Artificial Intelligence

Solving problems by searching II

"Artificial Intelligence - A Modern Approach", Chapters 3 and 4

POLITÉCNICO DE LEIRIA

Slides by Carlos Grilo

Iterative improvement

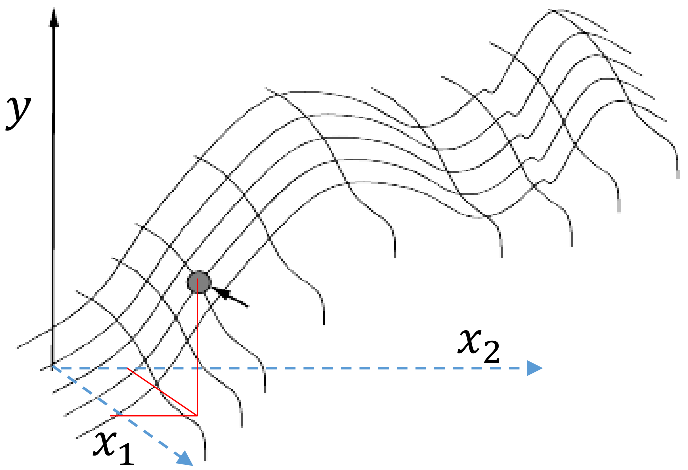
- In these slides we will address algorithms to solve iterative improvement problems
- In this type of problem, the solution consists in a state with some specific properties and not a sequence of actions
- The algorithms used in these problems start with a complete configuration of the problem (state) and proceed by making modifications to that configuration in order to improve it
- These algorithms belong to a class of algorithms called iterative improvement search

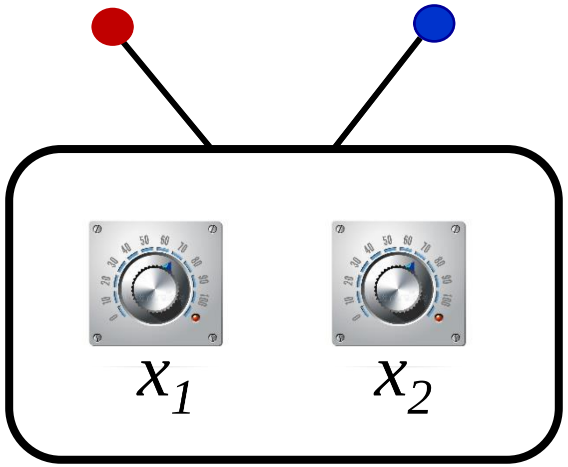
Nomenclature

- In the previous slide, you may have noticed that the therms state, configuration and solution are used reciprocally
- Depending on the algorithm or the context, the terms node and individual may also be used with the same meaning
- In these slides we will use the term solution

Iterative improvement

- Analogy: the solutions are represented as points in a landscape





x_1	x_2

- The elevation of each point in the landscape corresponds to the value of the corresponding solution as a solution to the problem

- Departing from an initial state the aim is to reach the highest or the lowest point in the landscape (the so called global optimum)

Iterative improvement

- Let us consider a function $y = f(x_1, x_2, \ldots, x_n)$
  - y may represent, for example, profit, nº of produced units, etc.
  - $x_1, x_2, \ldots, x_n$ may represent, for example, the pressure level, temperature, etc.

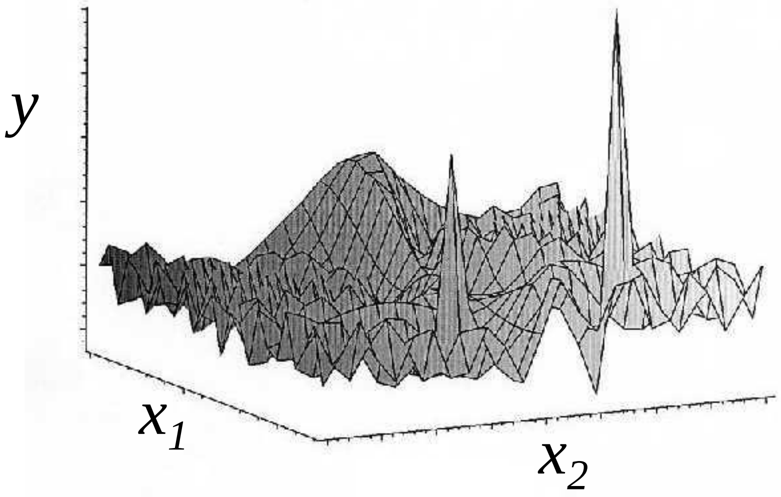
- Suppose that we want to know the combination of $x_1, x_2, \ldots, x_n$ values that maximizes the value of y
  - Please, notice that each combination of $x_1, x_2, \ldots, x_n$ represents a possible solution

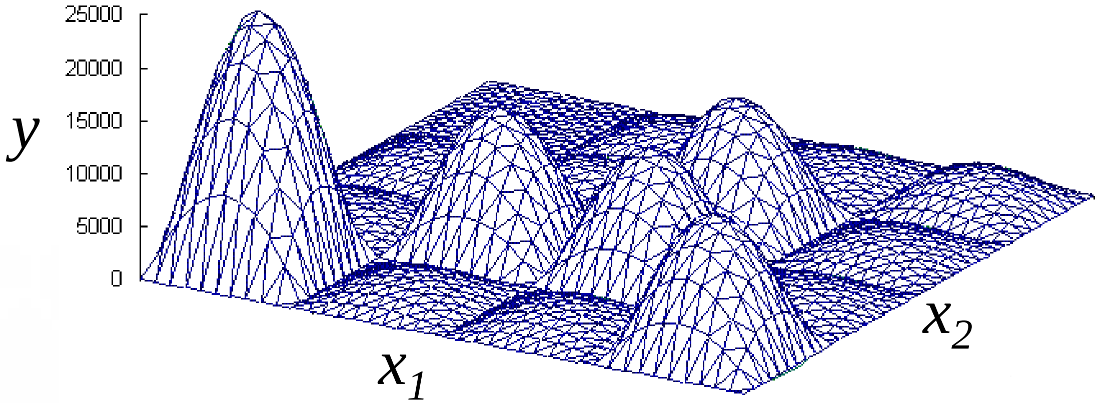
- Question: How can we do this?

- We will call evaluation function to f since it evaluates the quality of the solution (it is also called the objective function)

Solution space

- Solution space: set of possible combinations of the x_1, x_2, ..., x_n values





Note: the solution space is also called search space

Exhaustive search

- Approach 1: exhaustive analysis of the solution space
- Problem: the solution space is often too large or even infinite, which turns impossible to exhaustively analyse it

Random search

- Approach 2: search the solution space randomly
- Problems:
  - Extremely inefficient
  - It doesn’t considers any information about already explored solutions

Differential calculus

- Approach 3: differential calculus
- Problems
  - What if $f(x_1, x_2, ..., x_n)$ is discontinuous, as it is common in real problems?
  - Local maxima/minima (see slide 15)

Hill-climbing

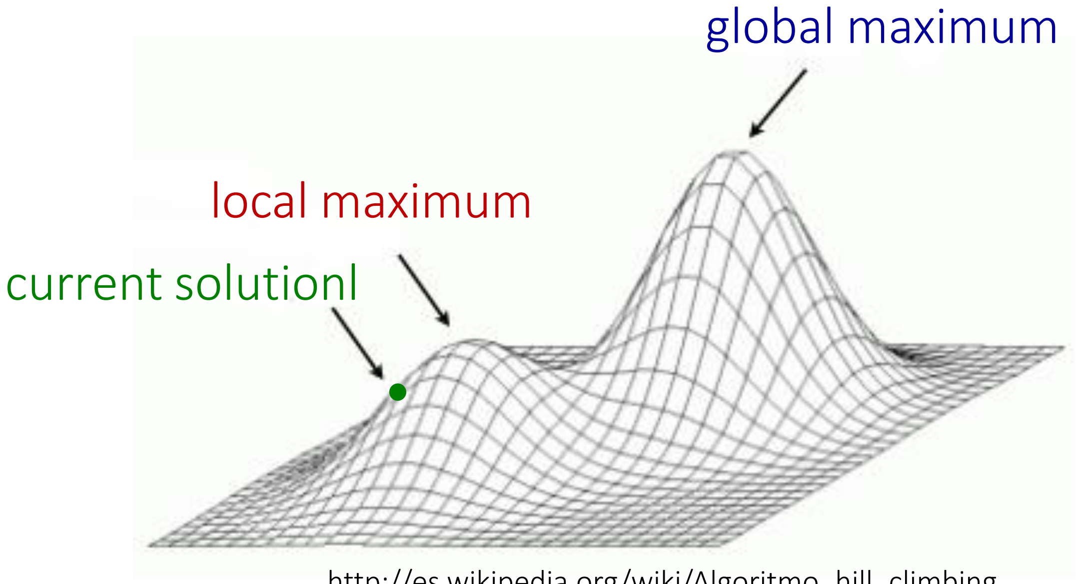
- Algorithm (steepest-ascent version):
  1. Generate a solution (point in the solution space) and evaluate it
  2. Apply operators to that solution generating *s* other solutions (the successors)
  3. Choose, among the successors, the one with the highest/lower *f* value and use it to continue the search
  4. Go to 2

- Fragilities: it may become trapped in
  - Local maxima/minima
  - plateaus
  - ridges

The first-choice hill climbing implements stochastic hill climbing by randomly selecting the next operator to apply in order to generate a successor until one is generated that is better than the current state. This is a good strategy when a state has many (e.g., thousands) of successors.

- In these situations the usual procedure is to restart the search in another randomly chosen point of the solution space

Global maxima



http://es.wikipedia.org/wiki/Algoritmo_hill_climbing

- A local maximum is a point in the space *x* for which the value of *f* is the highest in the neighbourhood of *x*, existing, however, another point outside the neighbourhood of *x* for which the value of *f* is largest
- Note: we can talk about local minima as well in minimization problems

Simulated annealing

- Principle: Allow that, once in a while, a solution worst than the current solution may be chosen
- Goal: escape to local maxima

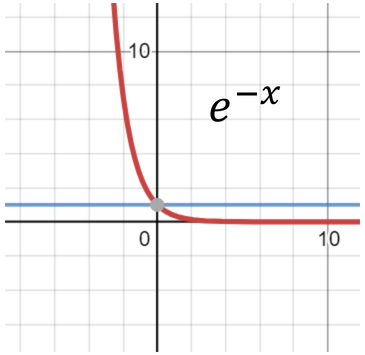
Simulated annealing

- Based on the first-choice hill climbing: it randomly chooses a successor s_i
- s_i is always chosen if it is best than the current solution
- But, if s_i is worst than the current solution, it is chosen with some probability which decreases exponentially with the difference between the f value of s_i and the f value of the current solution
- The probability of s_i being chosen depends also on a parameter T (temperature). The largest the value of T, the largest the probability that s_i is chosen. The value of T diminishes during the search process according to some function

Simulated annealing

- Probability of s_i being chosen if it is worst than S (the current solution)

$$e^{-\frac{(f(s_i)-f(S))}{T}}$$



Note: we are considering a minimization problem, that is, s_i is better than S if $f(s_i) < f(S)$

Beam search

- Maintains a population of *n* solutions instead of just one

- In each iteration: for each solution from the current population, generate *k* successors, resulting in *k* x *n* successors; then, choose the best *n* successors

- Note: in this course you learn about another (different) algorithm with the same name

Other iterative improvement algorithms

- Tabu search
- Evolutionary algorithms (genetic algorithms, etc.)
- Ant colony optimization
- Particle swarm optimization
- Bees algorithm
- Etc., etc…