2 - Main principles of metaheuristics, neighbourhood, movements, exploration operator, population metaheuristics

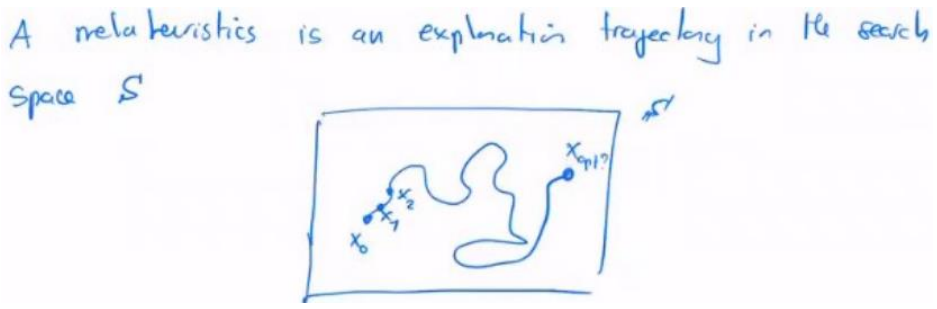
GOAL of metaheuristics: explore a large search space in a clever way. It is needed to solve large problems, for which only exponential algorithms are known

- ➔ It is a compromise between CPU time and quality of solution
- ➔ No guarantee on the quality of the solution
- ➔ As opposed to heuristic where the algorithm is specific to a problem, metaheuristic algorithm can be applied to many different problems

Characterization:

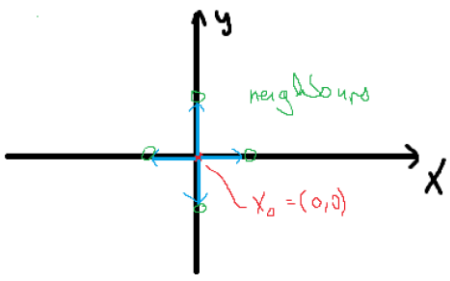
- no hypothesis on the mathematical properties of the fitness function
- Require guiding parameters that defines how the space should be explored. The quality of the solution will depend on how good these parameters are
- need a starting point (usually random)
- need a stopping condition (iteration, no more improvement, time, value …)
- inspired by natural processes (ant system, beehive …)
- can be parallelisable
- most of them are stochastic, use random numbers to guide a search

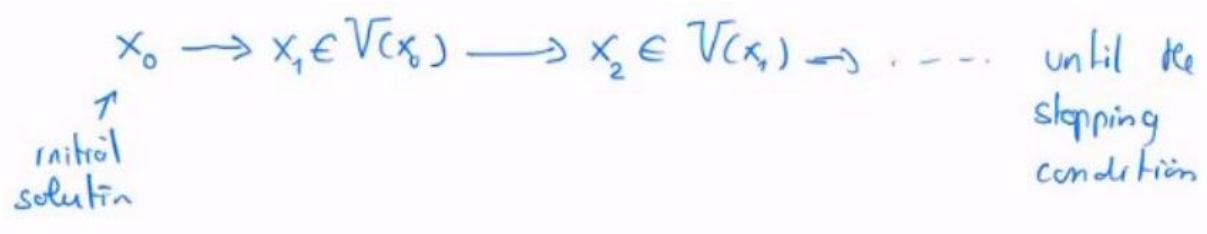
Main principle:



To move across the search space, we use the concept of neighbourhood.

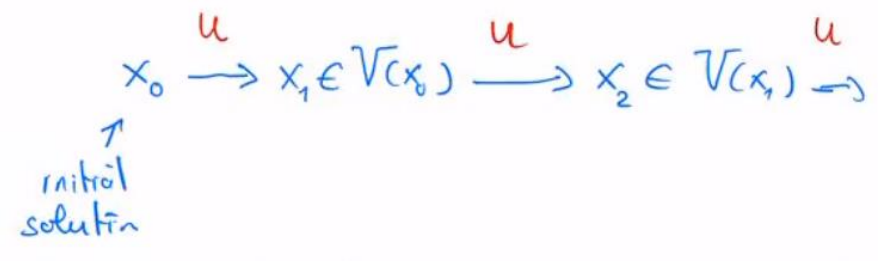
- ➔ If we are currently exploring a point x, the next point we explore is a neighbour of x
- ➔ The neighbourhood is found by doing elementary transformations on the current explored state. neighbourhood of x is called V(x)





Search operator:

- ➔ When we are at a state x, we find its neighbourhood V(x), and now we need to choose which neighbour to explore next.
- ➔ This is done via the search operator -> $u : V(x) \rightarrow y \in S$



Population metaheuristics

So far we assumed that at each iteration we consider only one possible solution:

$x_0 \rightarrow x_1 \in V(x_0) \rightarrow x_2 \in V(x_1) \dots$

But we could also consider a population of solutions:

$P_0 = \{ x_1^{(0)}, x_2^{(0)}, \dots x_M^{(0)} \}$ M possible solution at iteration 0