

7D-10 classifier system による「猫と鼠」追跡ゲーム

堂田 敏文 上田 雄悟 星野 力

筑波大学 構造工学系

はじめに

classifier system は classifier と呼ばれる多数のルールが集まりであり、そのルールの強度を学習するシステムである [1]。本研究では、classifier system(以下 CS)に遺伝的アルゴリズム(以下 GA)により新しいルールを発生させ、「猫と鼠」と名づけたゲーム環境において学習効果を調べた。

ゲームルール

図1にゲームシステムを示す。ゲームを管理しているのは GameMaster であり、2人のプレイヤーはそれぞれ Entry1, Entry2 というエントリープログラムを用意する。ゲームのルールは

1. Entry1 を猫、Entry2 を鼠とし、鬼ごっこをさせる。
2. 猫は鼠を捕まえると、得点を得る。
3. 両者ともに同じ量の体力を初期設定してあり、それを使い切ると死ぬ。
4. 相手が死ぬと、生き残ったものが得点を得る。
5. どちらかが得点を得ると1セット終了し、立場を入れかえ次のセットを行なう。
6. GameMaster とエントリープログラムが一回通信するのを1ステップとし、1セットは70ステップまででそれまでに終わらない時は引き分けとなる。
7. 1ステップごとにエントリープログラムには GameMaster から猫と鼠の現在の状況入力があり、それをもとに移動のための力 $\langle fx, fy \rangle$ を出力する。

ここでは猫の方が鼠より足が速いが、小回りが効かず体力の減りも早く設定してある。各々の体力は、 $\langle fx, fy \rangle$ と鼠をとるアクションに応じて減少する。

classifier system

1つのエントリープログラムを図2のように CS と GA を用いて作成した。CS は

if $\langle message == 条件部 \rangle$ then $\langle action 部 \rangle$

の形をとる。環境からの入力(自分と相手の座標、速度、残り体力)は Message (相手との距離、方向、速度、残り体力等)に $\langle 0, 1 \rangle$ の形で変換される。Message は条件部とパターンマッチングを行なう。各 classifier の条件部は $\langle 0, 1, \# \rangle$ からなり、action 部は $\langle 0, 1 \rangle$ からなる。条件部の # は don't care で相手が0でも1でもマッチするので、許容度の高いマッチングができる。Message と条件部がマッチした classifier は発火する。classifier は強度と呼ばれるパラメータをもっており、その classifier の有効性を示す。発火した classifier 群から相対的強度に比例した確率で1つの classifier (行動 classifier) が選ばれ、その action を $\langle fx, fy \rangle$ に変換して環境に出力する。

classifier system の学習のメカニズム

行動 classifier の強度は、ある一定の係数 c ($0 < c < 1$) をかけられ、その値は一時的に減少する。表1のような報酬条件があらかじめ設定してあり、各ステップごとにその行動 classifier が条件を満たしたかを判定し、条件に応じた一定の報酬を得る。また猫が鼠を捕まえた時には、Bucket Brigade アルゴリズムに従い、その行動 classifier だけでなく、それ以前数ステップの間の行動 classifier にも報酬が与えられる。

新しいルールの生成

GA は生物の遺伝、適応、進化をモデルにしたアルゴリズムで [1]、ここでは classifier の多様性を保ち、新しい classifier を生成する目的で使用する。GA は 3000 ステップに一度起動する。また

$$GP = (\text{強度が初期値よりも大きくなっている classifier 数}) / (\text{全 classifier 数})$$

とし、GP によって淘汰圧を変化させている。次の世代に残す classifier の割合を PP としたとき、PP は

$$GP \leq \text{MinP のとき } PP = \text{MinPP}$$

$$\text{MinPP} < GP < \text{MaxPP のとき } PP = GP$$

$$\text{MaxPP} \leq GP \text{ のとき } PP = \text{MaxPP}$$

(ただし PP の最大最小を MinPP, 最大値を MaxPP として設定する。)

とする。強度の順に classifier をソートし低い順に (1 - PP) まで淘汰された後、生き残った PP の中から両親をランダムに選び、classifier を遺伝子として交叉と突然変異により新しい classifier を 1 つ生成する。これをもとの classifier 数に戻るまで行なう。また、ある Message に対して classifier 群が 1 つも発火できなかったステップ数の割合に比例して突然変異率を上げている。これにより有効な classifier が少ないときには、多くの新しい classifier を生成し、増えてくるとそれに応じて、淘汰圧を下げるようにしてある。

対戦と結果

このような CS+GA を用いたエントリプログラムを他のプログラム (猫は鼠のあとを追いつけ、鼠はランダムに動き続けるもの) と対戦させてみた。ゲームの進行状況は、Xwindow 上でリアルタイムに表示した。対戦初期の classifier はランダムであり、動きはデタラメであるが、学習を続けることによって次第に鼠を追う猫に成長していった。ランダムに動く相手には、どの相手にも通用すると思われるルール (相手との距離が極めて近い時に捕まえる行動を起こす等) を学習して捕まえることができた。なお、この研究は筑波大学工学システム学類自由科目「知的ソフトウェアコンテスト」の準備作業として行なった。

参考文献 D.E.Goldberg, "Genetic Algorithms in Search, Optimization, and Machine Learning", Addison Wesley, (1989)

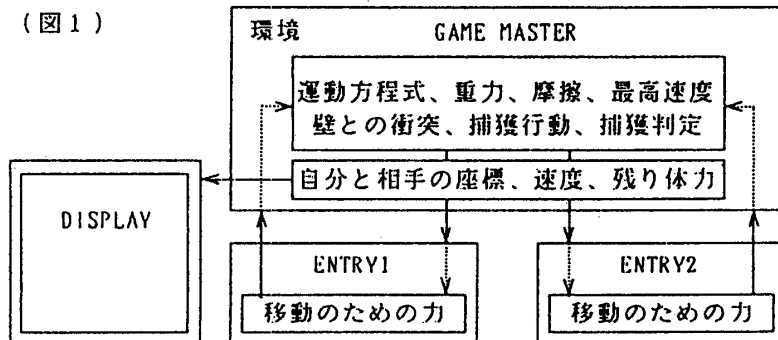


表 1

猫	相手との距離を縮めたとき (5 段階) かつ相手との距離がある値以下のとき (3 段階)
鼠	相手と危険な距離にいてもその状況下で逃げ続けているとき 相手との距離をのばしたとき 危険な距離から離脱したとき 相手との距離が離れているときの行動で危険な距離に移動しなかったとき 体力の減りが少ないとき

