METAHEURISTICS

INF273

#6: Simulated Annealing

AHMAD HEMMATI

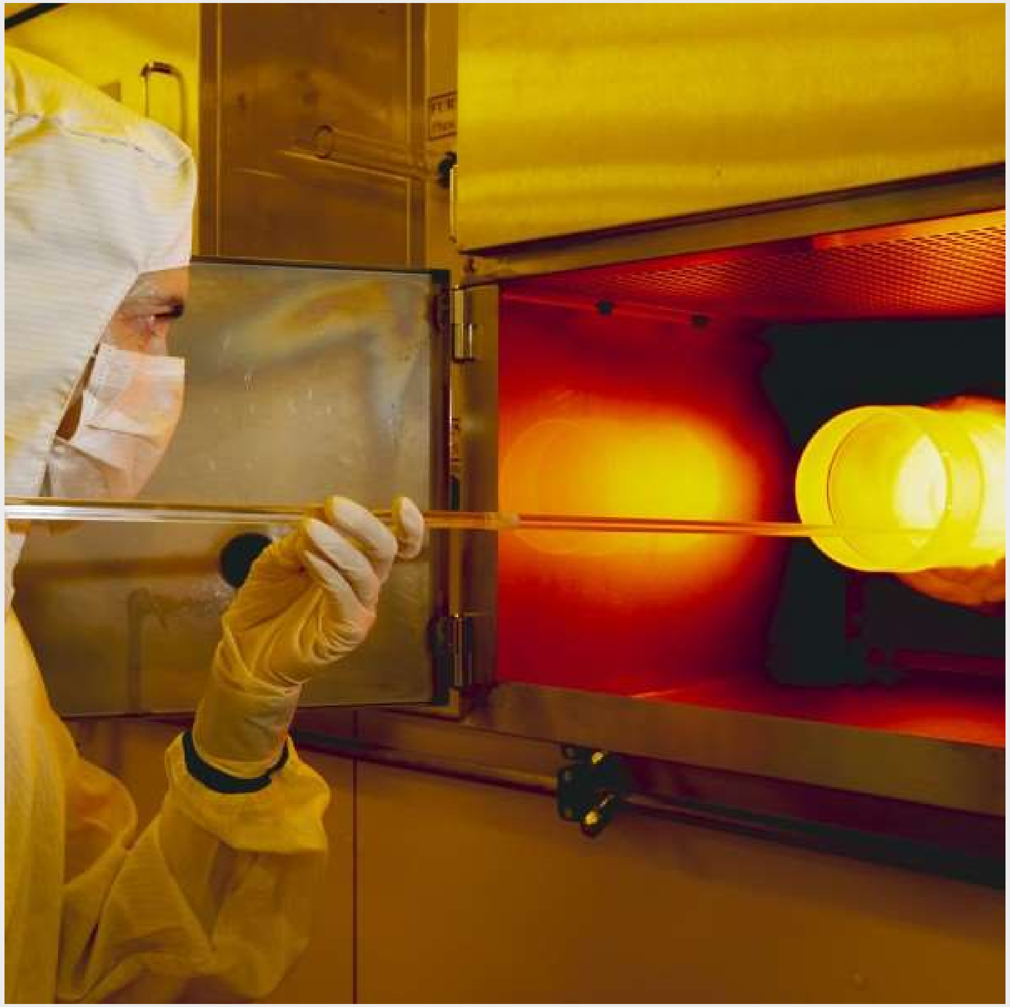
Optimization Group
Dept. of Informatics
University of Bergen

Spring Semester
2022

AGENDA

- Introduction: Annealing
- Simulated Annealing
  - Temperature
  - Cooling Schedule
  - Acceptance Criteria

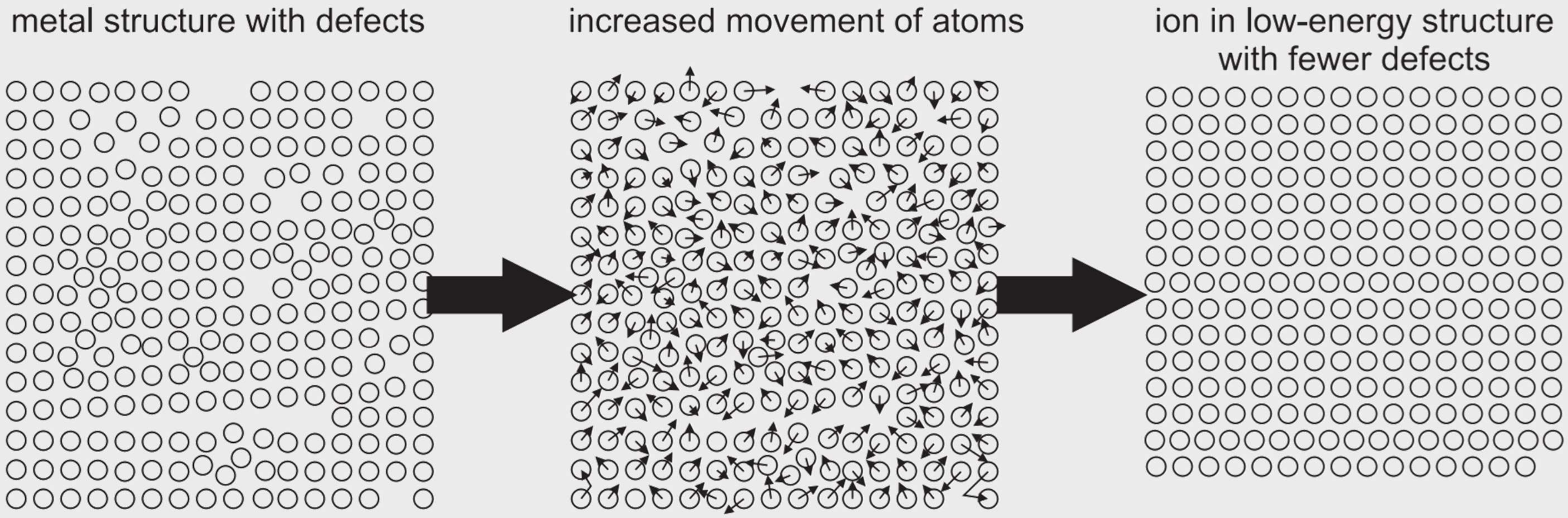
Introduction: Annealing



Introduction: Annealing

- Cold working metal causes/increases defects in crystal structure
- After cold working, *annealing* is performed
- The metal is heated to, like 0.4 * melting temperature
- Ions inside metal can move around
- Metal is slowly cooled down, ions assume low-energy, stable positions in crystal → metal becomes more stable
- Due to their movement, ions may temporarily assume positions of high energy
- An initial, brittle crystal structure is transformed to a much better configuration by stepping over good and bad states

Introduction: Annealing



Simulated Annealing

- Metropolis' algorithm (1953)
  - Algorithm to simulate energy changes in physical systems when cooling and shows how physical systems find states of low energy
  - The (simulated) physical system escapes local optima by accepting worse solutions from time to time.
  - This could be a remedy for the premature convergence problem of local search/hill climbing!

Simulated Annealing

- Kirkpatrick, Gelatt and Vecchi (1983)
  - Suggested to use the same type of simulation to look for good solutions in a COP

- Simulated Annealing = hill climbing + sometimes accept worse states following Metropolis' method

  ⇒ lower risk of premature convergence

Simulated Annealing

- A lot of literature
- Can be interpreted as a Stochastic Local Search
  - Choose a neighbor solution
  - Improving moves are always accepted
  - Deteriorating moves are accepted with a probability that depends on the amount of the deterioration and on the temperature (a parameter that decreases with time)
- Can escape local optima

SIMULATED ANNEALING

Initialization

Parameters Setting (T_0 , T_f, Max_Ite, Cooling_Schedule)

Generate an *Initial_solution* (randomly)

Current_solution =Initial_solution; Best_solution =Initial_solution; $T = T_0$

While ($T > T_f$) do

For i = 1 to *Max_Ite* do

Generate *New_solution* = Neighbor(*Current_solution*)

Evaluate the change in objective function level ΔE

If $\Delta E < 0$ then

Current_solution = *New_solution*

If f (*Current_solution*) < f (*Best_solution*) **then** *Best_solution* = *Current_solution*

Else

Accept *Current_solution* = *New_solution* with probability $p = e^{\frac{-\Delta E}{T}}$

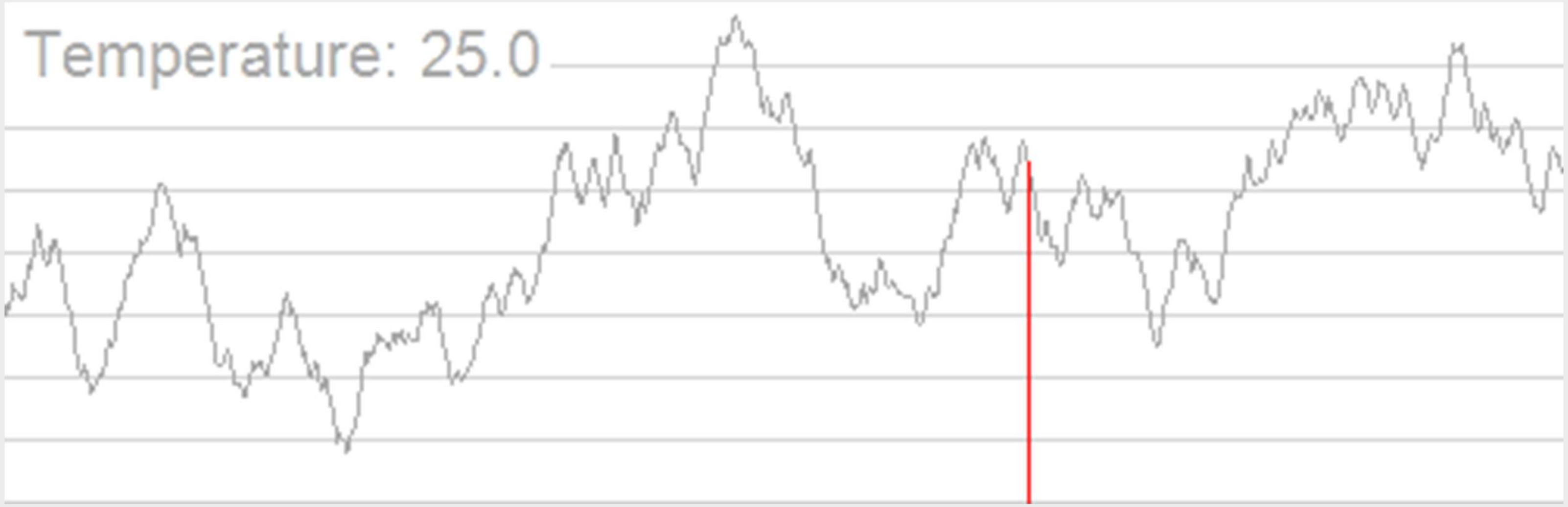
End If

End For

Decrease parameter T according to *Cooling_Schedule*

End While

Simulated annealing searching for a maximum



Acceptance Criteria

- We assume a minimization problem
- Set ΔE = Obj(neighbour solution) – Obj(current solution)
- If ΔE < 0 $\Longrightarrow$ accept (we have an improving move)
- Else accept if
- Boltzmann probability function: $p = e^{\frac{-\Delta E}{T}}$

$$Random(0,1) < e^{\frac{-\Delta E}{T}}$$

- If the move is not accepted: try another neighbor

Example 1:

Probability of acceptance for two different worse solutions (compared with current solution) at the same temperature:

$$(x_c, f(x_c) = 2000) \rightarrow (x_1, f(x_1) = 2100) \Longrightarrow p_1 = e^{\frac{-\Delta E}{T}} = e^{\frac{-(2100-2000)}{100}} = e^{-1} \cong 0.368$$

$$(x_c, f(x_c) = 2000) \rightarrow (x_2, f(x_2) = 2200) \Longrightarrow p_2 = e^{\frac{-\Delta E}{T}} = e^{\frac{-(2200-2000)}{100}} = e^{-2} \cong 0.135$$

$f(x_i)$	2050	2100	2200	2300	2400	2500	2600
p_i	0.606	0.368	0.135	0.050	0.018	0.007	0.002

Example 2:

Probability of acceptance for a worse solution (compared with current solution) at two different temperatures:

$T_1 = 100; \; (x_c, f(x_c) = 2000) \rightarrow (x_1, f(x_1) = 2200) \Longrightarrow p_1 = e^{\frac{-\Delta E}{T}} = e^{\frac{-(2200-2000)}{100}} = e^{-2} \cong 0.135$

$T_2 = 50; \quad (x_c, f(x_c) = 2000) \rightarrow (x_1, f(x_1) = 2200) \Longrightarrow p_2 = e^{\frac{-\Delta E}{T}} = e^{\frac{-(2200-2000)}{50}} = e^{-4} \cong 0.018$

T_i	300	200	150	100	75	50	40
p_i	0.513	0.368	0.263	0.135	0.069	0.018	0.007

Temperature

- *Initial* and *Final* temperatures (T_0, T_f) depend on the scale of the changes in objective function value (ΔE) when we move from one solution to anouther

$$p = e^{\frac{-\Delta E}{T}} \Longrightarrow \ln p = \frac{-\Delta E}{T} \Longrightarrow T = \frac{-\Delta E}{\ln p}$$

- Intermediate temperatures depend on the *cooling schedule*

Example 1:

$$p = e^{\frac{-\Delta E}{T}} \Longrightarrow \ln p = \frac{-\Delta E}{T} \Longrightarrow T = \frac{-\Delta E}{\ln p}$$

$p_{max} = 0.99$ $\qquad$ $p_{min} = 0.01$ $\qquad$ $\Delta E \in (0.01, 0.05)$

$$\Longrightarrow T_1 = \frac{-0.01}{\ln\ .99} \cong 0.99 \qquad T_2 = \frac{-0.05}{\ln 0.99} \cong 4.97 \qquad T_3 = \frac{-0.01}{\ln 0.01} \cong 0.002 \qquad T_4 = \frac{-0.05}{\ln 0.01} \cong 0.01$$

$$\Longrightarrow T_0 = \max\{T_1 \cong 0.99, T_2 \cong 4.97, \ T_3 \cong 0.002, T_4 \cong 0.01\} + \varepsilon = 5$$

$$\Longrightarrow T_f = \min\{T_1 \cong 0.99, T_2 \cong 4.97, \ T_3 \cong 0.002, T_4 \cong 0.01\} - \varepsilon = 0.001$$

Example 2:

$$p = e^{\frac{-\Delta E}{T}} \Longrightarrow \ln p = \frac{-\Delta E}{T} \Longrightarrow T = \frac{-\Delta E}{\ln p}$$

$p_{max} = 0.99$ $\quad$ $p_{min} = 0.01$ $\quad$ $\Delta E \in (40{,}50)$

$$\Longrightarrow T_1 = \frac{-40}{\ln 0.99} \cong 3980 \qquad T_2 = \frac{-5}{\ln 0.99} \cong 4975 \qquad T_3 = \frac{-40}{\ln 0.01} \cong 8.8 \qquad T_4 = \frac{-50}{\ln 0.01} \cong 11$$

$$\Longrightarrow T_0 = \max\,\{T_1 \cong 3980\,, T_2 \cong 4975,\ \ T_3 \cong 8.8\,, T_4 \cong 11\} + \varepsilon = 5000$$

$$\Longrightarrow T_f = \min\,\{T_1 \cong 3980\,, T_2 \cong 4975,\ \ T_3 \cong 8.8\,, T_4 \cong 11\} - \varepsilon = 8$$

Cooling Schedule

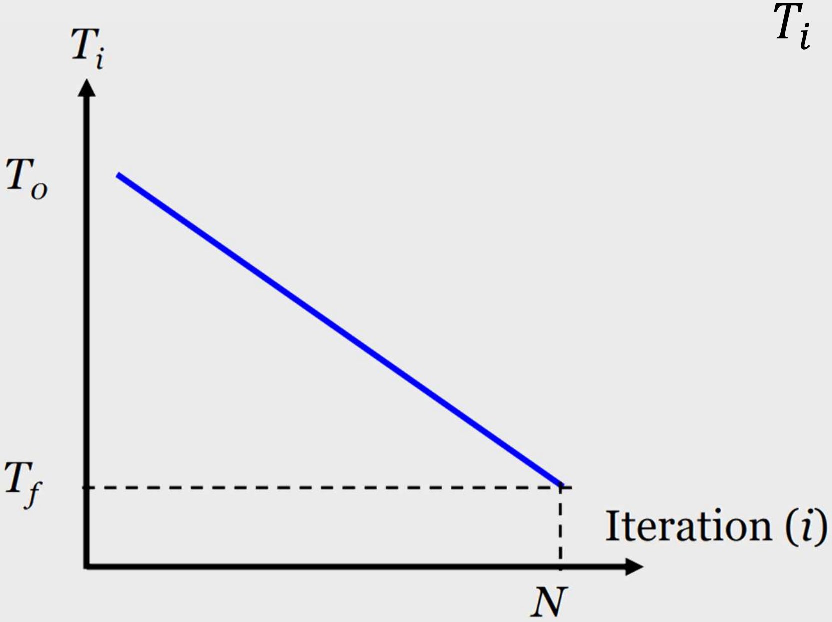
- The cooling schedule defines how the temperature parameter T in the *Simulated Annealing* process is set.

- The operator maps the current iteration index i to a (positive) real temperature value T.

- The temperature schedule allows for a smooth transition of *SA* algorithm behavior from "like *Random Search*" (high temperature) to "like *hill climbing*" (low temperature).

Cooling Schedule

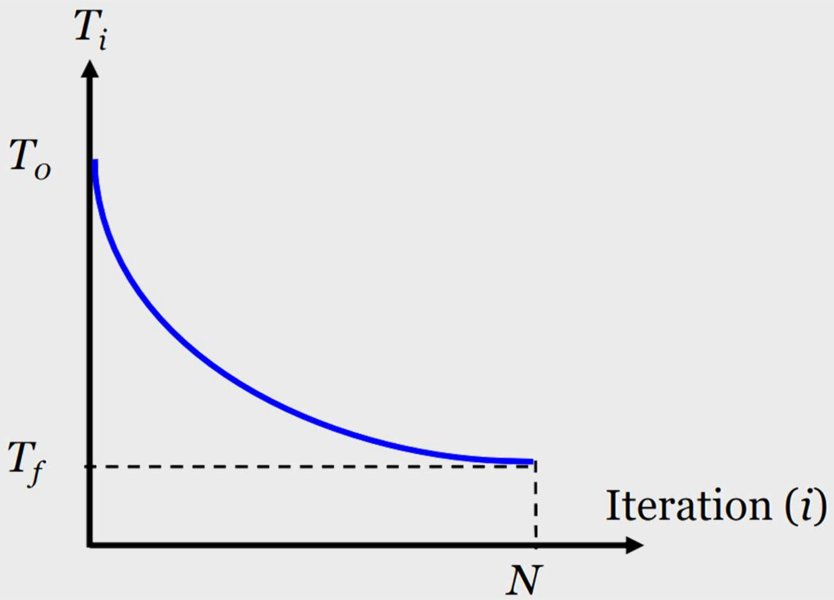
- Linear



$$T_i = T_0 - i\left(\frac{T_0 - T_f}{N}\right)$$

Cooling Schedule

- Exponential

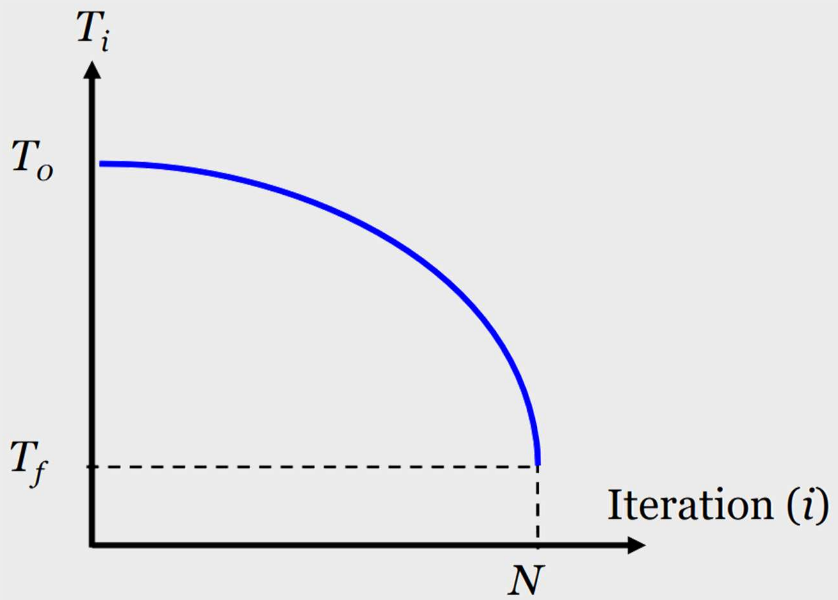


$$T_{i+1} = \alpha\, T_i$$

$$0 < \alpha < 1$$

- Logarithmic

$$T_i = T_0 - i^A$$

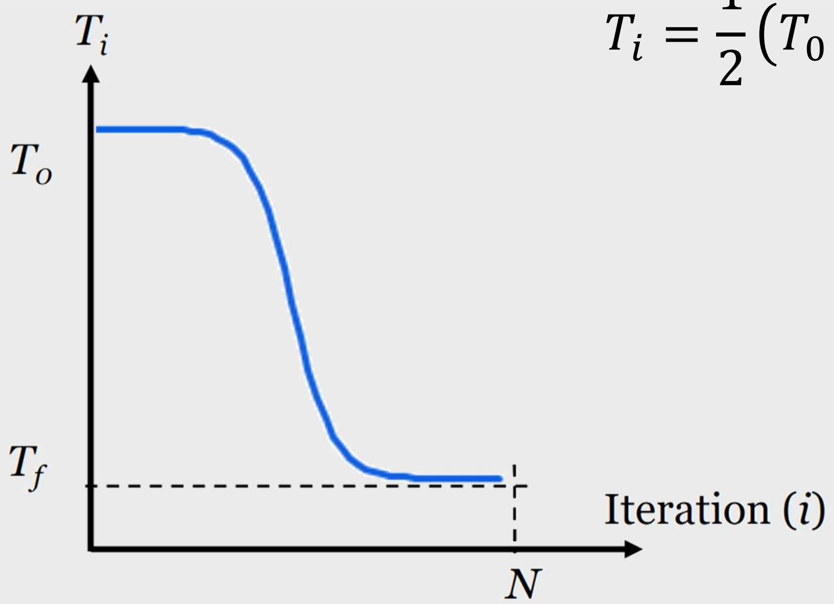


$$A = \frac{\ln(T_0 - T_f)}{\ln(N)}$$

Cooling Schedule

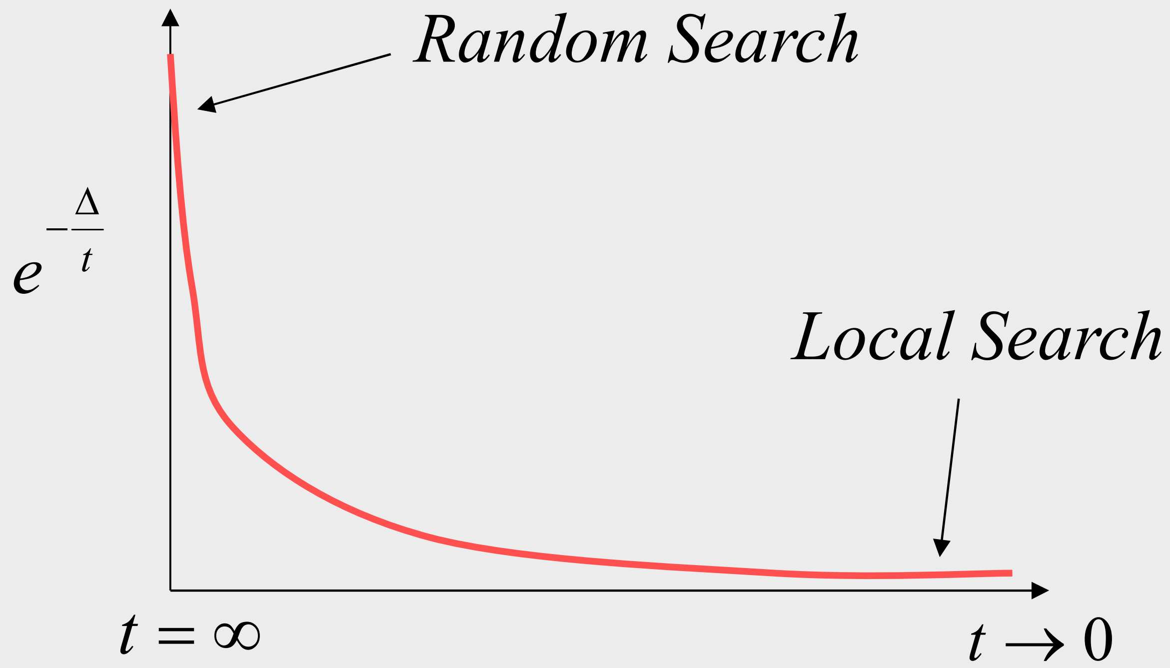
- Hyperbolic

$$T_i = \frac{1}{2}\left(T_0 - T_f\right)\left(1 - tanh\left(\frac{10\ i}{N} - 5\right)\right) + T_f$$



SIMULATED ANNEALING – REMARKS

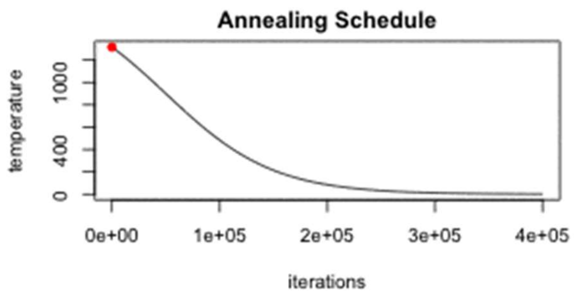
- *Initial temperature* T_0
  - (if $\infty \rightarrow$ *Random Search*)
  - (if $\sim 0 \rightarrow$ *Local Search*)



SIMULATED ANNEALING – REMARKS

- *Initial temperature* T_0
  - (if $\infty \rightarrow$ *Random Search*)
  - (if $\sim 0 \rightarrow$ *Local Search*)
- *Cooling schedule* – how to change temperature over time
  - If too fast $\rightarrow$ stops in some local optimum too early
  - If too slow $\rightarrow$ too slow convergence
- Number of iterations at each temperature
- Choice of neighborhood structure is important
- Stopping criterion
- Solution quality/speed dependents on the choices made


Distance: 43,499 miles
Temperature: 1,316
Iterations: 0


Annealing Schedule
temperature
0
400
1000
0e+00
1e+05
2e+05
3e+05
4e+05
iterations

Lecture #7:

Tabu Search