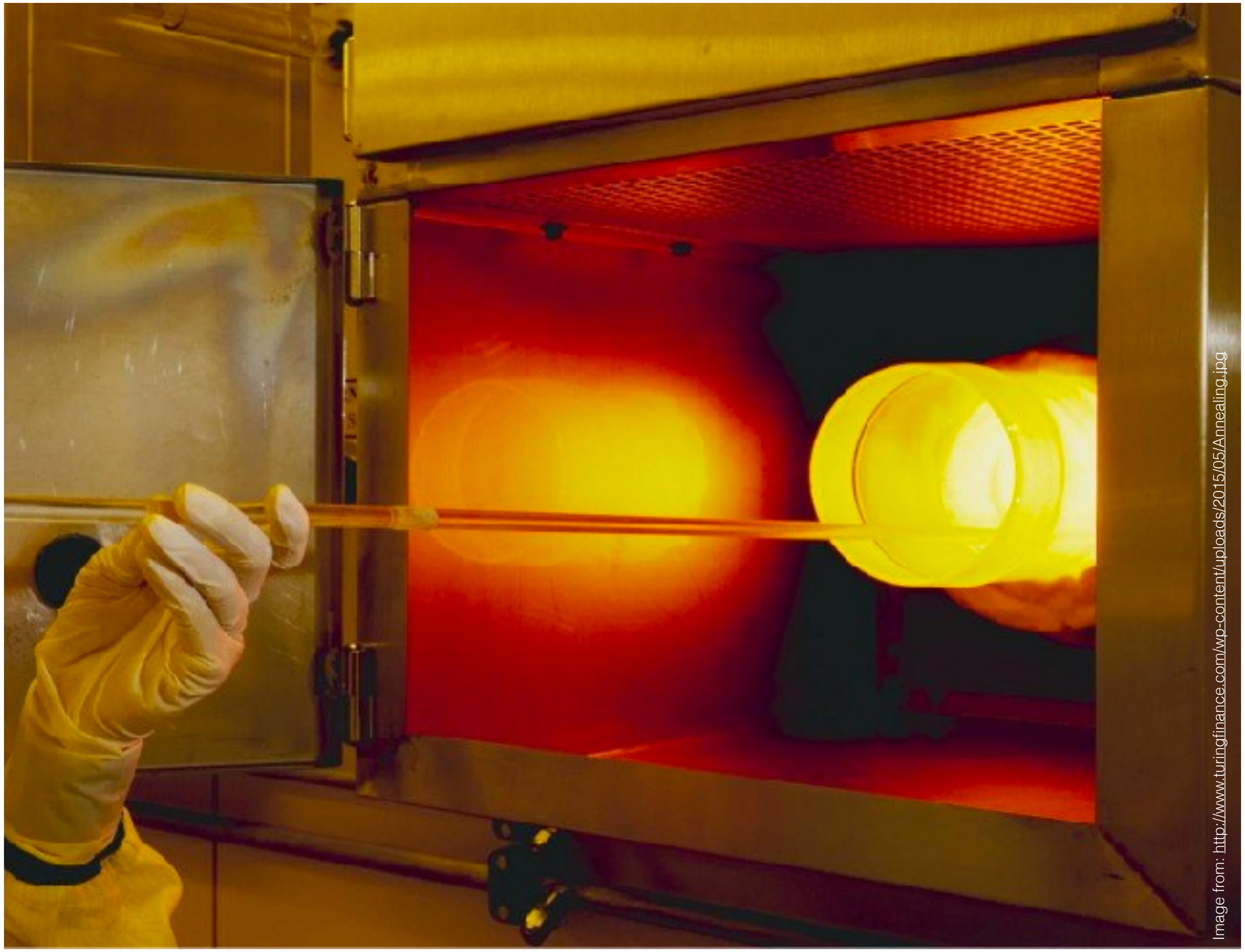
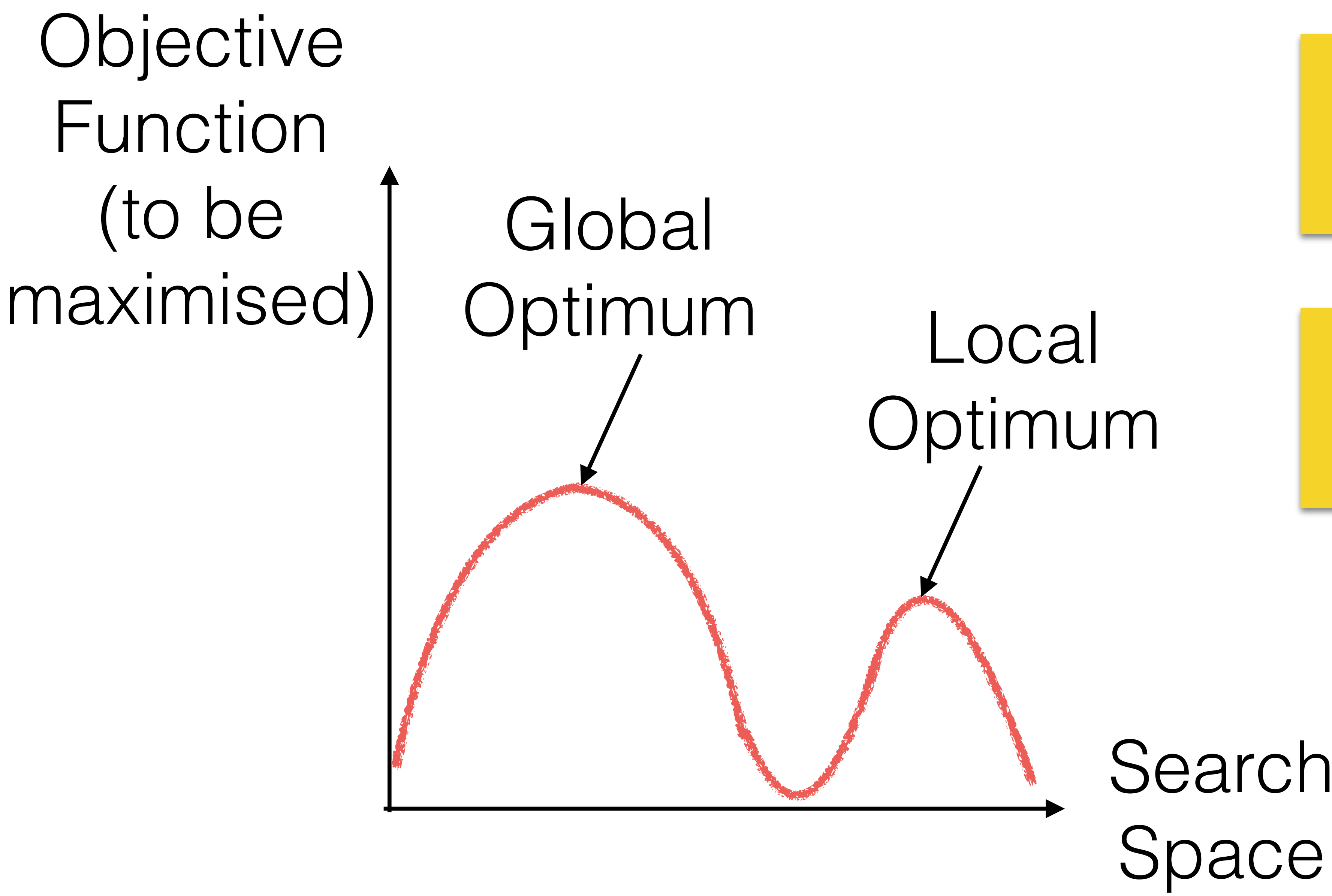


Image from: http://www.turingfinance.com/wp-content/uploads/2015/05/Annealing.jpg

Simulated Annealing - Part 1

Leandro L. Minku

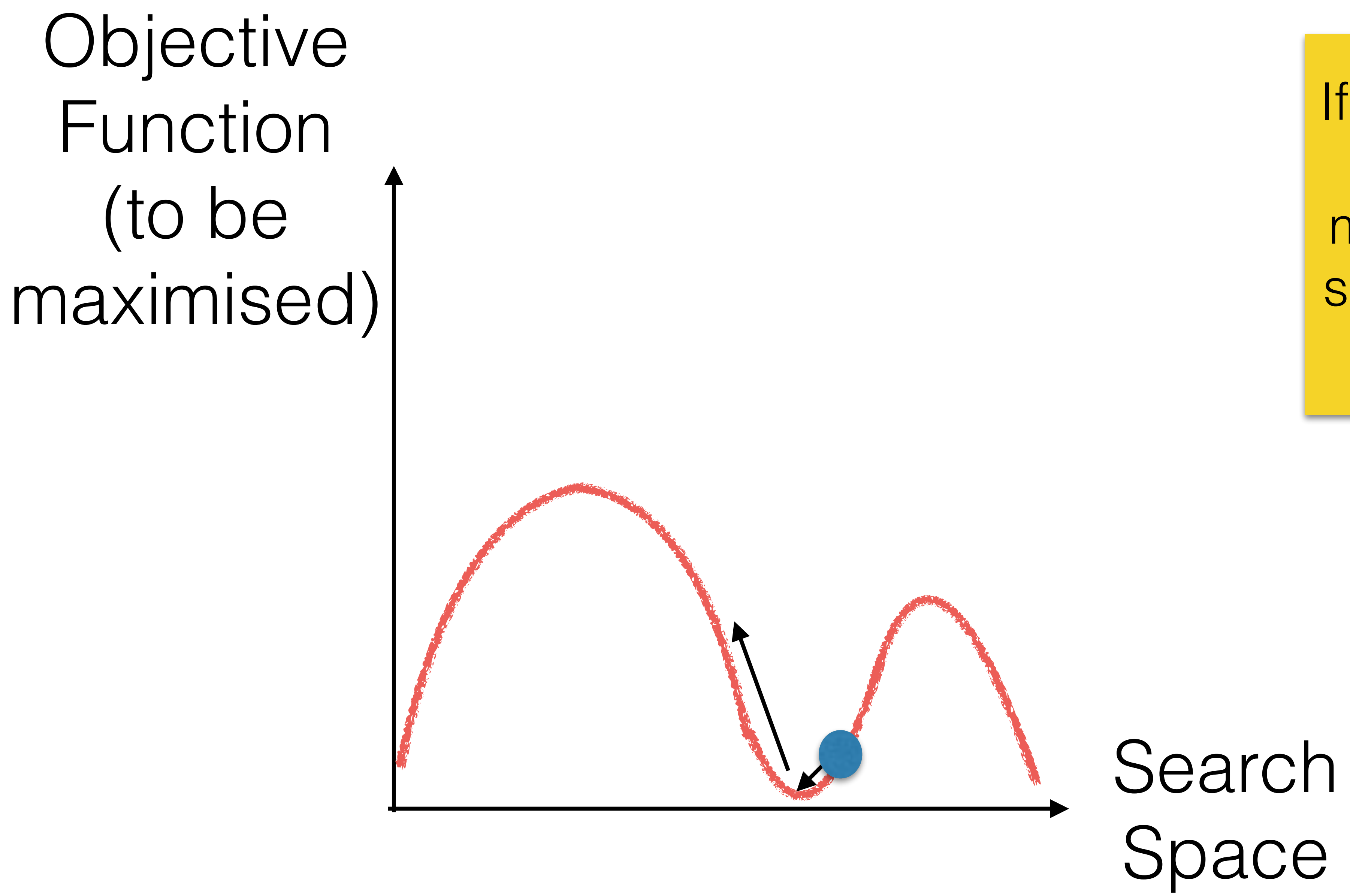
Motivation



Hill-climbing may get trapped in a local optimum.

Heuristic = informed guess

Motivation



If we could sometimes accept a downward move, we would have some chance to move to another hill.

Hill-Climbing

Hill-Climbing (assuming maximisation)

1. current_solution = generate initial solution randomly

2. Repeat:

    2.1 generate neighbour solutions (differ from current solution by a single element)

    2.2 best_neighbour = get highest quality neighbour of current_solution

    2.3 If quality(best_neighbour) <= quality(current_solution)

        2.3.1 Return current_solution

    2.4 current_solution = best_neighbour

    Until a maximum number of iterations

In simulated annealing, instead of taking the best neighbour, we pick a random neighbour.

Hill-Climbing

Hill-Climbing (assuming maximisation)

1. current_solution = generate initial solution randomly

2. Repeat:

   2.1 generate neighbour solutions (differ from current solution by a single element)

   2.2 best_neighbour = get highest quality neighbour of current_solution

   2.3 If quality(best_neighbour) <= quality(current_solution)

      2.3.1 Return current_solution

   2.4 current_solution = best_neighbour

   Until a maximum number of iterations

Simulated annealing will give some chance to accept a bad neighbour.

Simulated Annealing

Simulated Annealing (assuming maximisation)

1. current_solution = generate initial solution randomly

2. Repeat:

   2.1 generate neighbour solutions (differ from current solution by a single element)

   2.2 rand_neighbour = get random neighbour of current_solution

   2.3 If quality(rand_neighbour) <= quality(current_solution) {

      2.3.1 With some probability,
      current_solution = rand_neighbour

   } Else current_solution = rand_neighbour

   Until a maximum number of iterations

Simulated Annealing

Simulated Annealing (assuming maximisation)

1. current_solution = generate initial solution randomly

2. Repeat:

   ~~2.1 generate neighbour solutions (differ from current solution by a single element)~~

   2.2 rand_neighbour = get random neighbour of current_solution

   2.3 If quality(rand_neighbour) <= quality(current_solution) {

      2.3.1 With some probability,
      current_solution = rand_neighbour

   } Else current_solution = rand_neighbour

   Until a maximum number of iterations

Simulated Annealing

Simulated Annealing (assuming maximisation)

1. current_solution = generate initial solution randomly

2. Repeat:

    2.1 rand_neighbour = generate random neighbour of current_solution

    2.2 If quality(rand_neighbour) <= quality(current_solution) {

        2.2.1 With some probability,
            current_solution = rand_neighbour

    } Else current_solution = rand_neighbour

   Until a maximum number of iterations

Simulated Annealing

Simulated Annealing (assuming maximisation)

1. current_solution = generate initial solution randomly

2. Repeat:

    2.1 rand_neighbour = generate random neighbour of current_solution

    2.2 If quality(rand_neighbour) <= quality(current_solution) {

        2.2.1 With some probability,
            current_solution = rand_neighbour

    } Else current_solution = rand_neighbour

  Until a maximum number of iterations

How Should the Probability be Set?

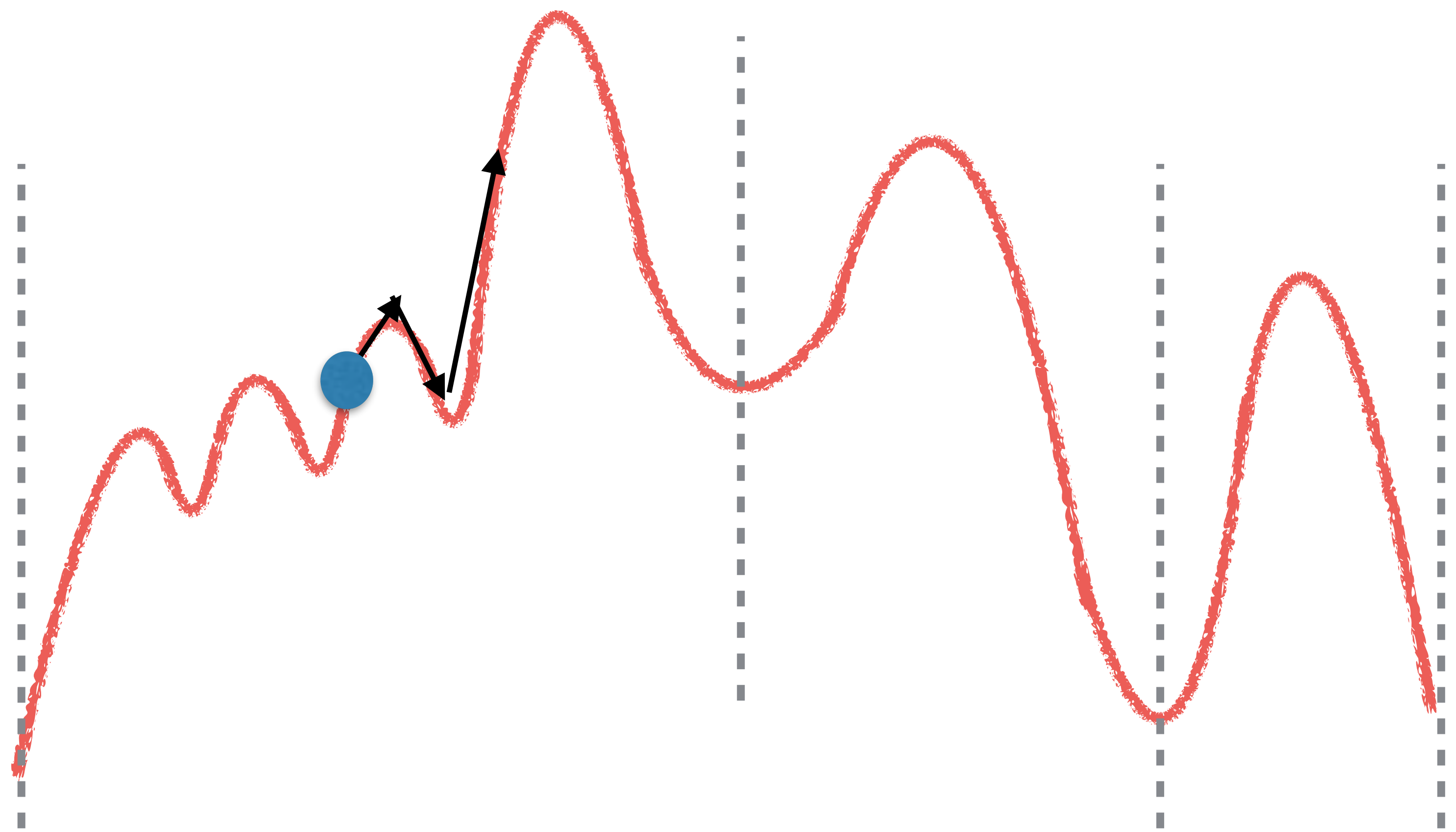
- Probability to accept solutions with much worse quality should be lower.
  - We don’t want to be dislodged from the optimum.

- High probability in the beginning.
  - More similar effect to random search.
  - Allows us to explore the search space.

- Lower probability as time goes by.
  - More similar effect to hill-climbing.
  - Allows us to exploit a hill.

How to Decrease the Probability?

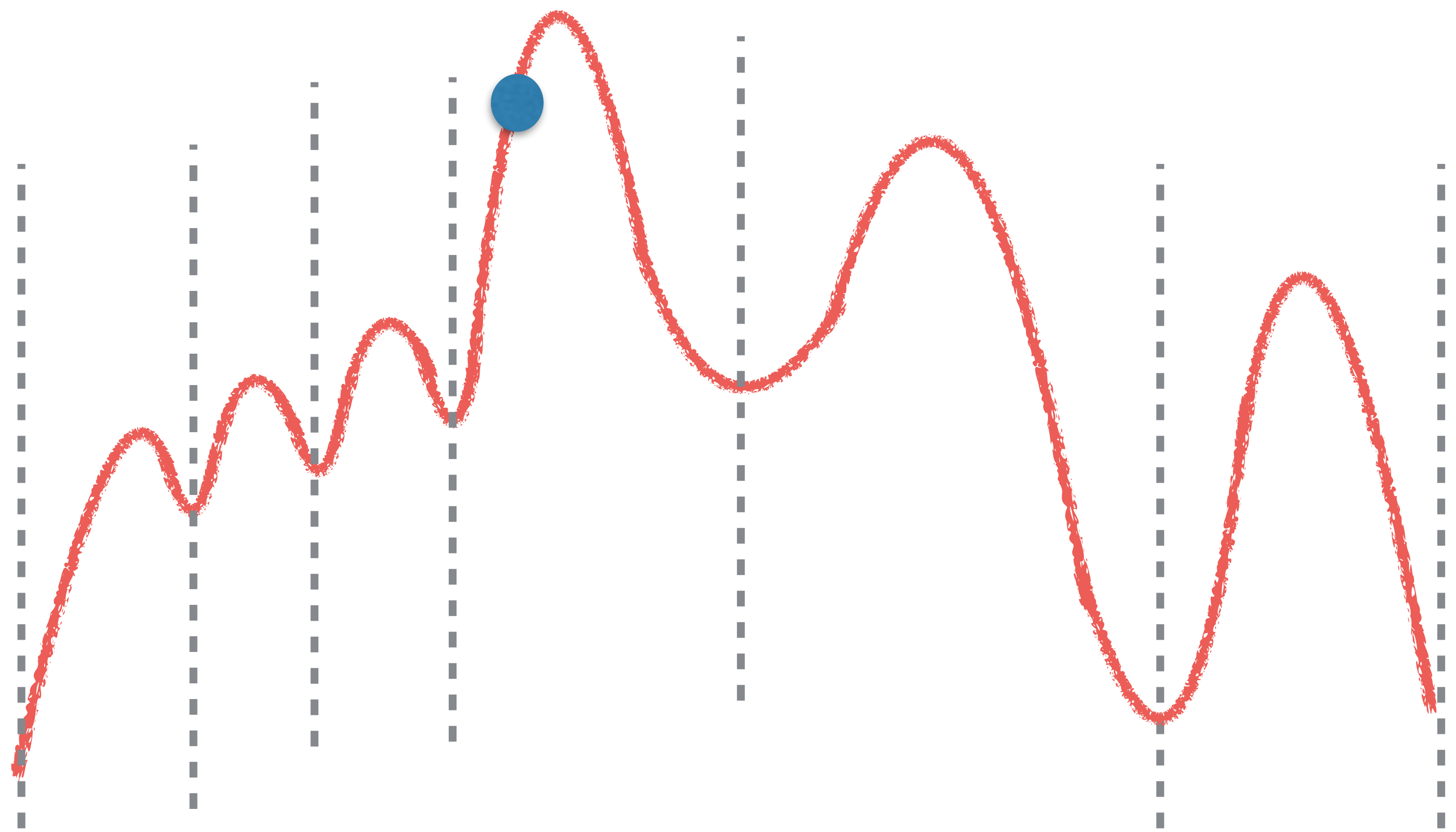
- We would like to decrease the probability slowly.



If you decrease the probability slowly, you start to form basis of attraction, but you can still walk over small hills initially.

How to Decrease the Probability?

- We would like to decrease the probability slowly.



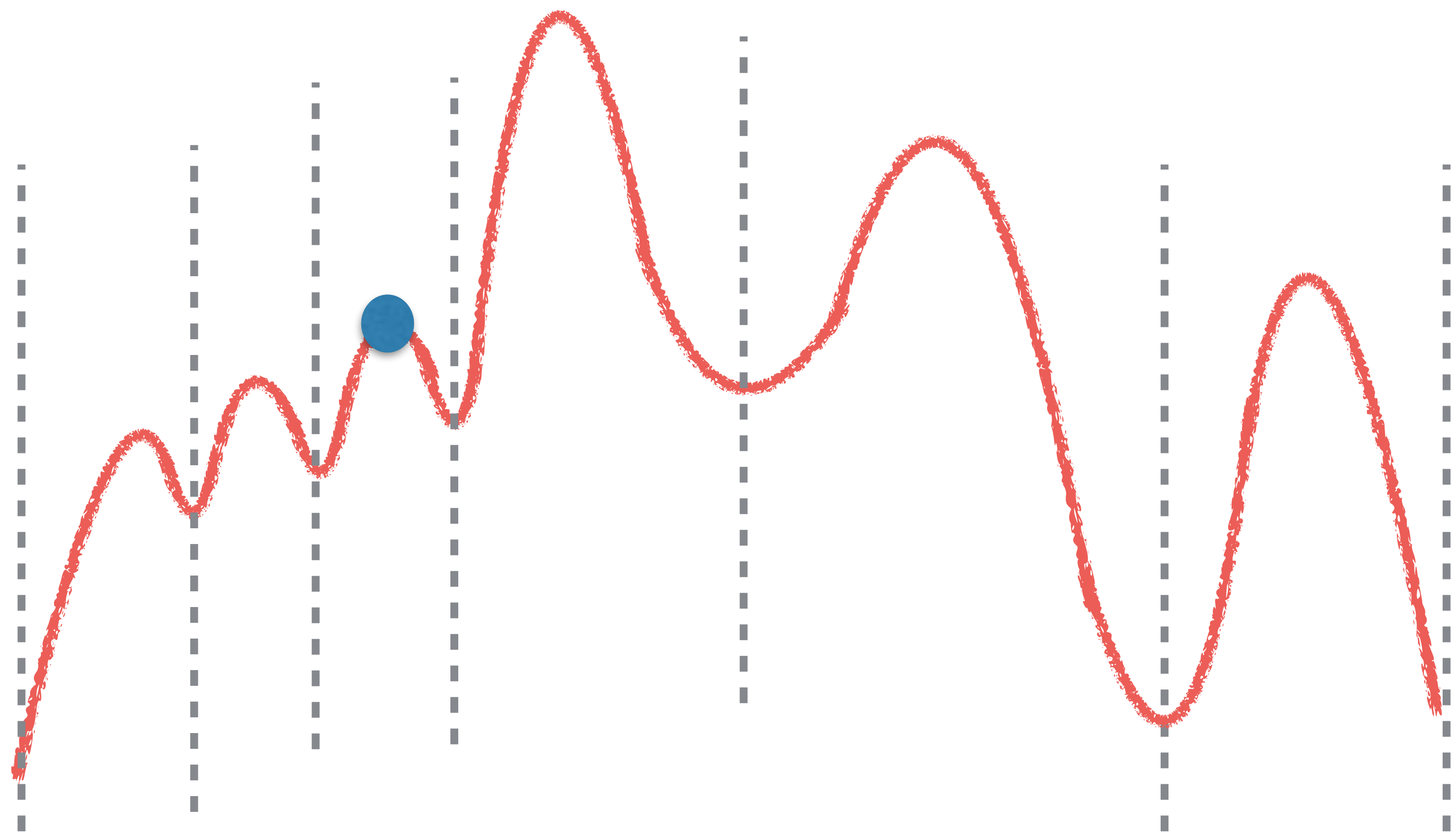
As the probability decreases further, the small hills start to form basis of attraction too.

But if you do so slowly enough, you give time to wander to the higher value hills before starting to exploit.

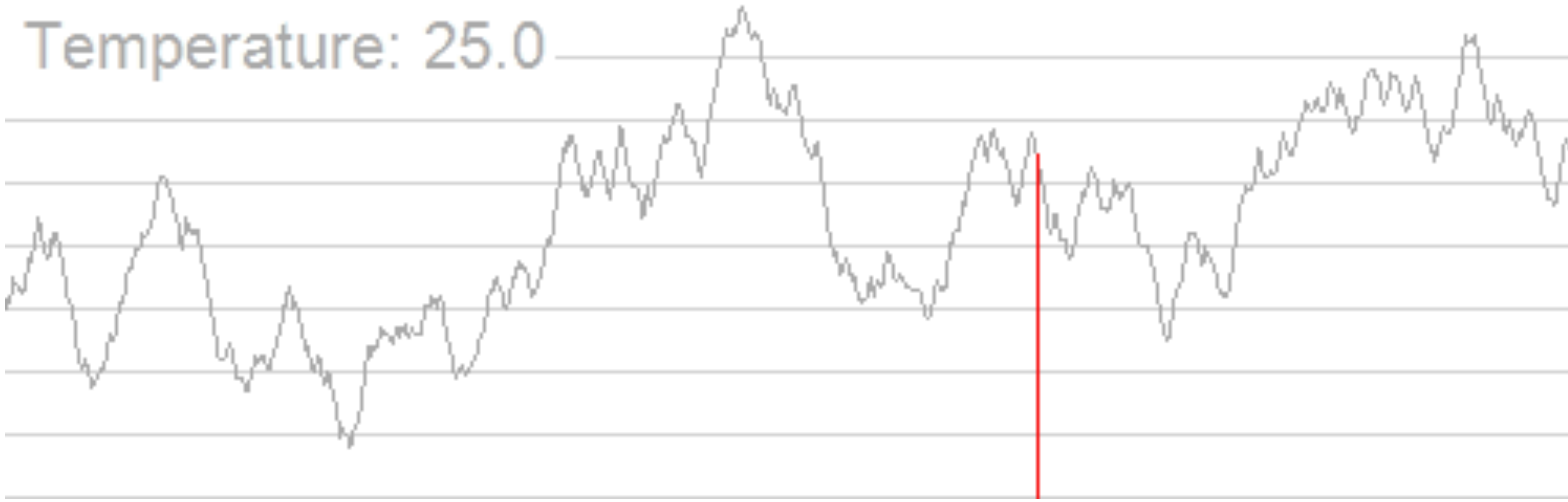
So, you can find the global optimum!

How to Decrease the Probability?

- We would like to decrease the probability slowly.



If you decrease too quickly, you can get trapped in local optima.

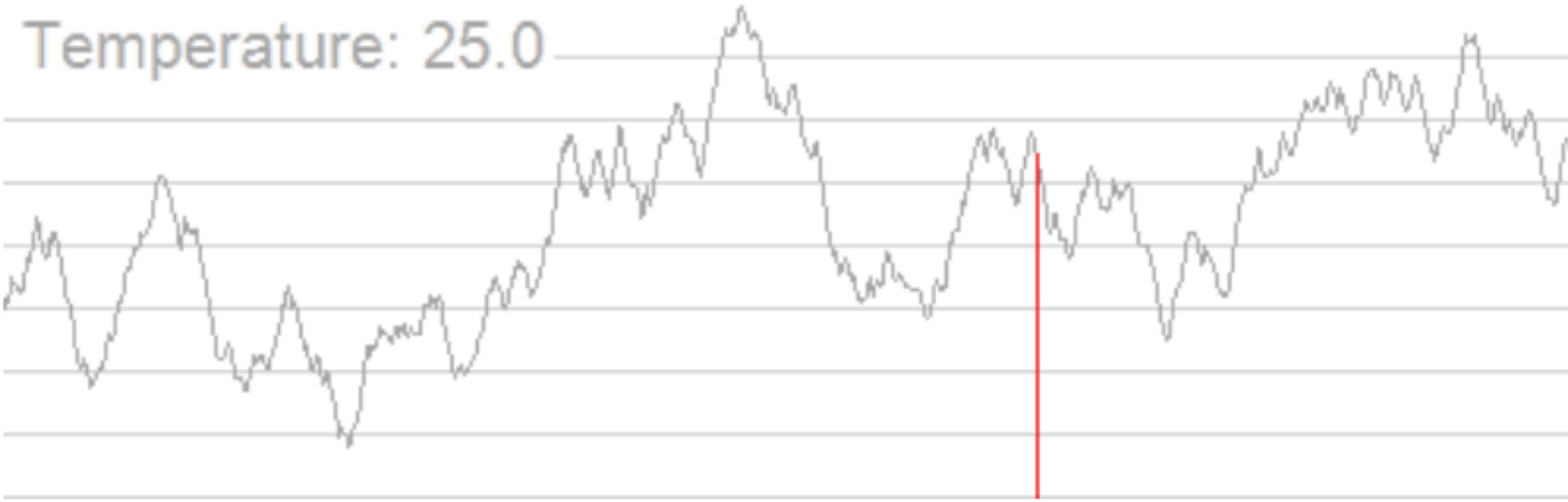


[By Kingpin13 - Own work, CC0, https://commons.wikimedia.org/w/index.php?curid=25010763]

Note 1: the line is not jumping to non-neighbouring positions, it's just moving very fast!

Note 2: in this example we have only 2 neighbours for each solution, but in real world problems we may have many neighbours for each solution.

Note 3: real world problems may have much more dimensions.



[By Kingpin13 - Own work, CC0, https://commons.wikimedia.org/w/index.php?curid=25010763]

Note 4: it may be very difficult to visualise the quality function in real world problems.

Note 5: the algorithm itself does not know the shape of the function beforehand.

Simulated Annealing

Simulated Annealing (assuming maximisation)

1. current_solution = generate initial solution randomly

2. Repeat:

    2.1 rand_neighbour = generate random neighbour of current_solution

    2.2 If quality(rand_neighbour) <= quality(current_solution) {

        2.2.1 With some probability,
        current_solution = rand_neighbour

    } Else current_solution = rand_neighbour

    2.3 Reduce probability

  Until a maximum number of iterations

Summary

- Simulated annealing gives some chance to accept a neighbour that has equal or worse quality than the current solution, in order to avoid getting trapped in local optima.

- The probability of accepting such bad neighbours should reduce over time.

- The worse the quality of this neighbour, the smaller the probability of accepting it should be.

Next

- How to calculate this probability?

- How to reduce it over time?