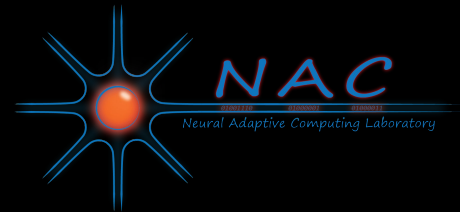


NEAT Implementation

William Gebhardt



Population

—

Population

- A collection of Genomes
- Broken up by species
- Controls which members of the population will produce offspring
- Runs the current generation through the task environment
- Agnostic to how Genomes are implemented and the topologies of each member

Speciation

- Used to protect suboptimal species from extinction
- Uses a defined compatibility distance and threshold to divide up genomes

$$\delta = \frac{c_1 E}{N} + \frac{c_2 D}{N} + c_3 \bar{W}$$

- Each species has a representative from the previous generation
 - Chosen randomly from amongst the genomes
- Genomes are assigned to the first species their compatibility distance falls under a certain threshold

Reproduction

- Each species is allotted a certain number of offspring
- Best members of the previous species are copied forward (Single Parent)
- Truncate the every member below a certain fitness
- Choose parents randomly from remaining members (Two Parents)

Single Parent

- The offspring will be an exact replica of the parent

Two Parents

- The offspring will inherit the parent with the high fitness's traits
- Shared genes will be merged

Extinction

- Just using the threshold to filter out bad species results in stagnation
- After every generation the average fitness of each species is tracked
- If a given species hasn't improved in the past n generations it will be culled from the population
- Removes species that have gotten stuck in a corner.

Start		
		Goal
		Stuck Species

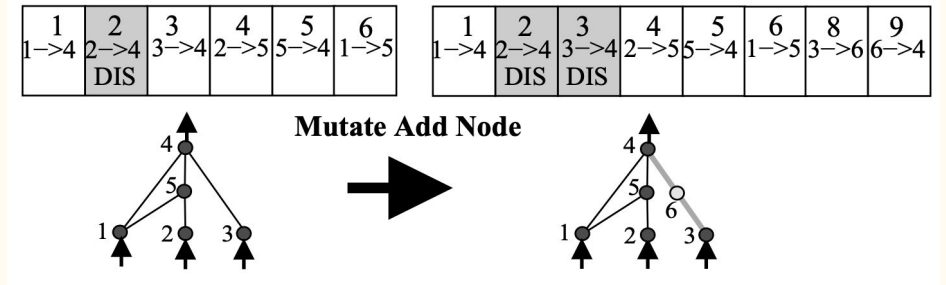
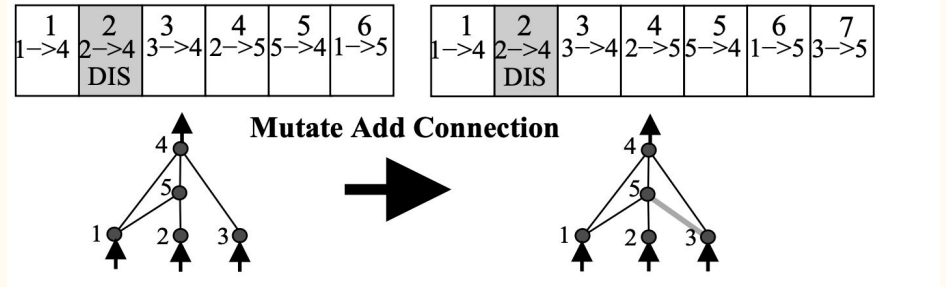
Genome

Initialization

- Each Genome has a unique id
- Node genes are created for each input and output
- Possible connection systems
 - Fully connected
 - Single input bound to each output
 - Random input bound to each output

Mutation

- Genome level mutations are
 - Connection manipulation
 - Node manipulation
- Mutations occur during reproduction



Genes



Node Genes

- A few additions beyond the initial values
- Activation function
 - tanh, sigmoid, summation
- Bias
- Unique Id
- Mutates by the bias by either a small amount or reinitializing it to a random value

Connection Genes

- Weight
- Active
- Innovation Number
- Mutation
 - Shifts the weight a small amount or reinitializing it to a random value
 - Small chance to flip the active status of the gene

Phenotype

Connection Matrix

- Have connections stored in a matrix and do a number of matrix multiplies to propagate all data to the output neurons
- Pros
 - Straight forward to construct
 - Parallelizable
- Cons
 - Scales exponentially
 - Lots of pointless math operations

Feedforward Graph

- Compute the order in which nodes of the graph must be calculated
- Pros
 - Less individual math operations
 - Allows for custom behavior at node level
- Cons
 - Threading is the only way to parallelize it
 - Does not scale to massive numbers of nodes due to its sequential nature

Creating a feedforward graph

- Start at the output of the graph
- Compute all nodes needed to compute those nodes
- Recursively work backwards from output to input to compute dependencies
- Output looks like a set of nodes to be computed in order that is guaranteed to be missing data


```
pip install neat-python
```

—