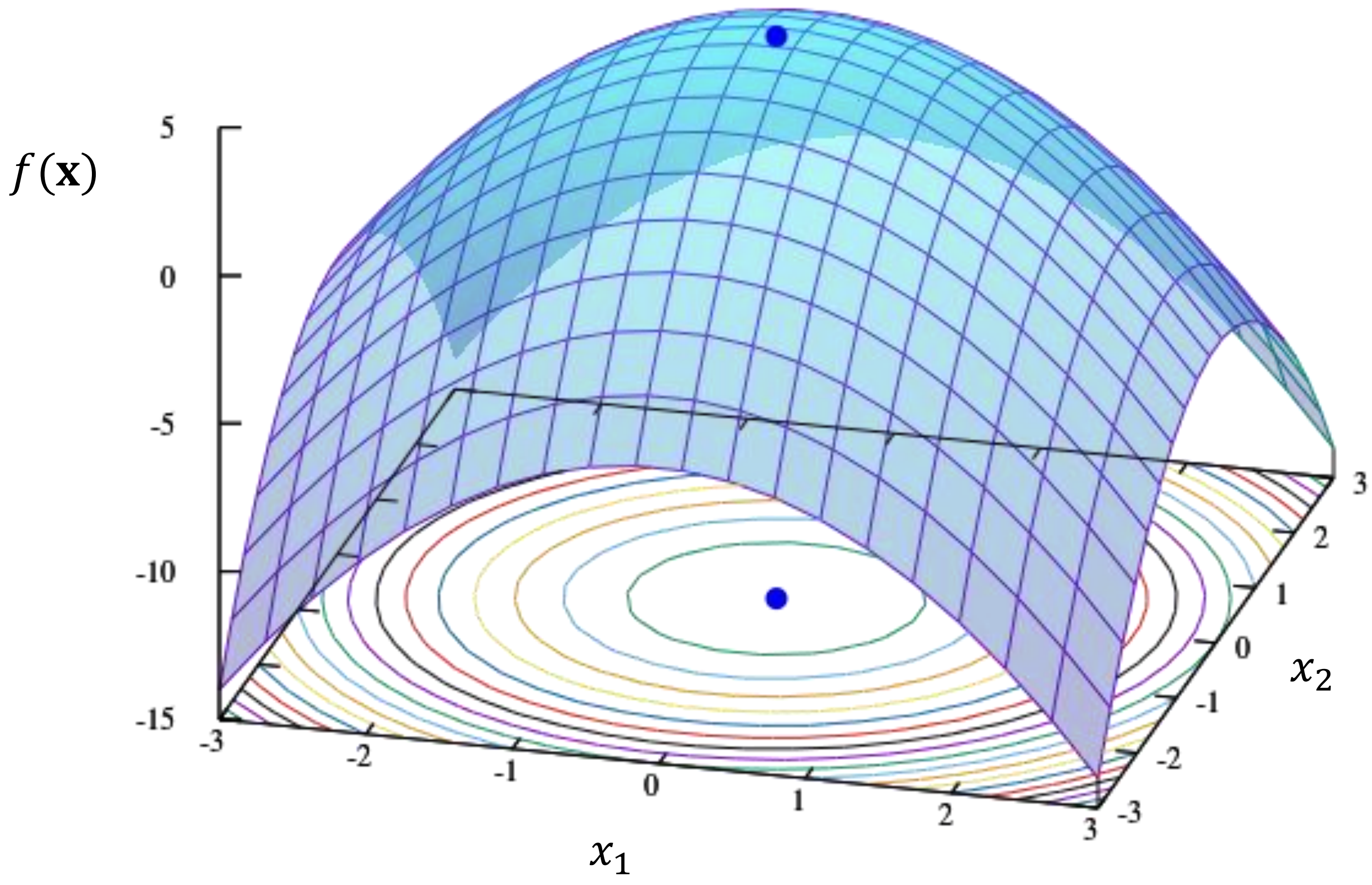
Global and local search

Stochastic optimization

$$\mathbf{x}^* = \text{argmax}(f(\mathbf{x}))$$

For differentiable loss functions – gradient descent.

For not differentiable of unknown loss functions – local and global search.



Global vs. Local

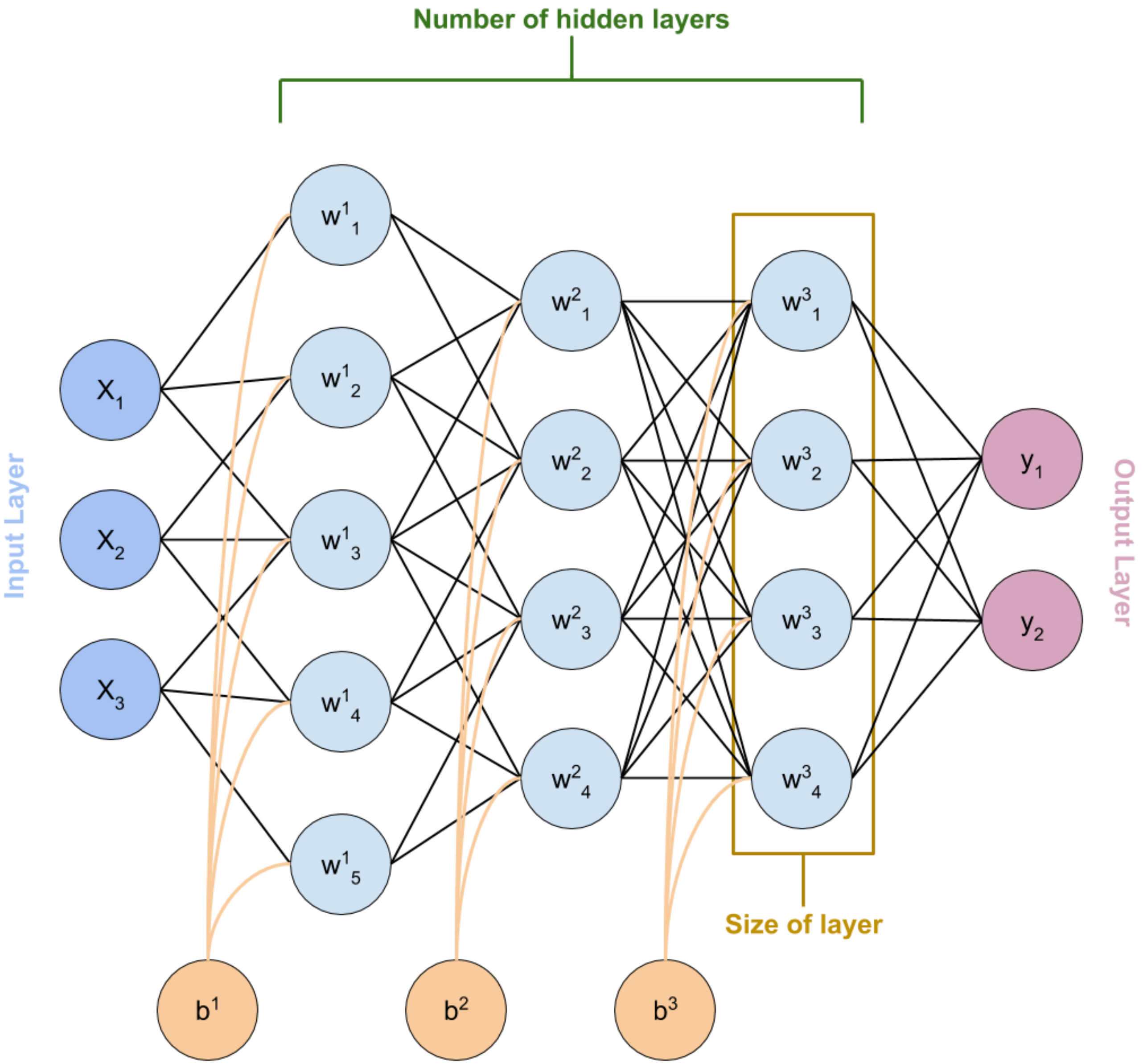
Global search does not analyze a local neighborhood of a point.

Examples: random search, random walk, grid search, Bayesian optimization, cross-entropy optimization.

Local search – choose a direction in the local neighborhood of a point.

Examples: hill climb, simulated annealing, genetic algorithm.

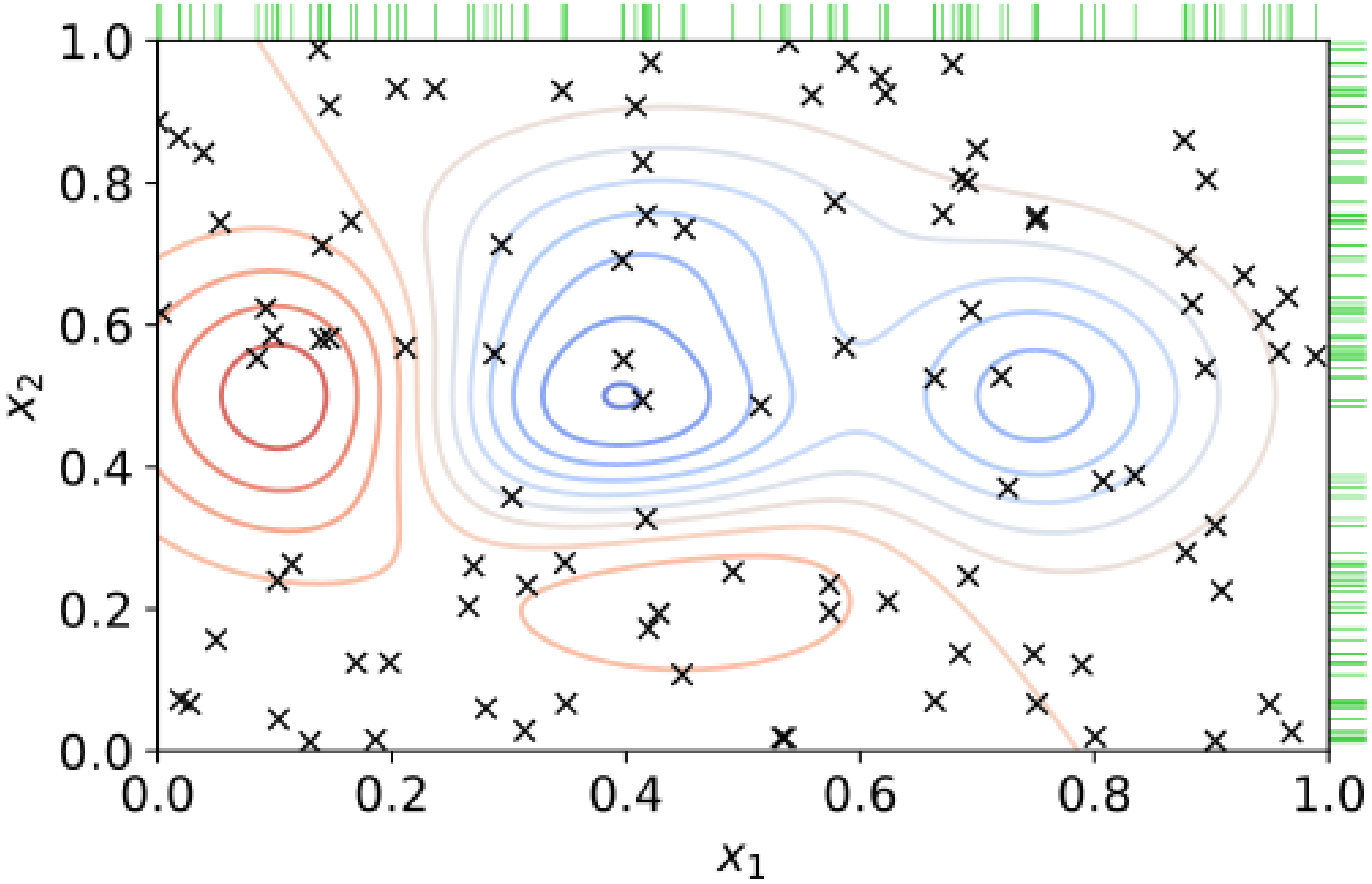
Hyperparameter optimization



Traveling Salesman Problem (TSP)

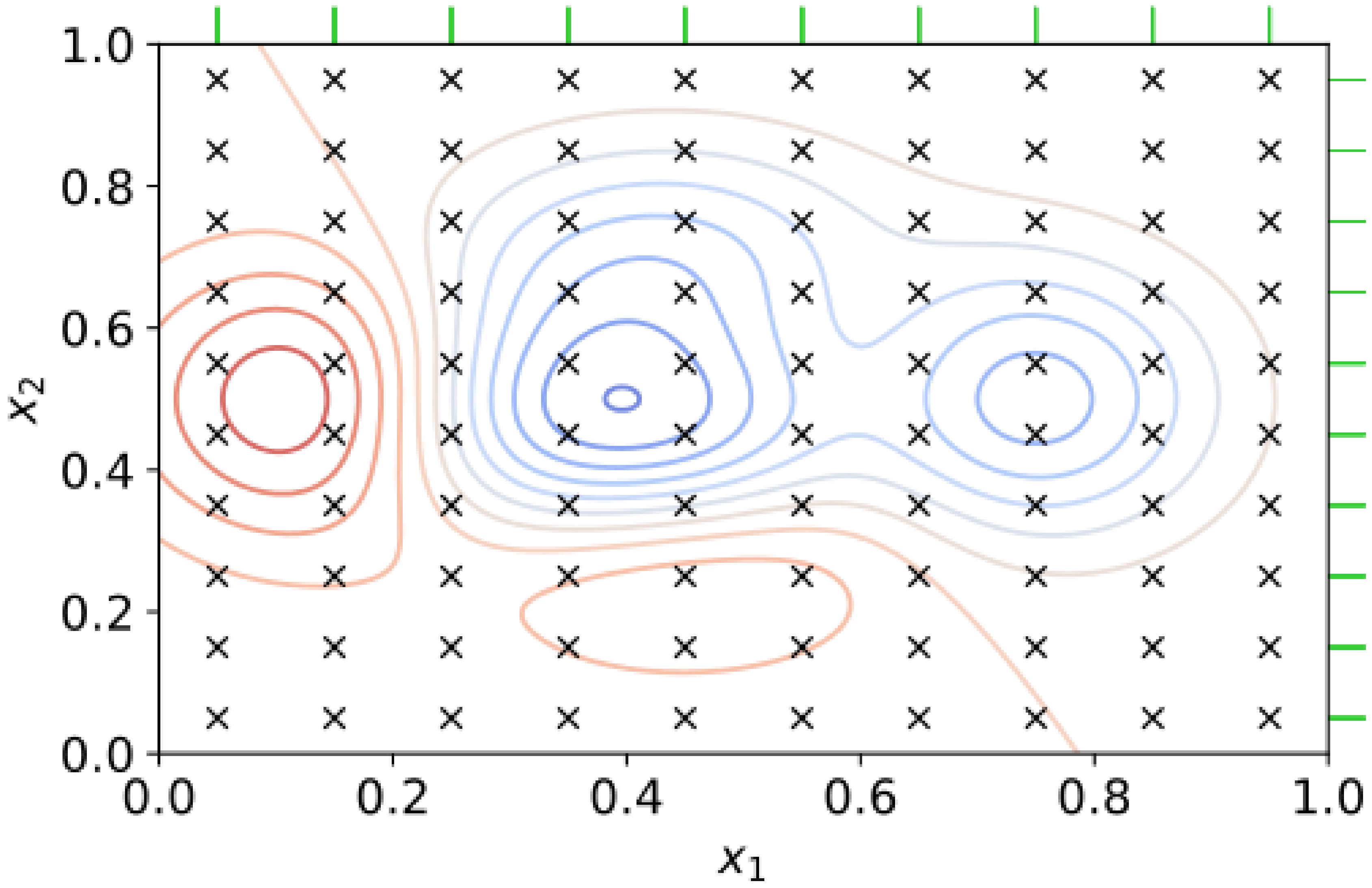
Random Search

Query the feature space many times at random and pick the best!



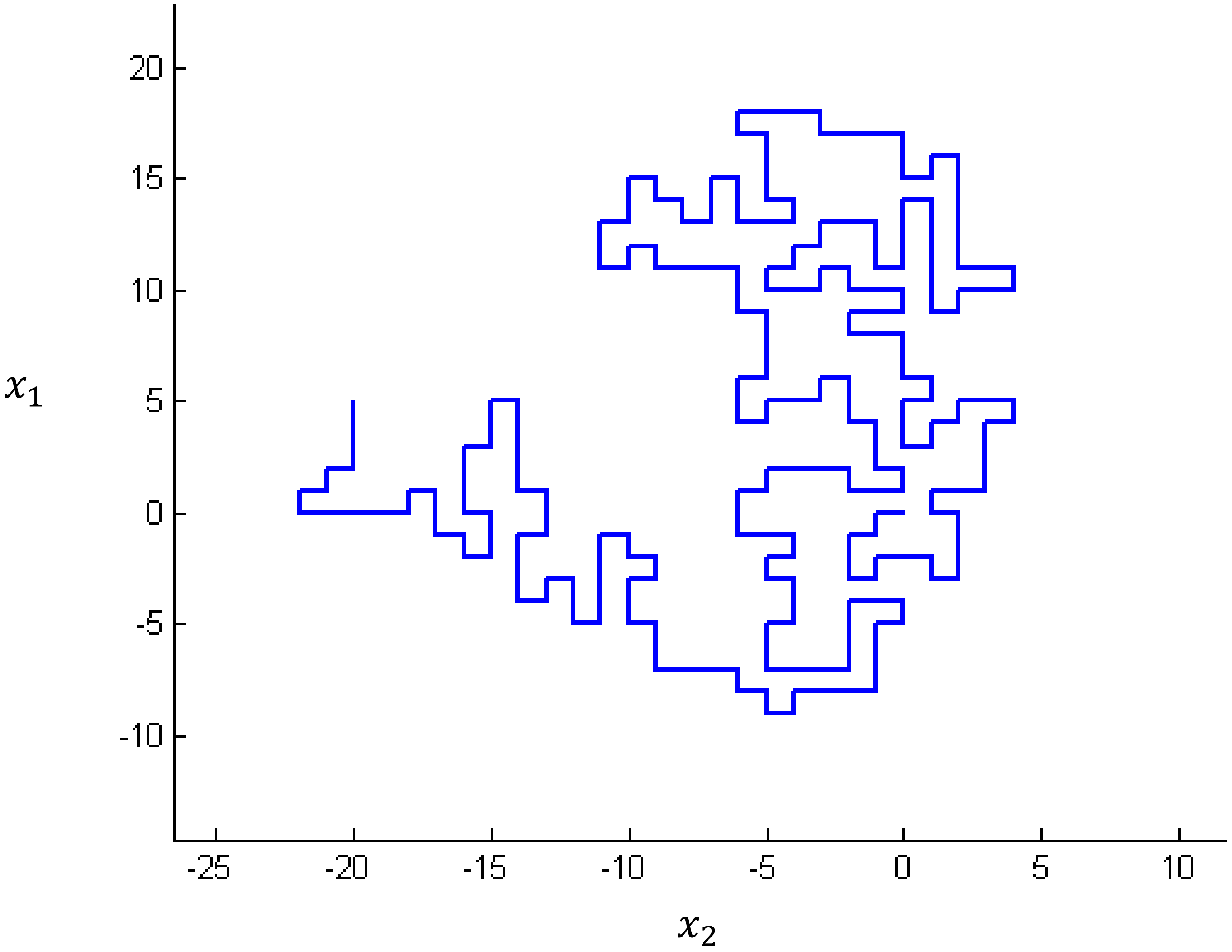
Grid Search

Sample with given intervals along all features.



Random Walk

Same as random search, but we do not have the ability to sample independently from previous point.



Bayesian optimization

$$P(f_{t+1}|\mathcal{D}_{1:t}, \mathbf{x}_{t+1}) = \mathcal{N}\left(\mu_t(\mathbf{x}_{t+1}), \sigma_t^2(\mathbf{x}_{t+1})\right)$$

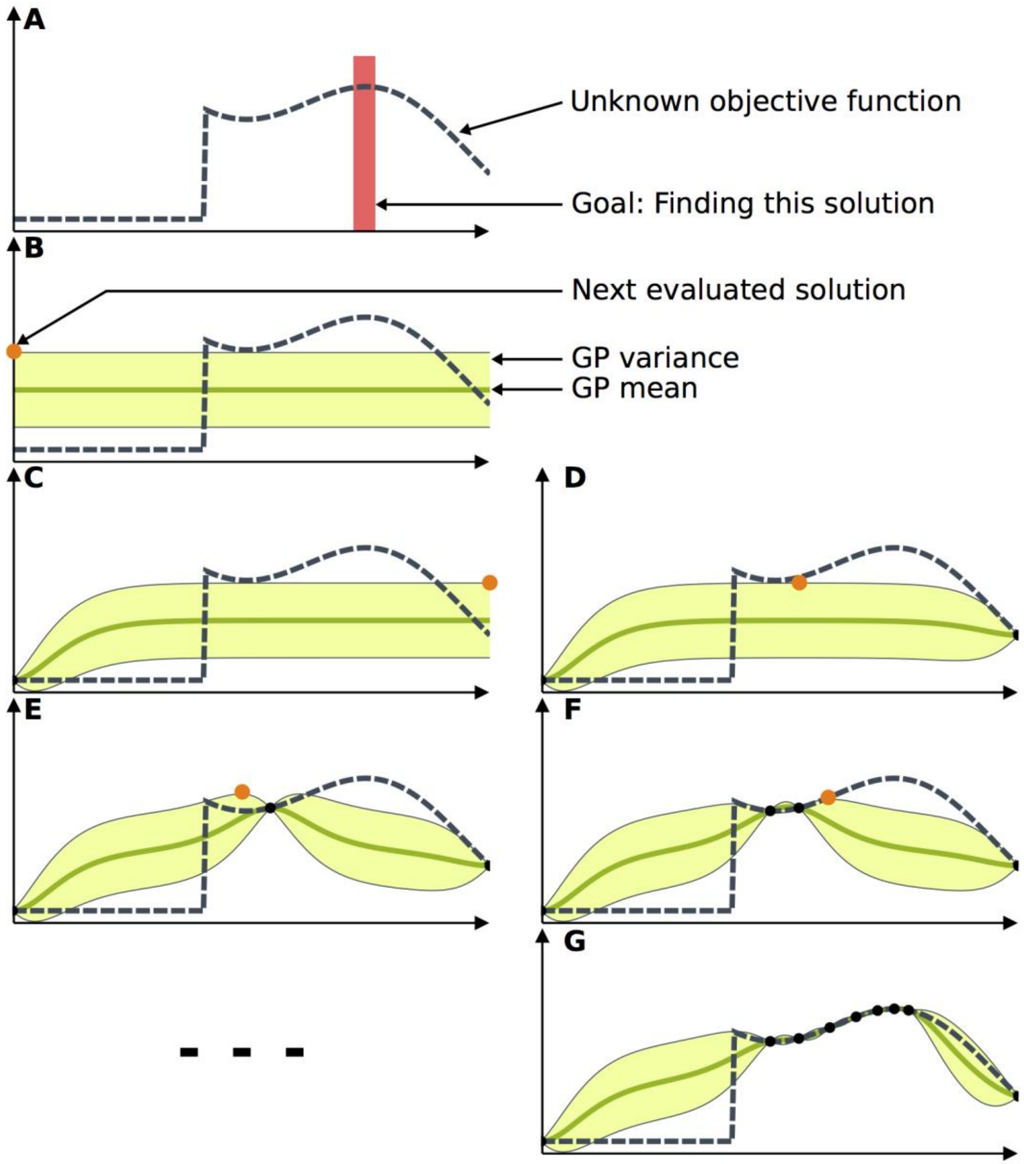
$$\begin{aligned}\mu_t(\mathbf{x}_{t+1}) &= \mathbf{k}^T\mathbf{K}^{-1}\mathbf{f}_{1:t}\\ \sigma_t^2(\mathbf{x}_{t+1}) &= k(\mathbf{x}_{t+1}, \mathbf{x}_{t+1}) - \mathbf{k}^T\mathbf{K}^{-1}\mathbf{k}\end{aligned}$$

$$\mathbf{K} = \begin{bmatrix} k(\mathbf{x}_1, \mathbf{x}_1) & \ldots & k(\mathbf{x}_1, \mathbf{x}_t) \\ \vdots & \ddots & \vdots \\ k(\mathbf{x}_t, \mathbf{x}_1) & \ldots & k(\mathbf{x}_t, \mathbf{x}_t) \end{bmatrix} + \sigma_{noise}^2 I$$

$$\mathbf{k} = \begin{bmatrix} k(\mathbf{x}_{t+1}, \mathbf{x}_1) & k(\mathbf{x}_{t+1}, \mathbf{x}_2) & \cdots & k(\mathbf{x}_{t+1}, \mathbf{x}_t) \end{bmatrix}$$

The most popular covariance function (Square Exponent):

$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{1}{2}\|\mathbf{x}_i - \mathbf{x}_j\|^2\right)$$



Acquisition functions:

- Expected improvement

$$\mathrm{EI}(x) = \mathbb{E}\max(f(\mathrm{x}) - f(\mathrm{x}_{\mathrm{best}}), 0)$$

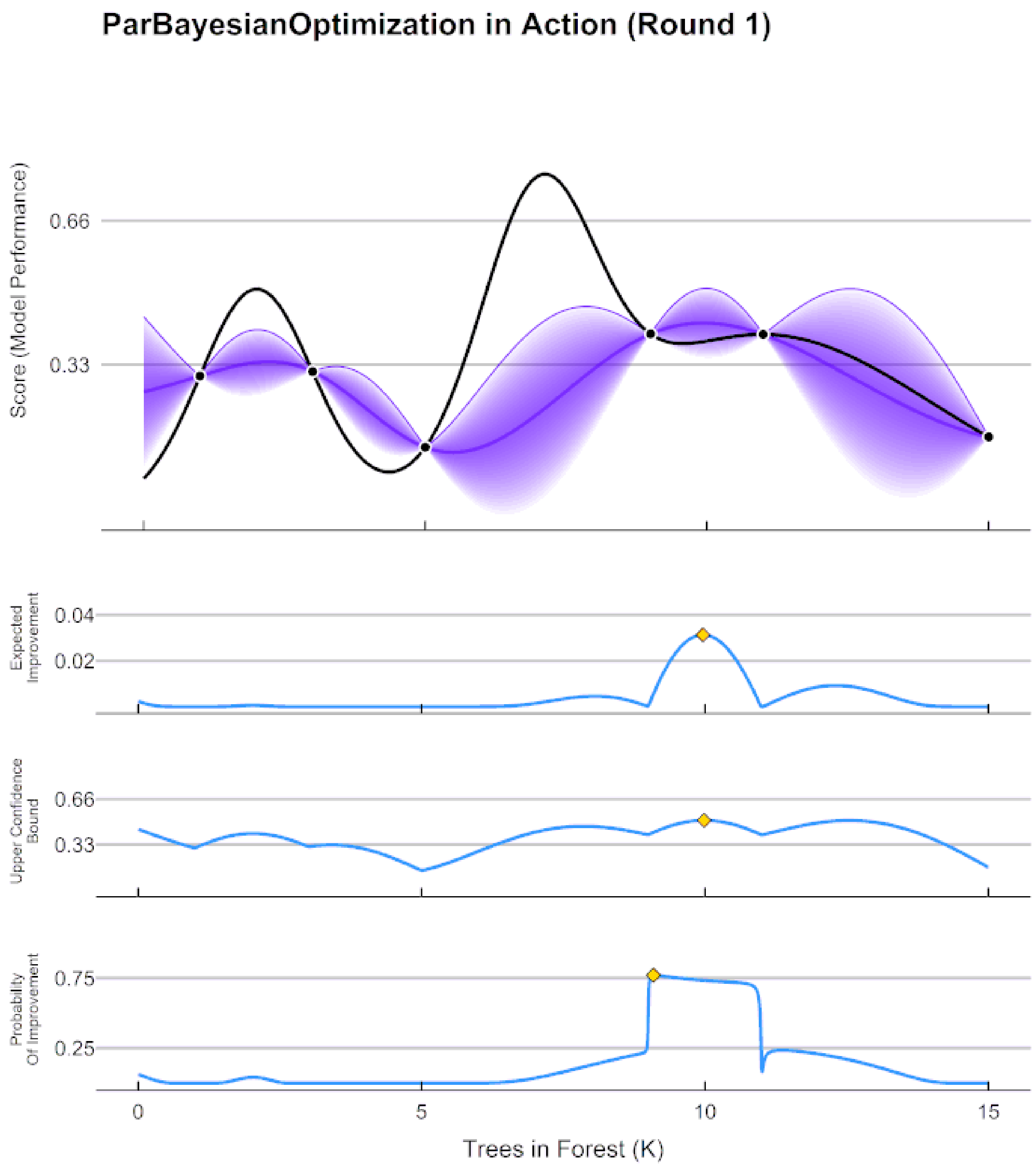
- Upper confidence bound

$$\mathrm{UCB}(\mathrm{x}) = \mu(\mathrm{x}) + \beta\sigma(\mathrm{x}),$$

where β is a user-defined parameter

- Probability of improvement

$$\mathrm{PI}(\mathrm{x}) = \mathbb{E}[f(\mathrm{x}) > f(\mathrm{x}_{best})]$$



Cross-entropy method

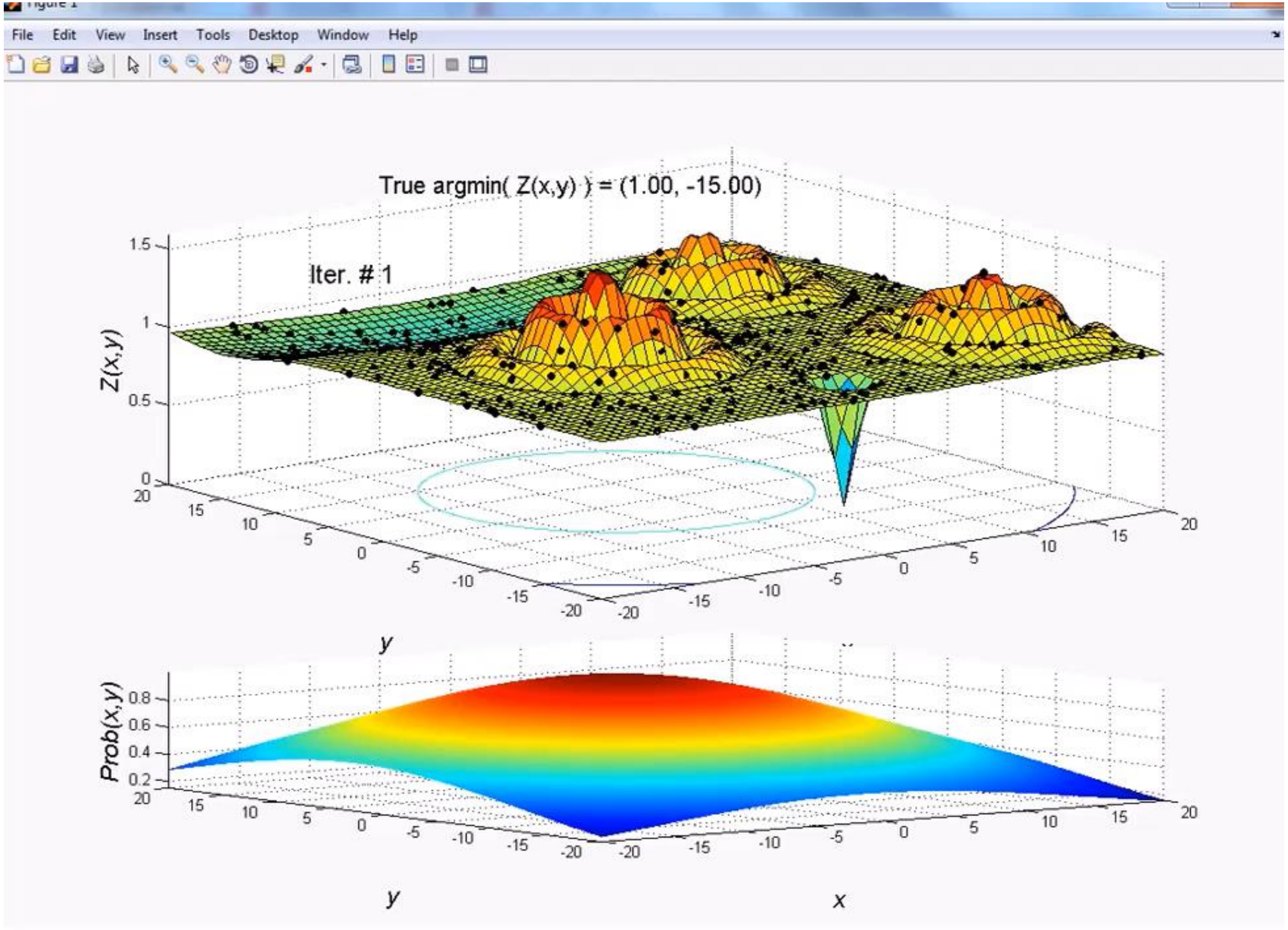
1. At step $t = 1$ we choose random parameters θ_0 for a distribution.

2. Generate a random sample $\mathrm{x}_1 \dots \mathrm{x}_N$ from distribution $p(\mathrm{x}; \theta_{t-1})$.

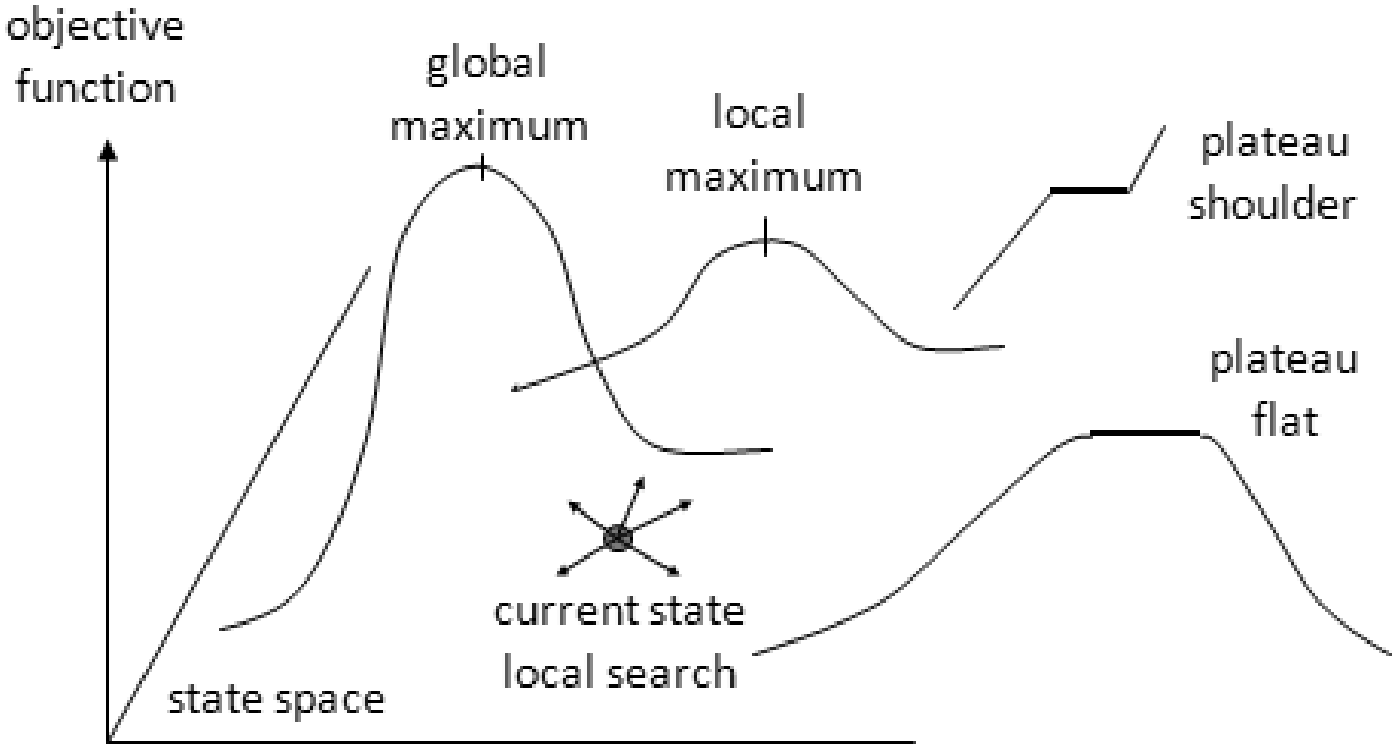
3. Optimize the next set of parameters based on the following criteria:

$$\theta_t = \arg\max_{\theta} \sum_{\mathrm{x}_i \in best\ k} y_i \frac{p(\mathrm{x}_i; \theta)}{p(\mathrm{x}_i; \theta_{t-1})} \log p(\mathrm{x}_i; \theta_{t-1})$$

Cross entropy method



Hill Climb



Similar to **Gradient Descent** method.

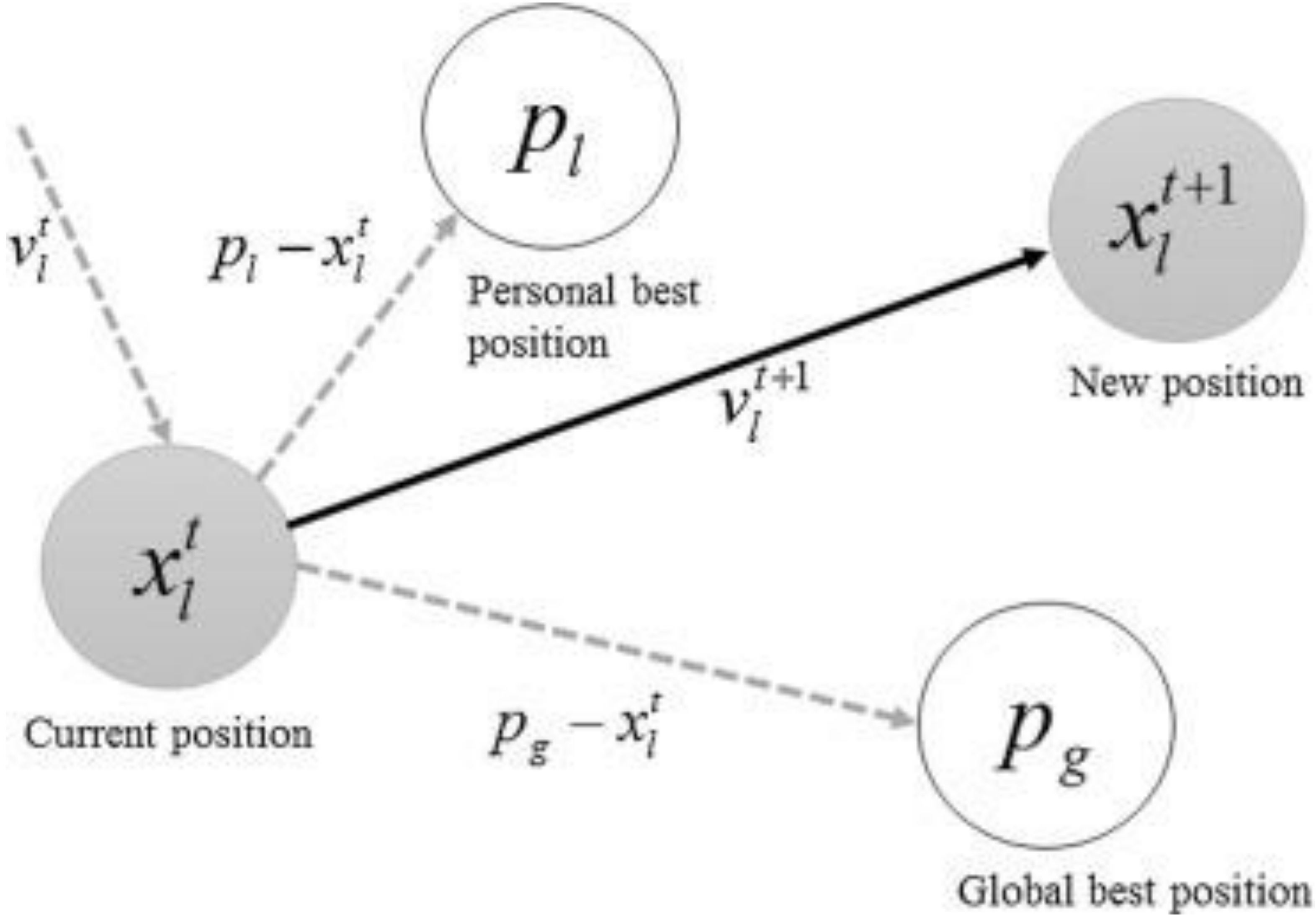
Calculate metric for possible shifts and step in a direction of the best value.

Stochastic Hill Climbing:
Step with probability proportional to increase in the metric (for example, with softmax).

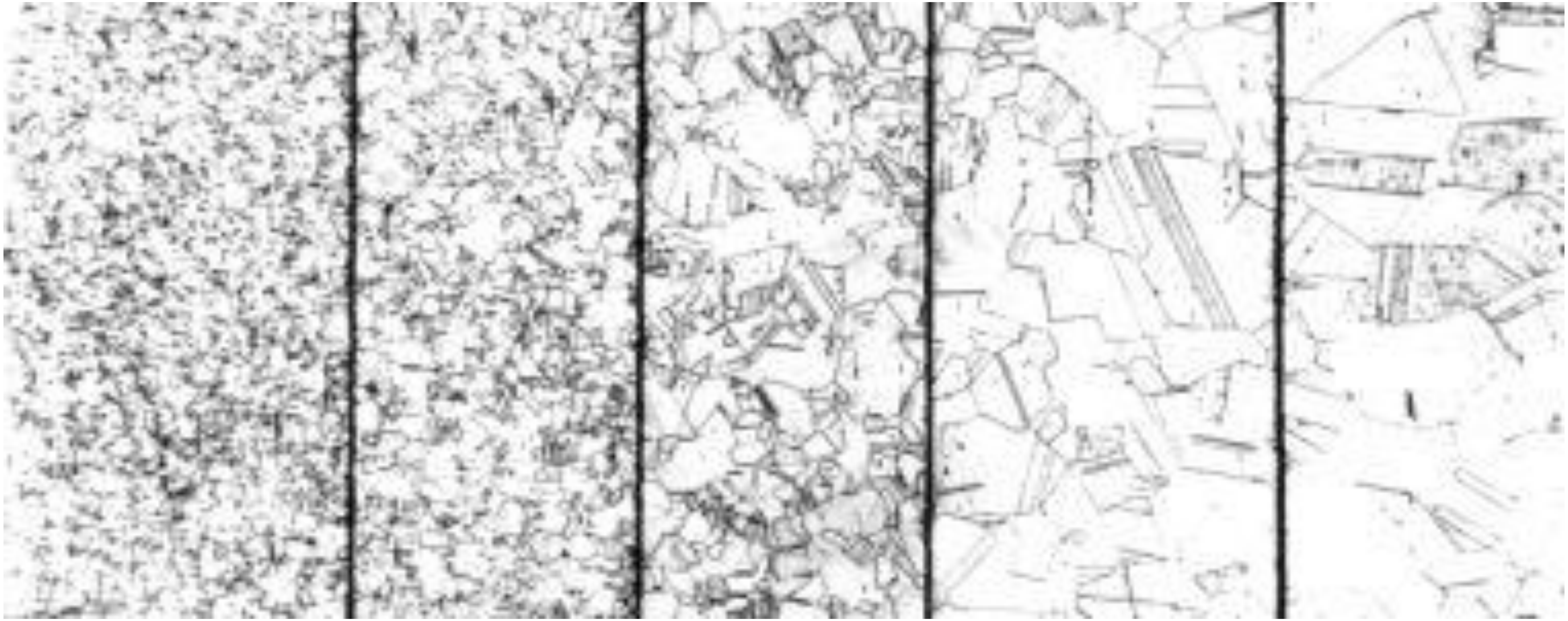
Taboo Search:
Do not return to visited states.

Particle swarm optimization:
Multiple “climbers” that pass information.

Particle swarm optimization



Simulated annealing



Annealing of metal

Simulated Annealing

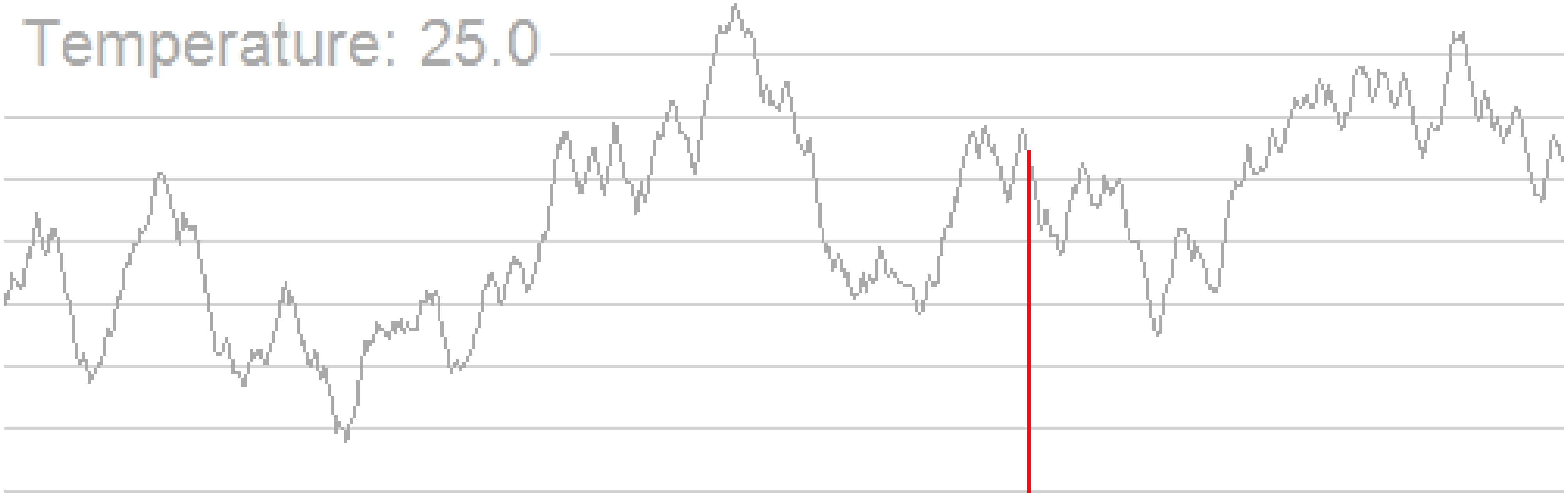
Introduce temperature to stochastic hill climb.

For Softmax:

$$P(s_i) = \frac{e^{\frac{\Delta E(s_i)}{T}}}{\sum e^{\frac{\Delta E(s_j)}{T}}}$$

	T - temperature			
	10000	10	1	0.1
ΔE	P(ΔE)	P(ΔE)	P(ΔE)	P(ΔE)
10	0.25016	0.43944	0.99325	1
5	0.25004	0.26653	0.00669	1.9287E-22
0	0.24991	0.16166	4.51E-05	3.7200E-44
-2	0.24986	0.13235	6.1E-06	7.6676E-53

Simulated Annealing

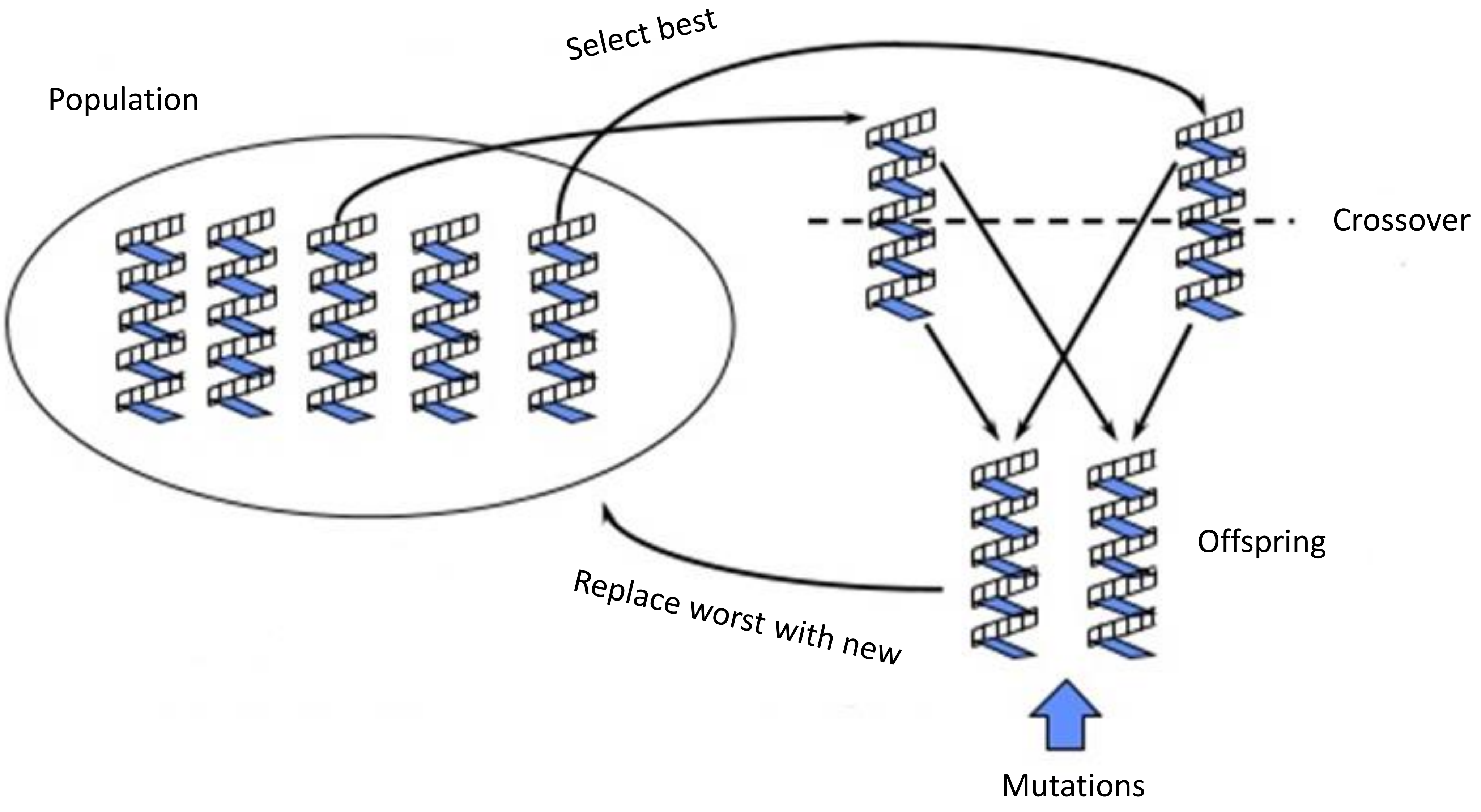


Genetic algorithm

“Evolved antenna”

An example of an evolved antenna is an X-band antenna evolved for a 2006 NASA mission called Space Technology 5.

Genetic algorithm



No Free Lunch Theorem

$$\sum_f P(d_m^y | f, m, a_1) = \sum_f P(d_m^y | f, m, a_2)$$

$d_m^y = \{y_1, y_2 \dots y_m\}$ - sequence of values of optimized function.

$y = f(\mathrm{x})$

a_1, a_2 $-$ algorithms.