Australian National University

Evolutionary Computation

Part of COMP4660/8420:
Neural Networks, Deep Learning and Bio-inspired Computing

2. Genetic Algorithms with Technical Details

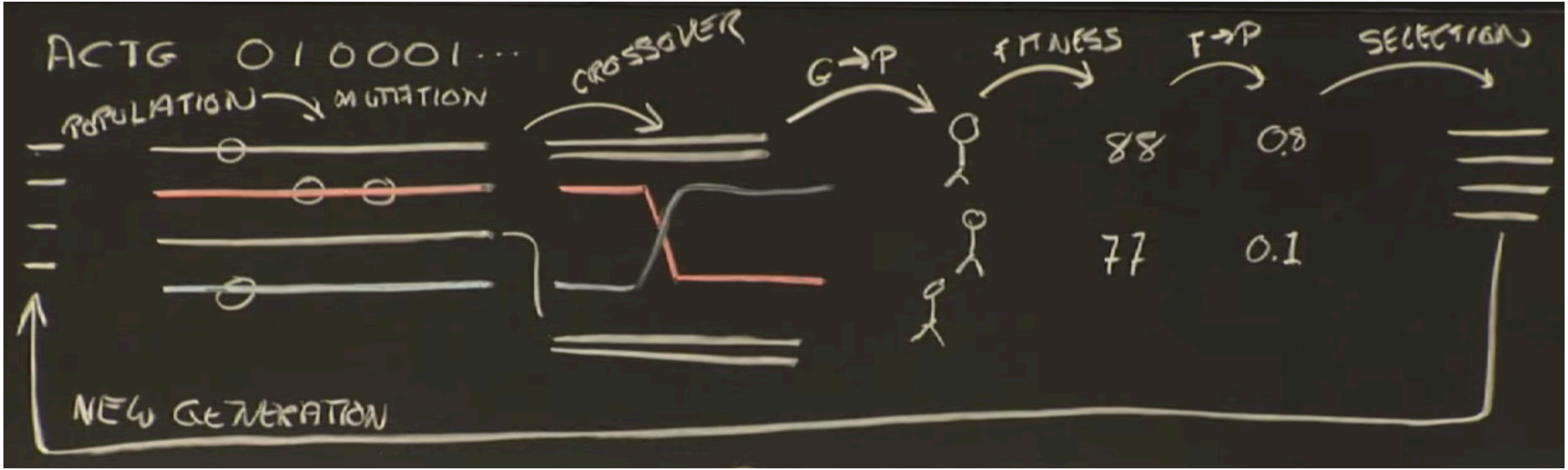
Prof. Tom Gedeon and Mr. Zhenyue Qin (秦震岳)
tom@cs.anu.edu.au; zhenyue.qin@anu.edu.au

Human Centered Computing (HCC) Laboratory
2021 Semester 1

Basic Idea Of Principle Of Natural Selection:

“Select The Best, Discard The Rest”

Procedures



Australian National University

Working Mechanism Of Evolutions and GAs

Evolution:

Initial Population Of Animals

Struggle For Existence-Survival Of the Fittest

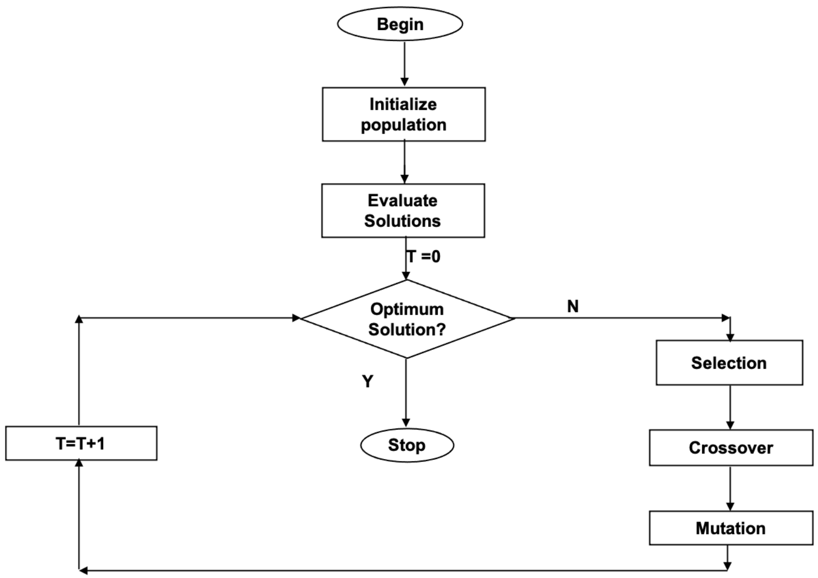
Surviving Individuals Reproduce, Propagate Favorable Characteristics

Millions Of Years

Evolved Species

(Favorable Characteristic Now A Trait Of Species)

GA:



Simple GA

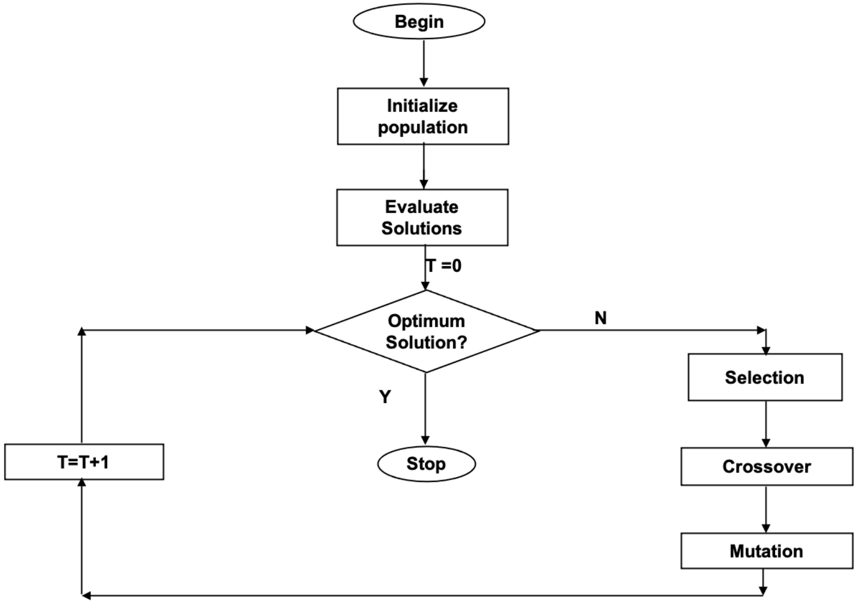
```
Simple_Genetic_Algorithm()
          {
          Initialize the Population;
          Calculate Fitness Function;

          While(Fitness Value != Optimal Value)
                    {
                    Selection;//Natural Selection, Survival Of Fittest

                    Crossover;//Reproduction, Propagate favorable characteristics

                    Mutation;//Mutation
                    Calculate Fitness Function;
                    }
          }
```

Nature to Computer Mapping

Nature	Computer
Population	Set of solutions.
Individual	Solution to a problem.
Fitness	Quality of a solution.
Chromosome	Encoding for a Solution.
Gene	Part of the encoding of a solution.
Reproduction	Crossover

Simulated Evolution

- We need the following:
  - Initialization.
  - Representation of an individual (Encoding).
  - Fitness function.
  - Recombination/Crossover for reproduction.
  - Selection criteria.



More Technical Parts

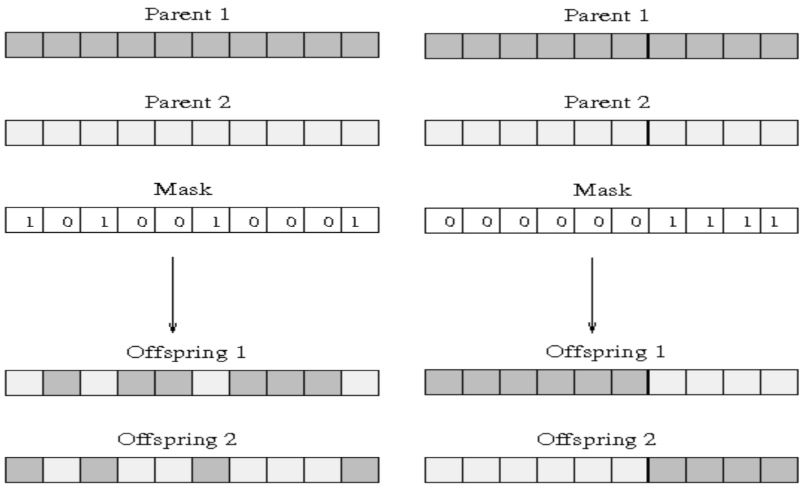
GA Operators and Parameters

Initialization

- Start with a population of randomly generated individuals.
- Use a previously saved population.
- A set of solutions provided by a human expert.
- A set of solutions provided by another heuristic algorithm.
- Randomness is important.

Encoding

- The process of representing the solution in the form of a string that conveys the necessary information.
  - Just as in a chromosome, each gene controls a particular characteristic of the individual, similarly, each bit in the string represents a characteristic of the solution.



(a) Uniform Crossover (b) One-point Crossover

Encoding Method 1

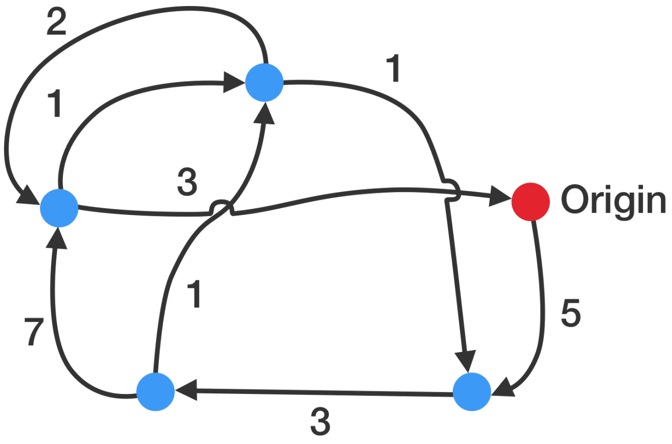
- **Binary Encoding** – Most common method of encoding. Chromosomes are strings of 1s and 0s and each position in the chromosome represents a particular characteristic of the problem.

Chromosome A	101100101100101011100101
Chromosome B	111111100000110000011111

- Example Problem: Knapsack problem.
- The problem: there are things with given value and size. The knapsack has given capacity. Select things to maximize the values.
- Encoding: Each bit says, if the corresponding thing is in the knapsack.

Encoding Method 2

- **Permutation Encoding** – Useful in ordering problems such as the Traveling Salesman Problem (TSP). Example. In TSP, every chromosome is a string of numbers, each of which represents a city to be visited.



Chromosome A	1 5 3 2 6 4 7 9 8
Chromosome B	8 5 6 7 2 3 1 4 9

Encoding Method 3

- **Value Encoding** – Used in problems where complicated values, such as real numbers, are used and where binary encoding would not suffice.
- Good for some problems, but *often necessary to develop some specific crossover and mutation techniques for these chromosomes*.

Chromosome A	1.235 5.323 0.454 2.321 2.454
Chromosome B	(left), (back), (left), (right), (forward)

Fitness Function

A fitness function quantifies the optimality of a solution (chromosome) so that that particular solutions may be ranked against all the other solutions.

- A fitness value is assigned to each solution depending on how close it actually is to solving the problem.
- Ideal fitness function correlates closely to goal + quickly computable.
- Example. In TSP, f(x) is sum of distances between the cities in solution. The lesser the value, the fitter the solution is.
- Similar with loss functions of deep learning programs.

Recombination / Crossover

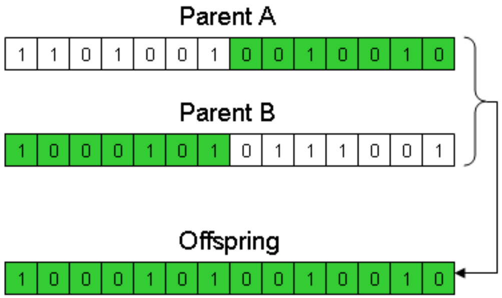
The process that determines which solutions are to be preserved and allowed to reproduce and which ones deserve to die out.

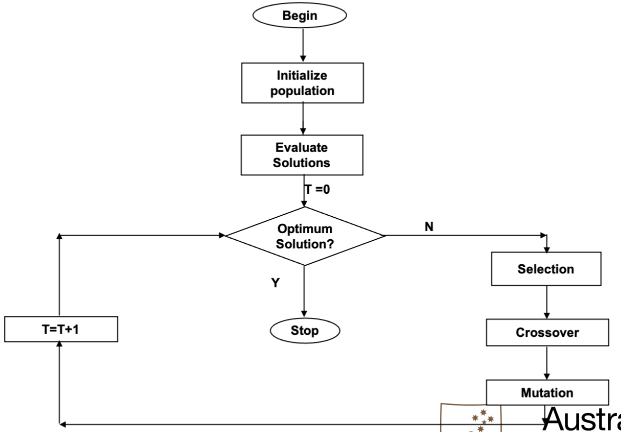
- The primary objective of the recombination operator is to emphasize the good solutions and eliminate the bad solutions in a population, while keeping the population size constant.
- "Selects The Best, Discards The Rest".
- "Recombination" is different from "Reproduction".

Recombination / Crossover

Process:

- Identify the good solutions in a population.
- Make multiple copies of the good solutions.
- Eliminate bad solutions from the population so that multiple copies of good solutions can be placed in the population.

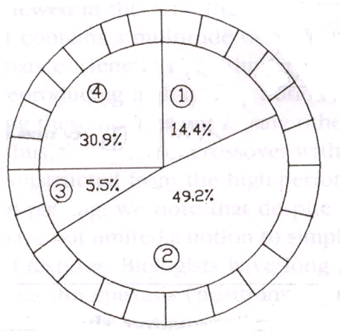




Australian National University

Proportional Selection

- Each current string in the population has a slot assigned to it which is in proportion to its fitness.
- We spin the weighted *roulette wheel* thus defined *n* times (where *n* is the total number of solutions).
- Each time Roulette Wheel stops, the string corresponding to that slot is created.
  - Strings that are fitter are assigned a larger slot and hence have a better chance of appearing in the new population.



$$\varphi_s(\mathbf{x}_i(t)) = \frac{f_\Upsilon(\mathbf{x}_i(t))}{\sum_{l=1}^{n_s} f_\Upsilon(\mathbf{x}_l(t))}$$

n_s is the total number of individuals in the population
$\varphi_s(\mathbf{x}_i)$ is the probability that $\mathbf{x}_i$ will be selected
$f_\Upsilon(\mathbf{x}_i)$ is the scaled fitness of $\mathbf{x}_i$, to produce a positive floating-point value

Australian National University

Proportional Selection Example

No.	String	Fitness	% Of Total
1	01101	169	?
2	11000	576	?
3	01000	64	?
4	10011	361	?
Total		1170	?

Proportional Selection Example

No.	String	Fitness	% Of Total
1	01101	169	14.4
2	11000	576	49.2
3	01000	64	5.5
4	10011	361	30.9
Total		1170	100.0

Proportional Selection

- What if all the fitness values are mutually similar?

Proportional Selection

- What if all the fitness values are mutually similar?
  - All the individuals will have similar chances to be selected.
  - Then, we may need rank-based selection methods.

Tournament Selection

1. Select k individuals from the population and perform a tournament amongst them
2. Select the best individual from the k individuals
3. Repeat process 1 and 2 until you have the desired amount of population

If the tournament size is 3, then what is the probability of selecting an Individual whose fitness is worse than the median?

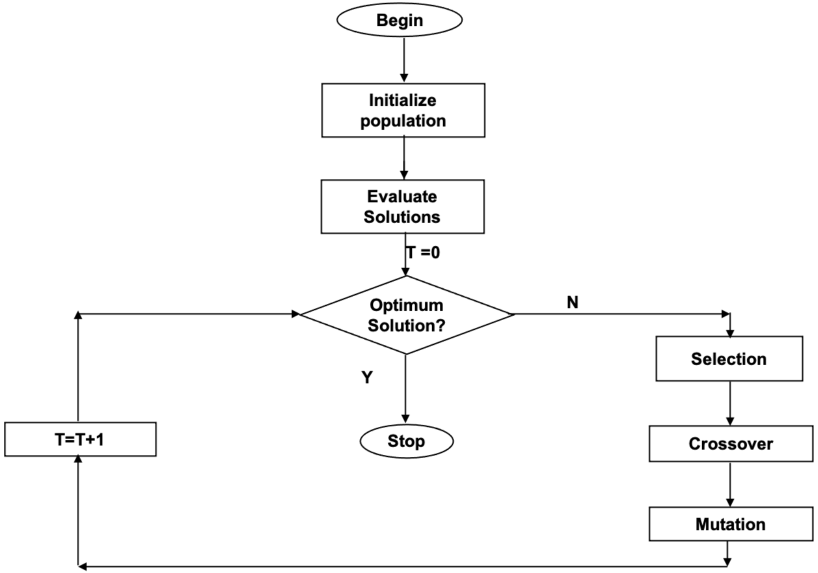
Tournament Selection

1. Select k individuals from the population and perform a tournament amongst them
2. Select the best individual from the k individuals
3. Repeat process 1 and 2 until you have the desired amount of population

If the tournament size is 3, then what is the probability of selecting an Individual whose fitness is worse than the median?
Answer: 1/8 (1/2 * 1/2 * 1/2)

Reproduction

- Producing new generations of improved solutions by selecting parents with higher fitness ratings or by giving such parents a greater probability of being contributors and by using random selection



Reproduction 1: Crossover

- It is the process in which two chromosomes (strings) combine their genetic material (bits) to produce a new offspring which possesses both their characteristics.

- Two strings are picked from the mating pool at random to cross over.
- The method chosen depends on the Encoding Method.

Crossover: Single Point Crossover

- **Single Point Crossover** - A random point is chosen on the individual chromosomes (strings) and the genetic material is exchanged at this point.

Chromosome1	11011 | 00100110110
Chromosome 2	11011 | 11000011110
Offspring 1	11011 | 11000011110
Offspring 2	11011 | 00100110110

Australian National University

Crossover: Two Point Crossover

- **Two Point Crossover** - Two random points are chosen on the individual chromosomes (strings) and the genetic material is exchanged at these points.

Chromosome1	**11011 | 00100 | 110110**
Chromosome 2	**10101 | 11000 | 011110**
Offspring 1	**10101 | 00100 | 011110**
Offspring 2	**11011 | 11000 | 110110**

NOTE: These chromosomes are different from the last example.

Uniform Crossover

- **Uniform Crossover** - Each gene (bit) is selected randomly (or by a mask) from one of the corresponding genes of the parent chromosomes.

Chromosome1	**11011 | 00100 | 110110**
Chromosome 2	**10101 | 11000 | 011110**
Offspring	**10111 | 00000 | 110110**

NOTE: Uniform Crossover yields ONLY 1 offspring.

Crossover (contd.)

- Crossover between 2 good solutions MAY NOT ALWAYS yield a better or as good a solution.
- Since parents are good, probability of the child being good is high.
- If offspring is not good (poor solution), it will be removed in the next iteration during “Selection”.

Elitism

Elitism is a method which copies the best chromosome to the new offspring population before crossover and mutation.

- When creating a new population by crossover or mutation the best chromosome might be lost.
- Forces GAs to retain some number of the best individuals at each generation.
- Has been found that elitism significantly improves performance.

Reproduction 2: Mutation

- It is the process by which a string is deliberately changed so as to maintain diversity in the population set.
- We saw in the giraffes' example, that mutations could be beneficial.
- Mutation Probability- determines how often the parts of a chromosome will be mutated.

Example Of Mutation

- For chromosomes using Binary Encoding, randomly selected bits are inverted.

Offspring	**11011 00100 110110**
Mutated Offspring	**11010 00100 100110**

NOTE: The number of bits to be inverted depends on the Mutation Probability.

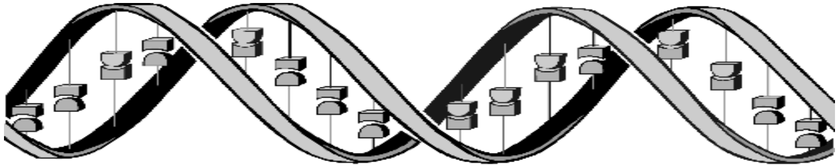
Termination Criteria

There exist three termination condition type:

- 1-Time:in seconds, in minutes and may be in hours according to the problem that you have it.
- 2-Number of generations: in hundreds, in thousands may be in millions according to the problem you have it.
- 3-convergence: when 95% of populations have the same fitness value we can say the convergence started to appear and the user can stop its genetic program to take the result.

When to Use a GA

- Alternate solutions are too slow or overly complicated.
- Need an exploratory tool to examine new approaches.
- Problem is similar to one that has already been successfully solved by using a GA.
- Want to hybridize with an existing solution.
- Benefits of the GA technology meet key problem requirements.



Genetic Algorithm Business Areas

- Schedule Assembly lines at Volvo Truck North America.
- Channel 4 Television (England) to schedule commercials.
- Driver scheduling in a public transportation system.
- Jobshop scheduling.
- Assignment of destinations to sources.
- Trading stocks.
- Productivity in whisky-making is increased.
- Often genetic algorithm hybrids with other AI methods.

Summary

- Genetic Algorithms (GAs) implement optimization strategies based on simulation of the natural law of evolution of a species by natural selection.

- GAs have been applied to a variety of function optimization problems, and have been shown to be highly effective in searching a large, poorly defined search space even in the presence of difficulties such as high-dimensionality, multi-modality, discontinuity and noise.

References

- https://www.freepik.com/free-vector/giraffe-standing-tall-eating-some-tree-leafs_1311438.htm
- Hasan OGUL’s genetic algorithm slides.