新しい個体を作成する。

<例> 個体1～個体10がもっているそれぞれの数値から個体4と個体7の交叉をする。

個体名	数値(10進)	数値(2進)
個体1	66	
個体2	217	
個体3	164	
個体4	11	00001011
個体5	89	
個体6	137	
個体7	214	11010110
個体8	213	
個体9	168	
個体10	161	



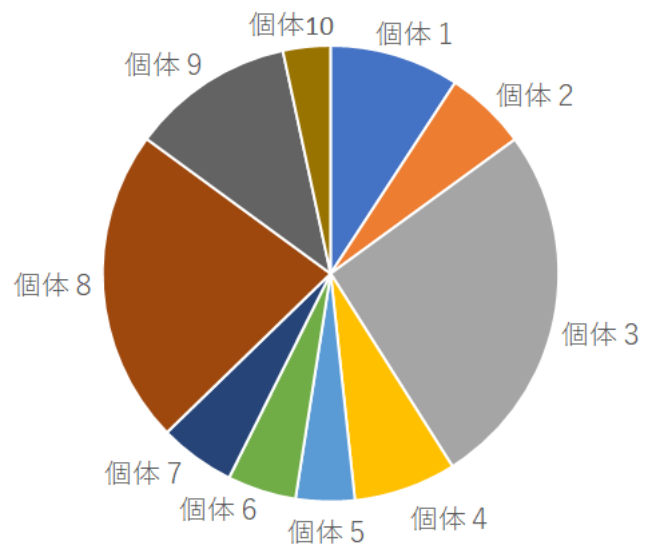
⑤ 突然変異

新しく生まれた2つの個体に対して一定の確率で、足の動かし方を変更する。地点aで足を上げるとなっていた個体が地点aでは足を下げるようになる、という変化を起こす。突然変異を加えることによって、最適化が遅れる場合がある一方、進化の方向が偏ることを防ぐことができる効果がある。

ロボットには8個の各数字に対して、値を変更するかどうかを確率5%の乱数で決定し、突然変異する桁の数字が「0」の場合は「1」、「1」の場合は「0」(反転)に変更して突然変異を発生させている。

⑥ 淘汰(世代交代)

新しく生まれた2つの個体と、すでにある2つの個体を入れ替える。この時、入れ替えの対象となる個体も「ルーレット方式」で選択するが、③の方法と逆に適合度の低い個体を選ばれやすいように処理を実行する。したがって、適合度が小さいほど選ばれやすいルーレットが必要となるため、確率は以下の式となる。



$$\text{各個体が「淘汰」される確率} = \frac{\text{各個体の適合度の逆数}}{\text{各個体の適合度の逆数の和}}$$

これによって、適合度の低い個体を淘汰し、新しい個体に置き換えを実行している。

②(評価)から⑥(淘汰・世代交代)までの処理をあらかじめ指定した分だけ繰り返し実行する。一定の世代交代を繰り返した後、もっとも適合度の高い組み合わせ(遺伝子)を持つ個体が、ブランコを漕ぐということに最も適合していると判断できる。

5. 考察

- (1) ロボットの駆動部（モータ）を動かすだけのプログラムを実行したときのロボットの動きについて観察した結果を記録する。

- (2) 遺伝アルゴリズムをもとに組まれたプログラムを実行したとき、次の段階について観察した結果を記録する。

- (3) L E Dが点灯したときのロボットはどのような状態か記録する。

6. 振り返り