An introduction to artificial intelligence and machine learning

M. Salomon
FEMTO-ST Institute - DISC Department - AND Team
Univ. Bourgogne Franche-Comté (UBFC), France

January 17, 2024

1st year Master in Computer Science - Internet of Things

Outline

1. What is Artificial Intelligence (AI)?
2. Artificial intelligence approaches.
3. What is Machine Learning (ML)?
4. ML example: Artificial Neural Networks (ANN concepts).
5. Optimization: how to determine “the” minimum of a function.

What is Artificial Intelligence (AI)?

- What is an intelligent task?
  - Understanding a natural language
  - Driving a car
  - Demonstrating new theorems
  - Solving mathematical equations
  - Playing games
  - etc.
- Giving a precise definition is difficult
  - Difficult to define intelligence
  - Many fields are concerned by AI
- A first rough definition

A system able to reproduce the human behavior

What is Artificial Intelligence (AI)?

- The pioneers: John McCarthy and Marvin Minsky
  - McCarthy coined the term *"artificial intelligence"* in 1955
  - Organized the first conference on AI in 1956
- A definition of AI according to Marvin Minsky

"The science and engineering that tries to make machines intelligent, trying to get them to understand human language and **to reach problems and goals as well as a human being**".

"The construction of programs that complete tasks that are, **for the moment**, **more satisfactorily performed by human beings** because they require high level mental processes such as perceptual **learning**, **memory organization**, and critical **reasoning**".

What is Artificial Intelligence (AI)?

- Several dimensions
  - Definition based on reasonableness (logic)
  - Definition centered on human (cognitive science)

Thinking and reasoning processes

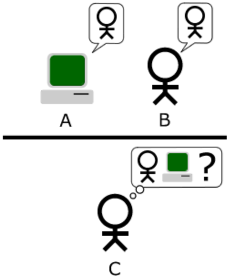
Human perf.	Systems thinking like humans	Systems having a rational thinking	Ideal perf.
	Systems acting like humans	Systems acting in a rational way	

Behavior

What is Artificial Intelligence (AI)?

- Systems thinking like humans
  - How does our brain work?
  - Implement models and compare with humans
- Systems acting like humans
  - If a machine exhibits an intelligent behavior, it is intelligent
  - Turing test (1950)
    - Imitation game

Player C, through a series of written questions, attempts to determine which of the two other players A and B is a computer



What is Artificial Intelligence (AI)?

- Systems having a rational thinking
  - Aristote logic: formal rules for correct reasoning
    - All men are mortal / Socrates is a man
      $\rightarrow$ Therefore, Socrates is mortal
  - Formal logic to prove or disprove things
- Systems acting in a rational way
  - Choose the action that maximizes a goal according to the available informations
  - Rational / intelligent agent (***sensing-reasoning-acting***)

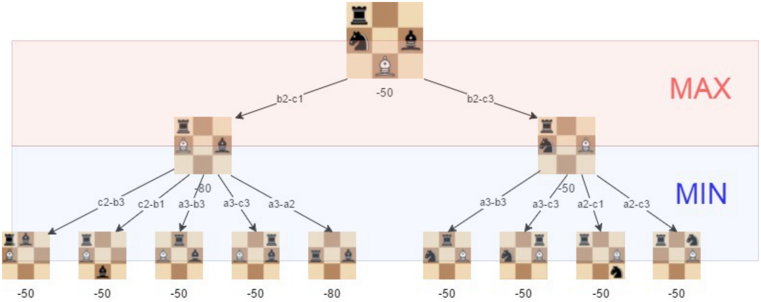
- an entity that perceives its environment (**observes**)
- acts (**able to decide**) to fulfill goals according to its
- abilities and knowledges (**reasoning**, **modelling**)
- adapts to change (**learning**)

Artificial intelligence approaches

- 1940 - 1970 : two streams appear
  - **Cognitive systems** → Making a mind
    - Simulate human thinking through symbolic manipulation
    - Knowledge-based systems
  - **Connectionist systems** → Modelling the brain
    - Simulate human brain activity
    - Artificial Neural Networks (ANN)
  - Some dates
    - 1943 (McCulloch et Pitts) → artificial neuron model
    - 1944 (von Neumann et Morgenstern) → gaming theory
    - 1955 (Newell et Simon) → **The Logic Theorist**
    - 1956 (Simon, Shaw, Newel) → The General Problem Solver
    - 1957 (Rosenblatt) → first neural network (perceptron)

Artificial intelligence approaches

- 1960 - 1970 : some progresses, but growing pessimism
  - 1965 (Feigenbaum) → first expert system
  - 1967 (Greenblatt) → first AI for Chess game
    - Satisfactory: able to beat an average player
    - Minimax algorithm (von Neumann)
  - 1969 (Minsky) → single perceptron's limitations



Artificial intelligence approaches

- 1970 - 1980 : dark age (AI winter), despite some progress
  - 1972 (Colmerauer) → Prolog
  - 1979 (Shortliffe) → first medical diagnosis expert system
- Since the 1980's: revival and diversification
  - Artificial neural networks
    - Backpropagation algorithm, etc.
  - Genetic algorithms
    - Simulate the process driving biological (natural) evolution
  - Multi-agents systems
    - Interacting agents with a collective behavior to solve a task
  - etc.
- Since the 2010's: boom of connectionism (deep neural net.)

Key reason of today success: increasing computing power and data

Solving "larger" problems; design of more complex AI

Some key successes of AI

- 1997 → Deep Blue defeats Kasparov (Chess game)
  - Won 3.5 - 2.5 (last game in 18 moves)
  - "Brute force" approach
  - Analysis of 200 millions de positions per second
- 2006 / 2009 → Monte-Carlo Tree Search (MCTS) / Google car
- 2011 → Watson won Jeopardy quiz
- 2016 → AlphaGo defeats Lee Sedol (Go game)
  - Merge Machine Learning → Deep Learning
    - Deep neural networks
    - 2 networks: one to evaluate and one to select (a move)
  - and tree search → MCTS
- 2017 → AlphaGo Zero and AlphaZero
  - "Simplification" of AlphaGo
  - AlphaZero : best IA for the Go, Chess, and Shogi games
- 2019 → AlphaStar
  - AI for Real-Time Strategy (RTS) game called StarCraft

Some key successes of AI

- 2023 $\rightarrow$ ChatGPT

M What are the departments of STGI UFR in UFC ?

The STGI (Sciences et Technologies pour l'Ingénieur) UFR (Unité de Formation et de Recherche) is a faculty of the University of Franche-Comté (UFC) in France, which offers undergraduate and graduate programs in engineering and technology. The STGI UFR is composed of four departments, which are:

1. Department of Computer Science (Informatique) - This department focuses on computer science and software engineering, including topics such as algorithms, programming languages, databases, artificial intelligence, and computer networks.
2. Department of Electrical Engineering (Electronique) - This department covers the study of electronics and electrical engineering, including topics such as circuit design, control systems, digital signal processing, and telecommunications.
3. Department of Mechanical Engineering (Mécanique) - This department focuses on mechanical engineering and related fields, including topics such as mechanics of

AI topics according to CNRS

(French National Centre for Scientific Research)

Several branches, more or less separated and specialized

GDR IA - CNRS sur Aspects Formels et Algorithmiques de l'IA

- Représentation des connaissances et modélisation des raisonnements
- Incertitude, modèles graphiques, réseaux bayésiens
- Contraintes et SAT
- Apprentissage
- Planification et recherche heuristique
- Systèmes multi-agents et décision collective

Applications of AI

- Knowledge-based systems
- Planning
- Automatic translation of natural language (NLP)
- Autonomous navigation (car, unmanned aerial vehicle, etc.)
- Medical diagnosis
- Robotics
- Games (Go, Chess, Starcraft, ...)
- etc.

What is Machine Learning (ML)?

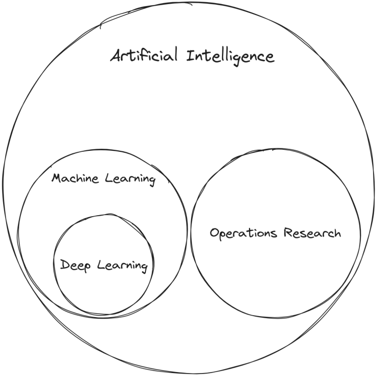
- A definition of ML

  "Machine learning is about making algorithms **learn from data or experience** to make **predictions** on previously **unseen input**".

- Interest of ML compared to Operations Research (OR)
  - OR methods excel when the model of the system is known and the problem can be mathematically formulated
  - ML is effective for
    - Problems with ambiguous or incomplete data
    - Systems where the underlying model is unknown or constantly changing
    - Problems where input features need to be inferred

What is Machine Learning (ML)?

- Venn Diagram representing the relationship between AI, ML, Deep Learning, and OR



What is Operations Research (OR)?

- A definition of OR

  "Operations research represents the application of analytical techniques to decision-making. Simply put, it is about making better (or the best) decisions".

- OR can be broadly categorized in two types of problems

  1. Feasibility prob. $\rightarrow$ find any solution that meets all criteria

     "Determine whether there exits a (any) solution that satisfies a given set of constraints".

  2. Optimization prob. $\rightarrow$ find the "best" possible solution

     "Find the best possible solution (may not be unique) according to some criterion (objective: maximize or minimize a specific measure), while still ensuring that the solution satisfies all given constraints".

What is Operations Research (OR)?

- Classification of optimization problems

- Optimization
  - Objective
    - Single objective optimization
    - Multiobjective optimization
    - Feasibility problem
  - Constraint
    - Unconstrained optimization
    - Constrained optimization
  - Landscape
    - Unimodal (including convex optimization)
    - Multimodal optimization
  - Function form
    - Linear programming
    - Nonlinear programming
      - Quadratic programming
      - ...
  - Variables/response
    - Discrete
      - Combinatorial optimization
      - ...
    - Integer
      - Integer programming
      - Binary integer programming
    - Continuous
    - Mixed integer programming
  - Determinacy
    - Deterministic (all the above)
    - Stochastic optimization (e.g. robust optimization)

What is Operations Research (OR)?

- Classification of optimization solvers

- Algorithms
  - Deterministic
    - Linear programming (simplex method, ...)
    - Integer programming (branch and bound, ...)
    - Convex optimization (QP, ...)
    - Nonlinear programming (Newton's method, ...)
    - ...
    - Gradient-based (Newton's method, ...)
    - Gradient-free (Nelder–Mead, ...)
  - Stochastic
    - Heuristic (evolution strategy, ...)
    - Metaheuristic
      - Population-based (PSO, FA, ...)
      - Trajectory-based (SA, ...)
  - Stochastic learning
    - Supervised learning (regression, ANN, ...)
    - Unsupervised learning (clustering, ANN, ...)

Types of Machine Learning / Training

ML algorithms are split into 4 classes / types

- Supervised learning

  $\longrightarrow$ Training set with the correct responses (targets or labels - pairs (X, Y^t))

  $\longrightarrow$ The algorithm generalizes to provide correct responses for new inputs X'

- Unsupervised learning

  $\longrightarrow$ Correct labels / responses are not available

- Semi-supervised learning

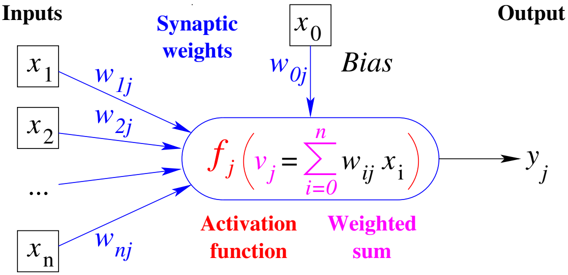
  $\longrightarrow$ Partially labeled input data are available

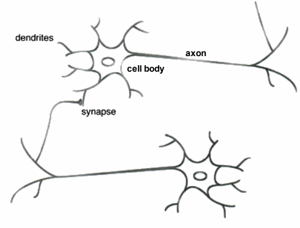
- Reinforcement learning

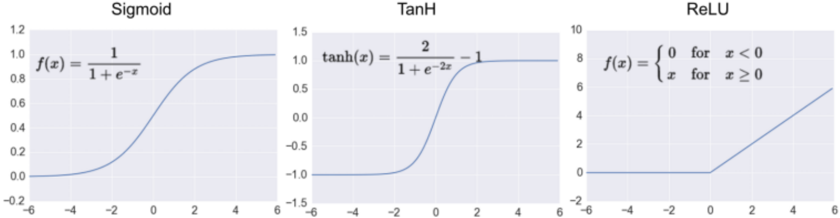
  $\longrightarrow$ The algorithm receives a measure of the quality of an action

  $\longrightarrow$ It explores and tries out different possibilities to find out how to correctly perform a given task (learning from trials and errors)

ANN - Basic component: the neuron

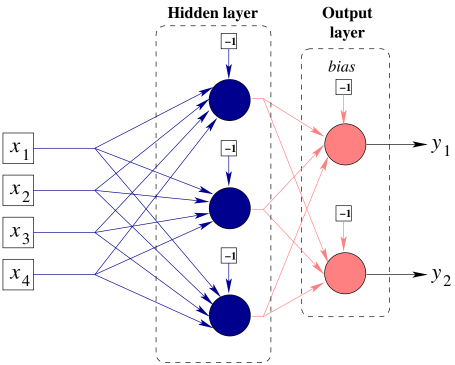






ANN - Multilayer Perceptron

- Many neurons organized in layers
- Classification (examples with Playground TensorFlow); regression

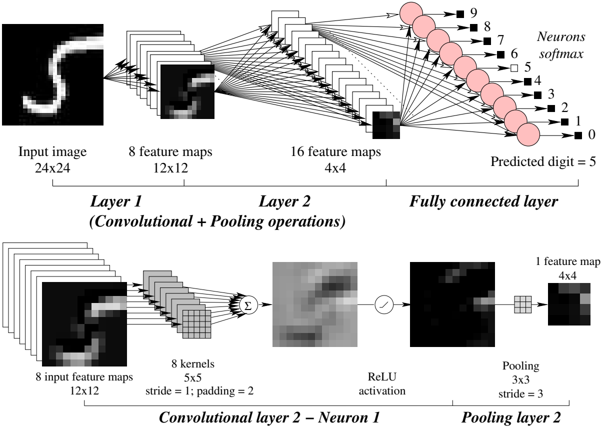


Supervised training of the weights using known data

$(X, Y^t) = ((x_1, x_2, x_3, x_4), (y_1^t, y_2^t))$ in order to minimize $\| Y - Y^t \|^2$

ANN - Deep learning (Apprentissage profond)

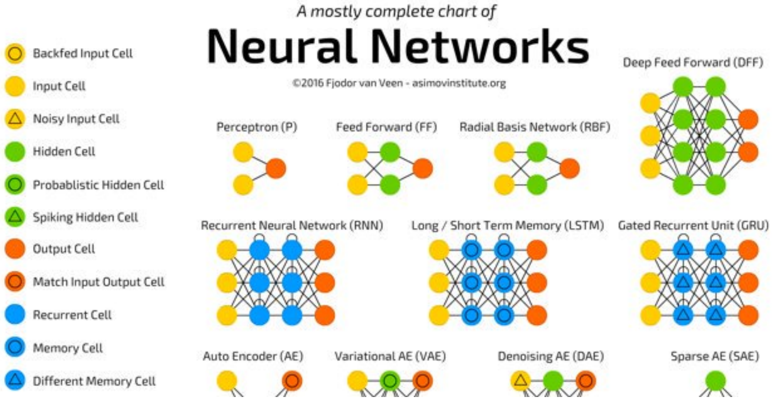
- "Many" layers, other kinds of neurons, ...
- Flagship architecture: Convolutional Neural Networks



MNIST problem - Synaptic weights = convolution kernels

ANN - Various architectures are available

- The neural network “zoo” is becoming larger and larger!
  - Feedforward (or static) ANN → *feedforward connections*
  - Recurrent (or dynamic) ANN → *feedback connections*



ANN - *Feedforward* vs. *Recurrent Networks*

Feedforward

- Connected graph without cycles
- One or several function(s)
- Universal function approximator
- More "easy" to train
- Widely used

Recurrent

- Connected graph with cycles
- A dynamical system
- "Has" some memory
- Universal approximator of dynamical systems
- More "difficult" to train
- Used to deal with "temporal" data

What topology for a neural network

- Data changing over time $\rightarrow$ recurrent network, but...
- Number of inputs and outputs $\rightarrow$ defined by the problem
- Number of layers, neurons, etc. $\rightarrow$ no methodology...

ANN - Training process - 1/2

What do we expect from an ANN?

Be able to learn from “stimulations” provided by its environment

What does the training process?

- It modifies the adjustable parameters of the network
  $\rightarrow$ synaptic weights and biases of the neurons
- using rules to update them
  $\rightarrow$ iterative process

What does mean "stimulation" or training an ANN?

To use input data to guide the update of the ANN parameters

What is supervised learning?

Optimization of the synaptic weights and biases
to minimize the output error (MSE = $\| Y - Y^t \|^2$)

ANN - Training process - 2/2

Training an ANN means solving an optimization problem usually nonlinear

- Any optimization method can be used
- In practice a gradient descent optimization method
  - Local optimization method
    $\rightarrow$ converge only to a local minimum
  - How are computed the gradients for weights and biases?
    $\rightarrow$ thanks to the backpropagation algorithm
  - Gradient descent variants
    $\rightarrow$ *Batch gradient descent*
    $\rightarrow$ *Stochastic Gradient Descent* (SGD)
    $\rightarrow$ *Mini-batch gradient descent*
  - Gradient descent optimization algorithms
    $\rightarrow$ *Momentum*, *Adagrad*, etc.

ANN - What is a good neural network?

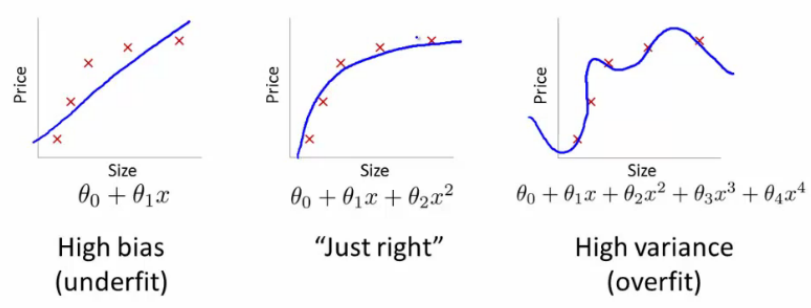
A neural network able to generalize
(gives good predictions for new encountered data)

- Some keys for a good training
  - Enough computing power (Datacenter, GPU, etc.)
  - A large and representative data set
  - Avoid overfitting (overadjustment of the network parameters)
    - Loss of the generalization ability (“memorizes”)
- How to avoid overfitting?
  $\rightarrow$ by dividing the data set in
  - *training / learning set* (60%; 70%; 70%)
  - *validation set* (20%; 15%; 0%)
  - *test set* (20%; 15%; 30%)

  $\rightarrow$ regularization; deactivation of neurons with dropout; etc.

ANN - Overfitting control

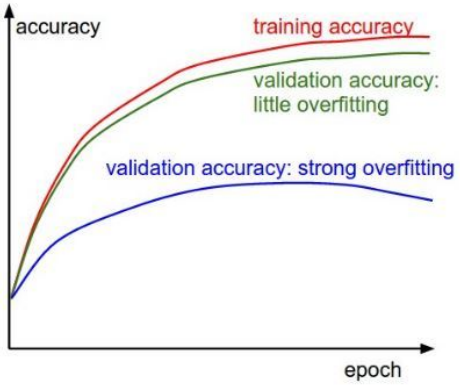
- Example (bias-variance tradeoff)



- High bias $\rightarrow$ ANN underfitted
- High variance $\rightarrow$ ANN sensitive to changes

ANN - Overfitting control

- Control using data → *early stopping*



- The *validation set* controls the accuracy of the predictions

Optimization - Terminology and methods

Definition of optimization problems

- Given a set Ω of
  - configurations;
  - or possibles solutions

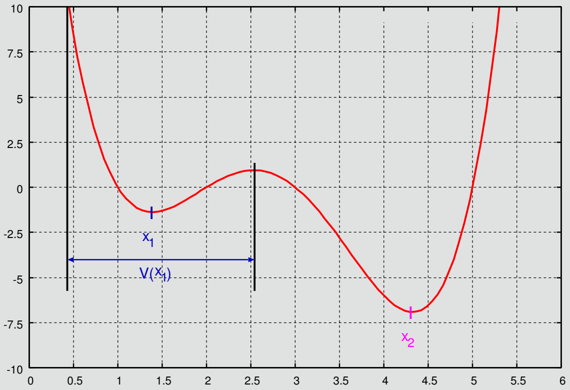
  of the problem to solve and an objective function f;
- find the minimum value x' of function f defined over Ω
- Which means x' such that $x' \in \Omega$ and $f(x') = \min_{x \in \Omega} f(x)$

Focus on some methods

- Local and deterministic optimization method
  - Gradient descent
- Global and stochastic optimization method
  - Simulated annealing

Optimization - Considered objective function

$f(x) = (x-1)\cdot(x-2)\cdot(x-3)\cdot(x-5)$ over $[0;6]$



- x_1 is a local minimum on neighborhood $V(x_1)$;
- x_2 is the global minimum

Gradient descent - One-dimensional objective function

- Allows to find a minimum if f is differentiable
- If the function can be derived, a minimum x' satisfies:

$$f'(x') = \frac{\partial f}{\partial x}(x') = 0$$

where the derivative $f'(x)$ is:

$$f'(x) = \frac{\partial f}{\partial x}(x) = 4 \cdot x^3 - 33 \cdot x^2 + 82 \cdot x - 61$$

- Directly solving $f'(x) = 0$ consists in finding polynomial roots where the polynomial is of degree 3 $\rightarrow$ difficult
- Gradient descent finds x' iteratively:
  - starting with an initial value x^0 (more or less well-chosen)
  - it builds a set of values x^k that converges towards a minimum of the objective function

Gradient descent - One-dimensional objective function

- Method based on the observation that considering a point a, function f will decrease in the direction opposite to f' value for a. Indeed:

$$f'(a) = \frac{\partial f}{\partial x}(a) = \lim_{h \to 0} \frac{f(a+h) - f(a)}{h}$$

thus:

  - if $f'(a) > 0 \Rightarrow f(a+h) > f(a)$
    $\rightarrow$ f decreases towards the x-axis with negative values
  - if $f'(a) < 0 \Rightarrow f(a+h) < f(a)$
    $\rightarrow$ f decreases towards the x-axis with positive values

- Finally, defining x^{k+1} as follows:

$$x^{k+1} = x^k - \gamma \cdot f'\left(x^k\right) = x^k - \gamma \cdot \frac{\partial f}{\partial x}\left(x^k\right) \qquad (1)$$

for $\gamma > 0$ small enough we have $f\left(x^{k+1}\right) \leq f\left(x^k\right)$

Gradient descent

Remarks on γ (often chosen between 0 and 1)

- It is called the learning rate;
- its value has an impact on the rate (speed) of convergence ;
- it also affects the minimum which will be found more or less close to the exact minimum;
- if it is set to large, the methode can have a chaotic behavior and even diverge;
- γ can decrease during optimization (rate decay)

Detection of convergence

- Stop computations when two successives x are close enough to each other $\left|x^{k+1} - x^k\right| \leq \epsilon$
- Stop when a maximum number of iterations is reached

Gradient descent - High-dimensional objective function

- Gradient descent finds a minimum x' iteratively:
  - starting with an initial value x^0 (more or less well-chosen)
  - it builds a set of values x^k that converges towards a minimum of the objective function
- The formula (1) defining x^{k+1} becomes:

$$x^{k+1} = x^k - \gamma \cdot \nabla f\left(x^k\right)$$

where ∇f is the gradient of f (that explains the name of the method)

- As

$$\nabla f(x) = \left(\frac{\partial f}{\partial x_1}(x), \ldots, \frac{\partial f}{\partial x_n}(x)\right)^T$$

we have the following update rule for i-th component of x^k

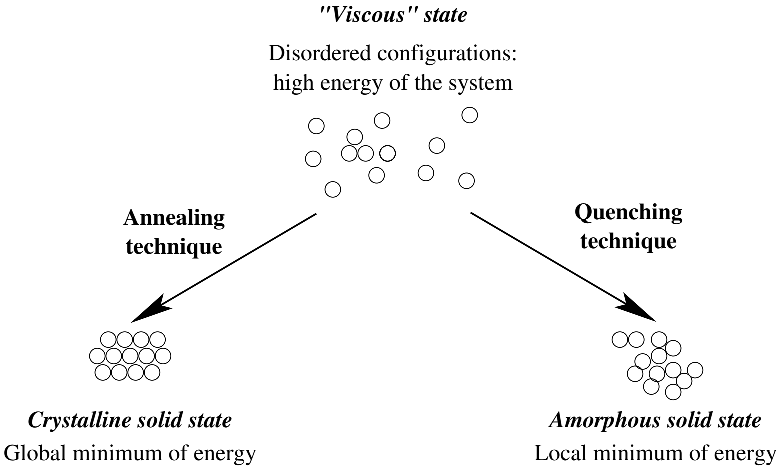
$$x_i^{k+1} = x_i^k - \gamma \cdot \frac{\partial f}{\partial x_i}\left(x^k\right)$$

Simulated annealing - Inspiration and principle

- Based on an analogy with annealing in metallurgy
  - a technique involving heating
  - and controlled cooling to increase the size of its crystals
- Principle
  - The material is first heated
  - Then it is slowly cooled (x denotes a configuration):
    - at high temperatures the atoms are agitated
      $\rightarrow$ all atomic configurations have same probability;
    - at low temperatures the atoms become ordered
      $\rightarrow$ reduce defects in crystal structure; lower energy states
  - Constraint
    - slow cooling $\rightarrow$ not being trapped by a local minimum

Simulated annealing - Inspiration and principle

- Inspiration (*Simulated Annealing*)



Simulated annealing - Probabilitic sampling

Gibbs / Boltzmann distribution

$$P(x) = \frac{1}{Z_T} \cdot \exp\left(-\frac{E(x)}{K_B \cdot T}\right)$$

where:

- $x \in \Omega$ (a configuration of physical system);
- E an energy function defined over Ω ($E(x) = f(x)$);
- T is the temperature

Metropolis Algorithm (1953) - fixed temperature T

- Sequence of configurations obtained through “small” changes

$$P(y|x) = \min\left(1, \exp\left(-\frac{E(y) - E(x)}{T}\right)\right)$$

Simulated annealing - Algorithm - 1/3

- Based on two steps
  - Distribution sampling process
    - Exploration (generate from x, a new candidate solution y)
      $\rightarrow$ Use a neighborhood ($y \in V(x)$)
    - Acceptance probability function
      $\rightarrow$ Use Metropolis algorithm ($P(y|x)$)
  - Cooling process
    - Use some annealing schedule to reduce the temperature

Simulated annealing - Algorithm - 2/3

- Ability to avoir local minimums
  - At begining high acceptance proba. for y s.t. $E(y) > E(x)$
  - The higher T, the higher the probability
- Choice of parameters (convergence in finite time)
  - T^0 such that almost all new configurations are accepted
  - Each temperature level (the number of state transitions) must be long enough
  - Slow temperature decrease between two levels
- Choice of parameters (in practice)
  - T^0 and temperature level length chosen after experiments
  - Exponential cooling schedule

$$T^k = T^0 \times \alpha^k, k \in \mathbb{N} \text{ et } 0 < \alpha < 1$$

  - Possible stopping criteria
    - percentage of accepted config. below a fixed value;
    - low variance of energy (objective function) values;
    - minimum temperature value

Simulated annealing - Algorithm - 2/3

Description of Metropolis version

$k = 0$
$T^k = T^0$ // Starting temperature
$x^k = x^0$ // Starting configuration
repeat
    repeat
        Draw randomly $y \in V(x^k)$ // A neighboring configuration of x^k
        if $\Delta E = \left(E(y) - E(x^k)\right) < 0$ or $\exp\left(-\frac{\Delta E}{T^k}\right) > \mu$, $\mu \in [0; 1]$ drawn
        from an uniform distribution **then**
            $x^{k+1} = y$
        else
            $x^{k+1} = x^k$
        end if
    until end of temperature level
    $T^{k+1} = g\left(T^k\right)$ // g is strictly decreasing
    $k = k + 1$
until stopping criterion satisfied