Transition States

Our program has many states namely

1. Rest
2. Walking
3. Action Dance
4. Flying

Before we begin describing to you how transition from one state to another state works in our program, let's first understand the mathematical aspect behind the change of states. So, first we will understand what a stochastic process is. A stochastic process is a family of random variables defined on a given probability space and indexed by a given parameter which varies over an index set. So our project is also a stochastic process i.e Transition from one state to another is a random function. We are using rand() function to generate a number between 0 and 1 which we are using as a probability. So transition from one state to another depends on the probability generated by rand() function.

Now consider a discrete parameter and discrete state space stochastic process that takes finite or countably infinite possible values. If future state depends on immediate present and not on the past, then the stochastic process is called markov process. In our project we have discrete state space (1. Walking, 2. Sitting, 3. Laying down, 4. Dancing and 5. Running) and discrete parameter space ( number of iterations of the function). Also future transition state depends on immediate present state and not on the past states(Also called as markov process). So our model is also a markov chain. The probability of transition is set constant so the model is homogenous markov chain.

So as we described the mathematical aspects, let's now understand the states and how transition takes place. Our program has several states as mentioned above. Also every state is divided into many substates depending upon the type of instruction required by the program. Each sub-state has three parts:- starting, mid part and termination.

# 1. REST

REST state is divided into 4 substates depending upon the type of action required to be performed by the program. The 4 substates are:

a. RESTV1 : In this substate, the virtual character stands still and move its tail gently.

b. RESTV2 : In this substate, the virtual character stands still without any motion whatsoever.

c. RESTV3S-RESTV3L1-RESTV3L2-RESTV3E : In this substate the character first sit down to its position and then as the loop goes the character stands up. Then lifts its hands a little in the loop part and put the hand back to its initial position after gently waving it.

d. RESTV4S-RESTV4L-RESTV4E : In this substate the lie down along its belly and roll over.



# 2. WALK

WALK state is divided into 2 substates depending upon the type of action required to be performed by the program. The 2 substates are:

a. WALKV1S-WALKV1L-WALK1E : In this substate, the character walks at a normal speed.

b. WALKV2S-WALKV2L-WALK2E: In this substate, the character runs at a significant speed.



# 3. Fly

FLY state is also divided into 2 substates depending upon the type of action required to be performed by the program. The 2 substates are:

a. FLYS1-FLYL1-FLYE1

b. FLYS2-FLYL2-FLYE2

In both of the substates, action of flying is depicted with slight modifications and describe how the character takes off from the ground and jumps in the air. Then the character flaps its wings and takes off to fly. Once the character does one loop of flying, another trigger of FLYE brings the character back to ground and landing sequence is triggered.



# 4. Action Dance

In this state, we don't have any substates. Action dance has three subparts:- start, mid and end of loop. Whenever program reaches to dancing state, it performs a constant set of instructions unless interrupted. Once finished, probability is calculated transition is made from dancing state to some other state.



Lets now take a look at the transition probability matrix of the program. As we know the probability of making a transition from one state is another is independent of previous state with constant probability. So let’s take a look at the flowchart and the TPM of the program.

