

第九回 空間の活用と時間の管理

前回復習と今回の趣旨

前回は、文字列型変数を用いることで、エージェントに複雑な個性を持たせることを学びました。そして、アクセルロッドの『文化伝播モデル簡易版』を作成してもらいました。課題では、オリジナルの文化伝播モデルを作ってもらいました。(トライしてみました?)

今回は、これまで沈黙してきた「空間」にひと働きしてもらいます。

また、モデルのなかでの時間の流れ(同期)について、少し厳密に考えてもらいます。

今回は、森林火災モデルを作成します。基本的なルールは、周りに燃えている木があると、自分も燃え上がるというルールで表現することができます。シンプル版は、HP から DL できます。みなさんには、もう少し複雑な、入り組んだモデルを作成してもらいます。

空間に変数を設定する

GetHistory

時間の流れを考える(同期)

今回作成するモデルでは、木が燃えると周辺の「空間」の温度があがり、熱が上昇するものとします。空間の熱は周囲に伝播します。そして、空間の温度があがると、燃えていない木も燃え上がるものとします。

[第一工程] モデルの枠組みを作る

- [01] 空間を作ります(名称は Forest、20*20 としましょう。他の設定はデフォルトのままでよいです)。木エージェントも作りましょう(名称は Tree、エージェント数は 0 のままでよいです)。
- [02] 今回のモデルは、コントロールパネルで木の本数を定めることができるものとしましょう。本数を表す変数(名称は N_Trees、整数型)を Universe の下に作ります。そして、コントロールパネル設定も忘れずにおきましょう。本数の範囲は、0 - 200 本としましょう。(コンパネ設定できるのは、Universe 下の変数のみです。)
- [03] 設定した本数だけ、木を生み出すルールを(Universe_Init のルールとして)書いてみましょう。もうそろそろ慣れてきてヒントが無くても書けるのではないのでしょうか。(レジュメのどこかに書いておきます。わからない人は探してね。)

- [04] 生みだされた木が配置されるルールを書いてみましょう。前は、RandomPutAgtsetCell を用いて、神様 (Universe) がばらまくやりかたを学びました。今回は、木が自分でばらばらに散らばるルールを書いてみましょう。そのために MoveToCenter() と MoveToSpaceOwnCell(整数) という関数を用います。
- [05] MoveToCenter は空間の真ん中へ動くというルールです。MoveToSpaceOwnCell(整数あ) は、自分の周りの視野 (あ) の範囲で空いている空間を見つけてそこに動くというルールです。(分居モデルのときに出てきましたね) Agt_Init のルールに、

```
MoveToCenter()  
MoveToSpaceOwnCell(7)
```

- [06] では、出力設定をしておきましょう。木のマーカは にでもしておいてください。

(注 !) 現在、MoveToSpaceOwnCell には障害があって木が重なってしまいます。なので、木の本数をいくら増やしても、密度 100% のびっしりした森はできません。来月には、治る予定なのでしばしお待ちください。

[第二工程] 空間に変数を設定する

- [07] ここで空間に働いてもらいましょう。artisoc では、空間の各セルにも変数を設定することができます。空間変数といいます。さっそく、空間に各セルの温度を表す変数を設定してみましょう。まず、空間を選択しておいて、変数を追加します (名称 Temp、実数型にしておいてください)。ツリー構造をよく確認しましょう。空間の下に (Universe でもエージェントでもなく) 変数ができていますか？
- [08] さらに空間変数についても出力設定をしておきましょう。出力設定で、すでに作成したマップ出力を編集します。マップ要素を追加して、出力対象としてさきほど追加した変数を選択してください。マーカーは「なし (= 全面表示になります)」にして、変数の範囲は 0 ~ 500 にしておいてください。そして、色はオレンジにしておいてください。

[第三工程] 木が燃え始めるルールを作ります

- [09] 木に燃焼状態をあらわす変数 (名称 Condition、整数型) を追加します。変数は、燃焼状態 (未燃、燃焼中、既燃) とともに、色 (緑、赤、黒) もあらわすことにします。
- [10] 木エージェントの初期ルール (Agt_Init) で、最初の燃焼状態を決めるルールを書き

ます。一本だけ火が付いているものとしましょう。エージェントは、それぞれ ID(背番号) を持っています。(ツリーで確認してください。0 から振られています) ID0 の木だけ、燃えている状態で始まることにしましょう。

[11] Condition にしたがって、色が表示されるように、出力設定しておきましょう。

```
If My.ID == 0    then
    My.Condition = Color_Red
else
    My.Condition = Color_Green
End if
```

保存したら、出力してみてください。

[11] Condition にしたがって、色が表示されるように、出力設定しておきましょう。

[第四工程] 木の行動ルールを書きます (1)

[12] 木の行動ルールは、「自分が燃えていたらその場の温度を上げる (暖める)」「周りが熱くなると燃え上がる (燃える)」「ある一定時間燃えると燃え尽きる (燃え尽きる)」という 3 つの部分でできているものとしましょう。

[13] 「暖めるルール」からいきましょう。燃えている木 (赤木) は、自分の存在するセル (格子) の温度を 200 度ずつ上げていくものとしましょう。空間変数 (今回のモデルでは Universe.Forest.Temp) は、セルの数だけ存在するので、座標を指定しておく必要があります。「暖めるルール」は以下のようになります。

```
If      My.Condition == Color_Red      then
    Universe.Forest.Temp(My.X, My.Y, 0) =
    Universe.Forest.Temp(My.X, My.Y, 0) + 200
End if
```

ステップ実行をして、燃えている木が空間を暖めている様子を確認しましょう。

[14] ちょっと都合もあって次は「燃え尽きるルール」に行きます。燃えている木 (赤木) も、一定期間燃え続けると、木が燃え尽きる (黒木) というルールです。最初のころにやったように、木に自分でカウントさせても良いです。ここでは、artisoc の便利な機能を使ってみます。artisoc には、変数の過去の値を記憶するという機能が付いています。この機能を使って、「5 ステップ前に燃焼していたら (赤木) 燃え尽きる (黒木)」というルールを書きます。

[15] まずは、変数の値を記憶させます。木エージェントの燃焼状態の変数 (Condition) のプロパティを開きます。そして、その記憶数を 5 にしてください (デフォルトは 0 です)。これで 5 ステップ前の値までを記憶します。

[16] 記憶を呼び起こすには、GetHistory(変数名, 整数) という関数をつかいます。たとえば、GetHistory(HENSU , 1) と書くことで、HENSU という変数の現在の値ではなく、前のステップの最後の時点での値を参照することができます。というわけで、「 5 ステップ前に燃焼していたら (赤木)、燃え尽きる (黒木) 」というルールは、

```
If      GetHistory(My.Condition, 5) == Color_Red      then
                                                .....5 ステップ前が赤だったら、
                My.Condition = Color_Black
End if                                                    となります。
```

[第五工程] 熱が伝播するルールを書きます (1)

[17] 空間を熱が伝播していくルールを書きます。空間は、エージェントと異なって、自分のルールを持つことはできません。なので、空間の変数の操作は、Universeで行うこととなります。空間のステップ (Univ_Step_Begin) に、空間変数の各セルでの値を操作するルールを書きます。

[18] 熱伝播は、各ステップに東西南北の隣接セルから 20% が伝播し、20% がもとのセルに残るというルールにしましょう。

[19] まず (ちょっと都合もあって) 20% ずつが周囲から伝播してくるルール (のみ) を書きましょう。(コピー & ペーストを駆使してね !)

```
dim i as integer
dim j as integer
For i = 1 to GetWidthSpace(Universe.Forest)-2
    For j = 1 to GetHeightSpace(Universe.Forest)-2
        Universe.Forest.Temp(i,j,0) =
            Universe.Forest.Temp(i,j,0)+
            Universe.Forest.Temp(i+1,j,0)*0.2+
            Universe.Forest.Temp(i-1,j,0)*0.2+
            Universe.Forest.Temp(i,j+1,0)*0.2+
            Universe.Forest.Temp(i,j-1,0)*0.2
    Next j
Next i
Next i
```

少し入り組んでいます。よくルールを読解してみてください。

ルールが書けたら実行してみてください。うまく熱は伝播していくでしょうか。

[第六工程] 熱が伝播するルールを書きます (2)

[20] 実は、このままでは熱はうまく伝播しません。熱は右上方向には急速に、左下方向にはゆっくりと伝播していったことと思います。これには artisoc の処理順序が関連しています。artisoc ではルールの実行順序が来ると、即座にその場でルールは処理されます。当たり前のようですが、複数のルールを同時に処理したい場合も、時間が自動的に同期されるわけではないということを意味します。例えば、A という行動ルールと B という行動ルールが、本来なら同時に動かなければならない場合も、行動ルール A が処理されてから行動ルール B が処理されるということになります。

[21] 今回のモデルの場合、空間のセルからセルに熱が伝わるというルールは、本来すべてのセルで同時に処理されなければなりません。しかし、このままでは、先に処理されたセル (左下) には先に周囲のセルから熱が伝播します。そして、後で処理されるセル (右上) は、それらの処理済み (= つまり熱が伝播済み) のセルから熱が伝播するために、早く熱が伝わってしまうというわけです。

[22] < 解決編です > 前口上が長くなってしまいました。解決するには (幾つかの方法がありますが) 今回ならった GetHistory を用います。伝播してくる熱の量を判断するときに、現時点での熱の量ではなく、前ステップの最後の時点での熱量を用いれば良いわけです。

[23] まず、温度をあらわす空間変数に、記憶を設定します。そして、熱伝播のルールに GetHistory を導入します。さらに、熱が伝播してしまっって冷えるルールも加えておきましょう (太字下線部)。

```
For i = 1 to GetWidthSpace(Universe.Forest)-2
  For j = 1 to GetHeightSpace(Universe.Forest)-2
    Universe.Forest.Temp(i,j,0) =
      GetHistory(Universe.Forest.Temp(i,j,0),1)*0.2+
      GetHistory(Universe.Forest.Temp(i+1,j,0),1)*0.2+
      GetHistory(Universe.Forest.Temp(i-1,j,0),1)*0.2+
      GetHistory(Universe.Forest.Temp(i,j+1,0),1)*0.2+
      GetHistory(Universe.Forest.Temp(i,j-1,0),1)*0.2
  Next j
```

Next i

[第七工程] 木の行動ルールを書きましょう (2)

[24] さいごに「燃えるルール」を書いておきましょう。自分のいるセル (格子) の温度が 100 度を超えたら、緑木は燃え上がることにしましょう。ここも自力で考えてみましょう。レジユメのどこかに、私案を書いておきます。分からないひとは探してみてください。

[25] さて、厳しく禁じられている火遊びを、PC のなかで思いっきり楽しんで下さい。本数をいろいろ変えてみてください。そして、飽きてきたら、空間を広げて楽しんで下さい。一度消えかけていた火災が、再開したりもします。

課題

空間の広さを 50×50 にしてください。そして木は視野 20 で広がることにします (41×41 の範囲)。木の本数は 1000 本くらいにしてください。

新しいエージェント「消防士」を加えます。消防士の数もコントロールパネルで設定できるようにしてください。数は 0 から 100 としてください。消防士は、ランダムに配置され、(水をまいて) 自分と周囲の 8 つのセルの温度を、毎ステップ 10 度ずつ下げつづけるようにしてください。

木が燃え上がるルール (案)

```
If      Universe.Forest.Temp(My.X, My.Y, 0) > 100
      and My.Condition == Color_Green                then
      My.Condition = Color_Red
End if
```

コンパネで設定した数だけ木を生み出すルール (案)

```
dim i as integer
For i = 0 to Universe.N_Trees-1
    CreateAgt(Universe.Forest.Tree)
Next i
```