# Complex Game Systems – Design

Fuzzy Logic State Machine Unity Proposal

By Jacob Cooper

# System

The fuzzy state machine is an extension to standard finite-state machines where it uses fuzzy logic to determine behaviours. Instead of processing crisp values, both the inputs and outputs are fuzzy values. This is advantageous for AI as it allows for a more intelligent feeling choice selection and this is because it allows the ability to blend different variables to choose a state.

The goal is to implement a modular Fuzzy State Machine into unity where you can design and have as many states as you like, and the system will process these states into actuality inside of unity which can be used by the AI.

* This is advantageous for AI as it allows for a more intelligent feeling choice selection and this is because it allows the ability to blend different variables to choose a state.  
  The intelligence comes from the felling of the ai making choices based on the graphical formulars used to determine the desired desire level,
* Look into the likelihood/confidence formular for Fuzzy Logic
* The goal is to provide a user the ability to use and create states

# Mathematical Operations

The mathematical operations involved in this system include:

* Converting a crisp set to a fuzzy set.
* Calculating a variables current desirability.
* Using the desirability in logic to output the correct state.

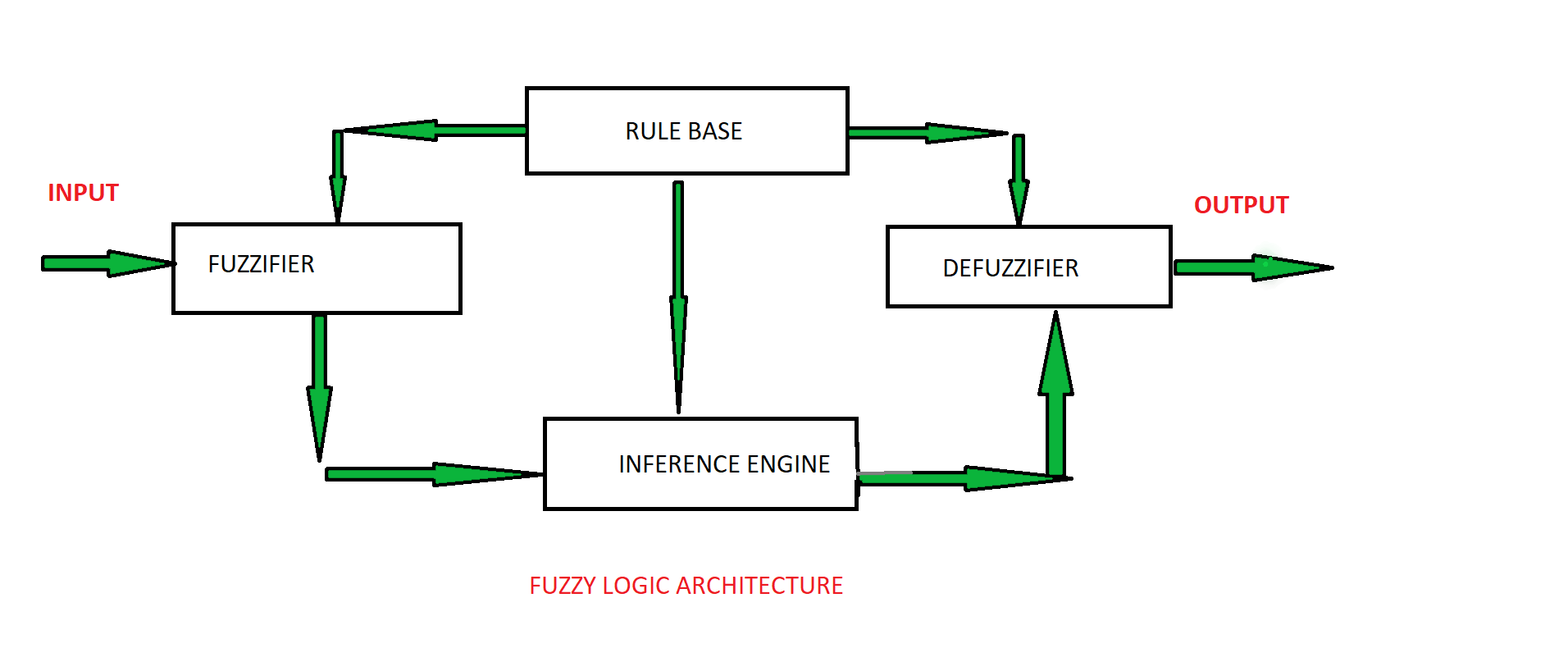
Logical Operators:

* (x and y) = min(x, y)
* (x or y) = max(x, y)
* Not x = 1 – x

# Advanced Algorithms

A fuzzy state machine algorithm will have to be included which calculates fuzziness and states all under a set of rules and then outputs the correct desirable state to the AI to execute.

* Provided an in-depth explanation on how crisp data becomes fuzzy and how to undo the fuzzy data to make choices



# Modularity

To make my system modular, I will be creating a Fuzzy State Machine object which can be edited using a node graph editor which will then be implemented onto objects. The states will have functional events connected using script variables. This allows my fuzzy state machine to be fully customizable and can be used anywhere for any AI inside of unity.

This system will be exported into a unity package and can be loaded in any unity application by simply importing it.

* Talk about how the system will work.
* Explain “full customisable”
* Explain how the graph window is created

# Integration

Integration of my system will be simple, you import the unity package, create the Fuzzy State Machine Object, open it in the Fuzzy State Machine Editor, create your states and assign variables and actions to them, then assign it to the AI object you wish it to run with and then you are done.

Also included is the ability to create your own State Machine nodes with code.

Instructions included in a readme and a demo inside of the project to help the user.

* create the Fuzzy State Machine Object, How?
* open it in the Fuzzy State Machine Editor, How?
* create your states, How? Also can you reuse states?
* assign variables and actions to them, How?
* then assign it to the AI object you wish it to run with. Much more detail required, mostly how this will work
* Also included is the ability to create your own State Machine nodes with code. , How?