

V5

- 1) Encoding - First step
→ Process of representing individual genes.
→ The process can be performed using bits, numbers, trees, arrays, lists or any other objects.
→ Encoding mainly depends on solving the prob.

Ways of encoding - BODH PVT
① Binary ② Octal ③ Hexadecimal ④ Permutation
⑤ Value ⑥ Tree

Binary Encoding -

- Every chromosome is a string of bits 0 or 1
eg Chromosome A → 1011001011010101
Chromosome B → 1111111001100111
→ Gives many possible chromosomes even with a small no. of alleles.
→ Eg of problem - Knapsack problem.

Things with given value & size, knapsack has given capacity.
Select sets to maximize value of things in knapsack.

Encoding:- Each bit says if the corresponding thing is in the Knapsack

Permutation Encoding:-

- Used in ordering prob,
→ Every chromosome is a string of nos, which represents no in a sequence
eg Ch A 1 5 3 2 6 4 7 9 8
Ch B 0 5 6 7 2 1 5 3 8

- Eg of problem - Travelling Salesman (TSP)
Cities to given distance b/w them TS has to visit all of them with min travel dist

Encoding → Chromosome says order of cities in which salesman should visit them.

- Value Encoding
- Problems with some complicated values such as real nos. use it.
 - Every ch is a string of some values.

ch A	1.23 - 5.32	0.45	1.12	
ch B	ABC/FDE/BJK/LMN			
ch C	(back),	(back),	(front),	(right)

- Eg of prob: finding weights for NN

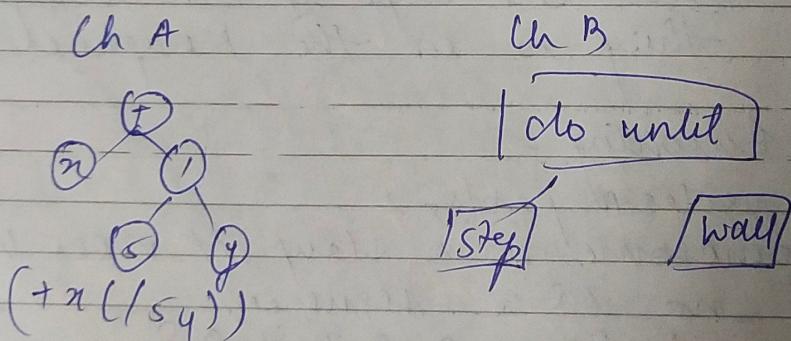
Some NN with given architecture.

Find weights for inputs of neurons to train the network for wanted op.

Encoding: Real values in ch represent corresponding weights for inputs

Tree Encoding

- Tree encoding for programs or expressions, for genetic alg.
- Every ch is a tree of some objects, such as func ~~or~~ commands in program



- Eg of prob - finding a func from given values If & Op values given. Find a func.

Encoding → Chromosomes are func represented in a tree.

Plan et not

Selection - selecting individuals (parents) with good fitness score, and allowing them to mate and recombine their genes on to the next gen (offspring)

Types - RR R³ TB SSE

- (1) Roulette wheel
- (2) Rank
- (3) Random
- (4) Tournament
- (5) Boltzmann
- (6) Steady state
- (7) Stochastic
- (8) Unbiased sampling
- (9) Elitism

Roulette wheel -

- Parents selected acc to fitness
- Better ch, more chances of selection
- ~~Elites~~ → ~~Stem~~

Rank -

- First rank the population and then only ch receive fitness from this ranking
- Worst will have fitness 1, 2nd worst 2 etc
- Best will have fitness N.

Steady State

- Big part of ch should survive to next gen
- In every gen, are selected a few (good - with high fitness) ch to create a new offspring
- then some (bad - with low fitness) ch are replaced by new offspring. Rest of population survives to next gen.

Elitism -

- First copies of best ch to new popn.
- Rest done in classical way, thus not losing the best ch. →