

世界モデル実習(Sr)/人間の安全保障実験実習 III(Gr) 2006 年度夏学期

第三回 エージェントに判断させる

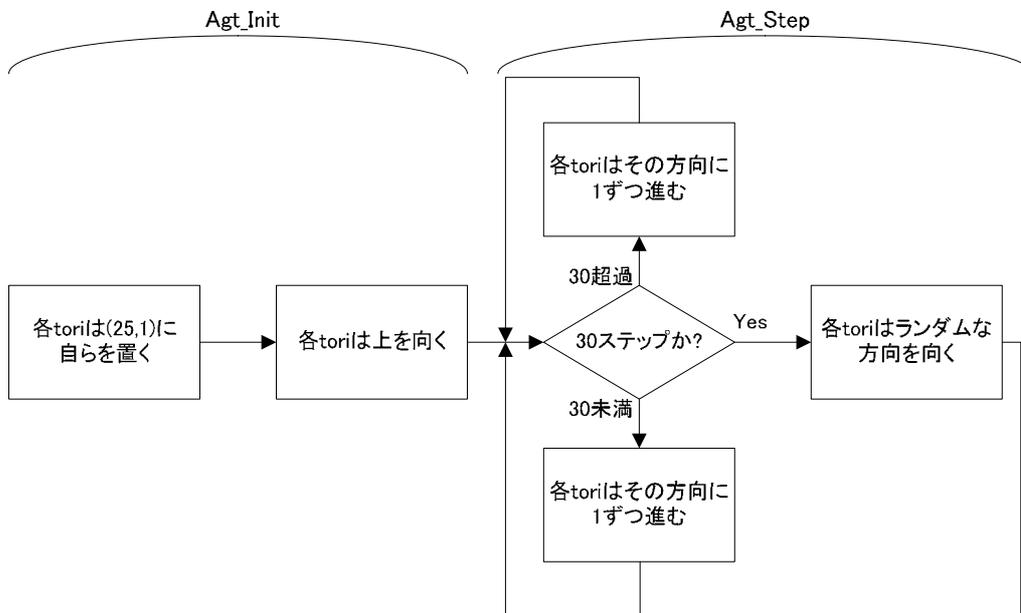
場合分けを設定して判断させる

* 場合分けを図示してみる

エージェントに「全ての tori が oozora の下端中央(25,1)にいる。そこから一斉に真上に飛び立つ。そして、毎ステップ、「1」ずつ、真上に飛び続ける。30 ステップ目に、各 tori は自分の好きな方角を向き、その後は毎ステップ、「1」ずつ、その方向に飛び続ける。」という行動をさせたいとしましょう。このとき、ルールとしては、

- (1) tori が oozora の下端中央(25,1)にいる。
- (2) 真上を向く
- (3) 毎ステップ、「1」ずつ、真上に飛び続ける。
- (4) 30 ステップになっていなければ、そのまま飛び続ける。
- (5) 30 ステップ目なら、飛んでいく方向を適当に決める。
- (6) 30 ステップを超えたら、毎ステップ、「1」ずつ、その方向に飛び続ける。

と書くこととなります。図にすると、こんな感じです。



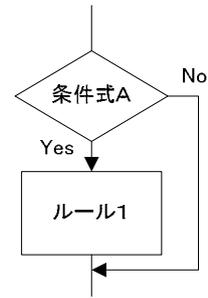
* 場合分けの書き方

場合分けは、if文と言われるものを使います。もっとも基本的な形は

```

if [ 条件式A ] then
    [ 実行したいルール<1> ]
end if

```



のように書きます。たとえば、「自分の Direction が 90 未満だったら 90 にする」としたいときには、

```

if my.Direction < 90 then
    my.Direction = 90
end if

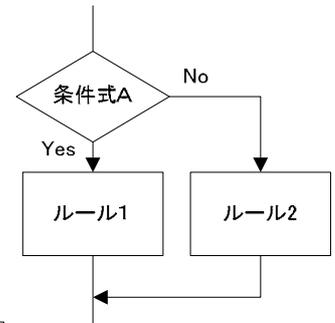
```

のように書きます。もう少し複雑な条件を設定したい時に便利な書き方もあります。

```

if [ 条件式A ] then
    [ 実行したいルール<1> ]
else
    [ 実行したいルール<2> ]
end if

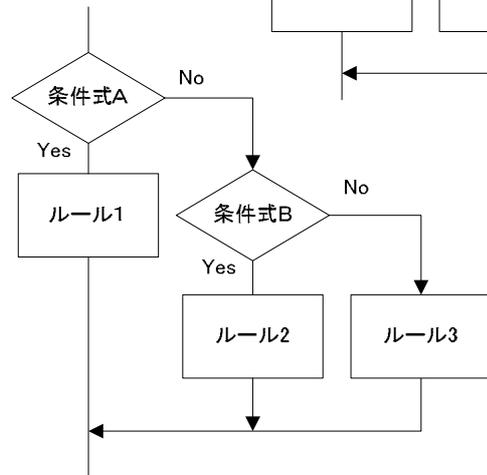
```



```

if [ 条件式A ] then
    [ 実行したいルール<1> ]
elseif [ 条件式B ] then
    [ 実行したいルール<2> ]
else
    [ 実行したいルール<3> ]
end if

```



elseif はいくつ続けてもかまいません。また、else なしで、elseif だけでもかまいません。if 文の中にもう if 文を入れ子にすることも出来ます。

* 条件式

条件式は、等式や不等式で表します。以下の記号が使えます。

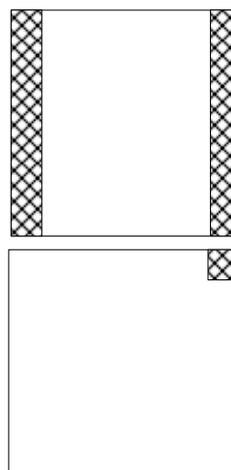
| | | | |
|---|-------|-----|---------------|
| = | 等しい | < > | 等しくない |
| < | より小さい | < = | より小さいか等しい(以下) |
| > | より大きい | > = | より大きいか等しい(以上) |

たとえば、「my.X == 20」や「my.Direction >= 90」のように使います。ちなみに、一つの

if文で複数の条件を設定することも出来ます。たとえば、

```
if my.X < 5 or 45 < my.X then
    my.X = 25
end if
```

```
if my.X > 45 and my.Y > 45 then
    my.X = 0
    my.Y = 0
end if
```



例題 1

トリモデルを開けて、別名(たとえば hanabi)で保存し、冒頭のモデルを作ってみましょう。

```
Agt_Init{
  my.X = 25
  my.Y = 1
  my.Direction = 90
}
Agt_Step{
  If 30 == getcountstep() then
    my.Direction = rnd()*360
  elseif 30 > getcountstep() then
    forward(1)
  elseif 30 < getcountstep() then
    forward(1)
  End if
}
```

冒頭のフローチャートと同じように作ると、こうなります。getcountstep()はステップ数を取得する関数です。あとは分かりますね。もう気づいていると思いますが、Agt_Stepの部分では下のように書き換えることも出来ます。

```
If 30 == getcountstep() then
  my.Direction = rnd()*360
else
  forward(1)
End if
```

例題 2

ステップ目では 50%の確率で左右に分かれ、さらに 40 ステップ目以降では、毎ステップ左右 10 度の範囲でランダムに方向を変える。なお、今までは、向きを変える 30 ステップ目では前方に飛びませんでした。ルールを簡単にするために、30 ステップ目も 40 ステップ目も前方に飛ぶようにします。

```
    If 30 == getcountstep() then
        If 0.5 > rnd() then
            my.Direction = 180
        Else
            my.Direction = 0
        End if
    End if
    If 40 <= getcountstep() then
        turn(rnd()*20 - 10)
    End if
    forward(1)
```

変数の追加

これまで、X,Y,Direction といった変数を使ってきました。変数は自分で追加することも出来ます。hanabi モデルに、進んだ回数を表す変数「kaisuu」を追加してみましょう。ツリで tori を選択して、変数の追加をクリックします。整数型を選択しましょう。

```
    If 30 == my.kaisuu then
        Turn(rnd()*360)
    else
        forward(1)
    End if
    my.kaisuu = my.kaisuu + 1
```

このように、自分で何ステップ進んだか数えることも出来ます。Turn()は方向を変える命令です。Turn(20)とすると、反時計回りに 20 度角度を変えます。上の例だとランダムな方向を向くのは分かりますね。変数の型には以下のようなものがあります。

| 型の種類 | 正式名称 | 中に入るもの | 値 |
|-----------|---------|-----------|-----------------------|
| ブール型 | Boolean | 真偽 | True, False |
| 整数型 | Integer | 整数 | -4, 235234, 0, 690 |
| 実数型 | Double | 実数 | 3.1415926, 3.33333 |
| 長整数型 | Long | 桁の大きい整数 | ± 20 億を超えるとき |
| 文字列型 | String | 文字列 | “Yamakage” “山影” |
| エージェント型 | Agt | 個別エージェント | tori(id0), tori(id1), |
| エージェント集合型 | AgtSet | エージェントの集合 | {tori1, tori2} |
| エージェント種別型 | AgtType | エージェントの種類 | tori, hito |
| 空間型 | Space | 空間の名前 | oozora |

課題

hanabi モデルを改造して、飛び散ったホシが徐々に下(270)に進行方向を変えてユラユラ落ちていくモデルを作ってみましょう。

ヒント： 右を向いている時は 90 未満か 270 超過

新しいモデルにエージェントを一つ作りましょう。(10,10)からスタートして、一辺が 30 の正方形の軌跡を描かせてください。三角形も作ってみてください。50%の確率で右回り、50%の確率で左回りに進むようにしてみましょう。

今日のポイント

- ・ if 文で場合分け
 - if ... then ... end if
 - else、elseif を使ってより細かくできる。
 - if 文は入れ子にできる。
 - rnd() を使って、確率で行動を決めることができる。
- ・ 条件式
 - 等号、不等号の書き方。「等しい」は「= = 」、 「 = 」は代入という点に注意。
 - and や or を使って、二つ以上の条件を付けられる。
- ・ 変数をツリーに追加して使える。