Introduction to
Evolutionary Programming
And Genetic Algorithms

After scientists became disillusioned with classical and neo-classical attempts at modelling intelligence, they looked in other directions.

Two prominent fields arose:

- Connectionism (neural networking, parallel processing)
- Evolutionary computing (Genetic Algorithms, Genetic Programming, etc…)

Artificial Intelligence takes example from natural solutions:

- Artificial Neural Networks The Brain
- Fuzzy Logic Reasoning and Experience
- Evolutionary Programming (inc. Genetic Algorithms) Natural Selection and Evolution

The Principles of Natural Selection and Evolution

Selection:

- If there is a pool of various individuals, those who are **fit** enough to copy themselves survive, if not, they extinguish.

- Reproducing by copy means that the **fittest** individuals populate the environment while the unfit eventually go extinct.

- But this only works if we have **variety** to start with.

- Natural Selection happens by letting the individuals perform (i.e. “live”) in an environment where they have to solve a problem (“survive” for long enough to be able to reproduce)

The Principles of Natural Selection and Evolution

Evolution:

Can you evolve by copying? Can you adapt this way?

If an organism **copies** itself to reproduce how can it evolve?

> **Mutation**.
>
> It is a renewable source of **variety**.
>
> But it is dangerous and absolutely random therefore an *effective* but not very *efficient* way to evolve.

The Principles of Natural Selection and Evolution

Evolution:

If different and already tested good treats could be shared it would be easier!

Cross Over Share information between already tested individuals

(→ Sexual Reproduction)

It is much safer and not so random, therefore more *efficient* than mutation.

But is it more *effective*?

It does not provide **renewable variety**.

(Once all combinations have been produced, there will be no more variety).

So we still **NEED Mutation** to maintain **variety**.

The Principles of Natural Selection and Evolution Applied to Problem Solving

Problem Solving:

From observing Natural Selection and Evolution we can see that:

- Neither Selection nor Evolution is a **solution** to a problem.

- They provide a way to **search** for a solution by evolving it.

- Therefore Evolutionary Programming can be seen as a methodology for searching solutions rather than a solution in itself.

- The solution is searched by trying it in the actual problem rather than trying to find the inverse model of the problem!

  It is a **Direct** solving method rather than an **Inverse** one.

Evolving Solutions to Problems

The Basic Genetic Algorithm

Symbolic AI vs Genetic Algorithms

- Most symbolic AI systems are very static, they can usually only solve one given specific problem
- If the problem were somehow to change, these systems could have a hard time adapting to them
- Genetic algorithms can combat these problems

- They are basically algorithms based on natural biological evolution
- The architecture of systems that implement genetic algorithms (or GA) are more able to adapt to a wide range of problems
- Genetic algorithms can be incredibly efficient if programmed correctly.

General Algorithm for Genetic Algorithms

Genetic algorithms are not too hard to program or understand, since they are biological based.

Thinking in terms of real-life evolution helps.

The general algorithm for a GA is:

- Generate a large set of possible solutions to a given problem (**initial population**)
- Evaluate each of those solutions, and decide on a "**fitness level**" ("survival of the fittest")
- From these solutions breed new solutions (the next **generation**)
    - The parent solutions that were more "fit" are more likely to reproduce
    - While those that were less "fit" are more unlikely to do so
- Solutions are **evolved** over time, by repeating the process each **generation**.
- **Terminate** when a solution has been found or other **termination criteria** has been met

General Algorithm for Genetic Algorithms