Learning Behaviours

COMP3431 Robot Software Architectures

Where are we?

- We are looking at robot architectures starting with simple mechanisms and progressing to more complex ones

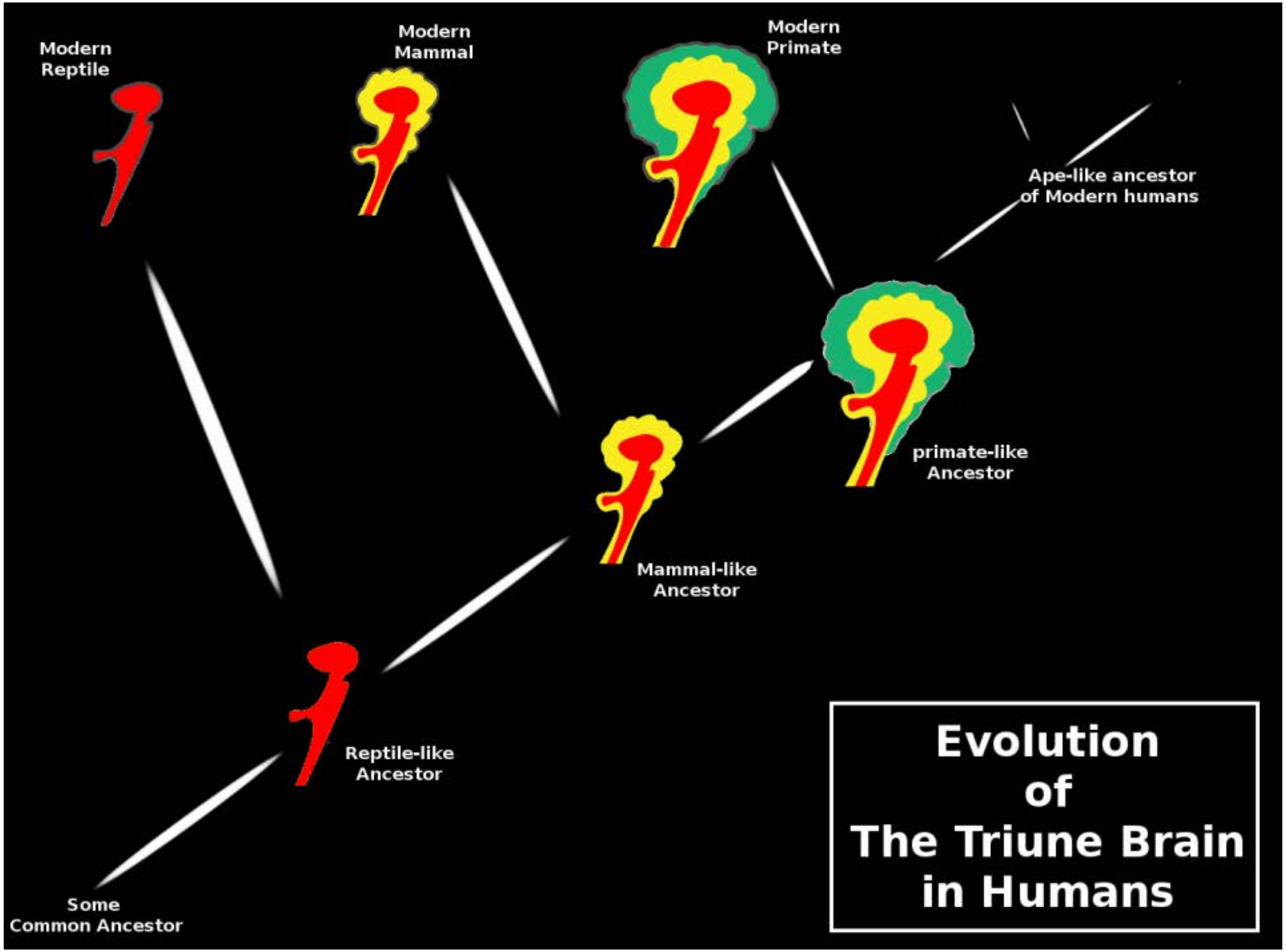
- Last week we saw how basic behaviour-based robots can be programmed by situation-action rules

- This week we look at how such behaviours can be learned

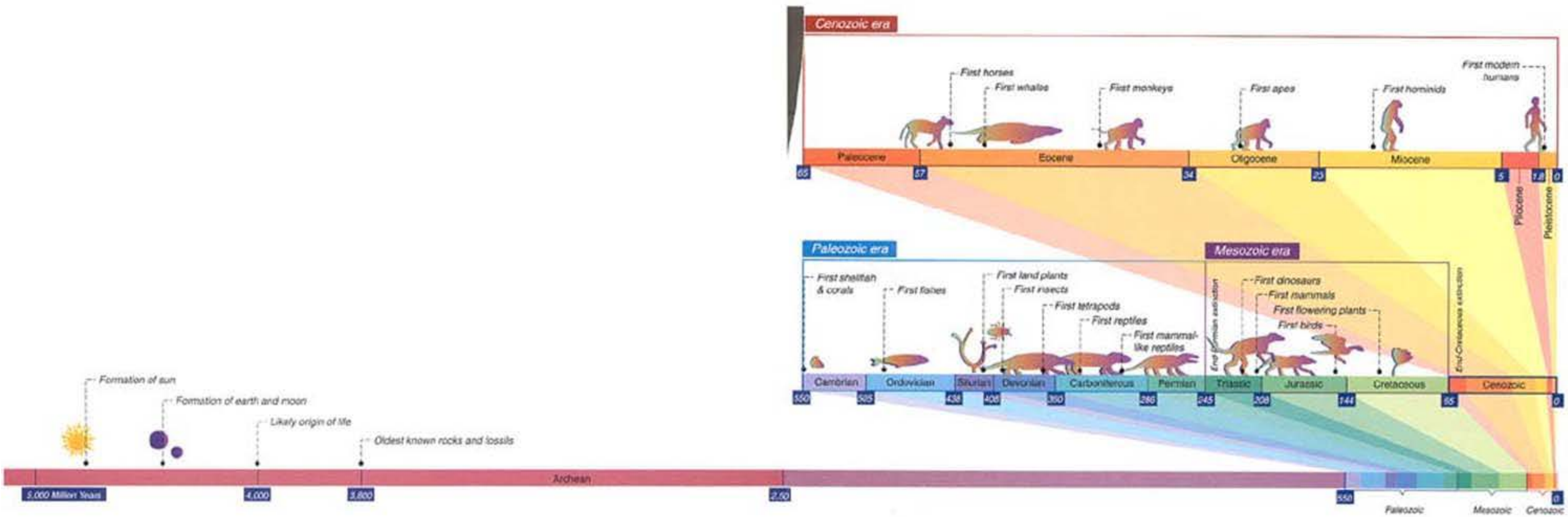
AI Focus

- Originally focussed on human-level intelligence
- Forgot that it is built on low-level intelligence

Evolution of the Brain

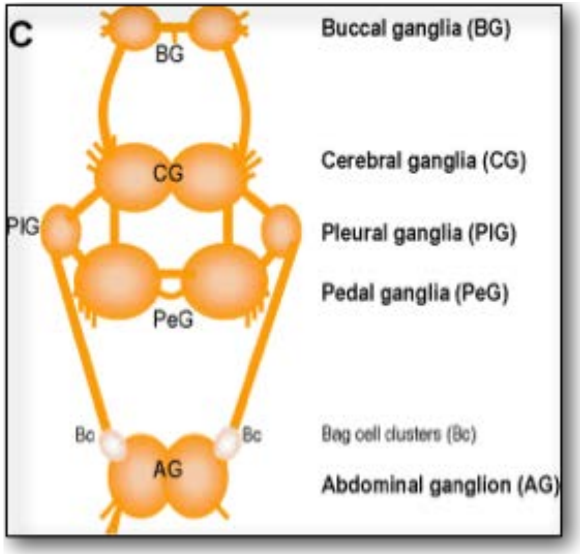


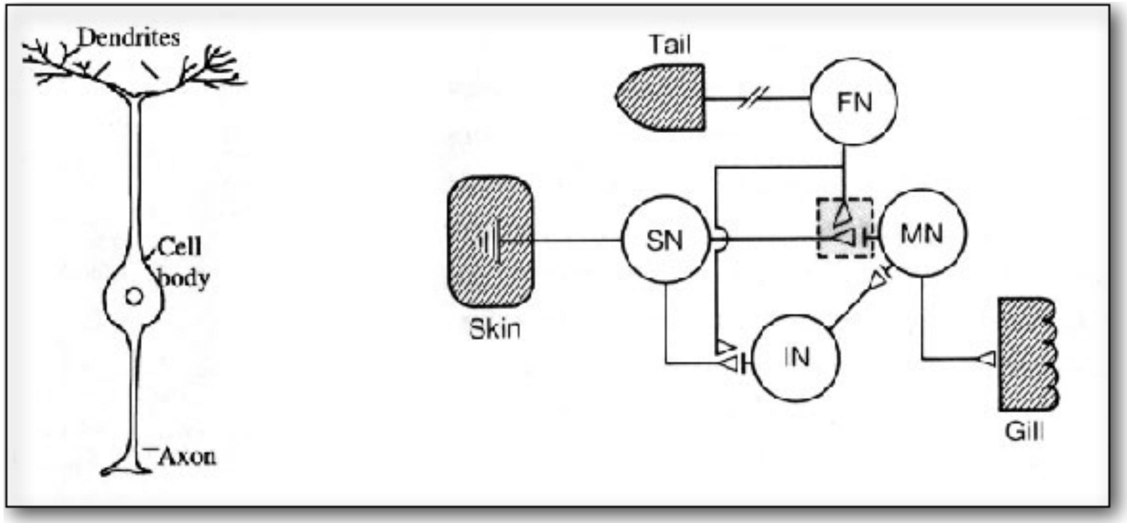
Evolution of Life on Earth



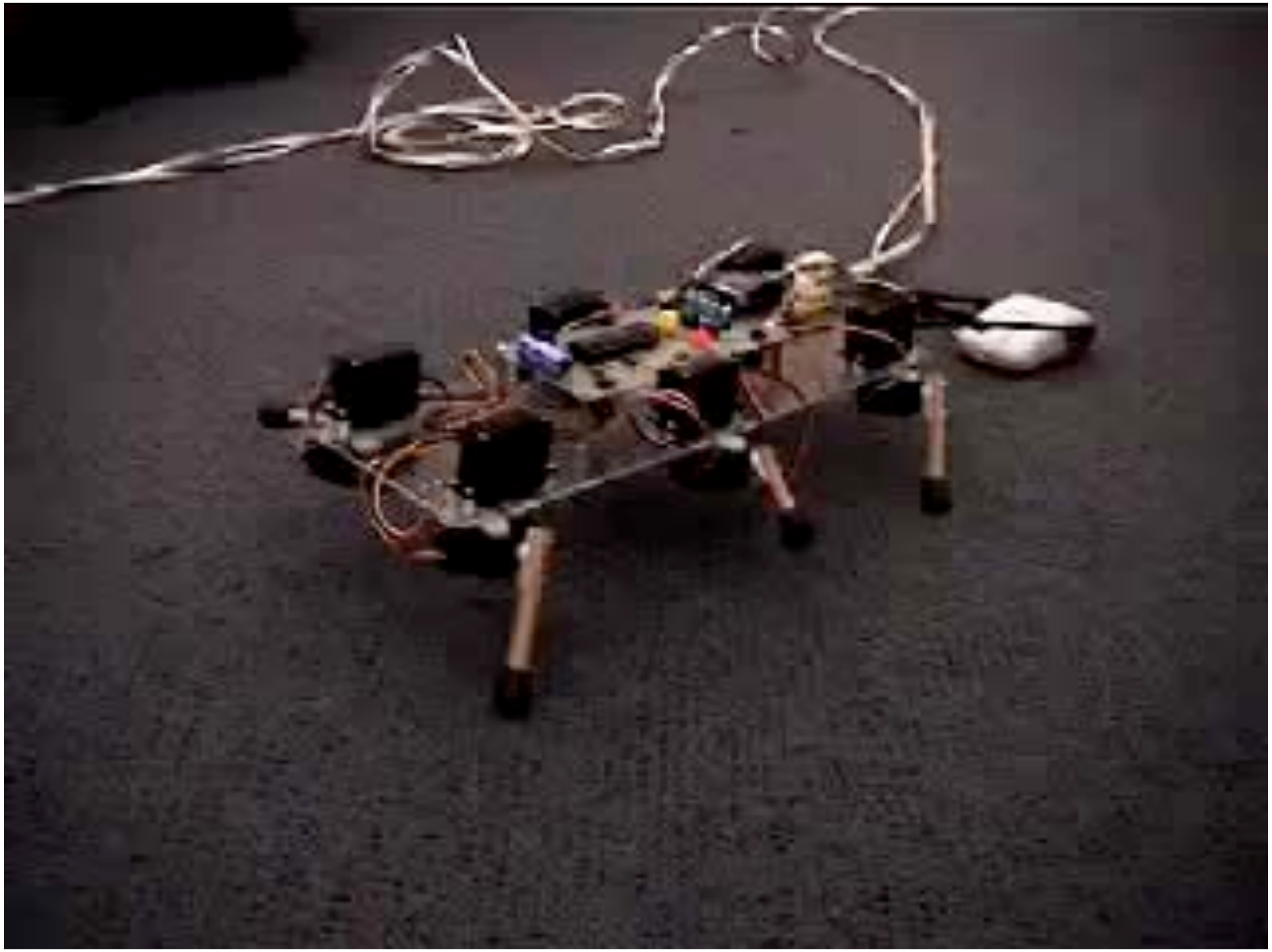
A Simple Organism

Aplysia californica

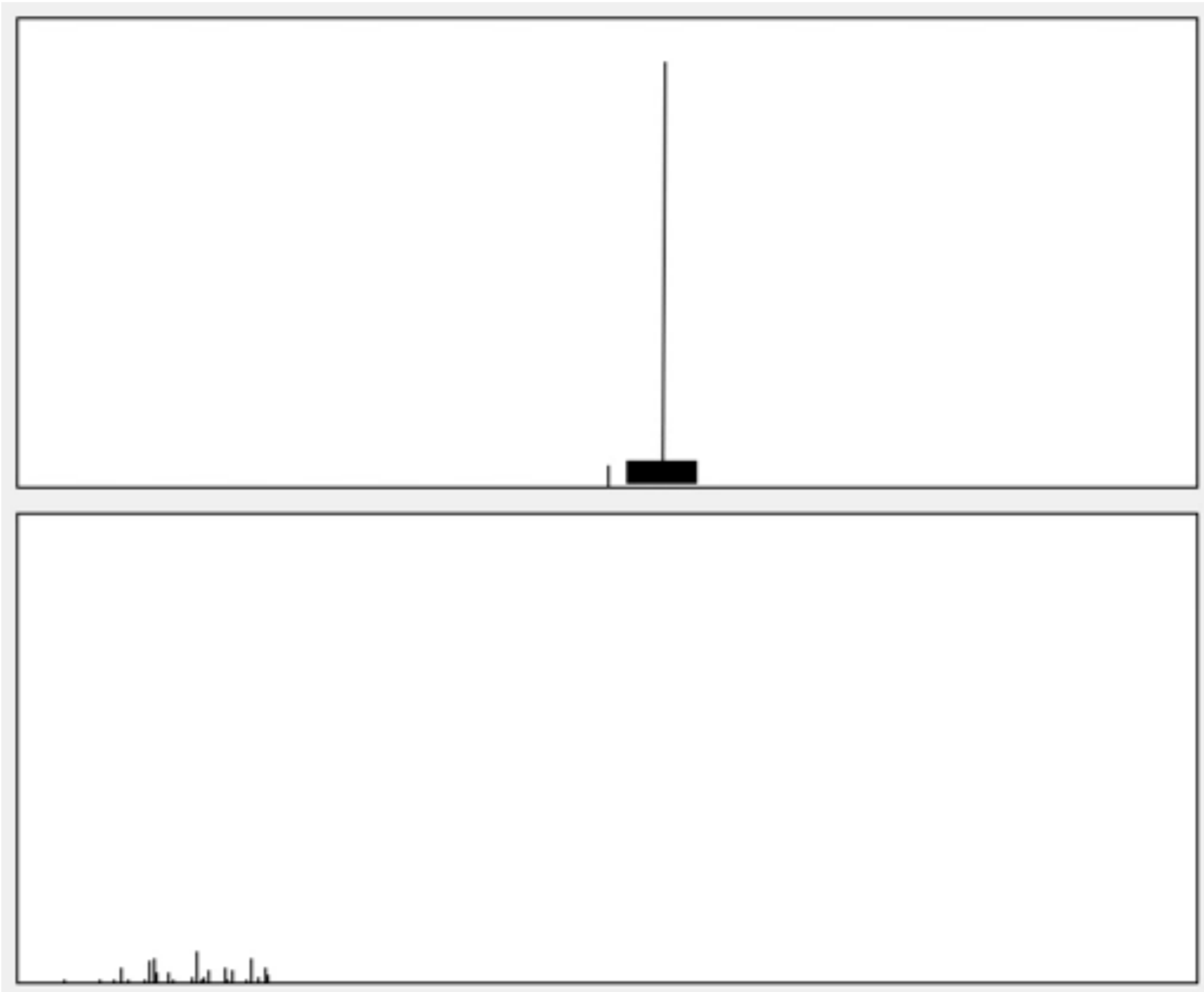




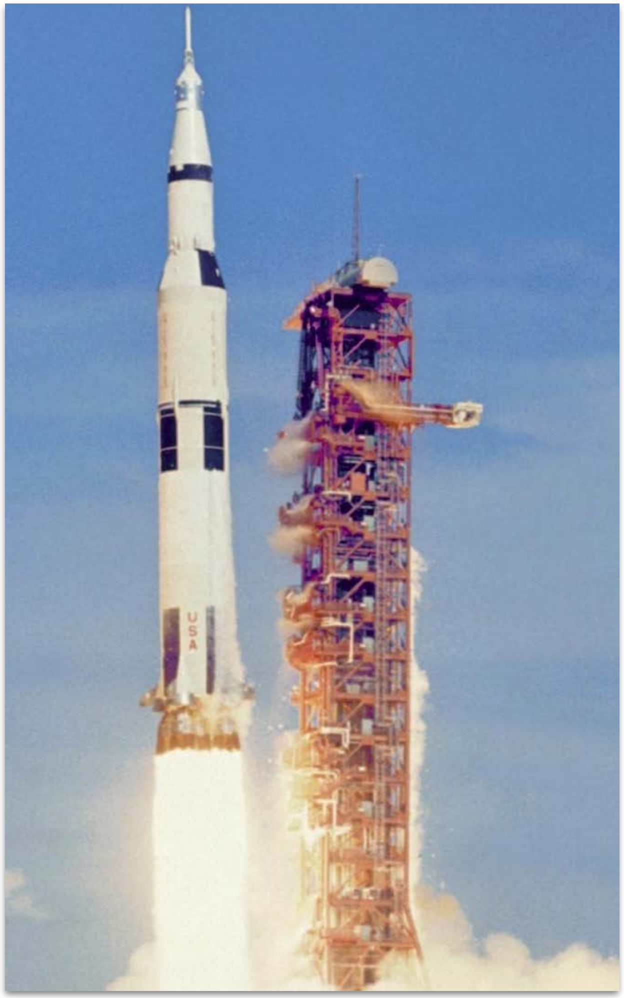
A Simple Robot



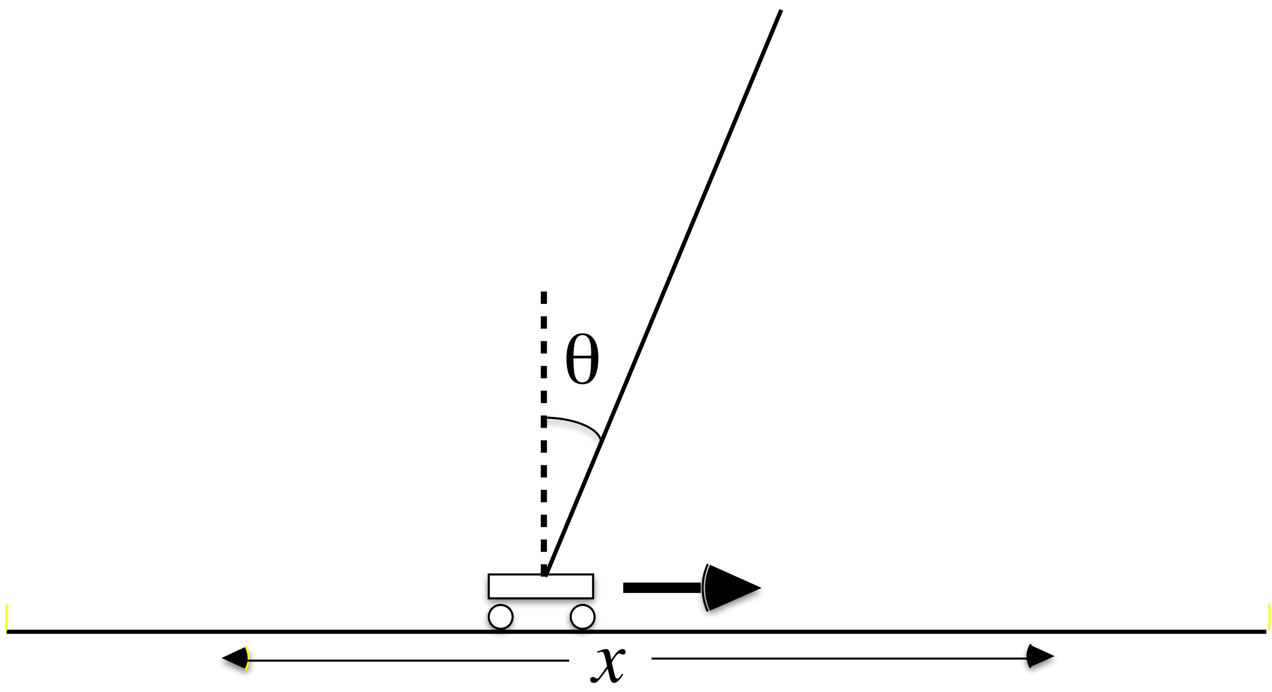
The Pole and Cart



Pole Balancing in Practice

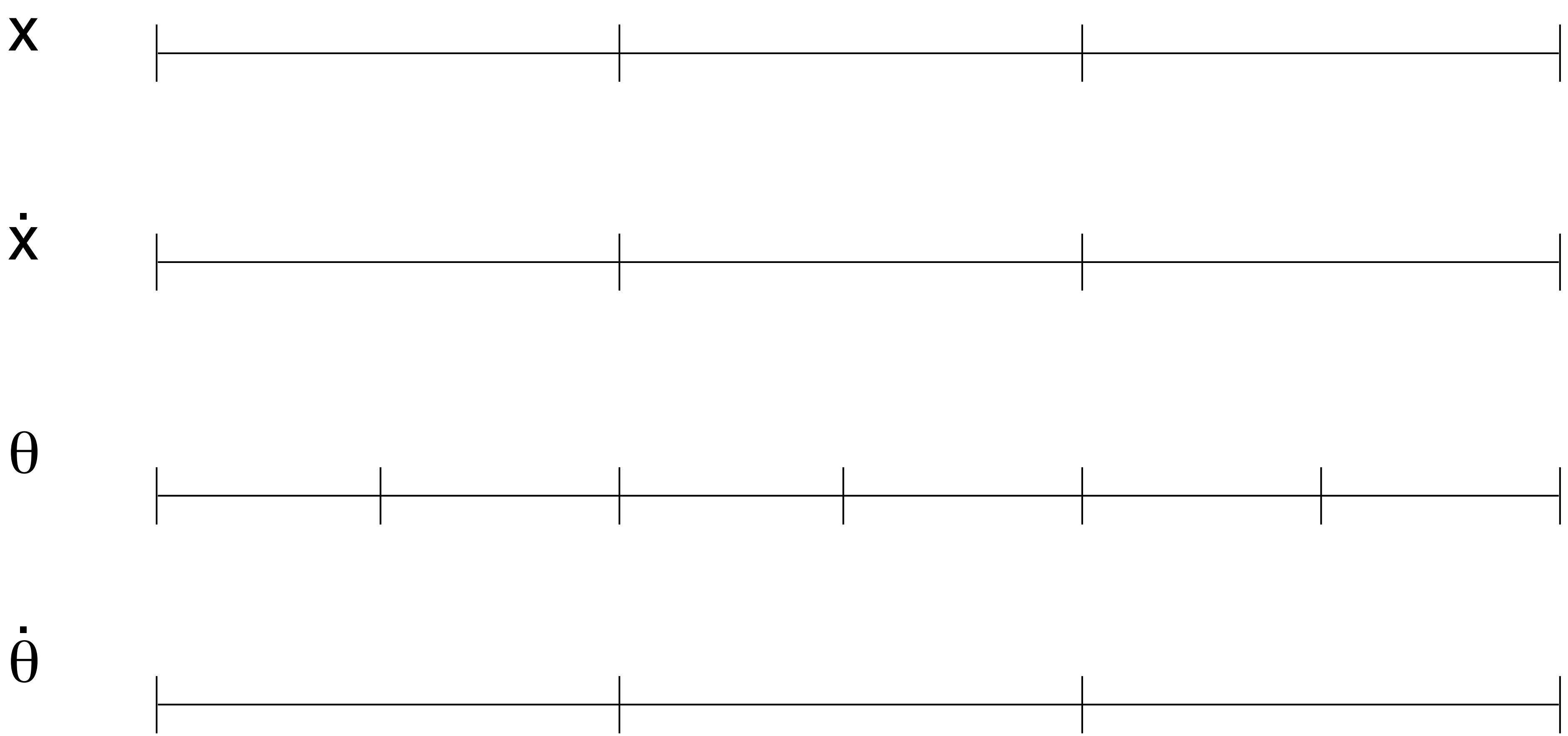


The Pole and Cart



- State is determined by: $x, \dot{x}, \theta, \dot{\theta}$

Discretising the State Space



$$3 \times 3 \times 6 \times 3 = 162$$

Representation

- Controller can be represented as a 162-bit binary string
- Each bit can take values:
  - 0 (meaning push left)
  - 1 (meaning push right)

Learning as Search

- There are 2^{162} possible settings for a controller
- Learning requires searching this huge space to find settings that work
- Must be more intelligent than random guess

Genetic Algorithms

- Genetic algorithms are adaptive general purpose search methods, based on genetic mechanisms.
- Start with a population of agents and modify them through successive generations to optimise some fitness function

Populations

- Each agent (controller) is a *chromosome*.
- A position, or set of positions in a chromosome is called a *gene*.
- The possible values (from a fixed set of symbols) of a gene are known as *alleles*.
- In most GA implementations the set of symbols is {0, 1} and chromosome lengths are fixed.
- Most implementations also use fixed population sizes.

Generations

- Each iteration in a genetic algorithm is called a generation.
- Each chromosome in a population is used to solve a problem.
- Its performance is evaluated and the chromosome is given some rating of fitness.
- The population is also given an overall fitness rating based on the performance of its members.
- The fitness value indicates how close a chromosome or population is to the required solution.

Reproduction

- New sets of chromosomes are produced from one generation to the next.
- Reproduction takes place when selected chromosomes from one generation are recombined with others to form chromosomes for the next generation.
- The new ones are called offspring.
- Selection of chromosomes for reproduction is based on their fitness values.

Genetic Operators

- Operators that recombine selected chromosomes are called *genetic operators*.

- Two common genetic operators are *crossover* and *mutation*.

Crossover

It exchanges portions of the pair to the right of a randomly chosen point called the *crossover point*.

Some Implementations have more than one crossover point.

X = 1001 01011   Y = 1110 10010

Crossover point = 4

O1 = 100110010   O2 = 1110 01011

Offspring produced by crossover cannot contain information that is not already in the population.

Mutation

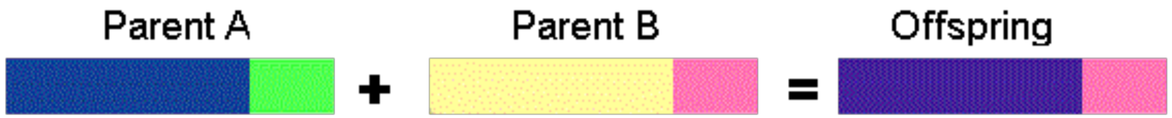
Mutation generates an offspring by randomly changing the values of genes at one or more gene positions of a selected chromosome.

Z = 100101011

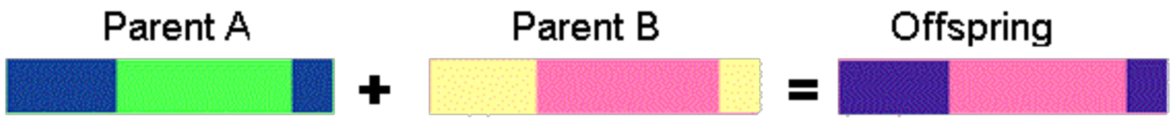
mutate at positions 2, 4 and 9

O = 110001010

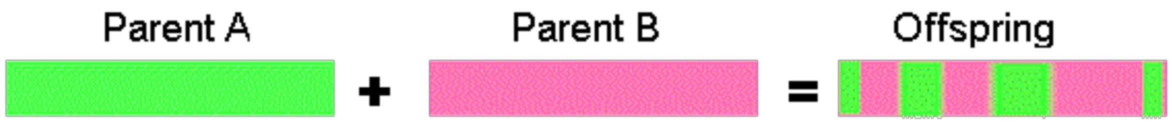
Crossover Variants



11001011+11011**111** = **11001111**



110010**11** + 11**0111**11 = **11011111**



1**10**010**11** + **11**0**111**01 = 11011111

Replacement Strategy

- The number of offspring produced for each new generation depends on how members are introduced so as to maintain a fixed population size.

- In a *pure* replacement strategy, the whole population is replaced by a new one.

- In an *elitist* strategy, a proportion of the population survives to the next generation.

Pole Balancing

- Genetic algorithms can be applied to learning real-time control tasks.
- Pole balancing is a “classic” task used to compare different algorithms

Population Size

- A small population size provides an insufficient sample size over the space of solutions for a problem.
- A large population requires a lot of evaluation and will be slow.
- 50 is usually a good number.

Fitness Value

The number of time steps that a chromosome is able to keep the pole balanced for

Replacement Strategy

- Calculate average fitness of population at end of each generation.
- Individuals whose fitness is below average are replaced by reproduction of above average chromosomes.
- The strategy must be modified if two few or two many chromosomes survive.
- E.g. at least 10% and at most 60% must survive.

Genetic Operators

- All offspring are created by crossover (except when more the 60% will survive for more than three generations when the rate is reduced to only 0.75 being produced by crossover).
- Mutation is a background operator which helps to sustain exploration.
- Each offspring produced by crossover has a probability of 0.01 of being mutated before it enters the population.
- If more then 60% will survive, the mutation rate is increased to 0.25.

Reproduction Strategy

- The number of offspring an individual can produce by crossover is proportional to its fitness:

$$\frac{\text{fitness value}}{\text{population fitness}} \times \text{\#children}$$

#children = total number of individuals to be replaced.

- Mates are chosen at random among the survivors.

Performance

Requires an average of 8165 trials before balancing pole.

Genetic Programming

- Tree Encoding: