

第八回 複雑な個性を与える

前回の復習と今回の趣旨

前回は、エージェント集合を操作する関数や ForEach 文を利用することで、他者を（少し高度に）認識し、さらに働きかける方法を学びました。

今回は、エージェントに少し複雑な個性を与えます。文字列を使って個性を表現します。また、その複雑な個性に基づいて、少しいろいろ考える行動ルールを書いてみます。

今回は、R.Axelrod の「文化流布 (Dissemination of Culture) モデル簡易版」を作ります。オリジナルは、R.Axelrod (1997) The Complexity of Cooperation (『対立と協調の科学：エージェント・ベース・モデルによる複雑系の解明』(ダイヤモンド社)) に収められています。簡易版といってもモデルの本質的な部分はほとんどそのままです(だと思います)。この本には他にも面白いモデルが紹介されているので、自作モデル作成に向けて読んでみると良いと思います。

文字列型変数

文字列を操作する関数

GetAgt

空間の大きさを調べる関数

If 文の入れ子構造 (再出)

今回作成するモデルは、ある王国に住む住民がさまざまな多様な文化を持っていて、それぞれの住民は近隣の住民の文化を見て、真似していくというものです。その際、もともとより似ているもの同士は真似しがちで、逆にあまり似ていないもの同士は真似しあわないというルールにします。

[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 Realm、設定はデフォルトのままで良いです)。住民エージェントも作ります (名称は Inhabitant、数は 0 でよいです)
- [02] まずは出力設定もしておきましょう。基本的にはデフォルトで問題ありません。
- [03] 空間の大きさだけ、住民を生み出して、王国にしきつめてみましょう。空間の広さだけの住民を生み出す方法ですが、ルールのなかで空間の広さを調べなくてはなりません。そのために、GetWidthSpace(空間名)、GetHeightSpace(空間名) という関数を使います。予想はつくと思いますが、空間の横幅と縦幅を数える関数です。

```
dim i as Integer
```

```
For i = 0 to GetWidthSpace(Universe.Realm)*GetHeightSpace(Universe.Realm)-1
```

.....-1 を忘れないように！

```
CreateAgt(Universe.Realm.Inhabitant)
```

Next i

- [04] つぎに、空間にしきつめる方法ですが、RandomPutAgtset を使って、空間にばらまいてやれば、artisoc が勝手にばらまいてくれます。便利な方法なので覚えましょう。

```
dim Population as Agtset
```

.....dim 文はルール文頭にまとめます

```
MakeAgtset(Population, Universe.Realm.Inhabitant)
```

```
RandomPutAgtsetCell(Population, False)
```

.....False は重ねずに配置する場合です

[第二工程] 複雑な個性を与える

- [05] さて、今回の講義ではいきなりメインテーマに入ります。住民エージェントに複雑な文化を持たせることにします。

- [06] それぞれの住民の文化は、3桁の文字列で表現されるものとします。一桁目の文字はその人の衣類に関する流儀、二桁目の文字はその人の食事に関する流儀、三桁目の文字はその人の住居に関する流儀を表現することとします。便宜的にそれぞれの分野（衣食住）で10種類の流儀があることとし、0から9の数字で表現します。いろいろな人々の文化が、001、905、562といった三桁の文字列で表現されます。

- [07] まず、住民エージェントに文化をあらわす変数（名称 Cululture、文字列型）を追加します。そして、まずは無作為に三桁の文化を与えることにしましょう。

- [08] 文字列の操作には少し注意が必要です。文字列をつなげるときには、&を用います。また文字列型変数に値を直接代入するときには、値を“ ”で囲んでおきます。my.name [文字列型変数] = “yamakage”&”susumu”というかんじに使います。また、今回は数字も文字列として用います。1+1では2になります。”1”&”1”は”11”となります。ちょっと注意してください。

- [09] また、変数の型を変える関数が必要となります。他のいろいろな型の変数を

実数型にするときは、 CDbI(元の変数)

整数型にするときは、 CInt(元の変数)

文字列型にするときは、 CStr(元の変数) というルールを用います。

- [10] 以上のこと（とこれまで習ったこと）をふまえ、住民に無作為に三桁の文化を与える方法を考えてみて下さい。（分からない人は、下を見てね。）

```
My.Culture = Cstr(int(Rnd()*10))&Cstr(int(Rnd()*10))&Cstr(int(Rnd()*10))
```

[第三工程] 複雑な色を表示する

[11] それぞれが持つ様々な文化に合わせた様々な色を表示させてみましょう。

[12] 文字列を操作するための関数を学びます。

Mid(文字列型変数 A, 整数 B, 整数 C) というルールで

A という文字列型変数の B 文字目から C 文字分の文字列を抜き出すという操作ができます。Mid("yamakagesusumu", 1, 8) とすれば、"yamakage" という文字列が抽出されます。

[13] 前回、色を表現する変数を設定して、赤や青など 8 つの色をエージェントに与える方法を学びました。今回は、もっと微妙な色を表現する方法を学びます。

RGB(整数 R, 整数 G, 整数 B) というルールで

整数 R は赤の成分、整数 G は緑の成分、整数 B は青の成分をあらわしていて、光の成分を混ぜる要領で色を作成します。各整数は 0-255 の範囲の値をとります。例えば、RGB(0,0,255) は青、RGB(0,0,0) は黒、RGB(255,255,255) は白となります。

[14] まず、住民エージェントに色をあらわす変数を設定してあげましょう(名称 Color、整数型)。そして、住民エージェントに、自分の文化に合わせて色を決めるルールを書いてやります。(自分で考えてみましょう。) 以下のようにかけば、各自が持つ三桁の文化要素の値に対応した、R , G , B の値を持つ色を備えることとなります。

My.Color = RGB(Cint(Mid(My.Culture,1,1))*25, Cint(Mid(My.Culture,2,1))*25, Cint(Mid(My.Culture,3,1))*25)

[15] では、出力設定で、今作成した色変数に合わせて色を表示するように設定しておきましょう。そして、(保存をちゃんとしてから) 実行してみてください。いろいろな文化をもった住民エージェントが敷き詰められたでしょうか。

[16] 空間が 50 * 50 では大きすぎるので、10 * 10 くらいにしておきましょう。

[第四工程] 真似する隣人を選び出す

[17] では、住民エージェントに各ステップの行動ルールを与えていきます。まず、隣人達のなかから、毎回一人だけ、真似をする対象を選び出します。エージェント集合のなかから一人だけエージェントを選び出す際には、

GetAgt(エージェント集合 , 整数 A) というルールを用います。

このエージェント集合の要素には、0 ~ n-1 (n は集合の大きさ) の番号がふられています。その番号を指定してやることで、エージェント集合のなかから一人のエージェントを取り出すことができます。

[18] 各住民エージェントが、周囲の 8 人の隣人のなかから一人を選び出すルールを書いてみましょう。

```

dim Neighbors as Agtset          .....隣人達          = エージェント集合型
dim Neighbor as Agt              .....隣人            = エージェント型
MakeAllAgtsetAroundOwnCell(Neighbors, 1, False)
                                .....視界一の範囲のエージェント(8人)を隣
                                人達として認識します。

Neighbor = GetAgt(Neighbors, Int(Rnd()*8))
                                .....そのなかの一人を無作為に隣人とします。

```

[第五工程] 選んだ隣人を真似る？真似ない？

[19] 住民は複雑な個性をもち、注目する隣人をすでに選択しています。あとは、真似るルールを与えてやれば、モデルは完成です。もともと、より似ている住民同士は真似しがちで、似ていない住民同士はあまり真似し合わないルールにします。

[20] 必要な文法事項はすでに全部学んでいます。学期末の自作モデル製作にむけて、少し自分で考えてみましょう。以下に住民エージェントの思考の流れを書いておきます。私なりのルールは、文書末にあります。うまくいかない(または無理無理)という場合は、参考にしてみてください。

- (1) 今回、隣人の文化について、関心のある分野を衣食住のなかから一つ選びます。
- (2) 今回、自分の文化について、変えてみようかなと思っている分野を同様に選んでおきます。
- (3) 関心をもった分野について、相手の文化要素と自分の文化要素を比べます。
- (4) もしそれが同じだったら(類似した文化をもった同類だと思って安心して)、真似をすることにします。
- (5) 変えてみようと思っていた分野を、その隣人と同じにします。
- (6) 自分の文化に合わせて、自分の色を変えるルールを書いておきましょう。

実は、(1) と (2) で選んだものが同じの場合は変化が起こりませんが、今回は気にしないことにしましょう。

(4) のルールがあるので、似ているものはより似やすい傾向があり、もともと似ていないものはなかなか似ていかない傾向があります。

[第六工程] 試行してみましょう

[21] もともと 1000 種類の文化がありうる世界ですが、相互作用を経て、幾つかの文化に集約していく様子が分かります。しかし必ずしも、一種類の文化にまとまるわけではありません。お互いに真似しあうわけですが、最終的に分かり合えない関係も残るわけですね。

- [22] 今回のモデルは空間の広さを変えても、自動的にそれに合わせて王国を作ってくれるルールを書きました。なので、空間のプロパティを使って自由に空間の広さを変えてみましょう。5 * 5 の王国や 3 * 3 の王国ではどんなことが起こるでしょう。15 * 15 の王国ではどうでしょう（わりと時間がかかります）。
- [23] 大きな王国のほうが、少ない種類の文化に集約されやすい気がしませんか。

[第七工程] 文化の種類を数える

- [24] 文化の種類を数えてみることにしましょう。少しテクニカルな部分があるので、まずはルールを掲載します。ルールの読解にトライしてみましょう。やろうとしていることが分かったら、ルールを書いてみて下さい。
- [25] Universe の下に文化のレポトリイ（名称 Var_Culture、文字列型）と文化の種類（名称 Num_Culture、整数型）をあらわす変数を作っておきます。

Univ_Step_Begin のルールに

```
Universe.Var_Culture = ""          .....文化のレポトリイを空にします
```

Agt_Step のルールに

```
If Instr(1,Universe.Var_Culture, My.Culture) == 0 then
```

```
    .....もし、レポトリイの中に自分の文化が無ければ
```

```
        Universe.Var_Culture = Universe.Var_Culture & "_" & My.Culture
```

```
    .....自分の文化をレポトリイに加えます
```

```
End if
```

Univ_Step_End のルールに

```
Universe.Num_Culture = Len(Universe.Var_Culture)/4
```

```
    .....レポトリイの長さを計算して4で割ります。
```

```
    .....それが文化の種類です。
```

- [26] (ヒント) 各住民は、文化のレポトリイを見て、自分の文化がその中にあるかどうかをチェックします。もし、ないときは、自分の文化をレポトリイに加えていきます。全員が加え終わると。。。_456_678_643 ってなかんじ。
- [27] 「値画面出力」が「時系列グラフ」で、文化の数を出力してみましょう。

課題

簡易版ではなく、Axelrod のモデルのオリジナルのモデルに近いものにしてみましょう。

変更点 1 (文化)

文化の分野は3つではなく、5つにしてみましょう。それに合わせて色の表示の仕方も変えておきましょう。(みなさんの自由にして良いです)

変更点2 (行動ルール)

住民エージェントは、お互いの文化的な類似性と等しい確率で、真似をします。真似をするときは、選択した隣人と自分とで異なる分野を一つ選び、その分野で真似をすることとします。たとえば、82330 という文化をもつ住民は、67730 という隣人と40%の類似性を持っています。なので、40%の確率で真似をすることにします。そのとき、一桁目か二桁目か三桁目のうちの一つを選択して、相手の真似をすることになります。

住民エージェントの行動ルール (サンプル)

```
dim P as Integer
```

```
dim Q as Double
```

```
P = Int(Rnd()*3+1)
```

```
Q = Int(Rnd()*3)
```

```
If Mid(My.Culture, P, 1) == Mid(Neighbor.Culture, P, 1) then
```

```
    If Q == 0 then
```

```
        My.Culture =
```

```
        Mid(Neighbor.Culture,1,1)&Mid(My.Culture,2,1)&Mid(My.Culture,3,1)
```

```
    ElseIf Q == 1 then
```

```
        My.Culture =
```

```
        Mid(My.Culture,1,1)&Mid(Neighbor.Culture,2,1)&Mid(My.Culture,3,1)
```

```
    Else
```

```
        My.Culture =
```

```
        Mid(My.Culture,1,1)&Mid(My.Culture,2,1)&Mid(Neighbor.Culture,3,1)
```

```
    End if
```

```
End if
```

```
My.Color = RGB(Cint(Mid(My.Culture,1,1))*25, Cint(Mid(My.Culture,2,1))*25,  
Cint(Mid(My.Culture,3,1))*25)
```