

MAS を用いた「ゲーム」—— 魚雷ゲーム (単純版、通常版、対戦版)

鈴木一敏

「実際にいじってみる」というのが MAS のプログラミングを覚える早道だと思います。しかし、全くの初学者は何をいじればいいのか分からないですし、コンピューターが単純なルールを実行するのを見るだけでは、すぐに飽きてしまうでしょう。特に、マルチエージェント的な面白い現象は、ほんの限られた条件下でしか見られないこともしばしばです。「改造したら、詰まらない結果しか出なくなった」なんて日常茶飯事です。これでは、やる気も失せるというものです。

全くの初心者でも試行や改造そのものに興味を持てるようなモデル、というのが意外に大切なのではないのでしょうか？

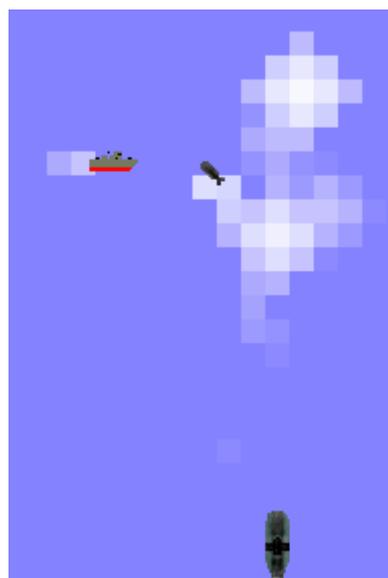
そこで本稿は「ゲーム」の利用を提案します。インタラクティブな「ゲーム」ならば実行者が実際に遊べるため、興味が持続しやすいでしょう。また、MAS のモデルはオブジェクト指向なので、部分的な改造が簡単なものをつくれる利点もあります。改造を施せばプレイ内容への影響がすぐに実感できますから、ルールを変えてもっと面白くしよう、という意欲もわきやすいはずです。

魚雷ゲームの概要

潜水艦から魚雷を発射、誘導して敵の工作船や駆逐艦の突破を阻止します。規定数以上の敵に作戦海域を突破されるとゲームオーバーです。また、工作船に体当たりされたり、駆逐艦に砲撃を受けて撃沈されてもゲームオーバーとなります。

コントロールパネルの再生ボタンを押すとゲームが始まります。発射ボタンを押して魚雷を発射してください。魚雷の舵はコントロールパネルのスライドバーで操作します。発射後にスライドバーにフォーカスを移しておいて、キーボードの矢印ボタンで操作するのが良いでしょう。

魚雷は複数発射できますが、舵は全て同じ方向にしか切れません。敵が複数出てきたときには、時間をずらして発射するなどの工夫が必要になります。



魚雷ゲームの特徴

このゲームを作るうえで注意した点がいくつかあります。

① MAS の関数をできる限り多く利用すること

初心者が学習するためのモデルですから、多くのパターンを使っていることが望まれます。そういった要請に応えるため、基本版では文字列型を除くすべての種類の関数を使用しています。ヘルプを見ただけでは使い方が分かりにくい関数でも、似た関数を実際に使っている部分をまねることで、ある程度使用できるようになるでしょう。

また、「エージェントを作成し、そのエージェントに情報を与える」「周りの情報を参照する」「外部ファイルの情報を読み書きする」など、よく使われる基本的なパターンもできる限りひとまとまりに書いて説明をつけました。

② プログラムが簡潔であること

前の項目と矛盾するようですが、初心者でも理解しやすいように、非常に少ない行数でも動く構造になっています。たとえば、プレイヤーの指示通り走行するだけの魚雷エージェントなら 4 行、さらに付近に敵がいたら捕捉して消し去る機能を付けるのならそれに 5 行プラスするだけで十分動作します。「単純版」は、動作の原理が理解しやすいように、極限まで機能を削ってあります。ここから改造を始めるのも良いかもしれません。

③ 改造しやすいプログラムであること

プログラムのどこで何をしているのかを分かりやすくするため、「何かにぶつかったら爆発する」「魚雷を見つけたら回避する」「自分の向きによってアイコンを変える」などできる限り機能ごとに独立して書いてあります。もし、「ぶつかっても爆発しない」としたければ該当部分を削除するだけで良いわけです。

人が操作する MAS モデル

このゲームの一番大きな特徴は、MAS のコントロールパネルを使って実行中に人間が入力を行うことです。これまで、人が操作する MAS モデルはあまりなかったように思います。もちろん、下らないから誰も作らなかつたという面もあるかもしれませんが、それでもなお、「MAS でこんなコトもできるんだ！」という例を示すことは重要だと思います。例を見ることで、新しいモデルの着想を得る人もいるかもしれないのですから。

実際、この手法は研究にも用いることができるかもしれません。たとえば、様々な戦略を持つエージェントに「囚人のジレンマ」を行わせてスコアの優劣を競うシミュレーショ

ンがあります。この競争の中に実際に人間が参加するようなモデルを作ったらどうでしょうか？ 観察者の視点から、シミュレーションに参加するマイクロな視点に入り込むことができます。あなたは、「しっぺ返し」のような単純なルールに勝てるでしょうか？ 人間は、何を基準にどのような反応をするのでしょうか？ 想像力を働かせれば、他にも活用例はたくさん見つかることでしょう。

そうした想像力の出発点として、いろんなタイプのサンプルルールを作ることは有用でしょう。こうした考えから、「人が操作する」という点をもう一步推し進めて、LAN 等を使って二人で対戦できる魚雷ゲームも作ってみました。2 つの MAS を連動させる試みです。フォルダの中にプレーヤー1 用(魚雷)、プレーヤー2 用(不審船)の二つの abs ファイルと、情報のやりとりを使うテキストファイルが入っています。LAN などをつかって、フォルダに直接アクセスし、2 つの abs ファイルを同時に実行状態にすると対戦が始まります。魚雷が不審船に追いついた時点でゲームオーバーです。

基本的な構造は以下の通りです。プレーヤー1 が魚雷を移動させ、テキストファイルに位置等の情報および自分のターンが終了したことを書き込みます。プレーヤー2 は、相手の位置を読み込んで表示し、不審船を移動させたあと、自分の位置情報とターン終了をファイルに書き込みます。これが連続して行われるようになっていきます。ひとつの MAS モデルを作るよりも多少手間がかかりますが、仕組み自体は単純です。別のモデルに組み込むことも容易にできるでしょう。

おわりに

本稿では、初心者でも試行や改造そのものに興味を持てるようなモデルを作ること、さらには、サンプルルールのより一層の多様性を確保することを目標に、魚雷ゲーム(単純版、通常版、対戦版)を紹介しました。エージェントが独立して自律的に行動し、結果的に思いがけない秩序を作ること、マルチエージェントシミュレーションの醍醐味です。しかし、MAS を使ってできることの引き出しは多いに越したことはありません。エージェント同士の秩序の中にヒトが入り込むことによって、新たな醍醐味が発見されるかもしれないのですから。