**+Workshop on World Modeling・Workshop on Methods of Human Security Studies**
**2005 Summer Semester**

Presiding Professor: Susumu Yamakage
TA; Takuto Sakamoto, Kazutoshi Suzuki,
Hiroyuki Hoshiro, Katsuma Mitsutsuji
Kazuya Yamamoto

# Lecture Seven: God is to Observe! (June 7th)

**●Outline**

○Previous Assignment and Review
○Agent Set Function（Make AgtSet series of functions）
○MoveTo Space: Rules for Moving
○Output of Time Series Graphs
○How to Use Debug
○Assignment

●Previous Assignment and Review

Six students presented their works.

●What We Learned Last Time.

☆Cell type space（Don't forget to modify the scope of space!）
☆Universe can store rules and variables.
☆How to create a control panel; only apply to variables set in Universe.
☆For sentence
☆CreateAgt （Type of agent to be created）
☆RandomPutAgtSetCell （AgtSetg variable, Is duplication possible?（True/False））
☆MakeAgtSet （AgtSet Variable to be produced, Agt Type）
☆MergeAgtSet （AgtSet Variable 1, AgtSet Variable 2）

●Today’s Target

Last time we created the segregation model, (T.Schelling) and we learned how to write rules in the Universe. This time we’ll complete the model and learn about another role of the Universe（How to observe the world!）

●This Week’s Grammar

☆Summary: Functions of「Make=AgtSet series」

We have learned about several functions that handle agent sets（agent set function）Especially noted is the function that creates the first agent set （just call it「Make=AgtSet series」）This is important in writing rules to make agents aware of their surroundings. There are numerous kinds, so let me teach you how they can be distinguished.（You don’t need to memorize them. We’ll proceed with the use of Help）

＞**MakeAgtSet**

As for this series, this is a bit of a special function. One kind of agent will be stored in a single agent set function.

＞**Make** **A** [One, All,None] **AgtSet** **B** [AroundOwn, AroundPosition] **C** [Cell, None]

Ex. **MakeOneAgtSetAroundOwnCell**（AgtSet Variable, Scope, Agt type, Include self？）

The name seems complicated but it is easy to comprehend when certain understandings are stipulated.

Ａ：We will designate what types of agents we are making in an agent set .

｛With just one type of agent, with all types of agents, with designated types of agents only｝

Ｂ：We will designate with which point of view we will create the agent set.

｛from oneself or from designated axis｝

Ｃ：We will designate with what kind of space awareness we will make the agent set.

｛cell type or real number space｝

Since the required arguments differ, please use them with the use of Help.

★Based on a previously-created model, let’s write the rules to have the turtles be aware of their surroundings. We'll write down the rules to count the surrounding red and blue turtles and their total.

○Let's make use of the Comment statement.
○Don't forget to write rules for both red and blue turtles.
○Those who have created variables in the tree must confirm that proper awareness is there through information display using output setting.

☆**MoveToSpaceOwnCell**

Up until now we have been using functions such as Turn and Forward to move the agents. They've moved after we had indicated the direction of movement and the distance. But in case of cell types, a less cumbersome（more sophisticated?）moving method is available. The rule states;「If there is open space, move to that space」

```
MoveToSpaceOwnCell( range )
```

**Ex.** MoveToSpace(2)

★Let's write the rule in the turtle's rule editor to let the turtle move if he is unsatisfied.

We let each turtle calculate his segregation rate. After this, we set the standard segregation rate (level of satisfaction, threshold) that applies to turtles in general and decide if this turtle is unsatisfied or not. If this level of satisfaction is set as a variable inside the Universe, it can be operated with the control panel. This is a much more simple method.＜Nevertheless, be aware that you cannot divide by zero. The denominator of a turtle with no one around him will be zero. Establish with if sentence so that if the surrounding turtle is zero turtle then, automatically, the segregation rate is zero.

☆Observe the System with the Universe.

By using the rules of the Universe, the system can be observed. Let's observe the number of unsatisfied turtles.（In the assignment we will ask you to calculate the segregation rate）The appropriate way is to write the rule of observing the system inside Step_End{} of the Universe. First of all, prepare a variable to aggregate. This time we will learn the method where all processing is done in the Universe. In this method all agents are asked to cooperate.

（１）Rather tacky but a simple method.

```
Univ_Step_End{
dim i as integer
//Initialize the value of the variable
Universe.Num_Unsatisfied = 0
// Count the number of unsatisfied turtles
        // Add up the unsatisfied red turtles
        For i = 0 to Universe.Num_Reds-1
                If Universe.Streets.RedTurtle.Seg_Rate(i)< Universe.Seg_Criteria then
                        Universe.Num_Unsatisfied = Universe.Num_Unsatisfied + 1
                end if
        Next i
        // Add up the unsatisfied blue turtles
        For i = 0 to Universe.Num_Blues-1
                If Universe.Streets.BlueTurtle.Seg_Rate(i)< Universe.Seg_Criteria then
                        Universe.Num_Unsatisfied = Universe.Num_Unsatisfied + 1
                end if
        Next i
}
```

（２）Fashionable smooth method

```
Univ_Step_Begin{
//Initialize the value of the variable
Universe.Num_Unsatisfied = 0
}

 （to the rules of the red and blue turtles）
Agt_Step{
// Add unsatisfied red turtles （blue turtles）
If My.Seg_Rate < Universe.Seg_Criteria then
        Universe.Num_Unsatisfied = Universe.Num_Unsatisfied + 1
end if
}
```

☆The use of Debug（A bit sophisticated）

KK-MAS has a tool for checking whether the created models accurately operate or not: Debug. With the use of Debug, we can execute the rule, line by line and stop it whenever we wish and check what values the variable is showing at that time.

○Try Step in（Step over）
○Breakpoint
○Check the values of the variable

●Assignment（Renovate the segregation model）

（Renovation 1 ☆） Calculation of average segregation rate.

In this lecture we have used the rules of the Universe and counted the number of unsatisfied turtles. Try next to compute and output the average segregation rate.

（Renovation 2）Individualization of satisfaction level.

In our segregation models so far, the value showing at what segregation rate one was satisfied was uniformly set in the Universe. Let's try to separately set the level of satisfaction for red and blue turtles.

（Renovation 3）Change the turtle rules

According to the segregation model, the rule was that unless there are a certain proportion of turtles in the surroundings, one had to move. What if the rule is changed; the turtle moves unwillingly when there are a certain number of different types of turtle surrounding one. How will the system react to this?