stochastics and probability

Lecture 10

Dr. Johannes Pahlke

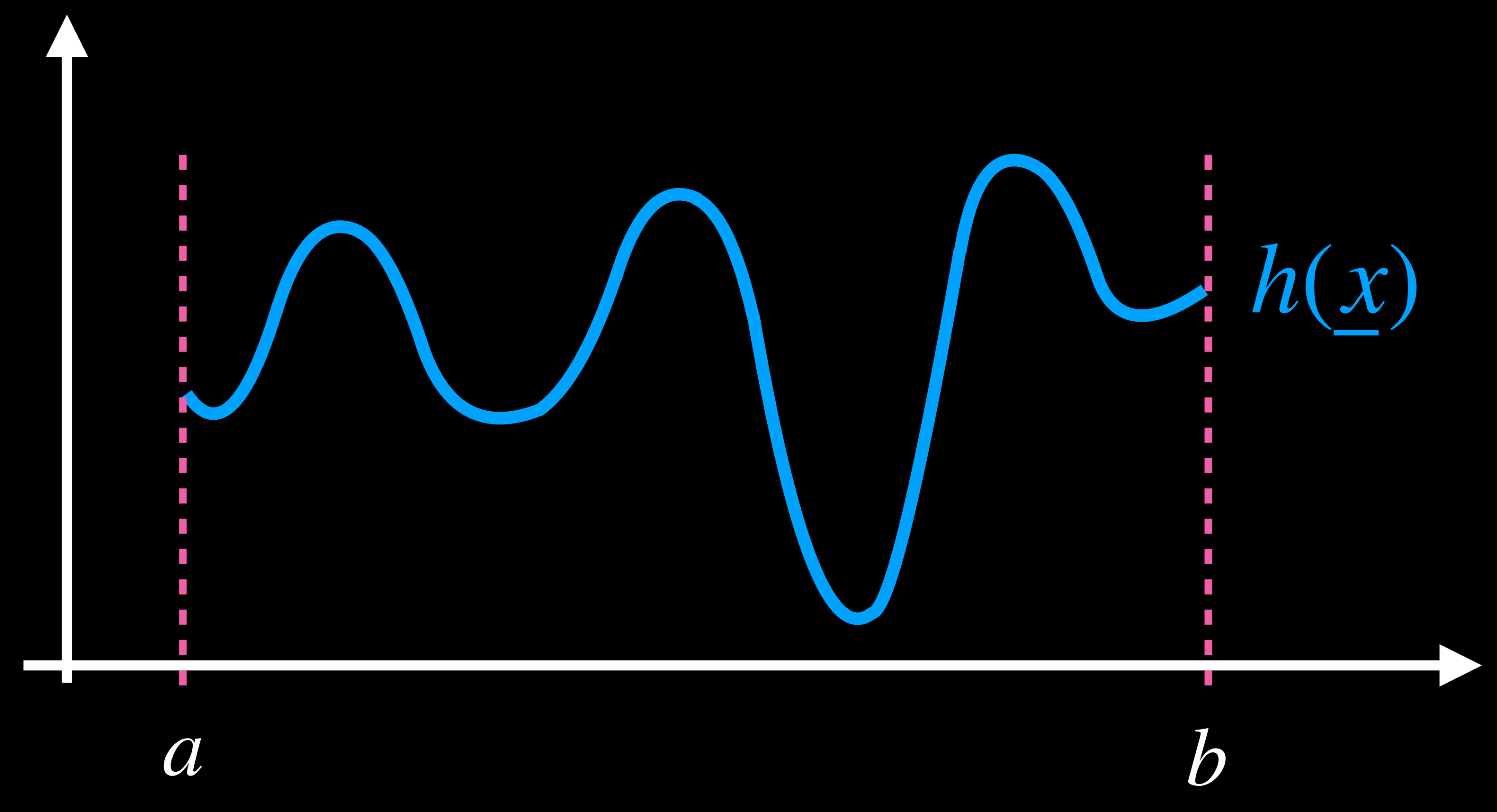
optimization

optimization

Goal:

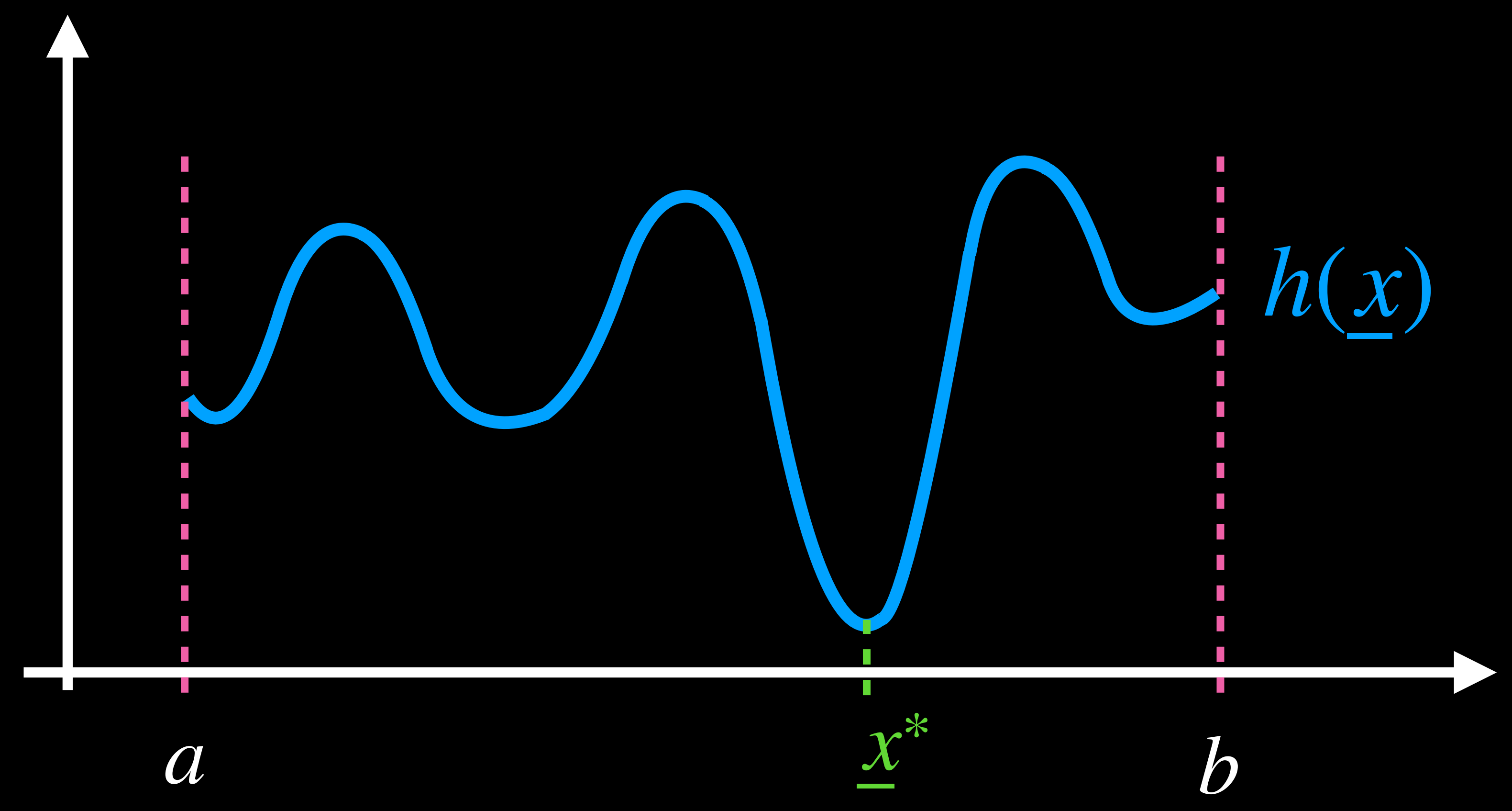
optimization

Goal:



optimization

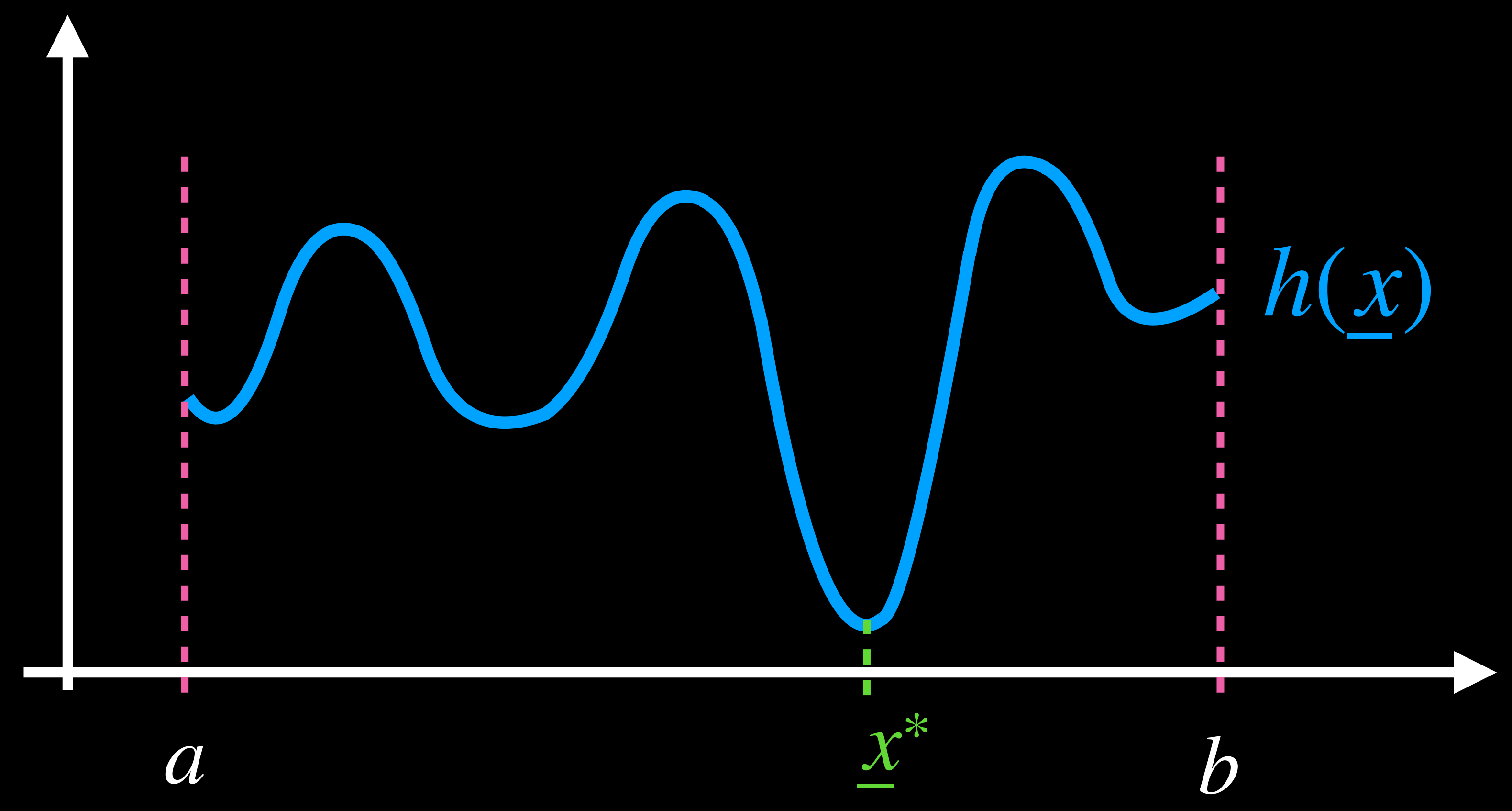
Goal:



optimization

Goal:

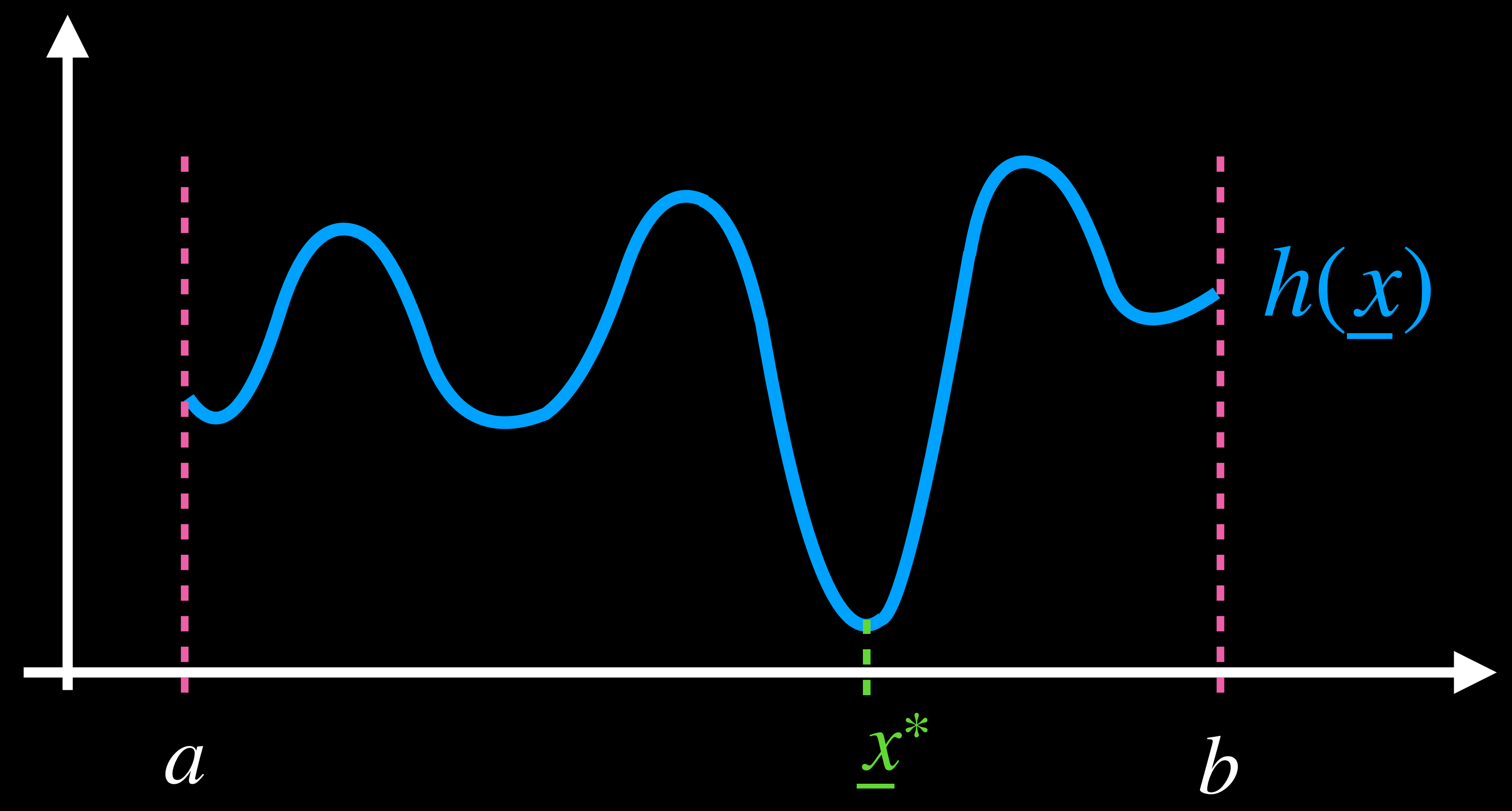
$\underline{x}^* :=$



optimization

Goal:

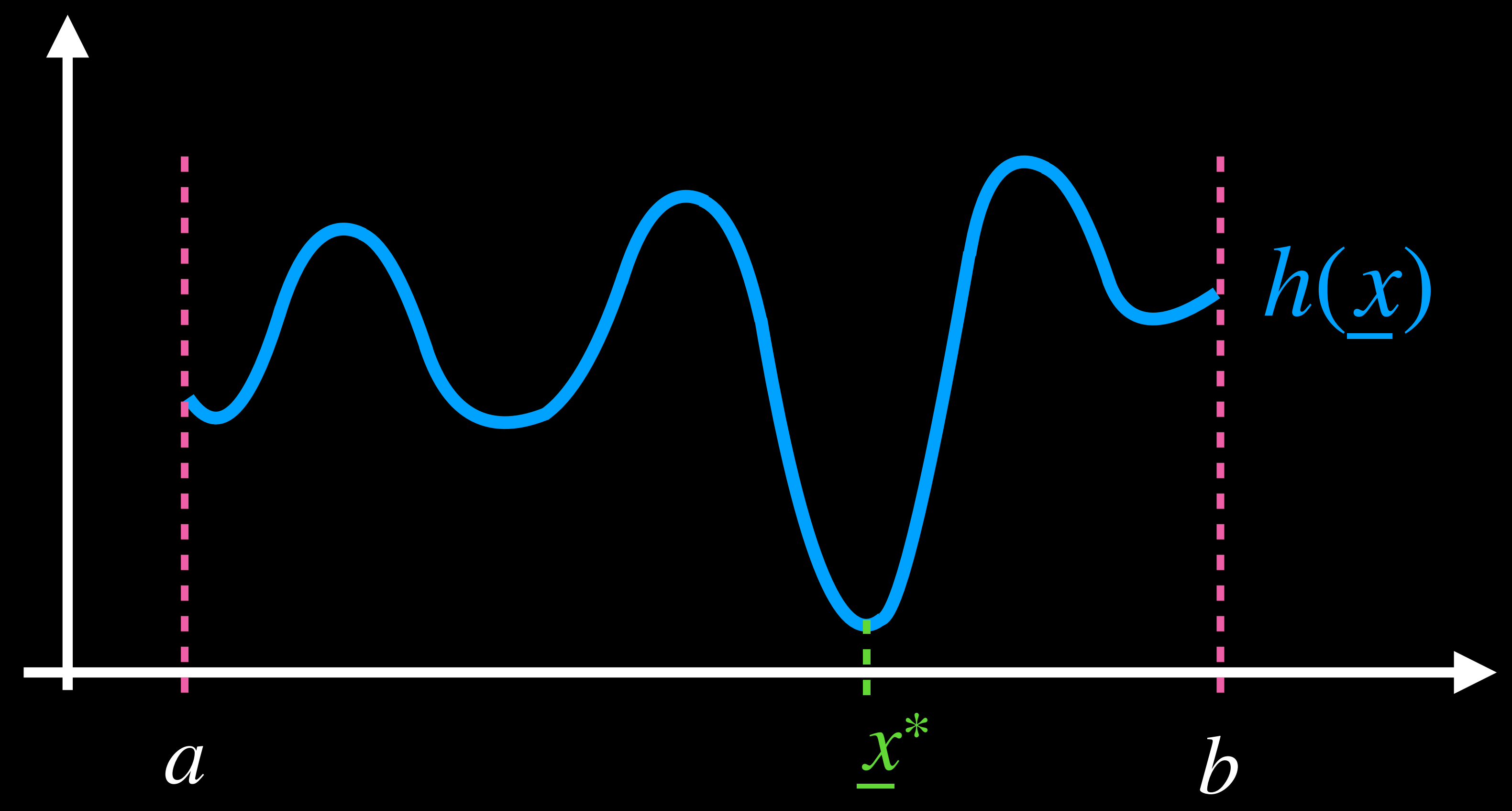
$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$



optimization

Goal:

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$

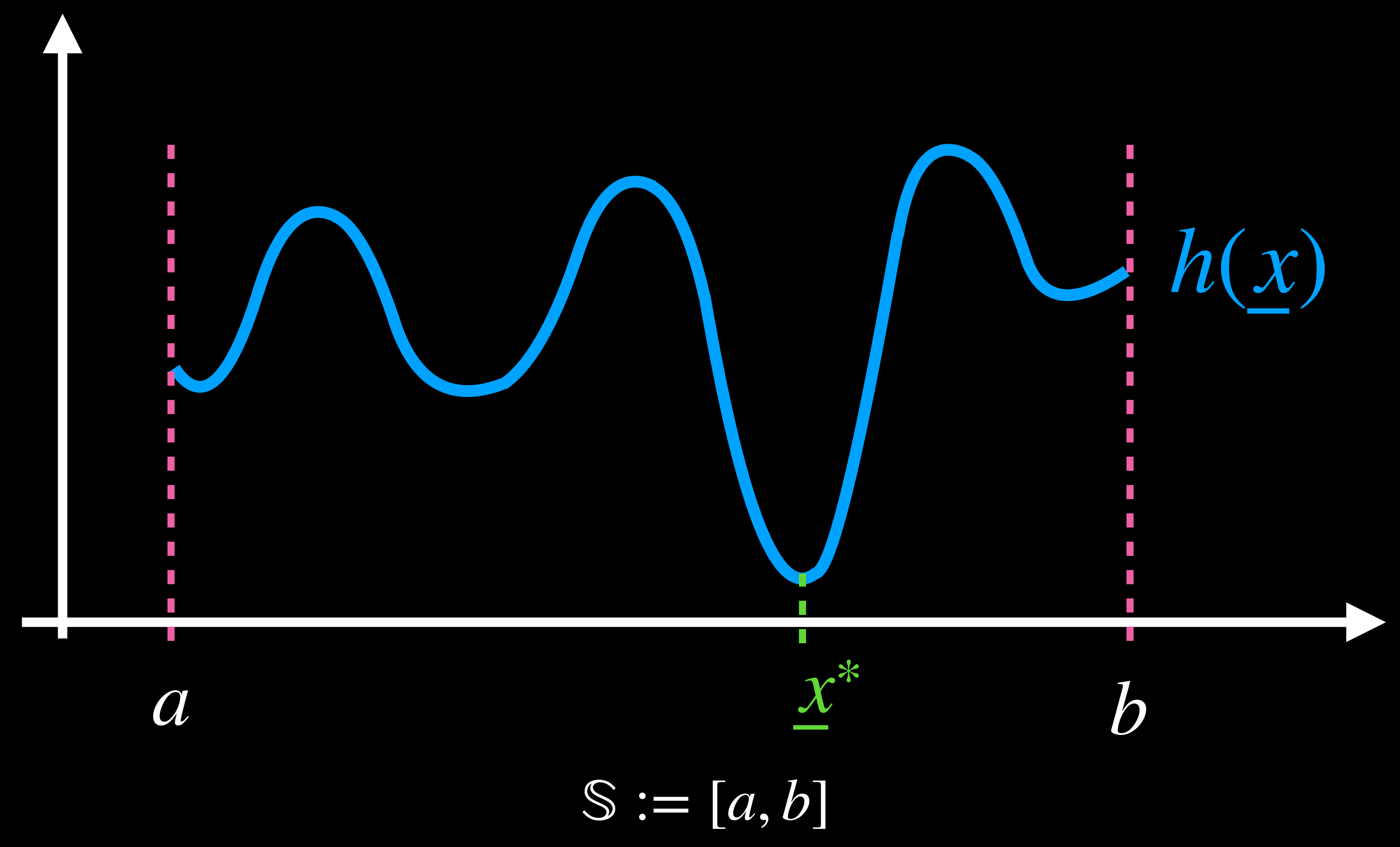


$\mathbb{S} := [a, b]$

optimization

Goal:

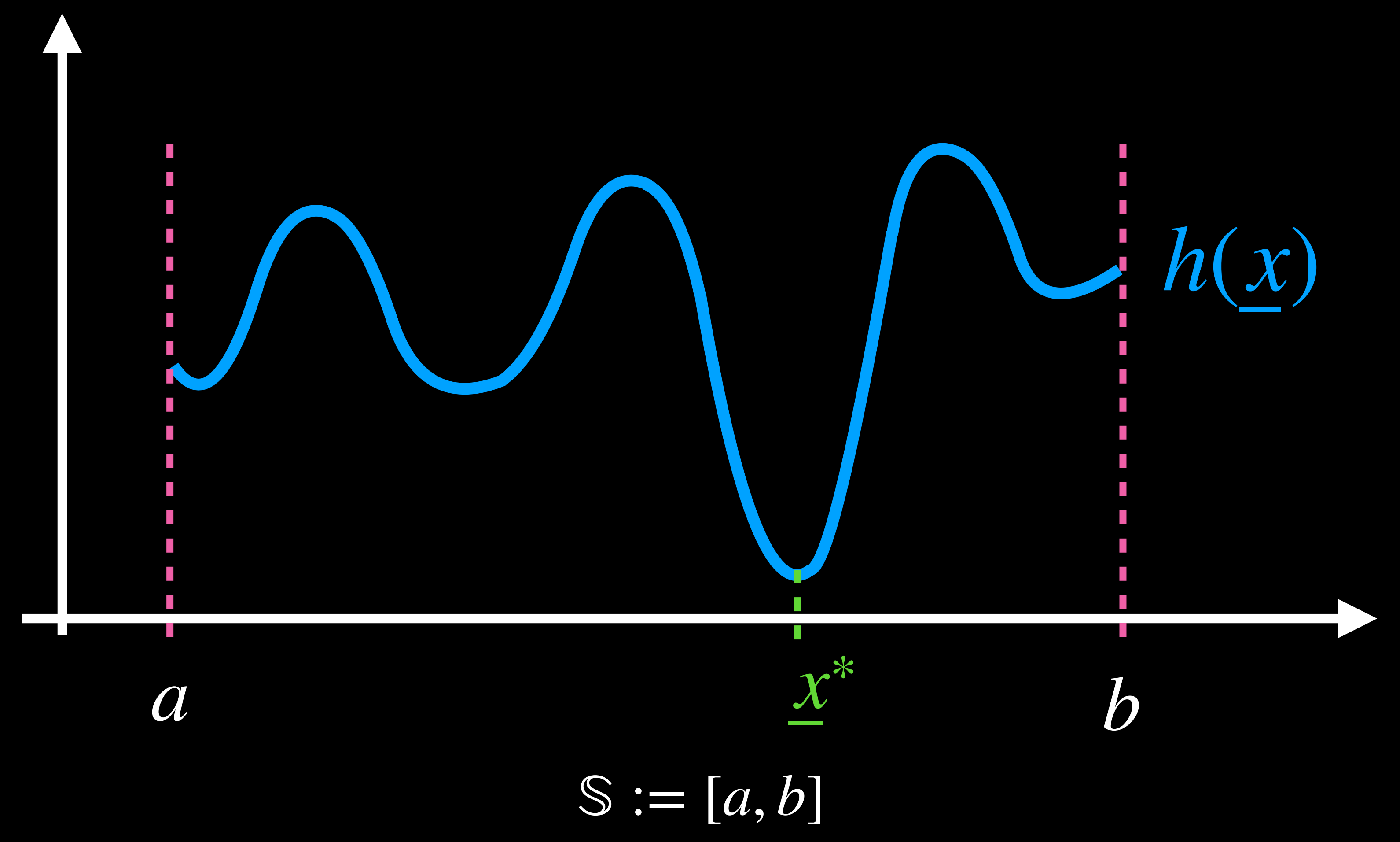
$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right) \qquad \left(\longleftrightarrow \qquad h(\underline{x}^*) = \min_{\underline{x}\in\mathbb{S}}\left(h(\underline{x})\right)\right)$$



optimization

Goal:

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right) \qquad \left(\longleftrightarrow \qquad h(\underline{x}^*) = \min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)\right)$$

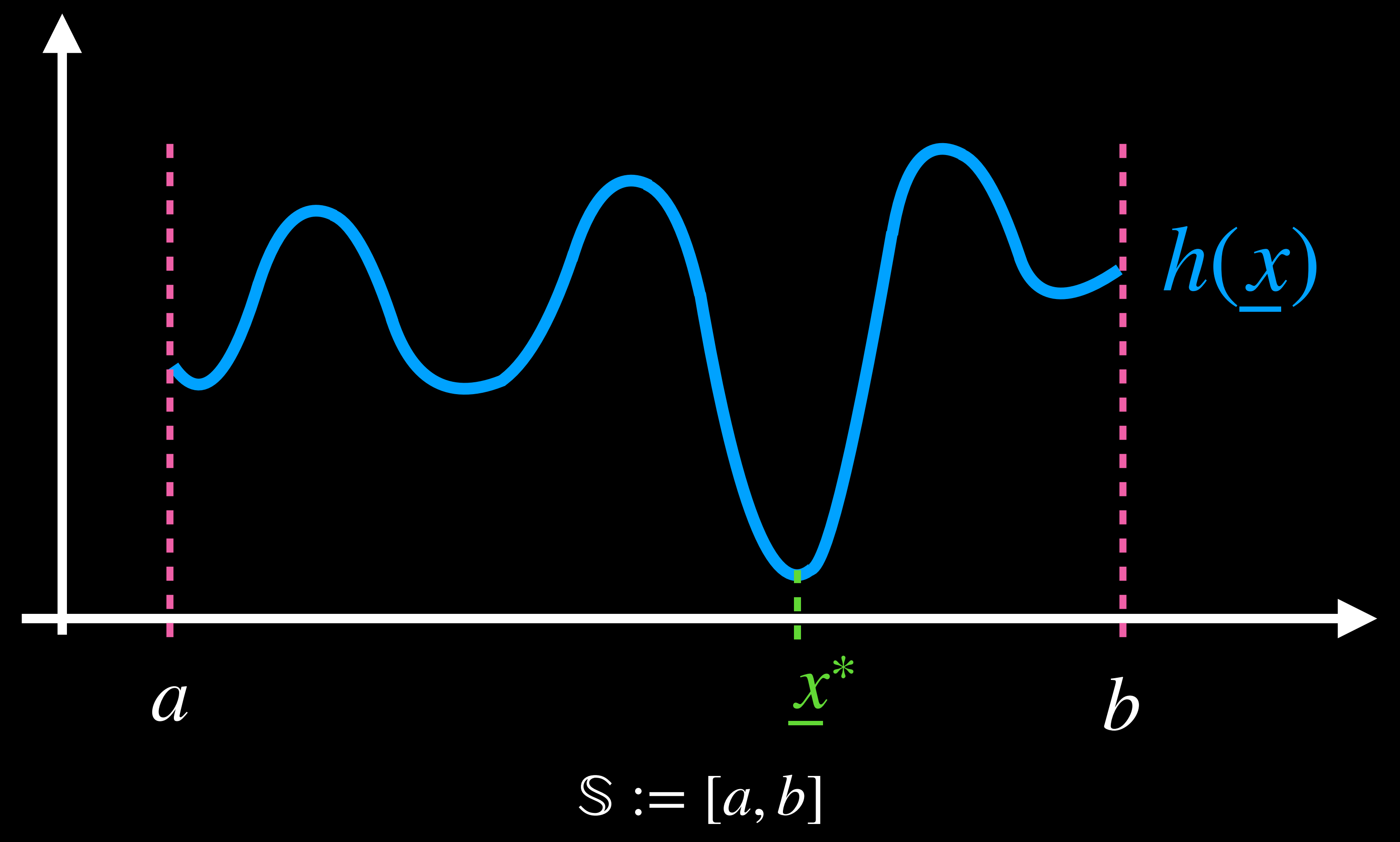


Challenges:

optimization

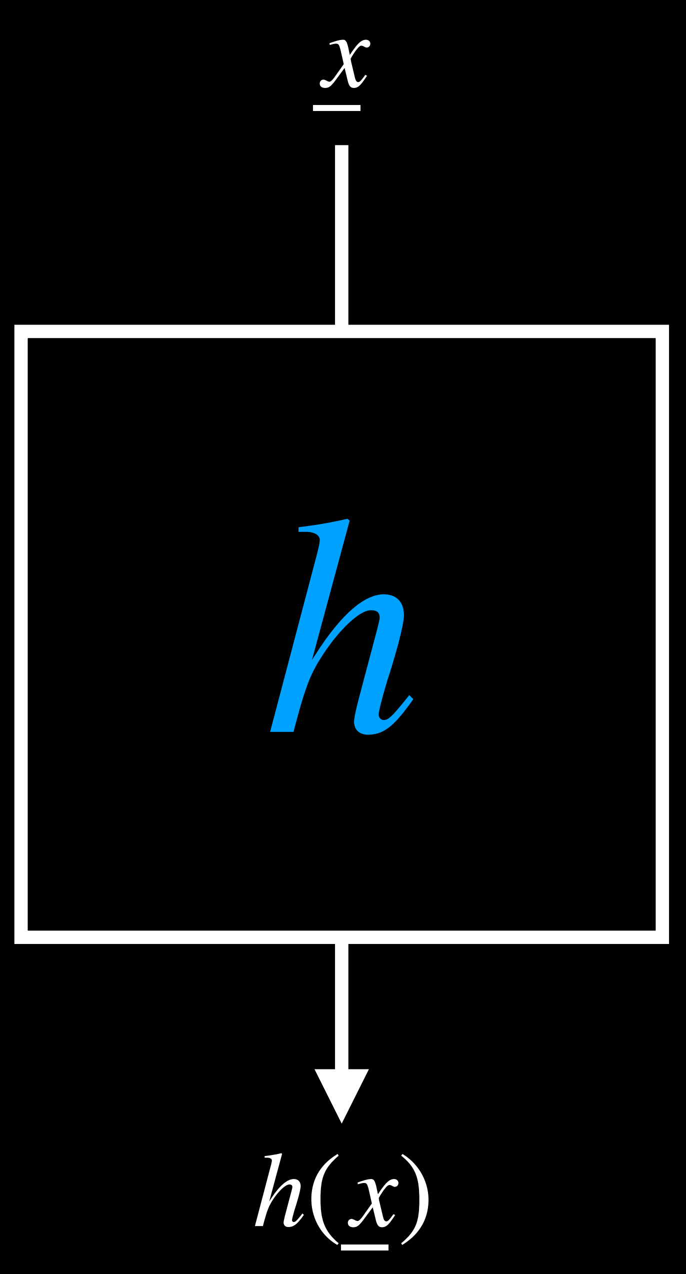
Goal:

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right) \qquad \left(\longleftrightarrow \qquad h(\underline{x}^*) = \min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right) \right)$$

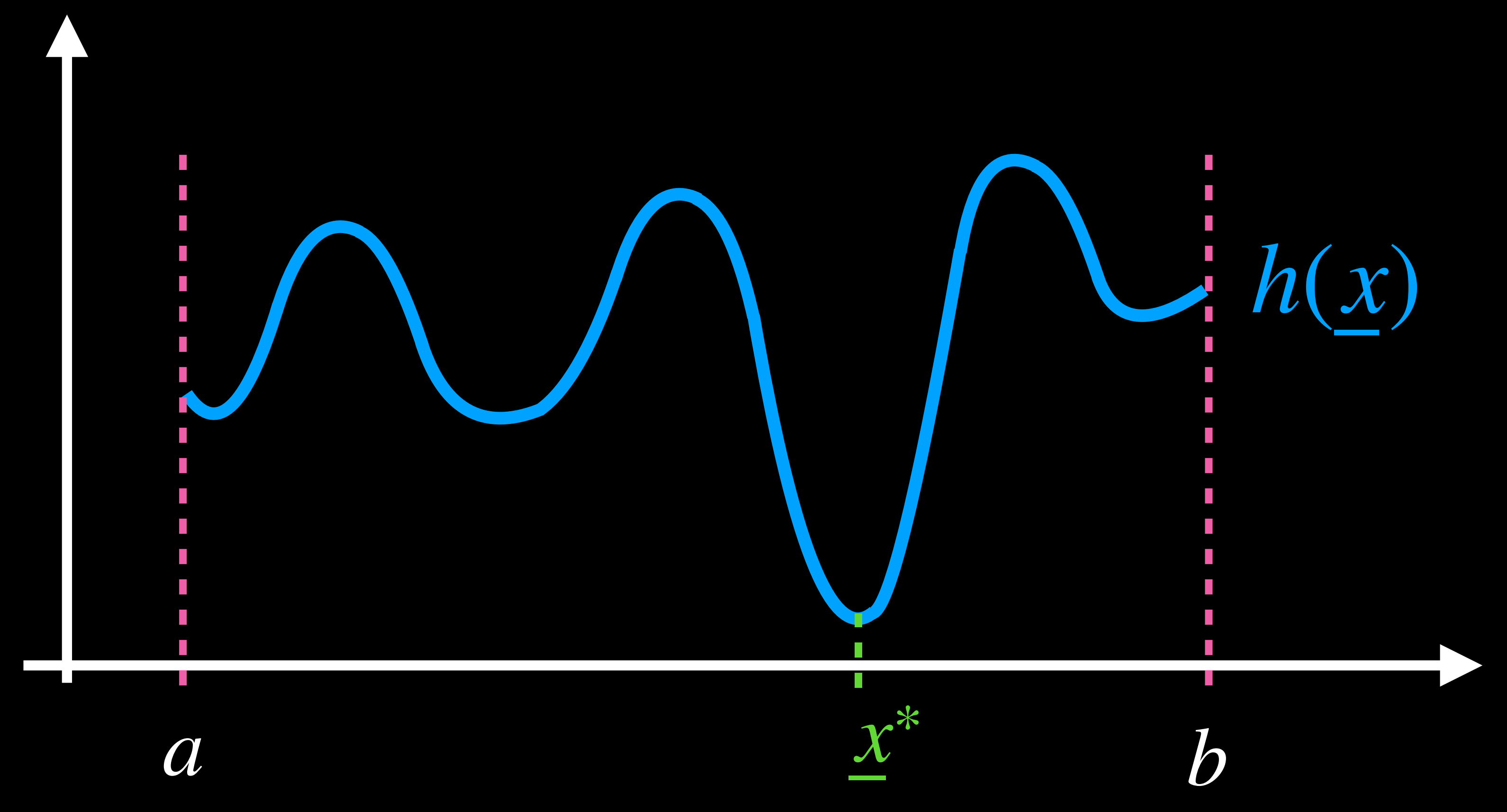


Challenges:

black box



optimization



$\mathbb{S} := [a, b]$

Goal:

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right) \quad \left(\longleftrightarrow \quad h(\underline{x}^*) = \min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right) \right)$$

Challenges:

black box

$\underline{x}$

$h(\underline{x})$

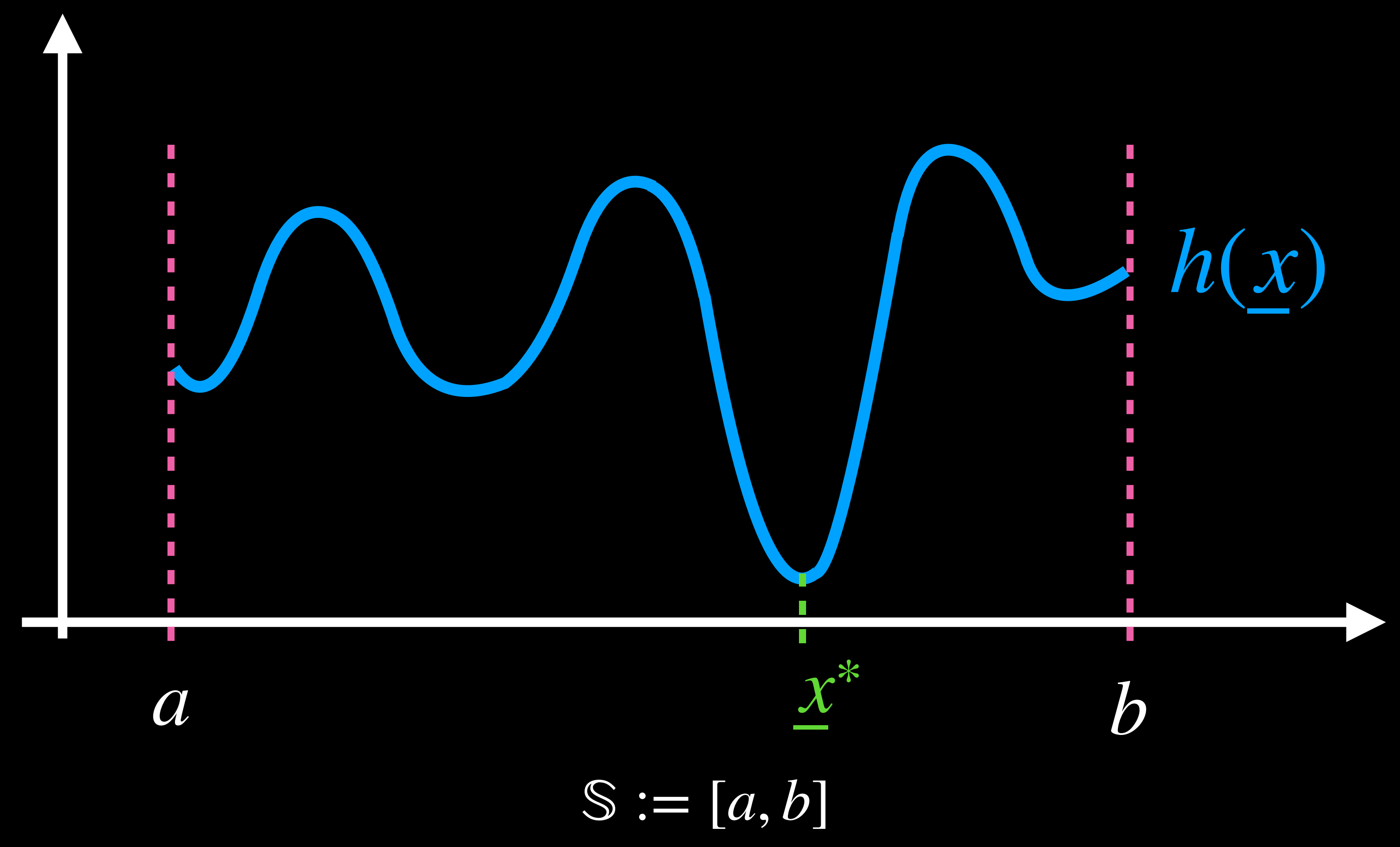
high dimentional

$h(x_1, x_2, x_3, \ldots, x_{100})$

optimization

Goal:

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right) \qquad \left(\longleftrightarrow \qquad h(\underline{x}^*) = \min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right) \right)$$



Challenges:

black box

high dimentional

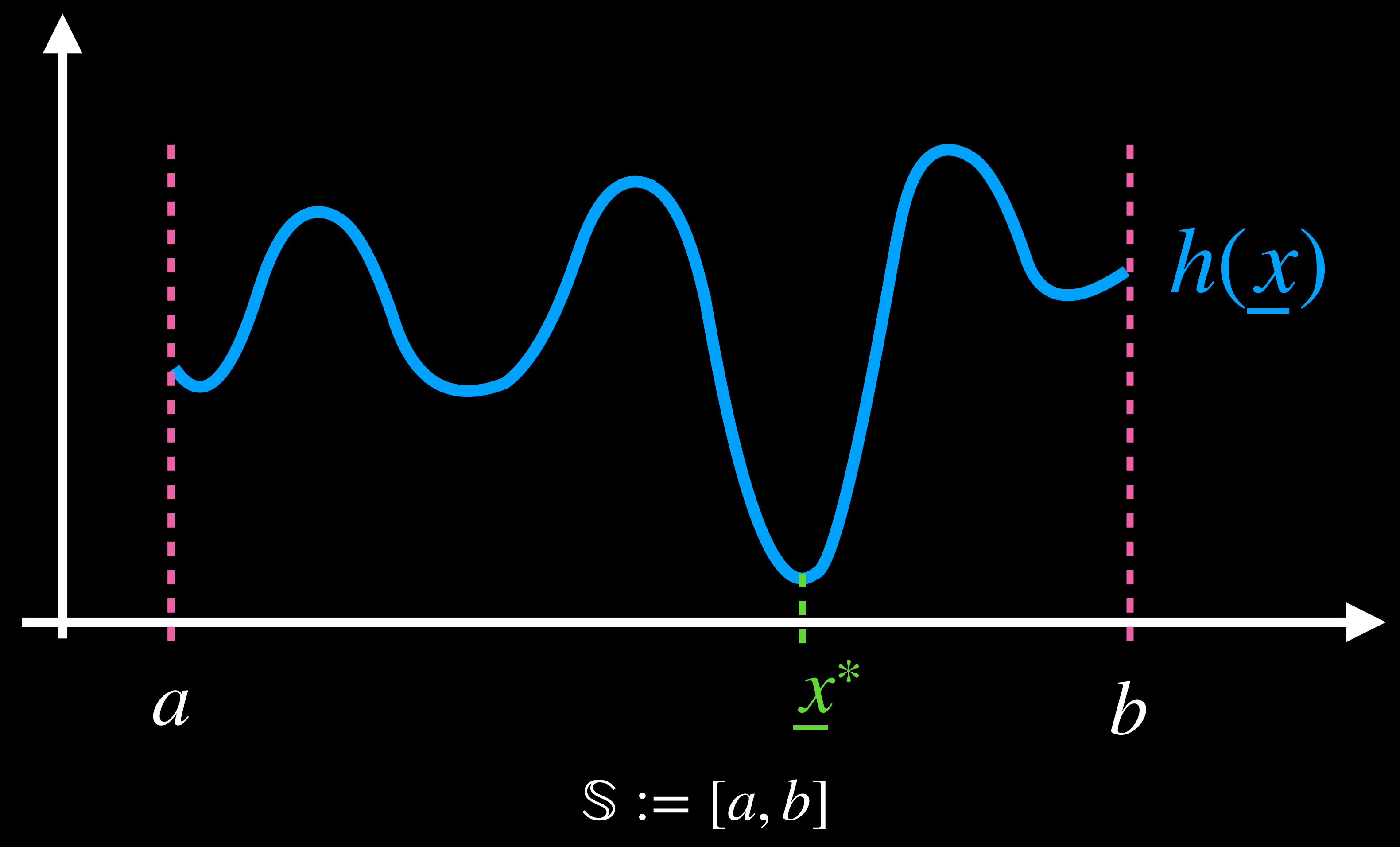
$h(x_1, x_2, x_3, \ldots, x_{100})$

function evaluation are expensive

optimization

Goal:

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right) \qquad \left(\longleftrightarrow \qquad h(\underline{x}^*) = \min_{\underline{x}\in\mathbb{S}}\left(h(\underline{x})\right)\right)$$



Challenges:

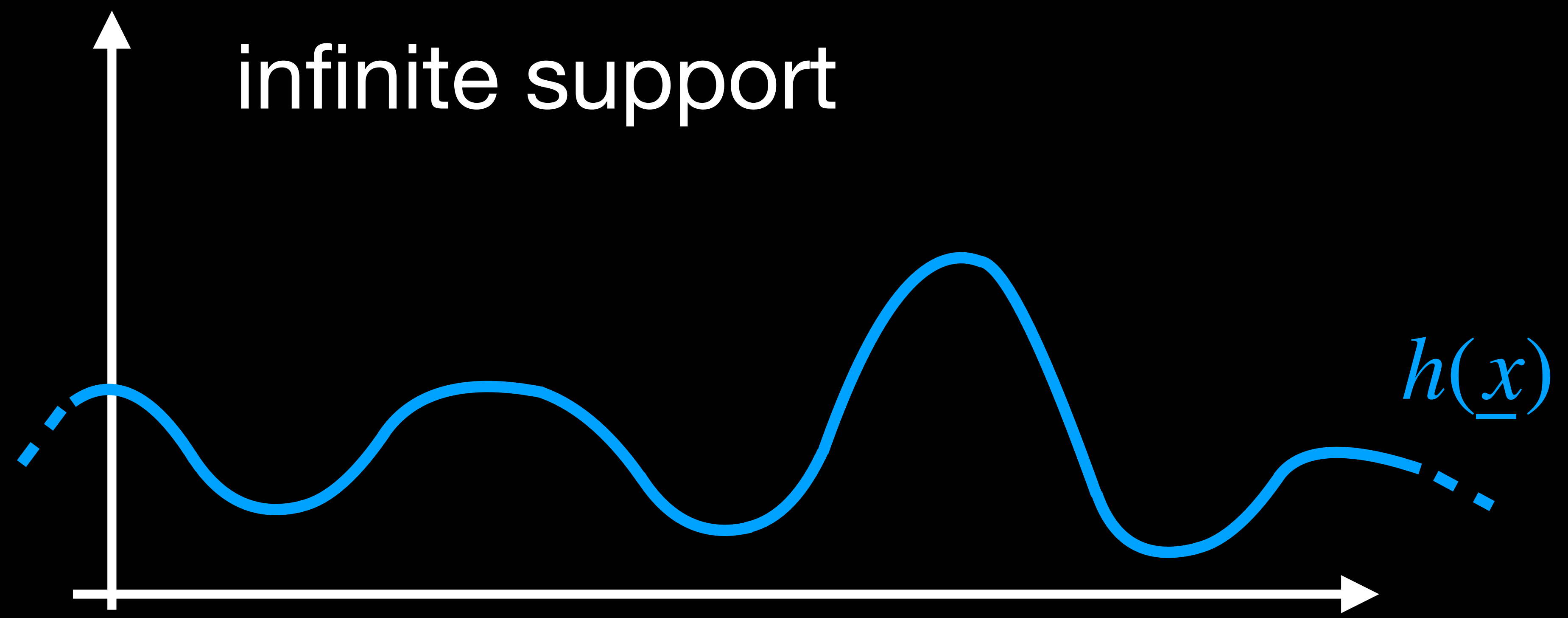
black box

high dimentional

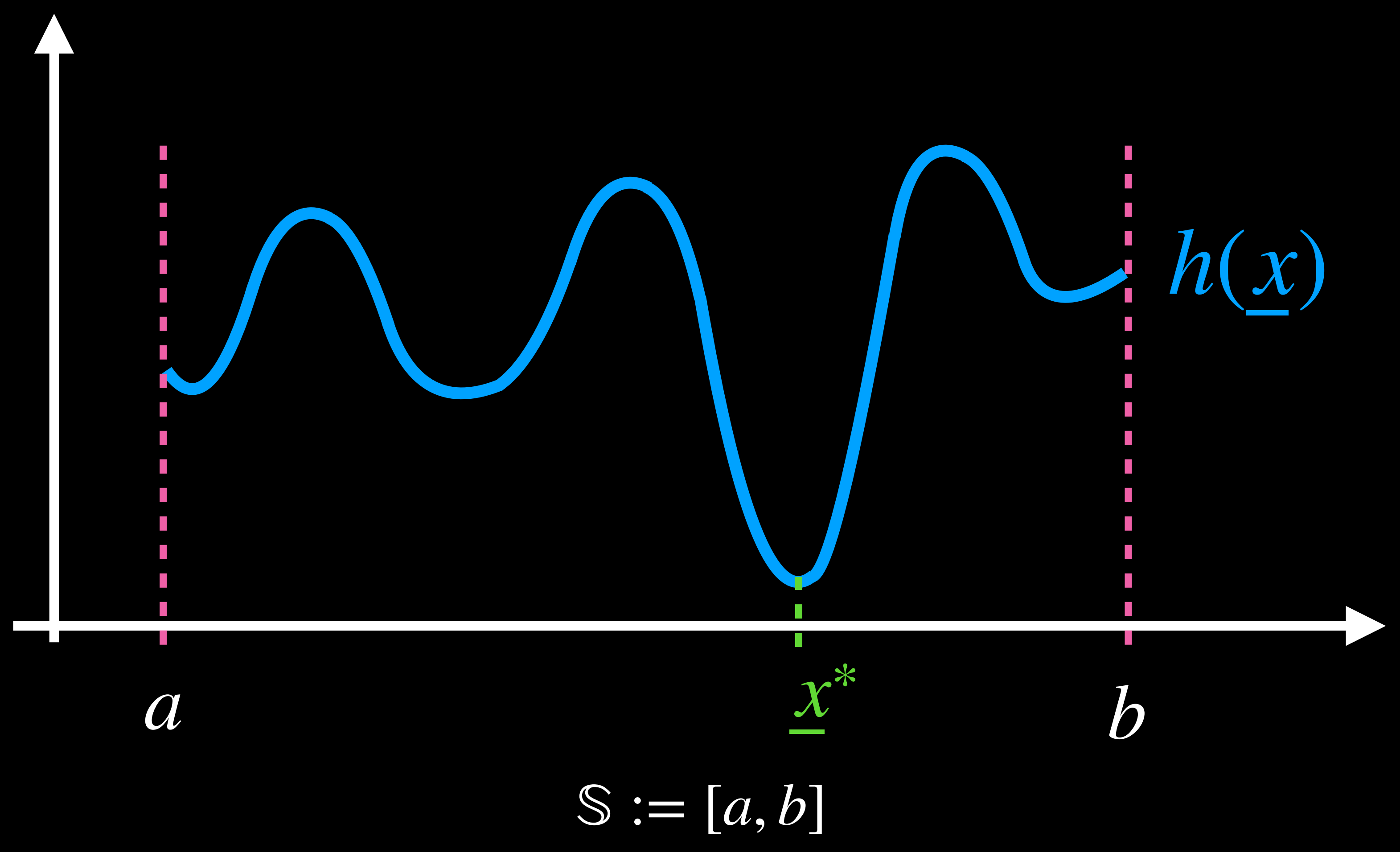
$h(x_1, x_2, x_3, \ldots, x_{100})$

function evaluation are expensive

infinite support



optimization



Goal:

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right) \qquad \left(\longleftrightarrow \qquad h(\underline{x}^*) = \min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right) \right)$$

Challenges:

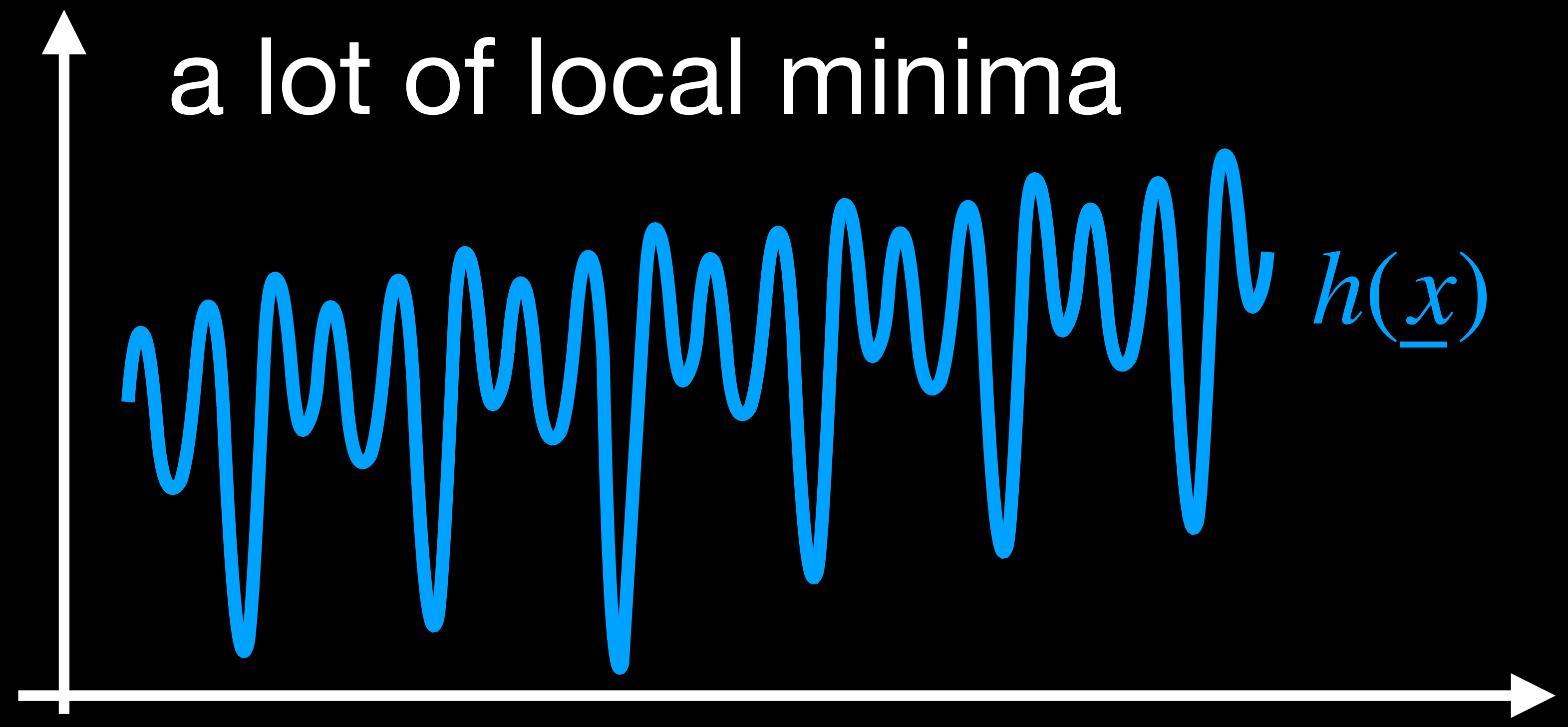
black box

high dimentional

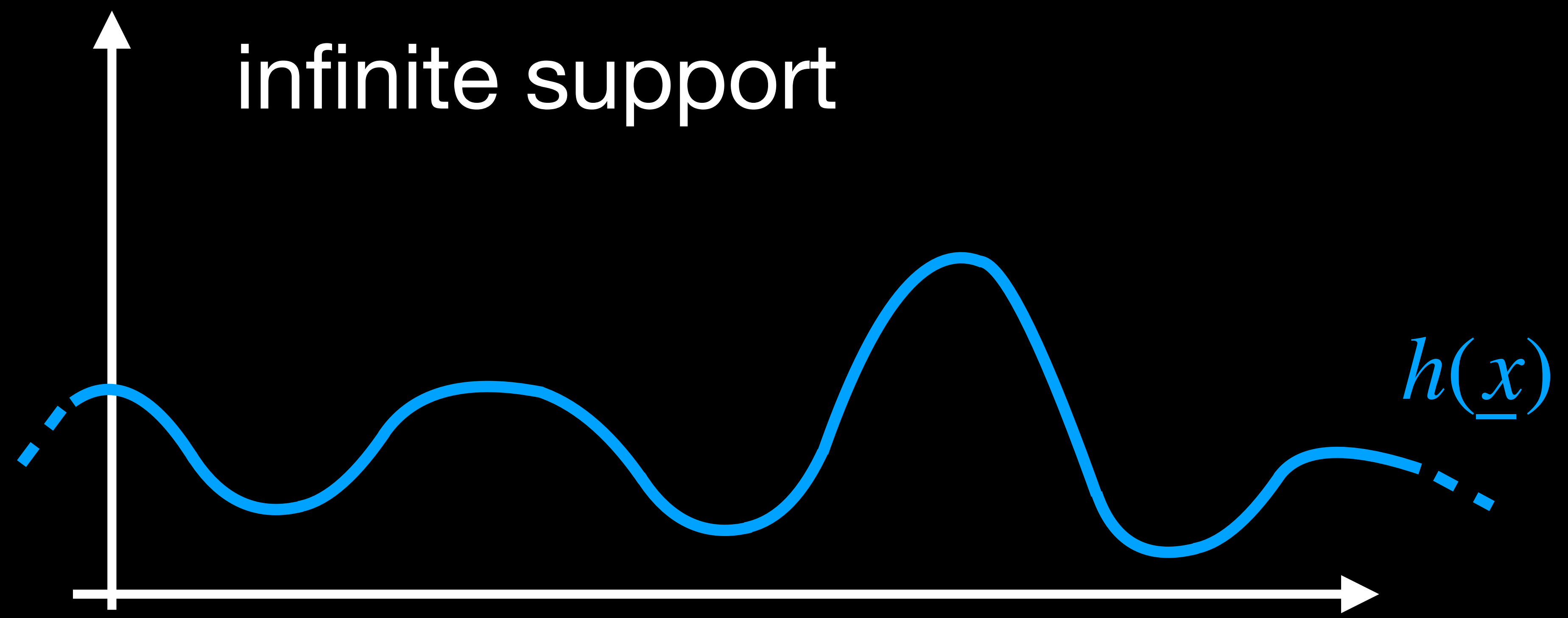
$h(x_1, x_2, x_3, \ldots, x_{100})$

function evaluation are expensive

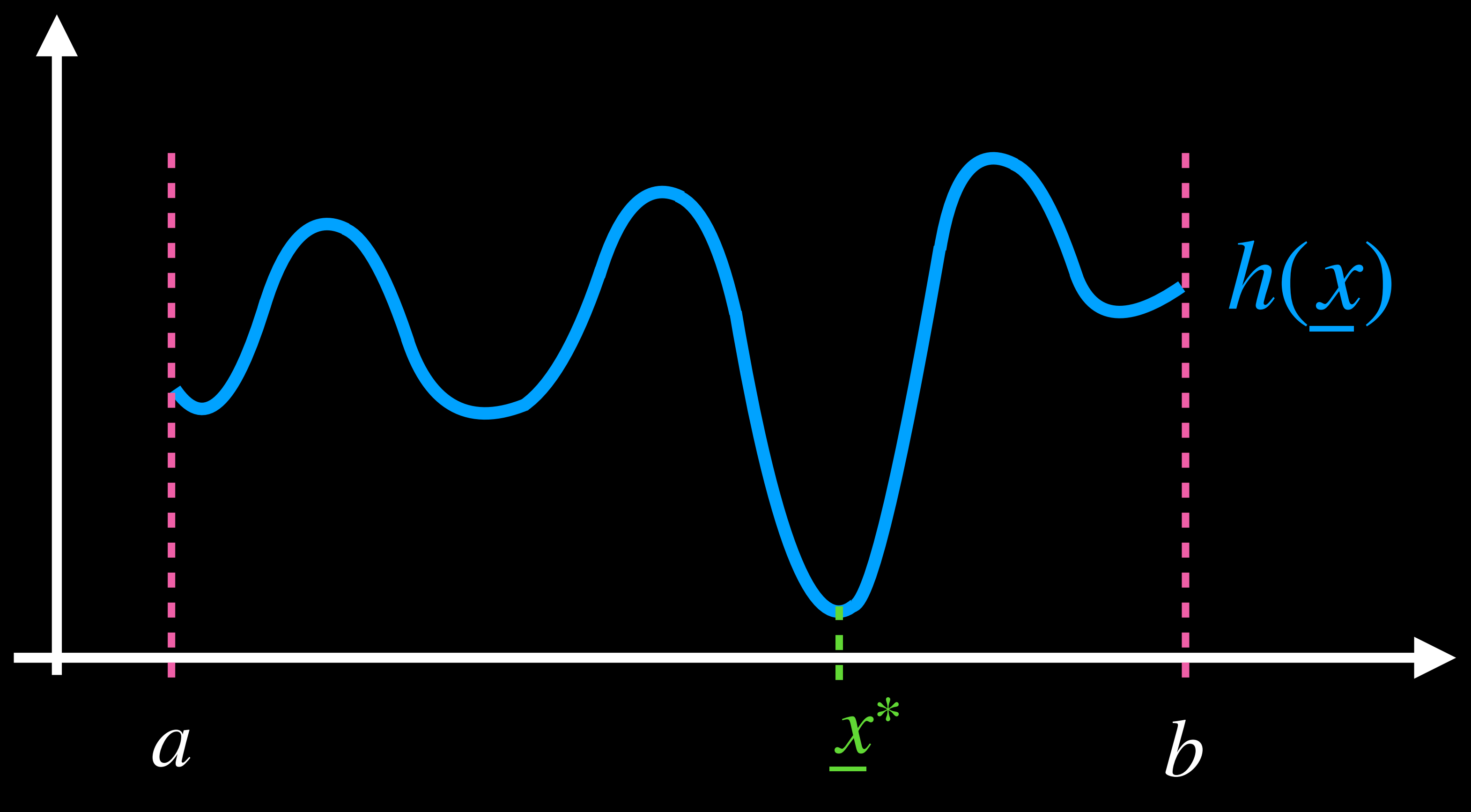
a lot of local minima



infinite support



optimization



$\mathbb{S} := [a, b]$

Goal:

$$\underline{x}^* := \underset{\underline{x} \in \mathbb{S}}{\arg\min} \left(h(\underline{x})\right) \qquad \left(\longleftrightarrow \qquad h(\underline{x}^*) = \min_{\underline{x} \in \mathbb{S}} \left(h(\underline{x})\right) \right)$$

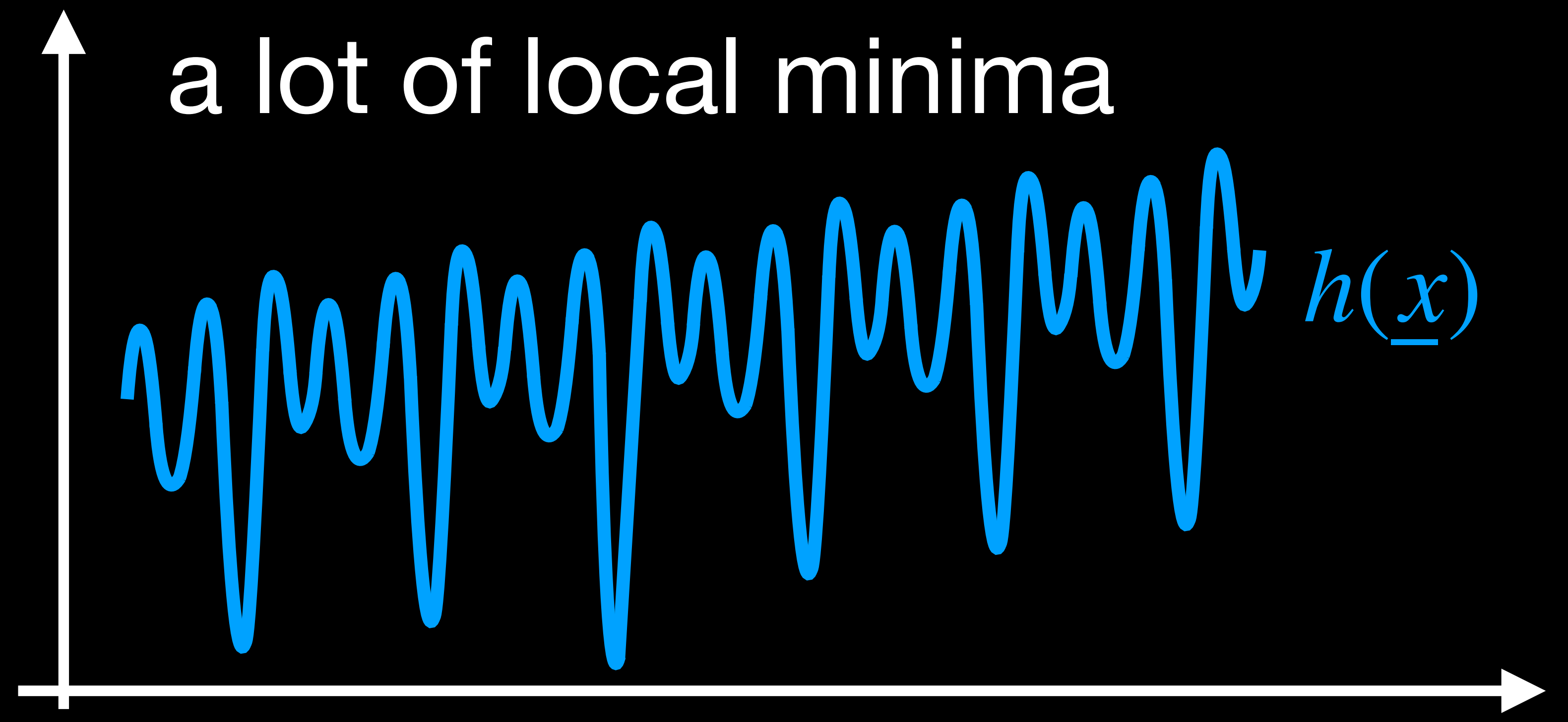
Challenges:

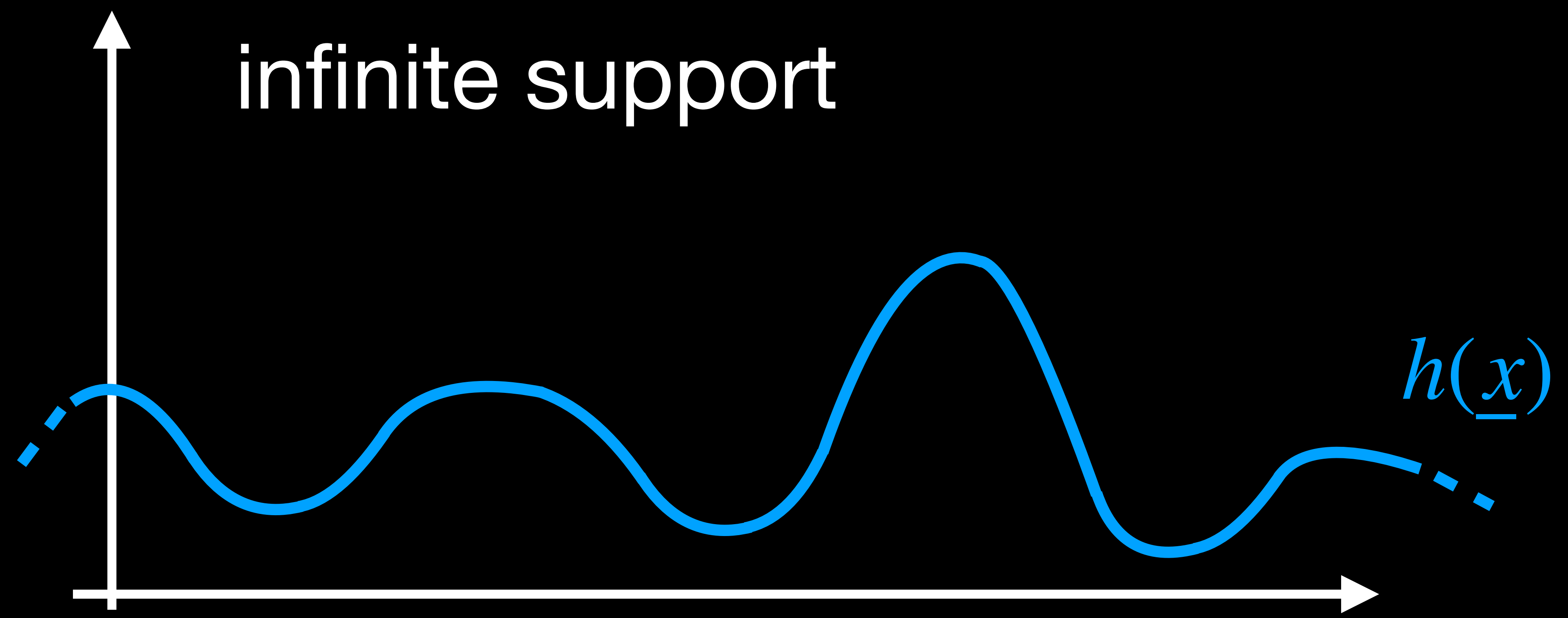
black box

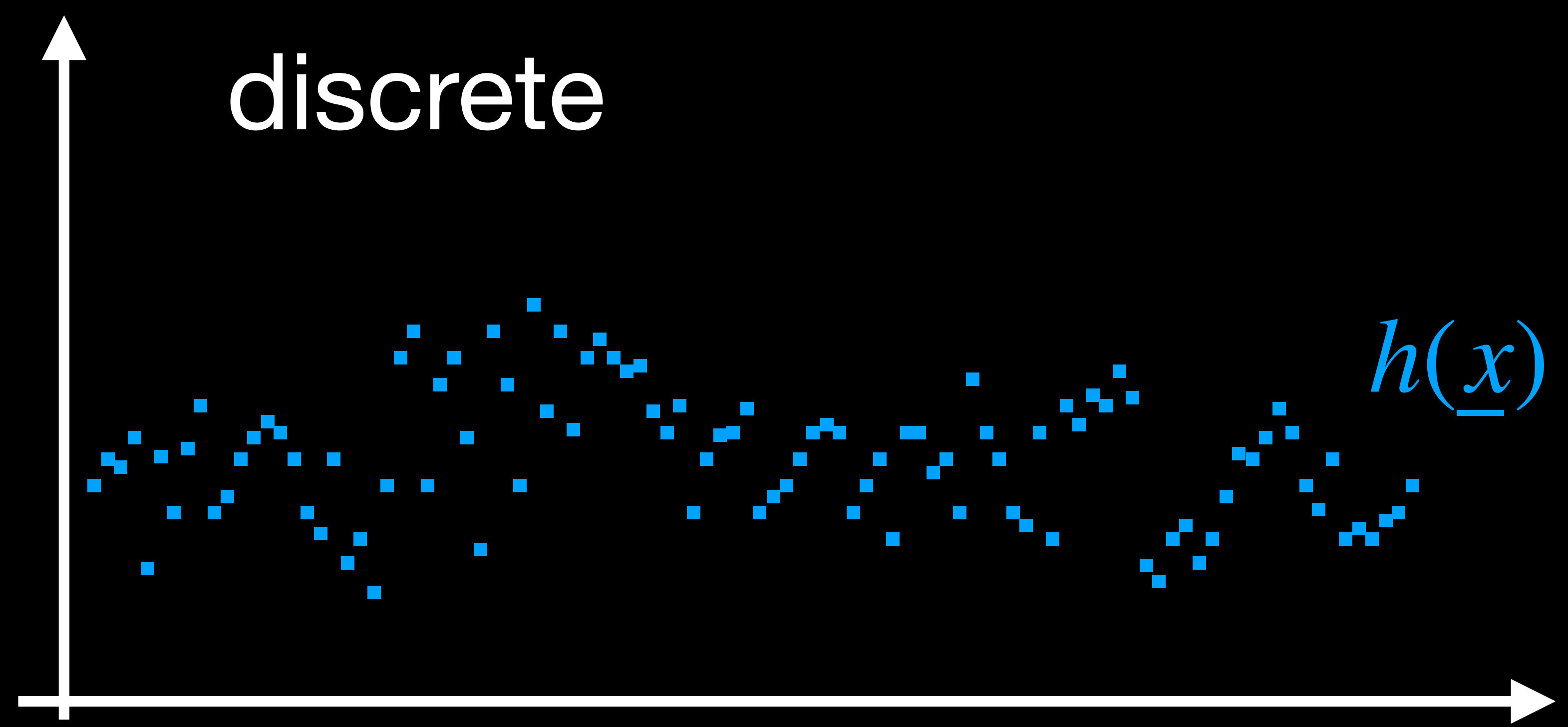
high dimentional

$h(x_1, x_2, x_3, \ldots, x_{100})$

function evaluation are expensive







algorithm 1: stochastic exploration

algorithm 1: stochastic exploration

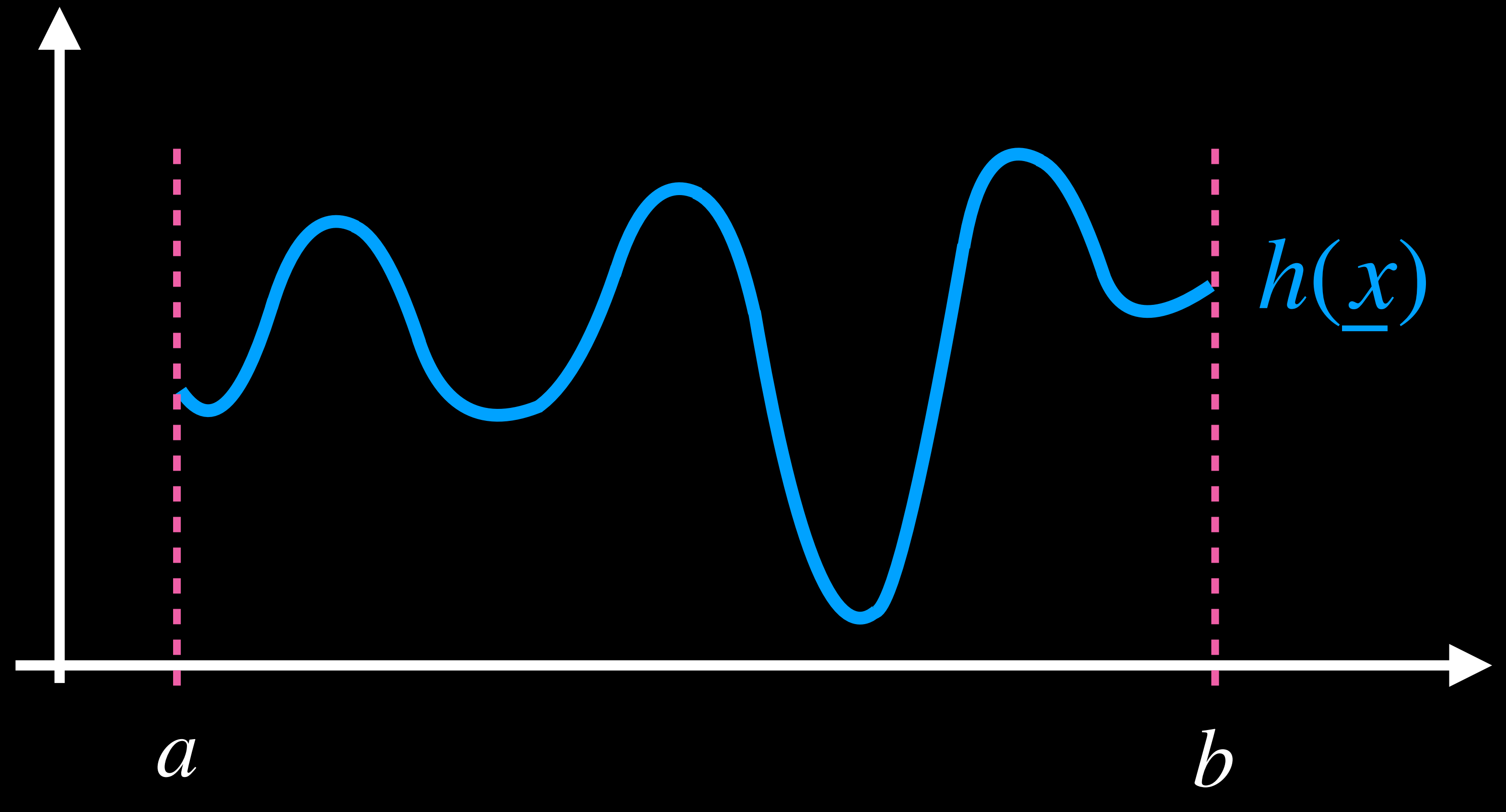
$\underline{x}^* :=$

algorithm 1: stochastic exploration

$$\underline{x}^* := \arg\min_{\underline{x} \in \mathbb{S}} \left(h(\underline{x}) \right)$$

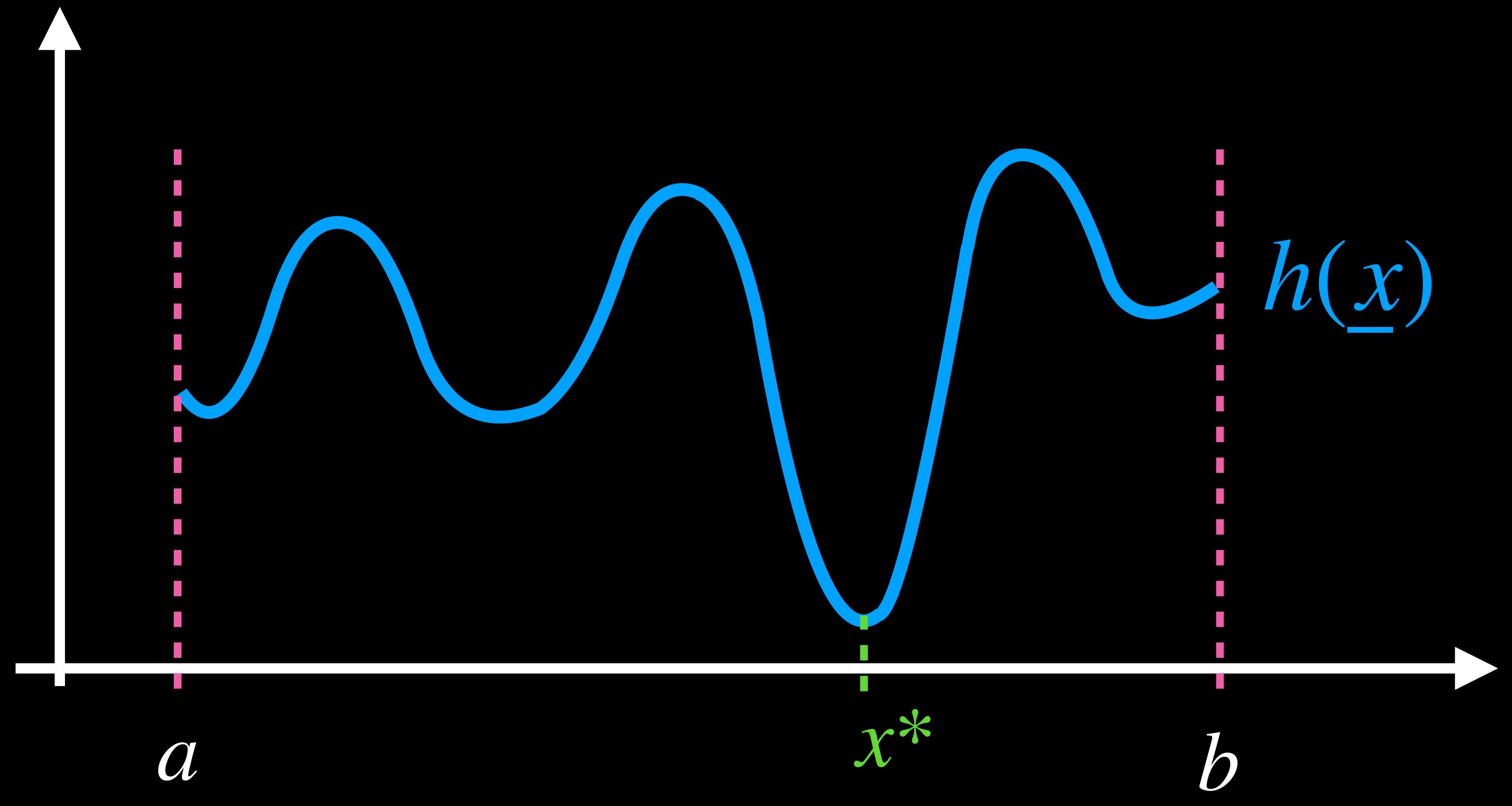
algorithm 1: stochastic exploration

$$\color{green}{\underline{x}^*} := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

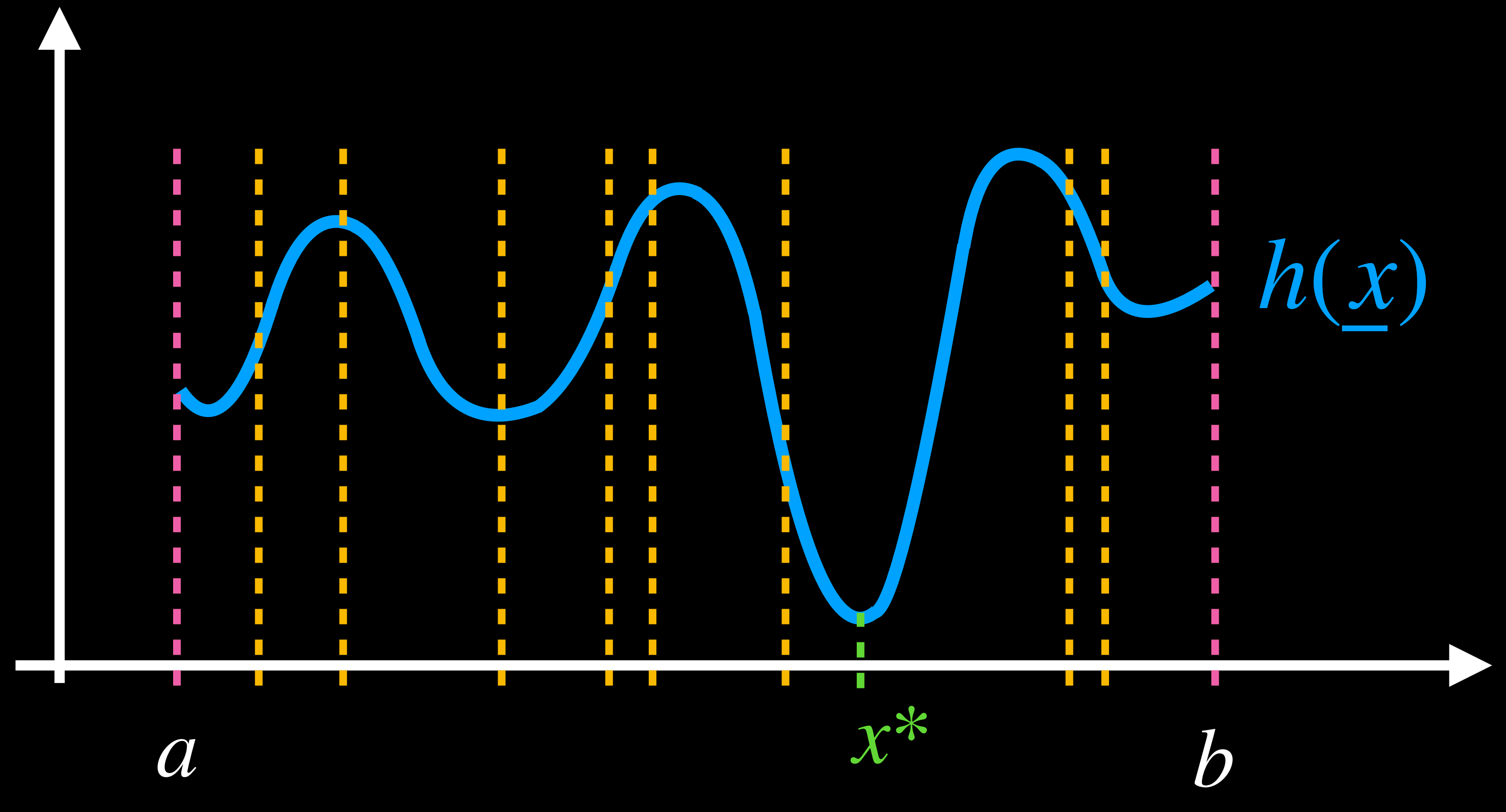


algorithm 1: stochastic exploration

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

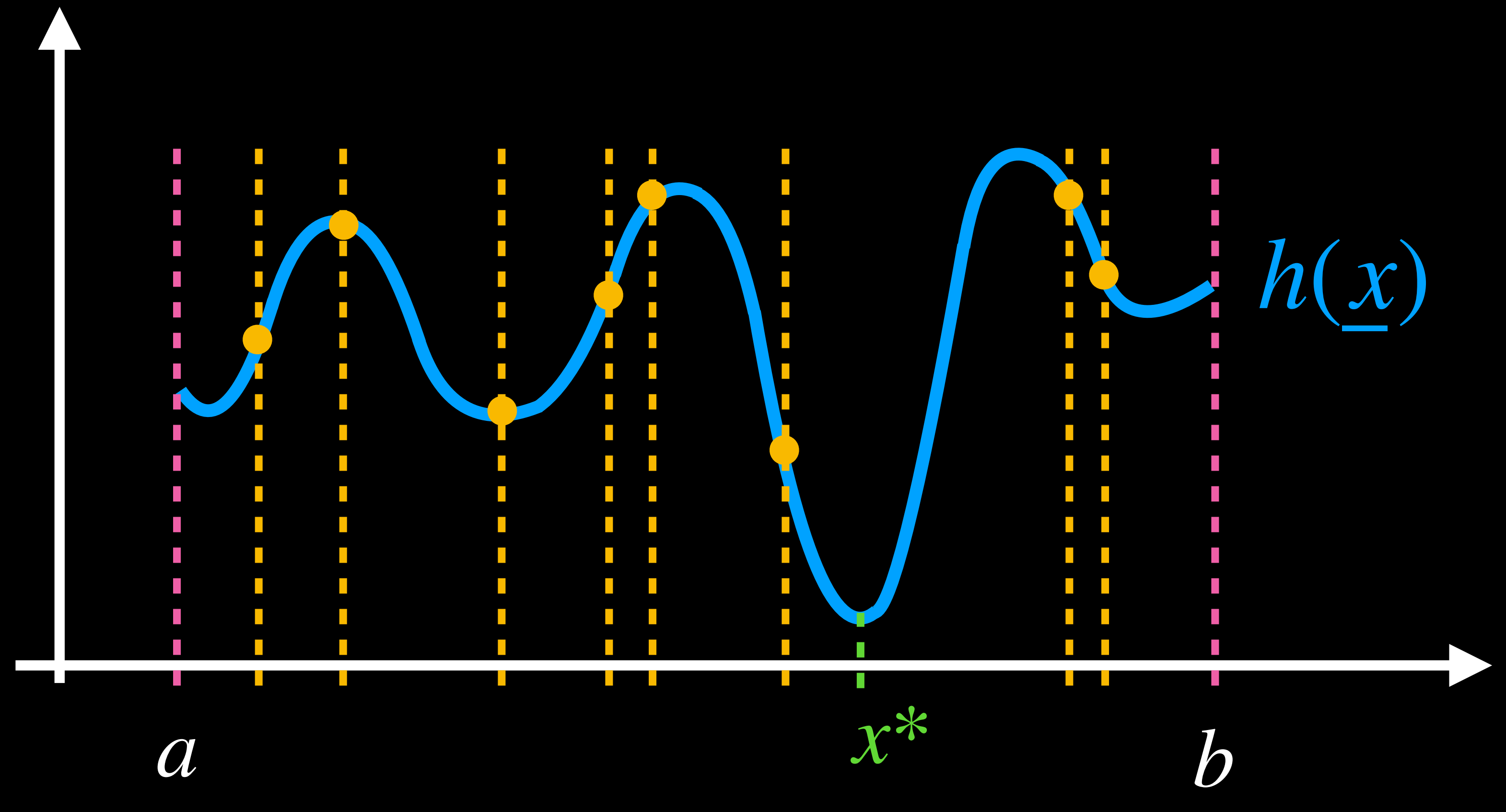


algorithm 1: stochastic exploration



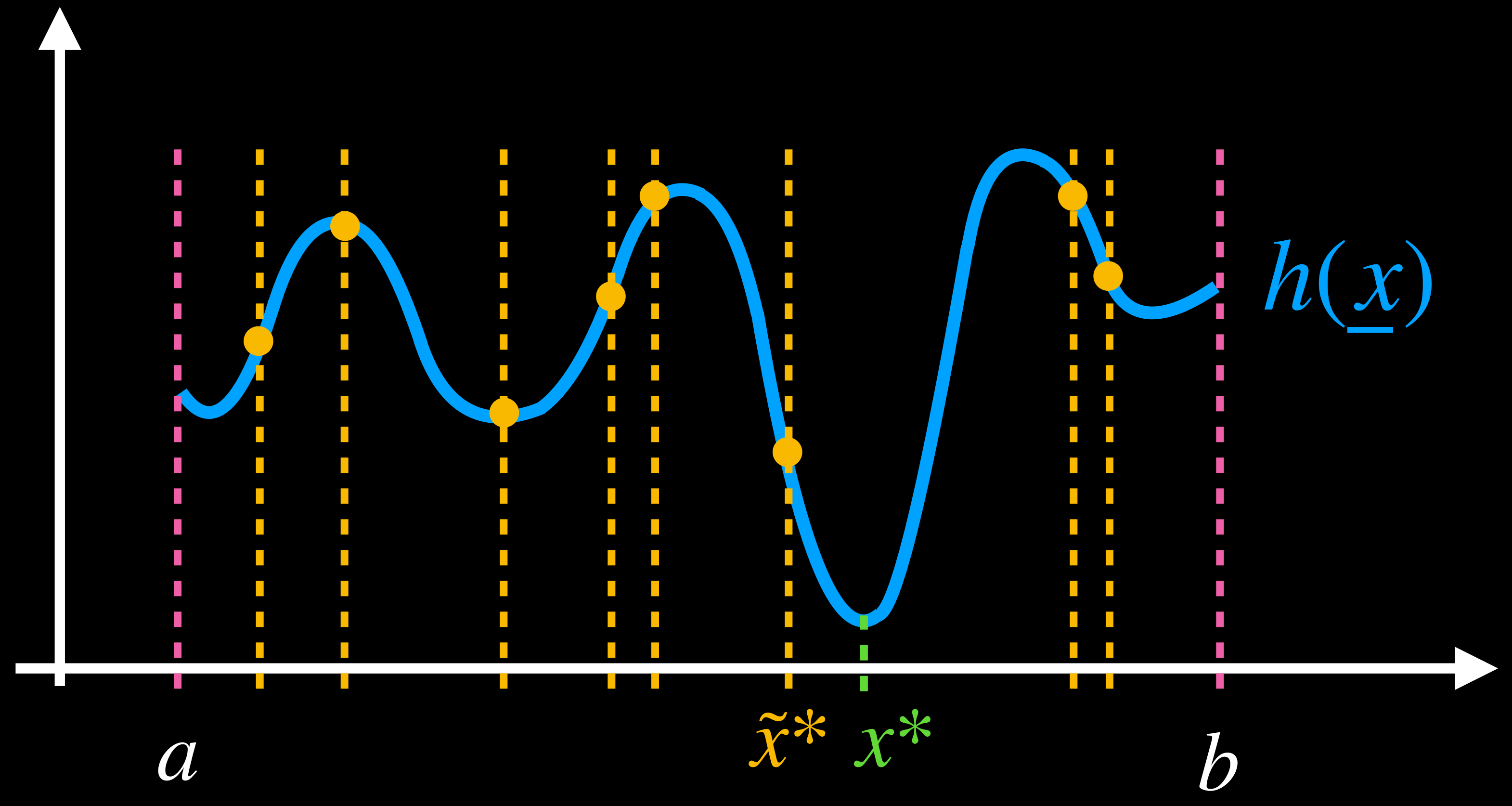
$$\underline{x}^* := \underset{\underline{x} \in \mathbb{S}}{\arg\min} \left(h(\underline{x}) \right)$$

algorithm 1: stochastic exploration



$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$

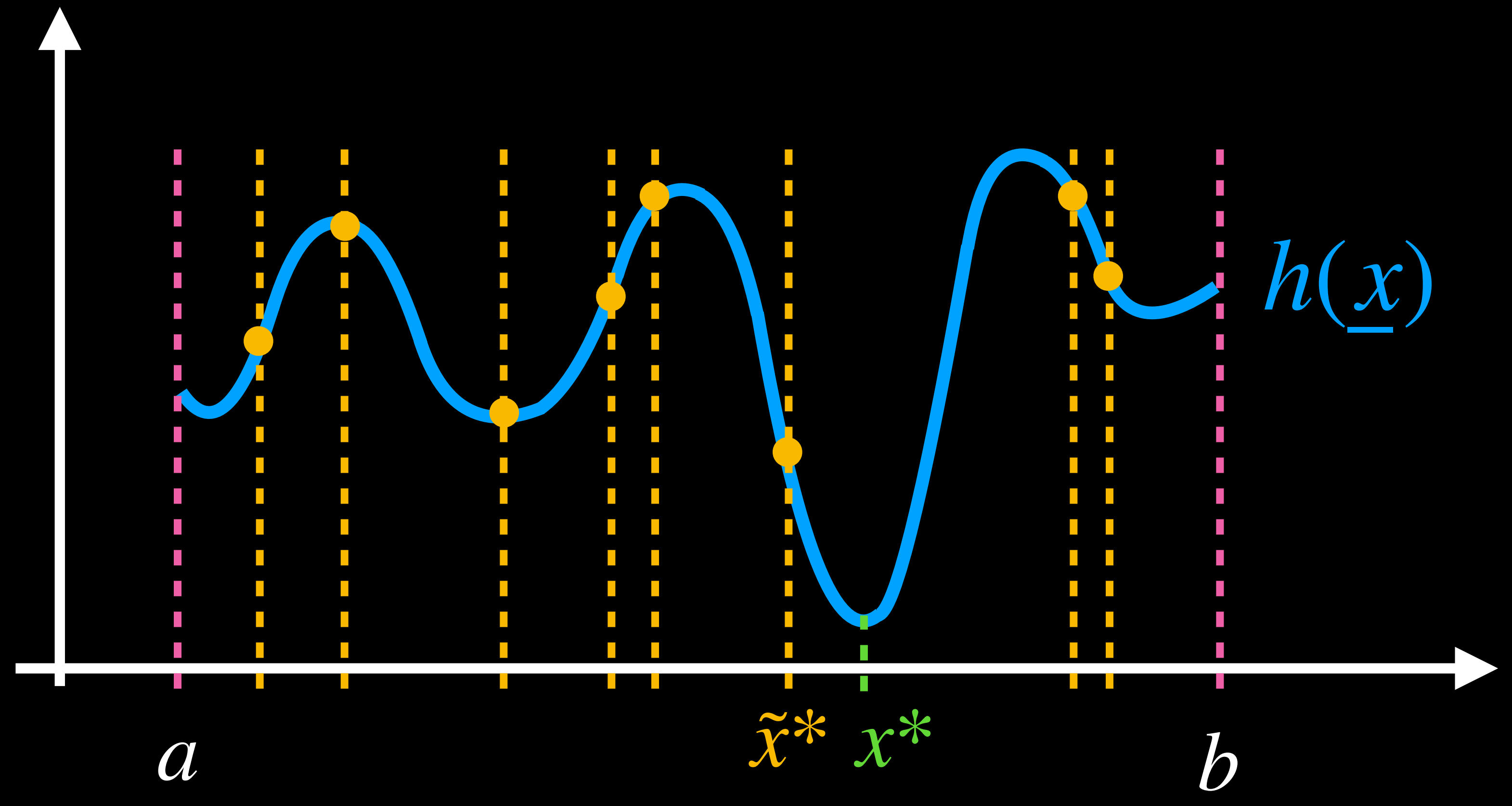
algorithm 1: stochastic exploration



$$\underline{x}^* := \underset{\underline{x} \in \mathbb{S}}{\arg\min} \left(h(\underline{x}) \right)$$

algorithm 1: stochastic exploration

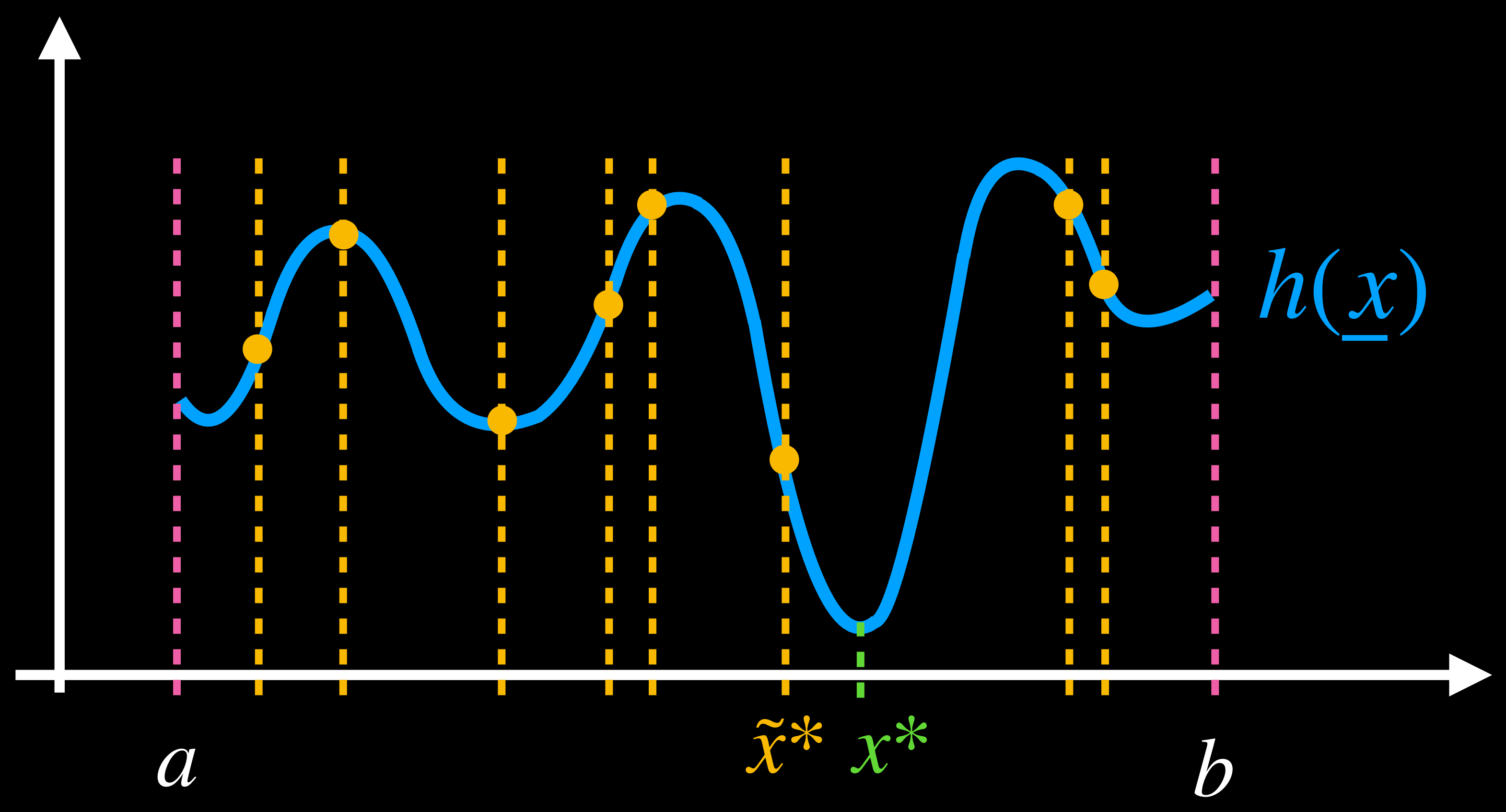
$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\,\big(h(\underline{x})\big) \approx \arg\min\Big(h(\underline{X}_1),\ldots,h(\underline{X}_n)\Big)$$



algorithm 1: stochastic exploration

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right) \approx \arg\min \left(h(\underline{X}_1), \ldots, h(\underline{X}_n)\right)$$

$$\underline{X}_t : \Omega \to \mathbb{S} \subseteq \mathbb{R}^d$$

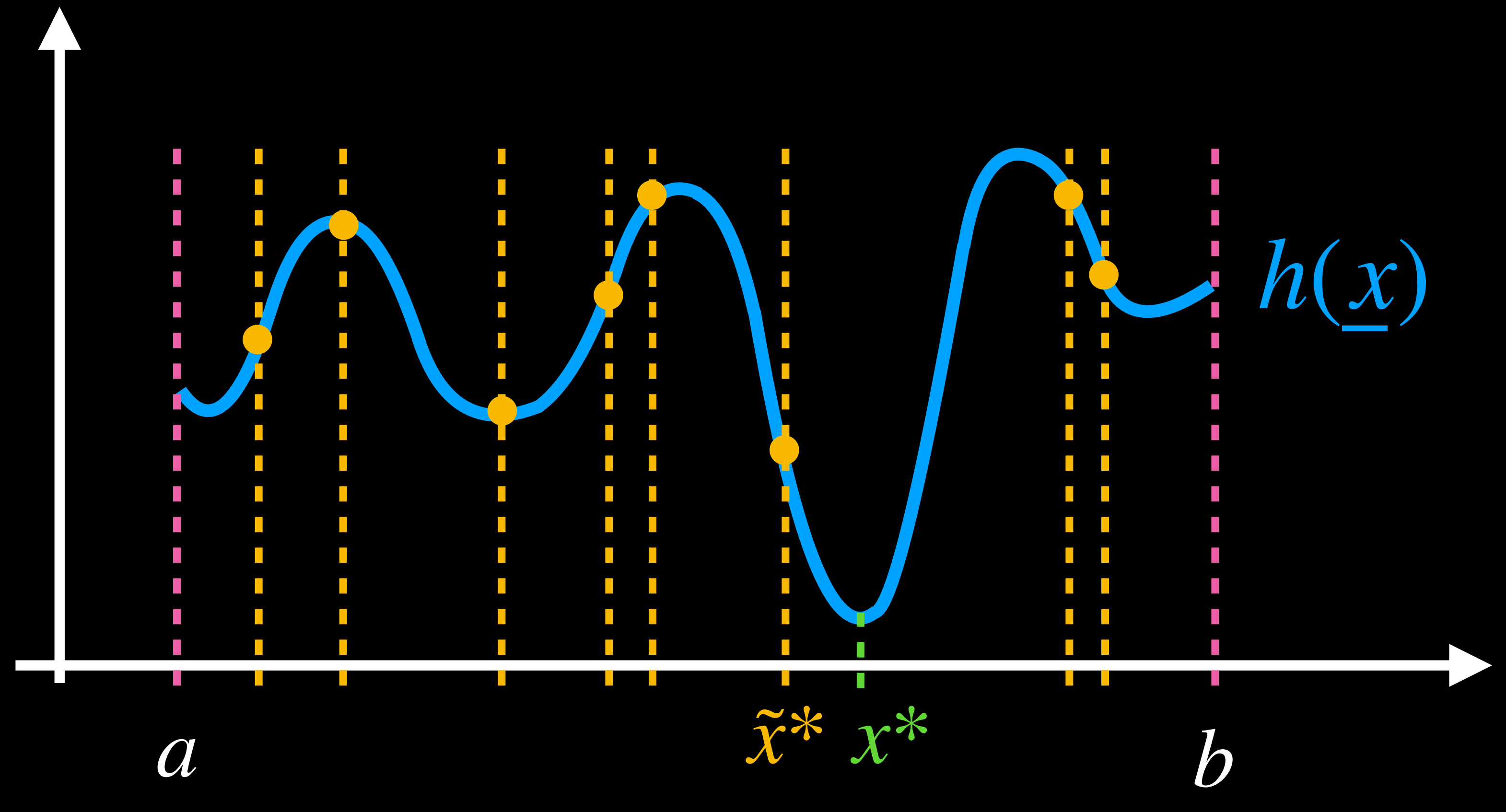


algorithm 1: stochastic exploration

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right) \approx \arg\min \left(h(\underline{X}_1), \ldots, h(\underline{X}_n)\right)$$

$$\underline{X}_t : \Omega \to \mathbb{S} \subseteq \mathbb{R}^d$$

$$\underline{X}_t \sim \mathcal{U}(\mathbb{S})$$

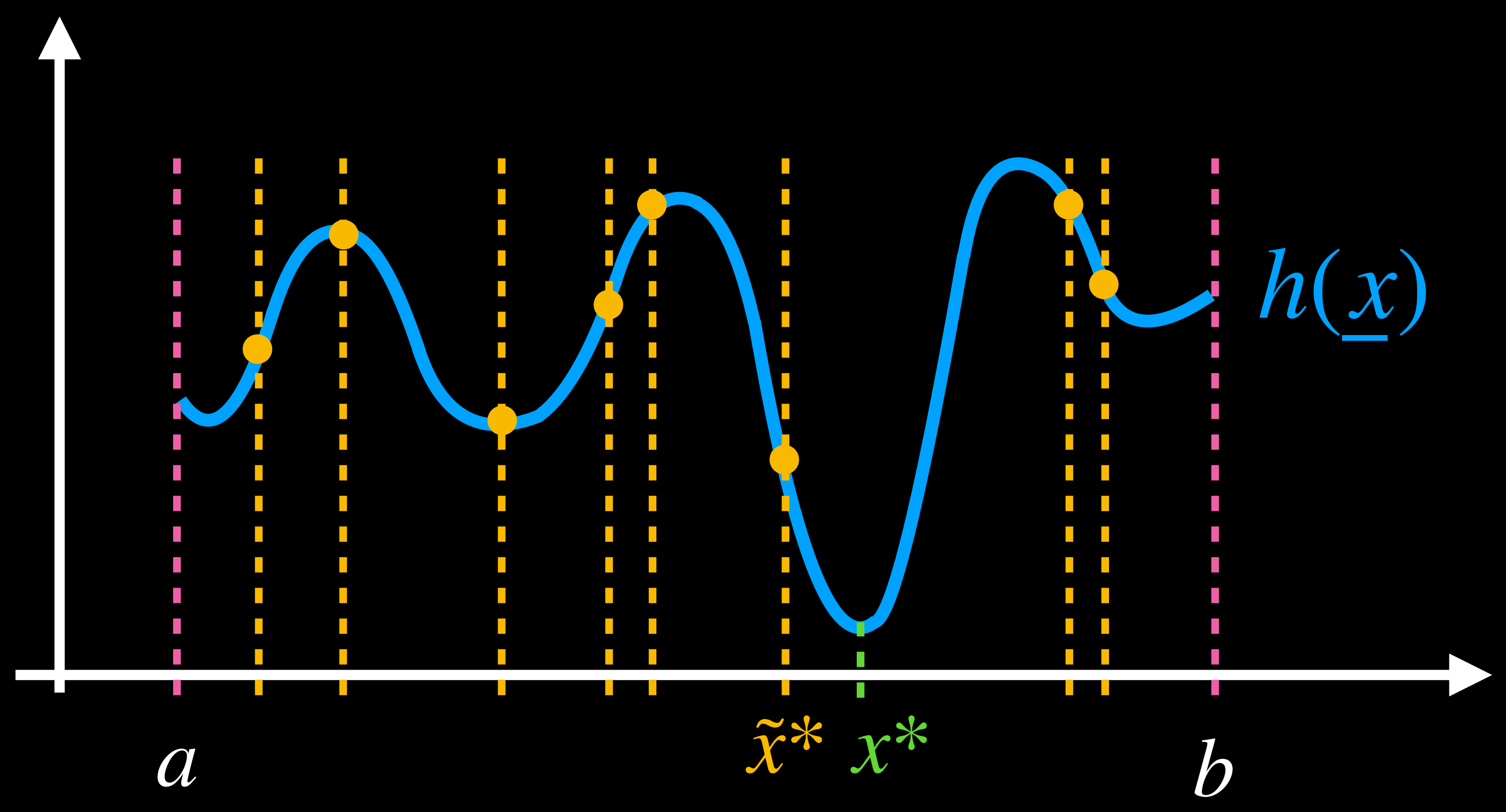


algorithm 1: stochastic exploration

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \big(h(\underline{x})\big) \approx \arg\min \Big(h(\underline{X}_1), \ldots, h(\underline{X}_n)\Big)$$

$$\underline{X}_t : \Omega \to \mathbb{S} \subseteq \mathbb{R}^d$$

$$\underline{X}_t \sim \mathcal{U}(\mathbb{S})$$



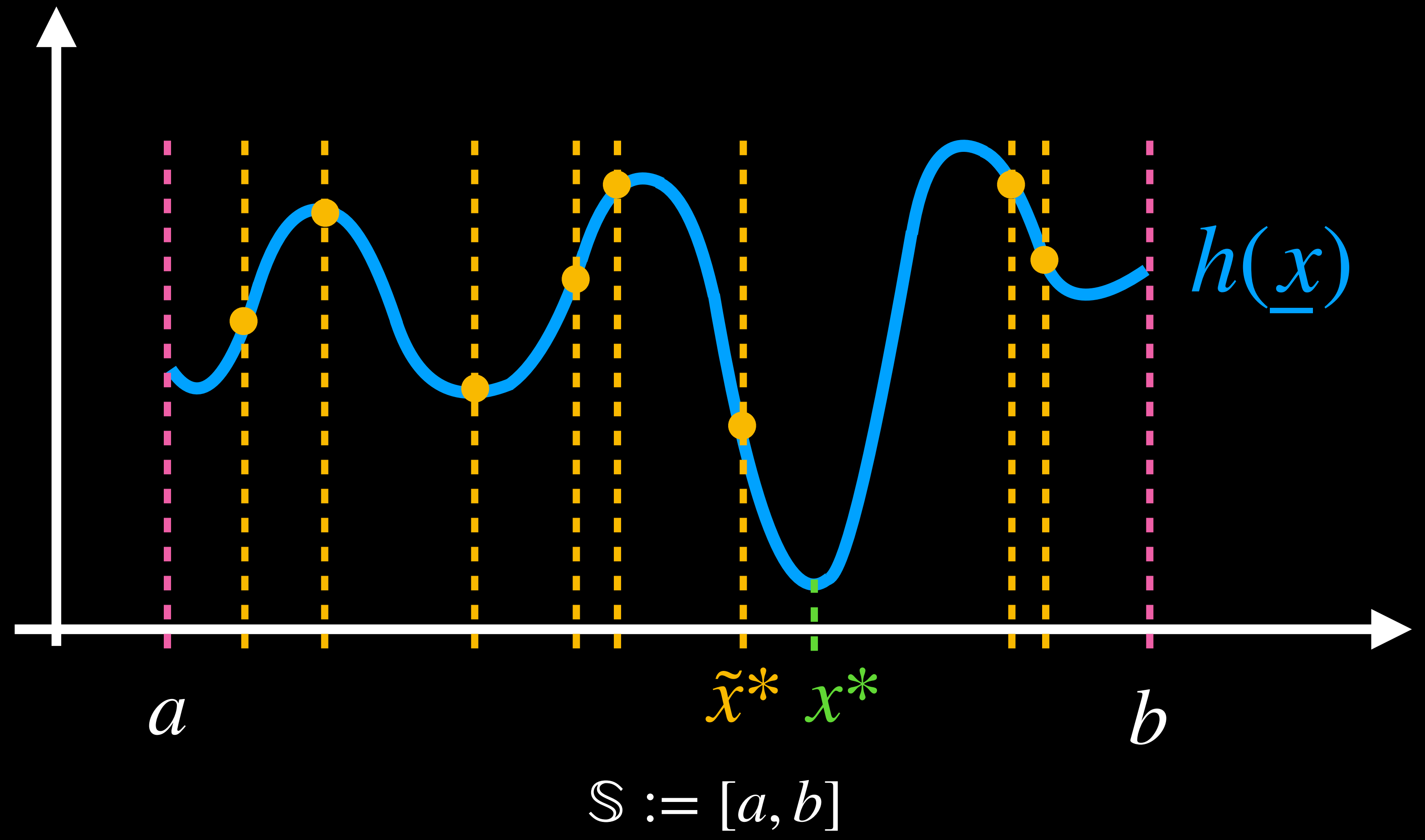
$\mathbb{S} := [a, b]$

algorithm 1: stochastic exploration

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\,\big(h(\underline{x})\big) \approx \arg\min\Big(h(\underline{X}_1),\ldots,h(\underline{X}_n)\Big) =: \underline{\tilde{X}}_n^*$$

$$\underline{X}_t : \Omega \to \mathbb{S} \subseteq \mathbb{R}^d$$

$$\underline{X}_t \sim \mathcal{U}(\mathbb{S})$$

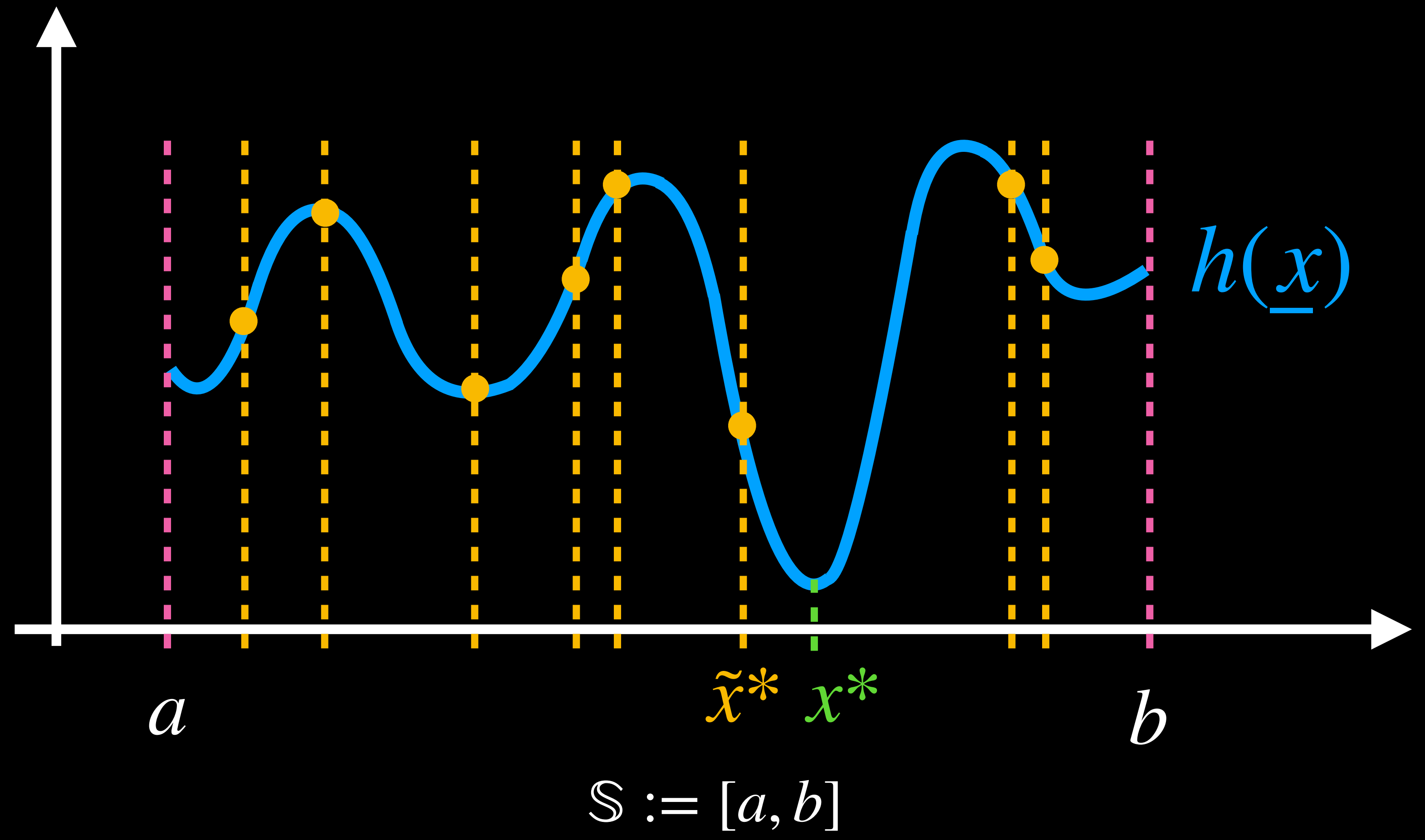


algorithm 1: stochastic exploration

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right) \approx \arg\min \left(h(\underline{X}_1), \ldots, h(\underline{X}_n)\right) =: \tilde{\underline{X}}_n^*$$

$$\underline{X}_t : \Omega \to \mathbb{S} \subseteq \mathbb{R}^d$$

$$\underline{X}_t \sim \mathcal{U}(\mathbb{S})$$



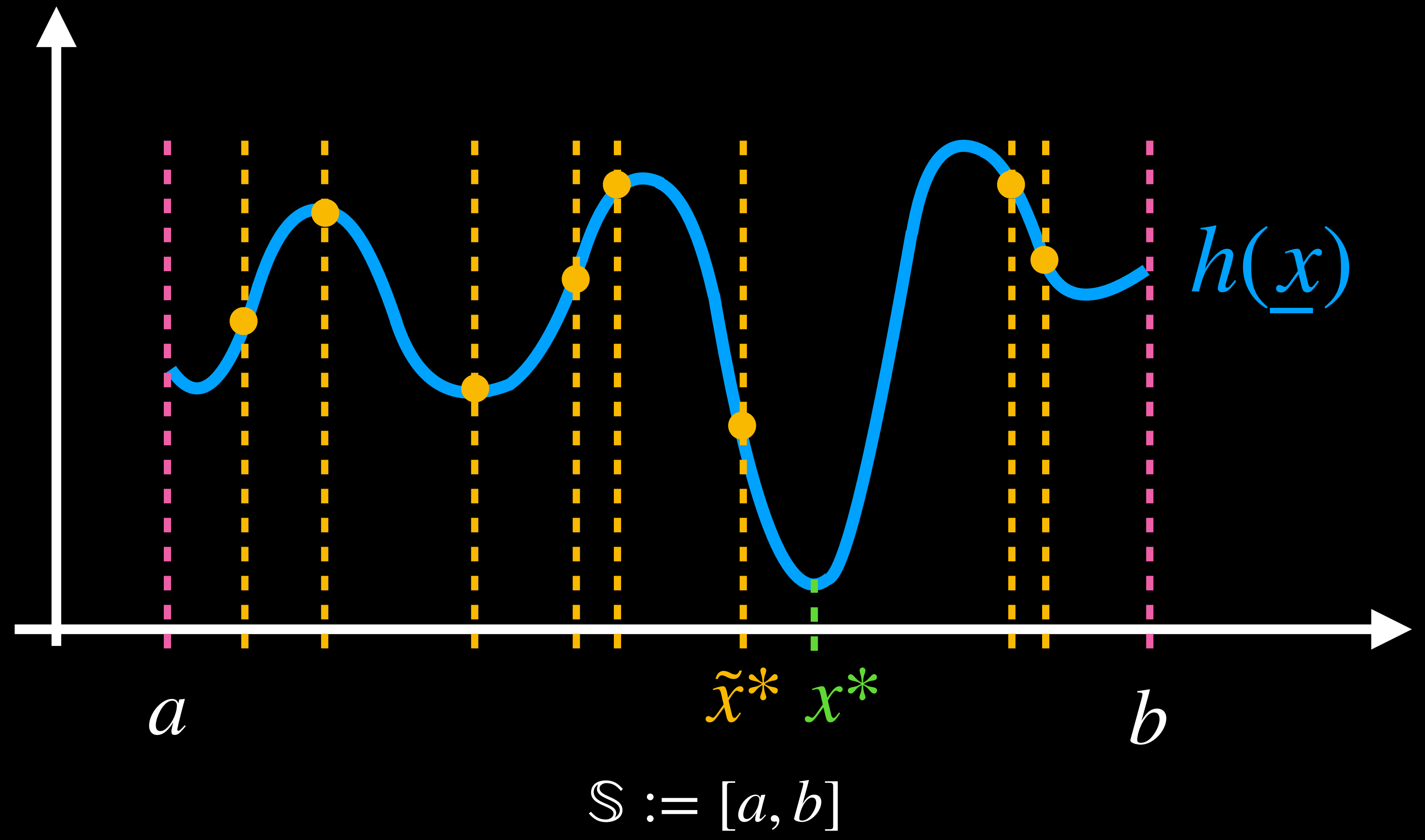
global convergence:

algorithm 1: stochastic exploration

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right) \approx \arg\min \left(h(\underline{X}_1), \ldots, h(\underline{X}_n)\right) =: \tilde{\underline{X}}^*_n$$

$$\underline{X}_t : \Omega \to \mathbb{S} \subseteq \mathbb{R}^d$$

$$\underline{X}_t \sim \mathcal{U}(\mathbb{S})$$



global convergence:

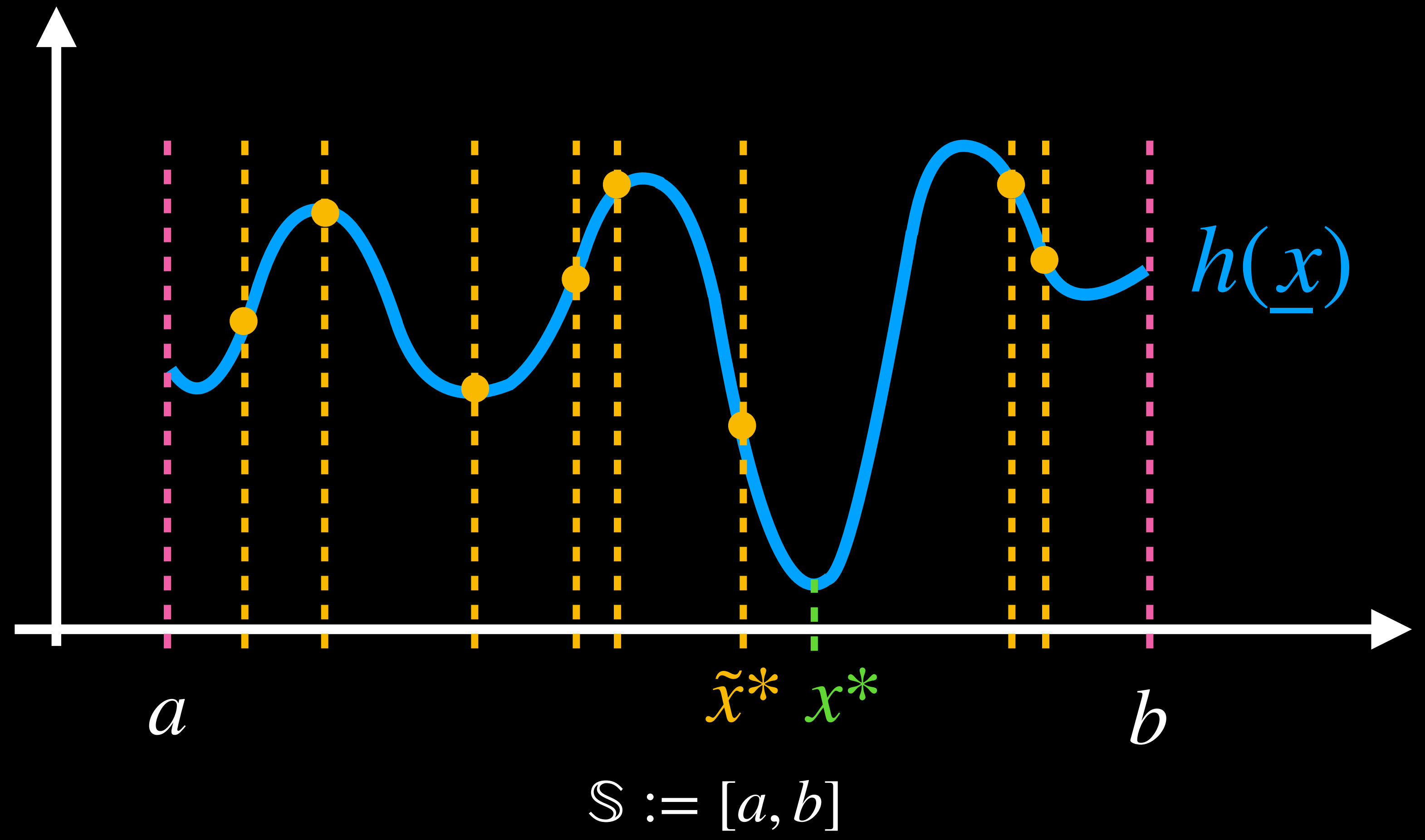
$\underline{x}^*$

algorithm 1: stochastic exploration

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right) \approx \arg\min \left(h(\underline{X}_1), \ldots, h(\underline{X}_n)\right) =: \tilde{\underline{X}}^*_n$$

$$\underline{X}_t : \Omega \to \mathbb{S} \subseteq \mathbb{R}^d$$

$$\underline{X}_t \sim \mathcal{U}(\mathbb{S})$$



global convergence:

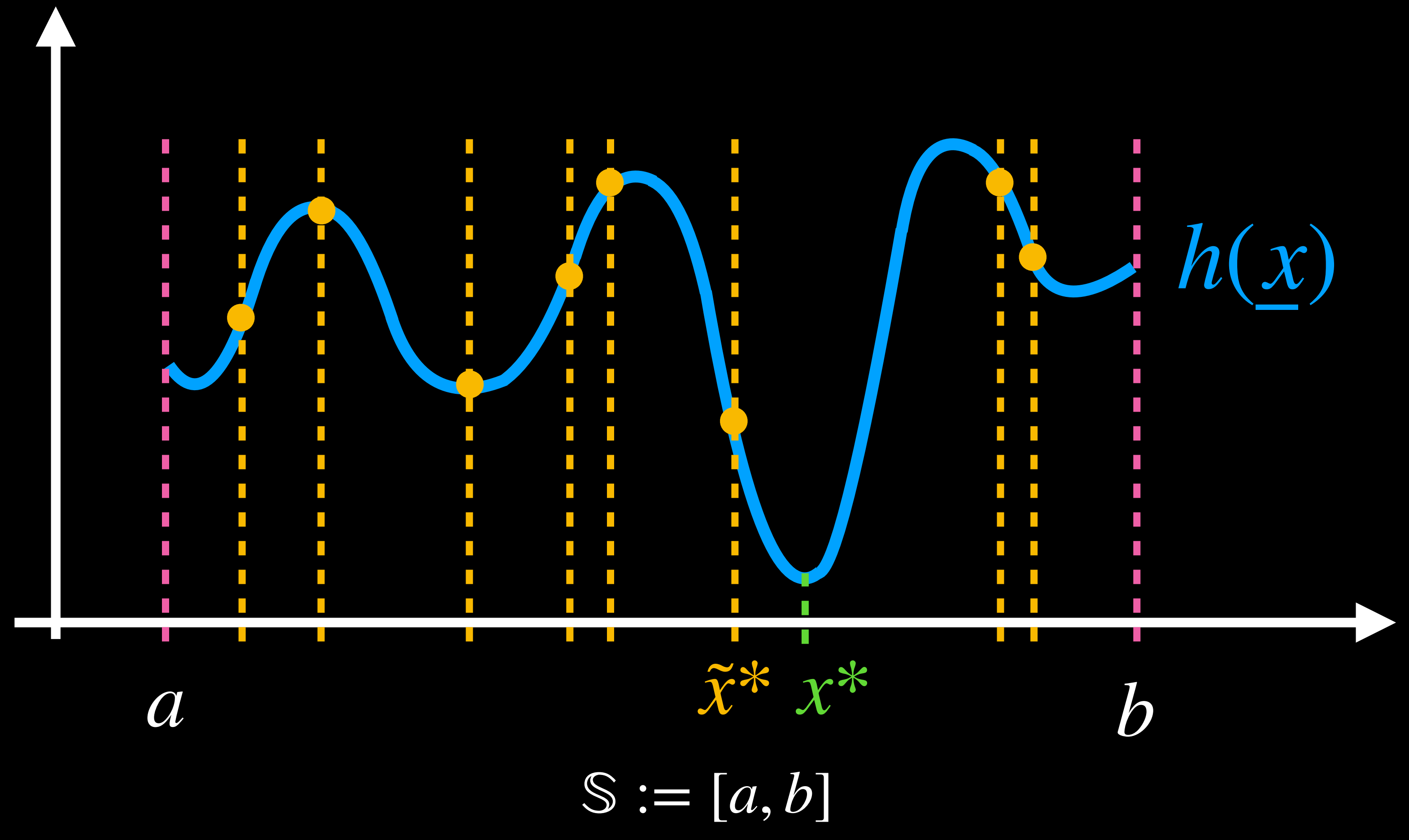
$$\underline{x}^* = \lim_{n\to\infty} \tilde{\underline{X}}^*_n$$

algorithm 1: stochastic exploration

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right) \approx \arg\min \left(h(\underline{X}_1), \ldots, h(\underline{X}_n)\right) =: \underline{\tilde{X}}_n^*$$

$$\underline{X}_t : \Omega \to \mathbb{S} \subseteq \mathbb{R}^d$$

$$\underline{X}_t \sim \mathcal{U}(\mathbb{S})$$

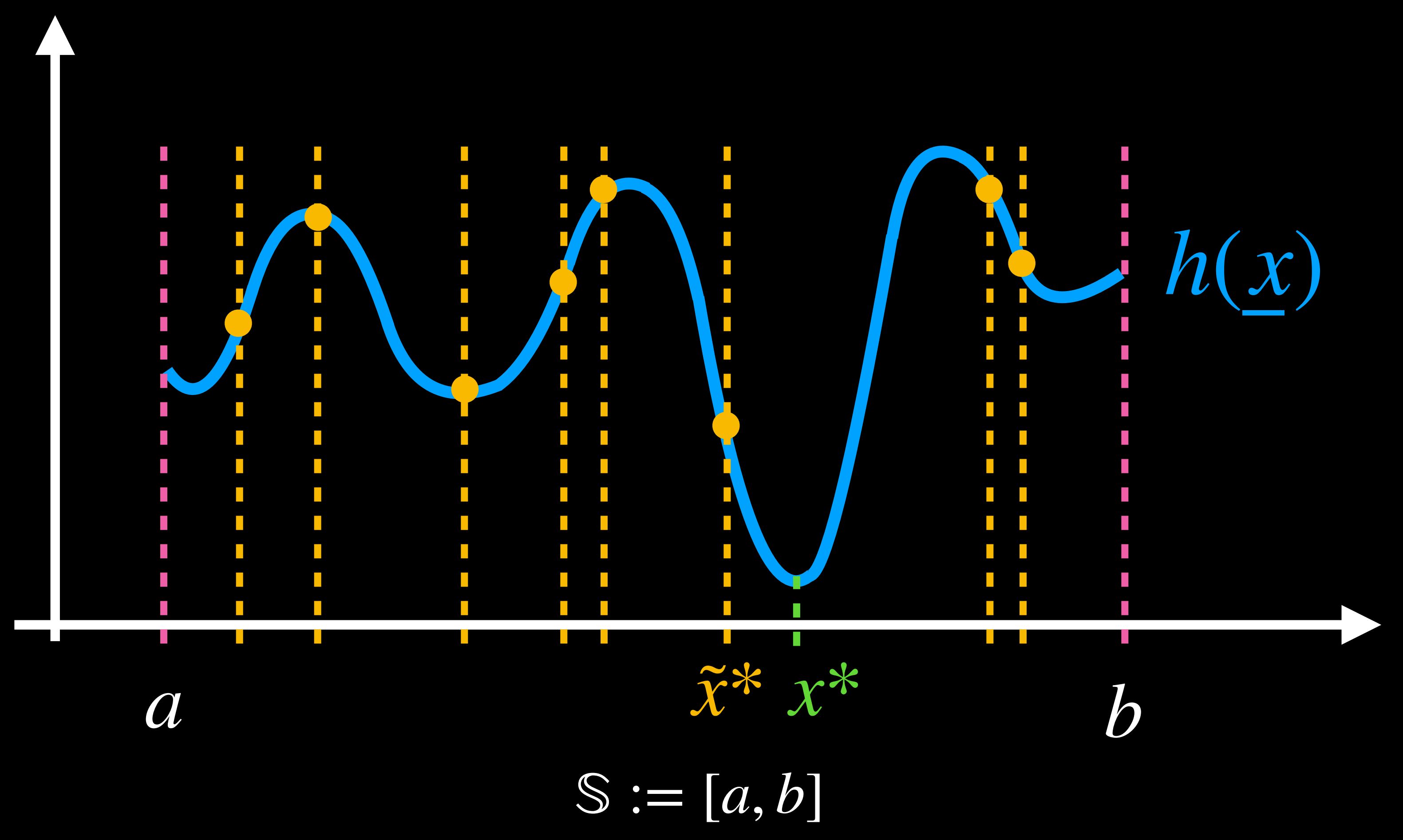


global convergence:

$$\underline{x}^* = \lim_{n\to\infty} \underline{\tilde{X}}_n^*$$

1d:

algorithm 1: stochastic exploration



$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \big(h(\underline{x})\big) \approx \arg\min \Big(h(\underline{X}_1), \ldots, h(\underline{X}_n)\Big) =: \underline{\tilde{X}}_n^*$$

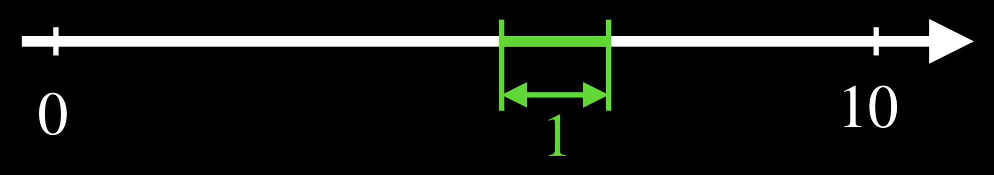
$$\underline{X}_t : \Omega \to \mathbb{S} \subseteq \mathbb{R}^d$$

$$\underline{X}_t \sim \mathscr{U}(\mathbb{S})$$

global convergence:

$$\underline{x}^* = \lim_{n\to\infty} \underline{\tilde{X}}_n^*$$

1d:

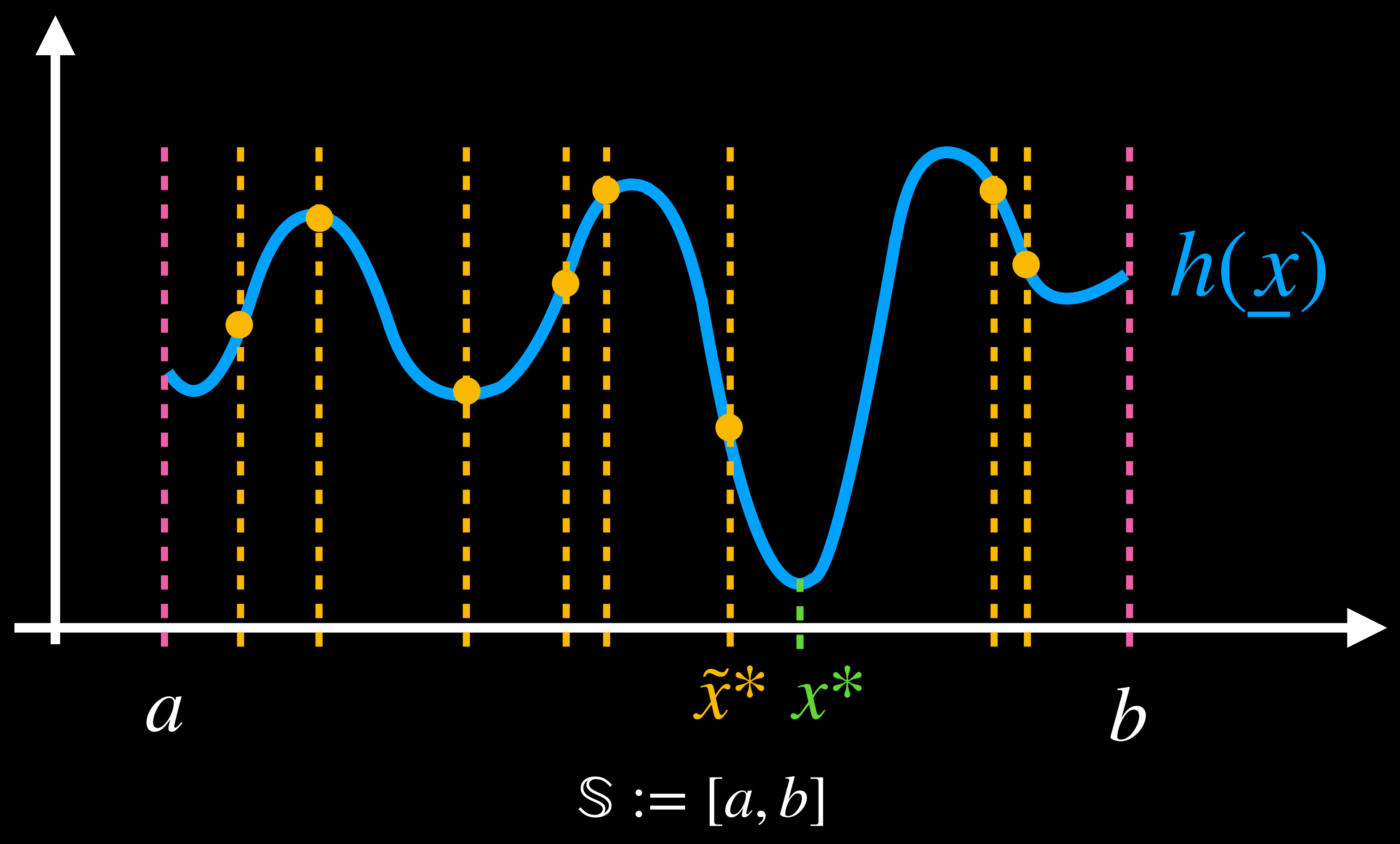


algorithm 1: stochastic exploration

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \big(h(\underline{x})\big) \approx \arg\min \Big(h(\underline{X}_1),\ldots,h(\underline{X}_n)\Big) =: \underline{\tilde{X}}_n^*$$

$$\underline{X}_t : \Omega \to \mathbb{S} \subseteq \mathbb{R}^d$$

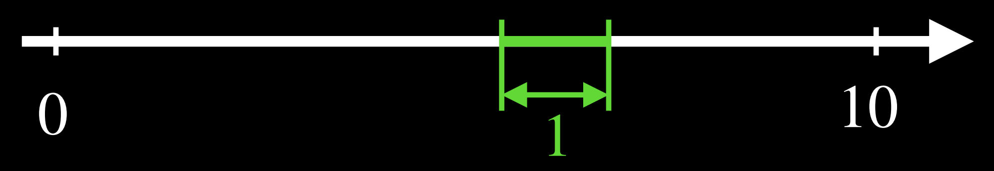
$$\underline{X}_t \sim \mathcal{U}(\mathbb{S})$$



global convergence:

$$\underline{x}^* = \lim_{n\to\infty} \underline{\tilde{X}}_n^*$$

1d:



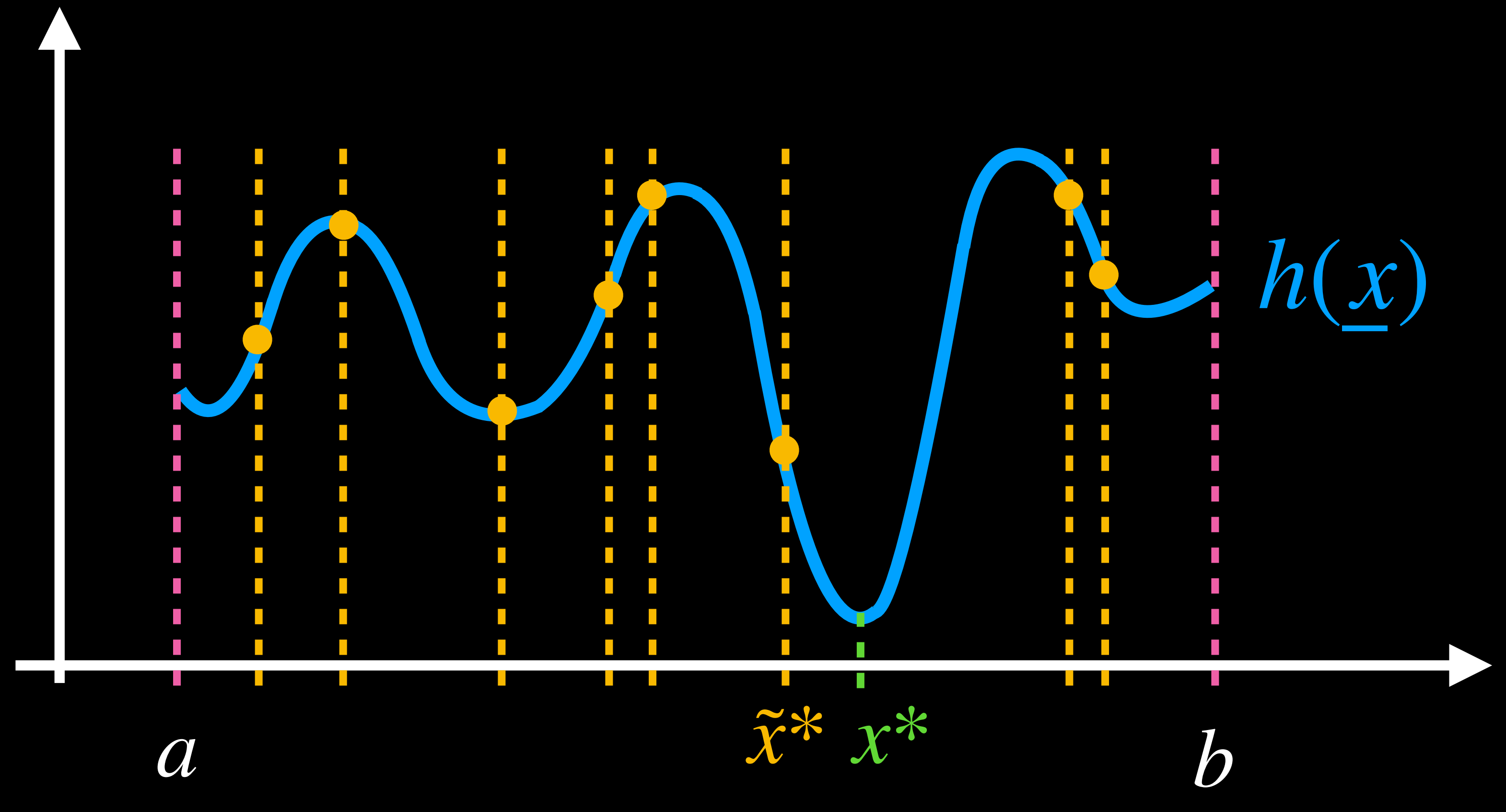
$P(X \in \text{—}) =$

algorithm 1: stochastic exploration

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \big(h(\underline{x})\big) \approx \arg\min \Big(h(\underline{X}_1), \ldots, h(\underline{X}_n)\Big) =: \underline{\tilde{X}}_n^*$$

$$\underline{X}_t : \Omega \to \mathbb{S} \subseteq \mathbb{R}^d$$

$$\underline{X}_t \sim \mathcal{U}(\mathbb{S})$$

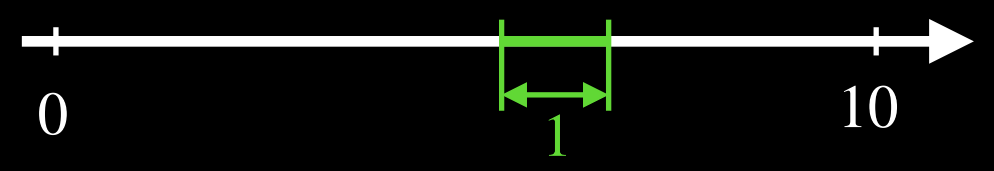


$\mathbb{S} := [a, b]$

global convergence:

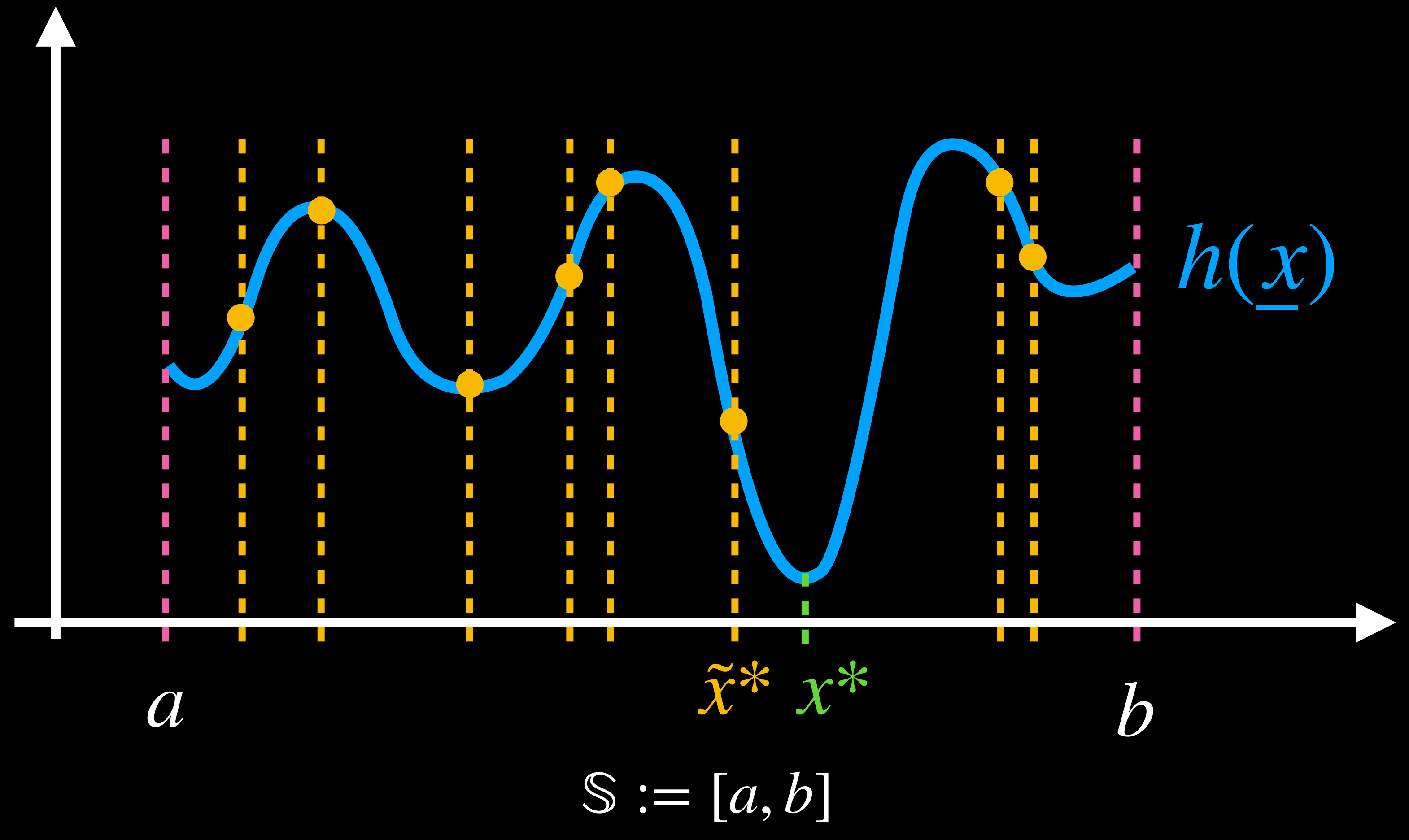
$$\underline{x}^* = \lim_{n\to\infty} \underline{\tilde{X}}_n^*$$

1d:



$$P(X \in \text{—}) = \frac{1}{10}$$

algorithm 1: stochastic exploration



$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right) \approx \arg\min\left(h(\underline{X}_1), \ldots, h(\underline{X}_n)\right) =: \tilde{\underline{X}}_n^*$$

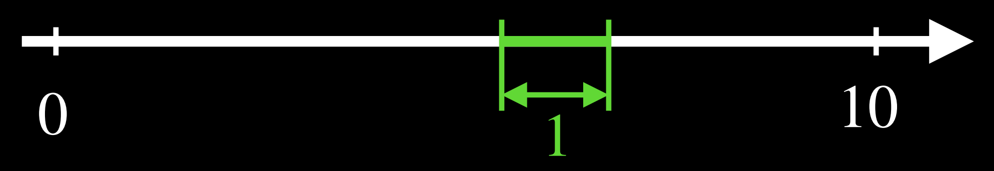
$$\underline{X}_t : \Omega \to \mathbb{S} \subseteq \mathbb{R}^d$$

$$\underline{X}_t \sim \mathcal{U}(\mathbb{S})$$

global convergence:

$$\underline{x}^* = \lim_{n\to\infty} \tilde{\underline{X}}_n^*$$

1d:



$$P(X \in \text{—}) = \frac{1}{10}$$

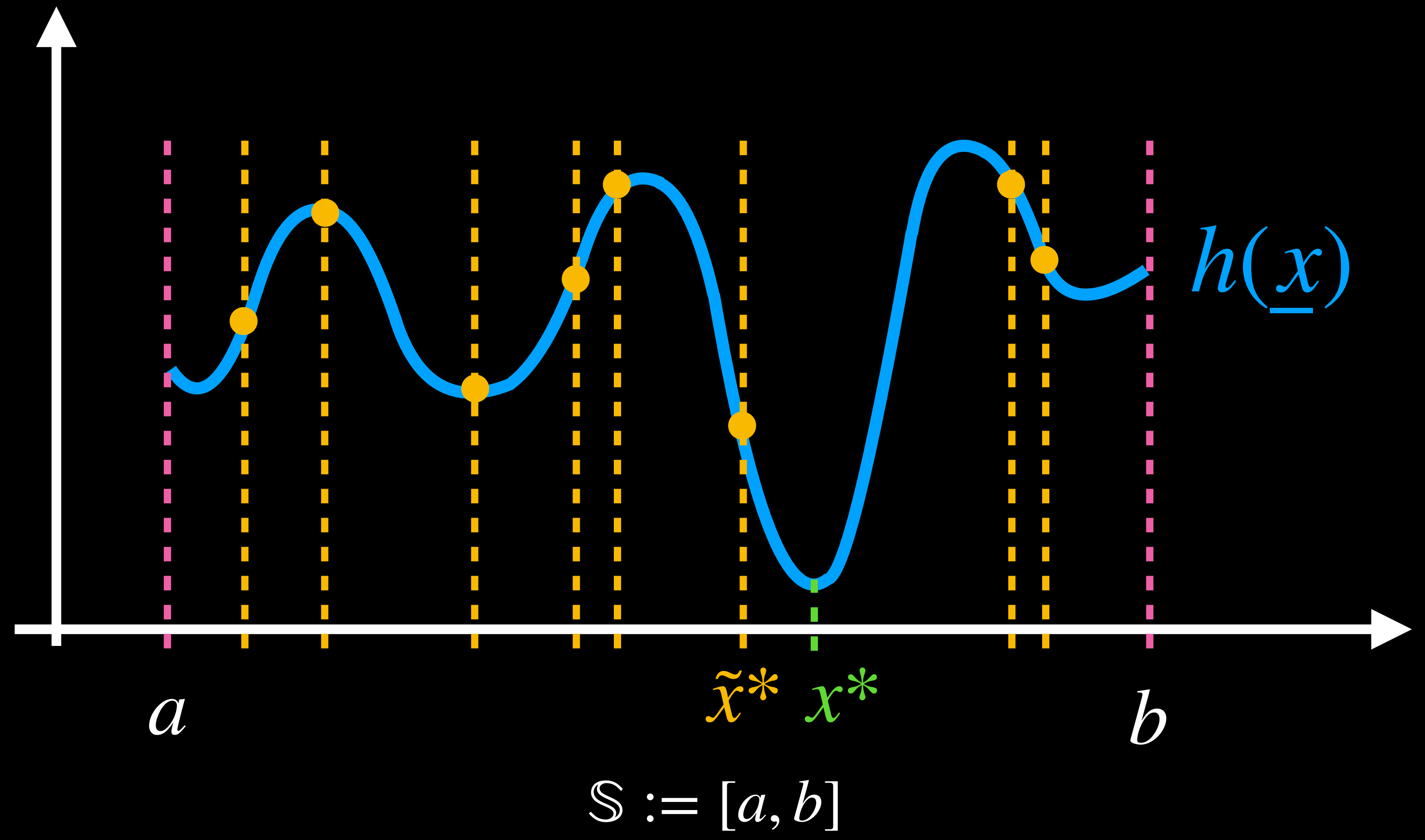
2d:

algorithm 1: stochastic exploration

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \big(h(\underline{x})\big) \approx \arg\min \Big(h(\underline{X}_1), \ldots, h(\underline{X}_n)\Big) =: \tilde{\underline{X}}^*_n$$

$$\underline{X}_t : \Omega \to \mathbb{S} \subseteq \mathbb{R}^d$$

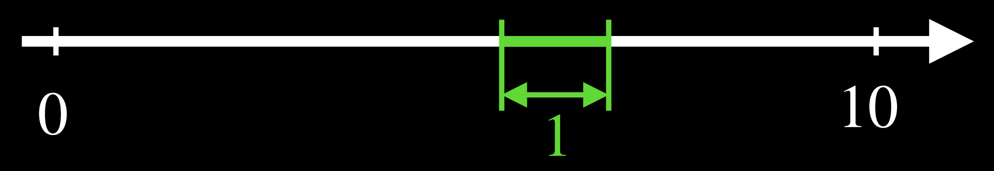
$$\underline{X}_t \sim \mathcal{U}(\mathbb{S})$$



global convergence:

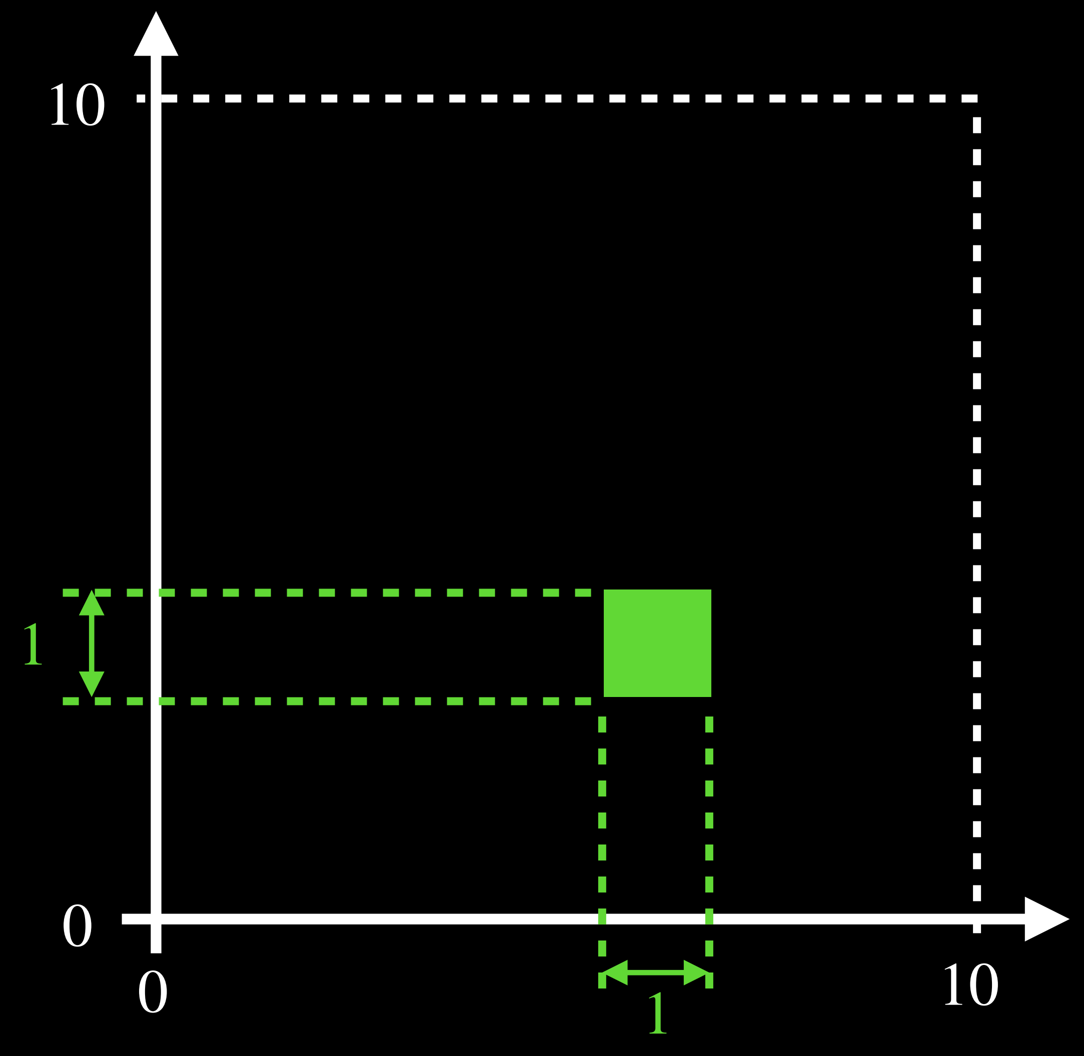
$$\underline{x}^* = \lim_{n\to\infty} \tilde{\underline{X}}^*_n$$

1d:



$$P(X \in \text{—}) = \frac{1}{10}$$

2d:

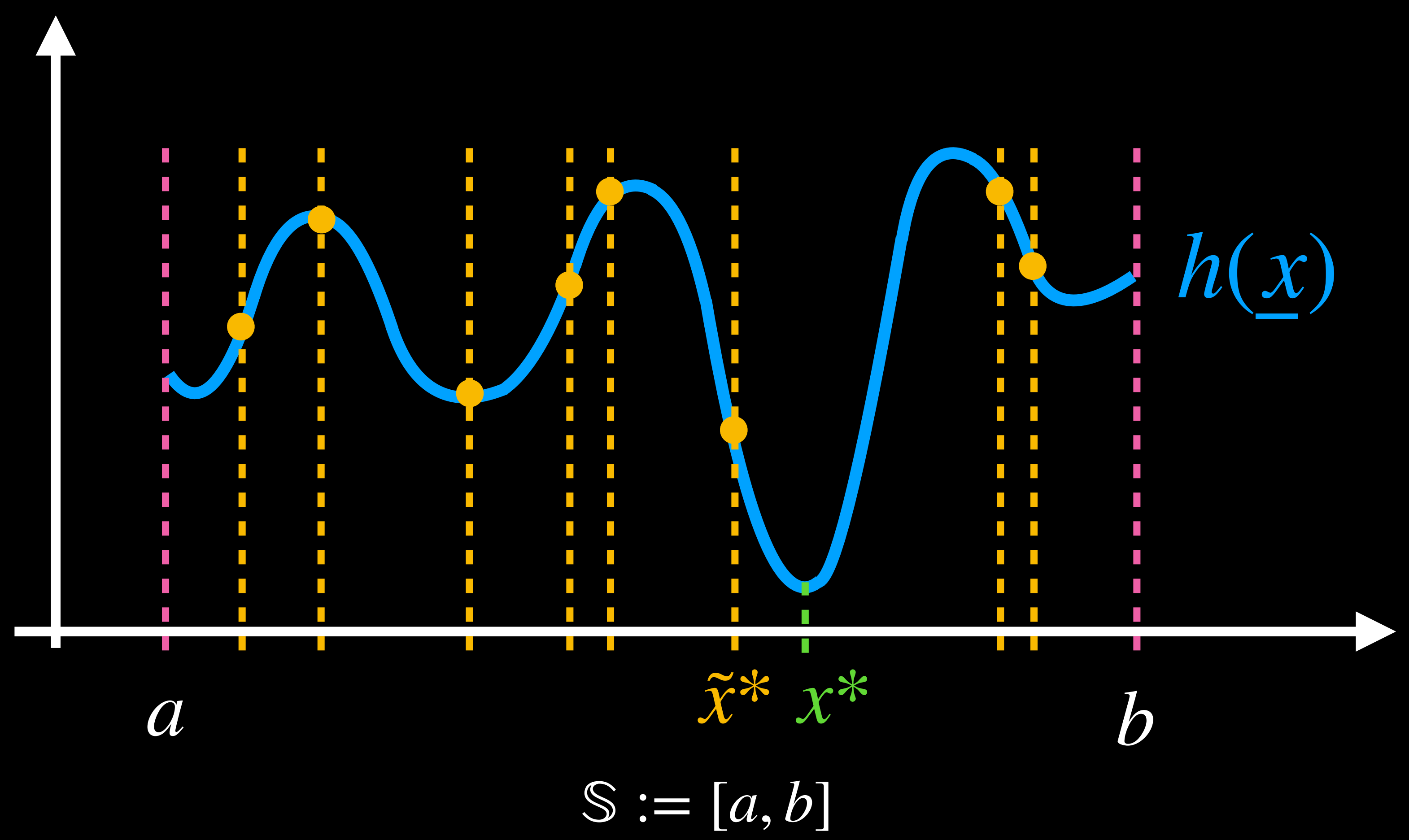


algorithm 1: stochastic exploration

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right) \approx \arg\min \left(h(\underline{X}_1), \ldots, h(\underline{X}_n)\right) =: \underline{\tilde{X}}_n^*$$

$$\underline{X}_t : \Omega \to \mathbb{S} \subseteq \mathbb{R}^d$$

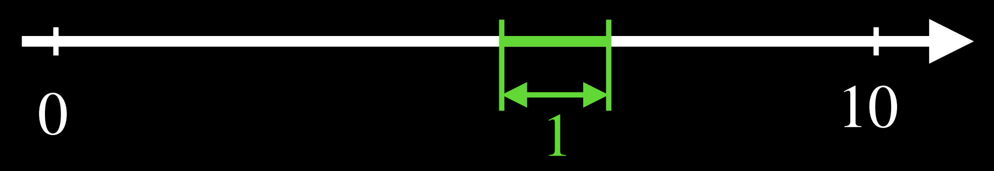
$$\underline{X}_t \sim \mathcal{U}(\mathbb{S})$$



global convergence:

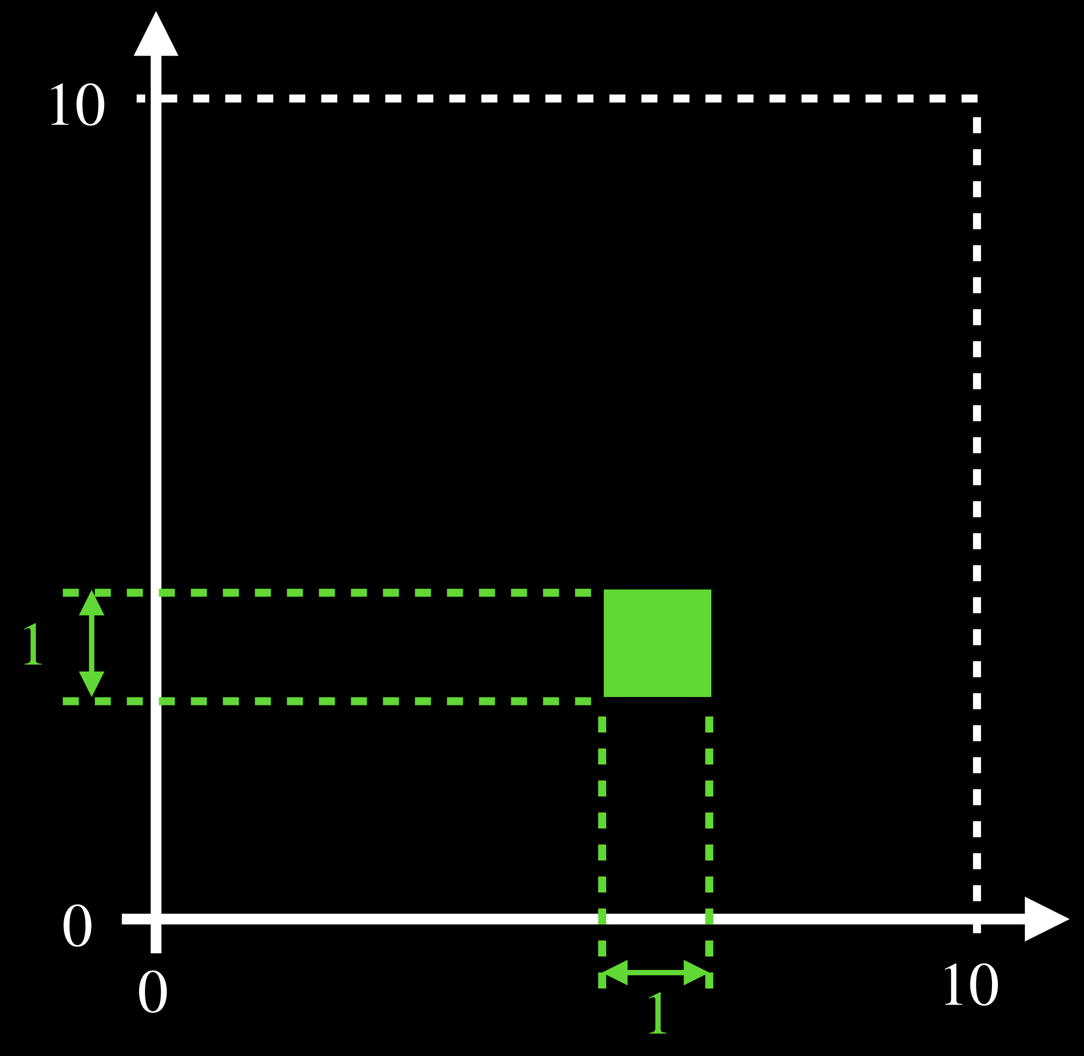
$$\underline{x}^* = \lim_{n\to\infty} \underline{\tilde{X}}_n^*$$

1d:



$P(X \in$ — $) = \dfrac{1}{10}$

2d:



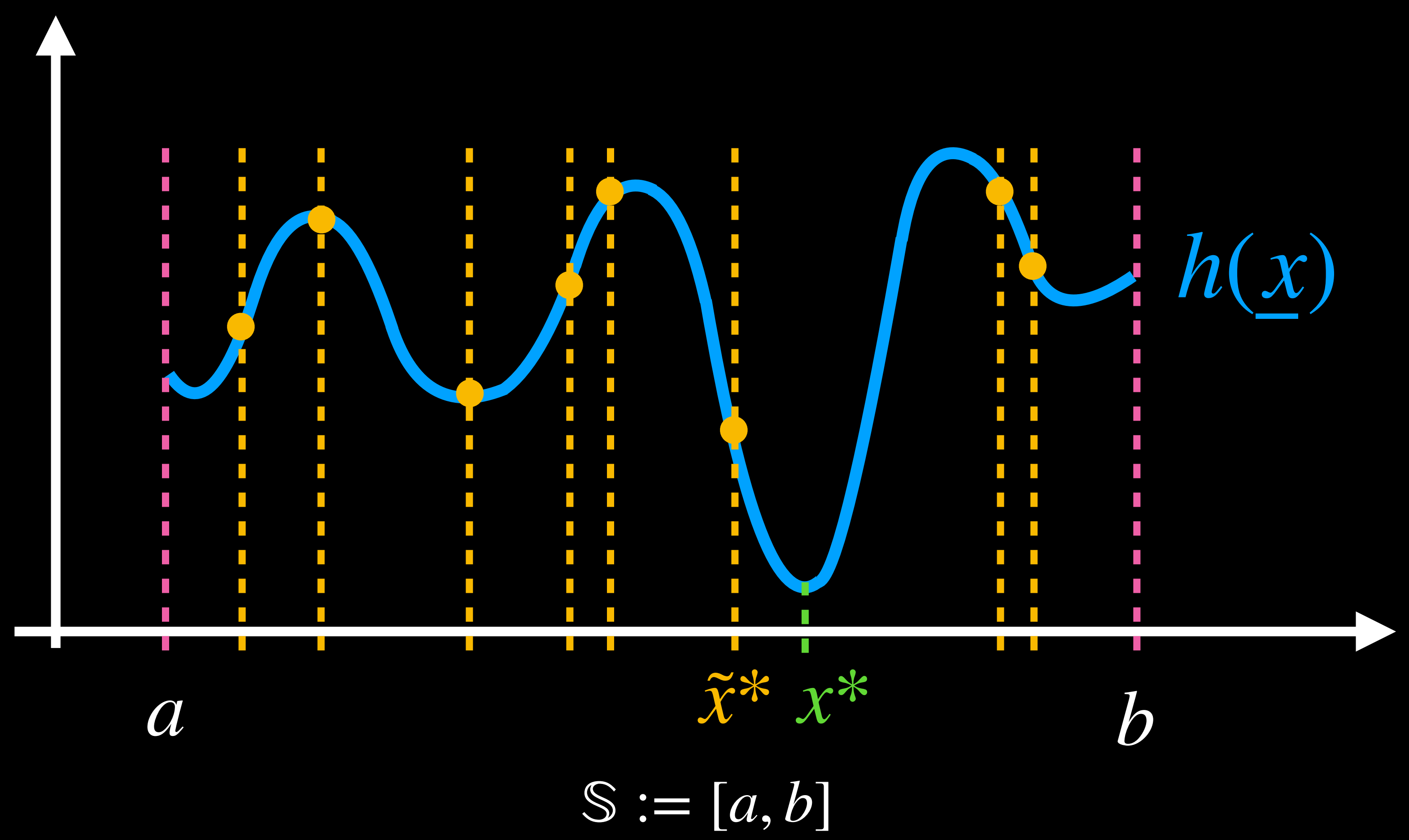
$P(X \in$ ■ $) =$

algorithm 1: stochastic exploration

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \big(h(\underline{x})\big) \approx \arg\min \Big(h(\underline{X}_1), \ldots, h(\underline{X}_n)\Big) =: \tilde{\underline{X}}_n^*$$

$$\underline{X}_t : \Omega \to \mathbb{S} \subseteq \mathbb{R}^d$$

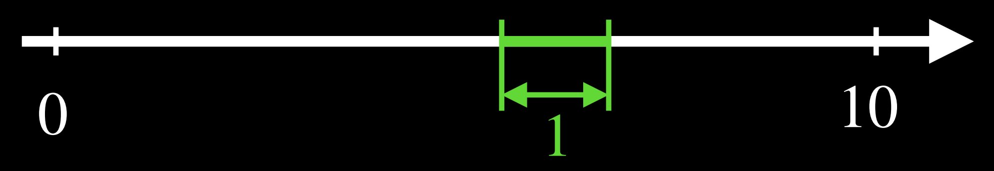
$$\underline{X}_t \sim \mathcal{U}(\mathbb{S})$$



global convergence:

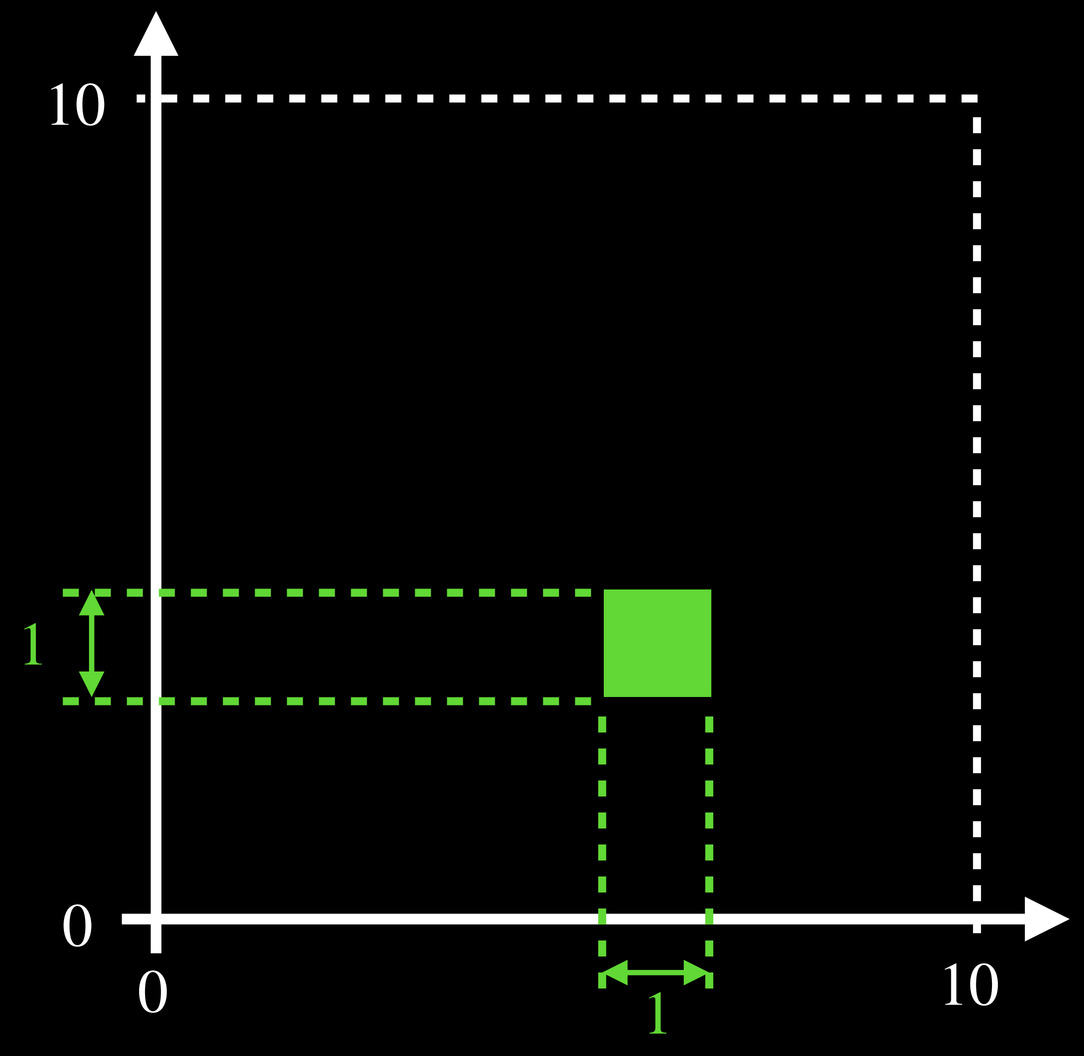
$$\underline{x}^* = \lim_{n\to\infty} \tilde{\underline{X}}_n^*$$

1d:



$$P(X \in \text{—}) = \frac{1}{10}$$

2d:



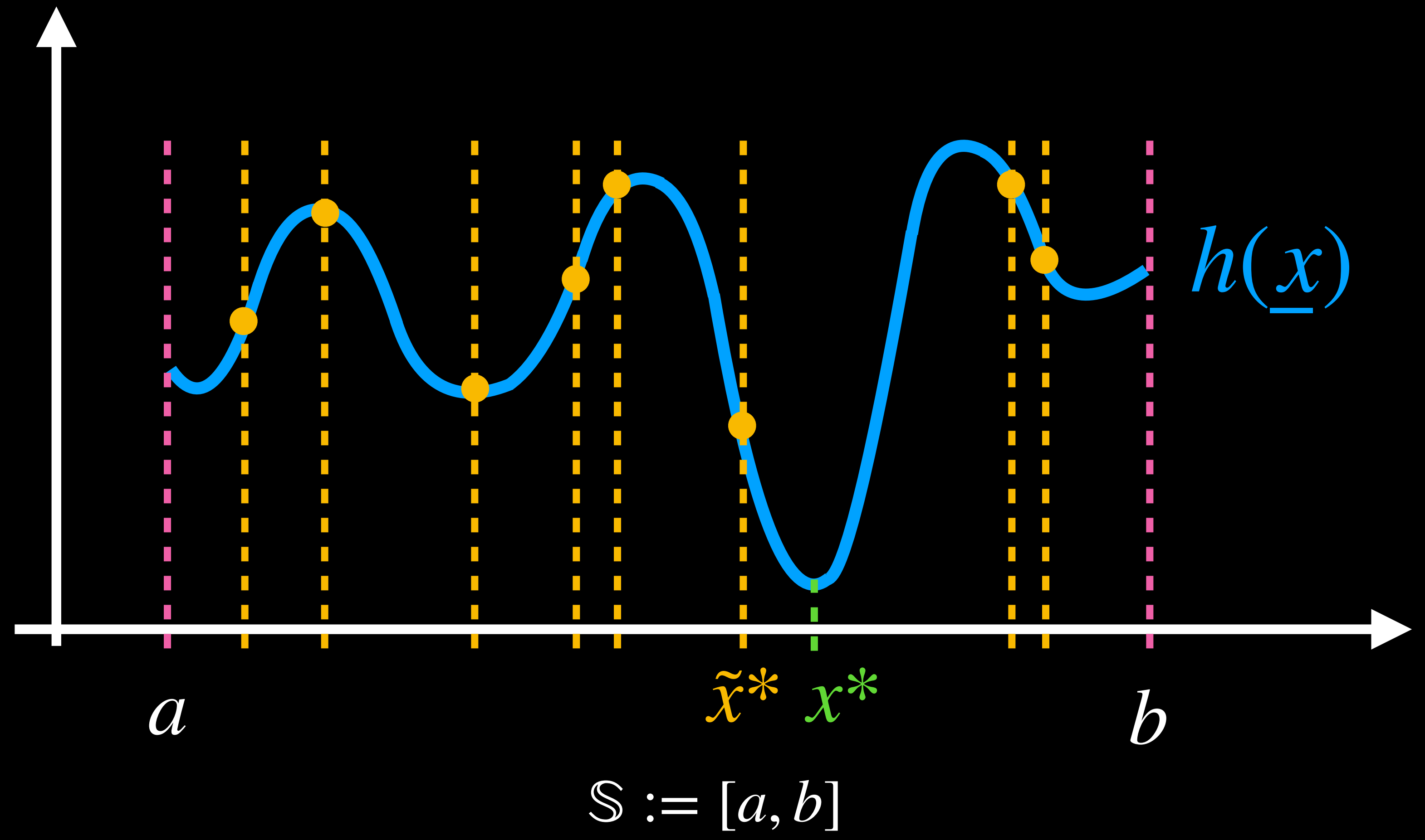
$$P(X \in \blacksquare) = \frac{1}{10^2}$$

algorithm 1: stochastic exploration

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right) \approx \arg\min \left(h(\underline{X}_1), \ldots, h(\underline{X}_n)\right) =: \tilde{\underline{X}}_n^*$$

$$\underline{X}_t : \Omega \to \mathbb{S} \subseteq \mathbb{R}^d$$

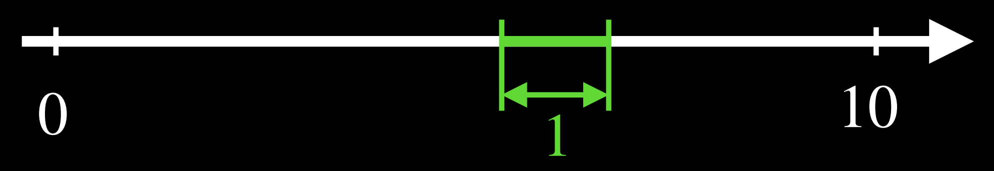
$$\underline{X}_t \sim \mathcal{U}(\mathbb{S})$$



global convergence:

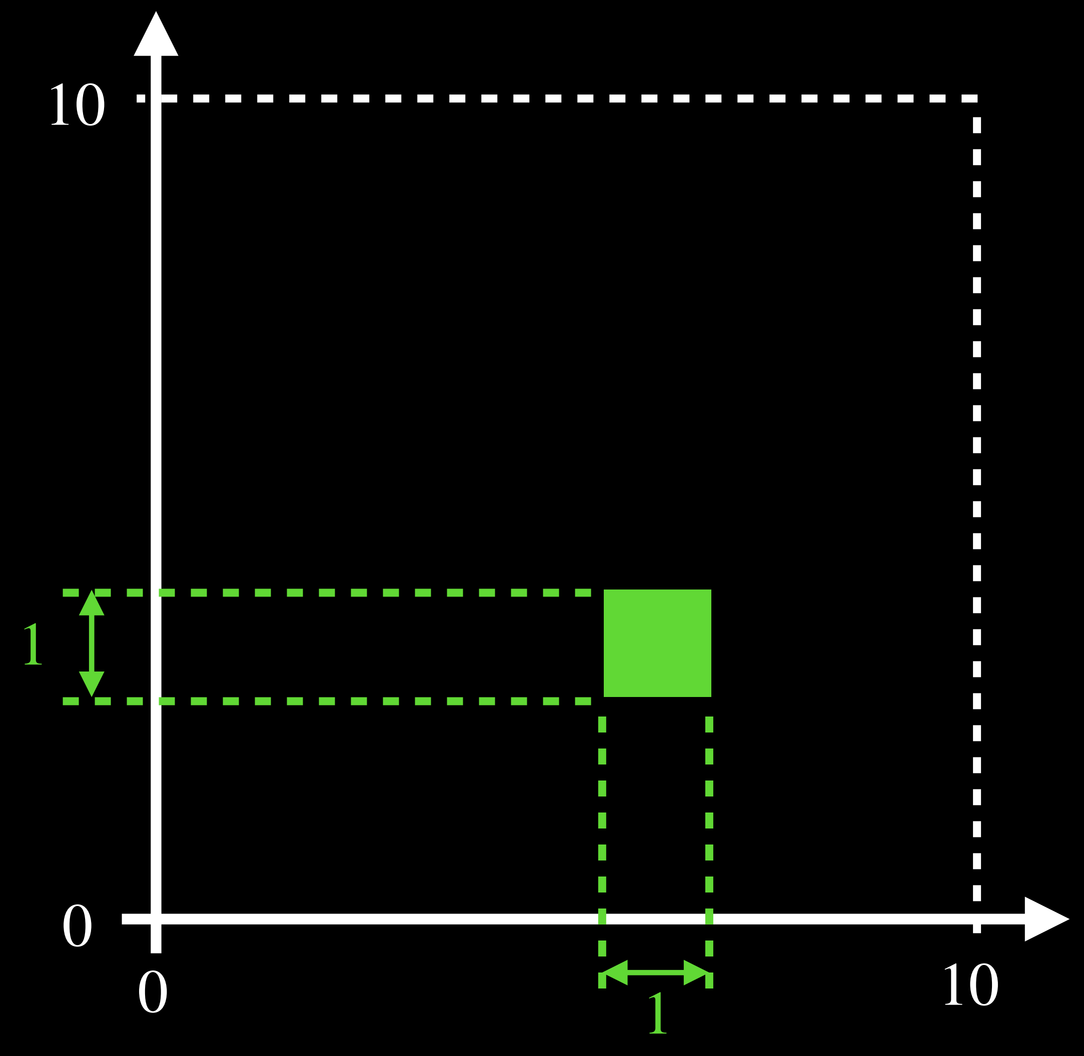
$$\underline{x}^* = \lim_{n\to\infty} \tilde{\underline{X}}_n^*$$

1d:



$$P(X \in \text{—}) = \frac{1}{10}$$

2d:



$$P(X \in \blacksquare) = \frac{1}{10^2}$$

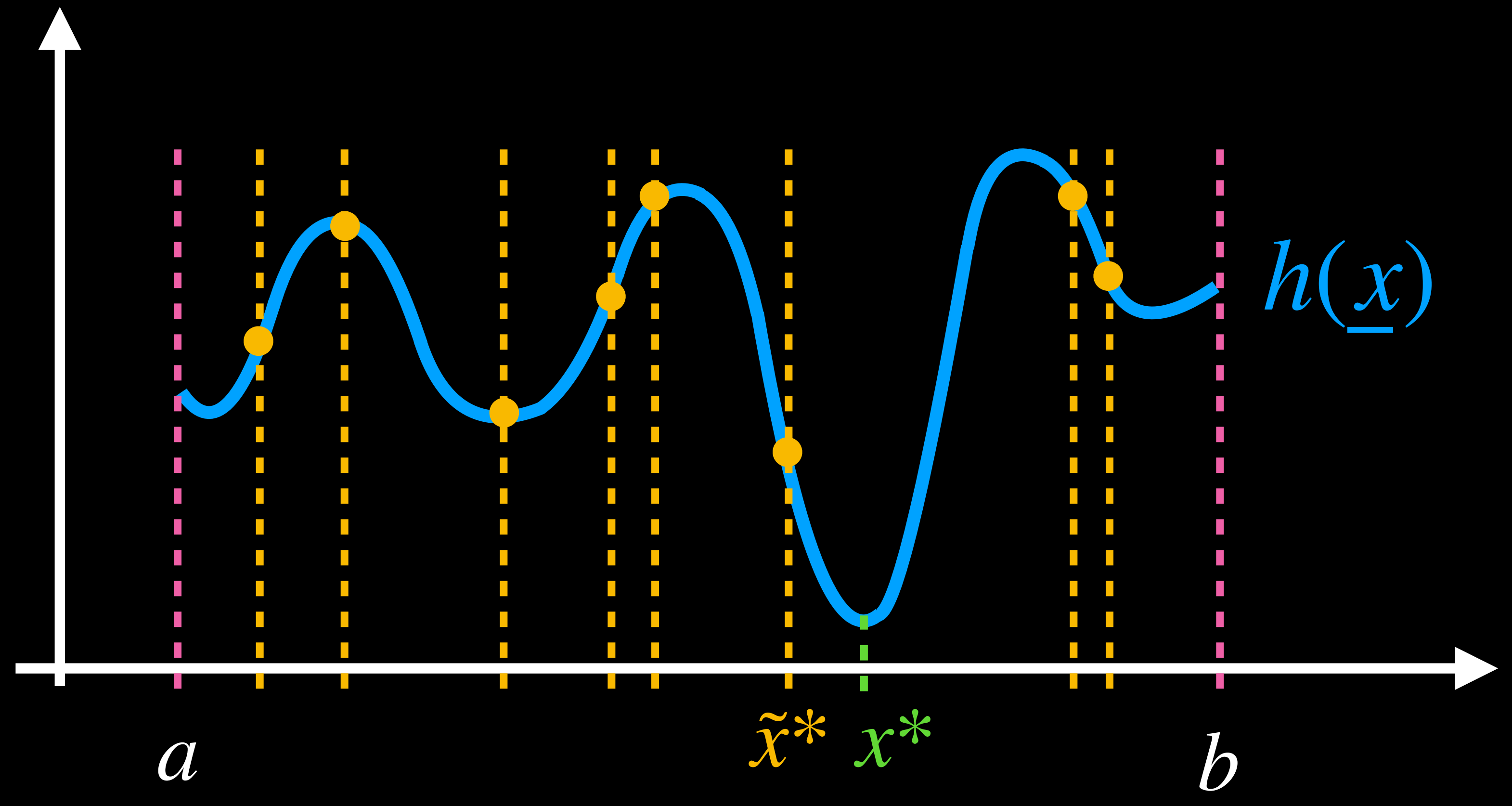
$$error \propto n^{-\frac{1}{d}}$$

algorithm 1: stochastic exploration

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \big(h(\underline{x})\big) \approx \arg\min \Big(h(\underline{X}_1),\ldots,h(\underline{X}_n)\Big) =: \underline{\tilde{X}}^*_n$$

$$\underline{X}_t : \Omega \to \mathbb{S} \subseteq \mathbb{R}^d$$

$$\underline{X}_t \sim \mathcal{U}(\mathbb{S})$$

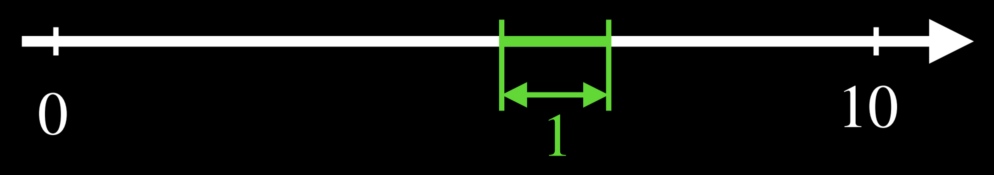


$\mathbb{S} := [a, b]$

global convergence:

$$\underline{x}^* = \lim_{n\to\infty} \underline{\tilde{X}}^*_n$$

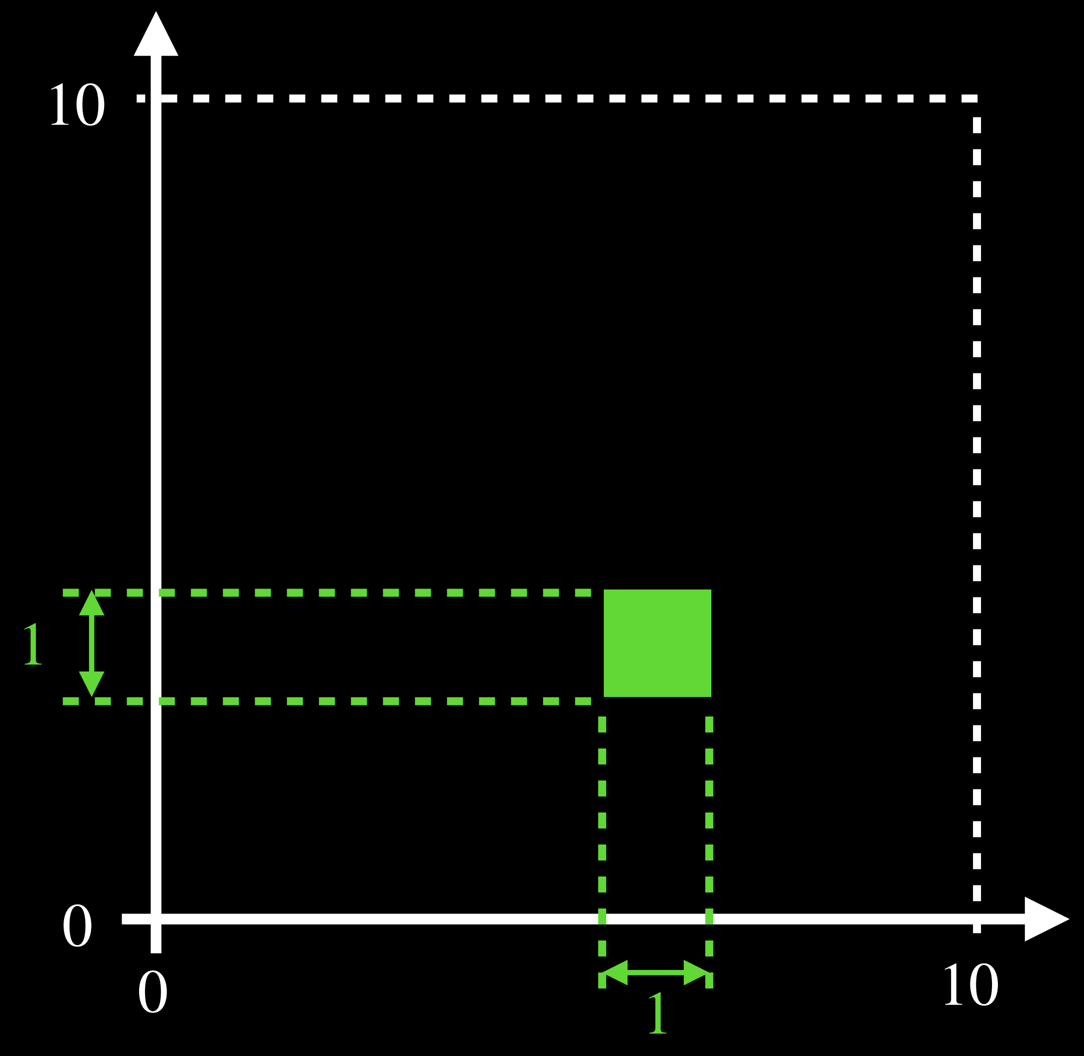
1d:



$$P(X \in \text{—}) = \frac{1}{10}$$

properties:

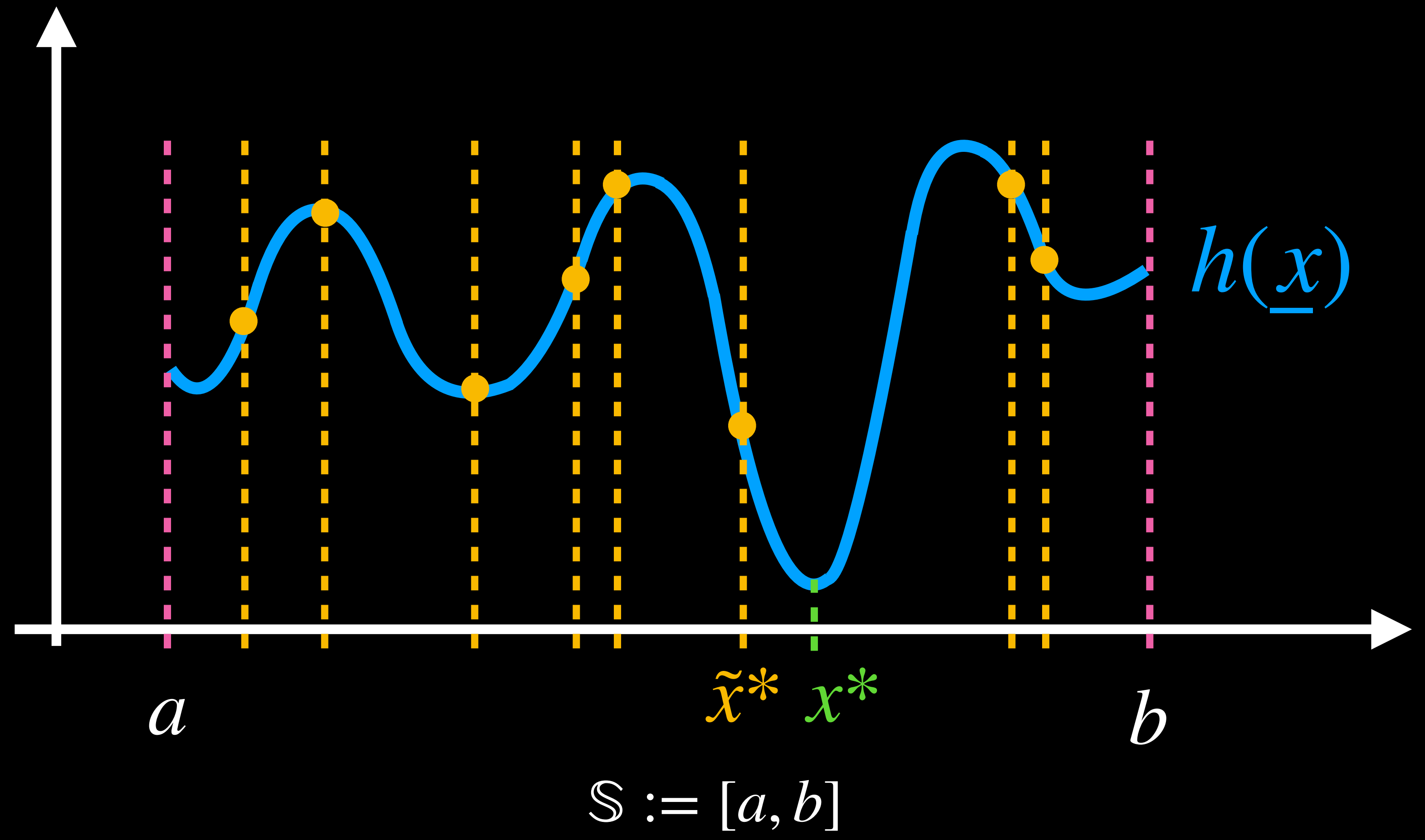
2d:



$$P(X \in \blacksquare) = \frac{1}{10^2}$$

$$error \propto n^{-\frac{1}{d}}$$

algorithm 1: stochastic exploration



$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right) \approx \arg\min \left(h(\underline{X}_1), \ldots, h(\underline{X}_n)\right) =: \underline{\tilde{X}}_n^*$$

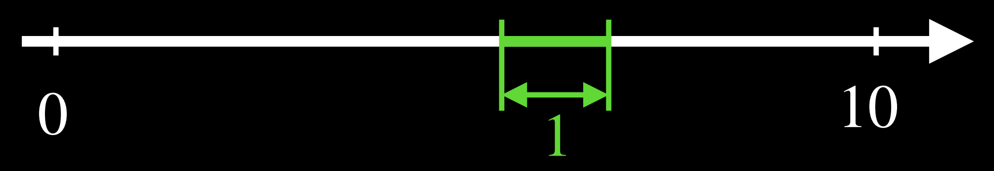
$$\underline{X}_t : \Omega \to \mathbb{S} \subseteq \mathbb{R}^d$$

$$\underline{X}_t \sim \mathcal{U}(\mathbb{S})$$

global convergence:

$$\underline{x}^* = \lim_{n\to\infty} \underline{\tilde{X}}_n^*$$

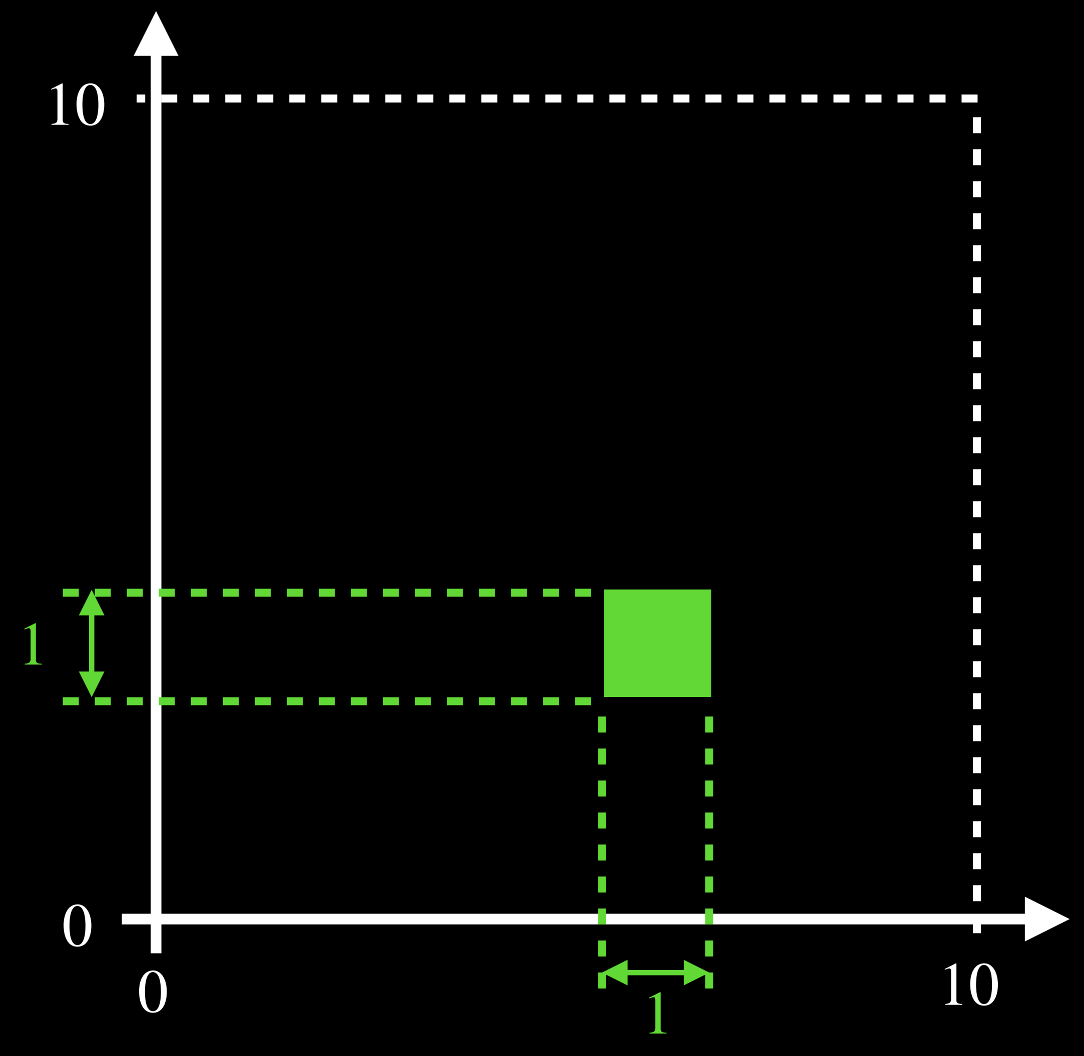
1d:



$$P(X \in \text{—}) = \frac{1}{10}$$

properties:
- very simple

2d:



$$P(X \in \blacksquare) = \frac{1}{10^2}$$

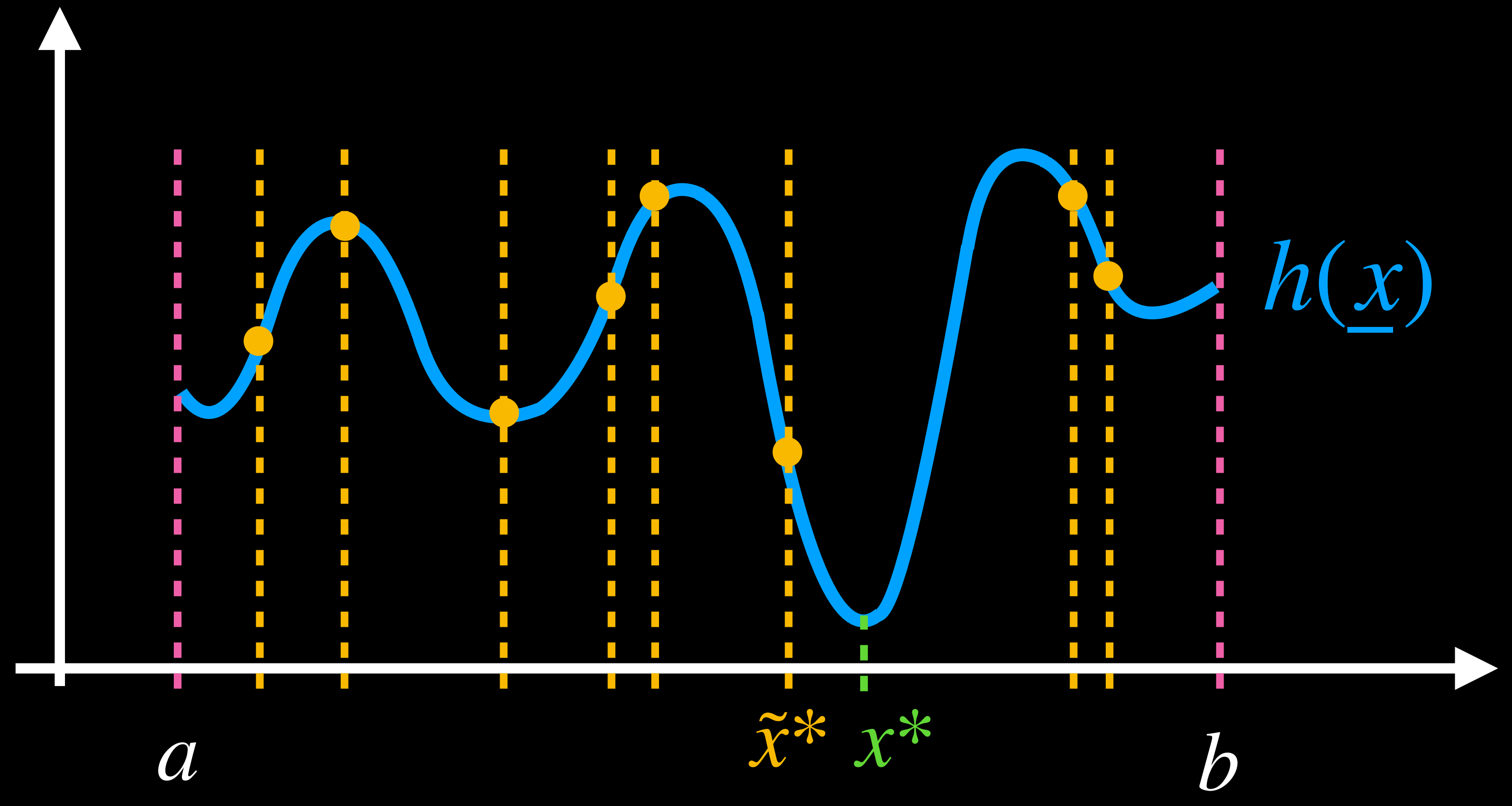
$$error \propto n^{-\frac{1}{d}}$$

algorithm 1: stochastic exploration

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right) \approx \arg\min \left(h(\underline{X}_1),\ldots,h(\underline{X}_n)\right) =: \tilde{\underline{X}}^*_n$$

$$\underline{X}_t : \Omega \to \mathbb{S} \subseteq \mathbb{R}^d$$

$$\underline{X}_t \sim \mathcal{U}(\mathbb{S})$$



$\mathbb{S} := [a, b]$

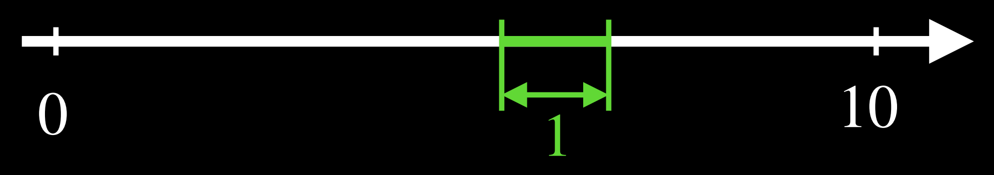
global convergence:

$$\underline{x}^* = \lim_{n\to\infty} \tilde{\underline{X}}^*_n$$

properties:

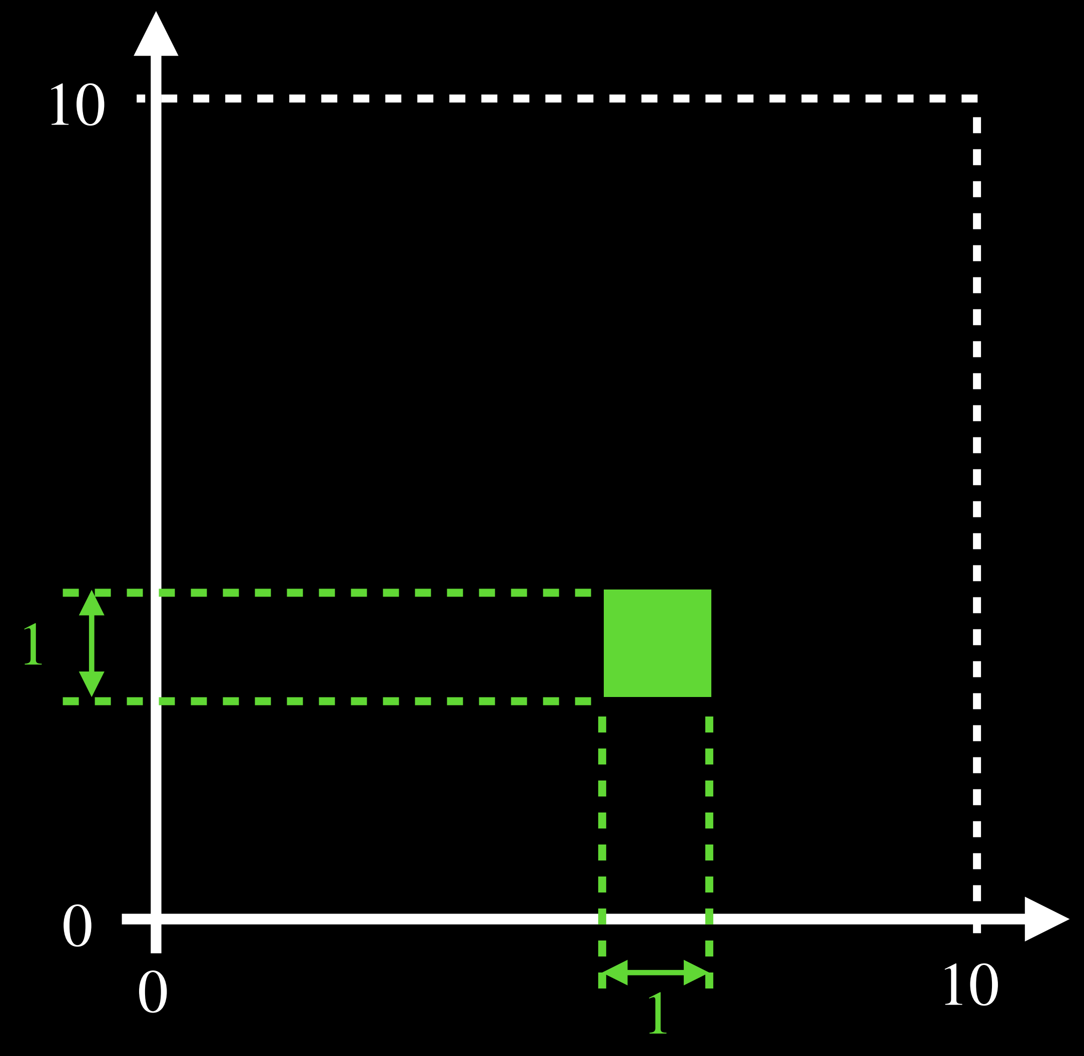
- very simple
- converges to $\underline{x}^*$

1d:



$$P(X \in \text{—}) = \frac{1}{10}$$

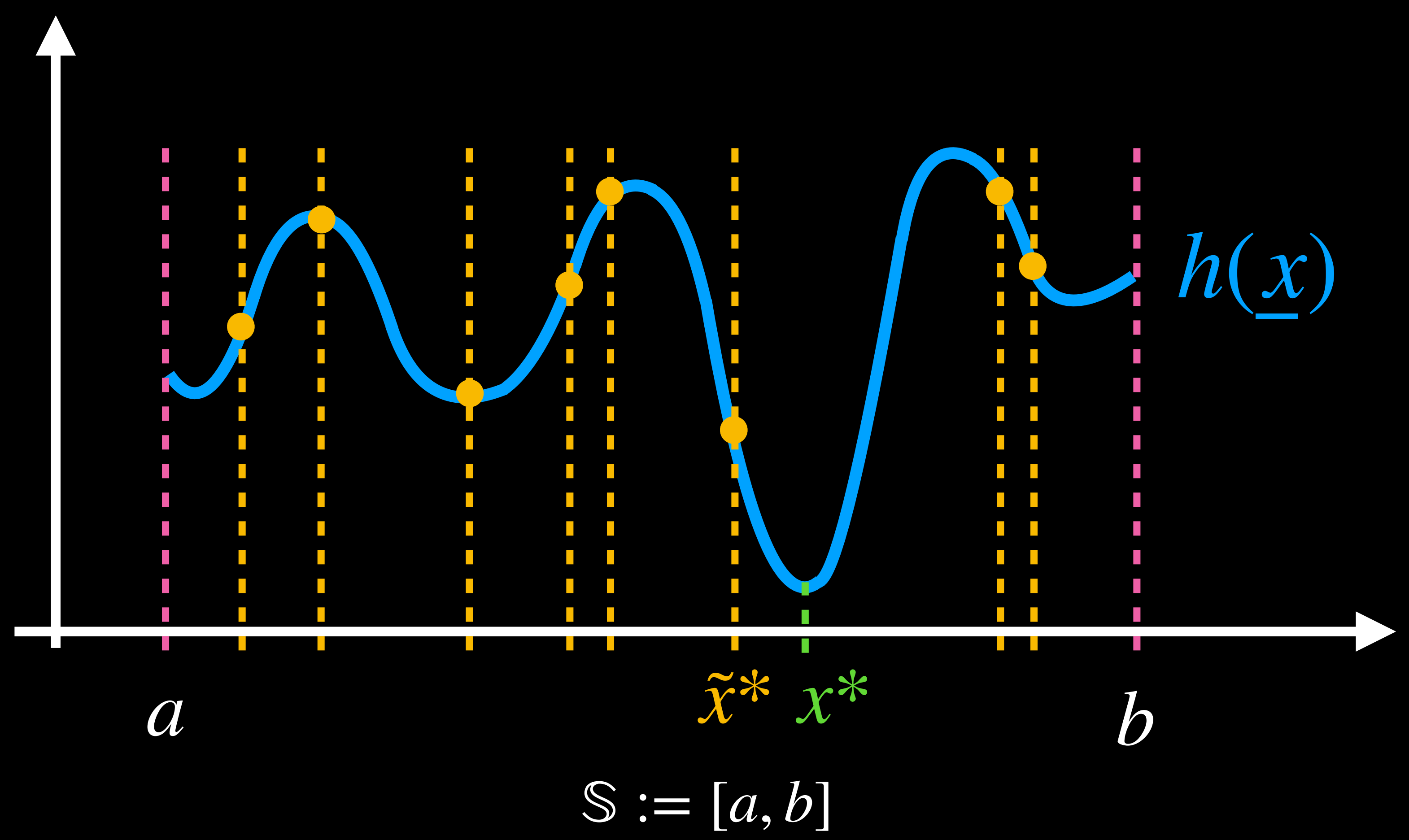
2d:



$$P(X \in \blacksquare) = \frac{1}{10^2}$$

$$error \propto n^{-\frac{1}{d}}$$

algorithm 1: stochastic exploration



$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right) \approx \arg\min \left(h(\underline{X}_1), \dots, h(\underline{X}_n)\right) =: \tilde{\underline{X}}^*_n$$

$\underline{X}_t : \Omega \to \mathbb{S} \subseteq \mathbb{R}^d$

$\underline{X}_t \sim \mathcal{U}(\mathbb{S})$

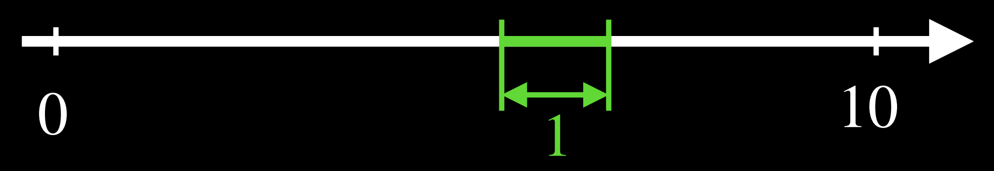
global convergence:

$$\underline{x}^* = \lim_{n\to\infty} \tilde{\underline{X}}^*_n$$

properties:
- very simple
- converges to $\underline{x}^*$

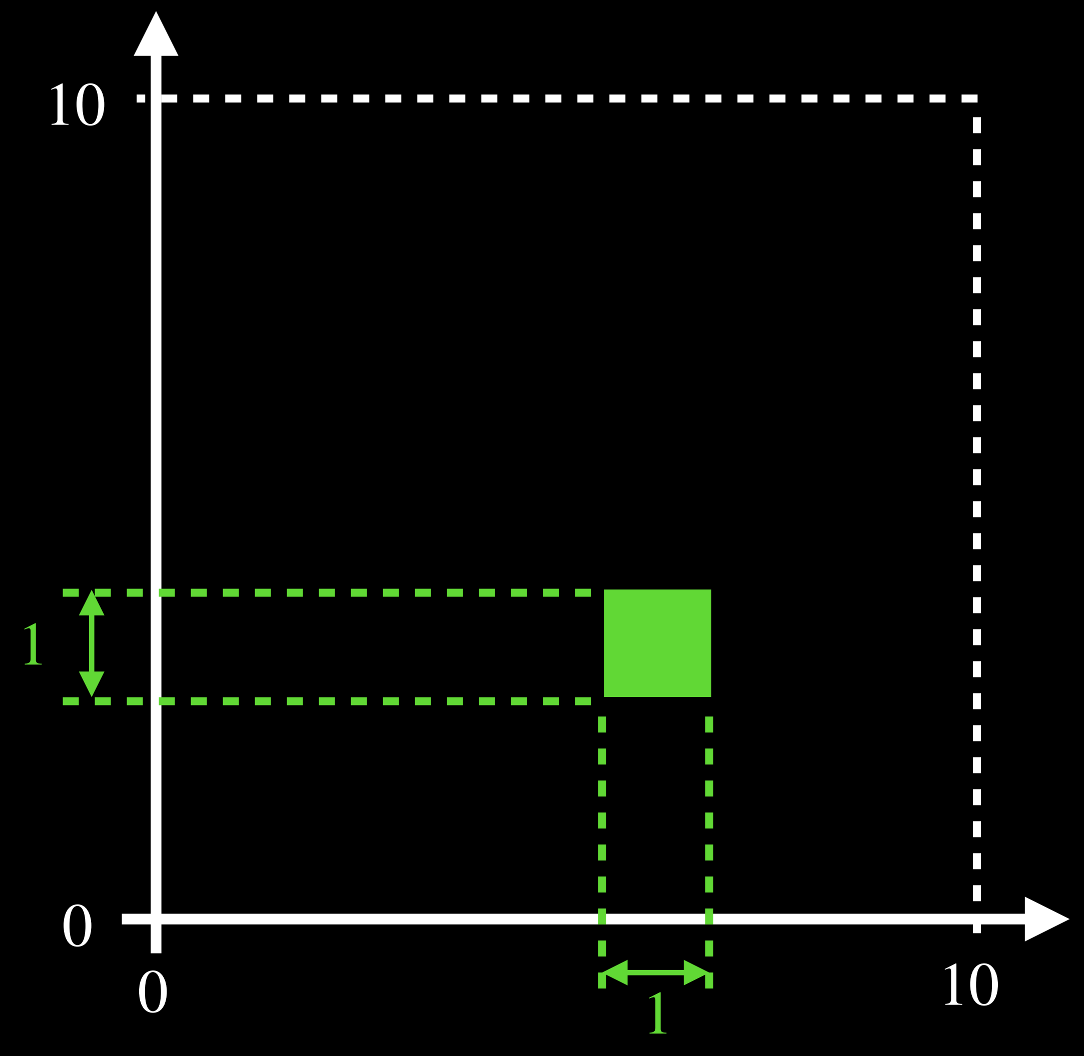
useful if:

1d:



$$P(X \in \text{—}) = \frac{1}{10}$$

2d:



$$P(X \in \blacksquare) = \frac{1}{10^2}$$

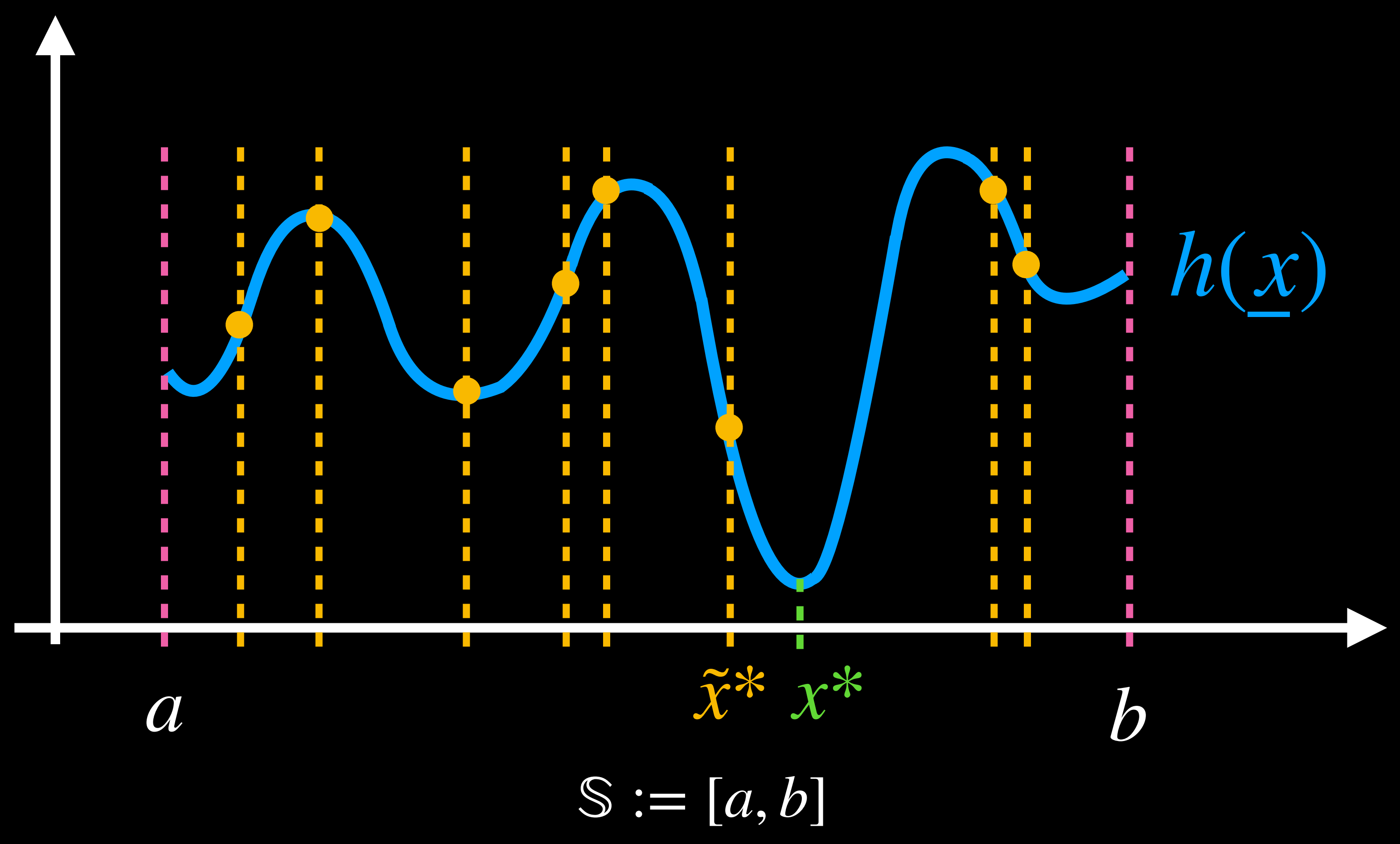
$$error \propto n^{-\frac{1}{d}}$$

algorithm 1: stochastic exploration

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right) \approx \arg\min \left(h(\underline{X}_1),\ldots,h(\underline{X}_n)\right) =: \tilde{\underline{X}}^*_n$$

$$\underline{X}_t : \Omega \to \mathbb{S} \subseteq \mathbb{R}^d$$

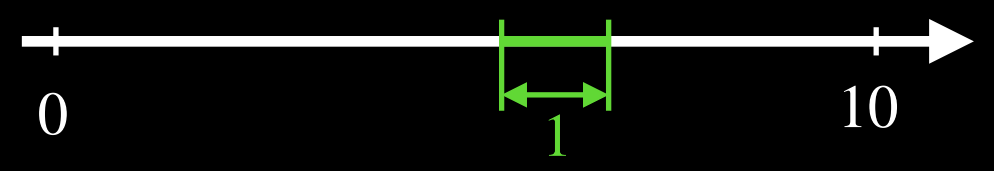
$$\underline{X}_t \sim \mathcal{U}(\mathbb{S})$$



global convergence:

$$\underline{x}^* = \lim_{n\to\infty} \tilde{\underline{X}}^*_n$$

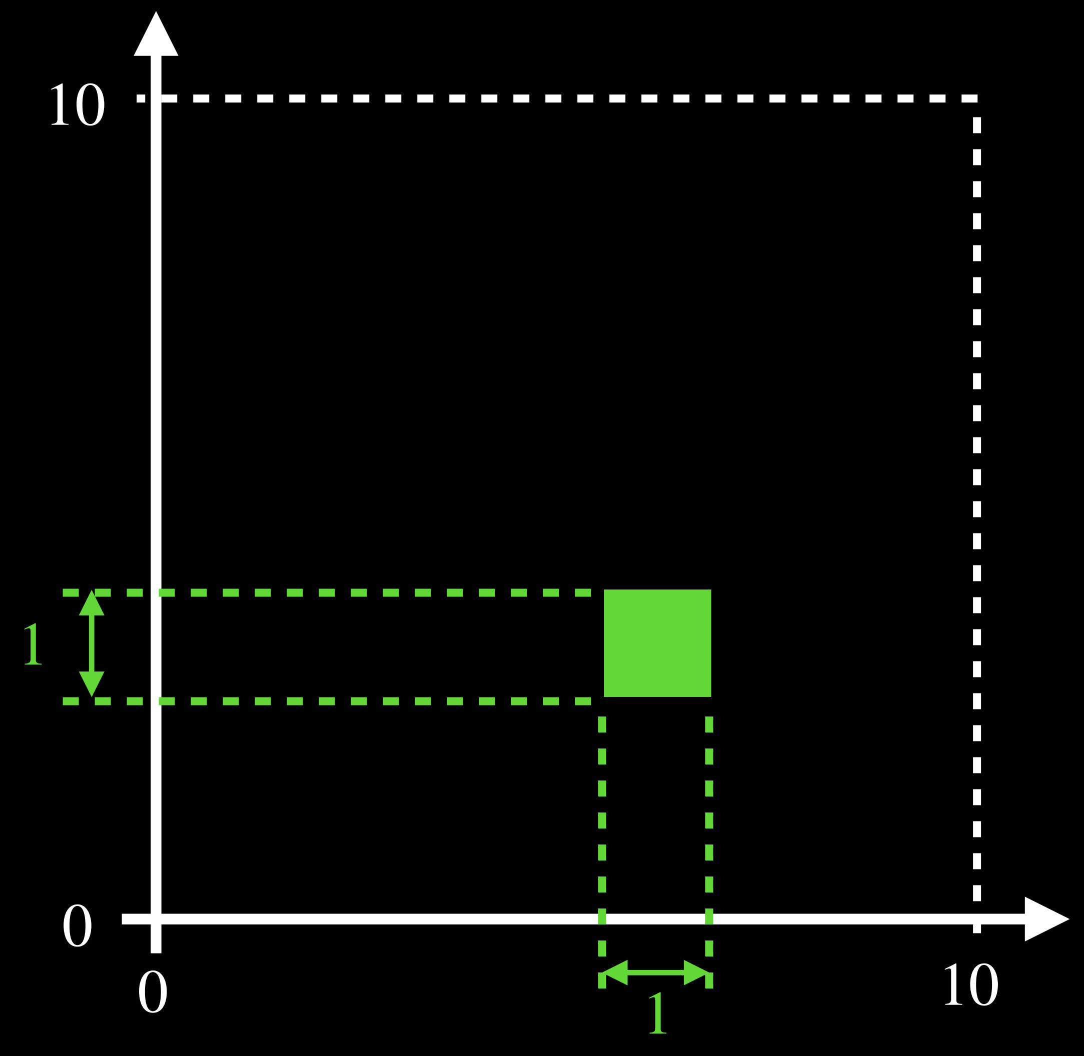
1d:



$$P(X \in \text{—}) = \frac{1}{10}$$

properties:
- very simple
- converges to $\underline{x}^*$

2d:



$$P(X \in \blacksquare) = \frac{1}{10^2}$$

useful if:
- $\mathbb{S}$ is low-dimensional and bounded

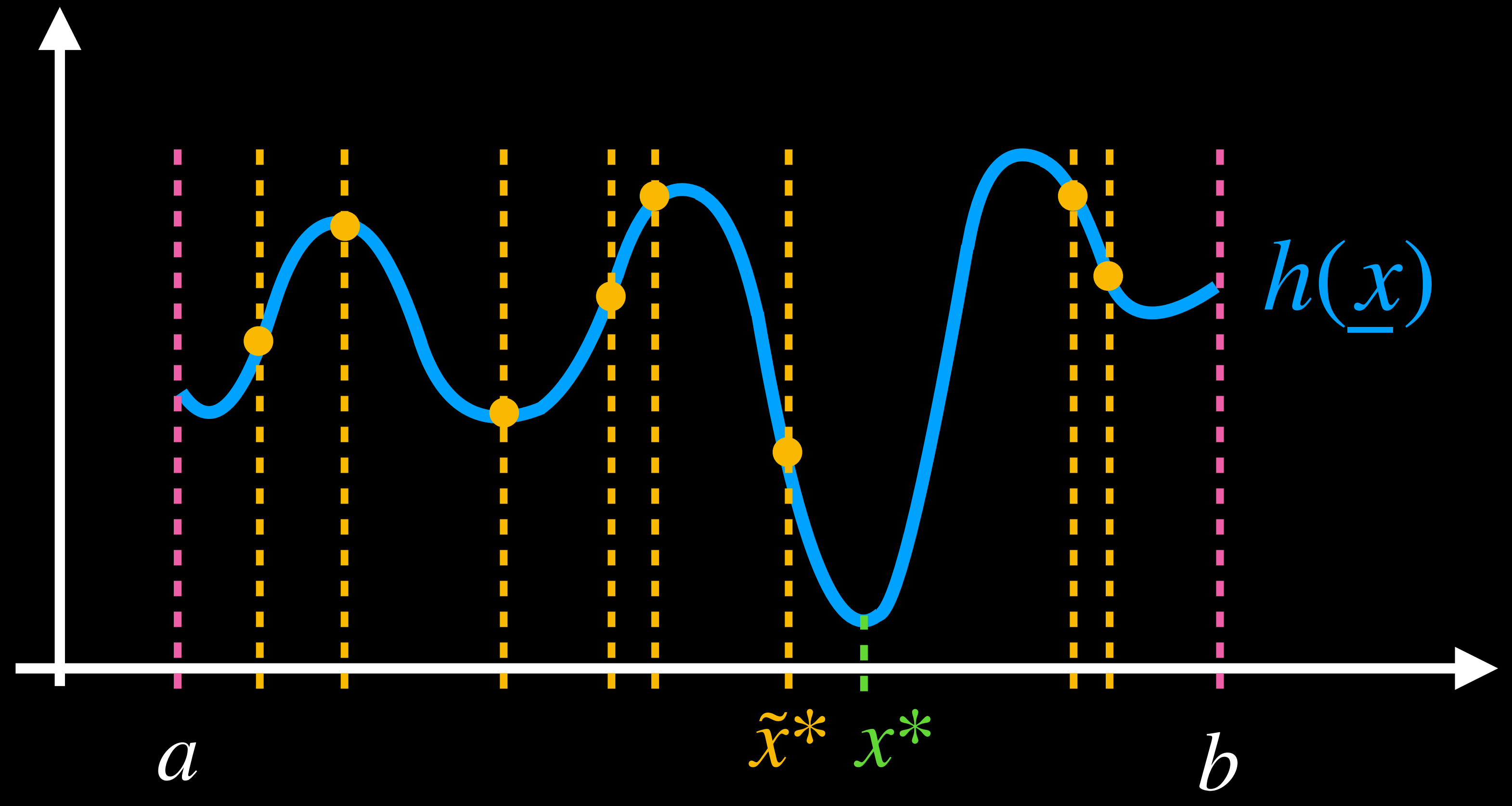
$$error \propto n^{-\frac{1}{d}}$$

algorithm 1: stochastic exploration

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right) \approx \arg\min\left(h(\underline{X}_1),\ldots,h(\underline{X}_n)\right) =: \underline{\tilde{X}}^*_n$$

$\underline{X}_t : \Omega \to \mathbb{S} \subseteq \mathbb{R}^d$

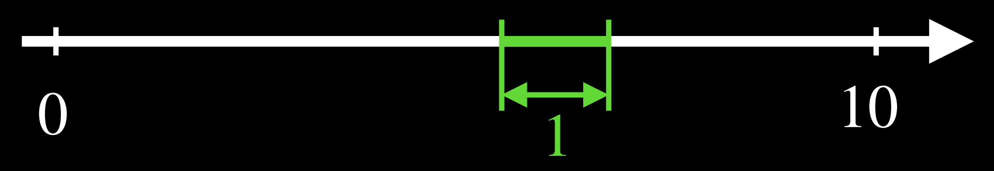
$\underline{X}_t \sim \mathcal{U}(\mathbb{S})$



global convergence:

$$\underline{x}^* = \lim_{n\to\infty} \underline{\tilde{X}}^*_n$$

1d:

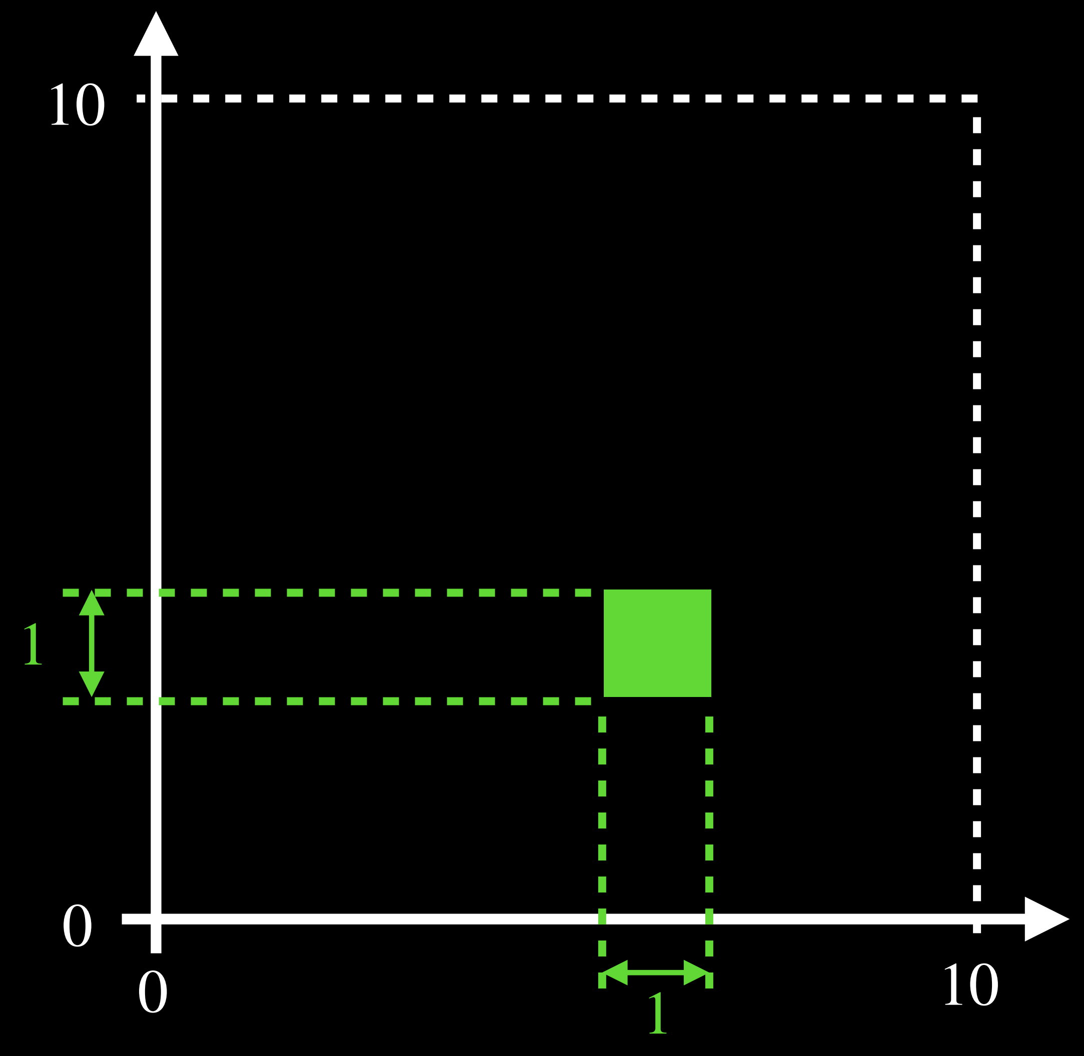


$P(X \in \text{—}) = \frac{1}{10}$

properties:

- very simple
- converges to $\underline{x}^*$

2d:



$P(X \in \blacksquare) = \frac{1}{10^2}$

useful if:

- $\mathbb{S}$ is low-dimensional and bounded
- (h is cheap to evaluate for higher dimensions)

$$error \propto n^{-\frac{1}{d}}$$

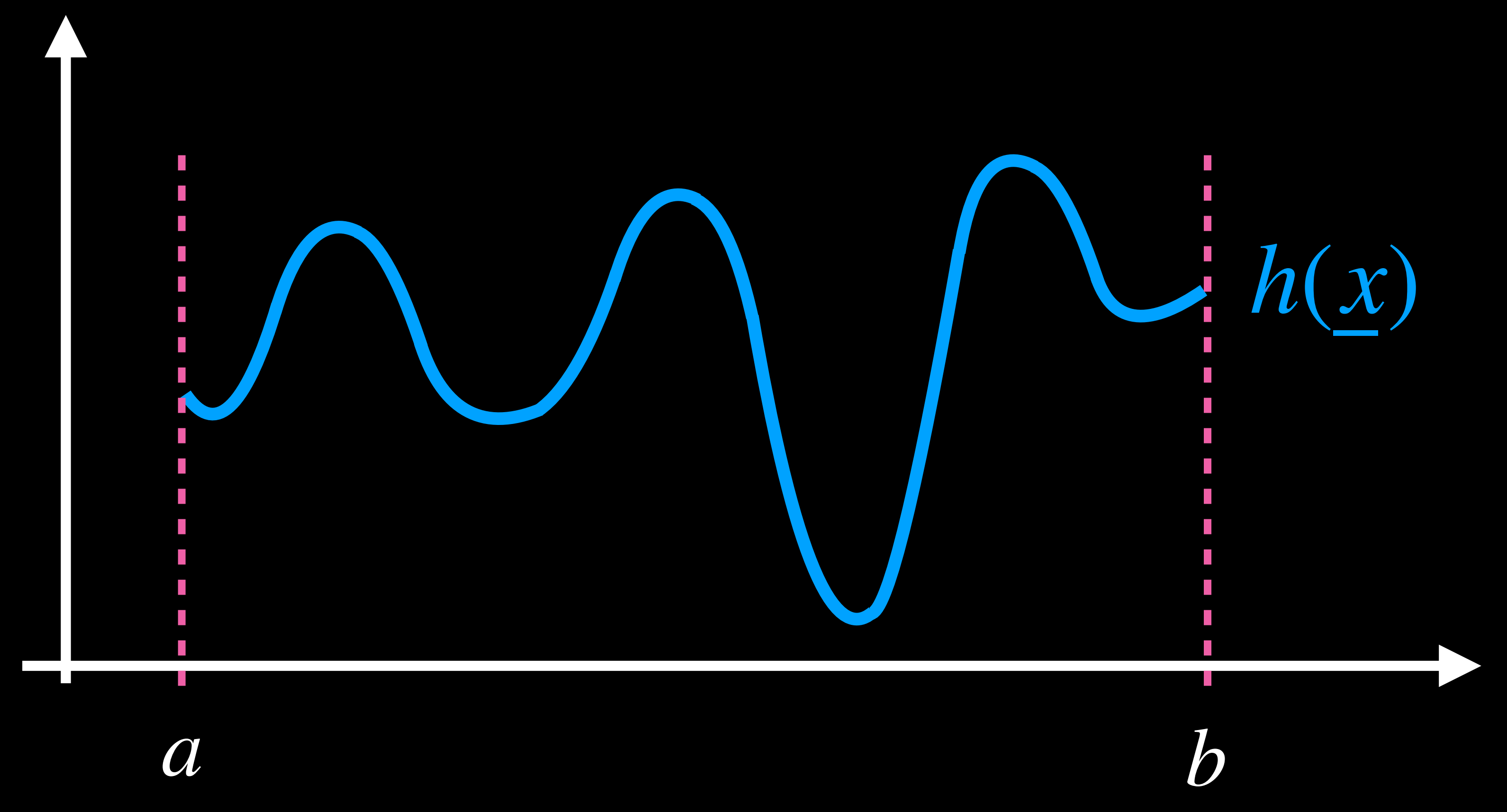
algorithm 2: stochastic descent

algorithm 2: stochastic descent

$$\underline{x}^* := \arg\min_{\underline{x} \in \mathbb{S}} \left(h(\underline{x}) \right)$$

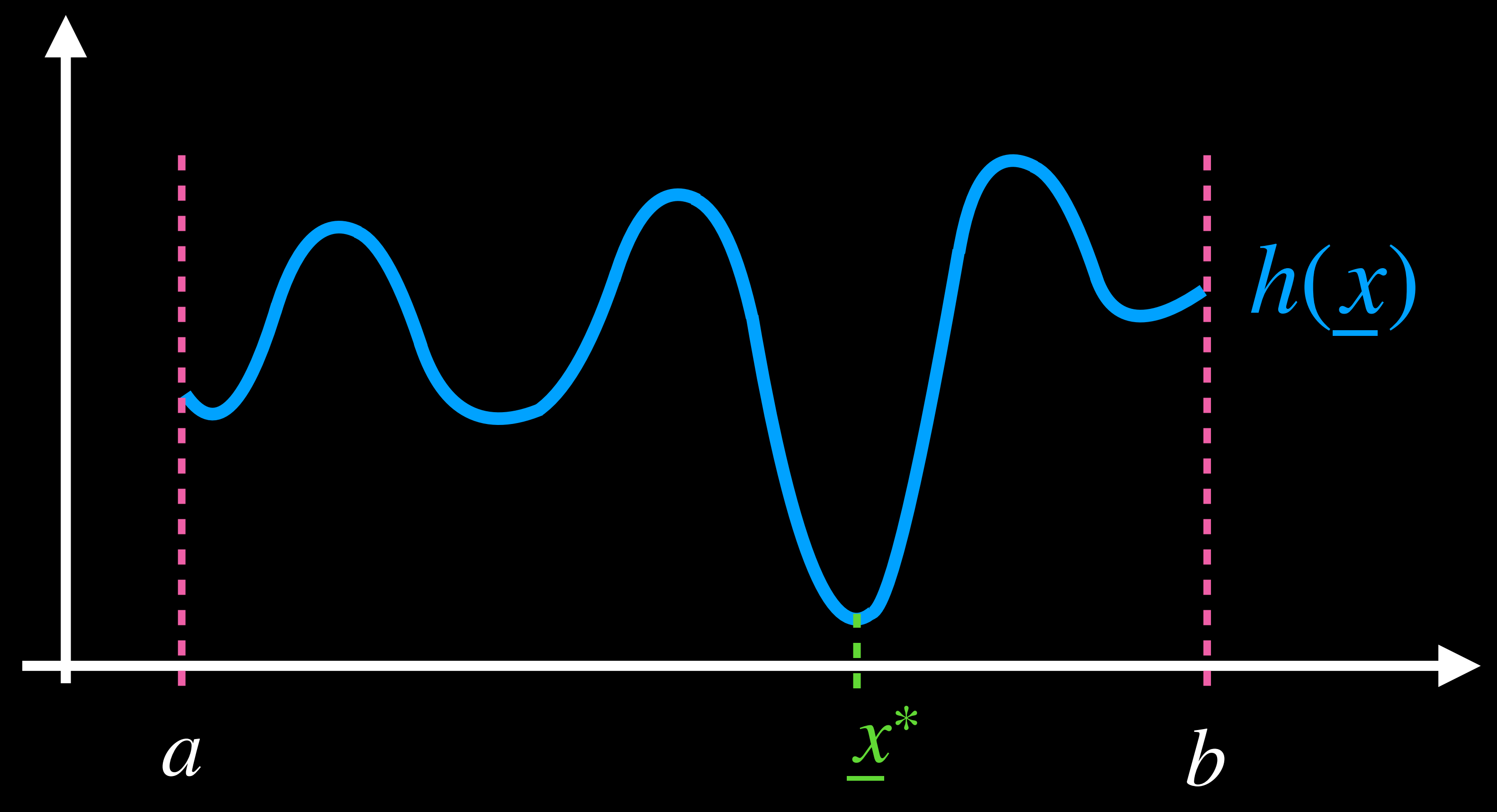
algorithm 2: stochastic descent

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$



algorithm 2: stochastic descent

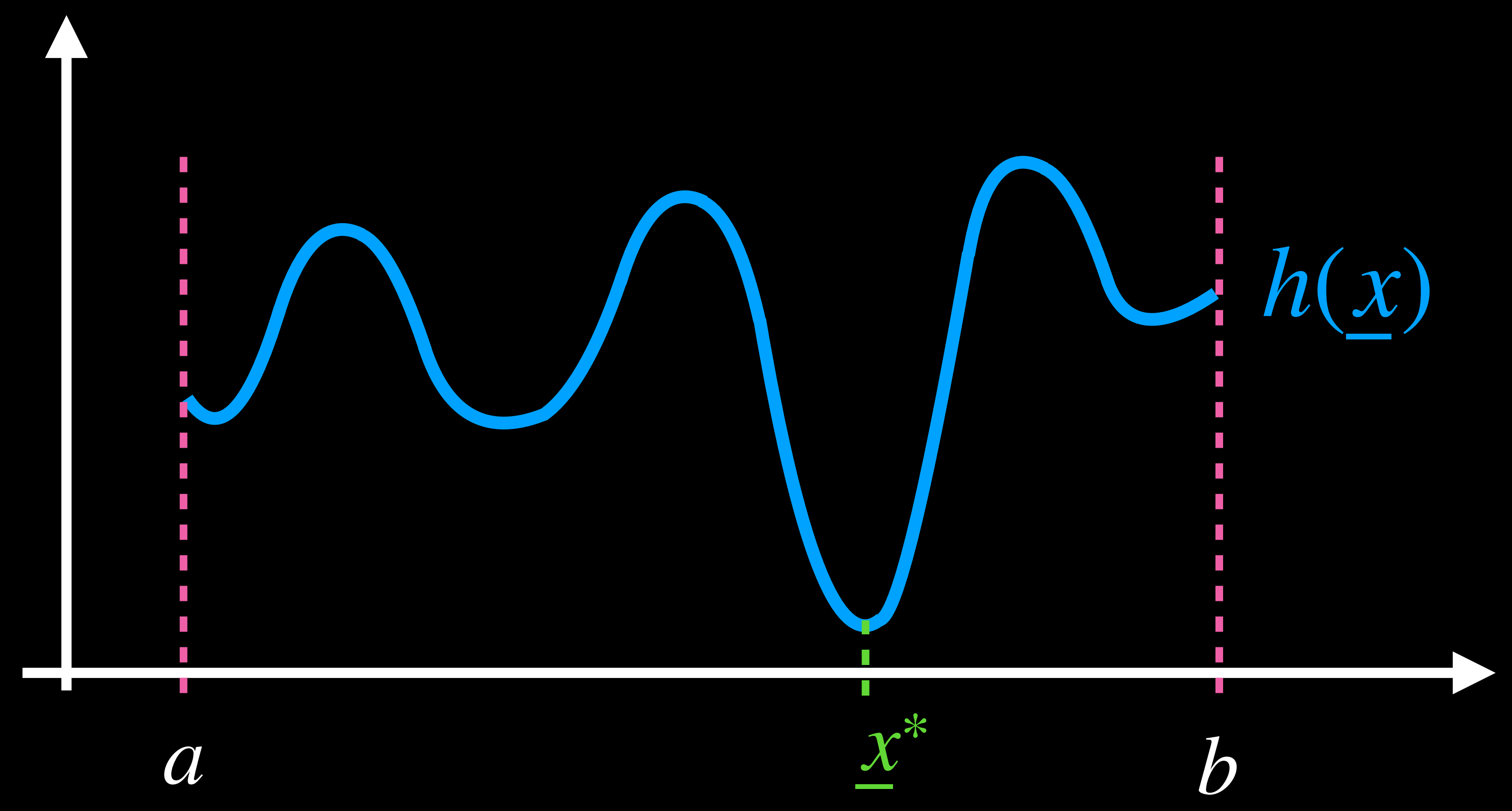
$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$



algorithm 2: stochastic descent

$$\underline{x}^* := \underset{\underline{x} \in \mathbb{S}}{\arg\min} \left(h(\underline{x}) \right)$$

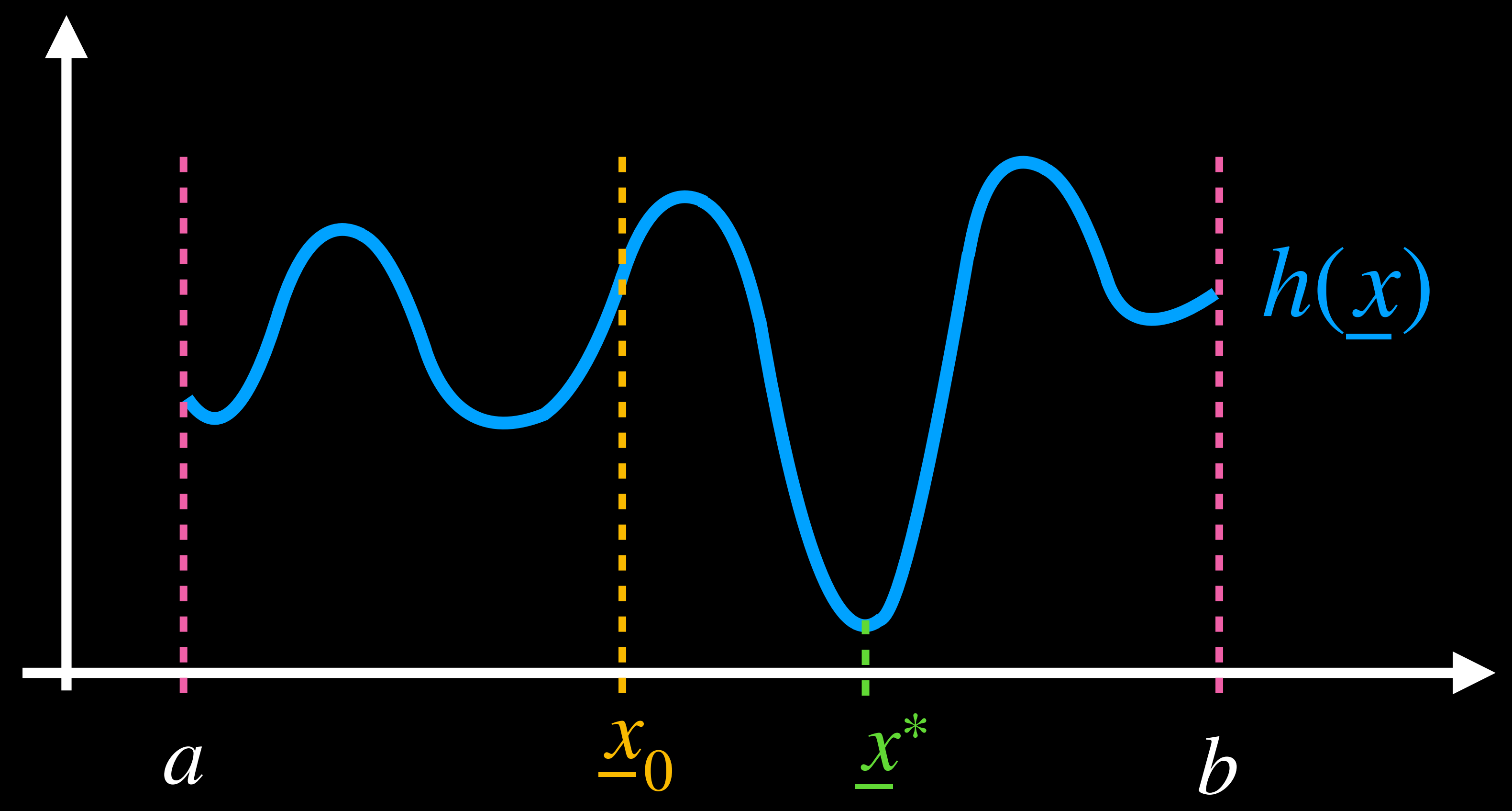
gradient descent:



algorithm 2: stochastic descent

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$

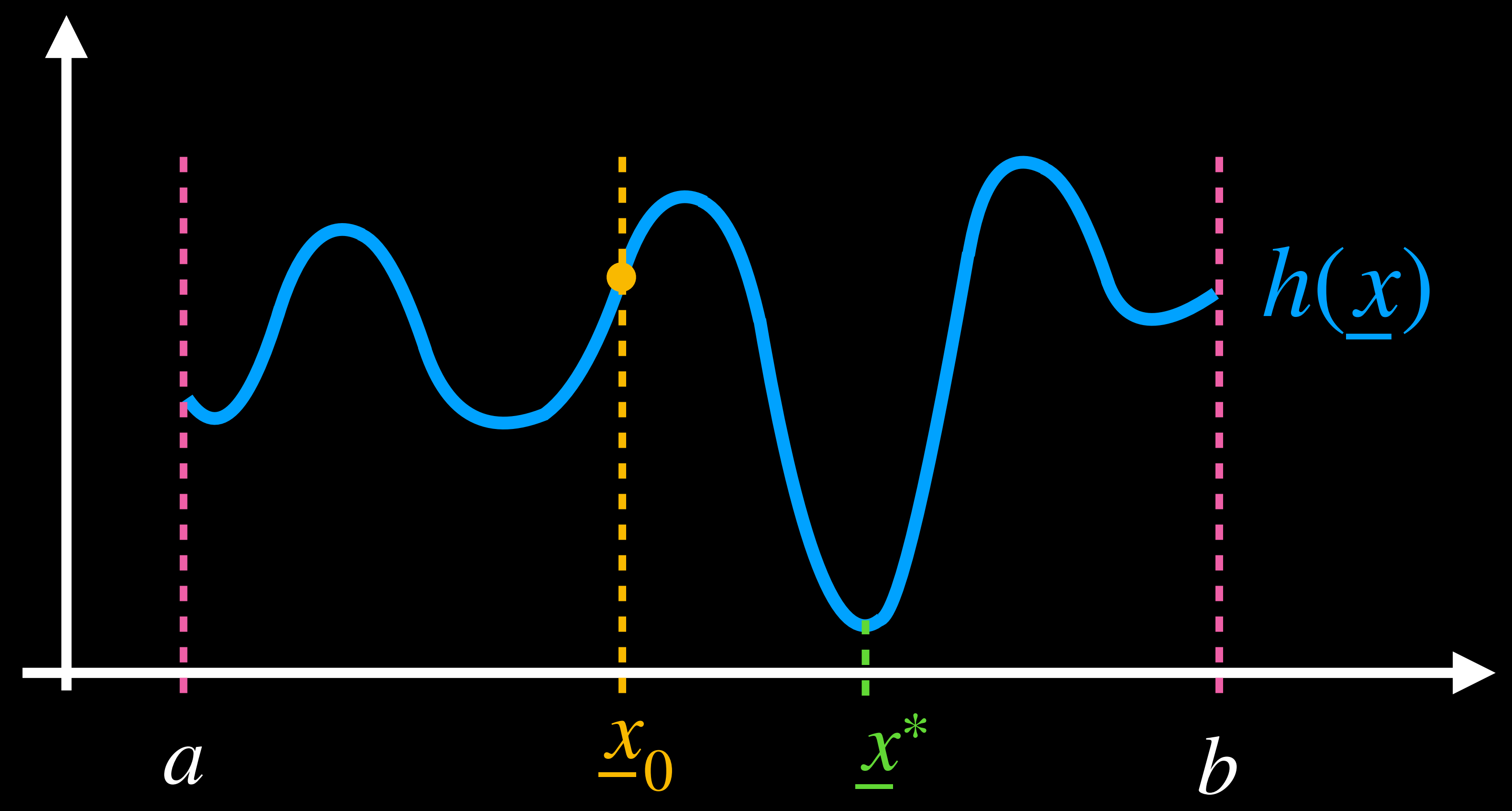
gradient descent:



algorithm 2: stochastic descent

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \big(h(\underline{x})\big)$$

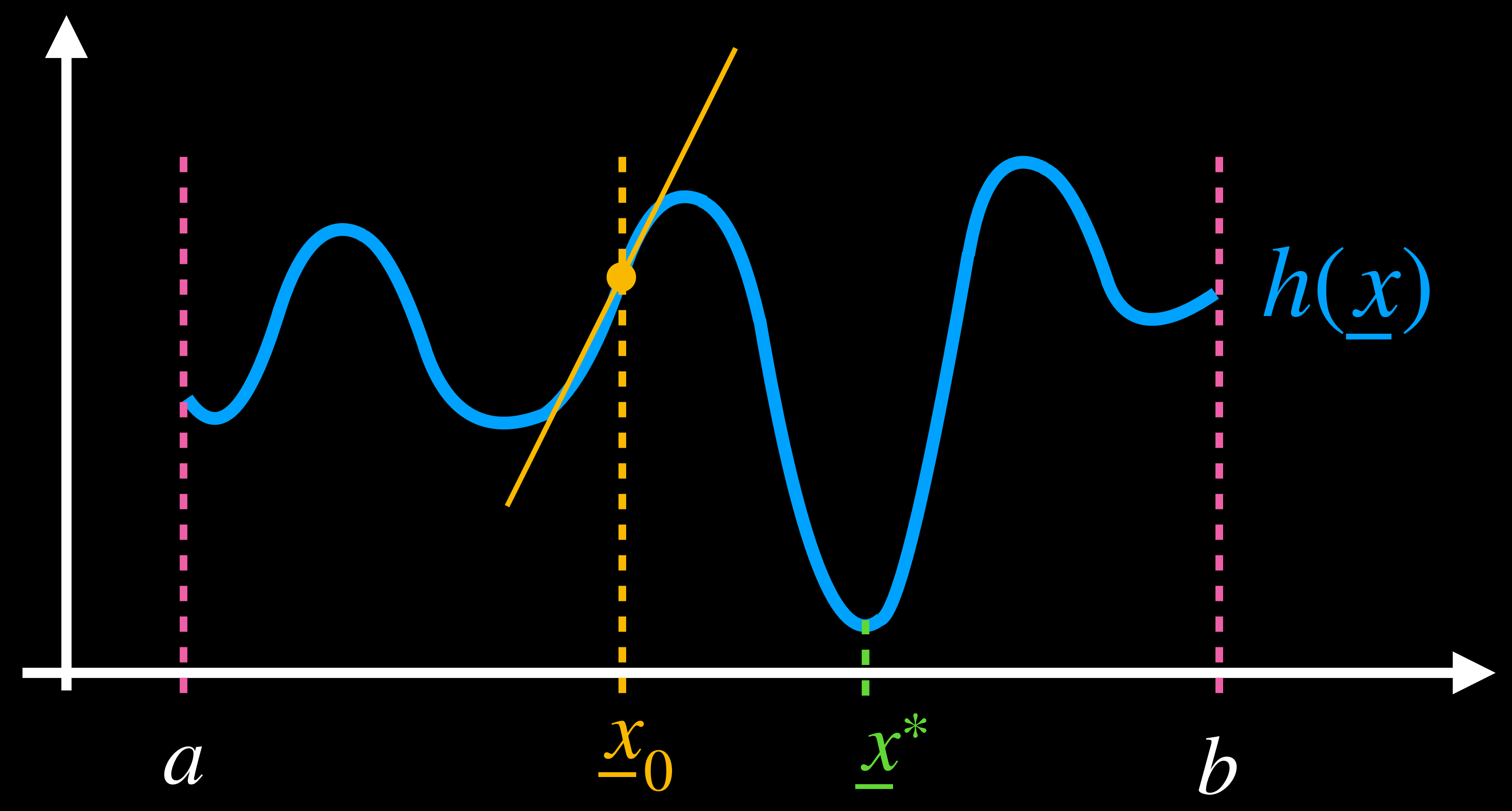
gradient descent:



algorithm 2: stochastic descent

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \big(h(\underline{x})\big)$$

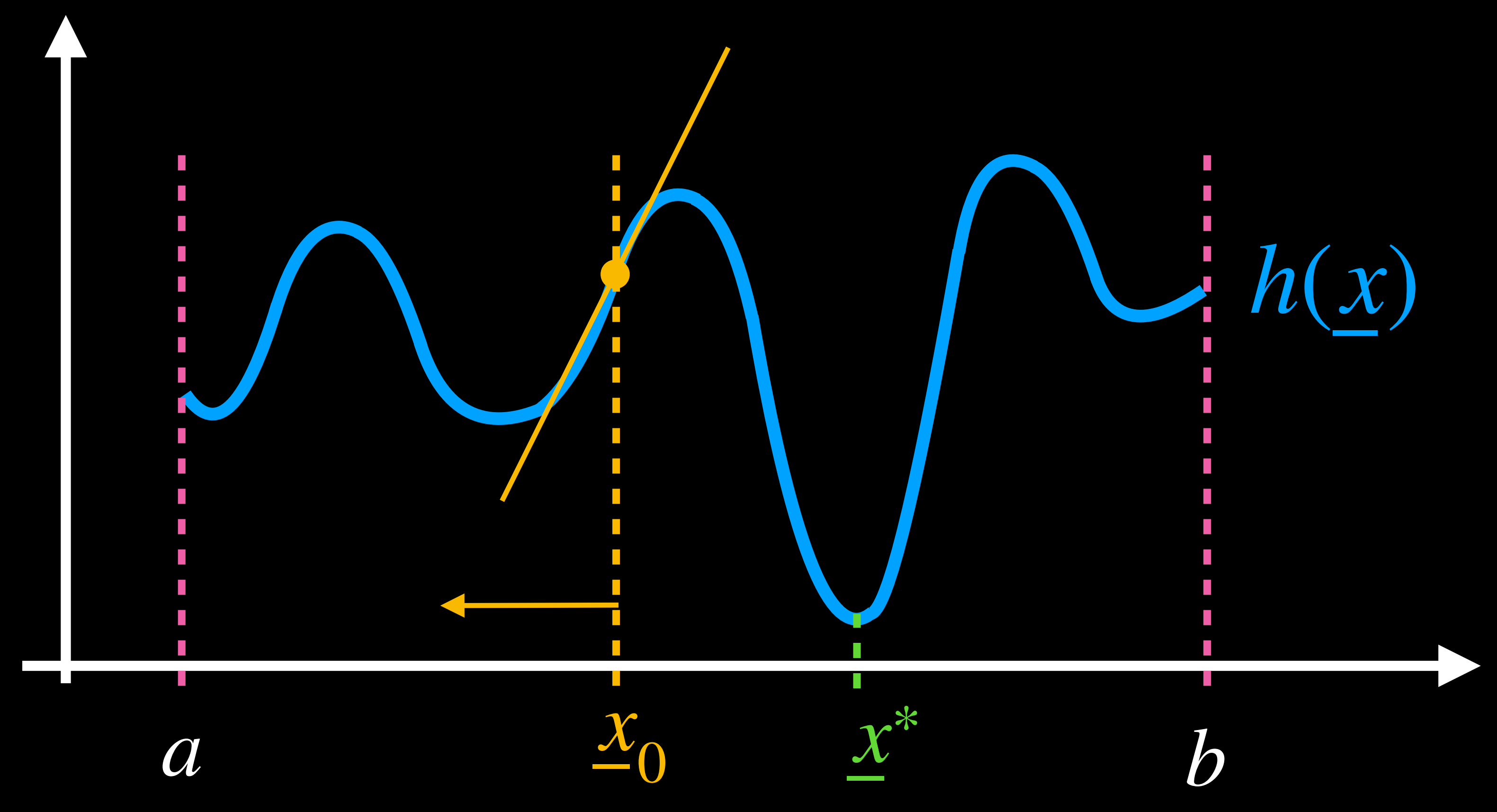
gradient descent:



algorithm 2: stochastic descent

$$\underline{x}^* := \underset{\underline{x} \in \mathbb{S}}{\arg\min} \left(h(\underline{x}) \right)$$

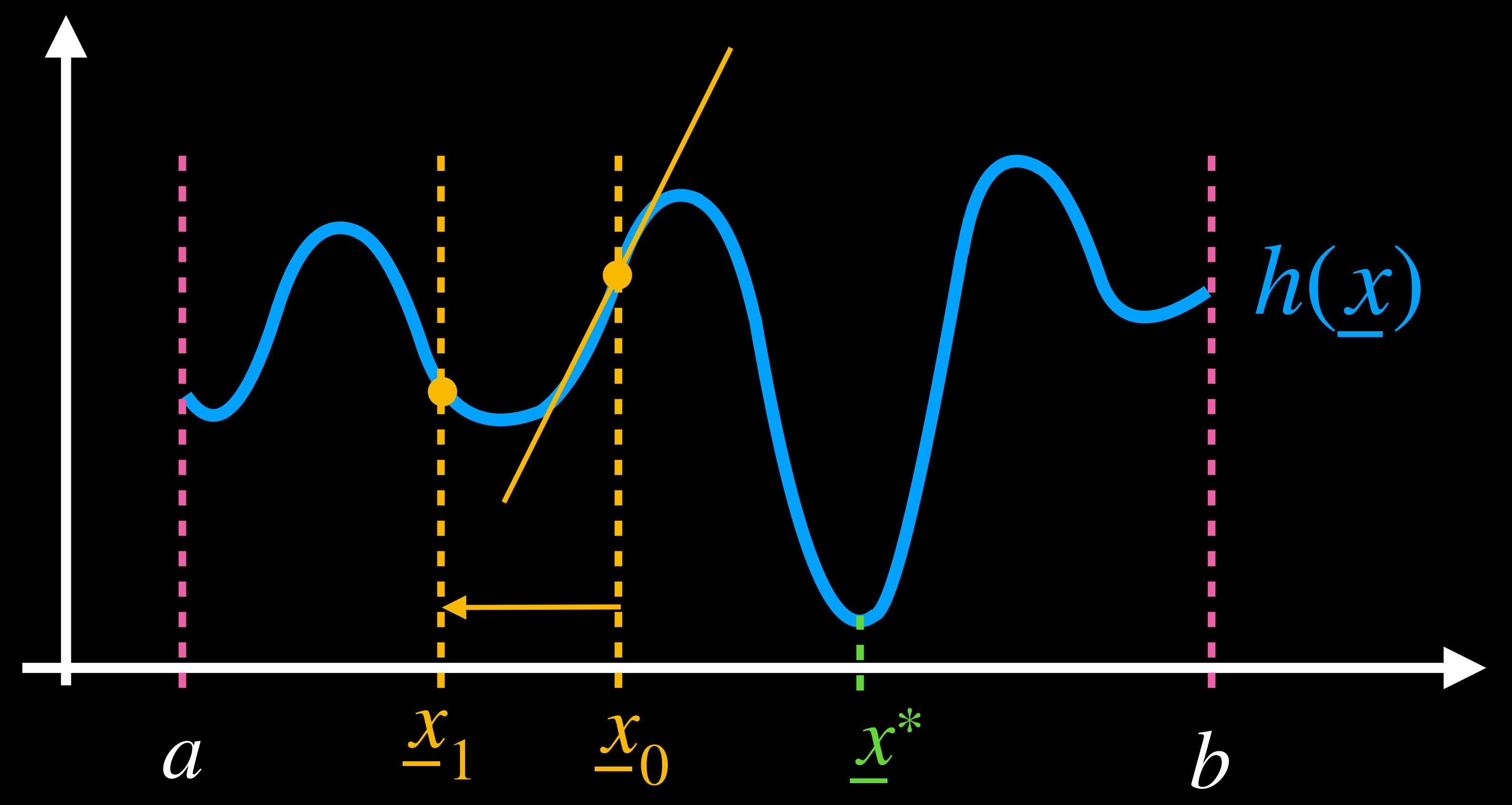
gradient descent:



algorithm 2: stochastic descent

$$\underline{x}^* := \underset{\underline{x} \in \mathbb{S}}{\arg\min} \left(h(\underline{x}) \right)$$

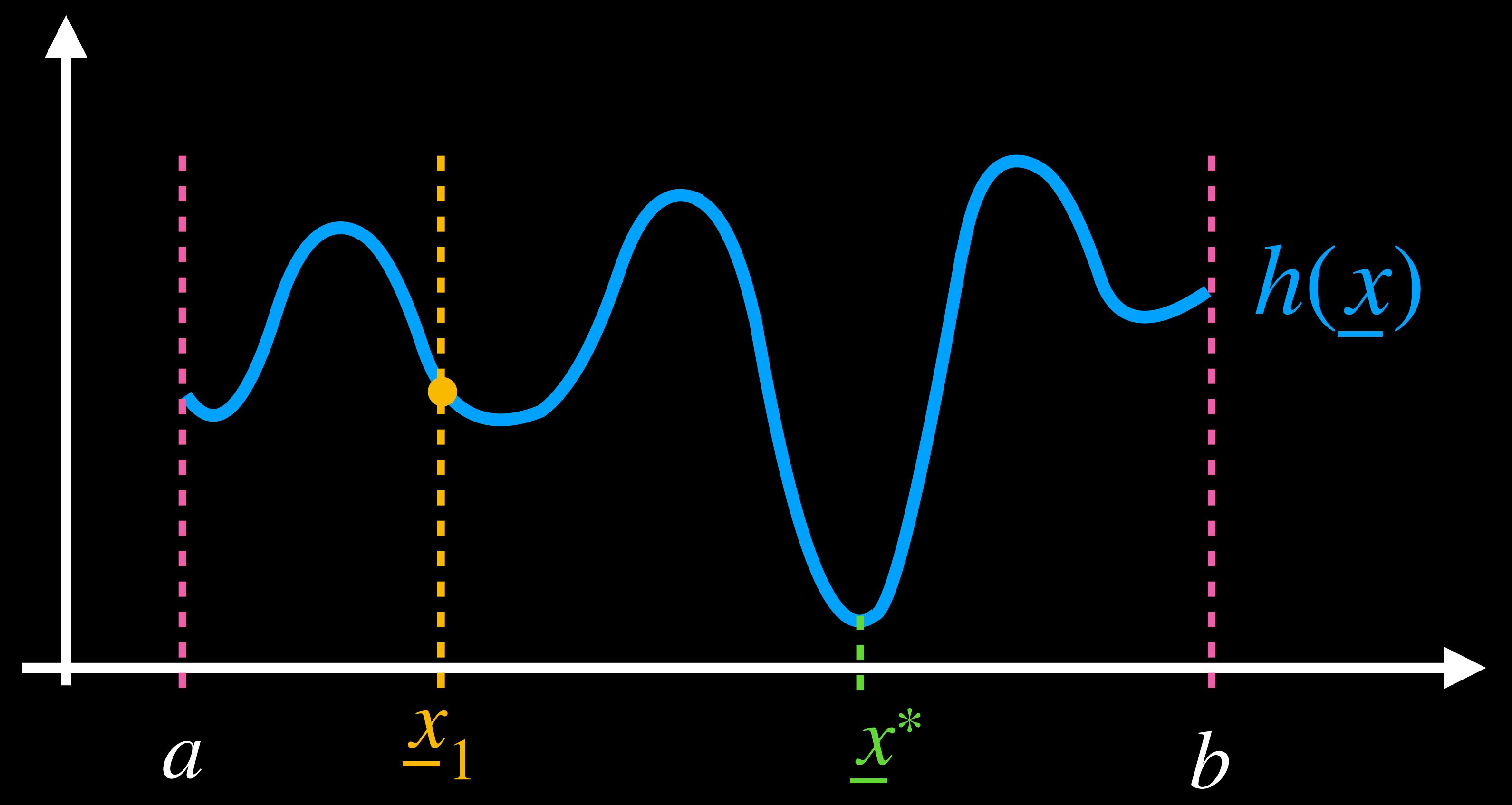
gradient descent:



algorithm 2: stochastic descent

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$

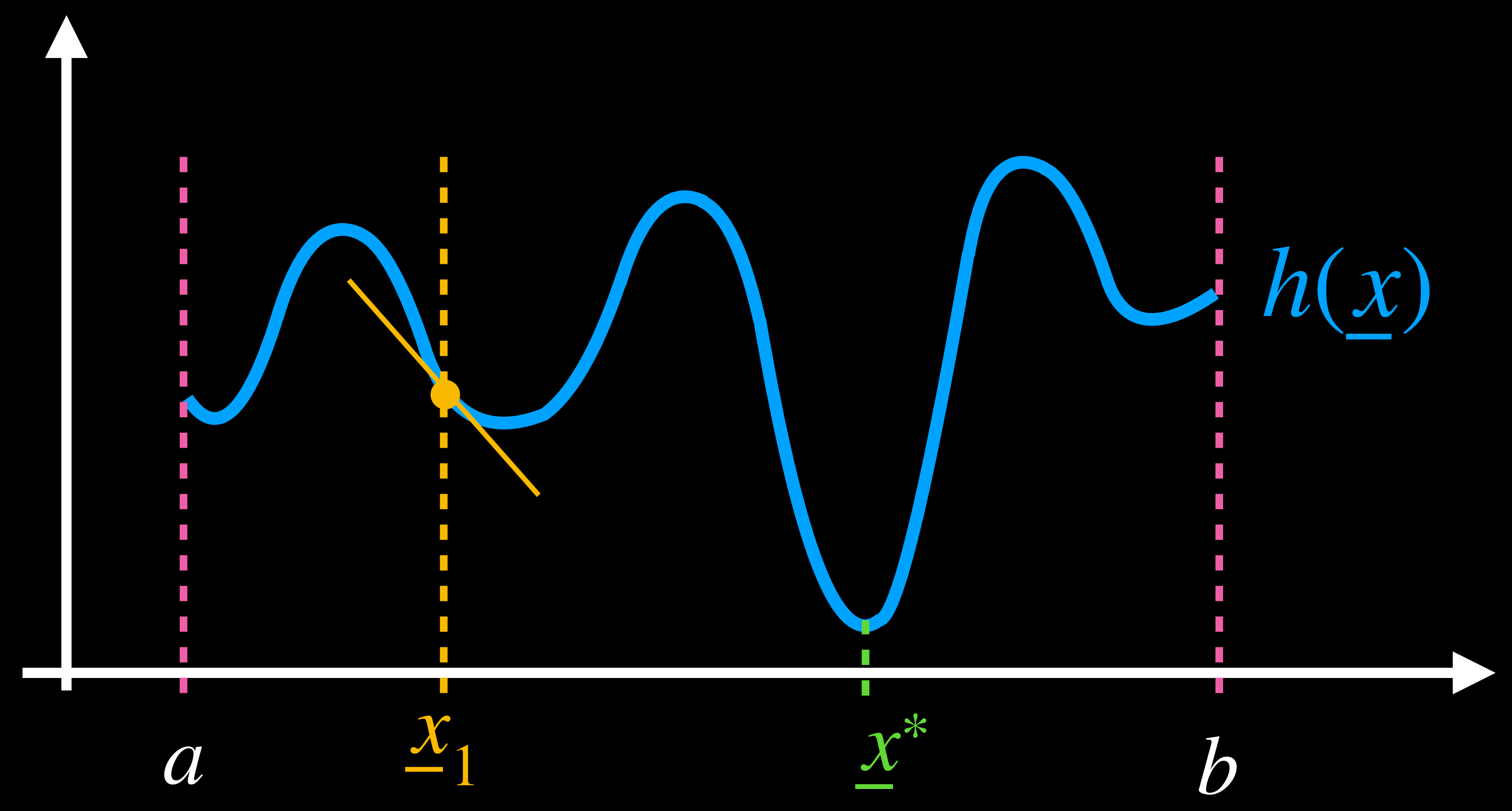
gradient descent:



algorithm 2: stochastic descent

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

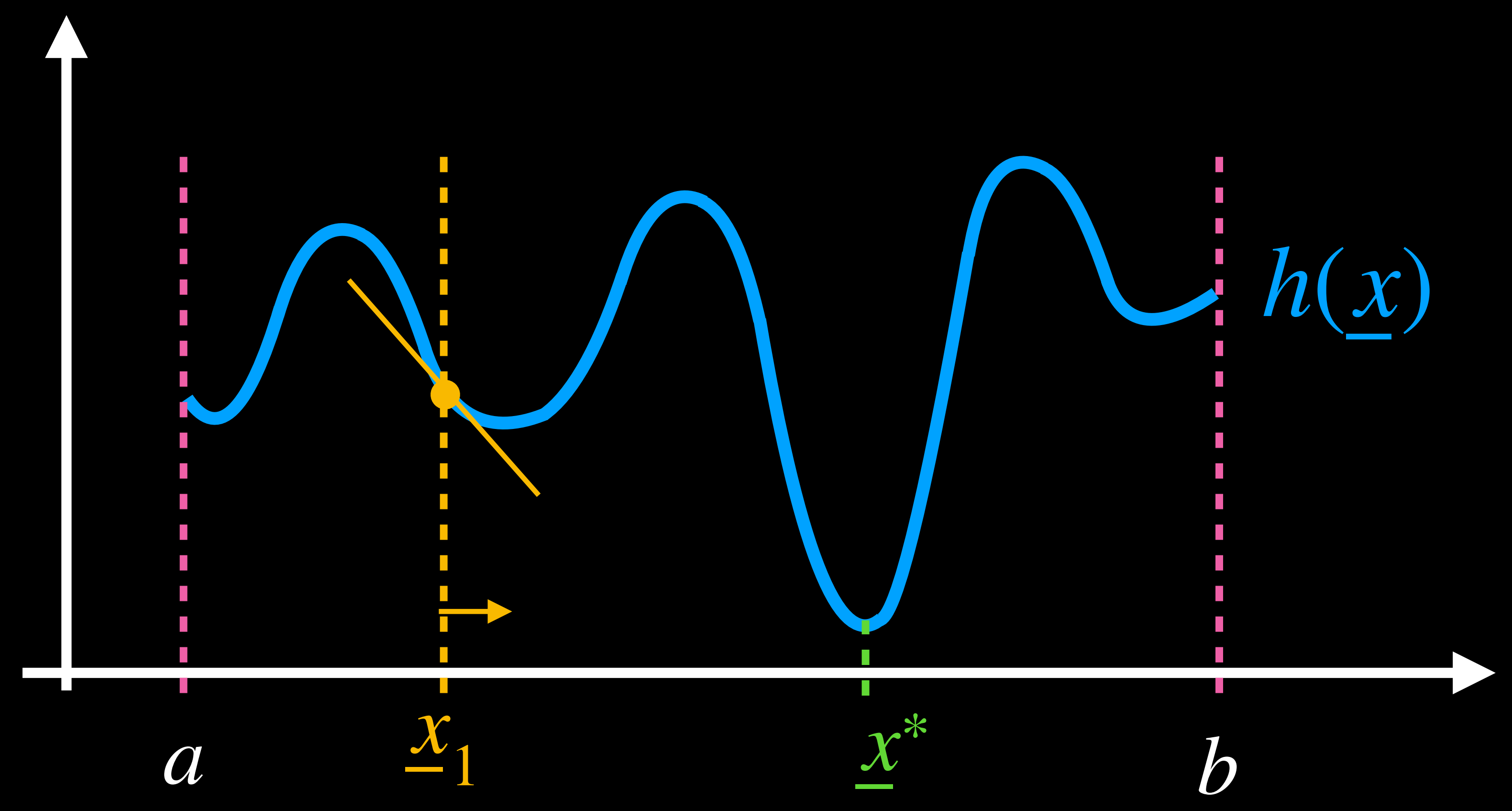
gradient descent:



algorithm 2: stochastic descent

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

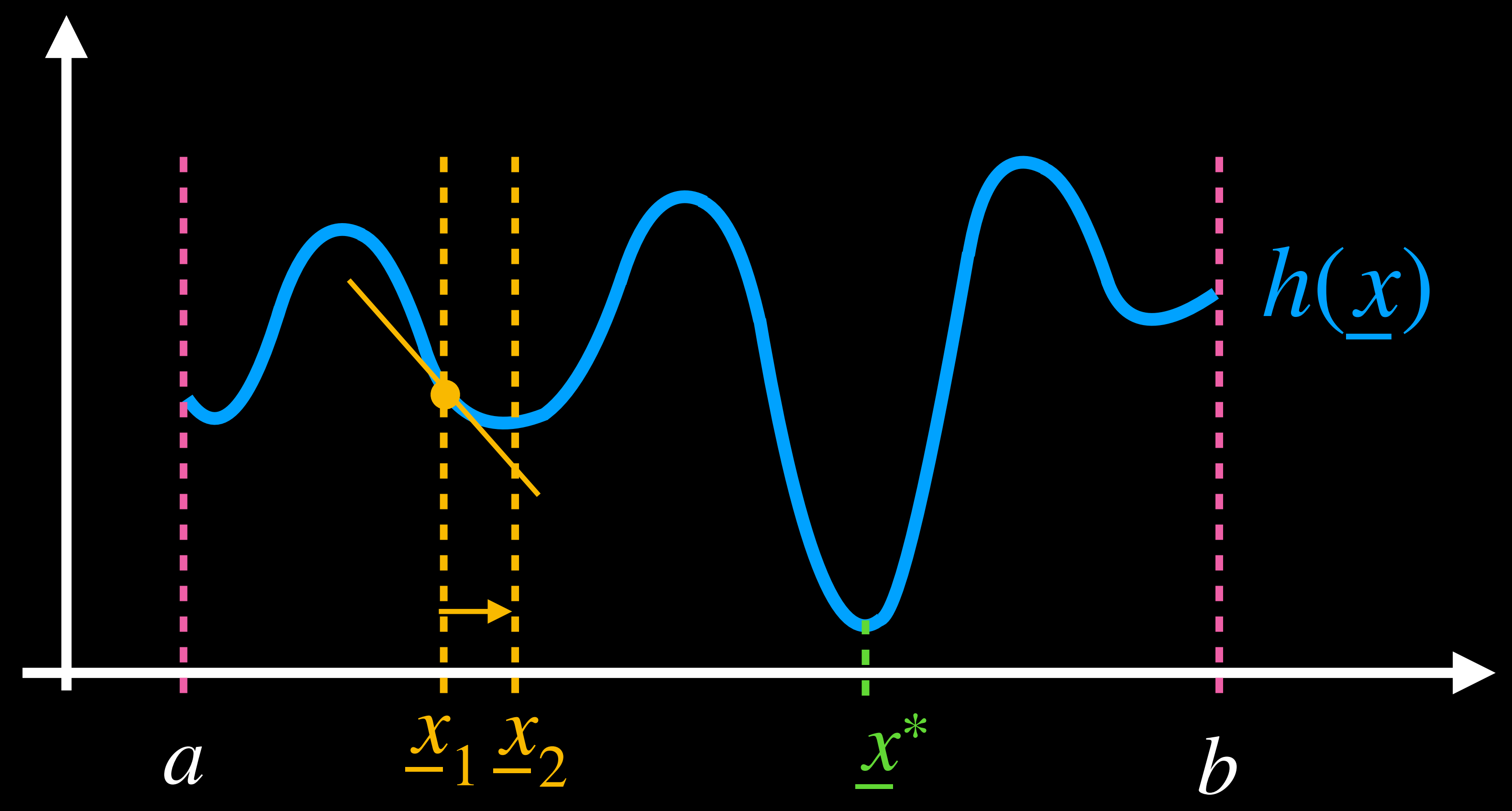
gradient descent:



algorithm 2: stochastic descent

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$

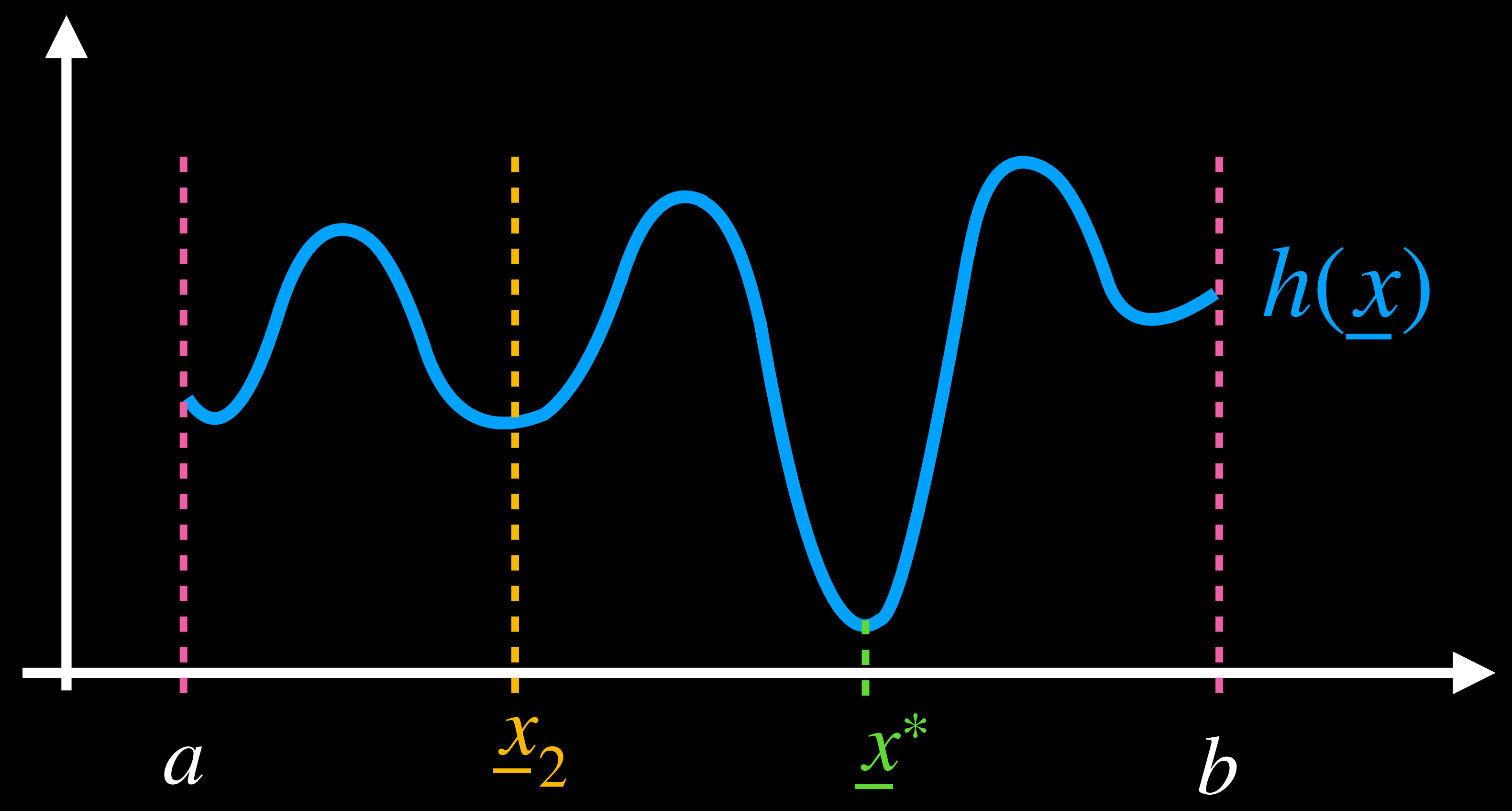
gradient descent:



algorithm 2: stochastic descent

$$\underline{x}^* := \underset{\underline{x} \in \mathbb{S}}{\arg\min} \left(h(\underline{x}) \right)$$

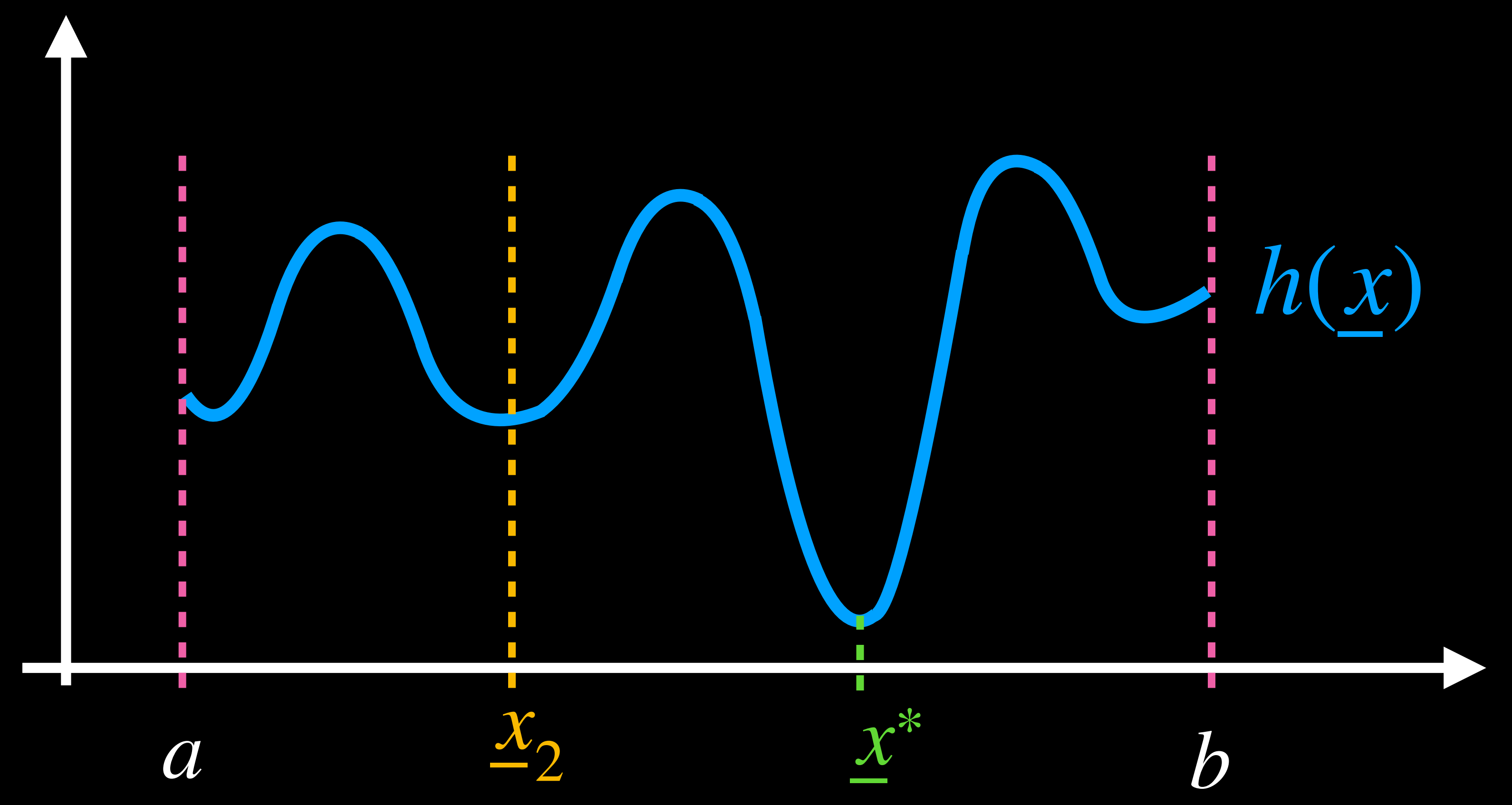
gradient descent:



algorithm 2: stochastic descent

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\,\big(h(\underline{x})\big)$$

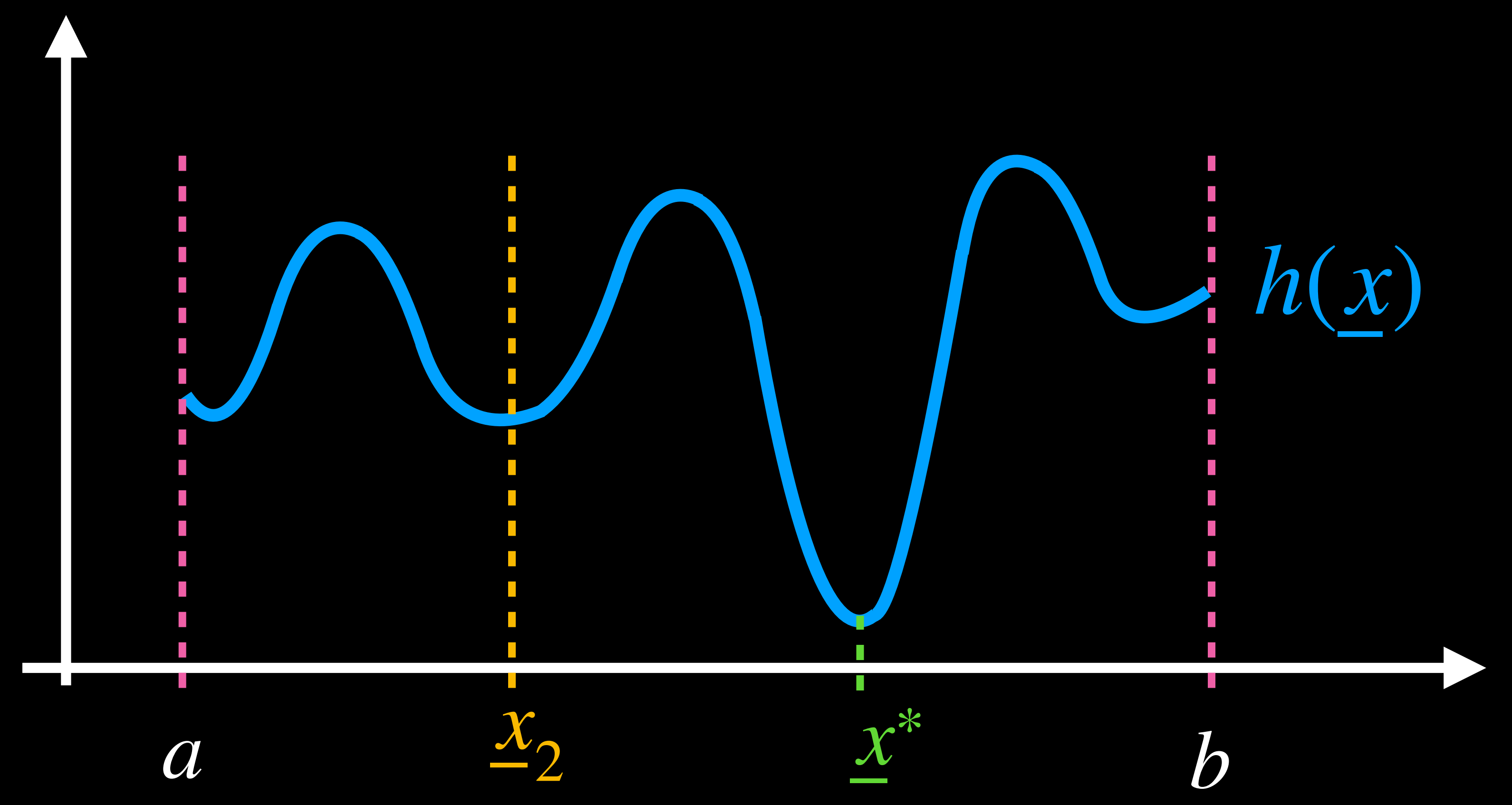
gradient descent: $\underline{x}_{t+1}$



algorithm 2: stochastic descent

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \big(h(\underline{x})\big)$$

gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \nabla h(\underline{x}_t)$

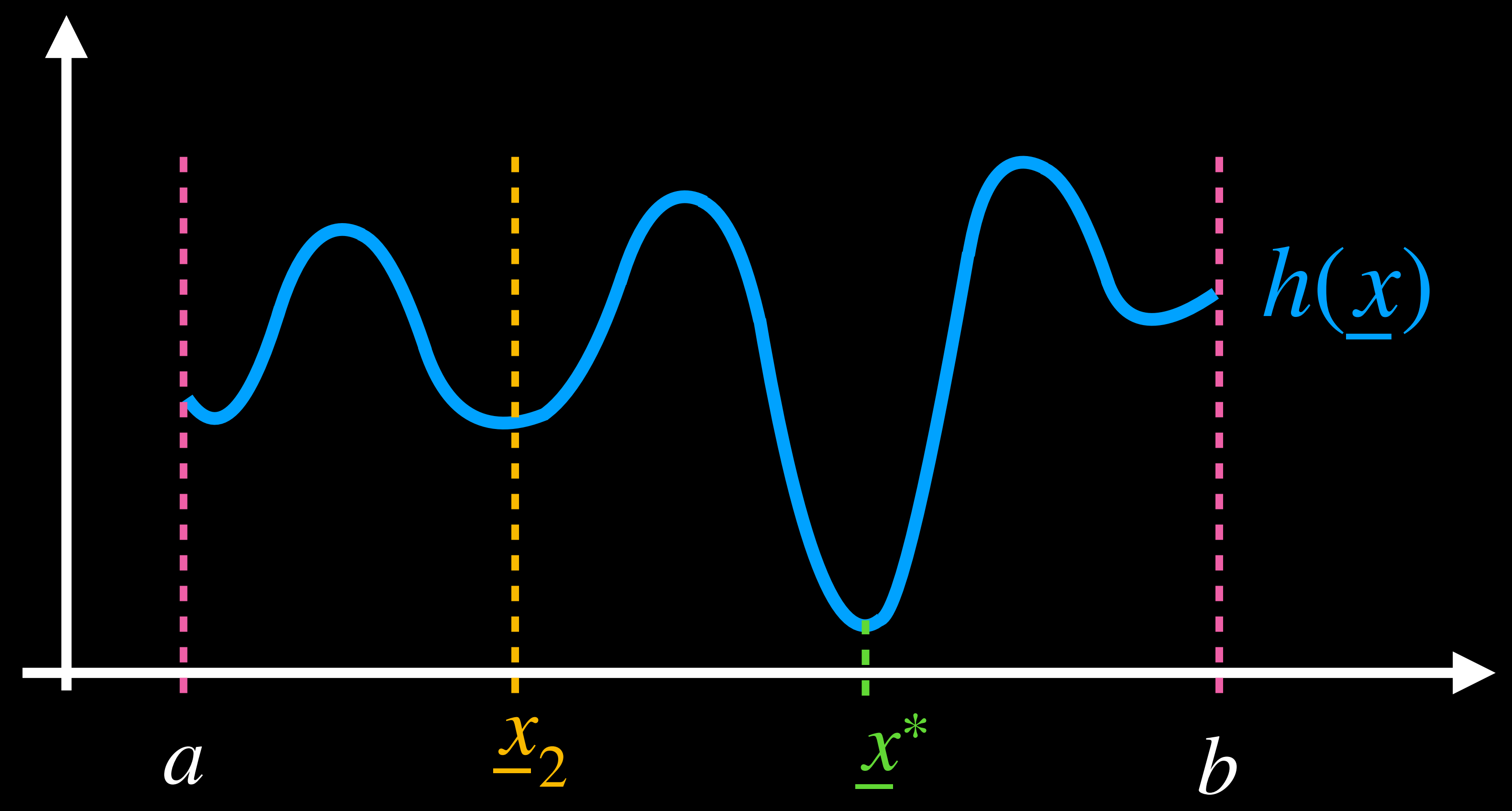


algorithm 2: stochastic descent

$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$

$t \in \{1,\ldots,n\}$ steps

gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \nabla h(\underline{x}_t)$



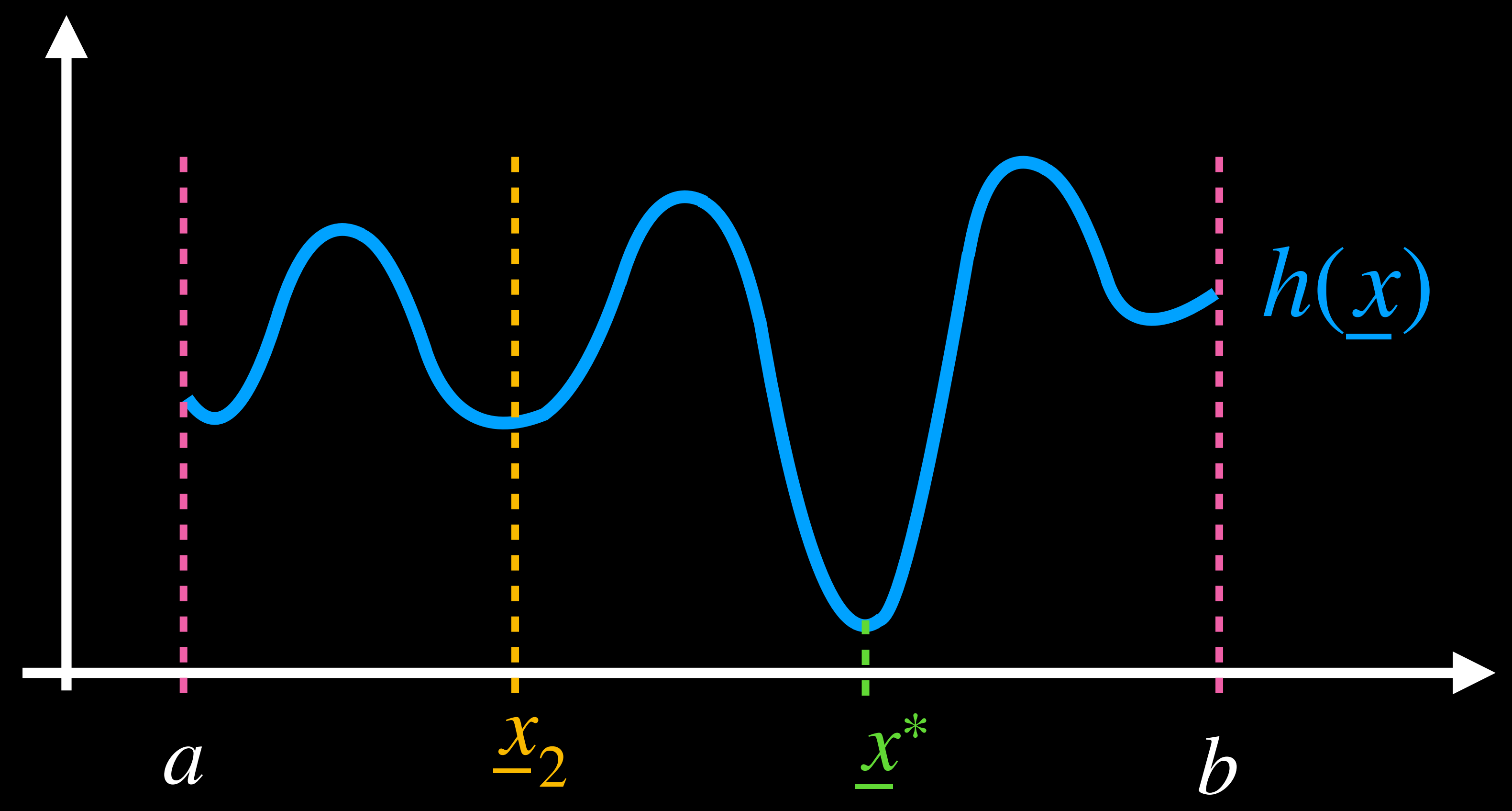
algorithm 2: stochastic descent

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \big(h(\underline{x})\big)$$

$t \in \{1,\ldots,n\}$ steps

$\alpha_t > 0$ step size

gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \nabla h(\underline{x}_t)$



algorithm 2: stochastic descent

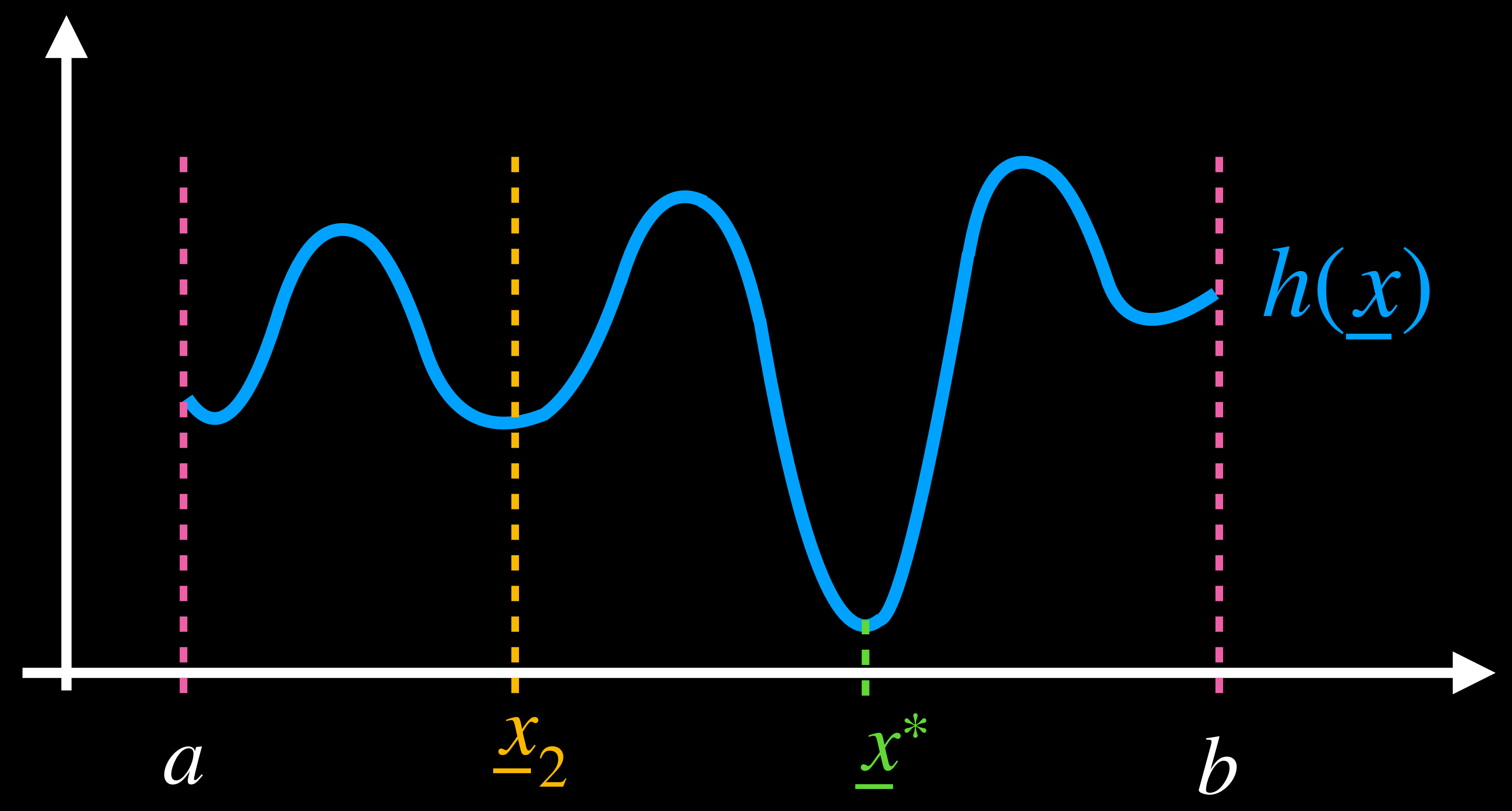
$\underline{x}^* := \underset{\underline{x} \in \mathbb{S}}{\arg\min} \left(h(\underline{x}) \right)$

gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \nabla h(\underline{x}_t)$

$t \in \{1, \ldots, n\}$ steps

$\alpha_t > 0$ step size

$-\nabla h(\underline{x}_t)$ vector of steepest decrease



algorithm 2: stochastic descent

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$

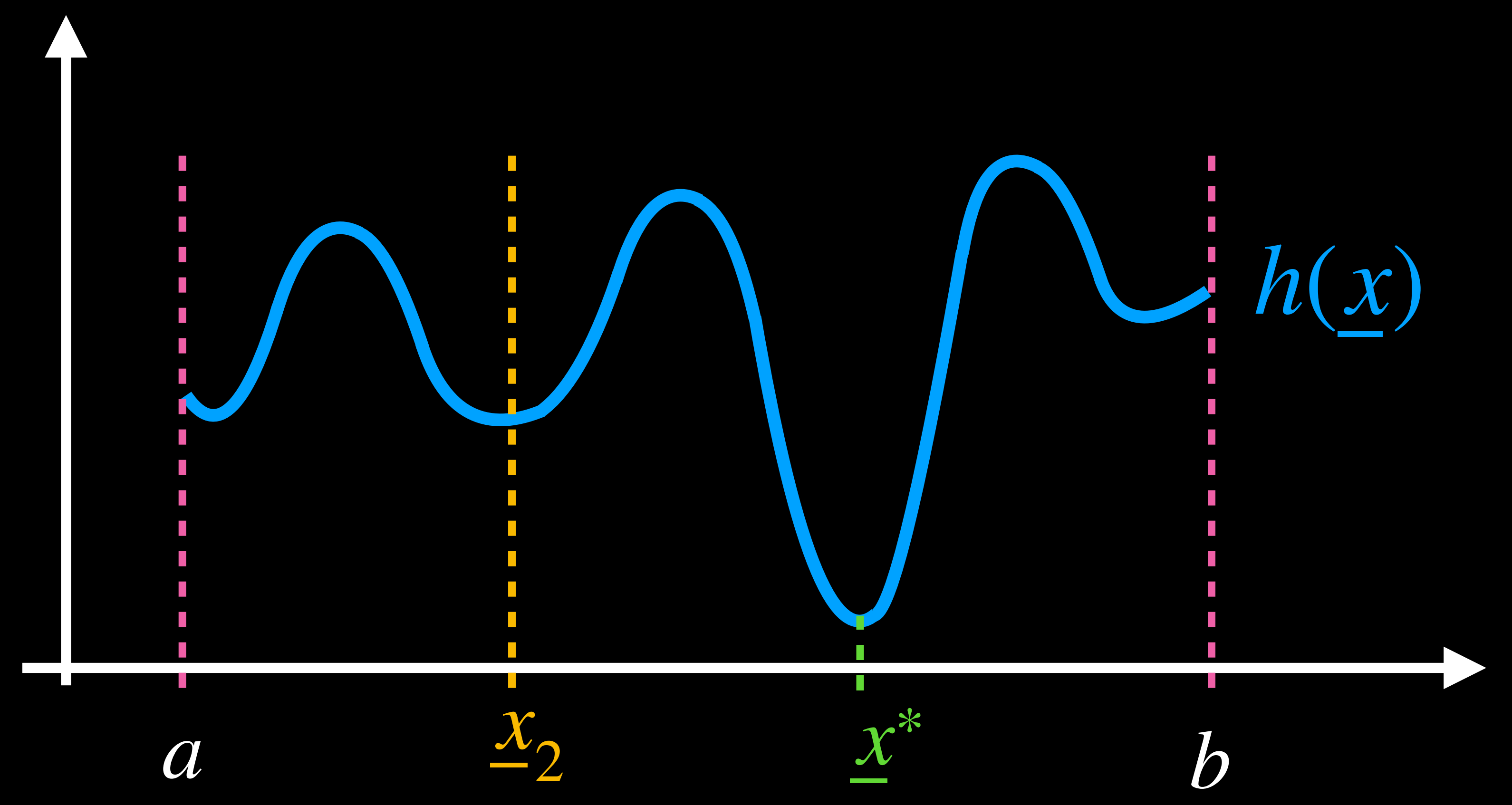
gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \nabla h(\underline{x}_t)$

$t \in \{1,\ldots,n\}$ steps

$\alpha_t > 0$ step size

$-\nabla h(\underline{x}_t)$ vector of steepest decrease

$\underline{x}_0$ starting point



algorithm 2: stochastic descent

$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \big(h(\underline{x})\big)$

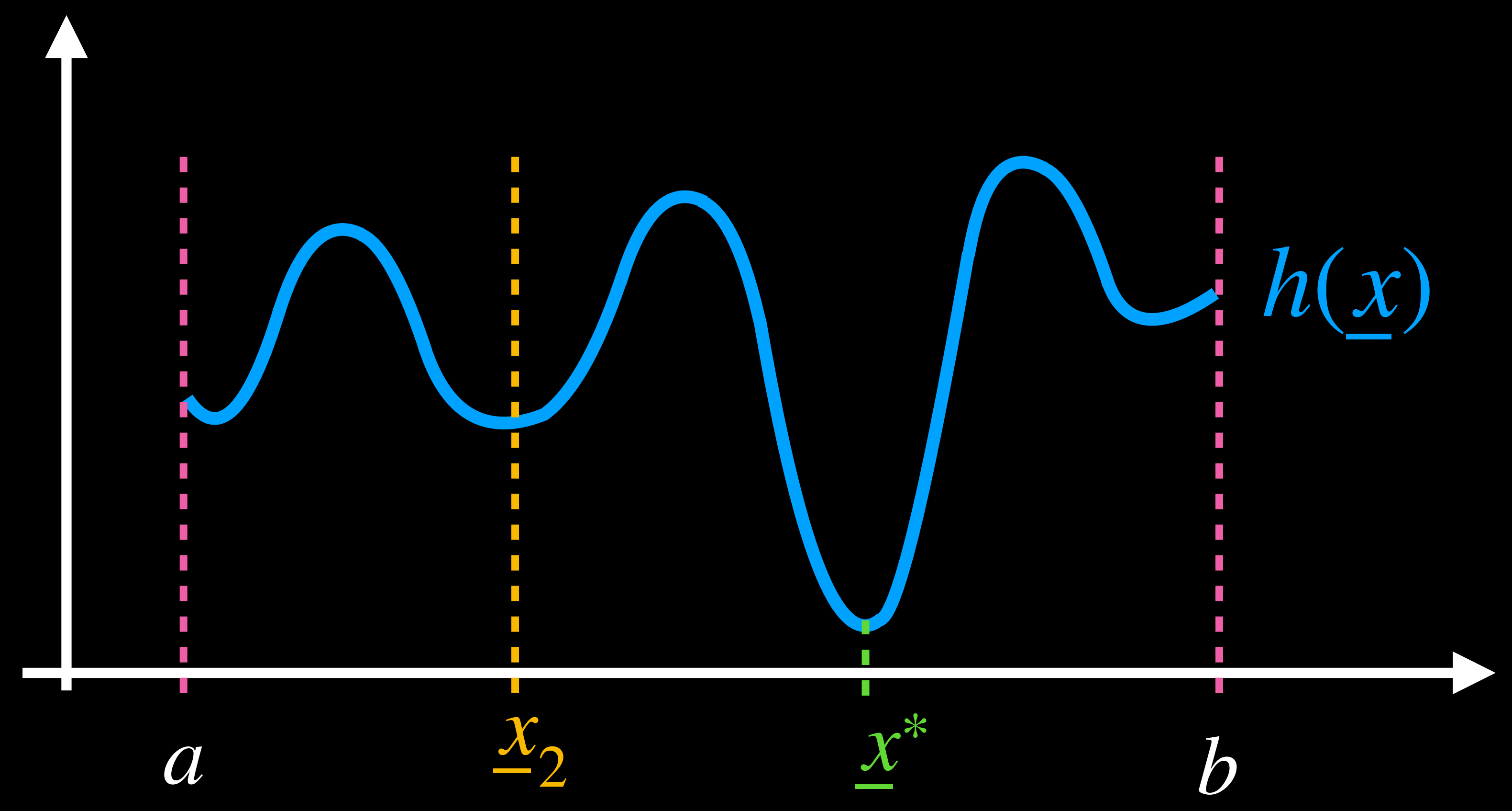
gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

$t \in \{1,\ldots,n\}$ steps

$\alpha_t > 0$ step size

$-\nabla h(\underline{x}_t)$ vector of steepest decrease

$\underline{x}_0$ starting point



algorithm 2: stochastic descent

$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$

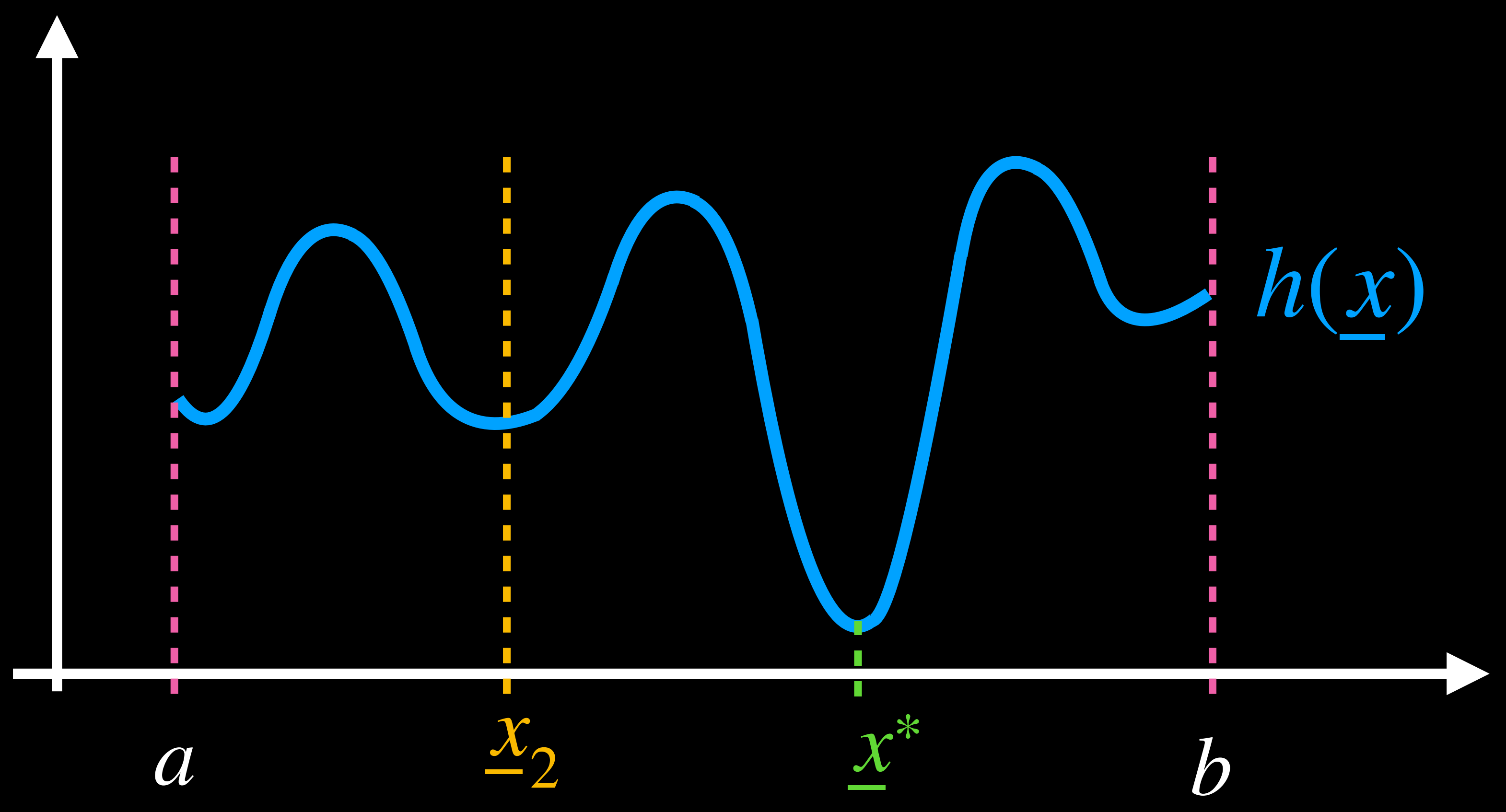
gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

$t \in \{1,\ldots,n\}$ steps

$\alpha_t > 0$ step size

$-\nabla h(\underline{x}_t)$ vector of steepest decrease

$\underline{x}_0$ starting point



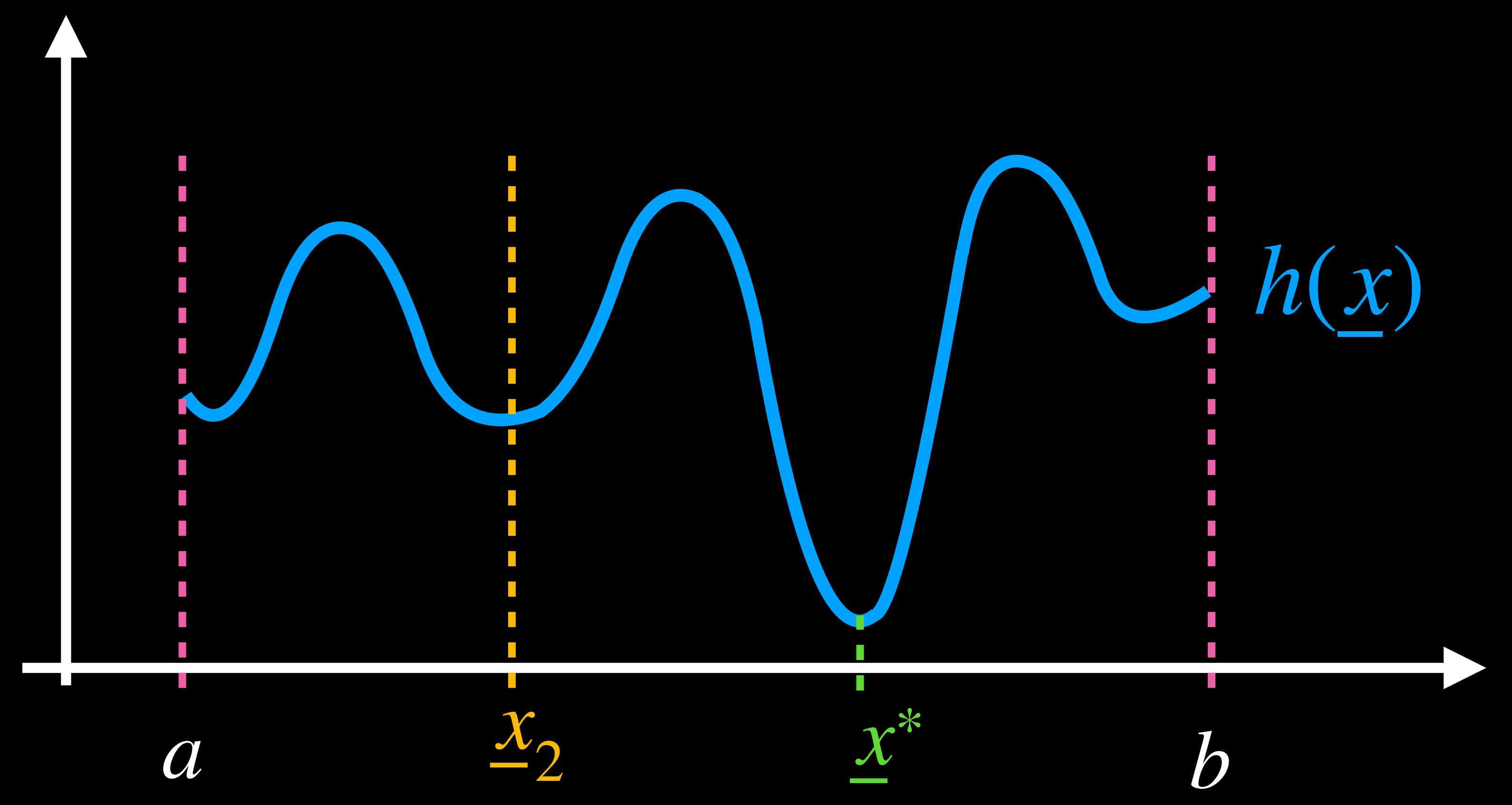
gradient:

algorithm 2: stochastic descent

$$\underline{x}^* := \arg\min_{\underline{x} \in \mathbb{S}} \left(h(\underline{x}) \right)$$

gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

$t \in \{1, \ldots, n\}$ steps
$\alpha_t > 0$ step size
$-\nabla h(\underline{x}_t)$ vector of steepest decrease
$\underline{x}_0$ starting point



gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix},$$

algorithm 2: stochastic descent

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$

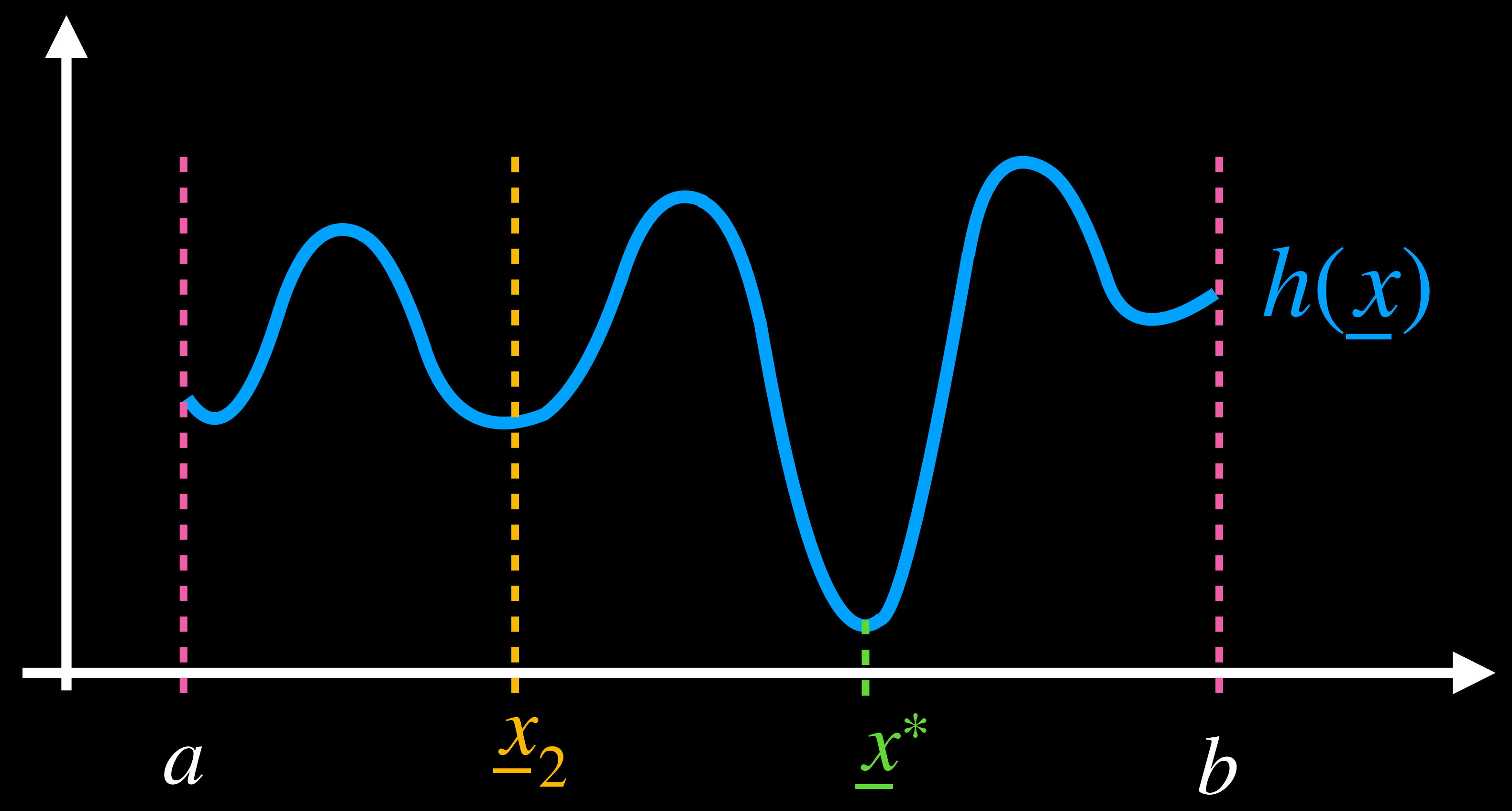
gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

$t \in \{1,\ldots,n\}$ steps

$\alpha_t > 0$ step size

$-\nabla h(\underline{x}_t)$ vector of steepest decrease

$\underline{x}_0$ starting point



gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix}\right) \cdot \frac{1}{2\Delta x}$$

algorithm 2: stochastic descent

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \big(h(\underline{x})\big)$$

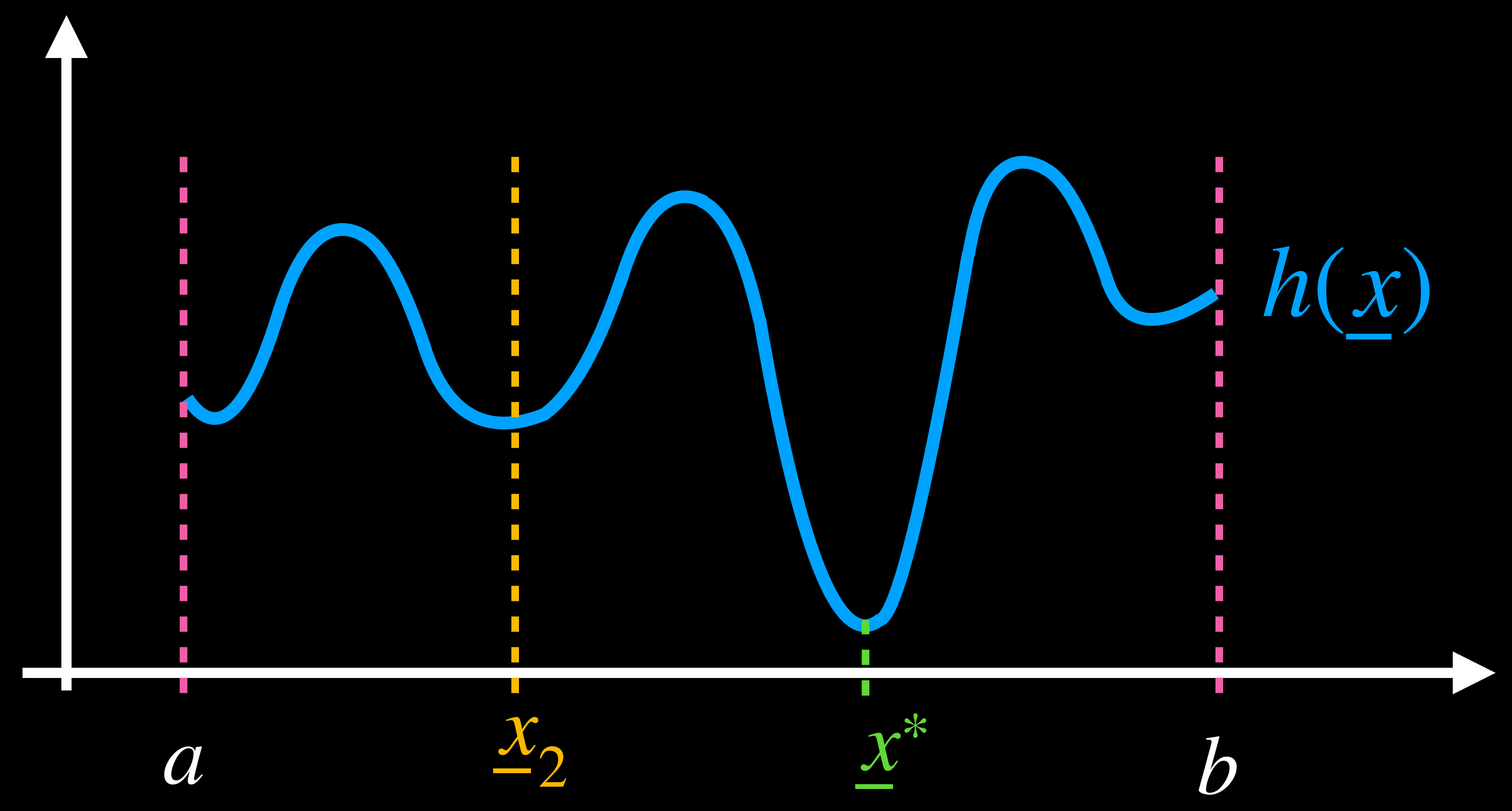
gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

$t \in \{1,\ldots,n\}$ steps

$\alpha_t > 0$ step size

$-\nabla h(\underline{x}_t)$ vector of steepest decrease

$\underline{x}_0$ starting point

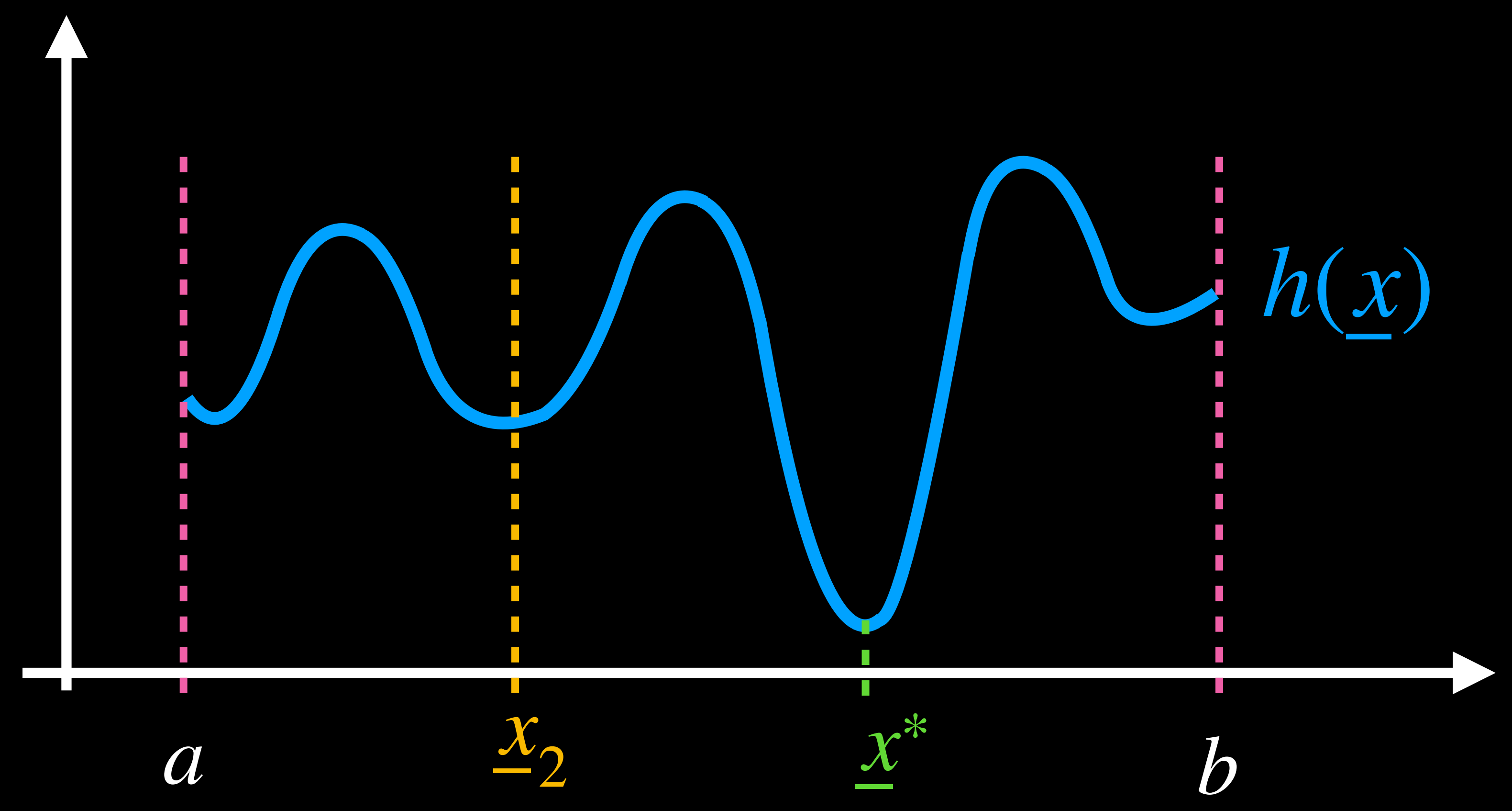


gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix}\right) \cdot \frac{1}{2\Delta x}$$

→ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

algorithm 2: stochastic descent



$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

$t \in \{1,\ldots,n\}$ steps

$\alpha_t > 0$ step size

$-\nabla h(\underline{x}_t)$ vector of steepest decrease

$\underline{x}_0$ starting point

gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix}\right) \cdot \frac{1}{2\Delta x}$$

$\rightarrow$ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

stochastic descent:

algorithm 2: stochastic descent

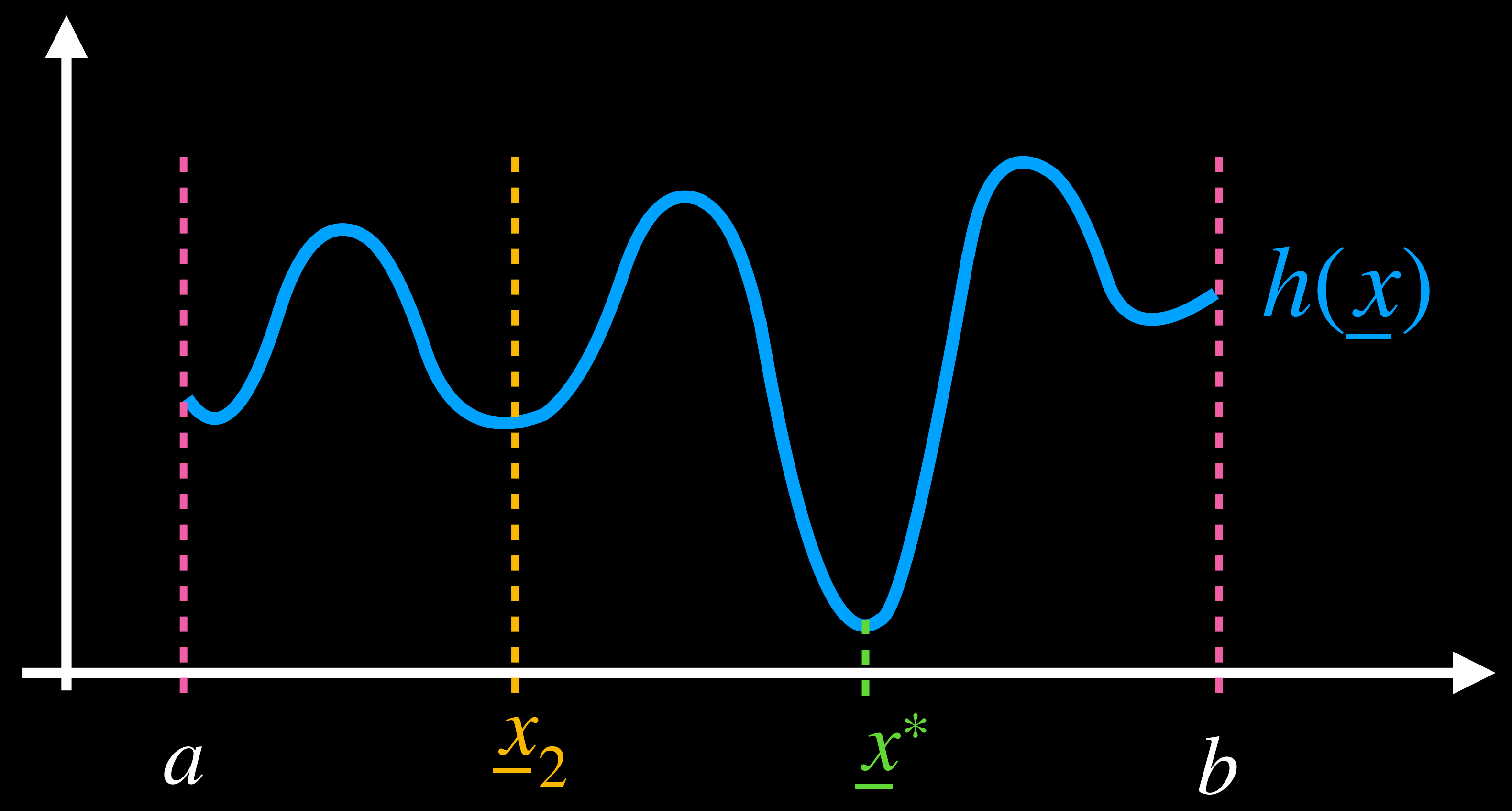
$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

$t \in \{1,\ldots,n\}$ steps

$\alpha_t > 0$ step size

$-\nabla h(\underline{x}_t)$ vector of steepest decrease

$\underline{x}_0$ starting point



gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} \right) \cdot \frac{1}{2\Delta x}$$

→ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

stochastic descent:

$\underline{X}_{t+1}$

algorithm 2: stochastic descent

$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$

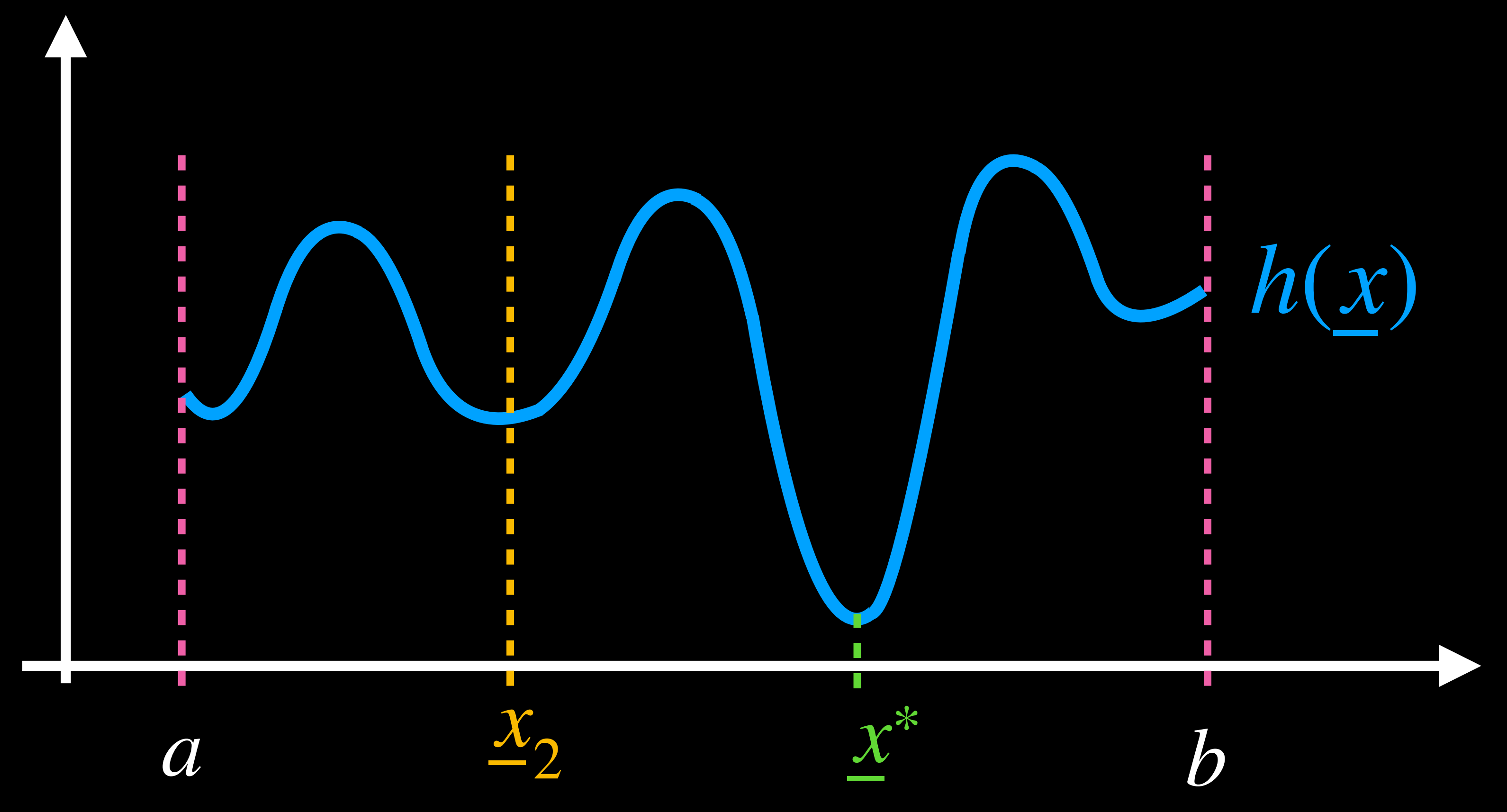
gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

$t \in \{1,\ldots,n\}$ steps

$\alpha_t > 0$ step size

$-\nabla h(\underline{x}_t)$ vector of steepest decrease

$\underline{x}_0$ starting point



gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix}\right) \cdot \frac{1}{2\Delta x}$$

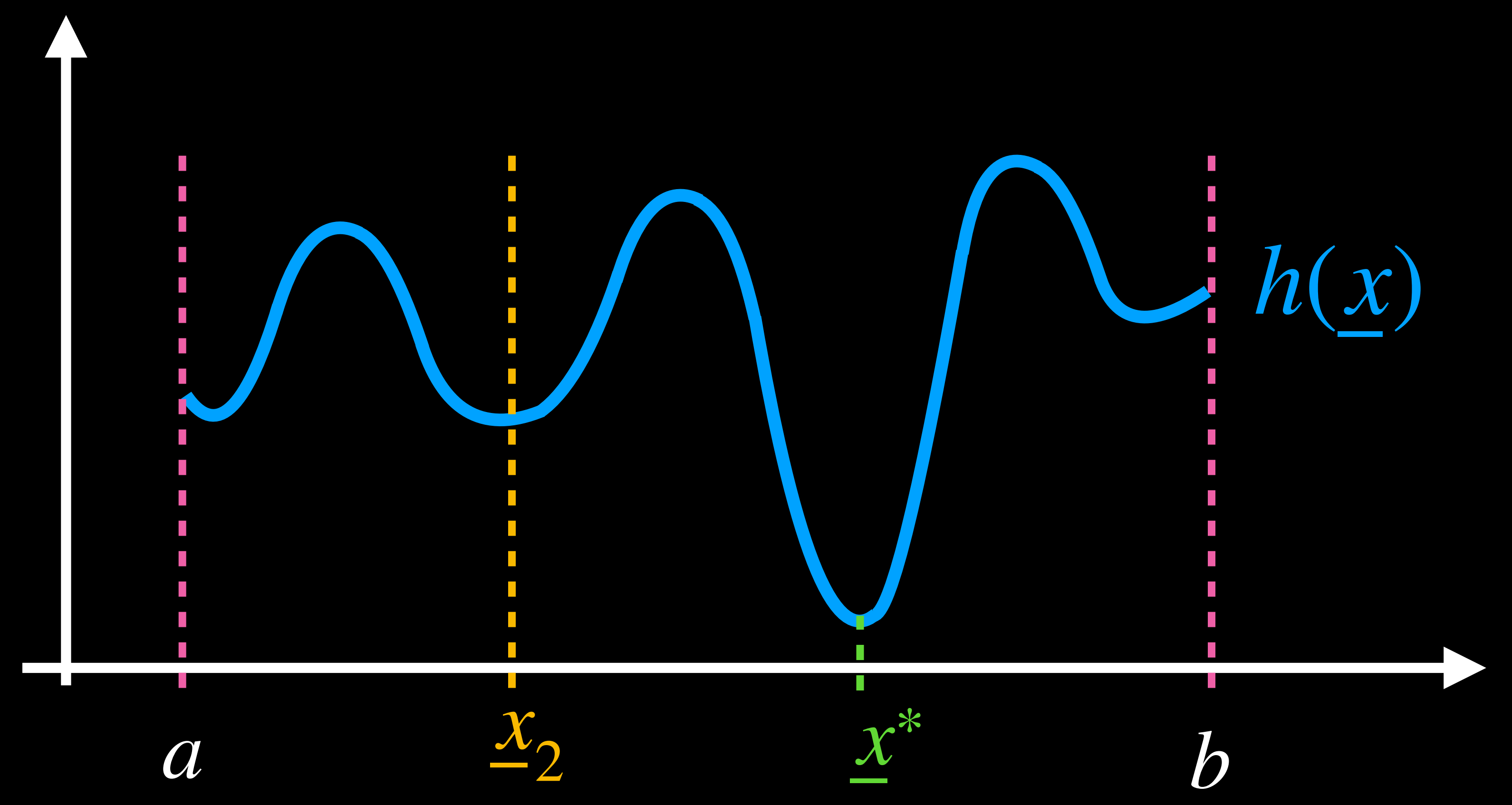
→ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

stochastic descent:

$$\underline{X}_{t+1} = \underline{X}_t + \alpha_t \underline{U}_t$$

algorithm 2: stochastic descent

$\underline{x}^* := \underset{\underline{x} \in \mathbb{S}}{\arg\min} \left(h(\underline{x}) \right)$



$t \in \{1, \ldots, n\}$ steps

$\alpha_t > 0$ step size

$-\nabla h(\underline{x}_t)$ vector of steepest decrease

$\underline{x}_0$ starting point

gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h \begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h \begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} \right) \cdot \frac{1}{2 \Delta x}$$

$\rightarrow$ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

stochastic descent:

$$\underline{X}_{t+1} = \underline{X}_t + \alpha_t \underline{U}_t \qquad \underline{U}_t \sim \mathcal{U}(\mathbb{S})$$

algorithm 2: stochastic descent

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\big(h(\underline{x})\big)$$

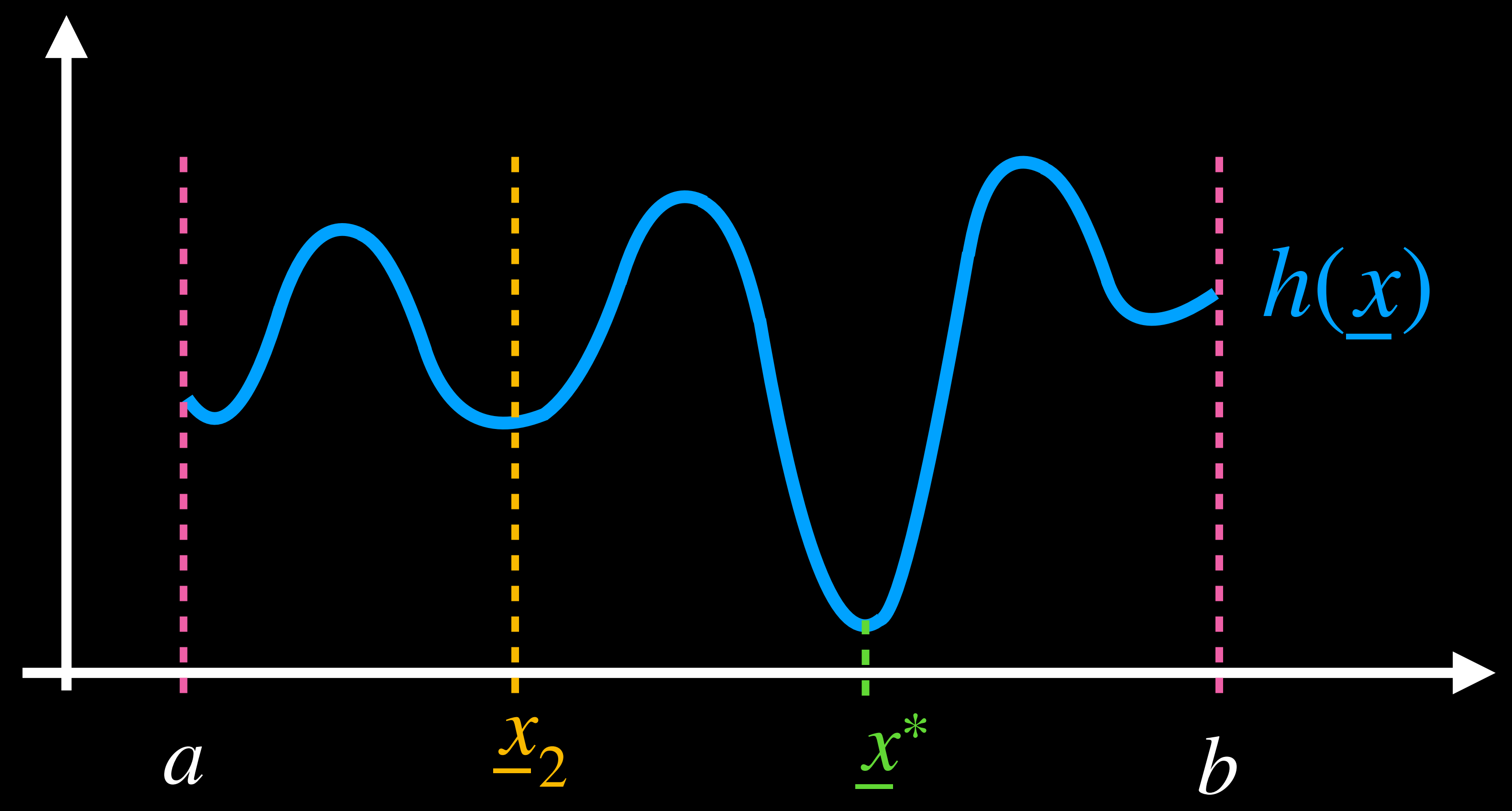
gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

$t \in \{1,\ldots,n\}$ steps

$\alpha_t > 0$ step size

$-\nabla h(\underline{x}_t)$ vector of steepest decrease

$\underline{x}_0$ starting point



gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} \right) \cdot \frac{1}{2\Delta x}$$

→ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

stochastic descent:

$$\underline{X}_{t+1} = \underline{X}_t + \alpha_t \underline{U}_t$$

$\underline{U}_t \sim \mathcal{U}(\mathbb{S})$

$\mathbb{S}$ be the d-dimensional unit sphere

algorithm 2: stochastic descent

$$\underline{x}^* := \underset{\underline{x} \in \mathbb{S}}{\arg\min} \big(h(\underline{x})\big)$$

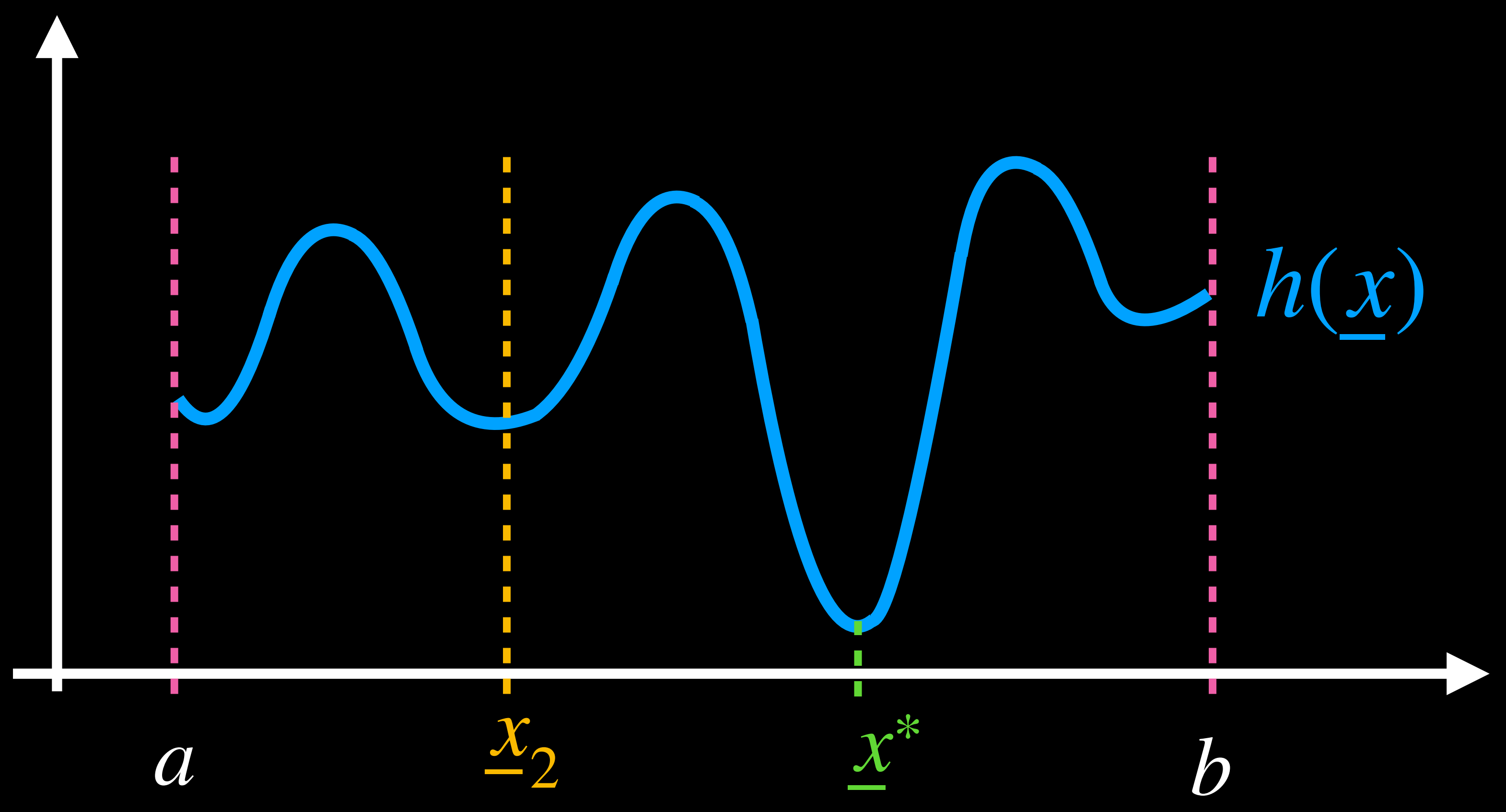
gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

$t \in \{1, \ldots, n\}$ steps

$\alpha_t > 0$ step size

$-\nabla h(\underline{x}_t)$ vector of steepest decrease

$\underline{x}_0$ starting point



gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} \right) \cdot \frac{1}{2\Delta x}$$

$\rightarrow$ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

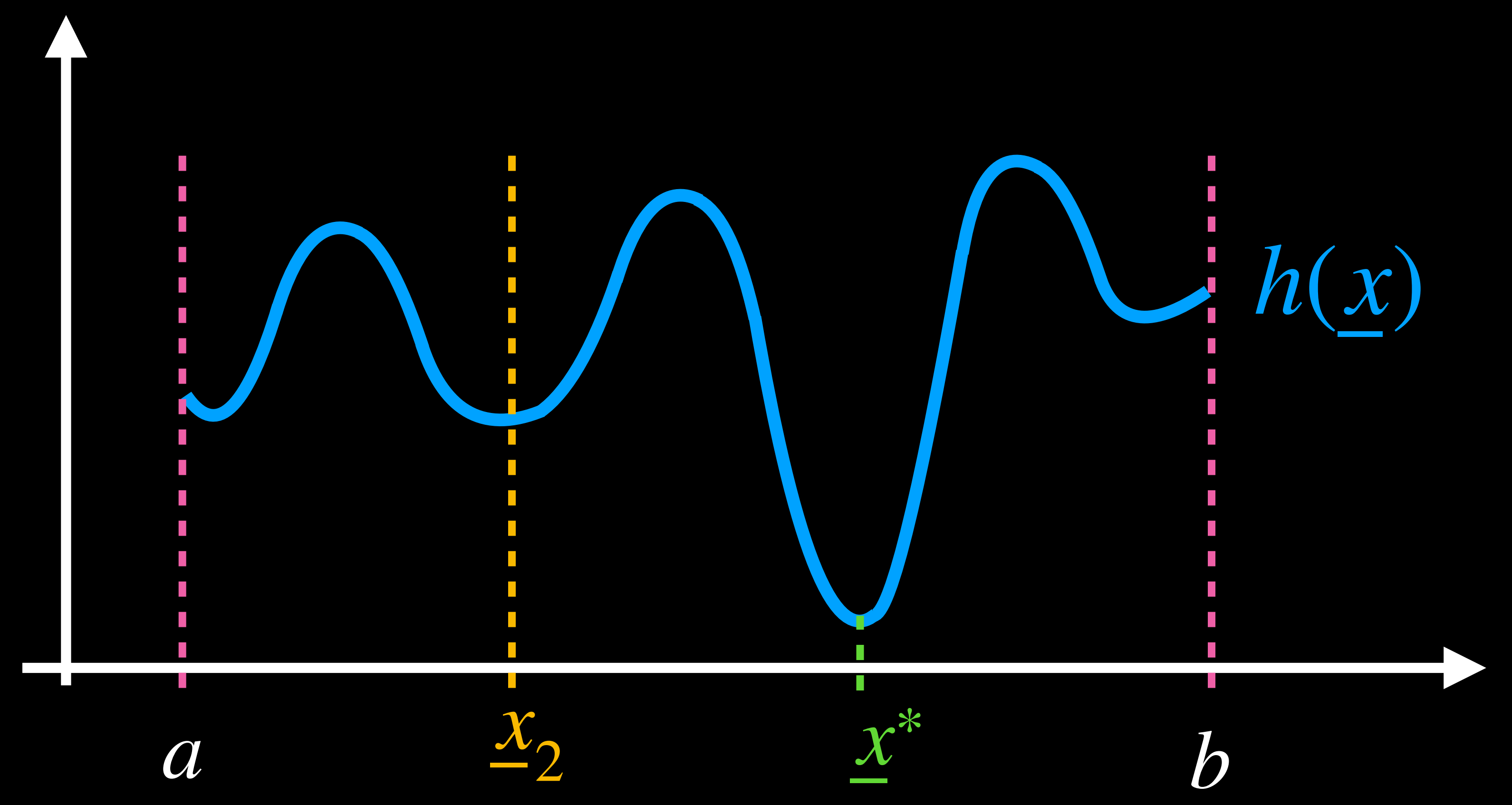
stochastic descent:

$$\underline{X}_{t+1} = \underline{X}_t - \alpha_t \left(\nabla h(\underline{X}_t) \cdot \underline{U}_t \right) \underline{U}_t$$

$\underline{U}_t \sim \mathcal{U}(\mathbb{S})$

$\mathbb{S}$ be the d-dimensional unit sphere

algorithm 2: stochastic descent



$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

$t \in \{1,\ldots,n\}$ steps

$\alpha_t > 0$ step size

$-\nabla h(\underline{x}_t)$ vector of steepest decrease

$\underline{x}_0$ starting point

gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix}\right) \cdot \frac{1}{2\Delta x}$$

$\rightarrow$ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

dot product:

stochastic descent:

$$\underline{X}_{t+1} = \underline{X}_t - \alpha_t \left(\nabla h(\underline{X}_t) \cdot \underline{U}_t \right) \underline{U}_t$$

$\underline{U}_t \sim \mathcal{U}(\mathbb{S})$

$\mathbb{S}$ be the d-dimensional unit sphere

algorithm 2: stochastic descent

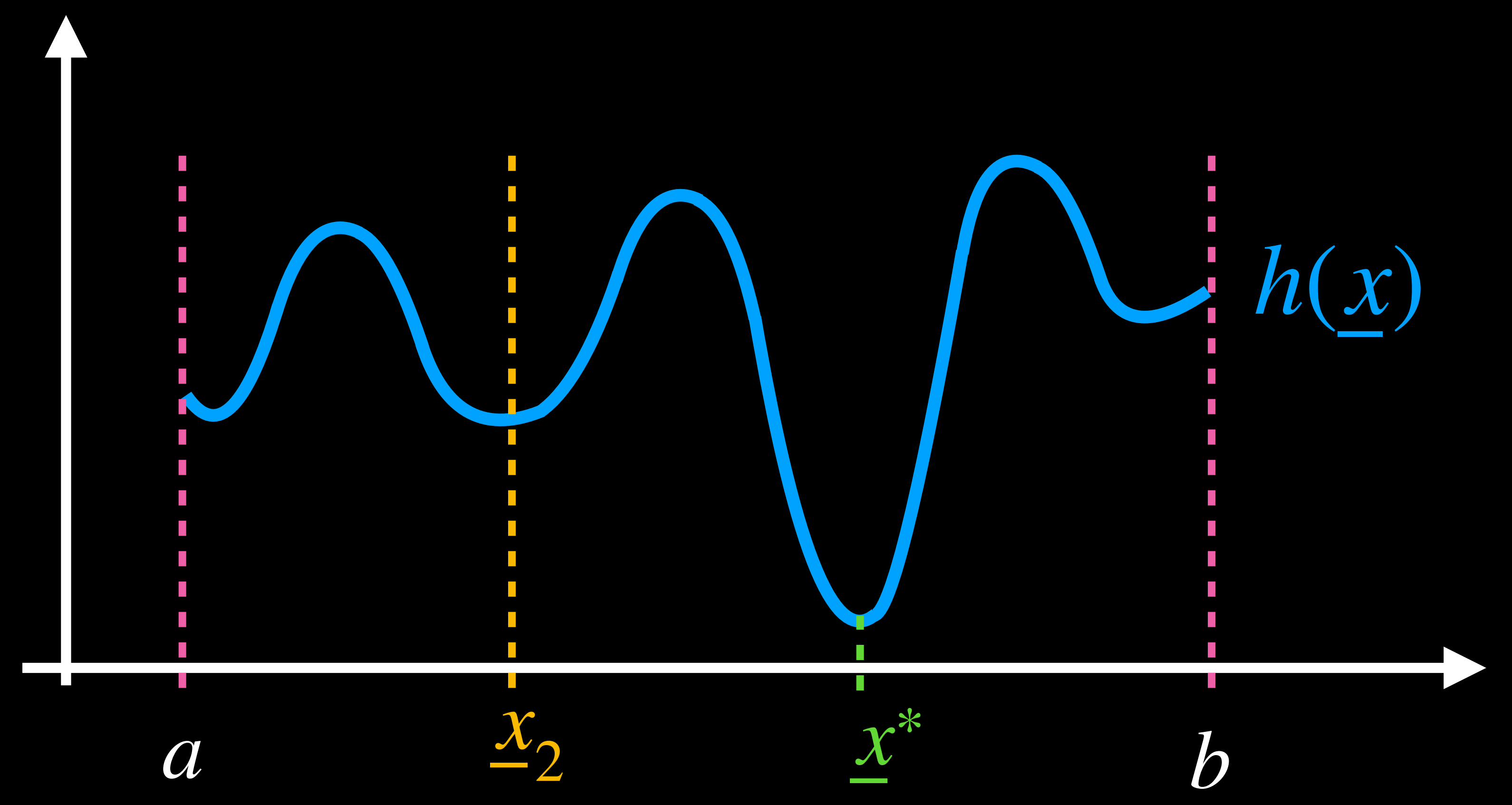
$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

$t \in \{1,\ldots,n\}$ steps

$\alpha_t > 0$ step size

$-\nabla h(\underline{x}_t)$ vector of steepest decrease

$\underline{x}_0$ starting point



gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} \right) \cdot \frac{1}{2\Delta x}$$

→ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

dot product: $\underline{x} \cdot \underline{y} = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_d \end{pmatrix}$

stochastic descent:

$$\underline{X}_{t+1} = \underline{X}_t - \alpha_t \left(\nabla h(\underline{X}_t) \cdot \underline{U}_t \right) \underline{U}_t$$

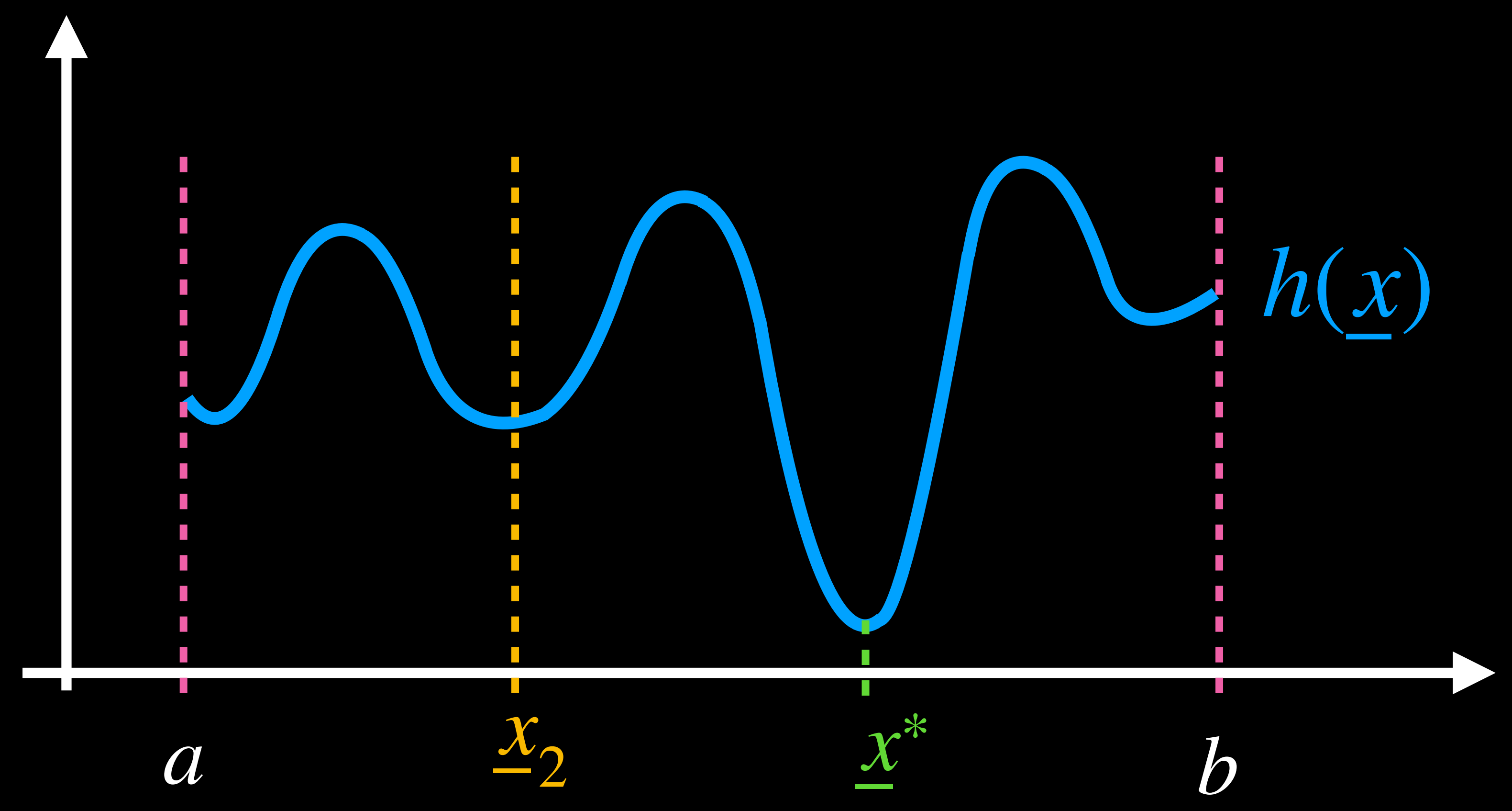
$\underline{U}_t \sim \mathcal{U}(\mathbb{S})$

$\mathbb{S}$ be the d-dimensional unit sphere

algorithm 2: stochastic descent

$\underline{x}^* := \underset{\underline{x} \in \mathbb{S}}{\arg\min} \left(h(\underline{x}) \right)$

$t \in \{1, \ldots, n\}$ steps
$\alpha_t > 0$ step size
$-\nabla h(\underline{x}_t)$ vector of steepest decrease
$\underline{x}_0$ starting point



gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h \begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h \begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} \right) \cdot \frac{1}{2\Delta x}$$

$\rightarrow$ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

dot product: $\underline{x} \cdot \underline{y} = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_d \end{pmatrix} = x_1 y_1 + \ldots + x_d y_d$

stochastic descent:

$$\underline{X}_{t+1} = \underline{X}_t - \alpha_t \left(\nabla h(\underline{X}_t) \cdot \underline{U}_t \right) \underline{U}_t$$

$\underline{U}_t \sim \mathcal{U}(\mathbb{S})$

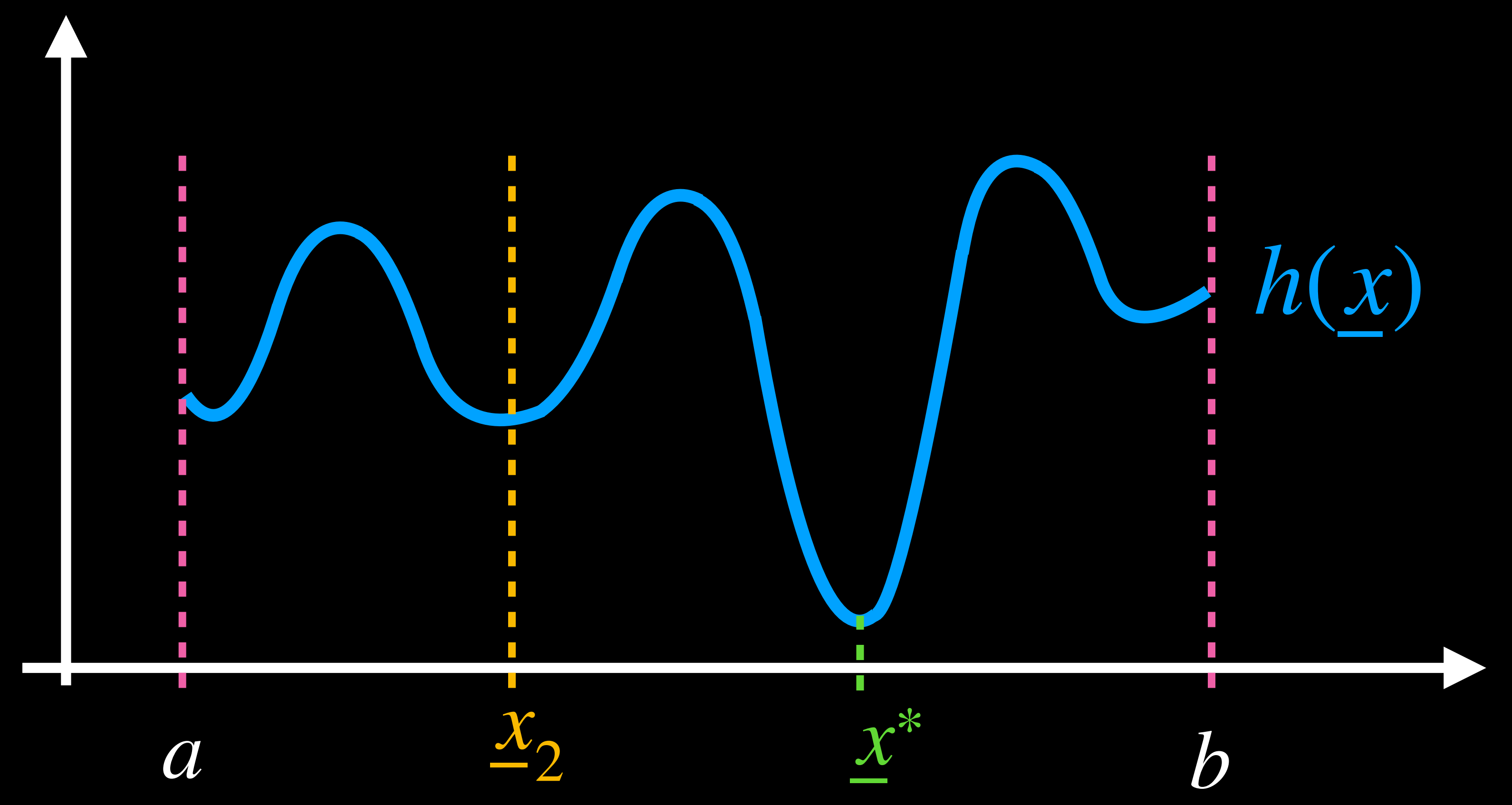
$\mathbb{S}$ be the d-dimensional unit sphere

algorithm 2: stochastic descent

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

$t \in \{1,\ldots,n\}$ steps
$\alpha_t > 0$ step size
$-\nabla h(\underline{x}_t)$ vector of steepest decrease
$\underline{x}_0$ starting point



gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} \right) \cdot \frac{1}{2\Delta x}$$

→ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

dot product: $\underline{x} \cdot \underline{y} = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_d \end{pmatrix} = x_1 y_1 + \ldots + x_d y_d$

directional derivative:

stochastic descent:

$$\underline{X}_{t+1} = \underline{X}_t - \alpha_t \left(\nabla h(\underline{X}_t) \cdot \underline{U}_t \right) \underline{U}_t$$

$\underline{U}_t \sim \mathcal{U}(\mathbb{S})$

$\mathbb{S}$ be the d-dimensional unit sphere

algorithm 2: stochastic descent

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\,\big(h(\underline{x})\big)$$

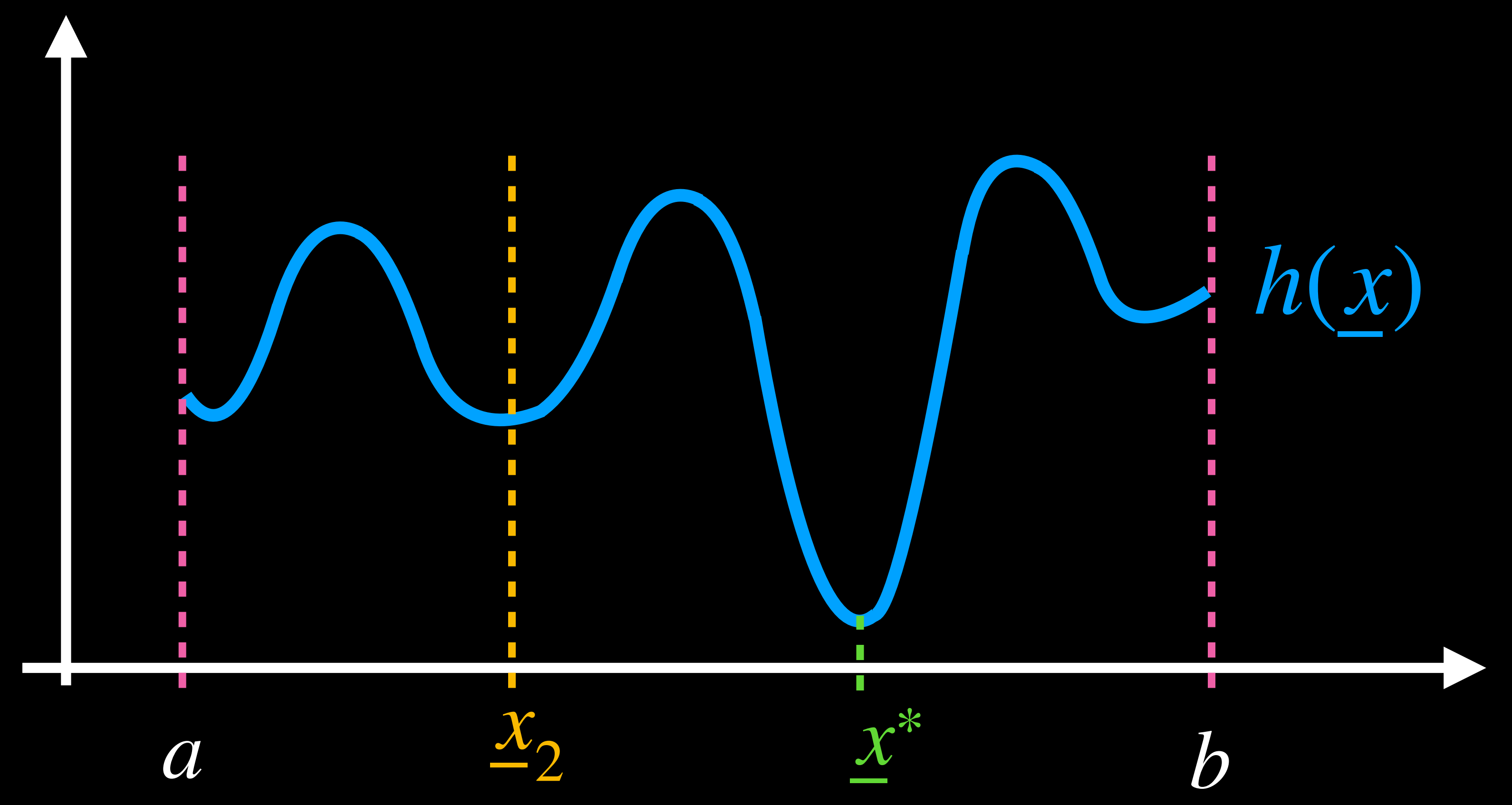
gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

$t \in \{1,\ldots,n\}$ steps

$\alpha_t > 0$ step size

$-\nabla h(\underline{x}_t)$ vector of steepest decrease

$\underline{x}_0$ starting point



gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1+\Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1-\Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix}\right) \cdot \frac{1}{2\Delta x}$$

→ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

dot product: $\underline{x}\cdot\underline{y} = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_d \end{pmatrix} = x_1 y_1 + \ldots + x_d y_d$

directional derivative:

$$\nabla h(\underline{x}) \cdot \frac{\underline{y}}{|\underline{y}|} \approx$$

stochastic descent:

$$\underline{X}_{t+1} = \underline{X}_t - \alpha_t \left(\nabla h(\underline{X}_t) \cdot \underline{U}_t \right) \underline{U}_t$$

$\underline{U}_t \sim \mathscr{U}(\mathbb{S})$

$\mathbb{S}$ be the d-dimensional unit sphere

algorithm 2: stochastic descent

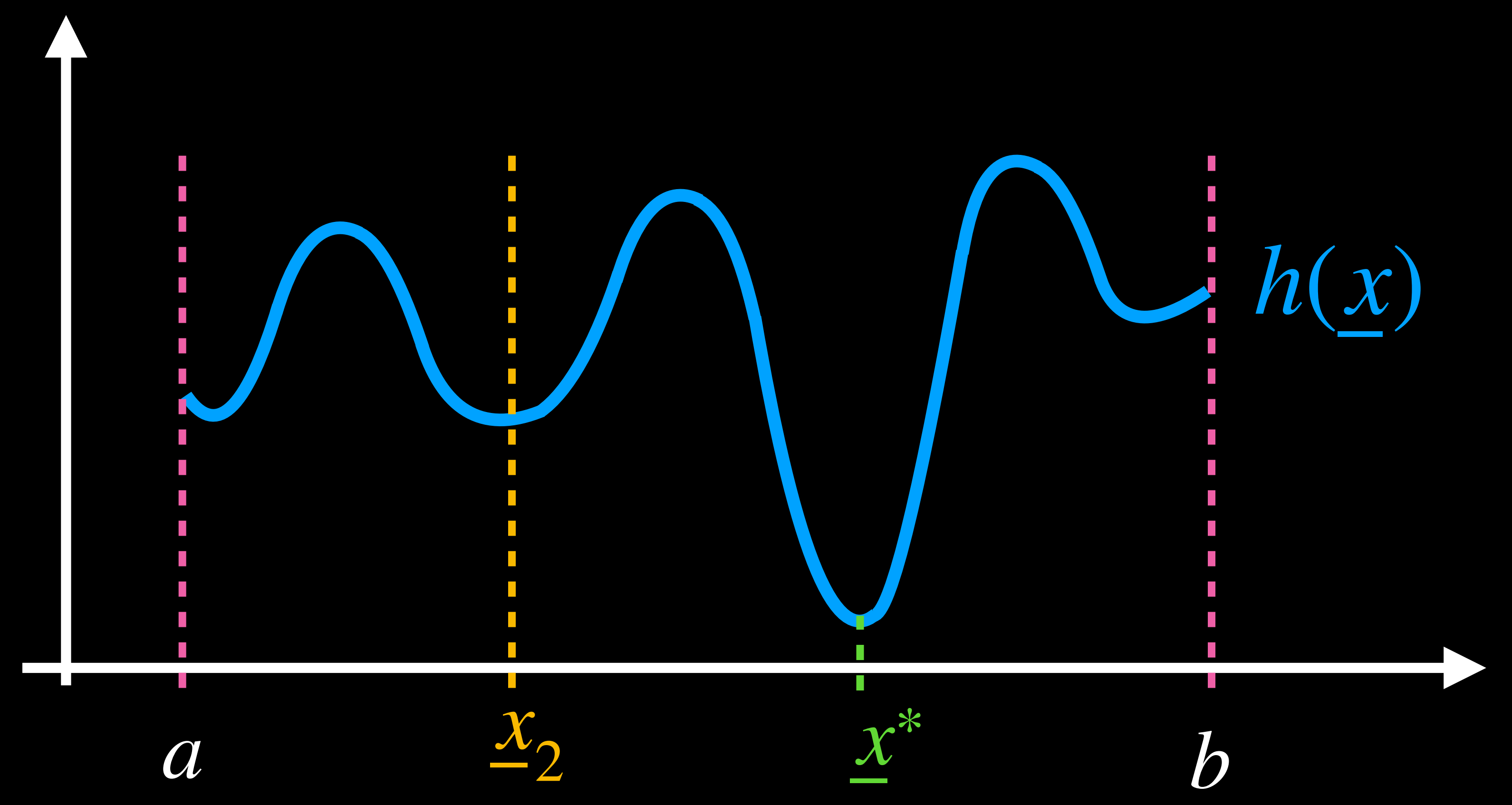
$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

$t \in \{1,\ldots,n\}$ steps

$\alpha_t > 0$ step size

$-\nabla h(\underline{x}_t)$ vector of steepest decrease

$\underline{x}_0$ starting point



gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} \right) \cdot \frac{1}{2\Delta x}$$

$\rightarrow$ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

dot product: $\underline{x} \cdot \underline{y} = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_d \end{pmatrix} = x_1 y_1 + \ldots + x_d y_d$

directional derivative:

$$\nabla h(\underline{x}) \cdot \frac{\underline{y}}{|\underline{y}|} \approx \frac{\Delta h(\underline{x}, \underline{y})}{2|\underline{y}|},$$

stochastic descent:

$$\underline{X}_{t+1} = \underline{X}_t - \alpha_t \left(\nabla h(\underline{X}_t) \cdot \underline{U}_t \right) \underline{U}_t$$

$\underline{U}_t \sim \mathcal{U}(\mathbb{S})$

$\mathbb{S}$ be the d-dimensional unit sphere

algorithm 2: stochastic descent

$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$

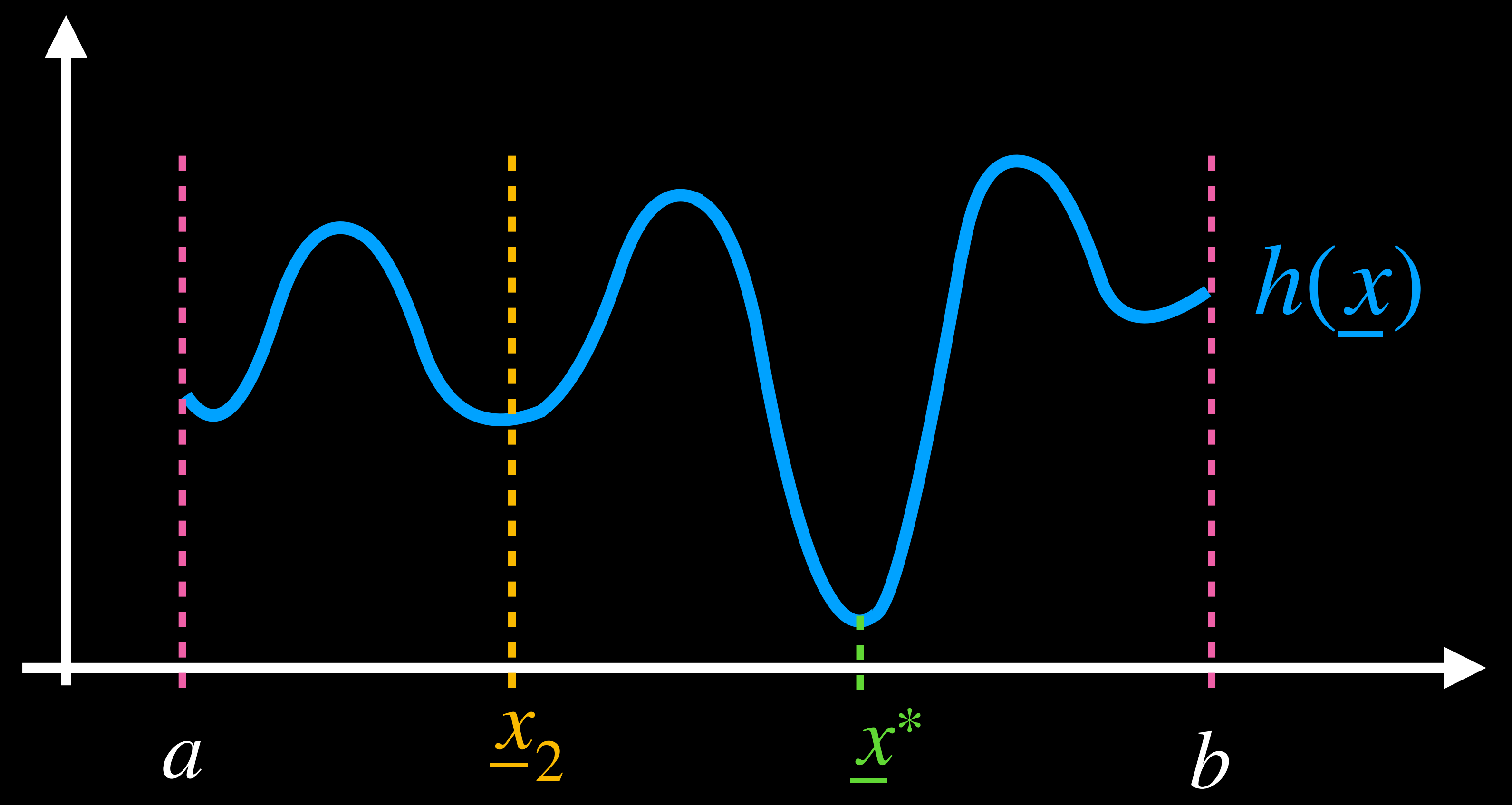
gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

$t \in \{1,\ldots,n\}$ steps

$\alpha_t > 0$ step size

$-\nabla h(\underline{x}_t)$ vector of steepest decrease

$\underline{x}_0$ starting point



gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix}\right) \cdot \frac{1}{2\Delta x}$$

$\rightarrow$ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

dot product: $\underline{x} \cdot \underline{y} = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_d \end{pmatrix} = x_1 y_1 + \ldots + x_d y_d$

stochastic descent:

$$\underline{X}_{t+1} = \underline{X}_t - \alpha_t \left(\nabla h(\underline{X}_t) \cdot \underline{U}_t \right) \underline{U}_t$$

$\underline{U}_t \sim \mathcal{U}(\mathbb{S})$

$\mathbb{S}$ be the d-dimensional unit sphere

directional derivative:

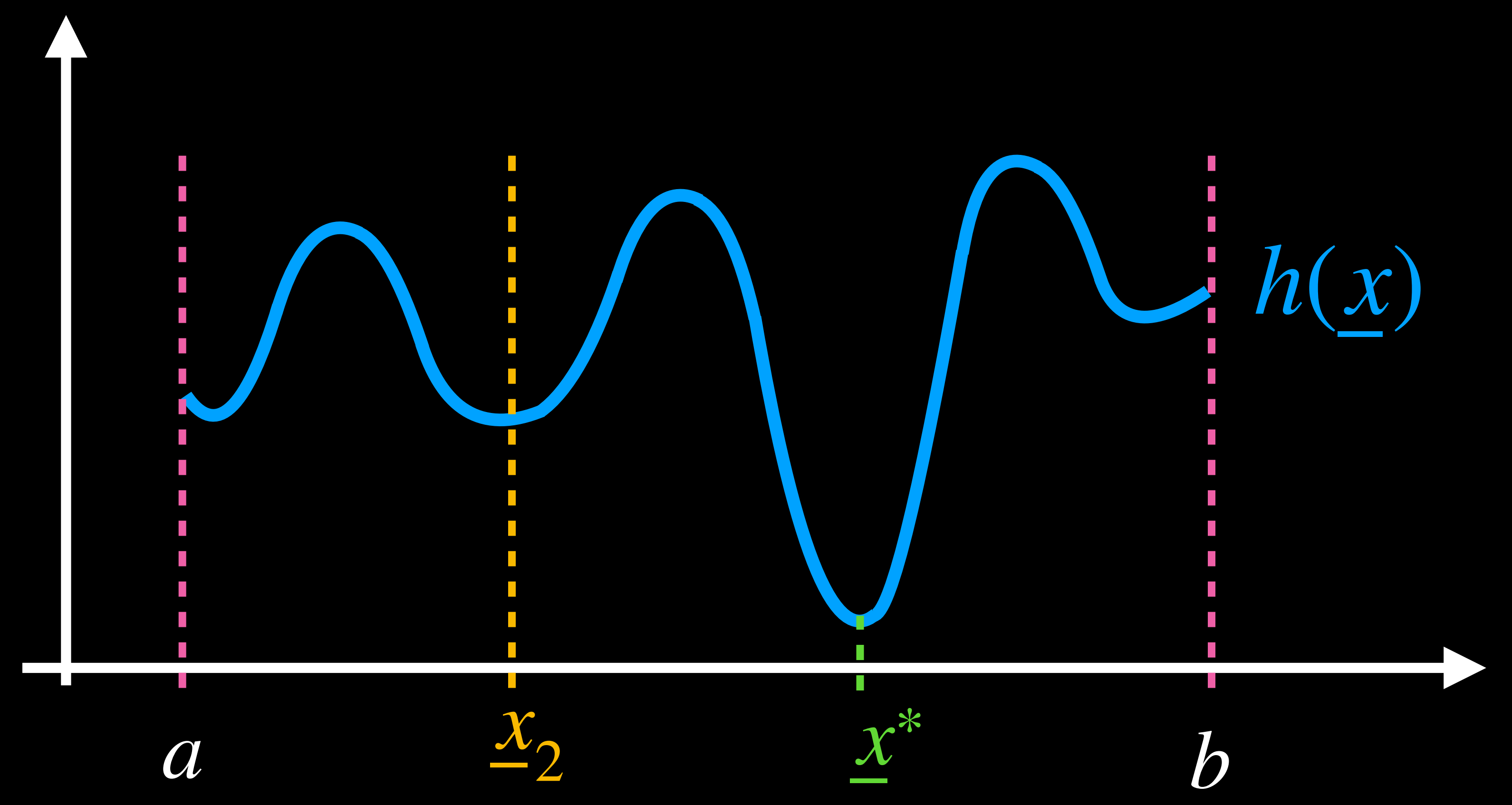
$$\nabla h(\underline{x}) \cdot \frac{\underline{y}}{|\underline{y}|} \approx \frac{\Delta h(\underline{x}, \underline{y})}{2|\underline{y}|}, \qquad \Delta h(\underline{x}, \underline{y}) := h(\underline{x} + \underline{y}) - h(\underline{x} - \underline{y})$$

algorithm 2: stochastic descent

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \big(h(\underline{x})\big)$$

gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

$t \in \{1,\ldots,n\}$ steps
$\alpha_t > 0$ step size
$-\nabla h(\underline{x}_t)$ vector of steepest decrease
$\underline{x}_0$ starting point



gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix}\right) \cdot \frac{1}{2\Delta x}$$

→ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

dot product: $\underline{x}\cdot\underline{y} = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_d \end{pmatrix} = x_1 y_1 + \ldots + x_d y_d$

directional derivative:

$$\nabla h(\underline{x}) \cdot \frac{\underline{y}}{|\underline{y}|} \approx \frac{\Delta h(\underline{x},\underline{y})}{2|\underline{y}|}, \qquad \Delta h(\underline{x},\underline{y}) := h(\underline{x}+\underline{y}) - h(\underline{x}-\underline{y})$$

→ approximation has 2 evaluations of h (cheaper)

stochastic descent:

$$\underline{X}_{t+1} = \underline{X}_t - \alpha_t \left(\nabla h(\underline{X}_t) \cdot \underline{U}_t \right) \underline{U}_t$$

$\underline{U}_t \sim \mathcal{U}(\mathbb{S})$

$\mathbb{S}$ be the d-dimensional unit sphere

algorithm 2: stochastic descent

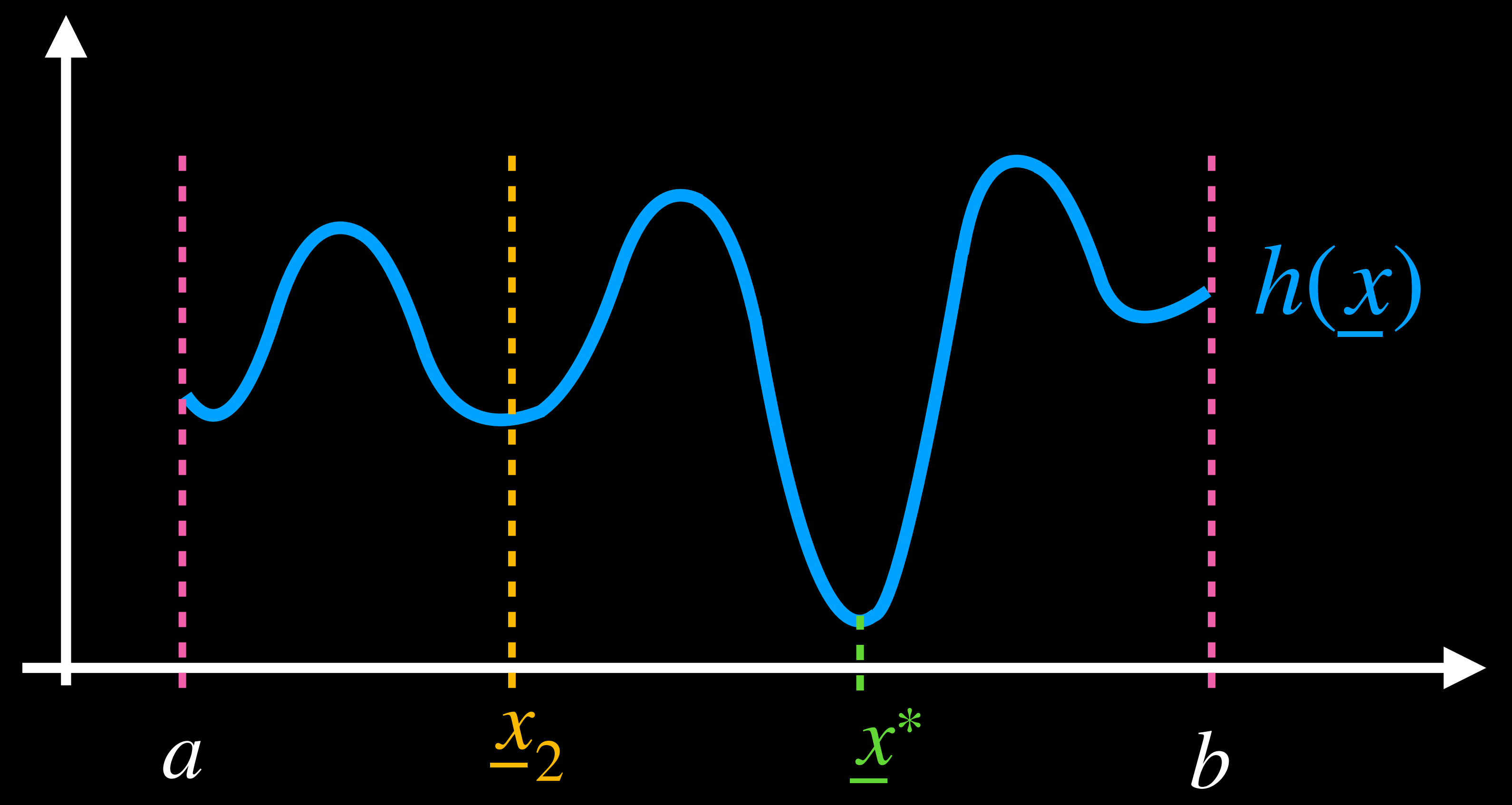
$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$

$t \in \{1,\ldots,n\}$ steps

$\alpha_t > 0$ step size

$-\nabla h(\underline{x}_t)$ vector of steepest decrease

$\underline{x}_0$ starting point



gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix}\right) \cdot \frac{1}{2\Delta x}$$

→ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

dot product: $\underline{x} \cdot \underline{y} = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_d \end{pmatrix} = x_1 y_1 + \ldots + x_d y_d$

directional derivative:

$$\nabla h(\underline{x}) \cdot \frac{\underline{y}}{|\underline{y}|} \approx \frac{\Delta h(\underline{x}, \underline{y})}{2|\underline{y}|}, \qquad \Delta h(\underline{x}, \underline{y}) := h(\underline{x} + \underline{y}) - h(\underline{x} - \underline{y})$$

→ approximation has 2 evaluations of h (cheaper)

stochastic descent:

$$\underline{X}_{t+1} = \underline{X}_t - \frac{\alpha_t}{2\beta_t} \Delta h(\underline{X}_t, \beta_t \underline{U}_t)\, \underline{U}_t$$

$\underline{U}_t \sim \mathscr{U}(\mathbb{S})$

$\mathbb{S}$ be the d-dimensional unit sphere

algorithm 2: stochastic descent

$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\,\big(h(\underline{x})\big)$

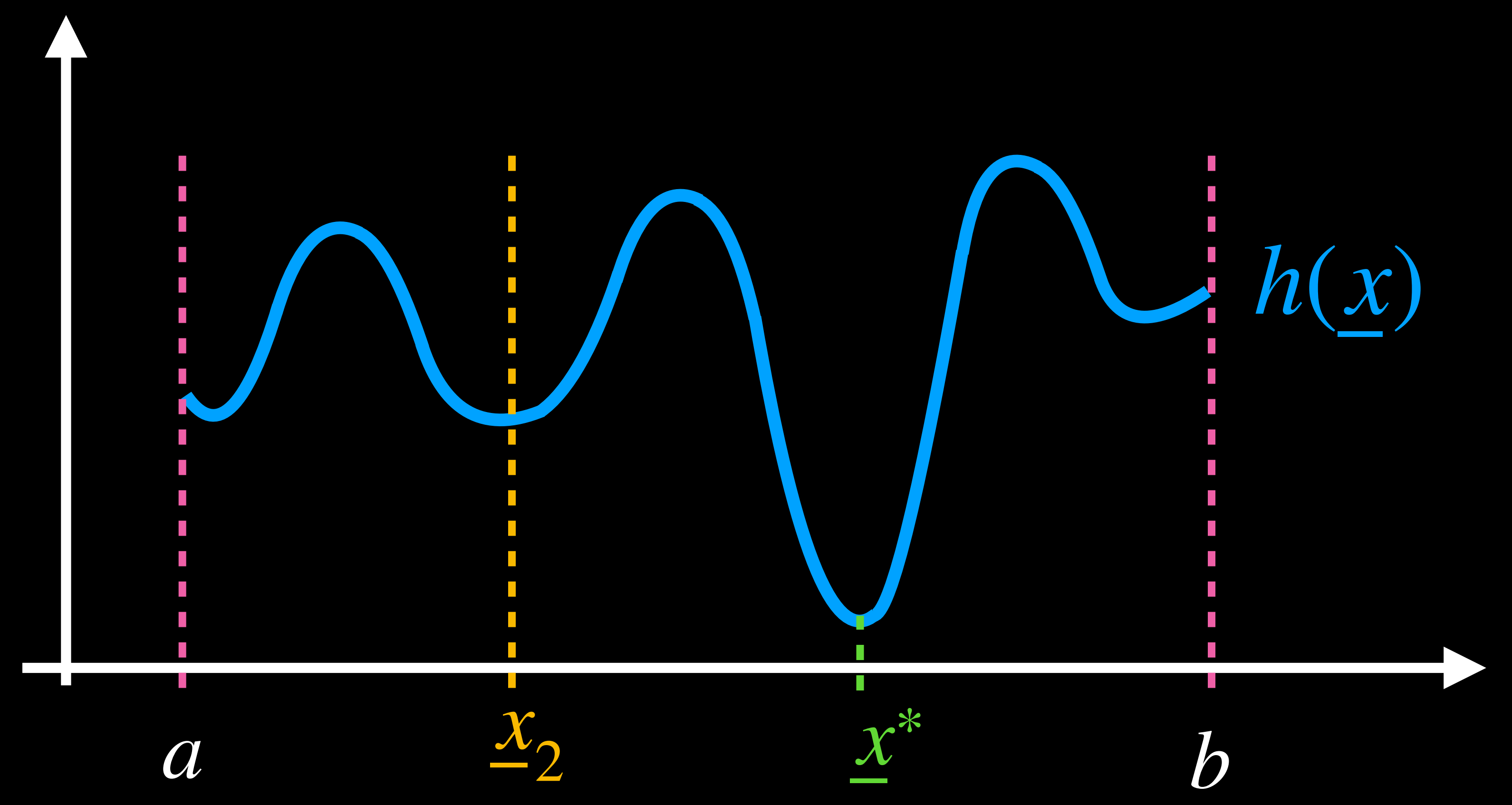
gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

$t \in \{1,\ldots,n\}$ steps

$\alpha_t > 0$ step size

$-\nabla h(\underline{x}_t)$ vector of steepest decrease

$\underline{x}_0$ starting point



gradient: $\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}$, $\quad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} \right) \cdot \frac{1}{2\Delta x}$

→ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

dot product: $\underline{x} \cdot \underline{y} = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_d \end{pmatrix} = x_1 y_1 + \ldots + x_d y_d$

directional derivative:

$\nabla h(\underline{x}) \cdot \frac{\underline{y}}{|\underline{y}|} \approx \frac{\Delta h(\underline{x}, \underline{y})}{2|\underline{y}|}, \qquad \Delta h(\underline{x}, \underline{y}) := h(\underline{x} + \underline{y}) - h(\underline{x} - \underline{y})$

→ approximation has 2 evaluations of h (cheaper)

stochastic descent:

$\underline{X}_{t+1} = \underline{X}_t - \frac{\alpha_t}{2\beta_t} \Delta h(\underline{X}_t, \beta_t \underline{U}_t)\, \underline{U}_t$

$\underline{U}_t \sim \mathcal{U}(\mathbb{S})$

$\mathbb{S}$ be the d-dimensional unit sphere

β_t sampling radius

algorithm 2: stochastic descent

$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\,\big(h(\underline{x})\big)$

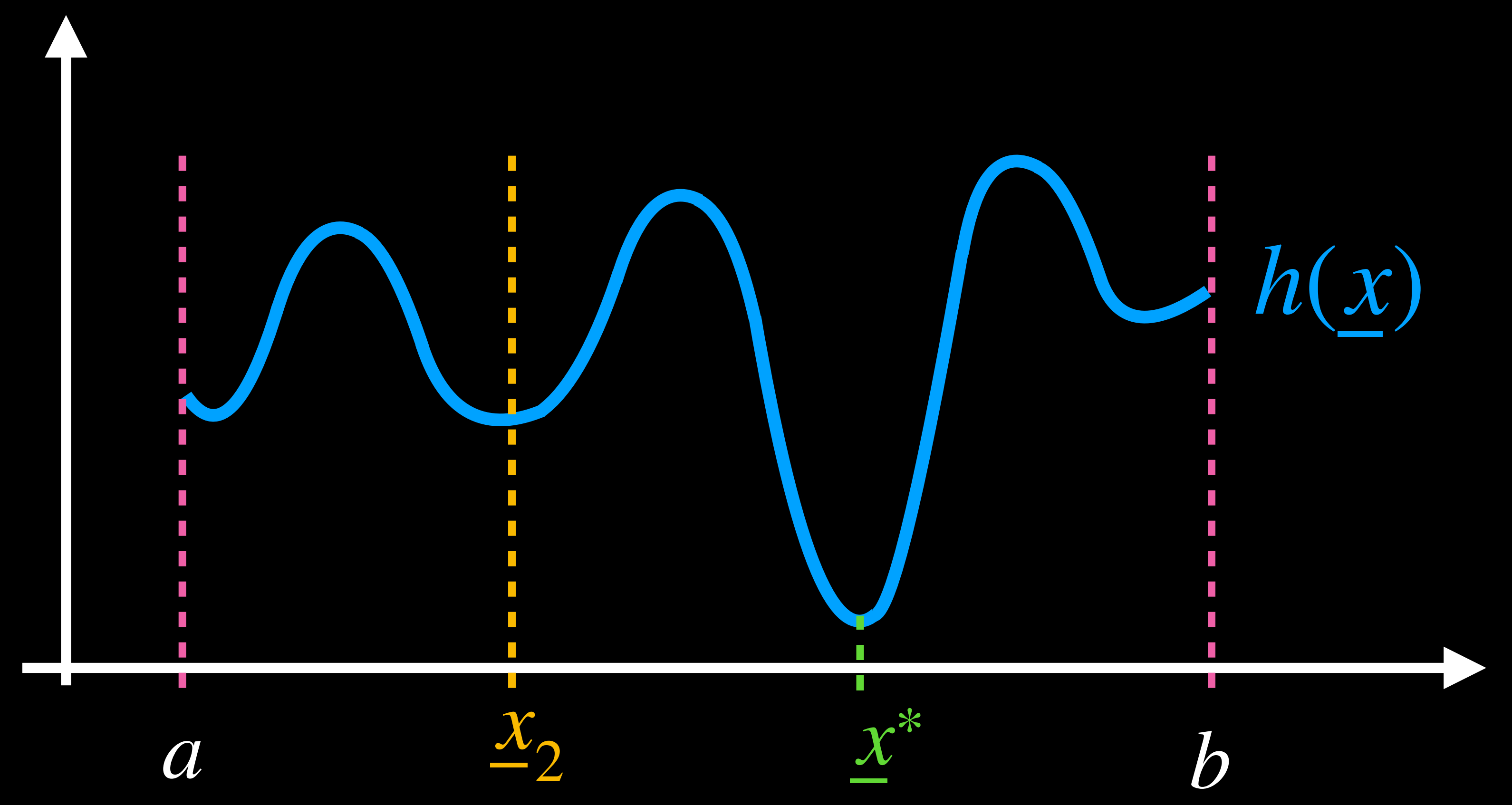
$t \in \{1,\ldots,n\}$ steps

$\alpha_t > 0$ step size

$-\nabla h(\underline{x}_t)$ vector of steepest decrease

$\underline{x}_0$ starting point

gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$



gradient: $\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}$, $\quad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1+\Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1-\Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix}\right)\cdot\frac{1}{2\Delta x}$

→ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

dot product: $\underline{x}\cdot\underline{y} = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_d \end{pmatrix} = x_1y_1 + \ldots + x_dy_d$

directional derivative:

$\nabla h(\underline{x}) \cdot \frac{\underline{y}}{|\underline{y}|} \approx \frac{\Delta h(\underline{x},\underline{y})}{2|\underline{y}|}, \quad \Delta h(\underline{x},\underline{y}) := h(\underline{x}+\underline{y}) - h(\underline{x}-\underline{y})$

→ approximation has 2 evaluations of h (cheaper)

stochastic descent:

$\underline{X}_{t+1} = \underline{X}_t - \frac{\alpha_t}{2\beta_t}\Delta h(\underline{X}_t, \beta_t\underline{U}_t)\,\underline{U}_t$

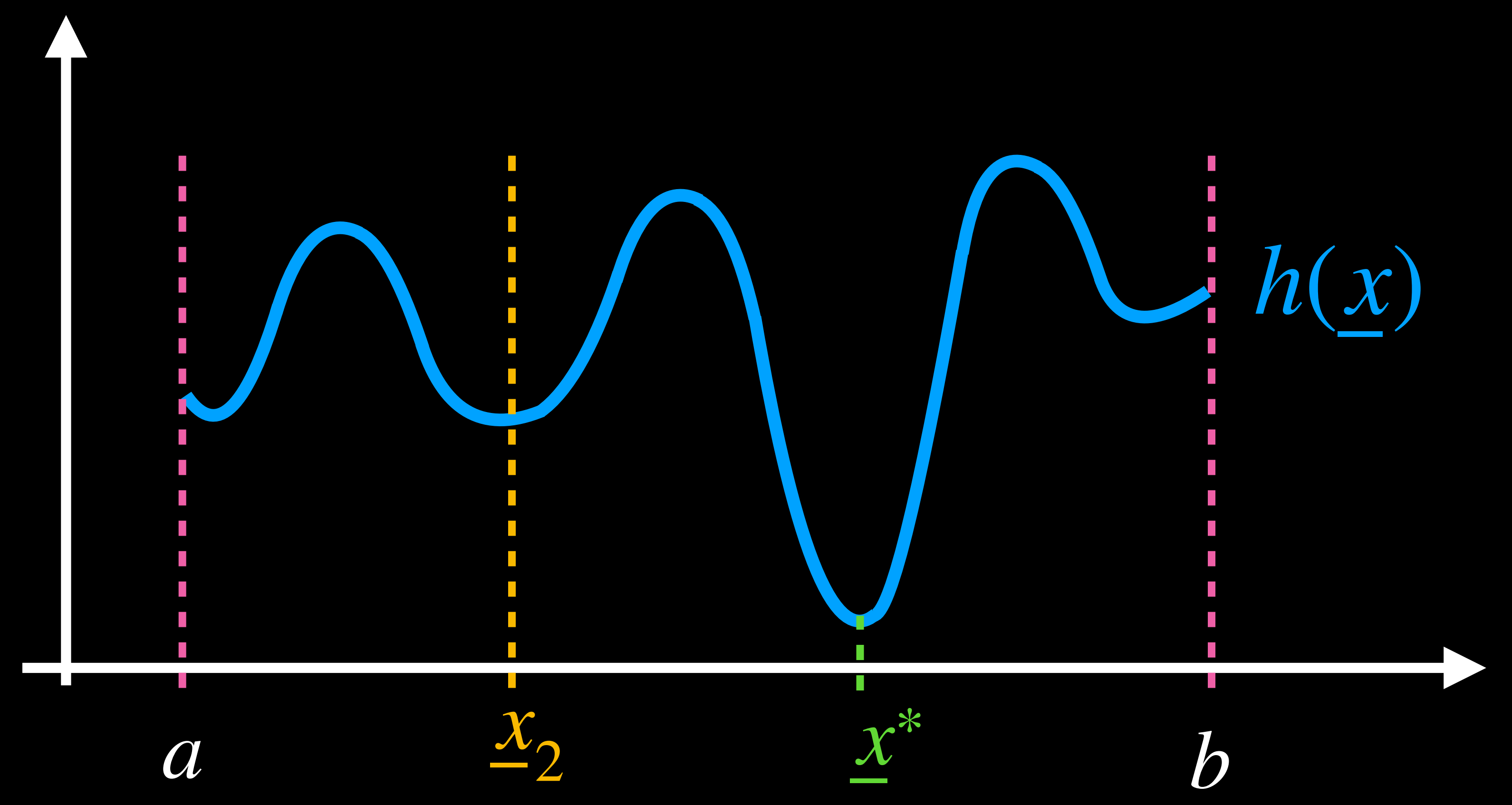
$\underline{U}_t \sim \mathscr{U}(\mathbb{S})$

$\mathbb{S}$ be the d-dimensional unit sphere

β_t sampling radius

properties:

algorithm 2: stochastic descent



$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$

gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

$t \in \{1,\ldots,n\}$ steps

$\alpha_t > 0$ step size

$-\nabla h(\underline{x}_t)$ vector of steepest decrease

$\underline{x}_0$ starting point

gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} \right) \cdot \frac{1}{2\Delta x}$$

→ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

dot product: $\underline{x} \cdot \underline{y} = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_d \end{pmatrix} = x_1 y_1 + \ldots + x_d y_d$

directional derivative:

$$\nabla h(\underline{x}) \cdot \frac{\underline{y}}{|\underline{y}|} \approx \frac{\Delta h(\underline{x}, \underline{y})}{2|\underline{y}|}, \qquad \Delta h(\underline{x}, \underline{y}) := h(\underline{x} + \underline{y}) - h(\underline{x} - \underline{y})$$

→ approximation has 2 evaluations of h (cheaper)

stochastic descent:

$$\underline{X}_{t+1} = \underline{X}_t - \frac{\alpha_t}{2\beta_t} \Delta h(\underline{X}_t, \beta_t \underline{U}_t) \underline{U}_t$$

$\underline{U}_t \sim \mathcal{U}(\mathbb{S})$

$\mathbb{S}$ be the d-dimensional unit sphere

β_t sampling radius

properties:

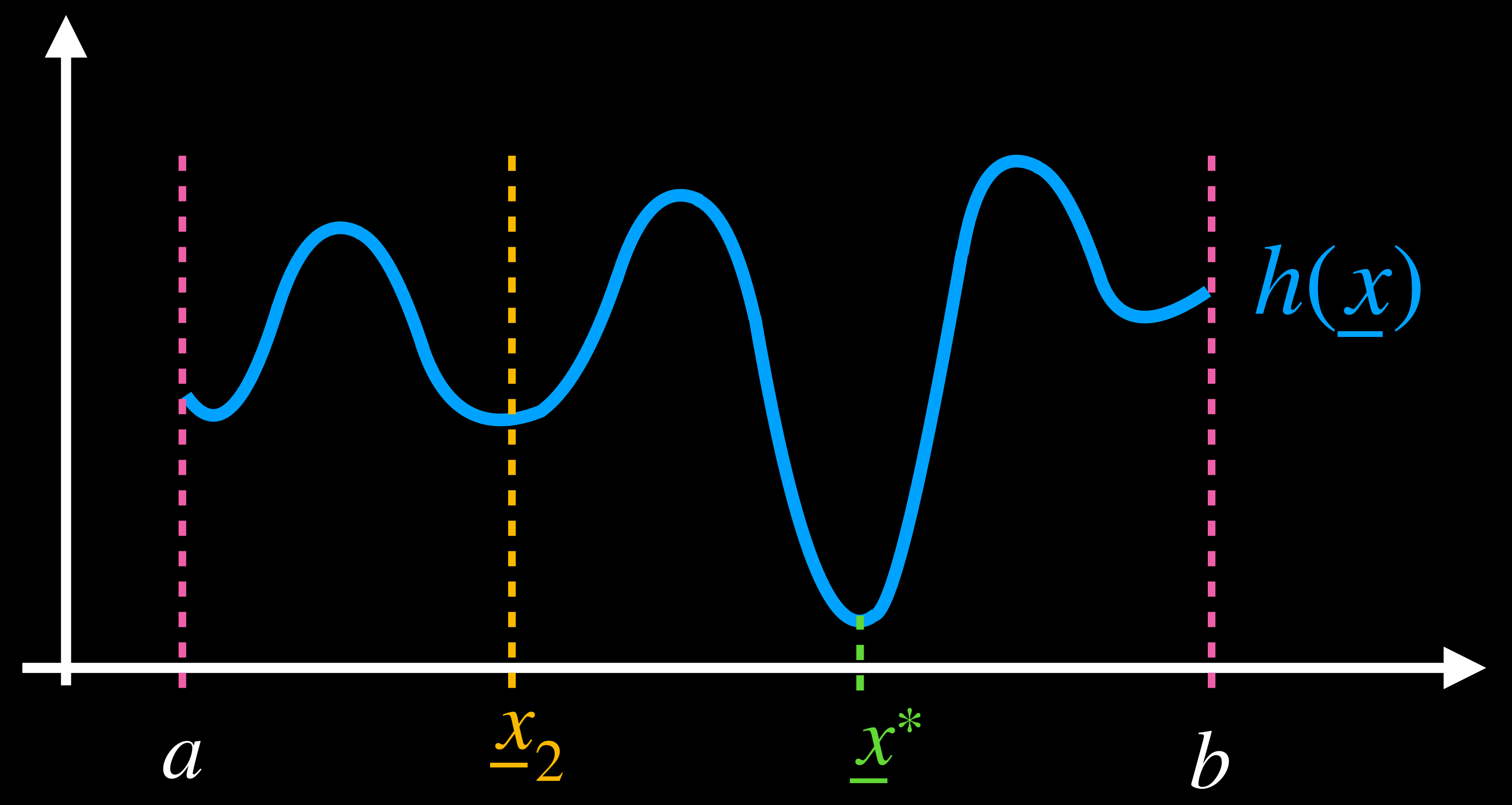
- simple

algorithm 2: stochastic descent

$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$

gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

$t \in \{1,\ldots,n\}$ steps
$\alpha_t > 0$ step size
$-\nabla h(\underline{x}_t)$ vector of steepest decrease
$\underline{x}_0$ starting point



gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix}\right) \cdot \frac{1}{2\Delta x}$$

→ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

dot product: $\underline{x} \cdot \underline{y} = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_d \end{pmatrix} = x_1 y_1 + \ldots + x_d y_d$

directional derivative:

$$\nabla h(\underline{x}) \cdot \frac{\underline{y}}{|\underline{y}|} \approx \frac{\Delta h(\underline{x}, \underline{y})}{2|\underline{y}|}, \qquad \Delta h(\underline{x}, \underline{y}) := h(\underline{x} + \underline{y}) - h(\underline{x} - \underline{y})$$

→ approximation has 2 evaluations of h (cheaper)

stochastic descent:

$$\underline{X}_{t+1} = \underline{X}_t - \frac{\alpha_t}{2\beta_t} \Delta h(\underline{X}_t, \beta_t \underline{U}_t)\, \underline{U}_t$$

$\underline{U}_t \sim \mathcal{U}(\mathbb{S})$
$\mathbb{S}$ be the d-dimensional unit sphere
β_t sampling radius

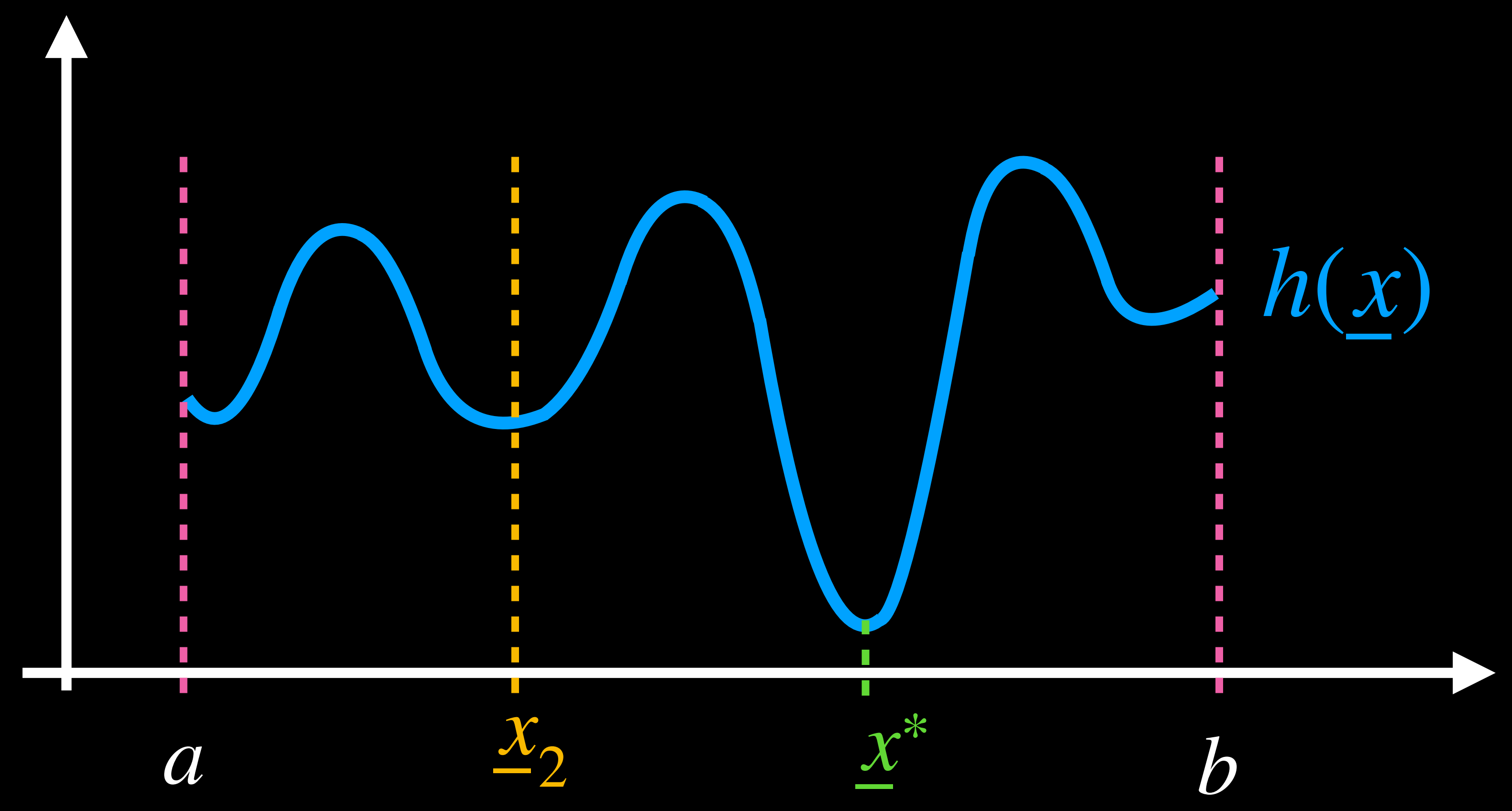
properties:
- simple
- no global convergence guarantee

algorithm 2: stochastic descent

$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$

gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

$t \in \{1,\ldots,n\}$ steps
$\alpha_t > 0$ step size
$-\nabla h(\underline{x}_t)$ vector of steepest decrease
$\underline{x}_0$ starting point



gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1+\Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1-\Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix}\right)\cdot \frac{1}{2\Delta x}$$

→ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

dot product: $\underline{x}\cdot\underline{y} = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_d \end{pmatrix} = x_1y_1 + \ldots + x_dy_d$

directional derivative:

$$\nabla h(\underline{x}) \cdot \frac{\underline{y}}{|\underline{y}|} \approx \frac{\Delta h(\underline{x},\underline{y})}{2|\underline{y}|}, \qquad \Delta h(\underline{x},\underline{y}) := h(\underline{x}+\underline{y}) - h(\underline{x}-\underline{y})$$

→ approximation has 2 evaluations of h (cheaper)

stochastic descent:

$$\underline{X}_{t+1} = \underline{X}_t - \frac{\alpha_t}{2\beta_t}\Delta h(\underline{X}_t, \beta_t\underline{U}_t)\,\underline{U}_t$$

$\underline{U}_t \sim \mathcal{U}(\mathbb{S})$
$\mathbb{S}$ be the d-dimensional unit sphere
β_t sampling radius

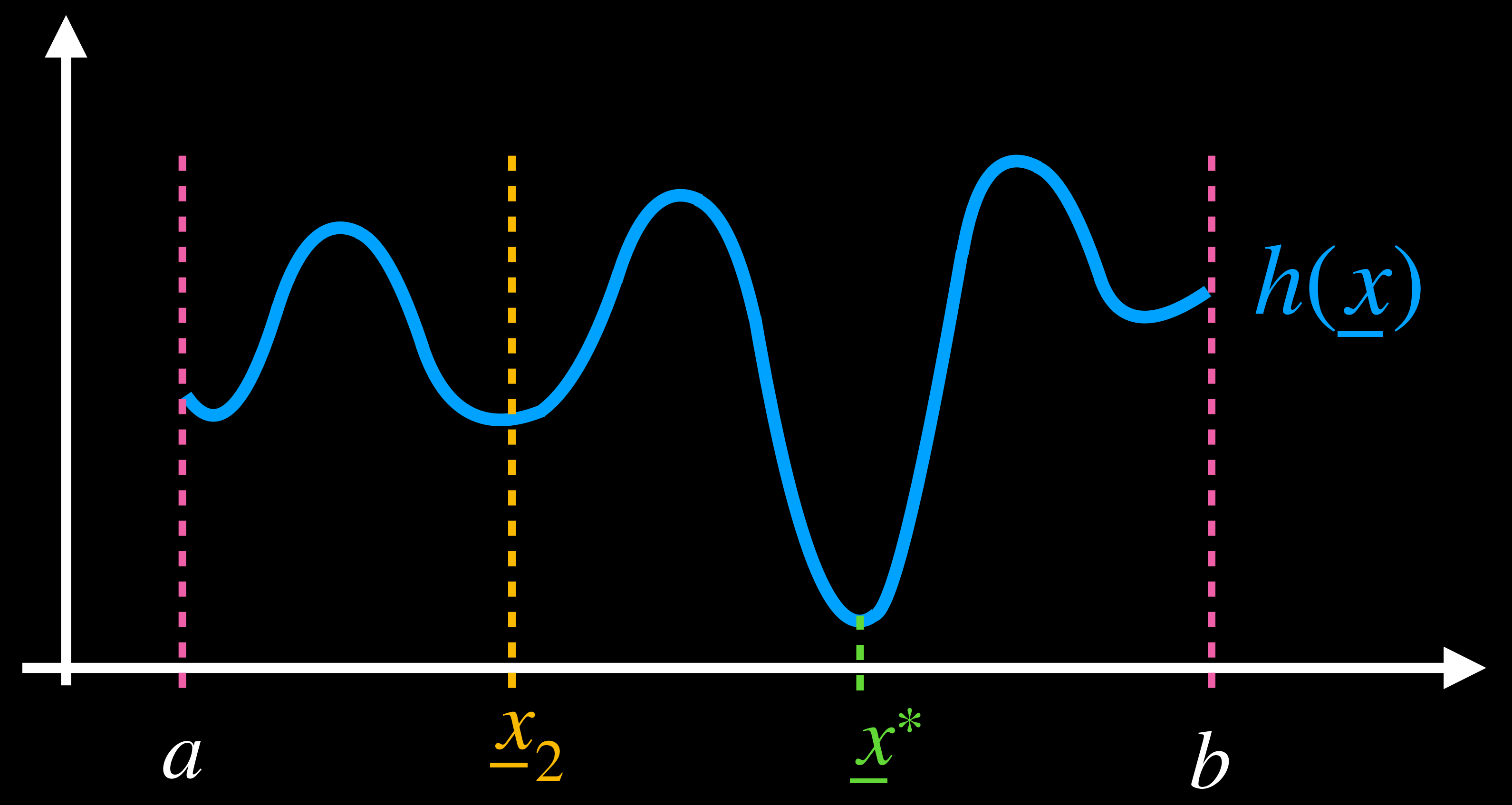
properties:
- simple
- no global convergence guarantee
- local convergence if $\lim_{n\to\infty} \alpha_n = 0$ and $\lim_{n\to\infty} \frac{\alpha_n}{\beta_n} = const.$

algorithm 2: stochastic descent

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

$t \in \{1,\ldots,n\}$ steps
$\alpha_t > 0$ step size
$-\nabla h(\underline{x}_t)$ vector of steepest decrease
$\underline{x}_0$ starting point



gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} \right) \cdot \frac{1}{2\Delta x}$$

→ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

dot product: $\underline{x} \cdot \underline{y} = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_d \end{pmatrix} = x_1 y_1 + \ldots + x_d y_d$

directional derivative:

$$\nabla h(\underline{x}) \cdot \frac{\underline{y}}{|\underline{y}|} \approx \frac{\Delta h(\underline{x}, \underline{y})}{2|\underline{y}|}, \qquad \Delta h(\underline{x}, \underline{y}) := h(\underline{x} + \underline{y}) - h(\underline{x} - \underline{y})$$

→ approximation has 2 evaluations of h (cheaper)

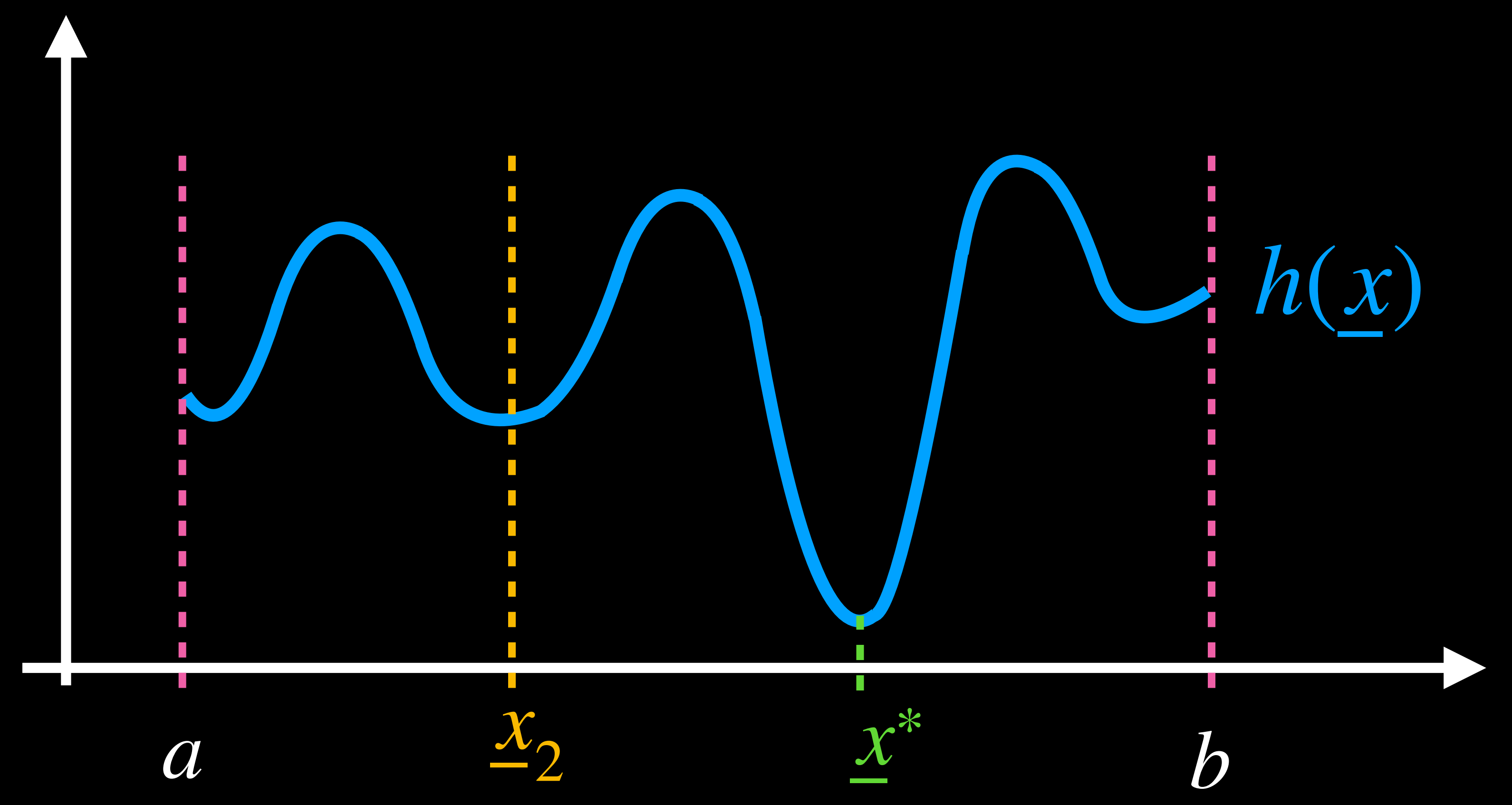
stochastic descent:

$$\underline{X}_{t+1} = \underline{X}_t - \frac{\alpha_t}{2\beta_t} \Delta h(\underline{X}_t, \beta_t \underline{U}_t)\, \underline{U}_t$$

$\underline{U}_t \sim \mathcal{U}(\mathbb{S})$
$\mathbb{S}$ be the d-dimensional unit sphere
β_t sampling radius

properties:
- simple
- no global convergence guarantee
- local convergence if $\lim_{n\to\infty} \alpha_n = 0$ and $\lim_{n\to\infty} \frac{\alpha_n}{\beta_n} = const.$
- converges fast ($\propto \frac{1}{n}$)

algorithm 2: stochastic descent



$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$

gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

$t \in \{1,\ldots,n\}$ steps
$\alpha_t > 0$ step size
$-\nabla h(\underline{x}_t)$ vector of steepest decrease
$\underline{x}_0$ starting point

gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix}\right) \cdot \frac{1}{2\Delta x}$$

→ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

dot product: $\underline{x} \cdot \underline{y} = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_d \end{pmatrix} = x_1 y_1 + \ldots + x_d y_d$

directional derivative:

$$\nabla h(\underline{x}) \cdot \frac{\underline{y}}{|\underline{y}|} \approx \frac{\Delta h(\underline{x}, \underline{y})}{2|\underline{y}|}, \qquad \Delta h(\underline{x}, \underline{y}) := h(\underline{x} + \underline{y}) - h(\underline{x} - \underline{y})$$

→ approximation has 2 evaluations of h (cheaper)

stochastic descent:

$$\underline{X}_{t+1} = \underline{X}_t - \frac{\alpha_t}{2\beta_t} \Delta h(\underline{X}_t, \beta_t \underline{U}_t)\, \underline{U}_t$$

$\underline{U}_t \sim \mathcal{U}(\mathbb{S})$

$\mathbb{S}$ be the d-dimensional unit sphere
β_t sampling radius

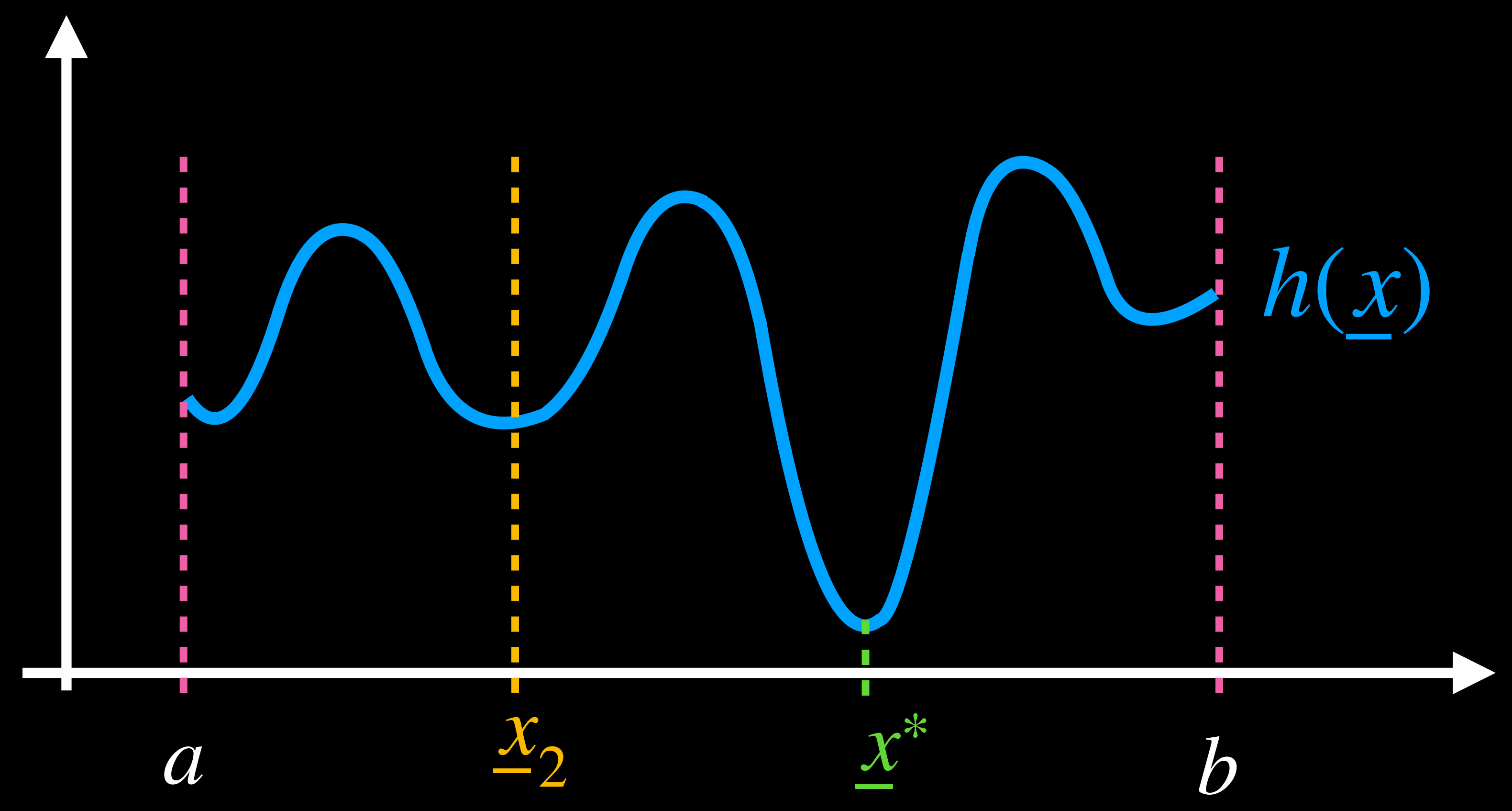
properties:
- simple
- no global convergence guarantee
- local convergence if $\lim_{n\to\infty} \alpha_n = 0$ and $\lim_{n\to\infty} \frac{\alpha_n}{\beta_n} = const.$
- converges fast ($\propto \frac{1}{n}$)
- needs uniform random numbers on the unit sphere

algorithm 2: stochastic descent

$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \big(h(\underline{x})\big)$

gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

$t \in \{1,\ldots,n\}$ steps
$\alpha_t > 0$ step size
$-\nabla h(\underline{x}_t)$ vector of steepest decrease
$\underline{x}_0$ starting point



gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} \right) \cdot \frac{1}{2\Delta x}$$

→ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

dot product: $\underline{x} \cdot \underline{y} = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_d \end{pmatrix} = x_1 y_1 + \ldots + x_d y_d$

directional derivative:

$$\nabla h(\underline{x}) \cdot \frac{\underline{y}}{|\underline{y}|} \approx \frac{\Delta h(\underline{x}, \underline{y})}{2|\underline{y}|}, \qquad \Delta h(\underline{x}, \underline{y}) := h(\underline{x} + \underline{y}) - h(\underline{x} - \underline{y})$$

→ approximation has 2 evaluations of h (cheaper)

stochastic descent:

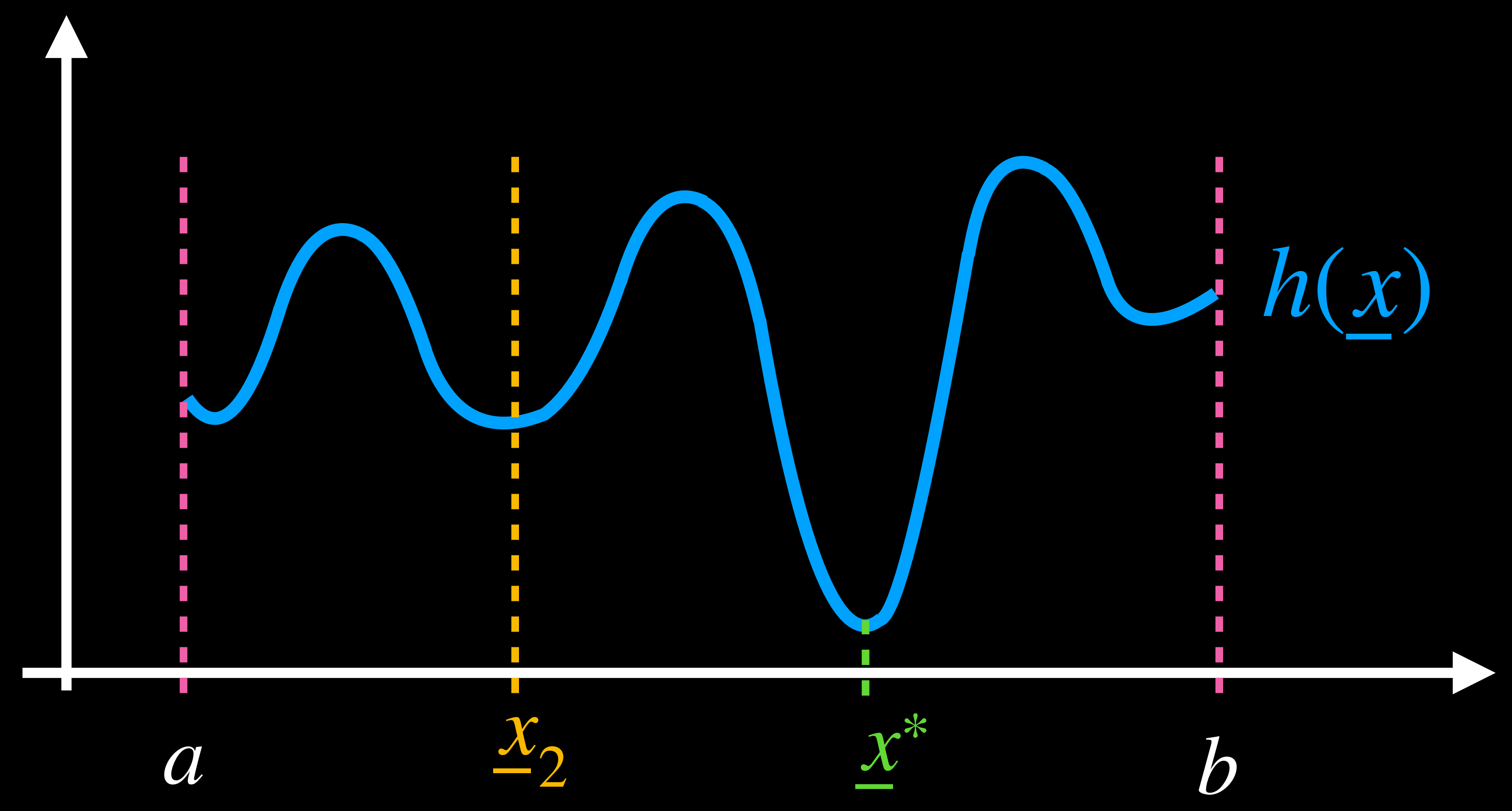
$$\underline{X}_{t+1} = \underline{X}_t - \frac{\alpha_t}{2\beta_t} \Delta h(\underline{X}_t, \beta_t \underline{U}_t)\, \underline{U}_t$$

$\underline{U}_t \sim \mathcal{U}(\mathbb{S})$
$\mathbb{S}$ be the d-dimensional unit sphere
β_t sampling radius

properties:

- simple
- no global convergence guarantee
- local convergence if $\lim_{n\to\infty} \alpha_n = 0$ and $\lim_{n\to\infty} \frac{\alpha_n}{\beta_n} = const.$
- converges fast ($\propto \frac{1}{n}$)
- needs uniform random numbers on the unit sphere
- need to choose $\alpha_t, \beta_t, \underline{X}_0$

algorithm 2: stochastic descent



$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$

$t \in \{1,\ldots,n\}$ steps
$\alpha_t > 0$ step size
$-\nabla h(\underline{x}_t)$ vector of steepest decrease
$\underline{x}_0$ starting point

gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix}\right) \cdot \frac{1}{2\Delta x}$$

→ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

dot product: $\underline{x} \cdot \underline{y} = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_d \end{pmatrix} = x_1 y_1 + \ldots + x_d y_d$

directional derivative:

$$\nabla h(\underline{x}) \cdot \frac{\underline{y}}{|\underline{y}|} \approx \frac{\Delta h(\underline{x}, \underline{y})}{2|\underline{y}|}, \qquad \Delta h(\underline{x}, \underline{y}) := h(\underline{x} + \underline{y}) - h(\underline{x} - \underline{y})$$

→ approximation has 2 evaluations of h (cheaper)

stochastic descent:

$$\underline{X}_{t+1} = \underline{X}_t - \frac{\alpha_t}{2\beta_t} \Delta h(\underline{X}_t, \beta_t \underline{U}_t)\, \underline{U}_t$$

$\underline{U}_t \sim \mathcal{U}(\mathbb{S})$
$\mathbb{S}$ be the d-dimensional unit sphere
β_t sampling radius

properties:

- simple
- no global convergence guarantee
- local convergence if $\lim_{n\to\infty} \alpha_n = 0$ and $\lim_{n\to\infty} \frac{\alpha_n}{\beta_n} = const.$
- converges fast ($\propto \frac{1}{n}$)
- needs uniform random numbers on the unit sphere
- need to choose $\alpha_t, \beta_t, \underline{X}_0$

useful if:

algorithm 2: stochastic descent

$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\,\big(h(\underline{x})\big)$

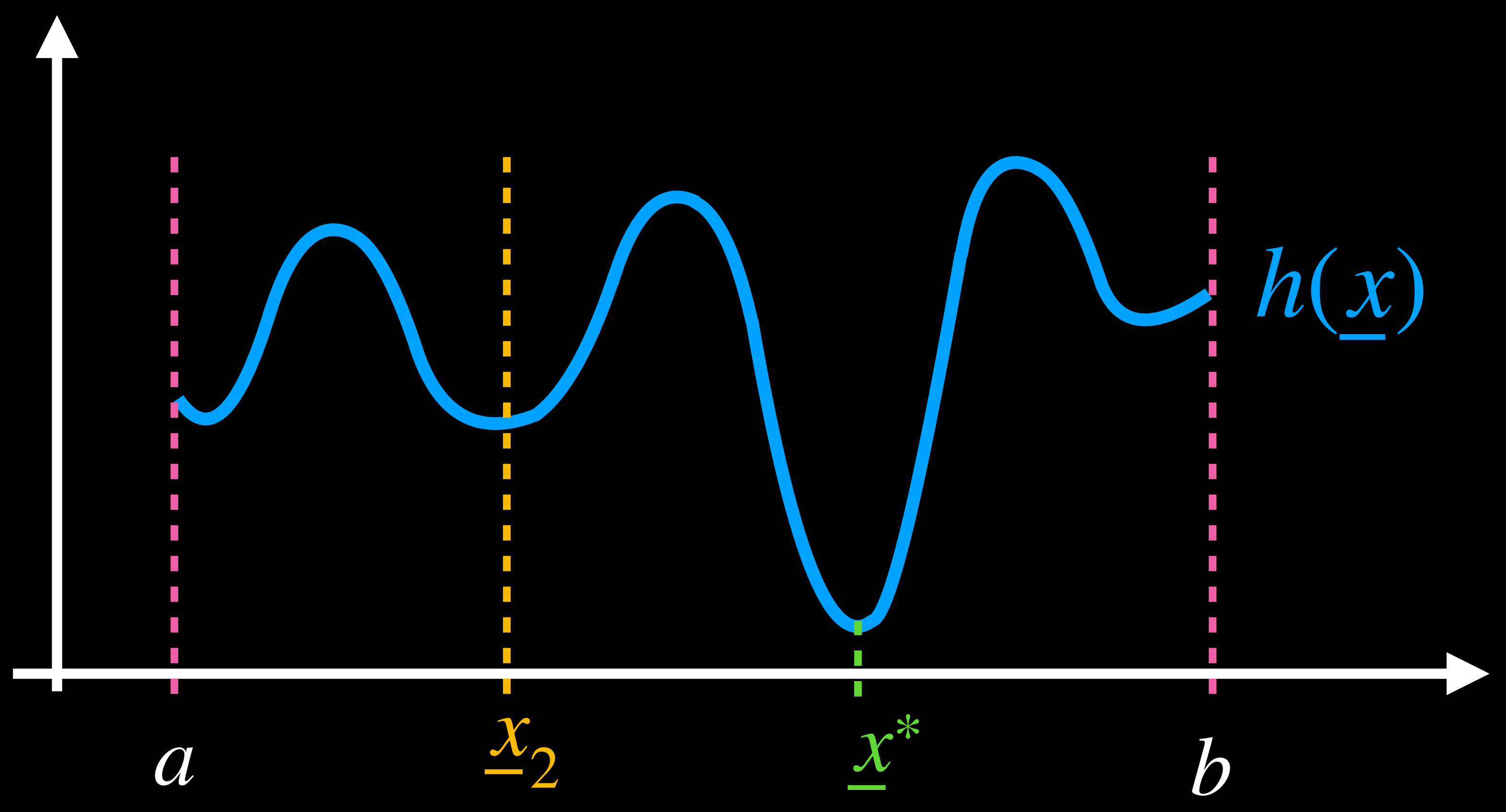
$t \in \{1,\ldots,n\}$ steps

$\alpha_t > 0$ step size

$-\nabla h(\underline{x}_t)$ vector of steepest decrease

$\underline{x}_0$ starting point

gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$



gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} \right) \cdot \frac{1}{2\Delta x}$$

→ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

dot product: $\underline{x} \cdot \underline{y} = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_d \end{pmatrix} = x_1 y_1 + \ldots + x_d y_d$

directional derivative:

$$\nabla h(\underline{x}) \cdot \frac{\underline{y}}{|\underline{y}|} \approx \frac{\Delta h(\underline{x},\underline{y})}{2|\underline{y}|}, \qquad \Delta h(\underline{x},\underline{y}) := h(\underline{x}+\underline{y}) - h(\underline{x}-\underline{y})$$

→ approximation has 2 evaluations of h (cheaper)

stochastic descent:

$$\underline{X}_{t+1} = \underline{X}_t - \frac{\alpha_t}{2\beta_t} \Delta h(\underline{X}_t, \beta_t \underline{U}_t)\, \underline{U}_t$$

$\underline{U}_t \sim \mathcal{U}(\mathbb{S})$

$\mathbb{S}$ be the d-dimensional unit sphere

β_t sampling radius

properties:

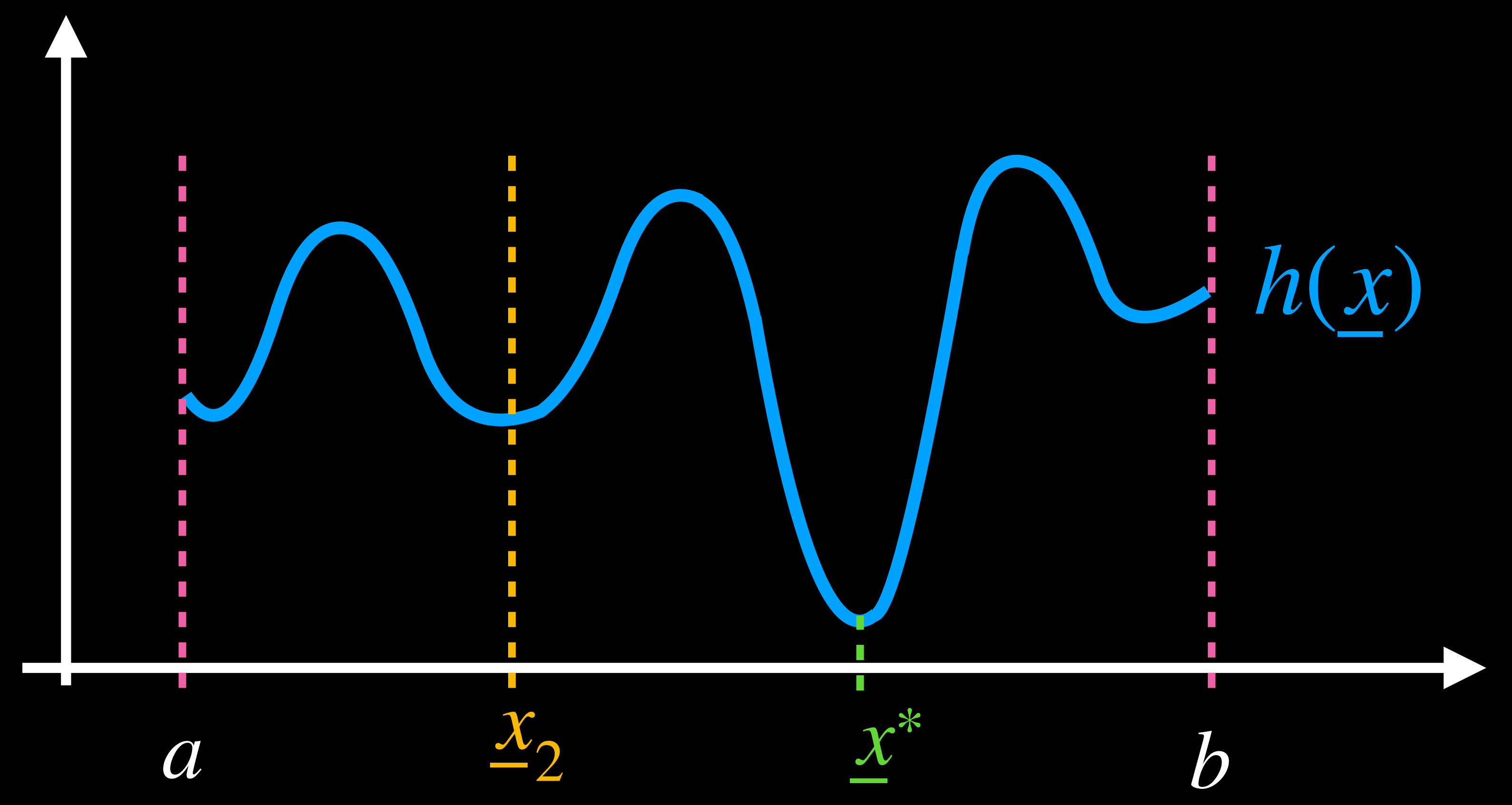
- simple
- no global convergence guarantee
- local convergence if $\lim_{n\to\infty} \alpha_n = 0$ and $\lim_{n\to\infty} \frac{\alpha_n}{\beta_n} = const$.
- converges fast ($\propto \frac{1}{n}$)
- needs uniform random numbers on the unit sphere
- need to choose $\alpha_t, \beta_t, \underline{X}_0$

useful if:

- $\mathbb{S}$ is high-dimensional and unbounded

algorithm 2: stochastic descent

$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$



gradient descent: $\underline{x}_{t+1} = \underline{x}_t - \alpha_t \boxed{\nabla h(\underline{x}_t)}$

$t \in \{1,\ldots,n\}$	steps
$\alpha_t > 0$	step size
$-\nabla h(\underline{x}_t)$	vector of steepest decrease
$\underline{x}_0$	starting point

gradient:

$$\nabla h(\underline{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} h(\underline{x}) \\ \vdots \\ \frac{\partial}{\partial x_d} h(\underline{x}) \end{pmatrix}, \qquad \frac{\partial}{\partial x_1} h(\underline{x}) \approx \left(h\begin{pmatrix} x_1 + \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix} - h\begin{pmatrix} x_1 - \Delta x \\ x_2 \\ \vdots \\ x_d \end{pmatrix}\right) \cdot \frac{1}{2\Delta x}$$

→ approximating $\nabla h(\underline{x})$ has $2d$ evaluations of h (expensive)

dot product: $\underline{x} \cdot \underline{y} = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_d \end{pmatrix} = x_1 y_1 + \ldots + x_d y_d$

directional derivative:

$$\nabla h(\underline{x}) \cdot \frac{\underline{y}}{|\underline{y}|} \approx \frac{\Delta h(\underline{x},\underline{y})}{2|\underline{y}|}, \qquad \Delta h(\underline{x},\underline{y}) := h(\underline{x}+\underline{y}) - h(\underline{x}-\underline{y})$$

→ approximation has 2 evaluations of h (cheaper)

stochastic descent:

$$\underline{X}_{t+1} = \underline{X}_t - \frac{\alpha_t}{2\beta_t} \Delta h(\underline{X}_t, \beta_t \underline{U}_t)\, \underline{U}_t$$

$\underline{U}_t \sim \mathcal{U}(\mathbb{S})$

$\mathbb{S}$ be the d-dimensional unit sphere

β_t sampling radius

properties:

- simple
- no global convergence guarantee
- local convergence if $\lim_{n\to\infty} \alpha_n = 0$ and $\lim_{n\to\infty} \frac{\alpha_n}{\beta_n} = const$.
- converges fast ($\propto \frac{1}{n}$)
- needs uniform random numbers on the unit sphere
- need to choose $\alpha_t, \beta_t, \underline{X}_0$

useful if:

- $\mathbb{S}$ is high-dimensional and unbounded
- h is convex or $\underline{X}_0$ is close to $\underline{x}^*$

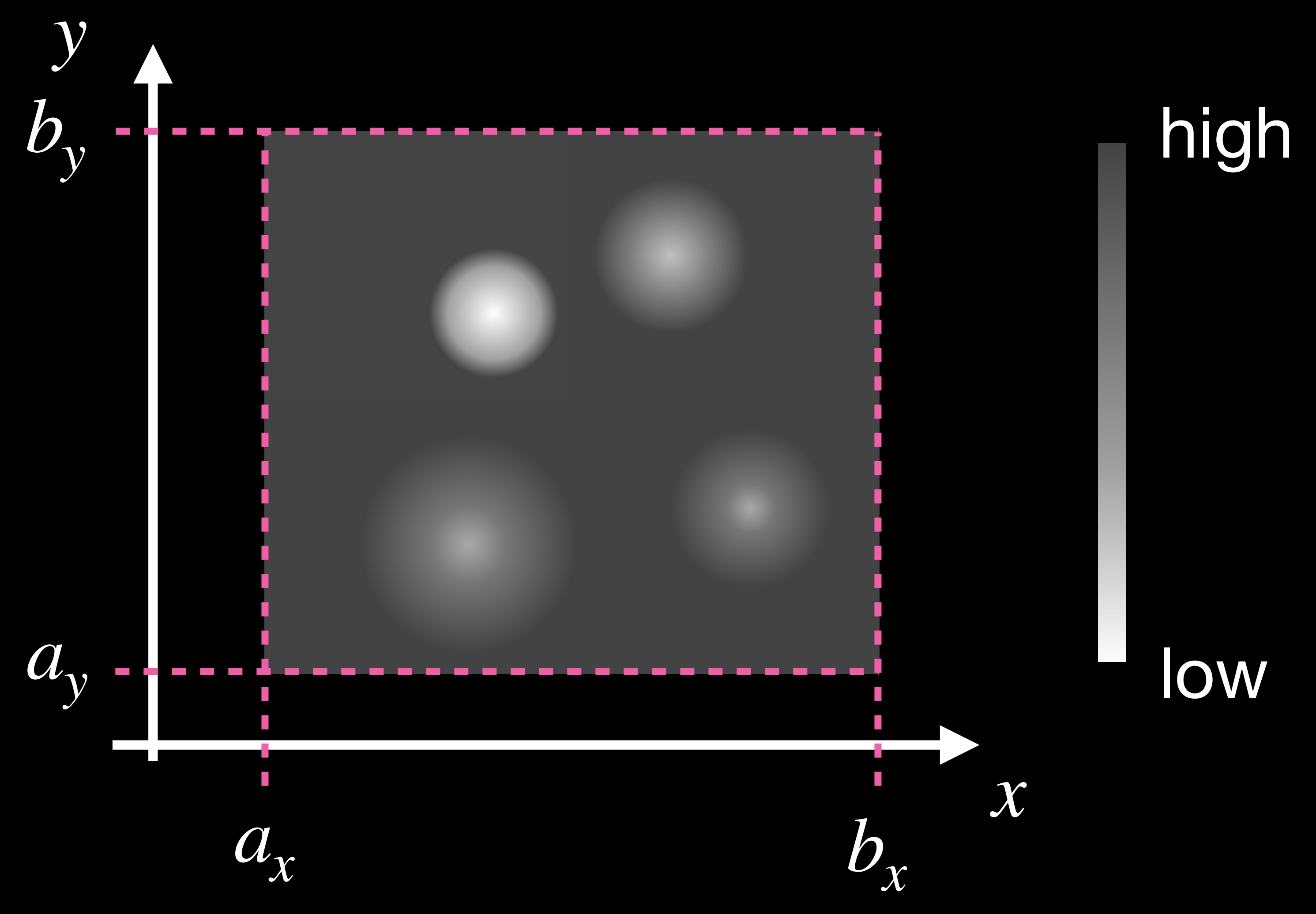
algorithm 3: random pursuit

algorithm 3: random pursuit

$$\underline{x}^* := \arg\min_{\underline{x} \in \mathbb{S}} \left(h(\underline{x}) \right)$$

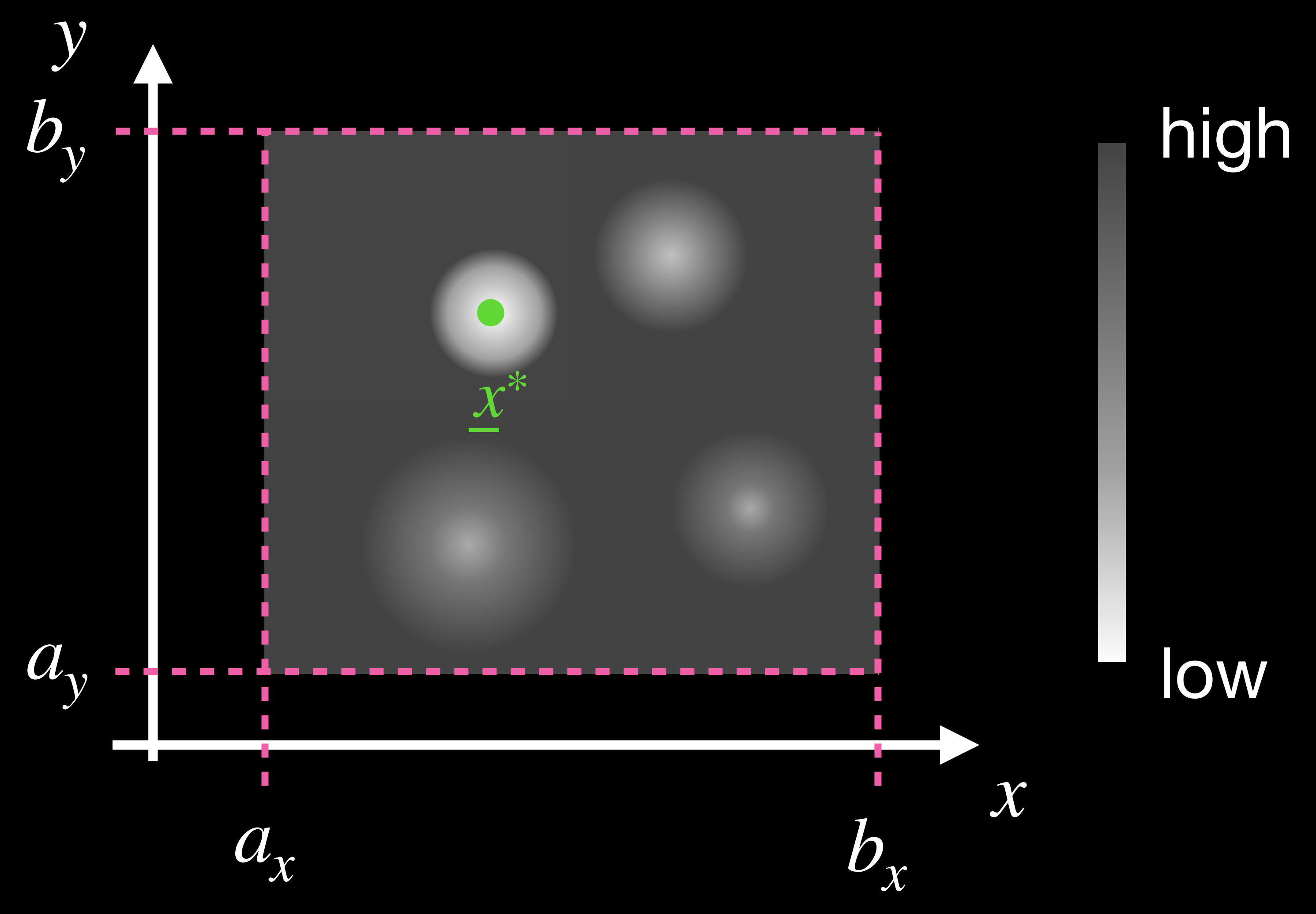
algorithm 3: random pursuit

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$



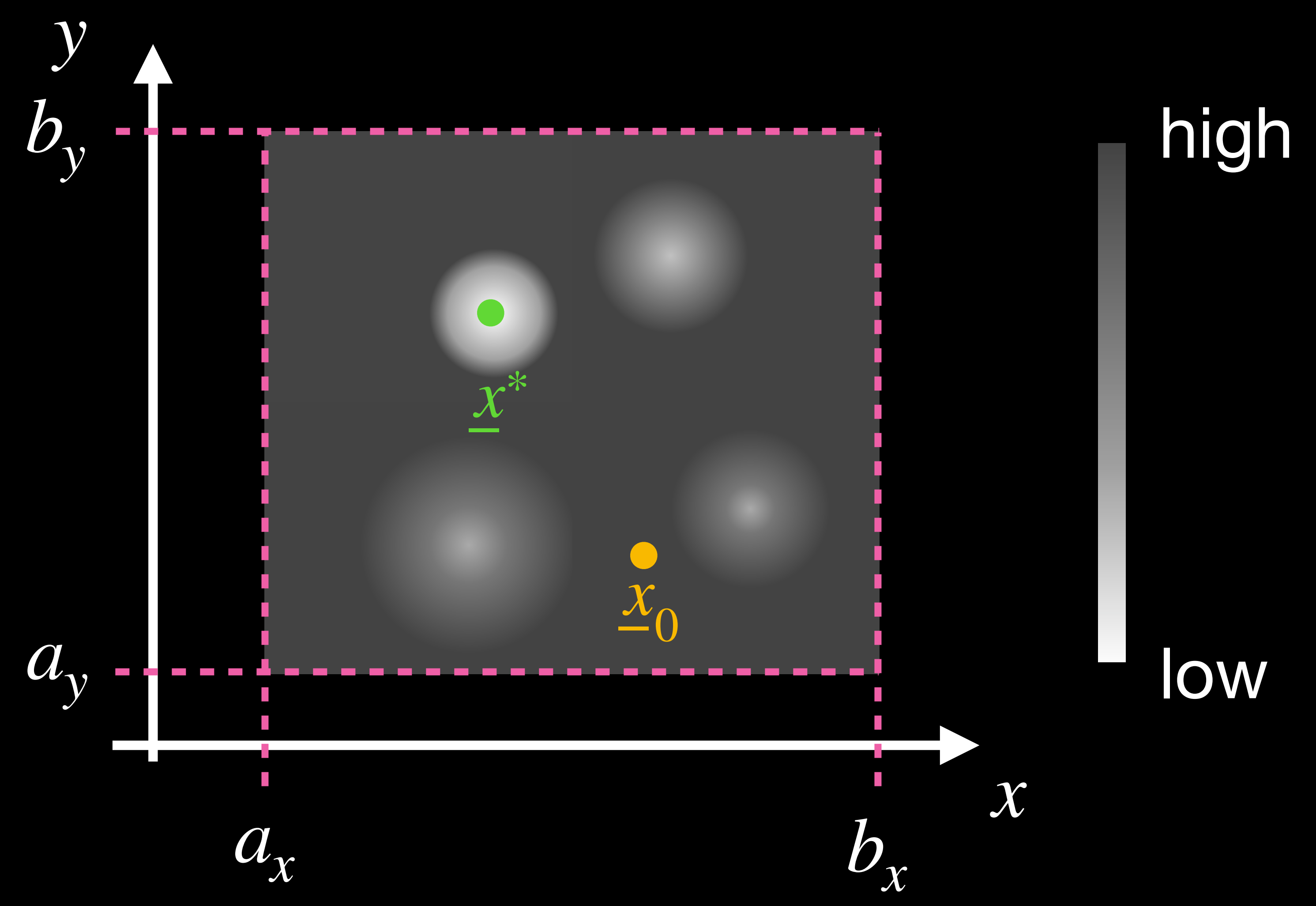
algorithm 3: random pursuit

$$\underline{x}^* := \arg\min_{\underline{x} \in \mathbb{S}} \left(h(\underline{x}) \right)$$



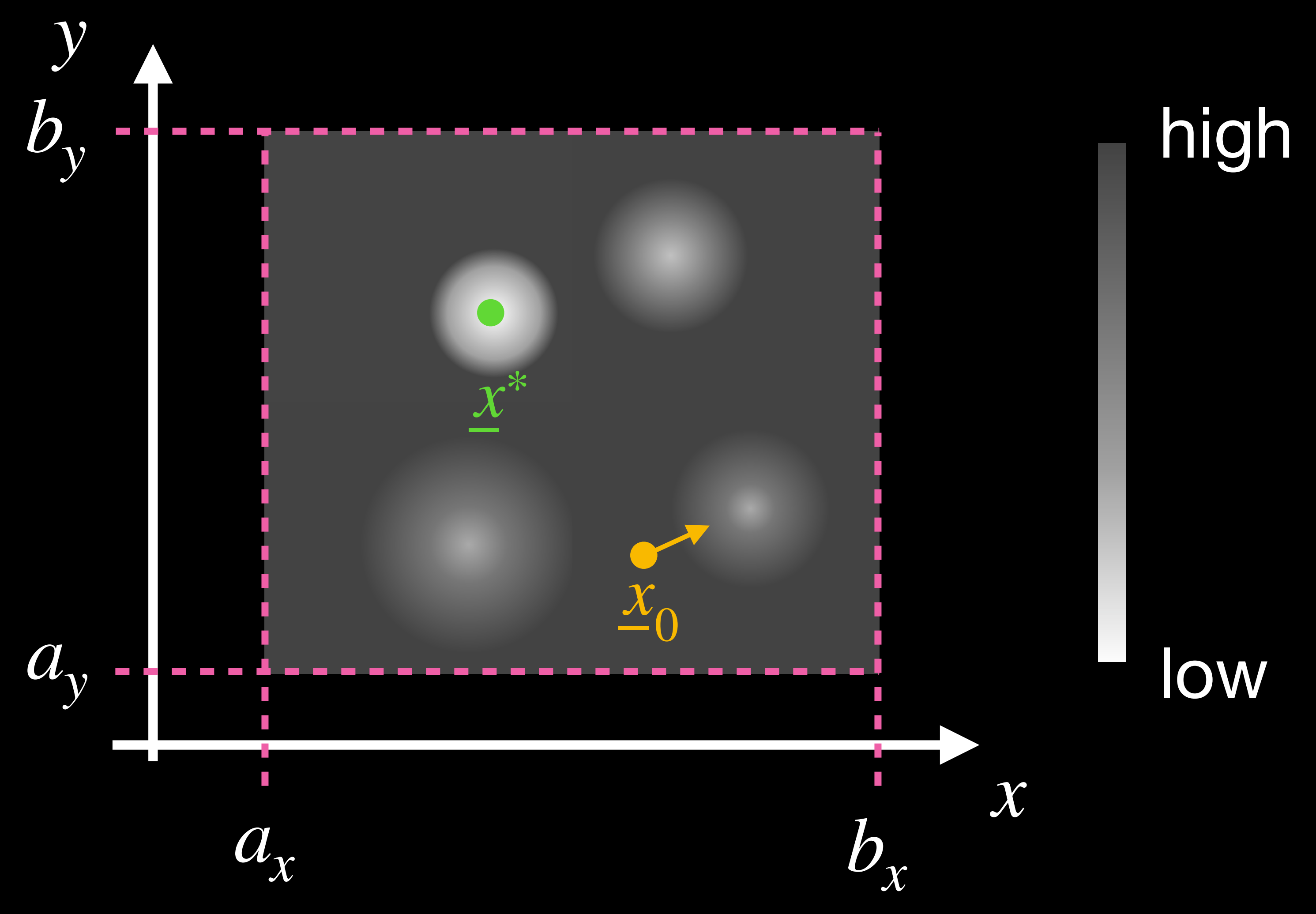
algorithm 3: random pursuit

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \big(h(\underline{x})\big)$$



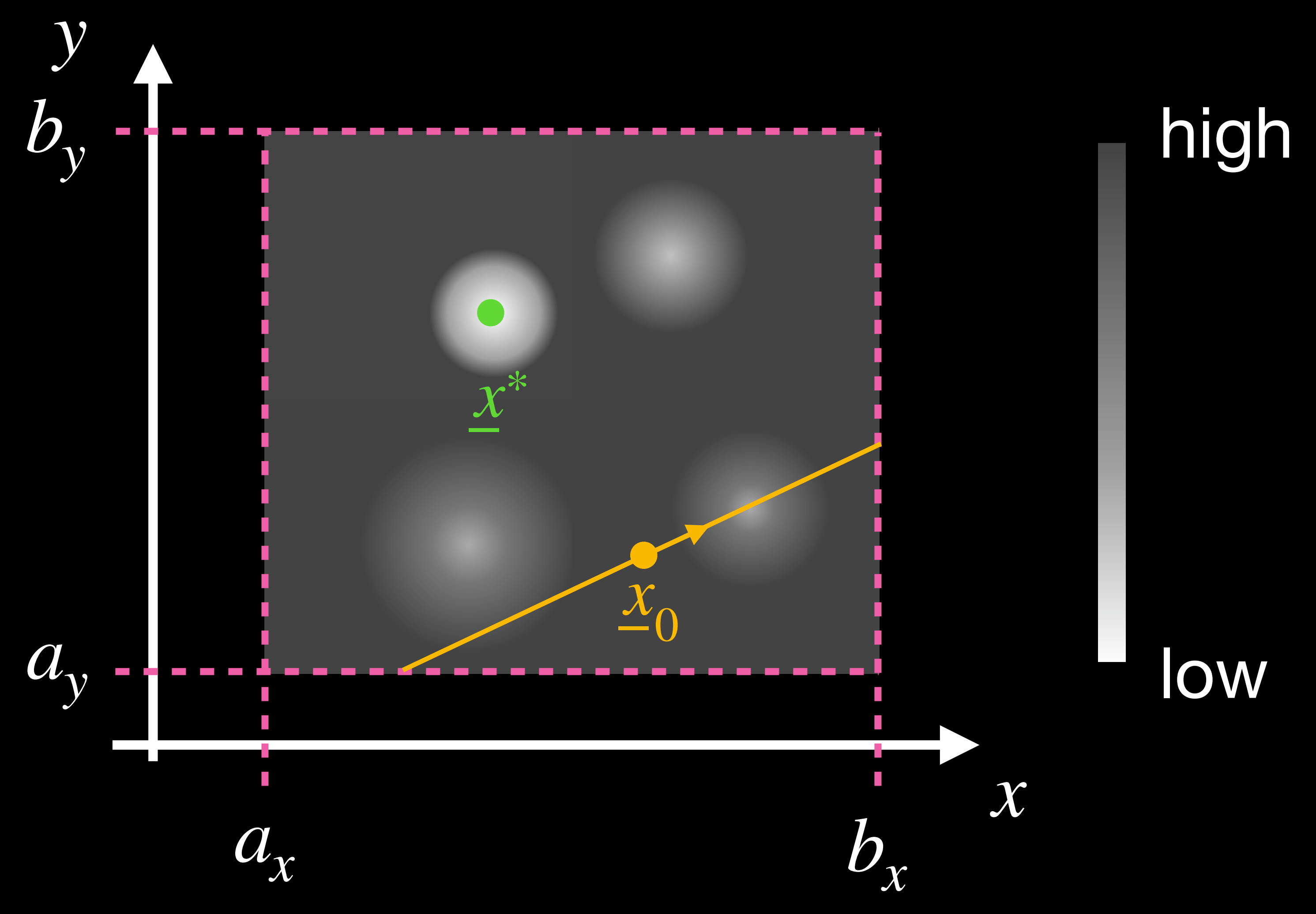
algorithm 3: random pursuit

$$\underline{x}^* := \underset{\underline{x} \in \mathbb{S}}{\arg\min} \left(h(\underline{x}) \right)$$



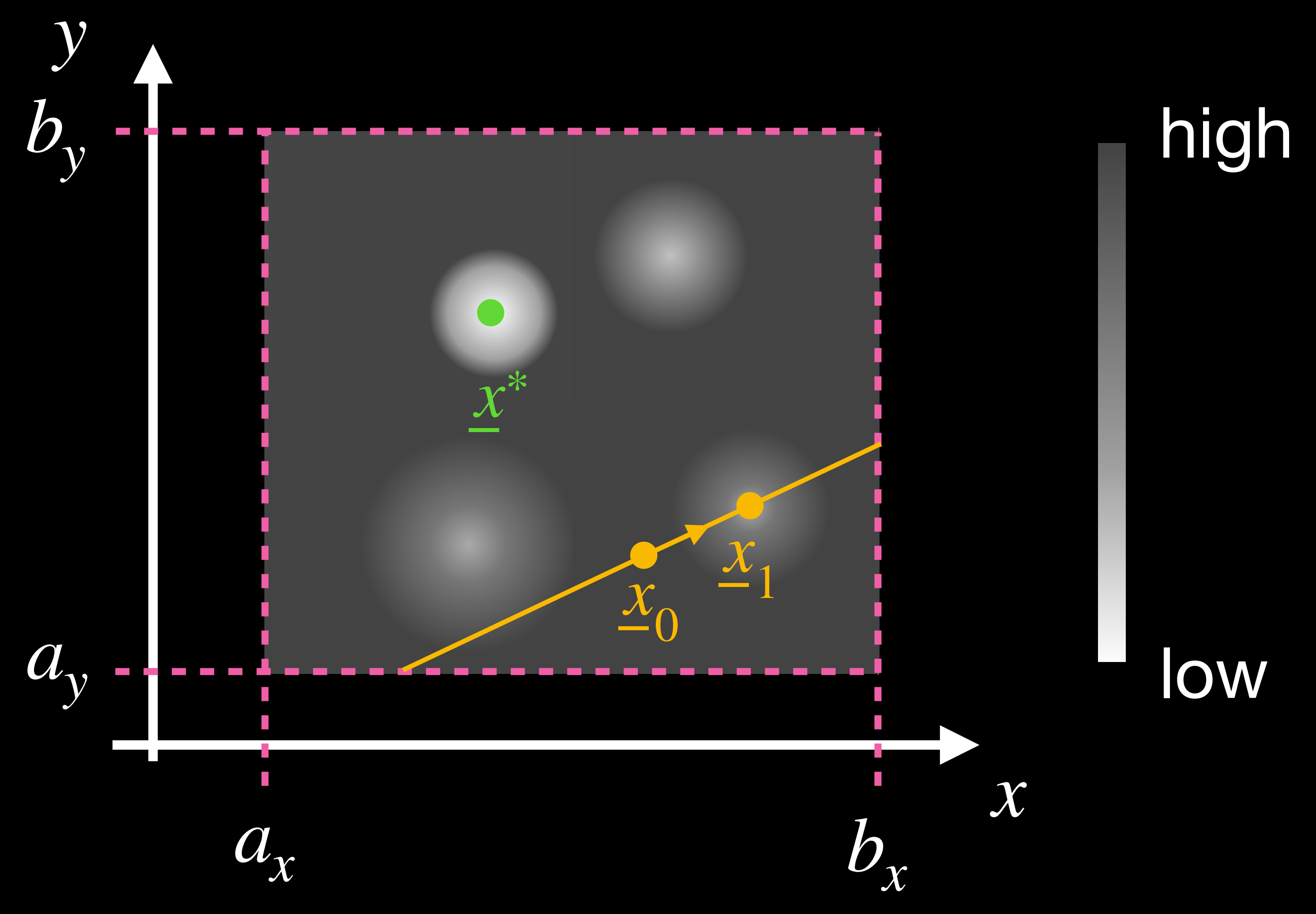
algorithm 3: random pursuit

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$



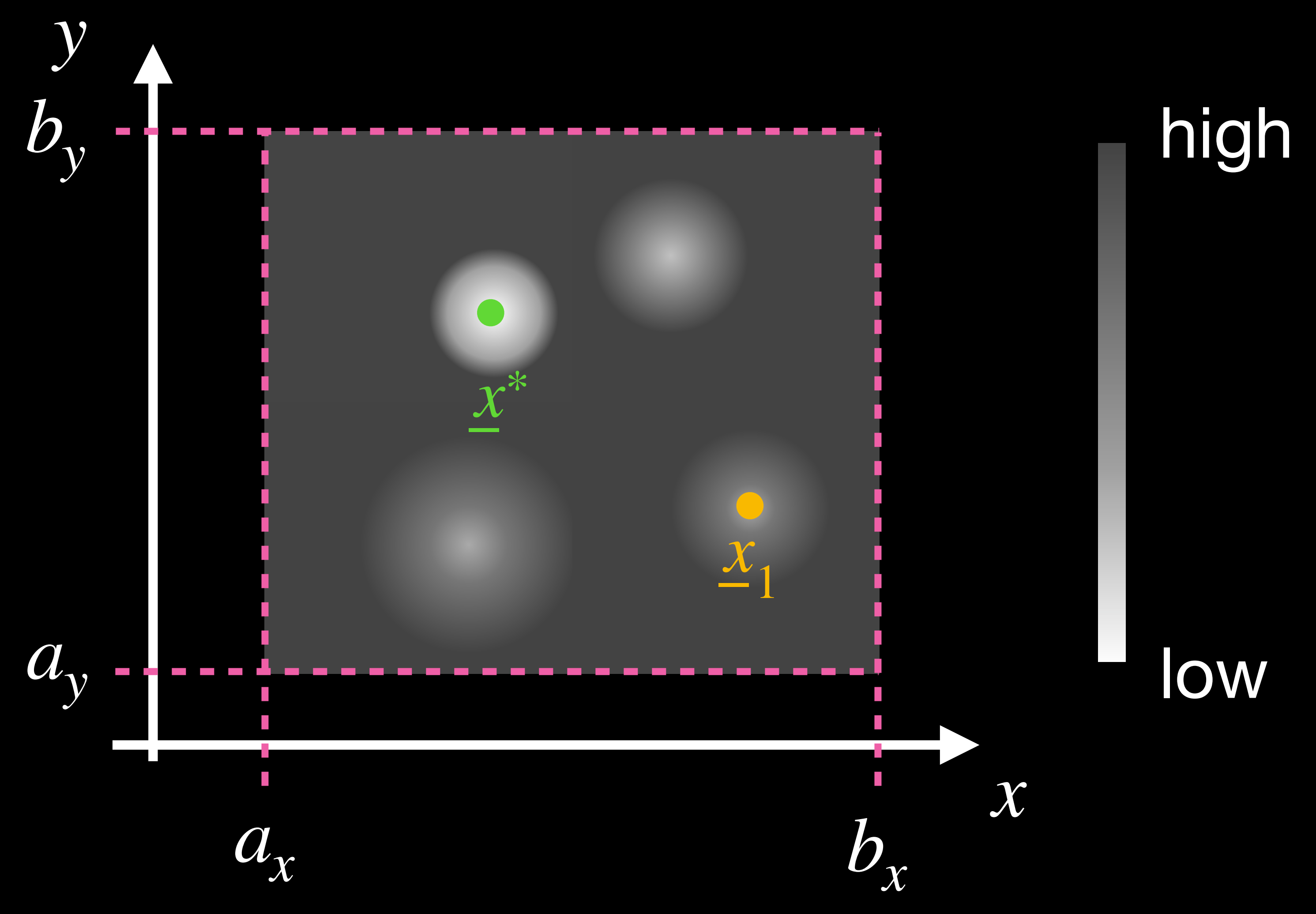
algorithm 3: random pursuit

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$



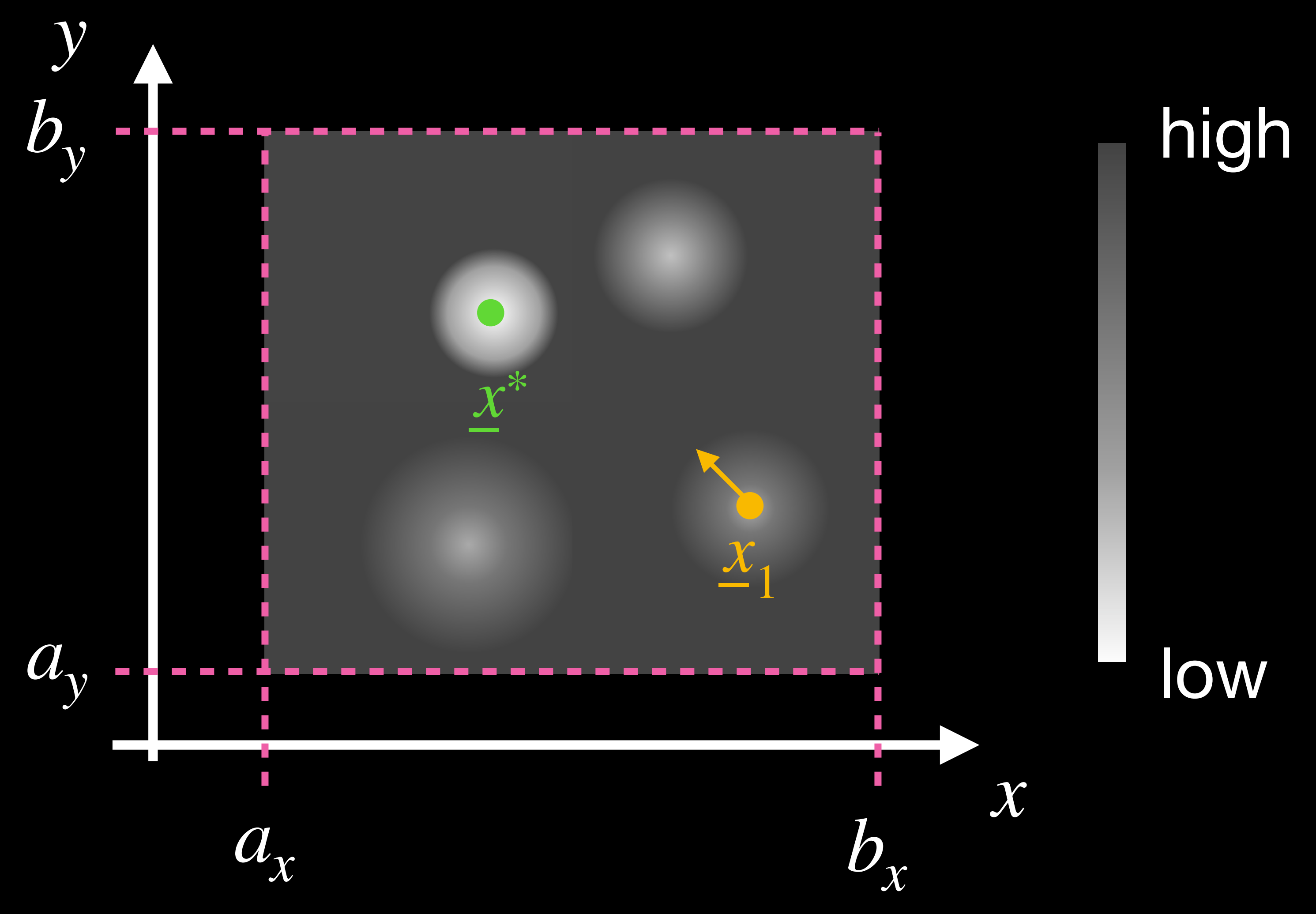
algorithm 3: random pursuit

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$



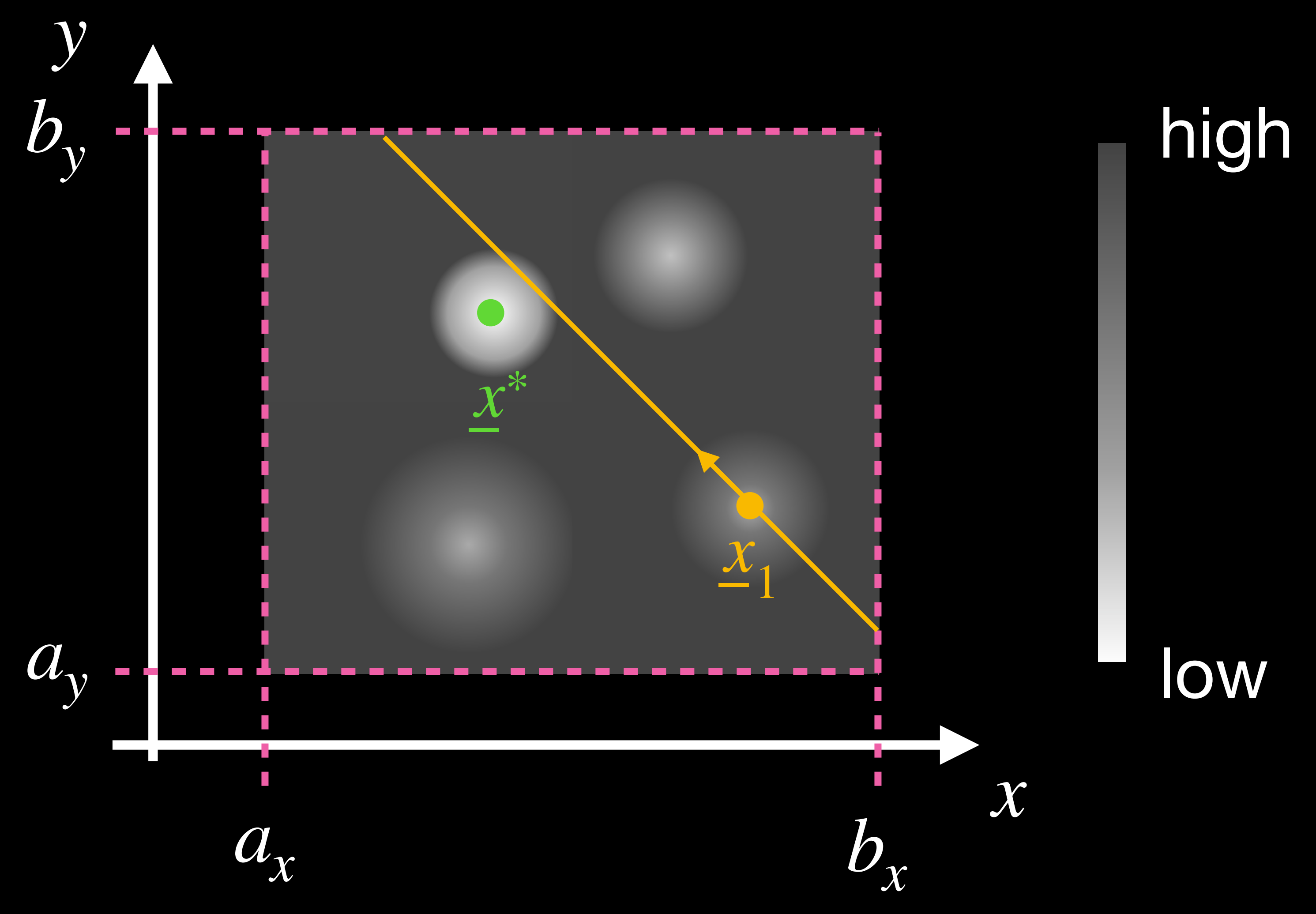
algorithm 3: random pursuit

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$



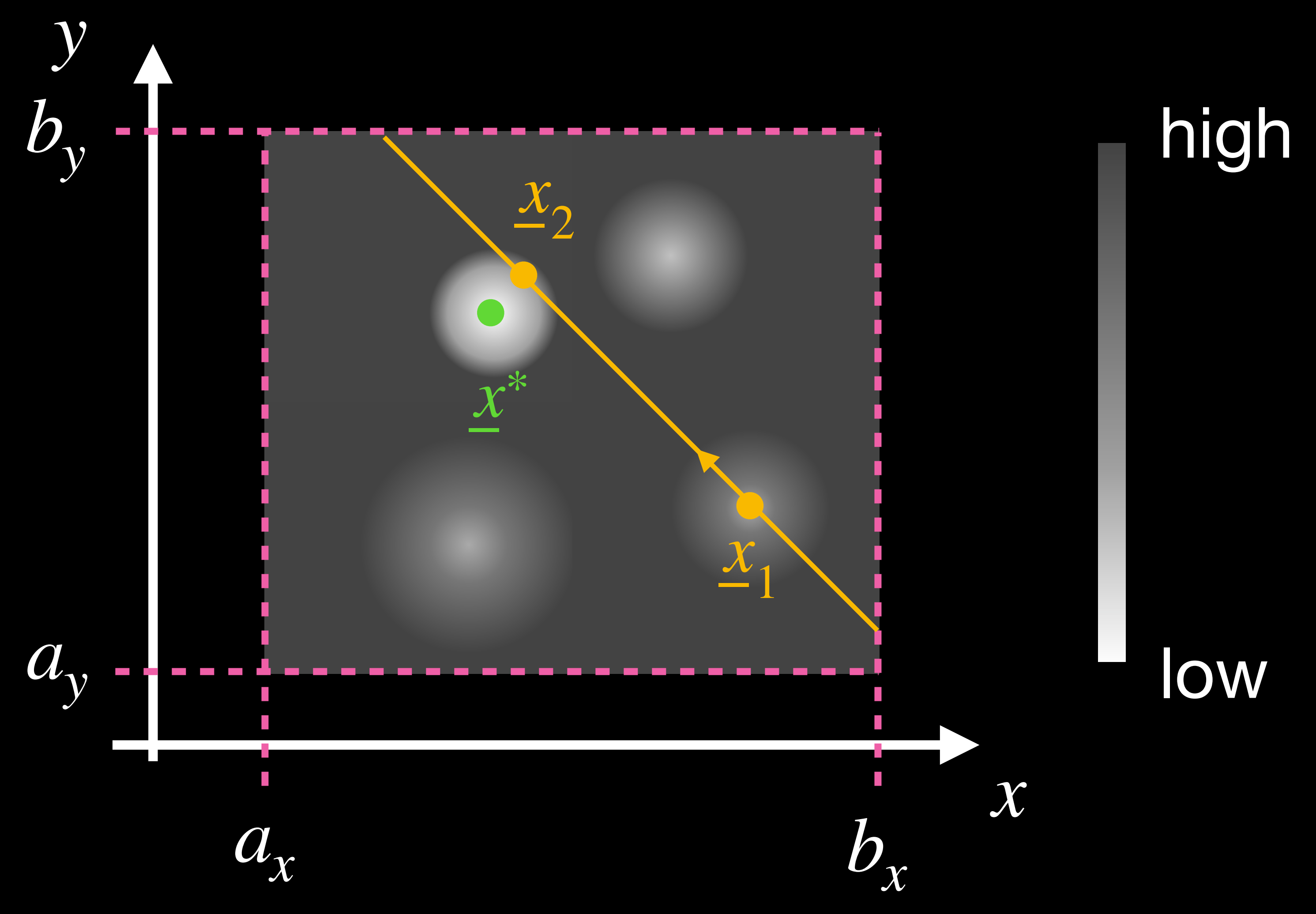
algorithm 3: random pursuit

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$



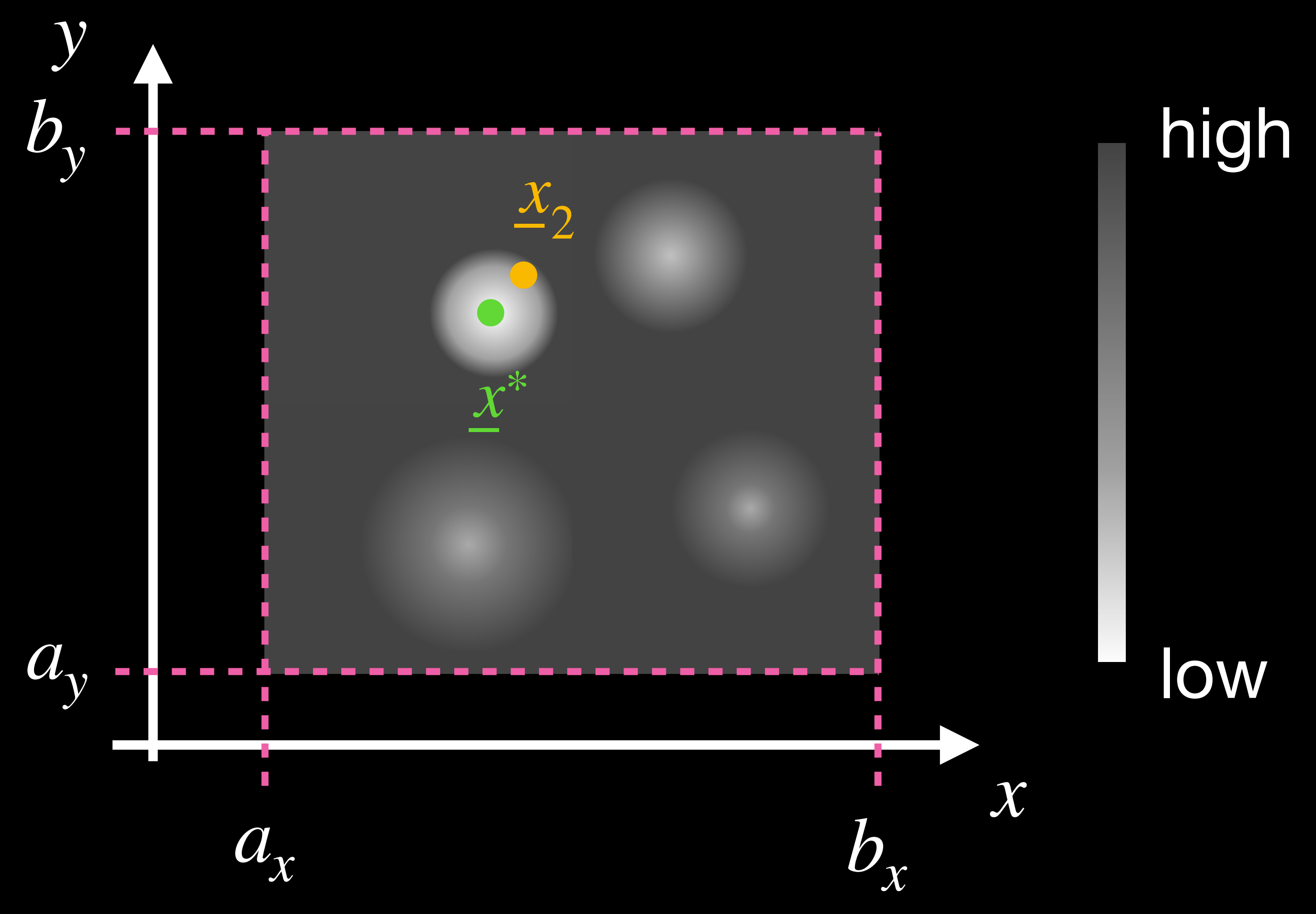
algorithm 3: random pursuit

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$



algorithm 3: random pursuit

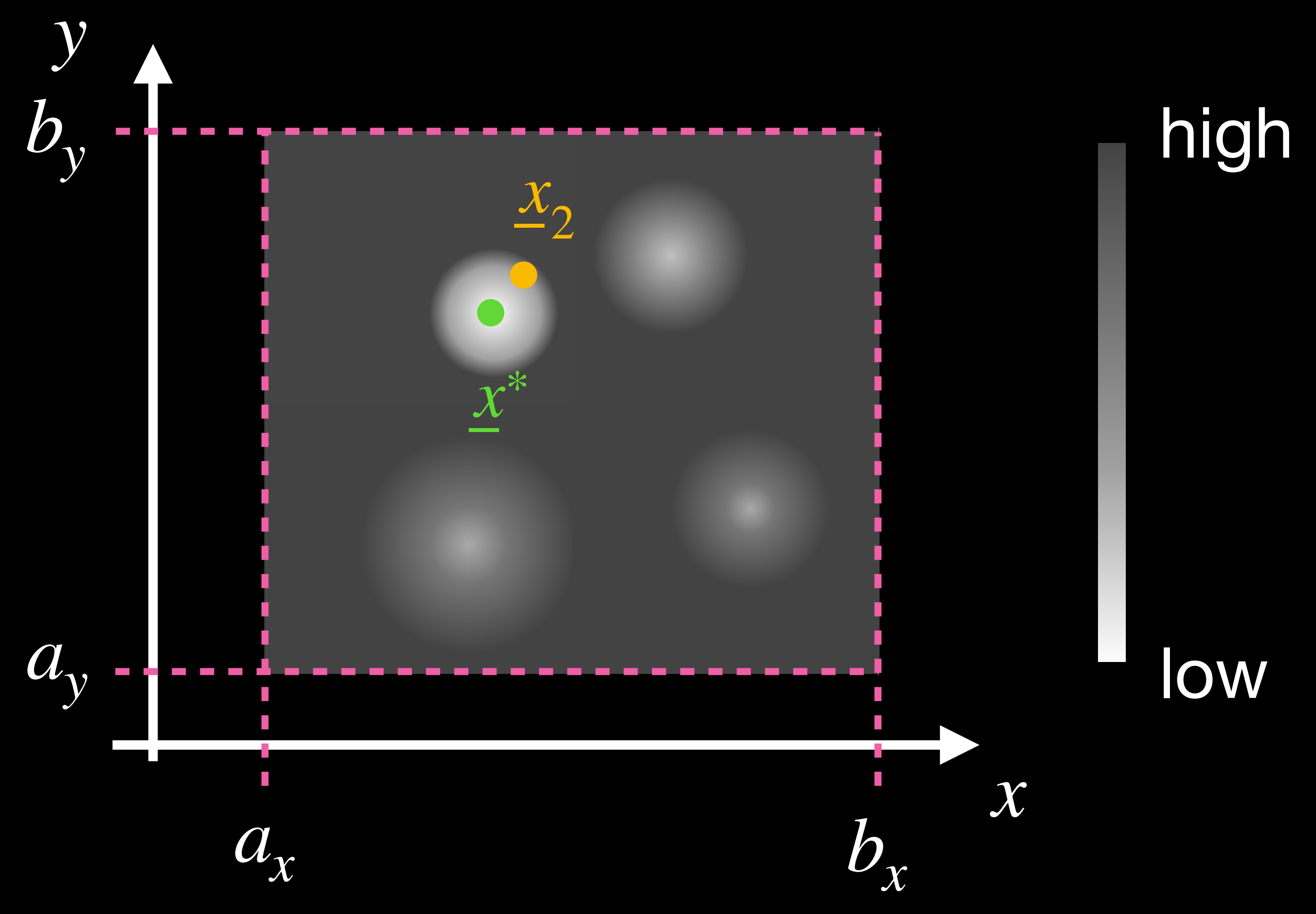
$$\underline{x}^* := \arg\min_{\underline{x} \in \mathbb{S}} \left(h(\underline{x}) \right)$$



algorithm 3: random pursuit

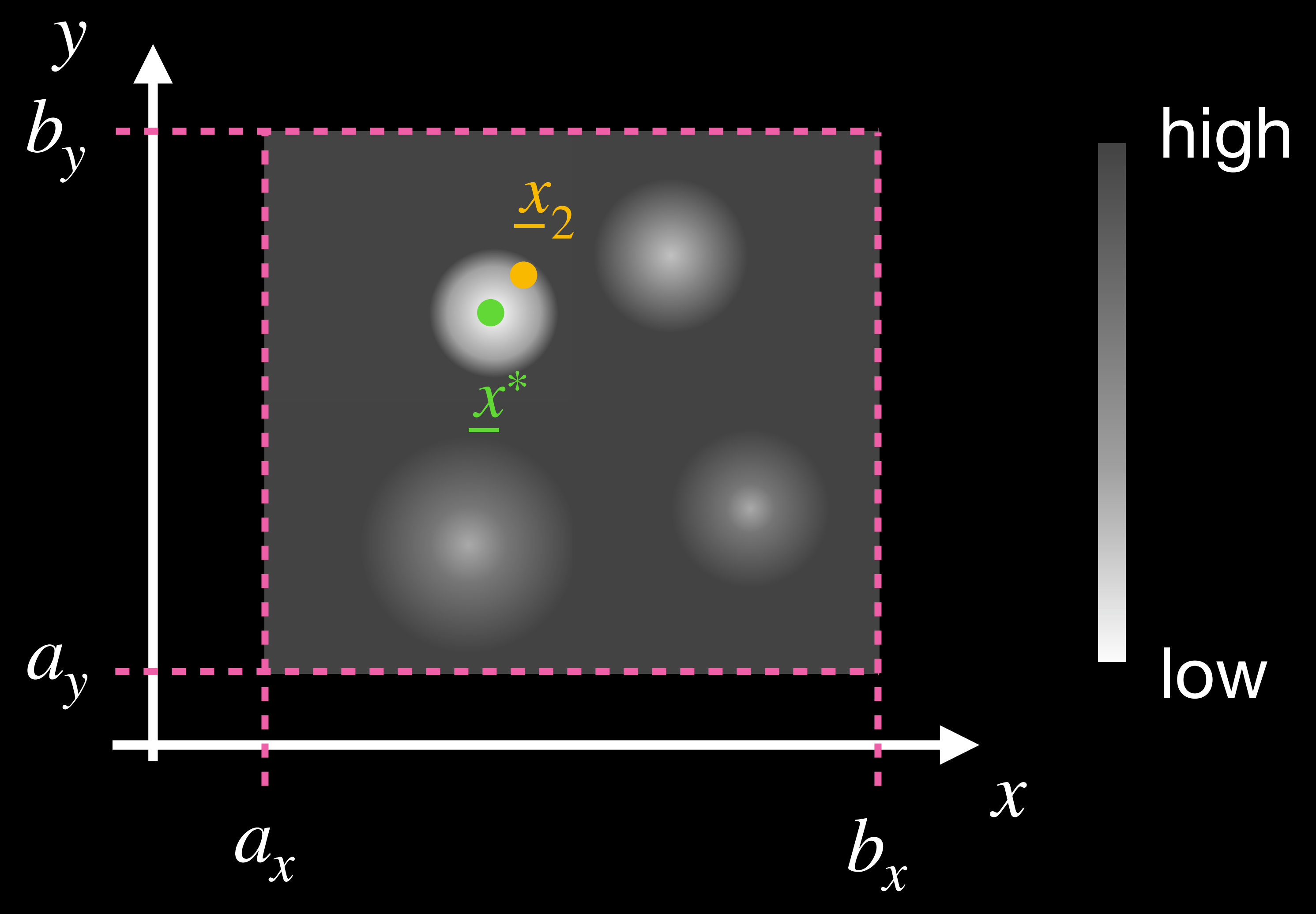
$$\underline{x}^* := \arg\min_{\underline{x} \in \mathbb{S}} \left(h(\underline{x}) \right)$$

$\underline{X}_{t+1}$



algorithm 3: random pursuit

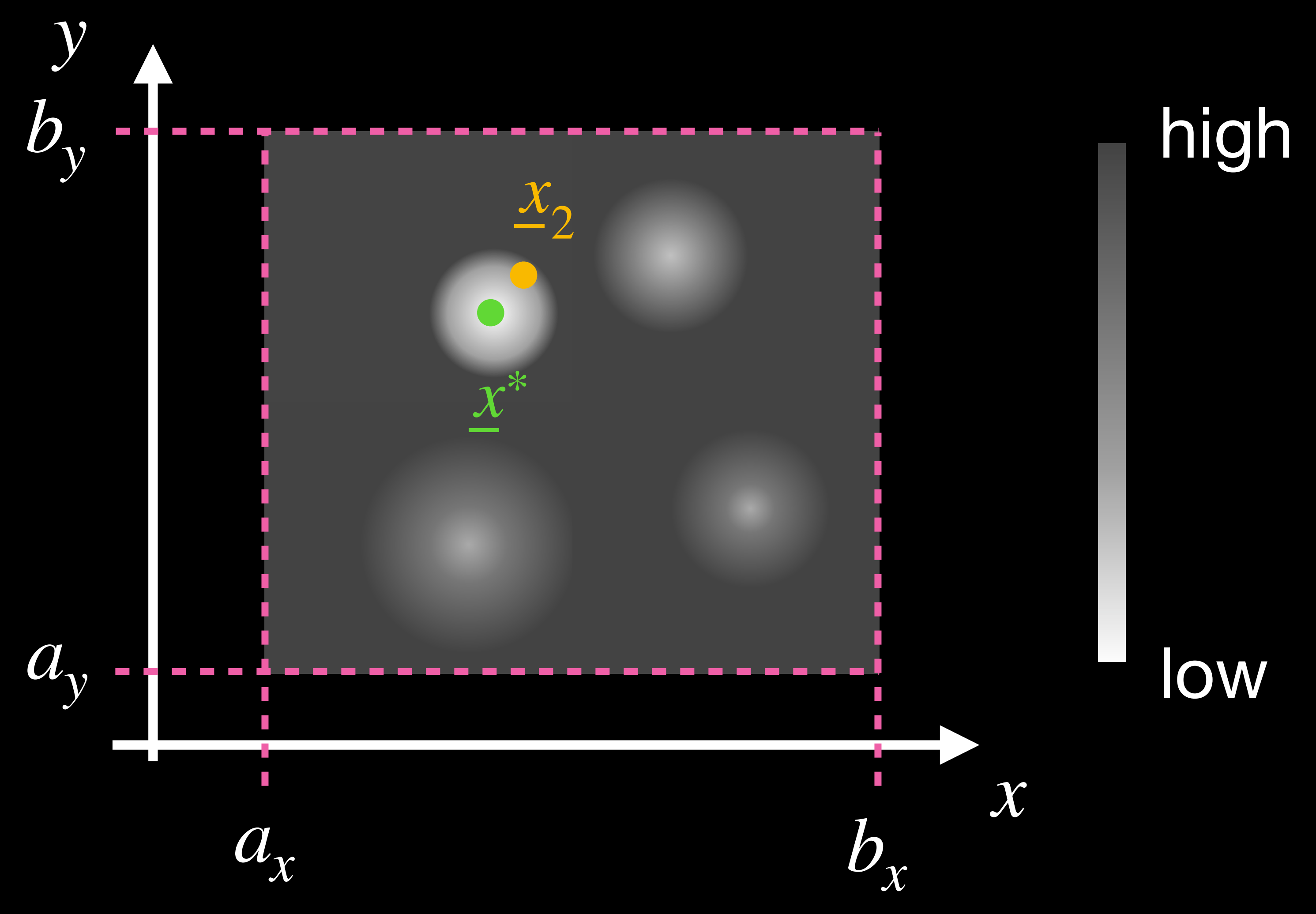
$$\underline{x}^* := \arg\min_{\underline{x} \in \mathbb{S}} \left(h(\underline{x}) \right)$$



$$\underline{X}_{t+1} = \underline{X}_t + linesearch(\underline{X}_t, \underline{U}_t)$$

algorithm 3: random pursuit

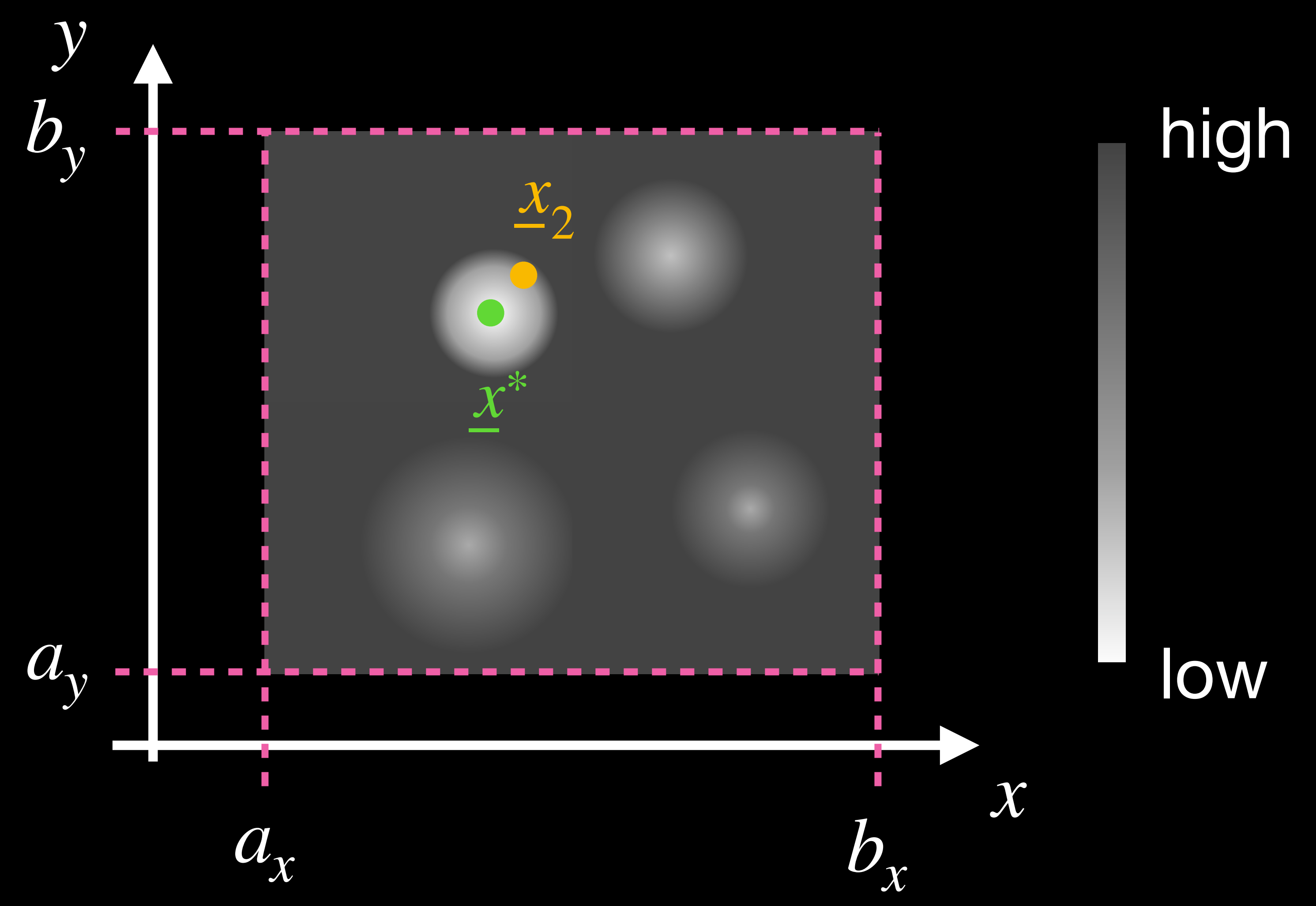
$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$



$$\underline{X}_{t+1} = \underline{X}_t + linesearch(\underline{X}_t, \underline{U}_t) \qquad \underline{U}_t \sim \mathcal{U}(\mathbb{S})$$

algorithm 3: random pursuit

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

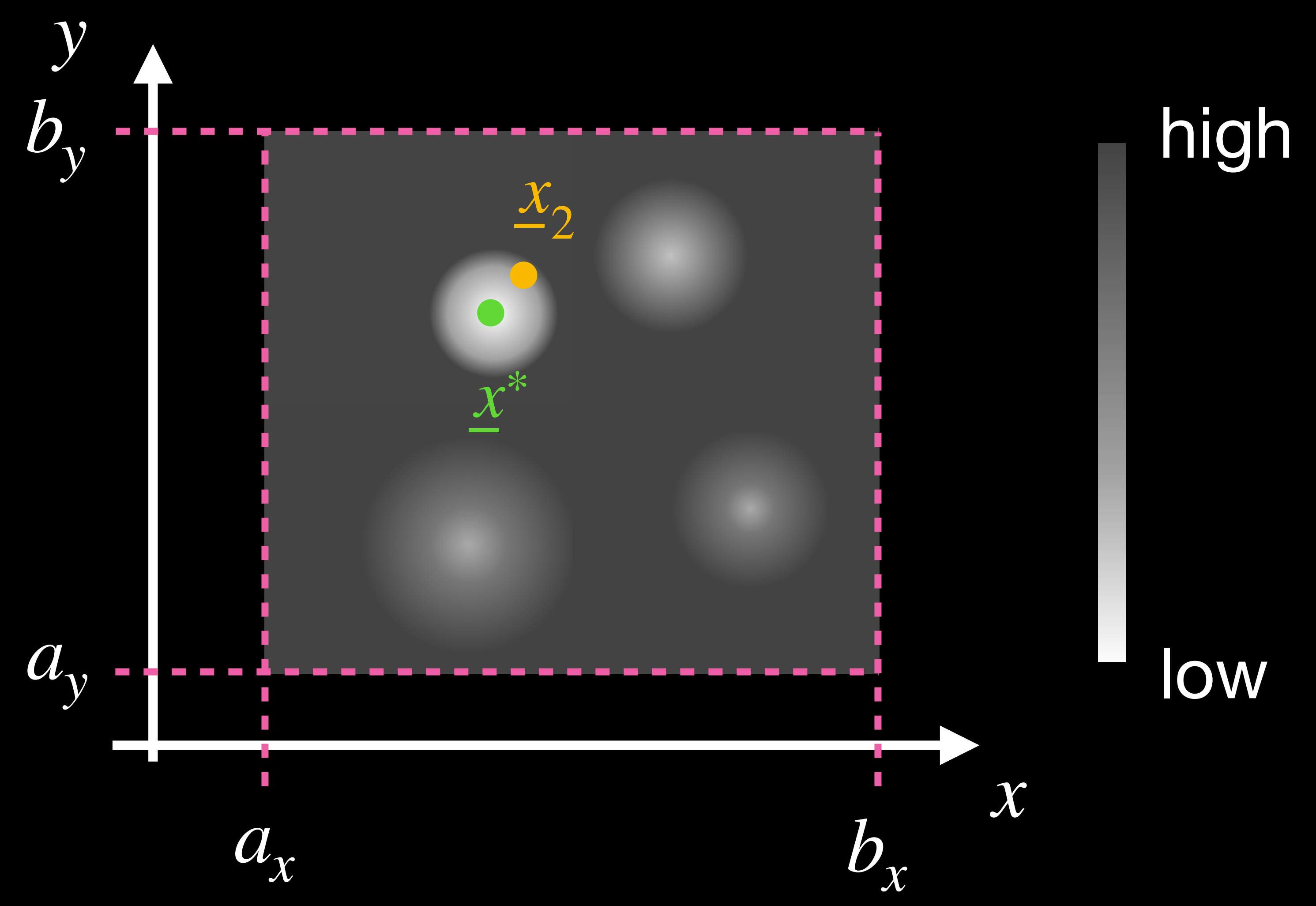


$$\underline{X}_{t+1} = \underline{X}_t + linesearch(\underline{X}_t, \underline{U}_t) \qquad \underline{U}_t \sim \mathcal{U}(\mathbb{S})$$

$\mathbb{S}$ be the d-dimensional unit sphere

algorithm 3: random pursuit

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$



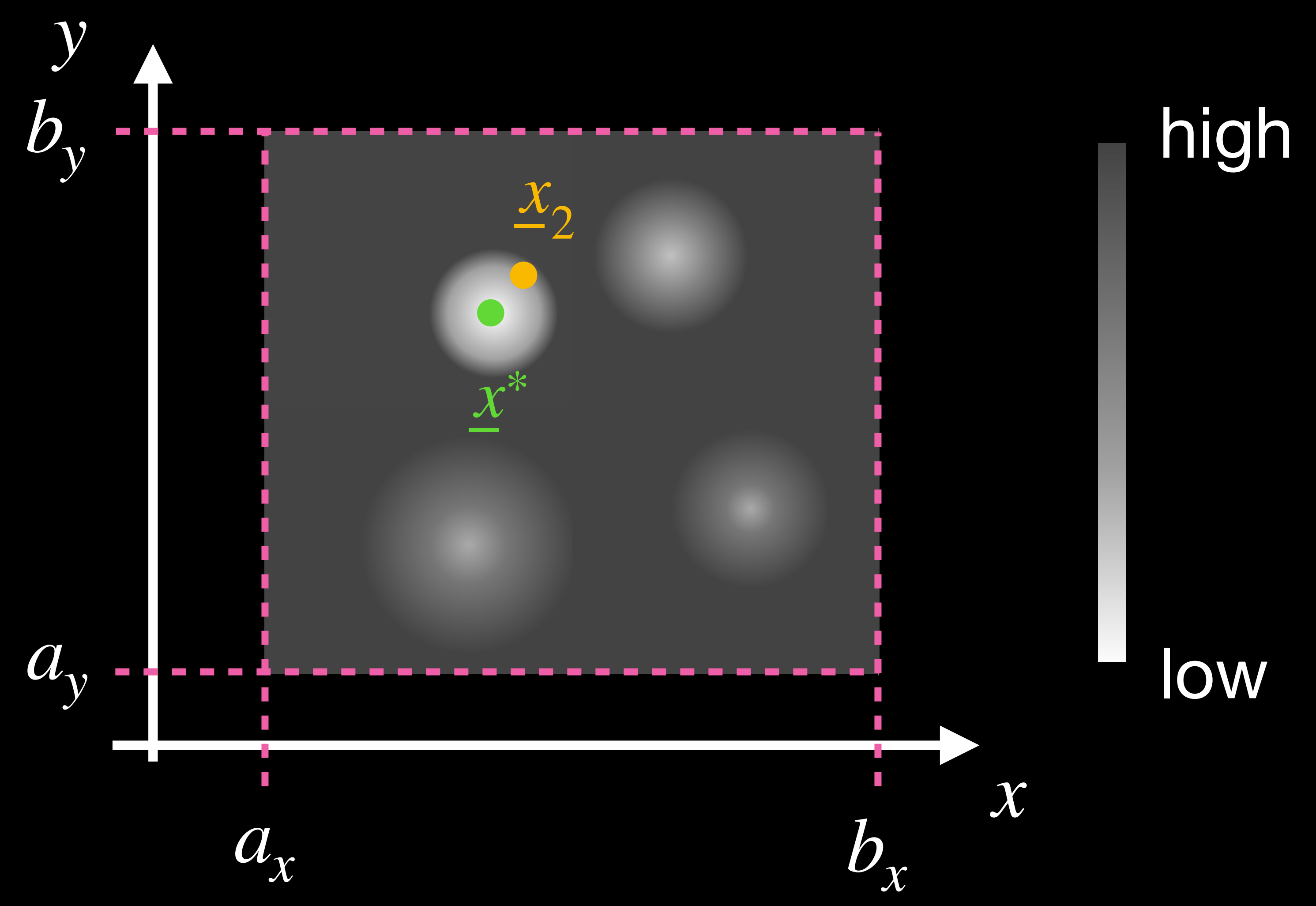
$$\underline{X}_{t+1} = \underline{X}_t + linesearch(\underline{X}_t, \underline{U}_t) \qquad \underline{U}_t \sim \mathcal{U}(\mathbb{S})$$

$\mathbb{S}$ be the d-dimensional unit sphere

$$linesearch_h(\underline{X}_t, \underline{U}_t) := \arg\min_{\beta} \left(h(\underline{X}_t + \beta \cdot \underline{U}_t)\right) \cdot \underline{U}_t$$

algorithm 3: random pursuit

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$



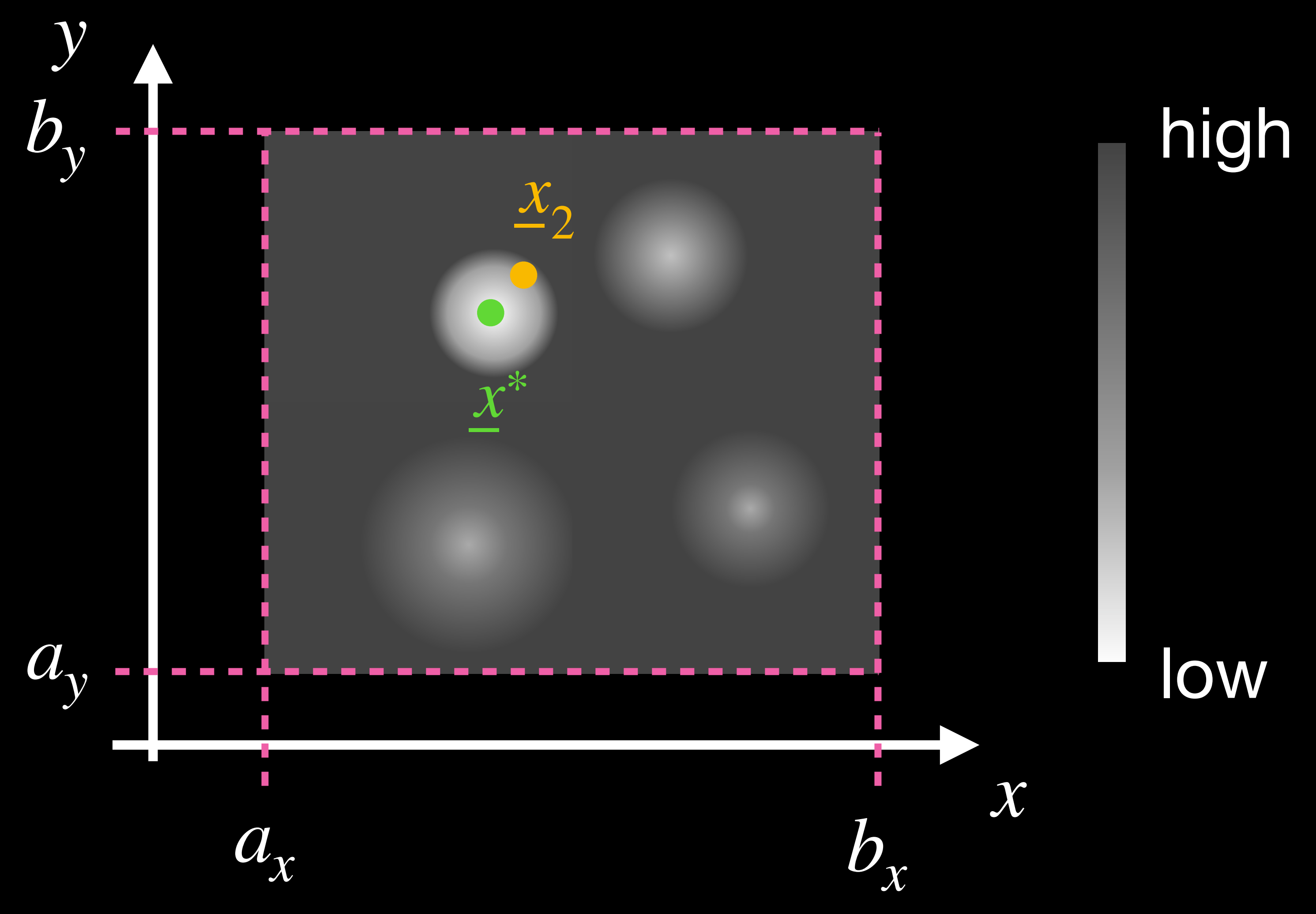
$$\underline{X}_{t+1} = \underline{X}_t + linesearch(\underline{X}_t, \underline{U}_t) \qquad \underline{U}_t \sim \mathcal{U}(\mathbb{S})$$

$\mathbb{S}$ be the d-dimensional unit sphere

$$linesearch_h(\underline{X}_t, \underline{U}_t) := \arg\min_{\beta} \left(h(\underline{X}_t + \beta \cdot \underline{U}_t)\right) \cdot \underline{U}_t$$

β is distance between X_t and X_{t+1}

algorithm 3: random pursuit



$$\underline{x}^* := \arg\min_{\underline{x} \in \mathbb{S}} \left(h(\underline{x})\right)$$

$$\underline{X}_{t+1} = \underline{X}_t + linesearch(\underline{X}_t, \underline{U}_t) \qquad \underline{U}_t \sim \mathscr{U}(\mathbb{S})$$

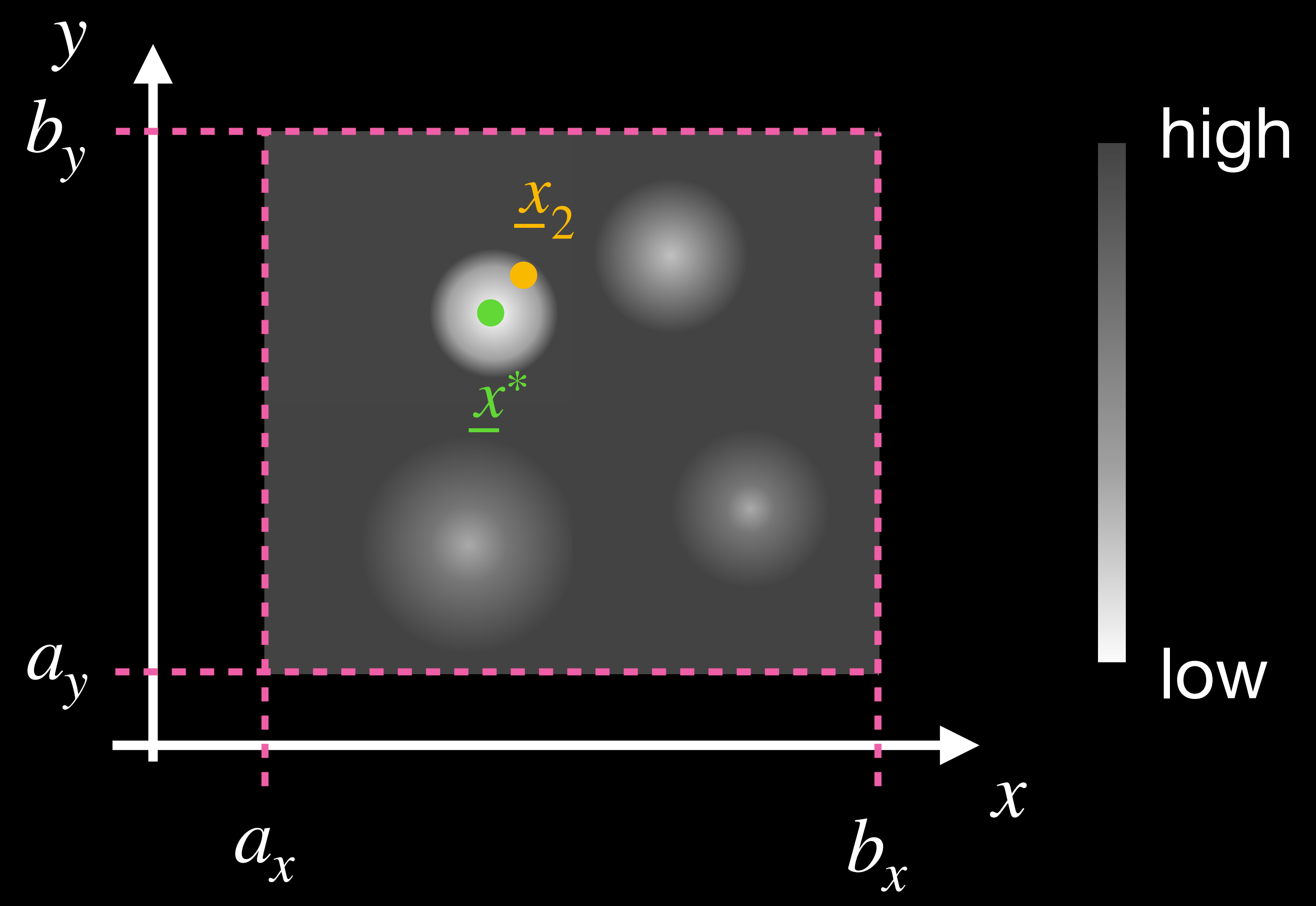
$\mathbb{S}$ be the d-dimensional unit sphere

$$linesearch_h(\underline{X}_t, \underline{U}_t) := \arg\min_{\beta} \left(h(\underline{X}_t + \beta \cdot \underline{U}_t)\right) \cdot \underline{U}_t$$

β is distance between X_t and X_{t+1}

(finds the vector to the point with minimal value on the line that goes through $\underline{X}_t$ in the direction $\underline{U}_t$)

algorithm 3: random pursuit



$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \big(h(\underline{x})\big)$$

$$\underline{X}_{t+1} = \underline{X}_t + linesearch(\underline{X}_t, \underline{U}_t) \qquad \underline{U}_t \sim \mathcal{U}(\mathbb{S})$$

$\mathbb{S}$ be the d-dimensional unit sphere

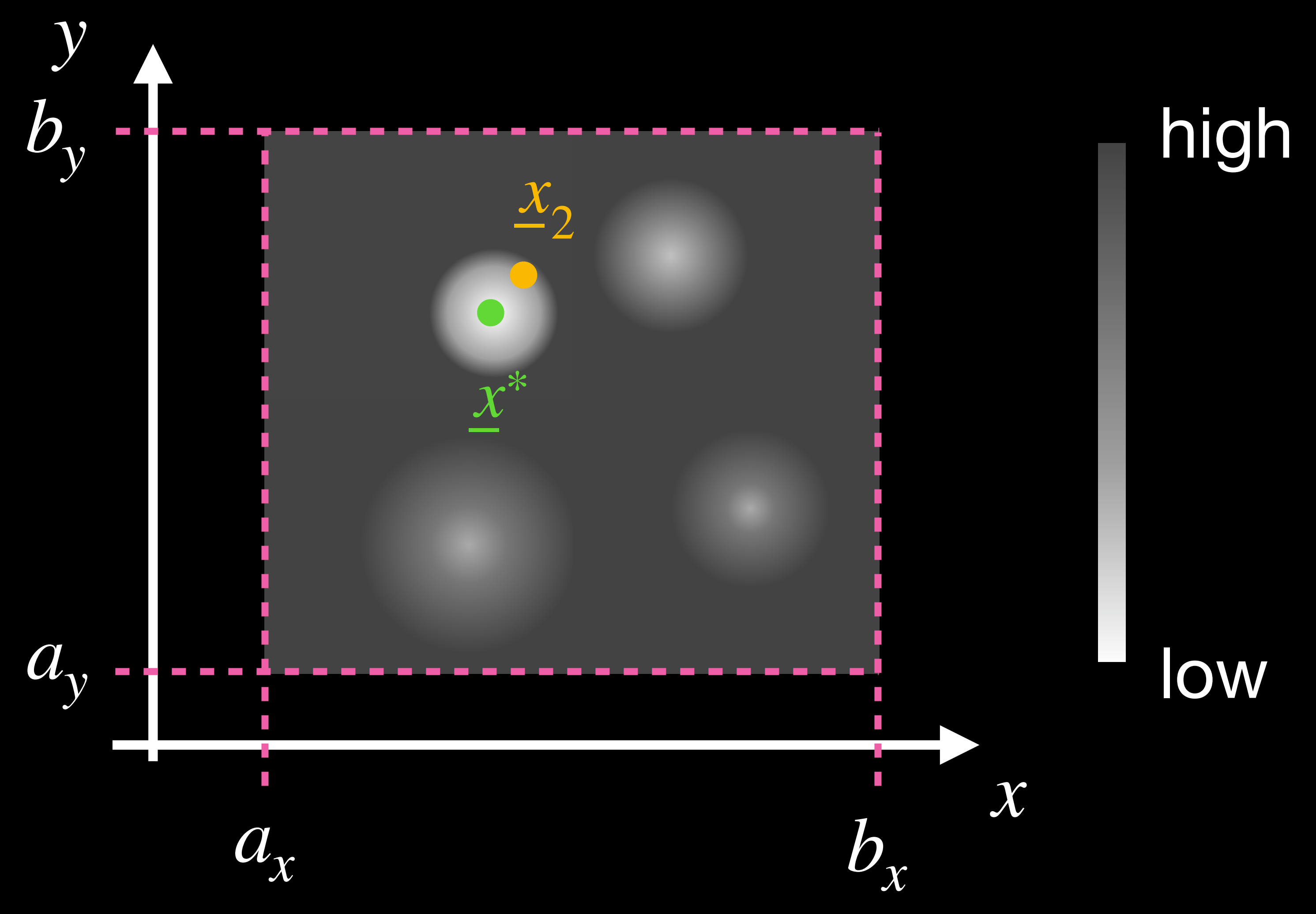
$$linesearch_h(\underline{X}_t, \underline{U}_t) := \arg\min_{\beta} \Big(h(\underline{X}_t + \beta \cdot \underline{U}_t)\Big) \cdot \underline{U}_t$$

β is distance between X_t and X_{t+1}

(finds the vector to the point with minimal value on the line that goes through $\underline{X}_t$ in the direction $\underline{U}_t$)

$\rightarrow$ reduction to 1d-optimisation problem

algorithm 3: random pursuit



$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

$$\underline{X}_{t+1} = \underline{X}_t + linesearch(\underline{X}_t, \underline{U}_t) \qquad \underline{U}_t \sim \mathcal{U}(\mathbb{S})$$

$\mathbb{S}$ be the d-dimensional unit sphere

$$linesearch_h(\underline{X}_t, \underline{U}_t) := \arg\min_{\beta} \left(h(\underline{X}_t + \beta \cdot \underline{U}_t)\right) \cdot \underline{U}_t$$

β is distance between X_t and X_{t+1}

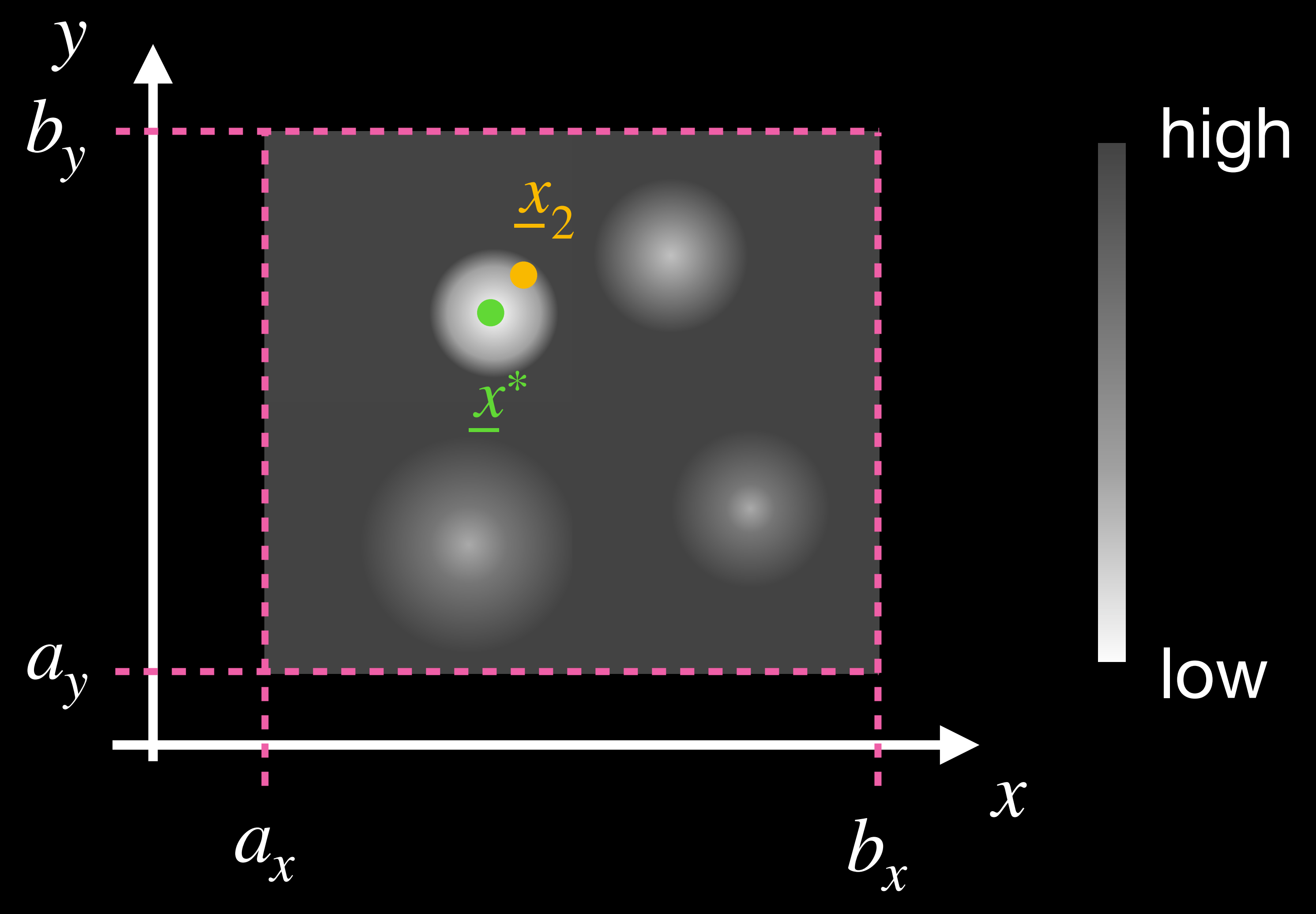
(finds the vector to the point with minimal value on the line that goes through $\underline{X}_t$ in the direction $\underline{U}_t$)

→ reduction to 1d-optimisation problem

properties:

algorithm 3: random pursuit

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$



$$\underline{X}_{t+1} = \underline{X}_t + linesearch(\underline{X}_t, \underline{U}_t) \qquad \underline{U}_t \sim \mathcal{U}(\mathbb{S})$$

$\mathbb{S}$ be the d-dimensional unit sphere

$$linesearch_h(\underline{X}_t, \underline{U}_t) := \arg\min_{\beta} \left(h(\underline{X}_t + \beta \cdot \underline{U}_t) \right) \cdot \underline{U}_t$$

β is distance between X_t and X_{t+1}

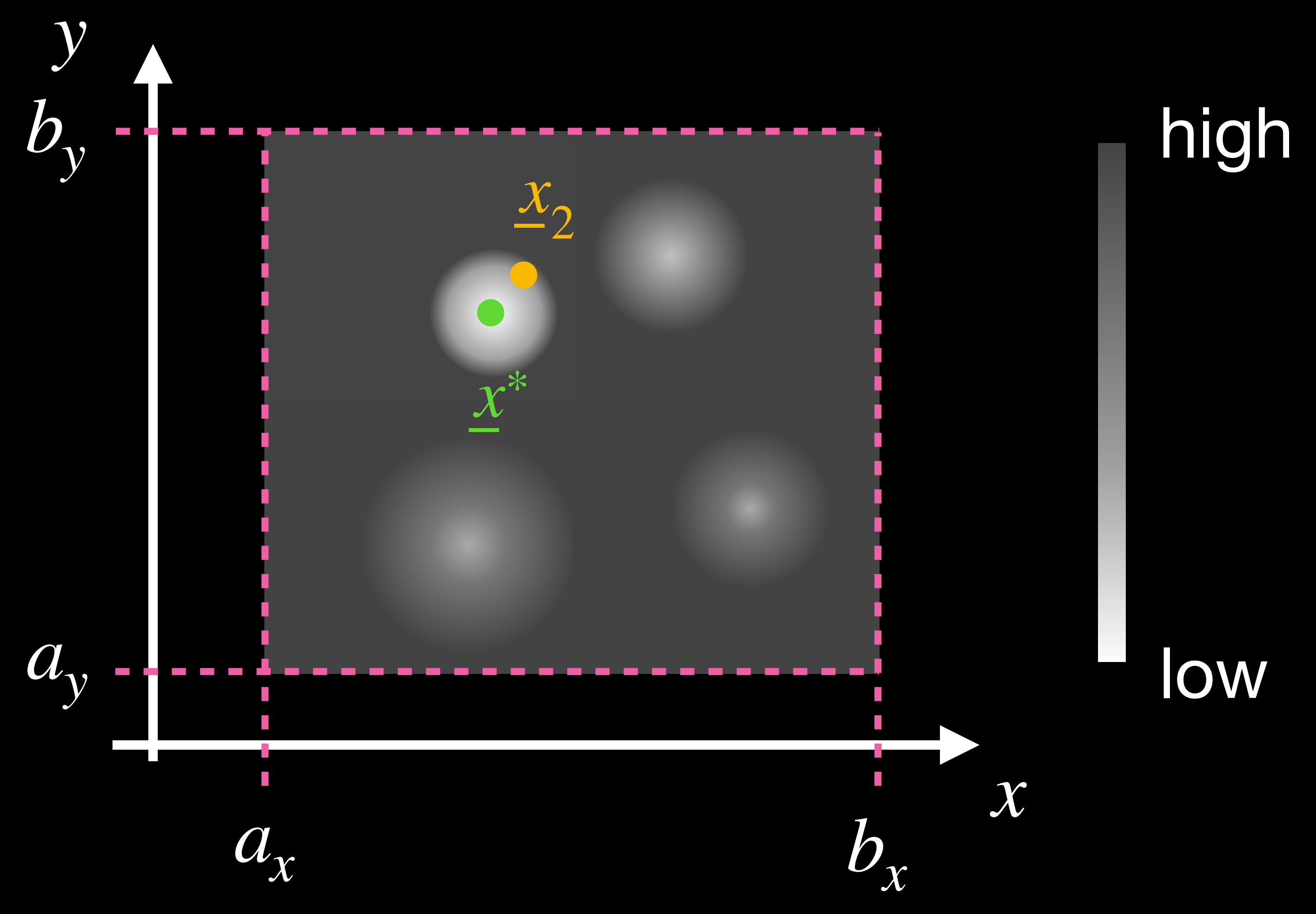
(finds the vector to the point with minimal value on the line that goes through $\underline{X}_t$ in the direction $\underline{U}_t$)

→ reduction to 1d-optimisation problem

properties:
- global convergence guarantee

algorithm 3: random pursuit

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$



$$\underline{X}_{t+1} = \underline{X}_t + linesearch(\underline{X}_t, \underline{U}_t) \qquad \underline{U}_t \sim \mathcal{U}(\mathbb{S})$$

$\mathbb{S}$ be the d-dimensional unit sphere

$$linesearch_h(\underline{X}_t, \underline{U}_t) := \underset{\beta}{\arg\min}\left(h(\underline{X}_t + \beta \cdot \underline{U}_t)\right) \cdot \underline{U}_t$$

β is distance between X_t and X_{t+1}

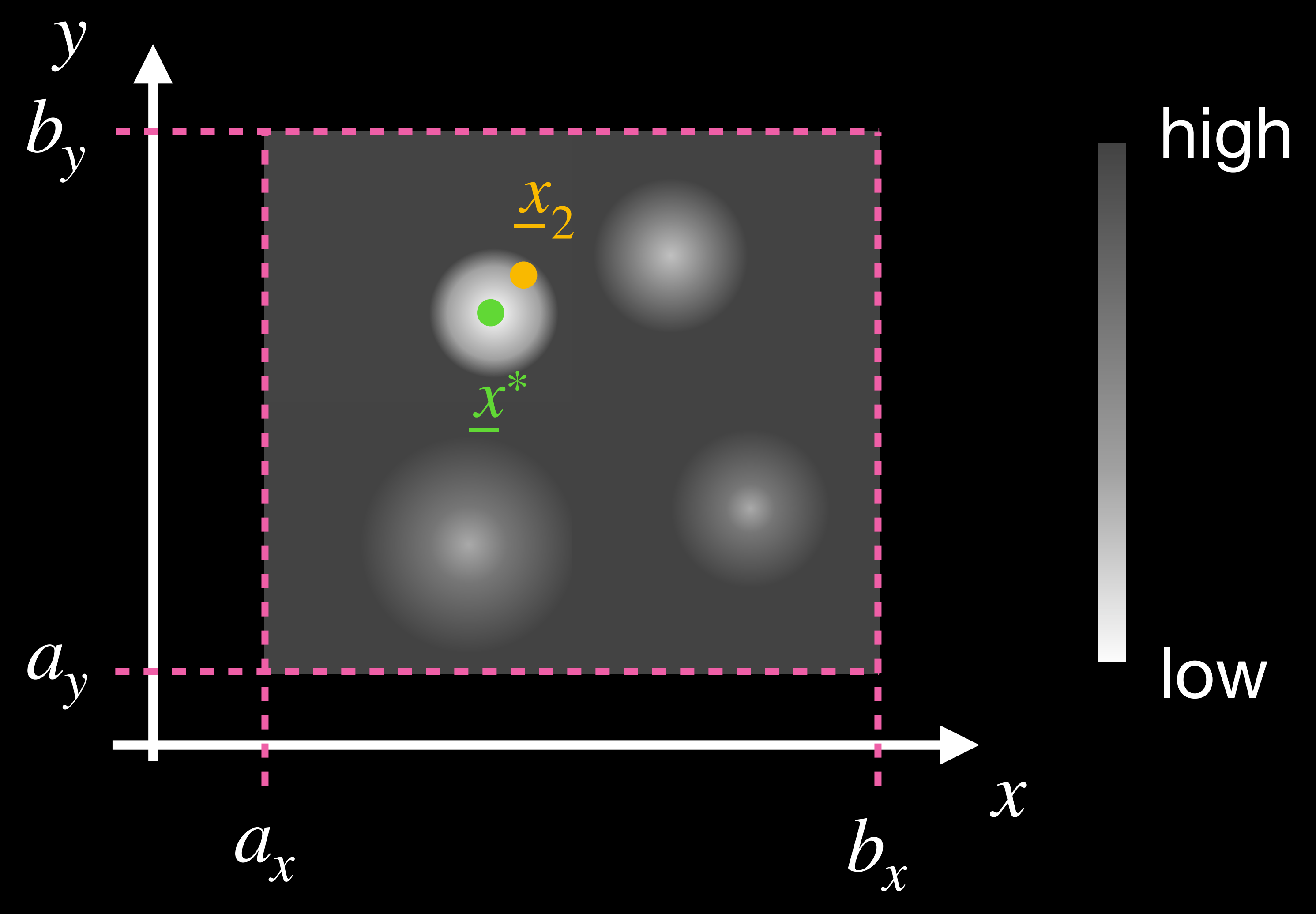
(finds the vector to the point with minimal value on the line that goes through $\underline{X}_t$ in the direction $\underline{U}_t$)

$\rightarrow$ reduction to 1d-optimisation problem

properties:
- global convergence guarantee
- converges fast

algorithm 3: random pursuit

$$\underline{x}^* := \arg\min_{\underline{x} \in \mathbb{S}} \left(h(\underline{x}) \right)$$



$$\underline{X}_{t+1} = \underline{X}_t + linesearch(\underline{X}_t, \underline{U}_t) \qquad \underline{U}_t \sim \mathcal{U}(\mathbb{S})$$

$\mathbb{S}$ be the d-dimensional unit sphere

$$linesearch_h(\underline{X}_t, \underline{U}_t) := \arg\min_{\beta} \left(h(\underline{X}_t + \beta \cdot \underline{U}_t) \right) \cdot \underline{U}_t$$

β is distance between X_t and X_{t+1}

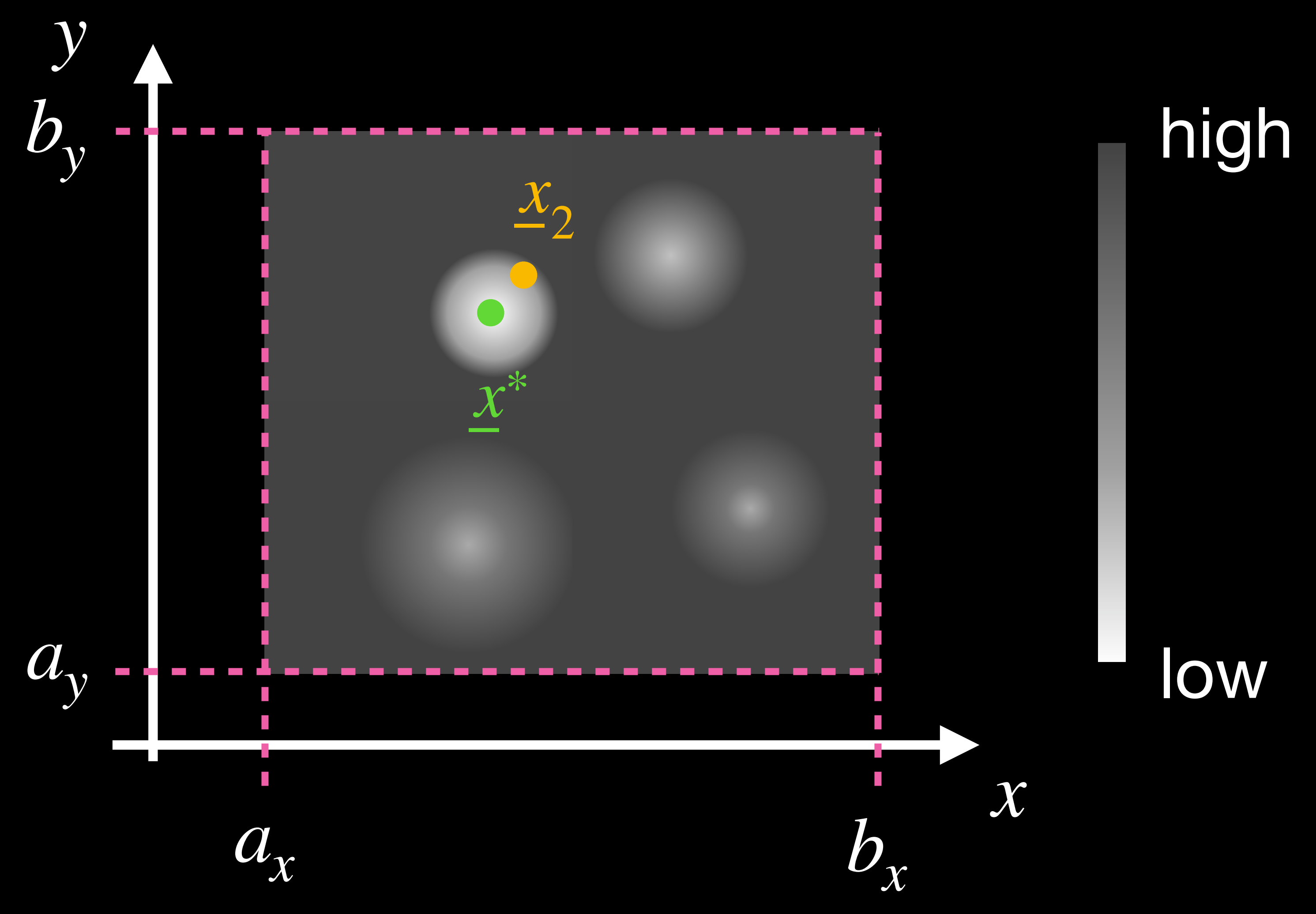
(finds the vector to the point with minimal value on the line that goes through $\underline{X}_t$ in the direction $\underline{U}_t$)

$\rightarrow$ reduction to 1d-optimisation problem

properties:
- global convergence guarantee
- converges fast
- needs uniform random numbers (on the unit sphere)

algorithm 3: random pursuit

$$\underline{x}^* := \arg\min_{\underline{x} \in \mathbb{S}} \left(h(\underline{x}) \right)$$



$$\underline{X}_{t+1} = \underline{X}_t + linesearch(\underline{X}_t, \underline{U}_t) \qquad \underline{U}_t \sim \mathcal{U}(\mathbb{S})$$

$\mathbb{S}$ be the d-dimensional unit sphere

$$linesearch_h(\underline{X}_t, \underline{U}_t) := \arg\min_{\beta} \left(h(\underline{X}_t + \beta \cdot \underline{U}_t) \right) \cdot \underline{U}_t$$

β is distance between X_t and X_{t+1}

(finds the vector to the point with minimal value on the line that goes through $\underline{X}_t$ in the direction $\underline{U}_t$)

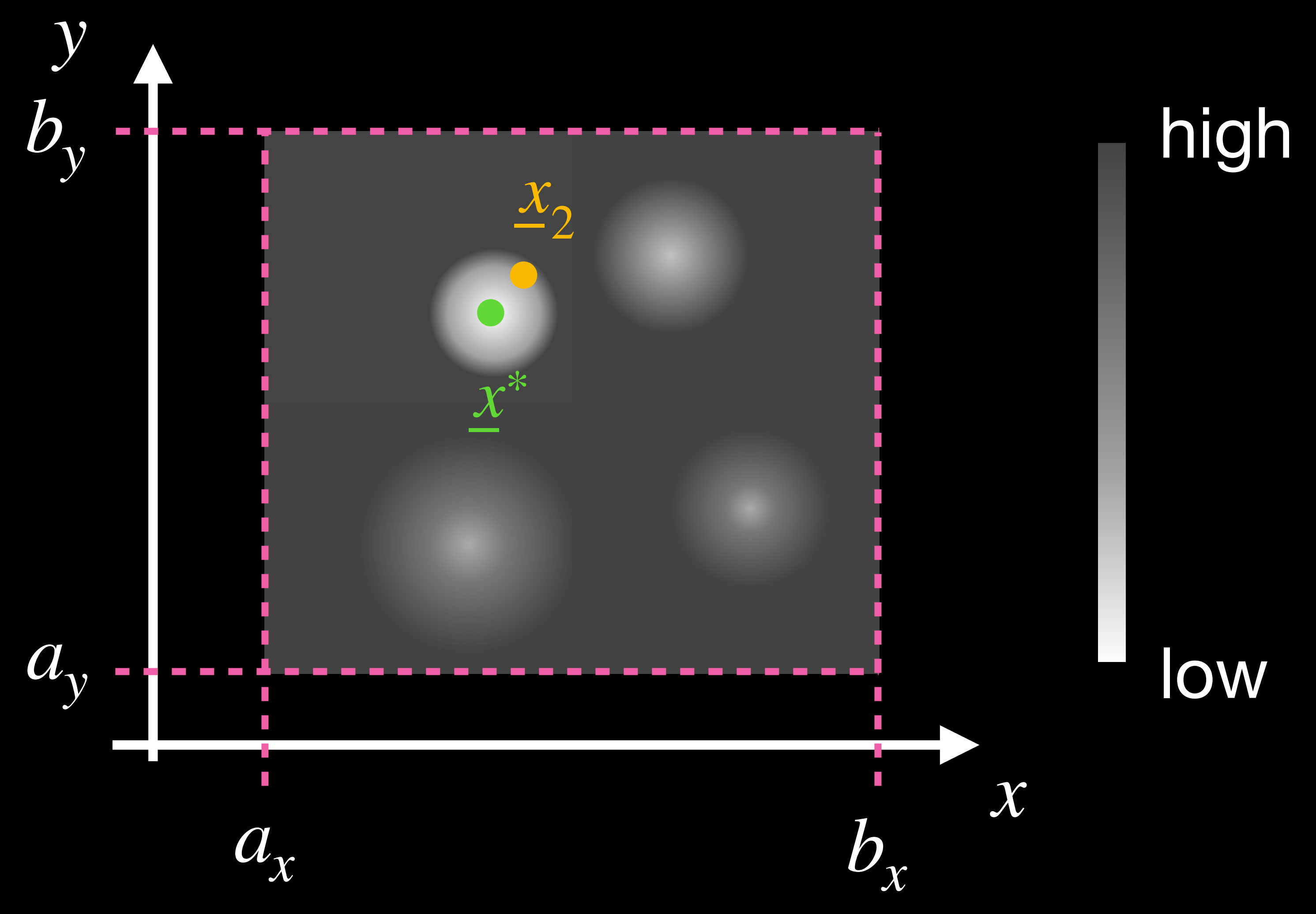
→ reduction to 1d-optimisation problem

properties:
- global convergence guarantee
- converges fast
- needs uniform random numbers (on the unit sphere)

useful if:

algorithm 3: random pursuit

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \big(h(\underline{x})\big)$$



$$\underline{X}_{t+1} = \underline{X}_t + linesearch(\underline{X}_t, \underline{U}_t) \qquad \underline{U}_t \sim \mathcal{U}(\mathbb{S})$$

$\mathbb{S}$ be the d-dimensional unit sphere

$$linesearch_h(\underline{X}_t, \underline{U}_t) := \arg\min_{\beta} \Big(h(\underline{X}_t + \beta \cdot \underline{U}_t)\Big) \cdot \underline{U}_t$$

β is distance between X_t and X_{t+1}

(finds the vector to the point with minimal value on the line that goes through $\underline{X}_t$ in the direction $\underline{U}_t$)

$\rightarrow$ reduction to 1d-optimisation problem

properties:
- global convergence guarantee
- converges fast
- needs uniform random numbers (on the unit sphere)

useful if:
- $\mathbb{S}$ is high-dimensional and bounded

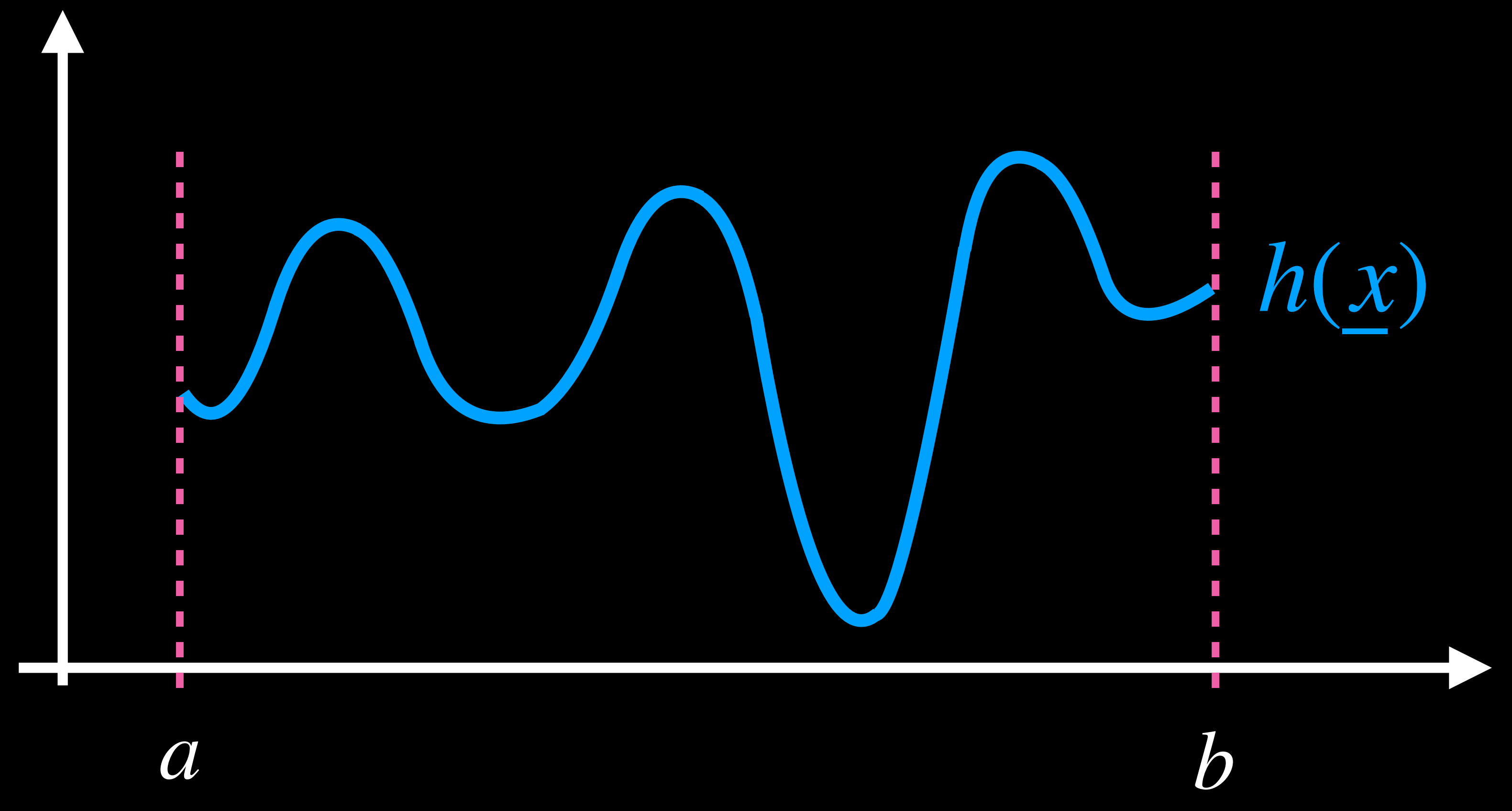
algorithm 4: simulated annealing

algorithm 4: simulated annealing

$$\underline{x}^* := \arg\min_{\underline{x} \in \mathbb{S}} \left(h(\underline{x}) \right)$$

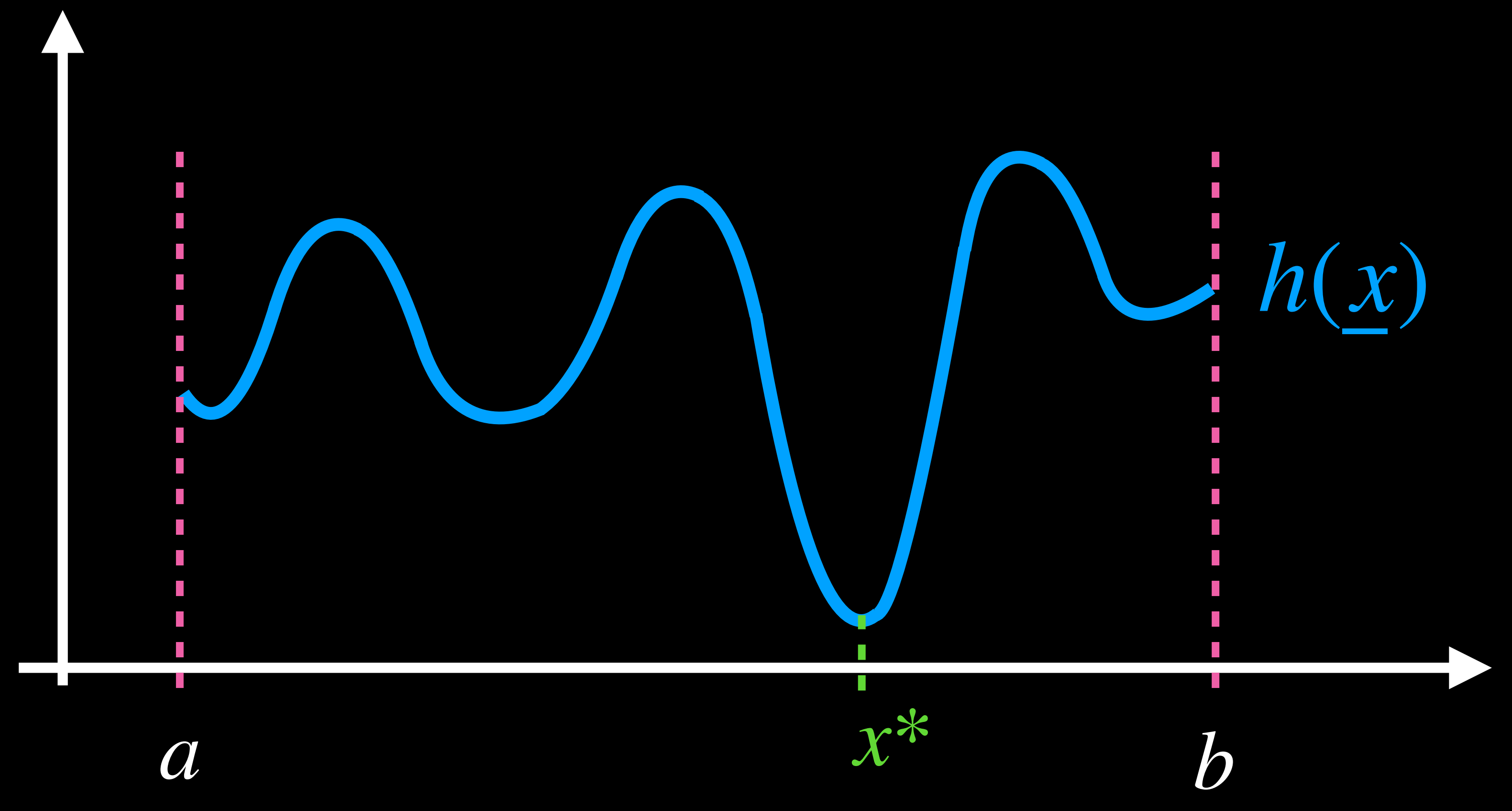
algorithm 4: simulated annealing

$$\underline{x}^* := \underset{\underline{x} \in \mathbb{S}}{\arg\min} \left(h(\underline{x}) \right)$$



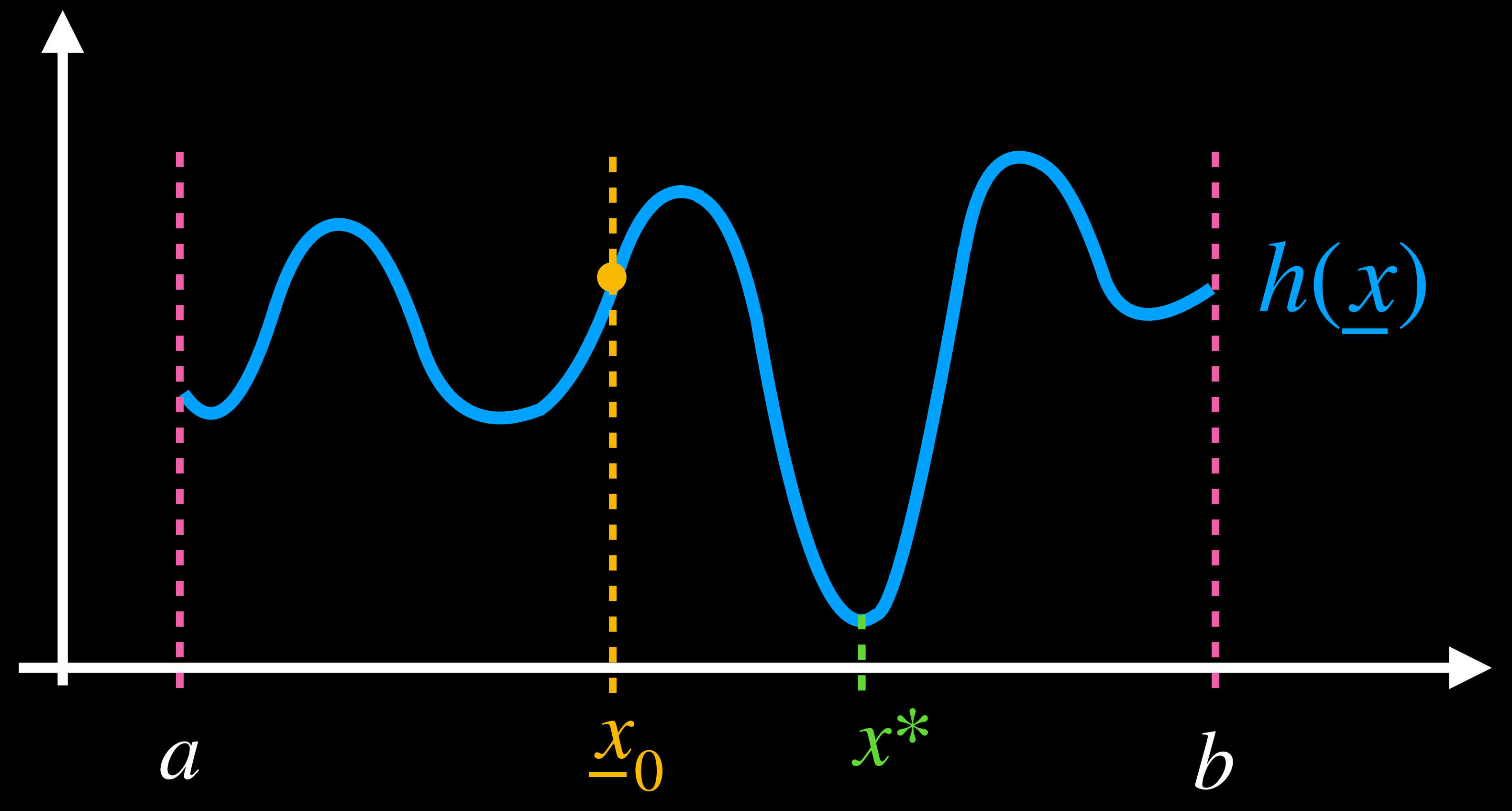
algorithm 4: simulated annealing

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$



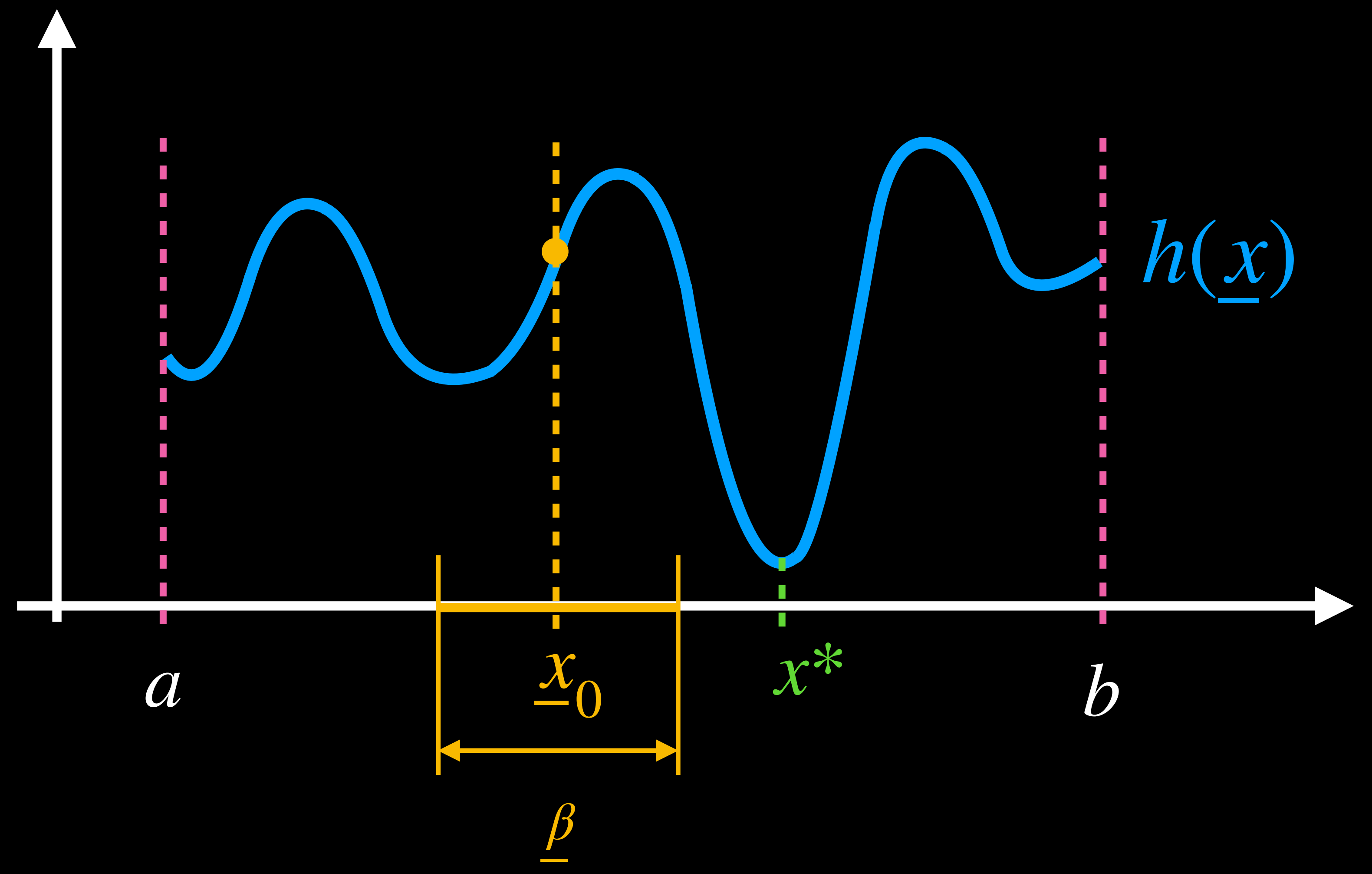
algorithm 4: simulated annealing

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$



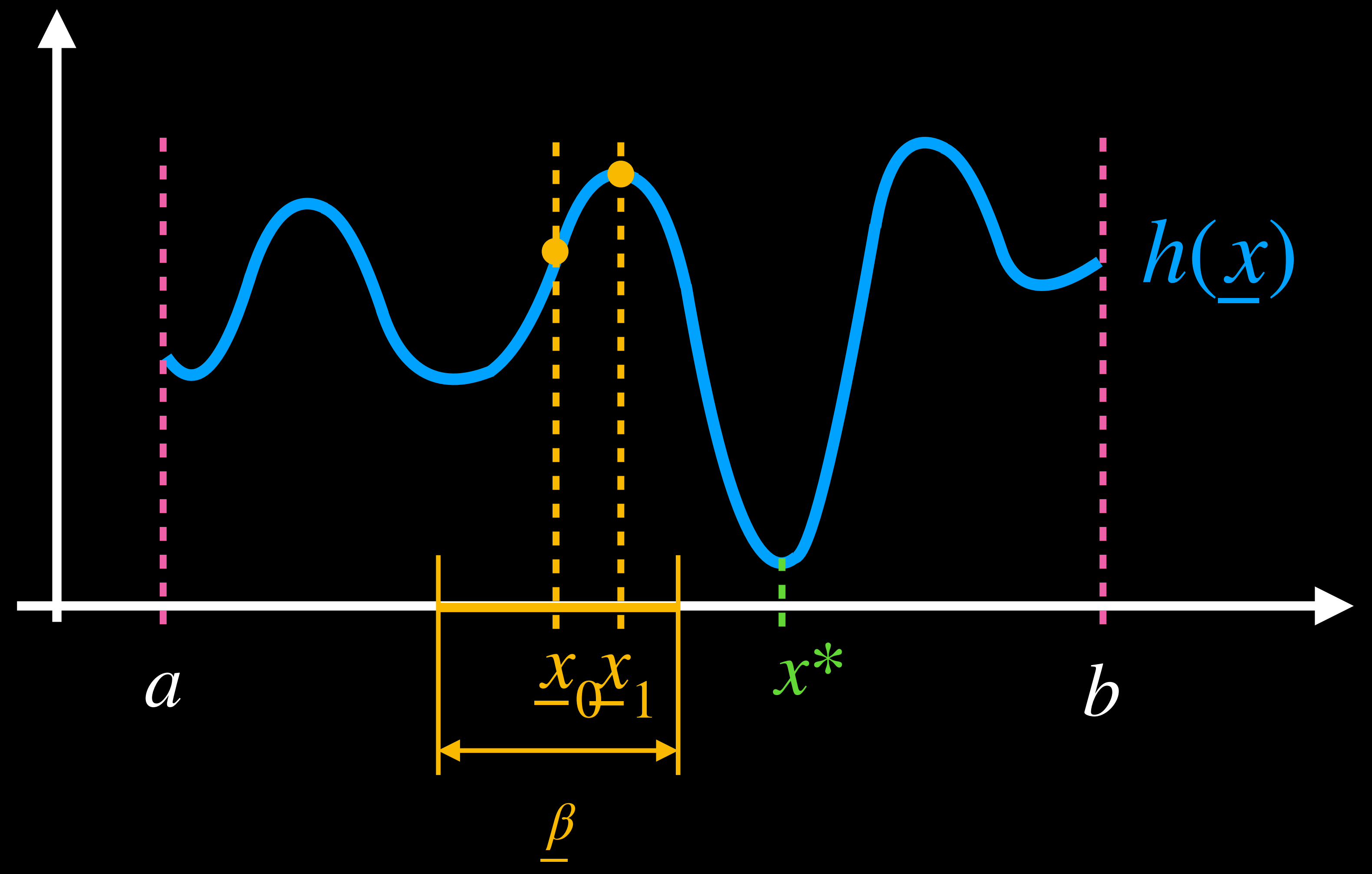
algorithm 4: simulated annealing

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$



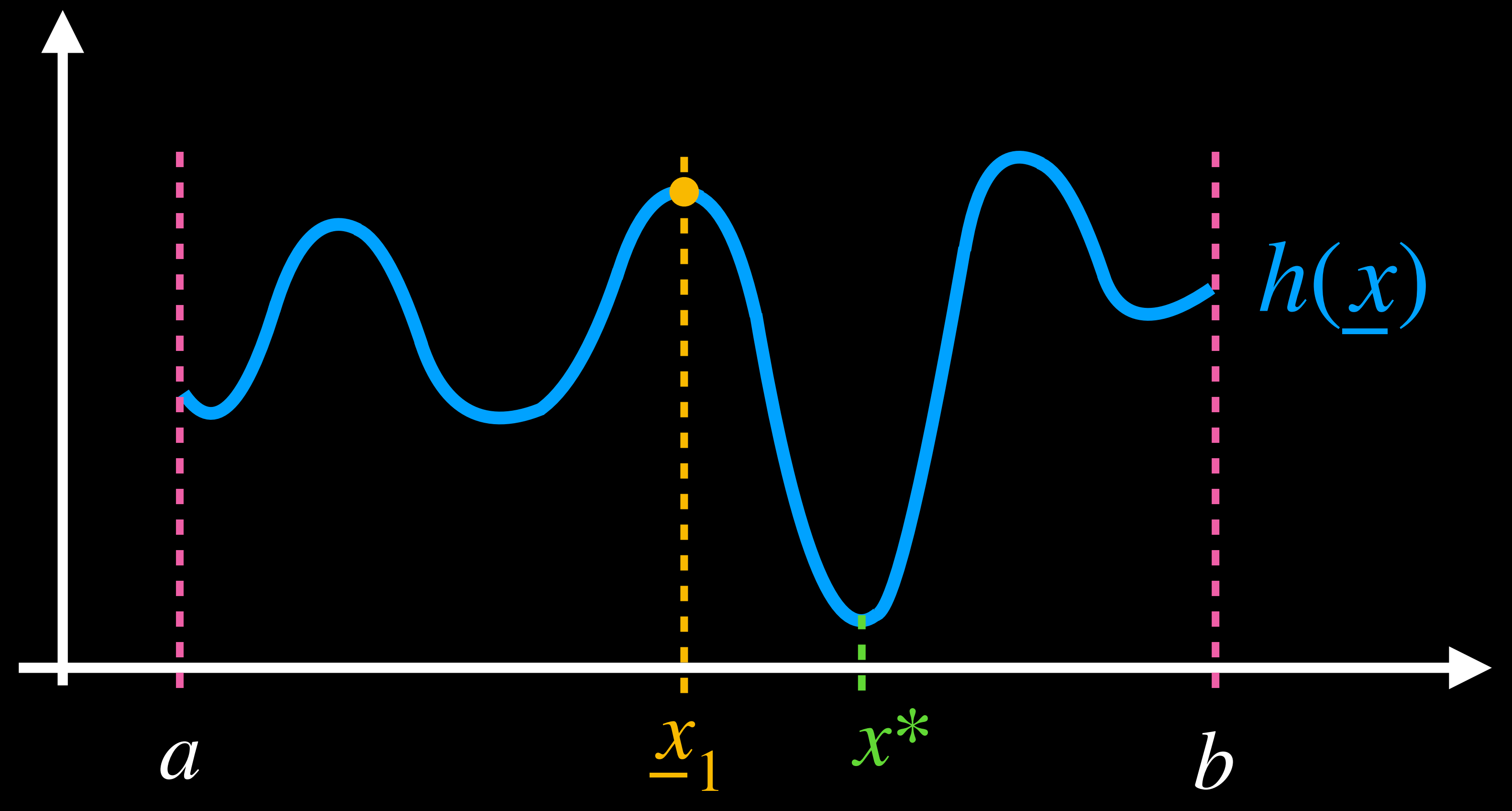
algorithm 4: simulated annealing

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$



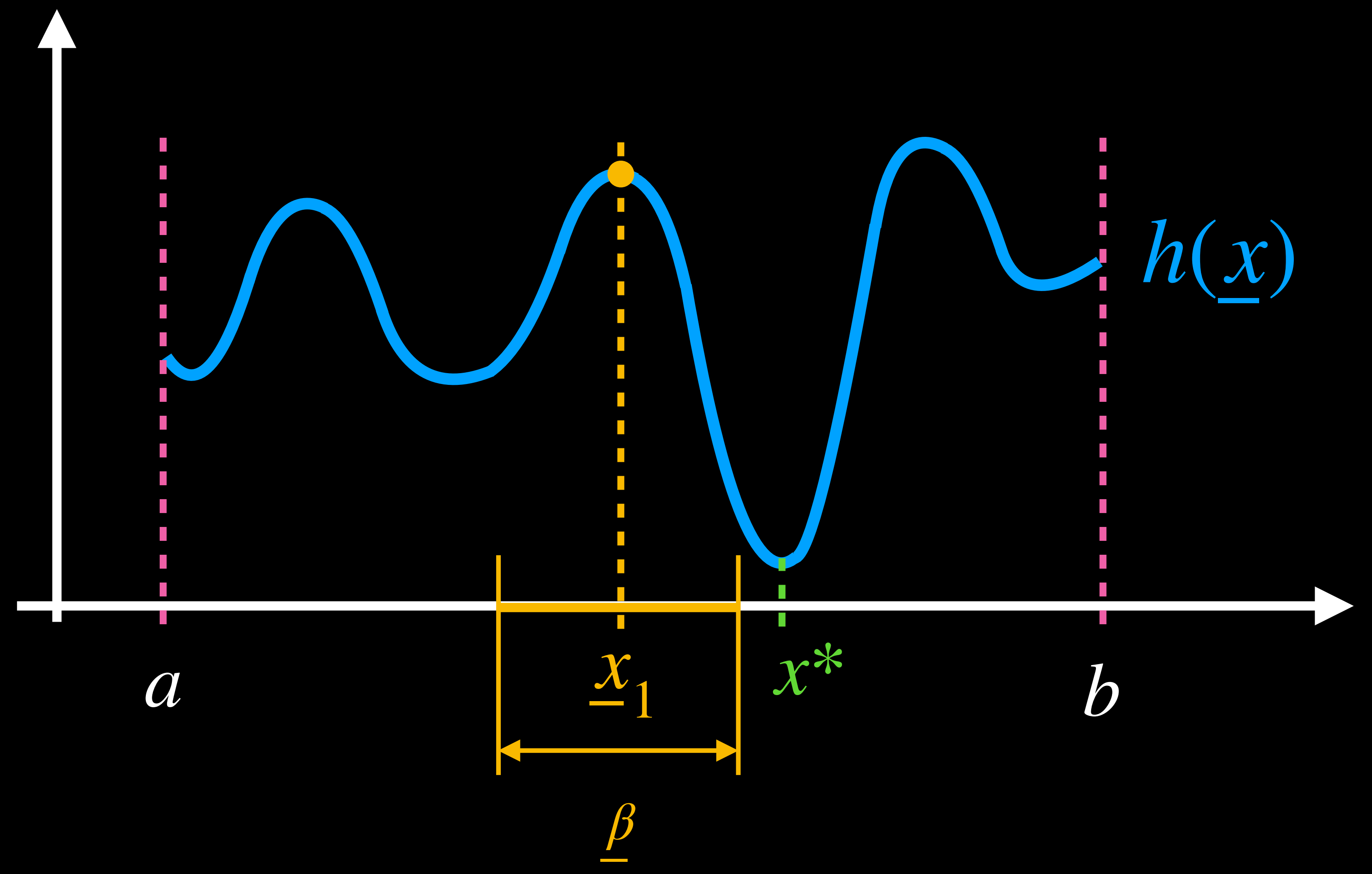
algorithm 4: simulated annealing

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$



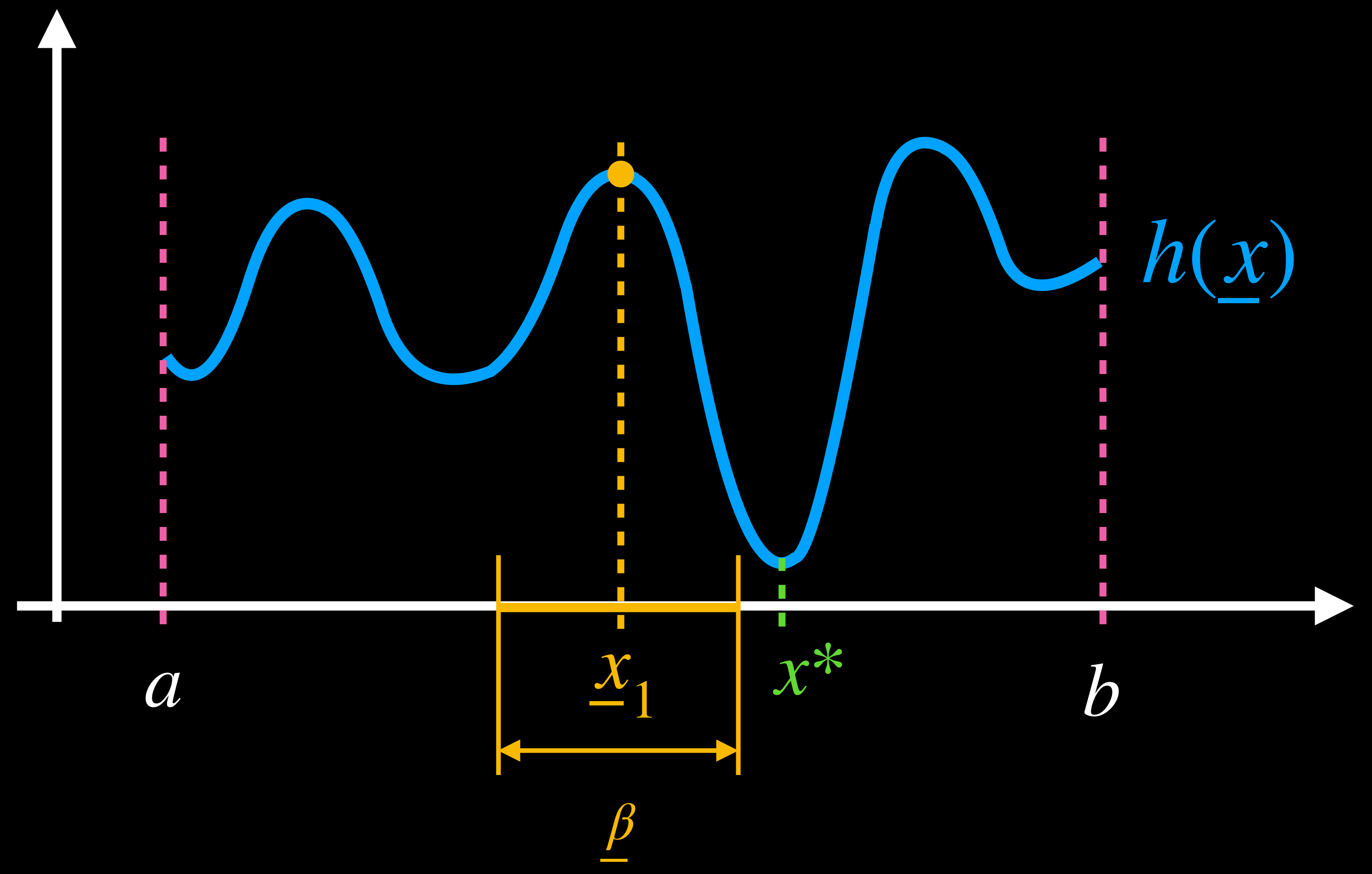
algorithm 4: simulated annealing

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$



algorithm 4: simulated annealing

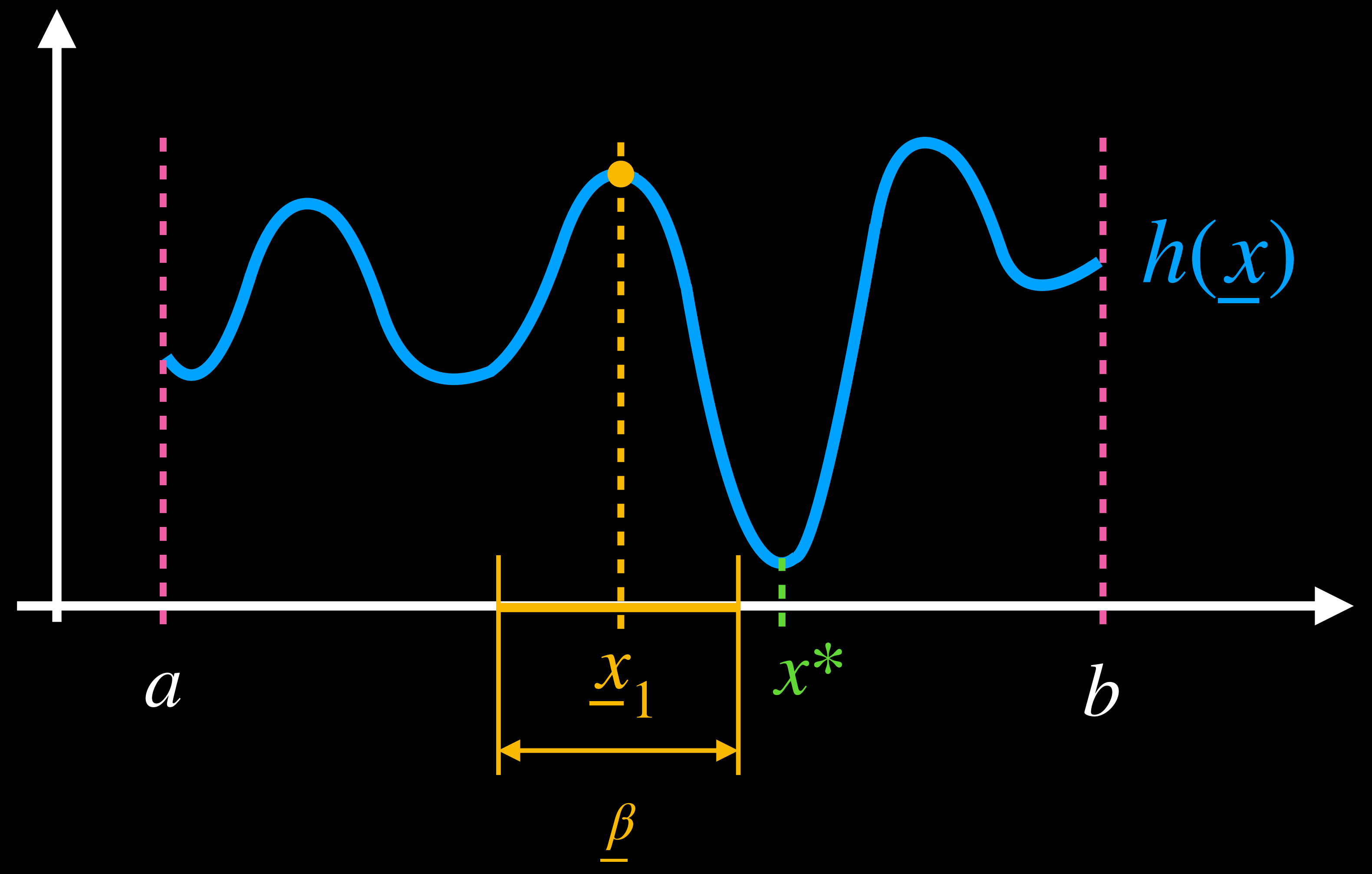
$$\underline{x}^* := \arg\min_{\underline{x} \in \mathbb{S}} \left(h(\underline{x}) \right)$$



$\underline{X}_{t+1}$

algorithm 4: simulated annealing

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

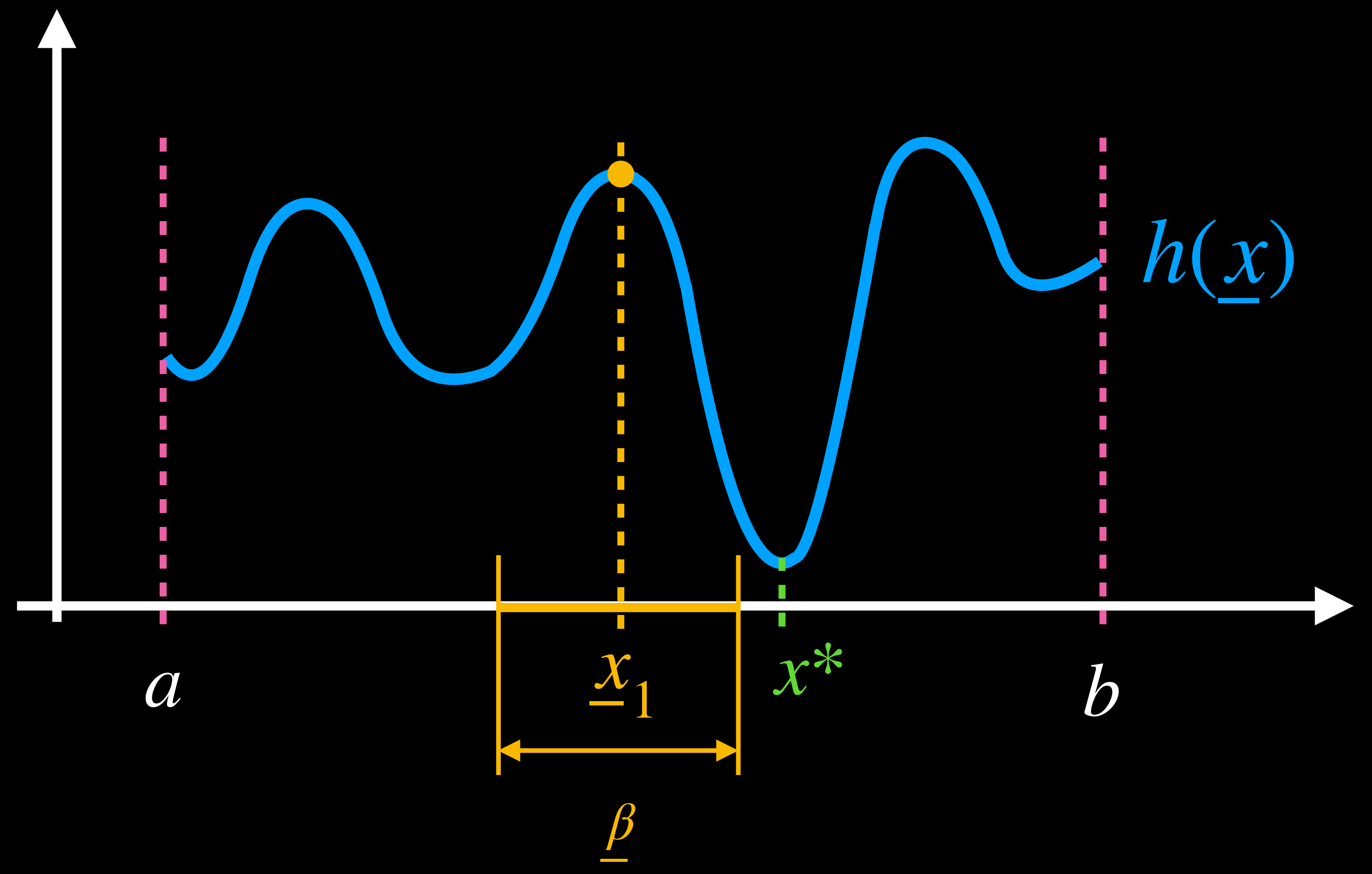


$\underline{X}_{t+1}$

$U_t \sim \mathcal{U}(0,1)$

algorithm 4: simulated annealing

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$



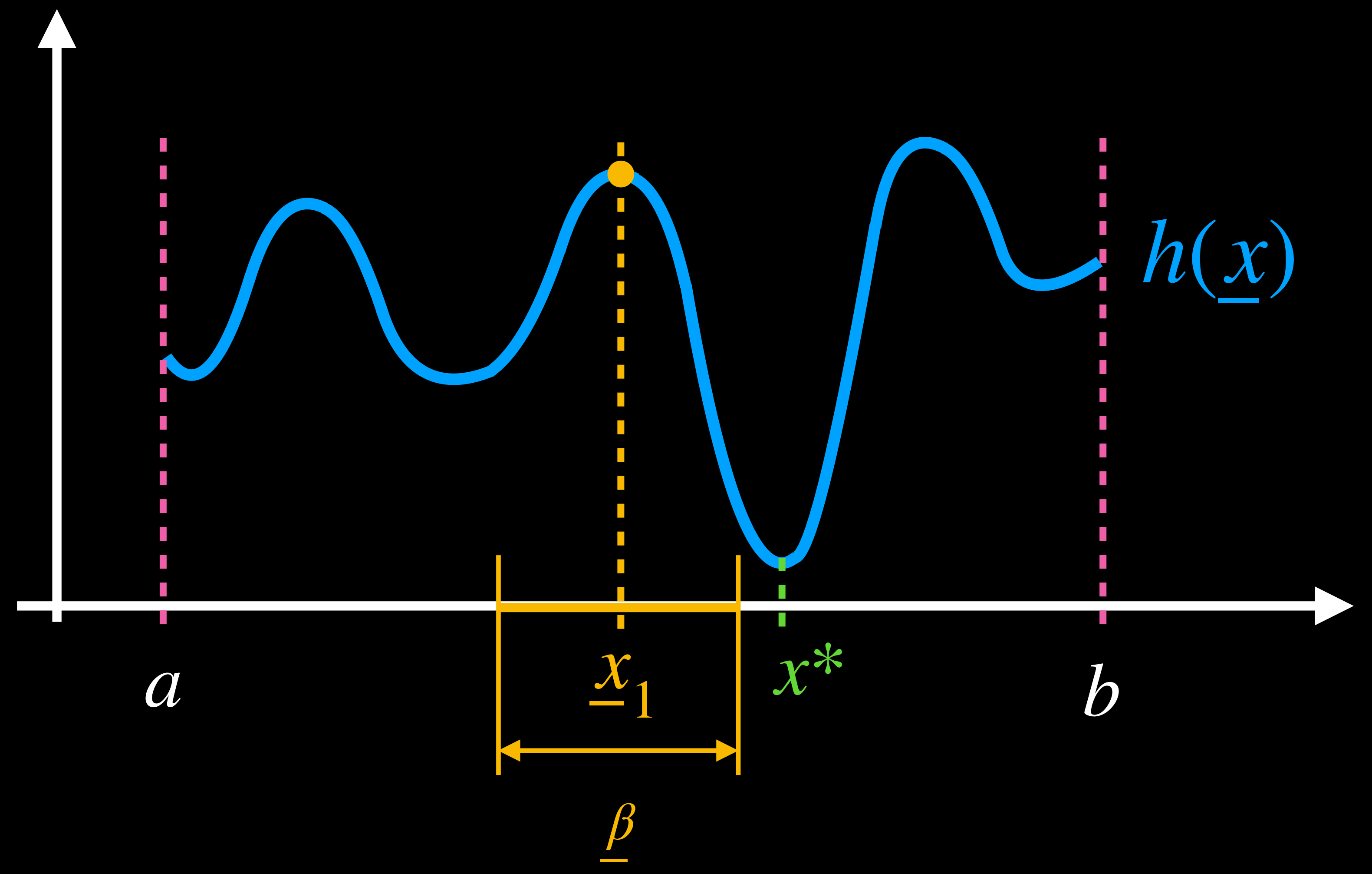
$$U_t \sim \mathcal{U}(0,1)$$

$$\underline{U}_t^\beta \sim \mathcal{U}(-\tfrac{1}{2}\underline{\beta}, \tfrac{1}{2}\underline{\beta})$$

$$\underline{X}_{t+1}$$

algorithm 4: simulated annealing

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$



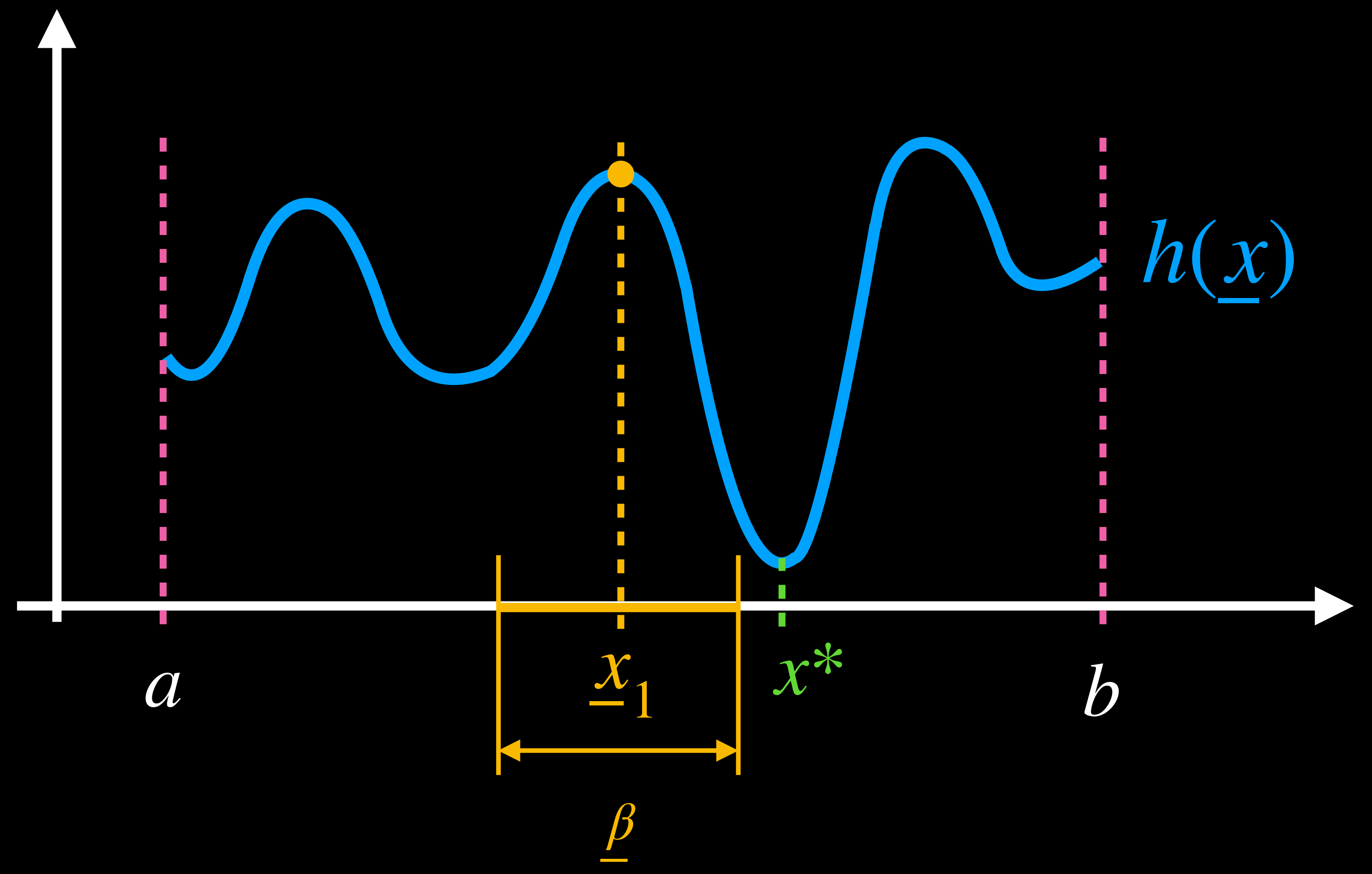
$$\underline{X}_{t+1} := \begin{cases} \underline{X}_t + \underline{U}_t^\beta & \text{if } U_t < e^{\frac{h(\underline{X}_t) - h(\underline{X}_t + \underline{U}_t^\beta)}{T_t}} \\ \underline{X}_t & \text{else} \end{cases}$$

$$U_t \sim \mathscr{U}(0,1)$$

$$\underline{U}_t^\beta \sim \mathscr{U}(-\tfrac{1}{2}\underline{\beta}, \tfrac{1}{2}\underline{\beta})$$

algorithm 4: simulated annealing

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$



$$\underline{X}_{t+1} := \begin{cases} \underline{X}_t + \underline{U}_t^\beta & \text{if } U_t < e^{\frac{h(\underline{X}_t) - h(\underline{X}_t + \underline{U}_t^\beta)}{T_t}} \\ \underline{X}_t & \text{else} \end{cases}$$

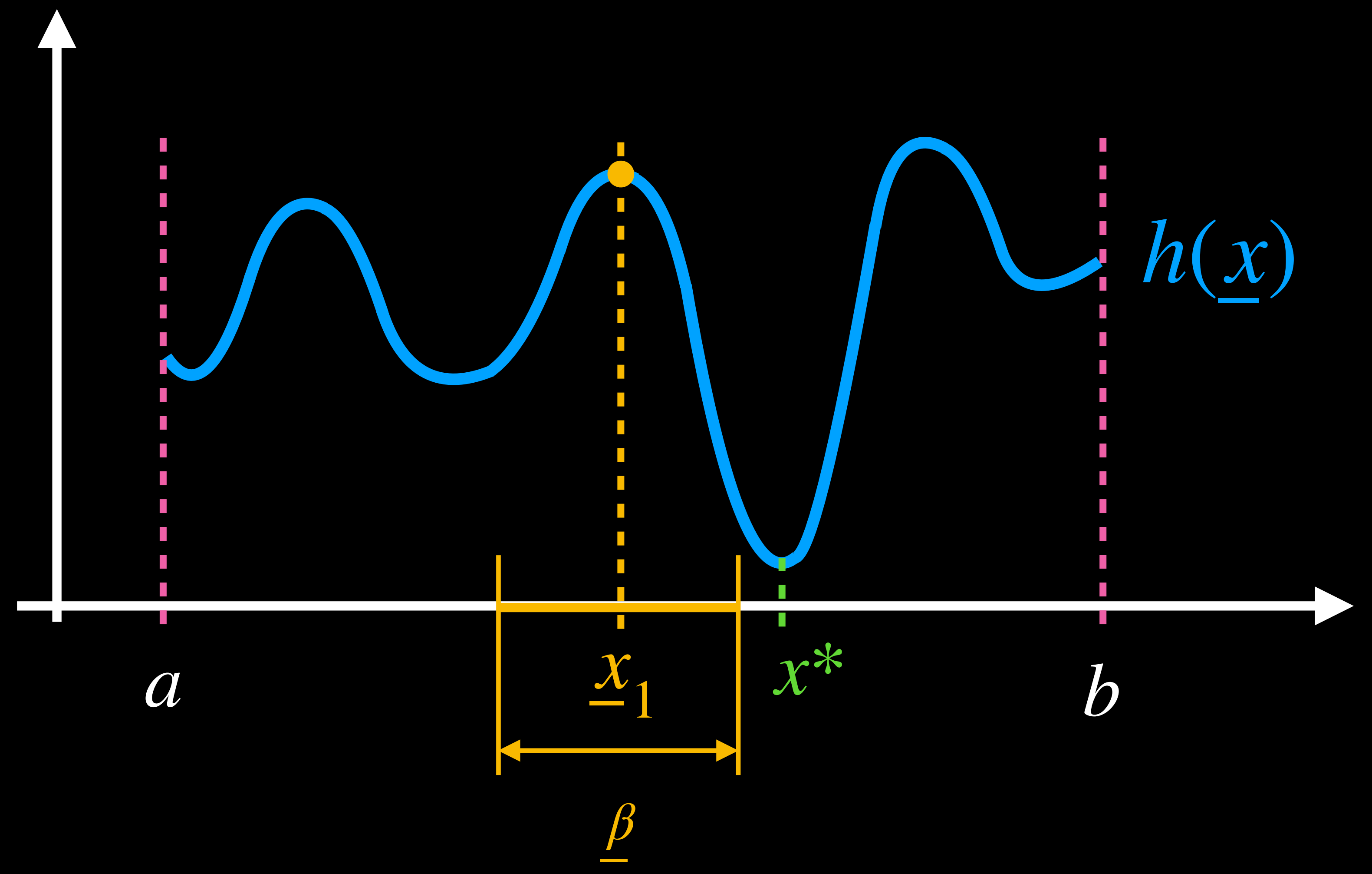
$$U_t \sim \mathcal{U}(0,1)$$

$$\underline{U}_t^\beta \sim \mathcal{U}(-\tfrac{1}{2}\underline{\beta}, \tfrac{1}{2}\underline{\beta})$$

$T_t > 0$ is the "temperature" (parameter for the chance to accept worse points)

algorithm 4: simulated annealing

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\,\big(h(\underline{x})\big)$$



$$\underline{X}_{t+1} := \begin{cases} \underline{X}_t + \underline{U}_t^\beta & \text{if } U_t < e^{\frac{h(\underline{X}_t) - h(\underline{X}_t + \underline{U}_t^\beta)}{T_t}} \\ \underline{X}_t & \text{else} \end{cases}$$

$$U_t \sim \mathcal{U}(0,1)$$

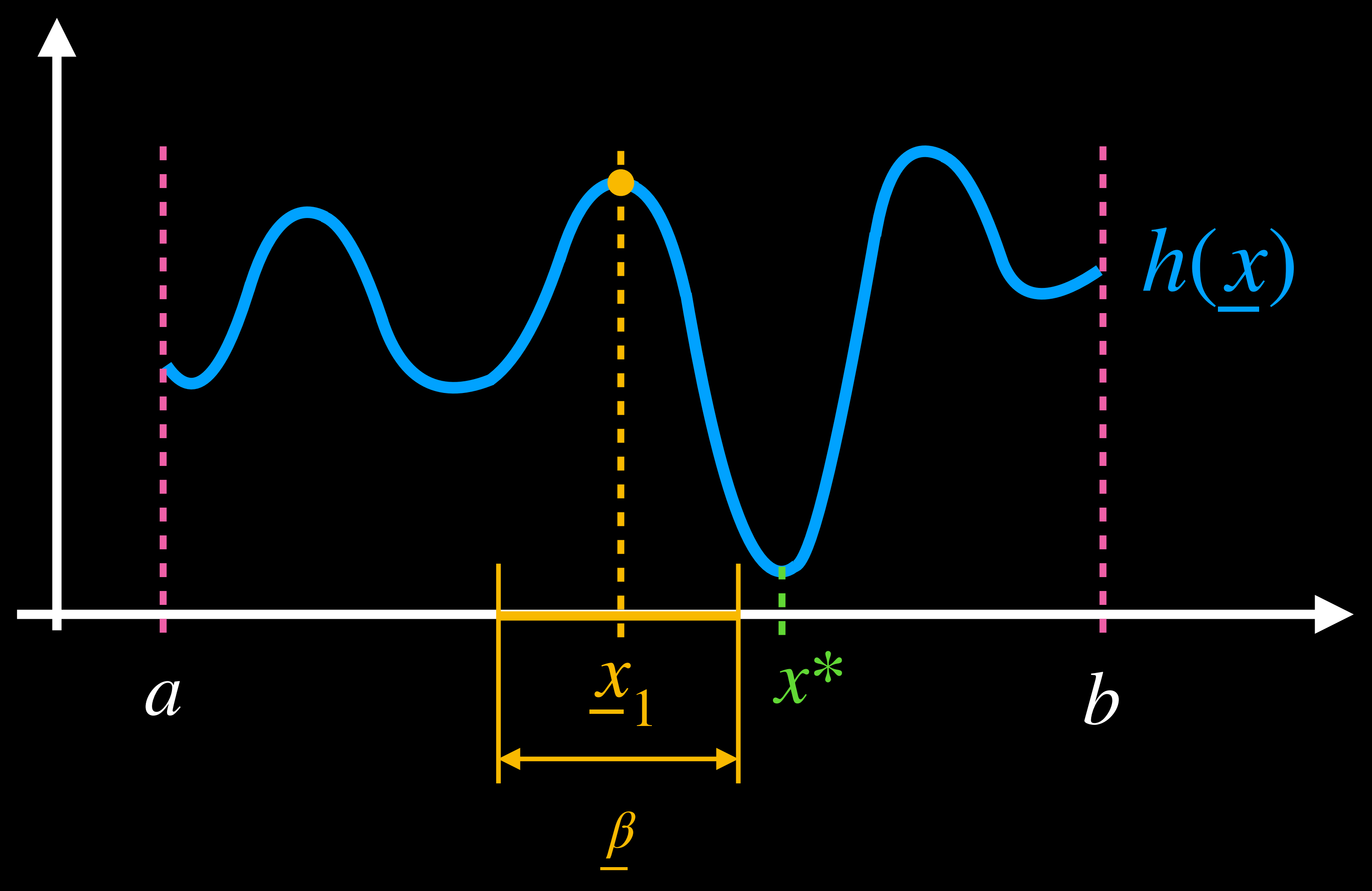
$$\underline{U}_t^\beta \sim \mathcal{U}(-\tfrac{1}{2}\underline{\beta}, \tfrac{1}{2}\underline{\beta})$$

$T_t > 0$ is the "temperature" (parameter for the chance to accept worse points)

(example for a cooling scheme: $T_{t+1} := 0.95\, T_t$)

algorithm 4: simulated annealing

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$



$$\underline{X}_{t+1} := \begin{cases} \underline{X}_t + \underline{U}_t^\beta & \text{if } U_t < e^{\frac{h(\underline{X}_t) - h(\underline{X}_t + \underline{U}_t^\beta)}{T_t}} \\ \underline{X}_t & \text{else} \end{cases}$$

$$U_t \sim \mathcal{U}(0,1)$$

$$\underline{U}_t^\beta \sim \mathcal{U}(-\tfrac{1}{2}\underline{\beta}, \tfrac{1}{2}\underline{\beta})$$

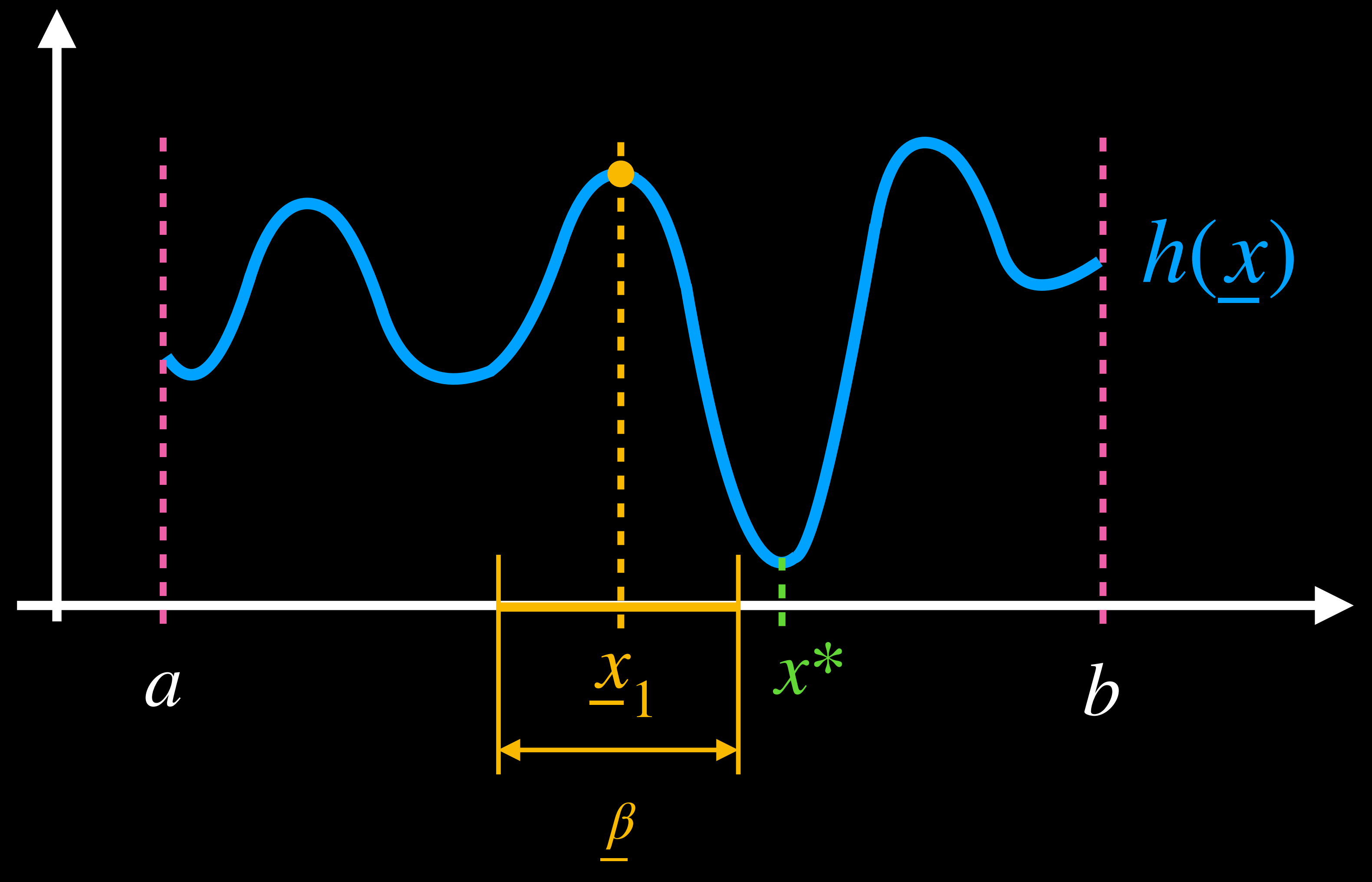
$T_t > 0$ is the "temperature" (parameter for the chance to accept worse points)

(example for a cooling scheme: $T_{t+1} := 0.95\, T_t$)

properties:

algorithm 4: simulated annealing

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$



$$\underline{X}_{t+1} := \begin{cases} \underline{X}_t + \underline{U}_t^\beta & \text{if } U_t < e^{\frac{h(\underline{X}_t) - h(\underline{X}_t + \underline{U}_t^\beta)}{T_t}} \\ \underline{X}_t & \text{else} \end{cases}$$

$$U_t \sim \mathcal{U}(0,1)$$

$$\underline{U}_t^\beta \sim \mathcal{U}(-\tfrac{1}{2}\underline{\beta}, \tfrac{1}{2}\underline{\beta})$$

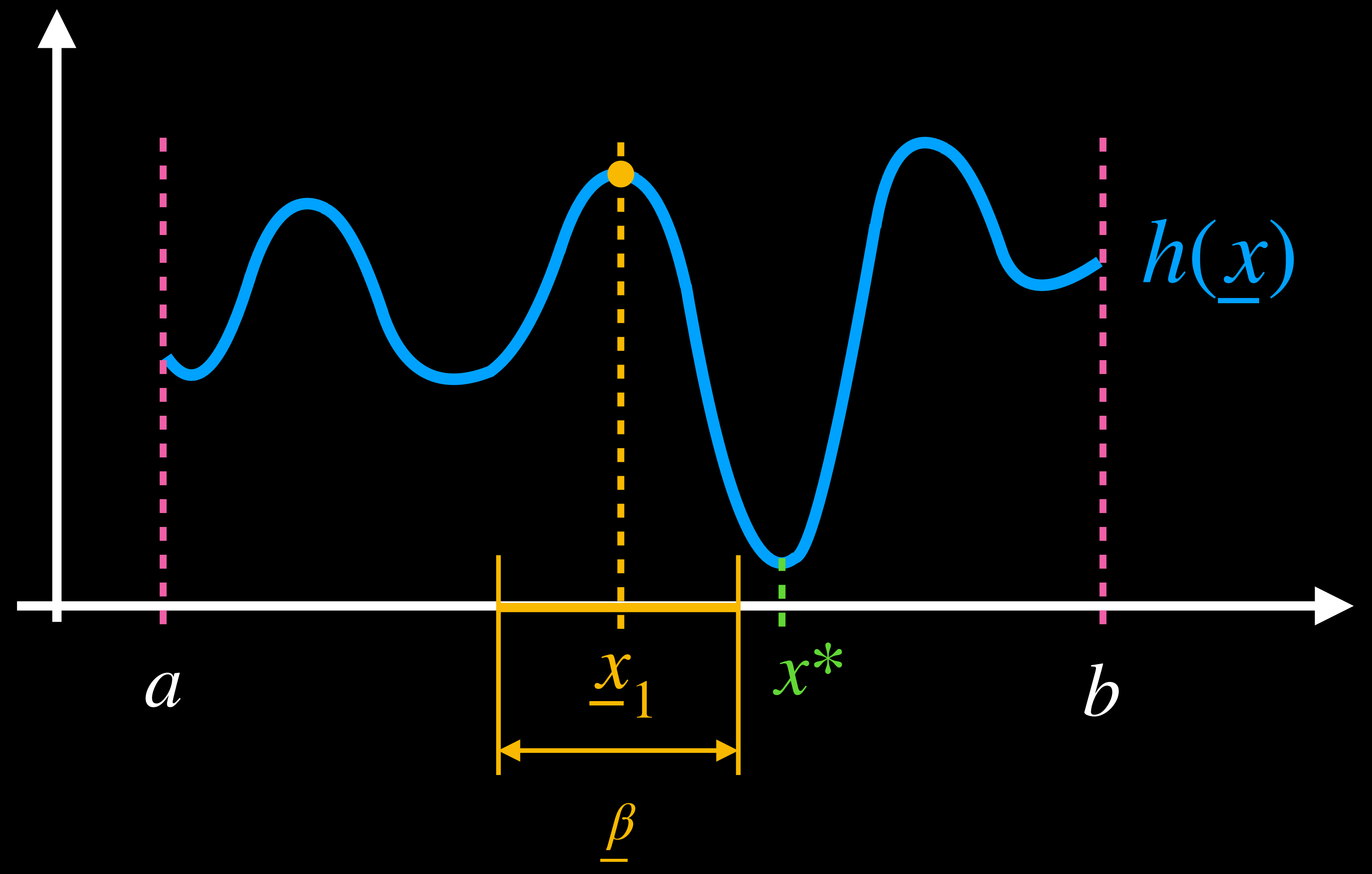
$T_t > 0$ is the "temperature" (parameter for the chance to accept worse points)

(example for a cooling scheme: $T_{t+1} := 0.95\, T_t$)

properties:

- simple

algorithm 4: simulated annealing



$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

$$\underline{X}_{t+1} := \begin{cases} \underline{X}_t + \underline{U}_t^{\beta} & \text{if } U_t < e^{\frac{h(\underline{X}_t) - h(\underline{X}_t + \underline{U}_t^{\beta})}{T_t}} \\ \underline{X}_t & \text{else} \end{cases}$$

$$U_t \sim \mathcal{U}(0,1)$$

$$\underline{U}_t^{\beta} \sim \mathcal{U}(-\tfrac{1}{2}\underline{\beta}, \tfrac{1}{2}\underline{\beta})$$

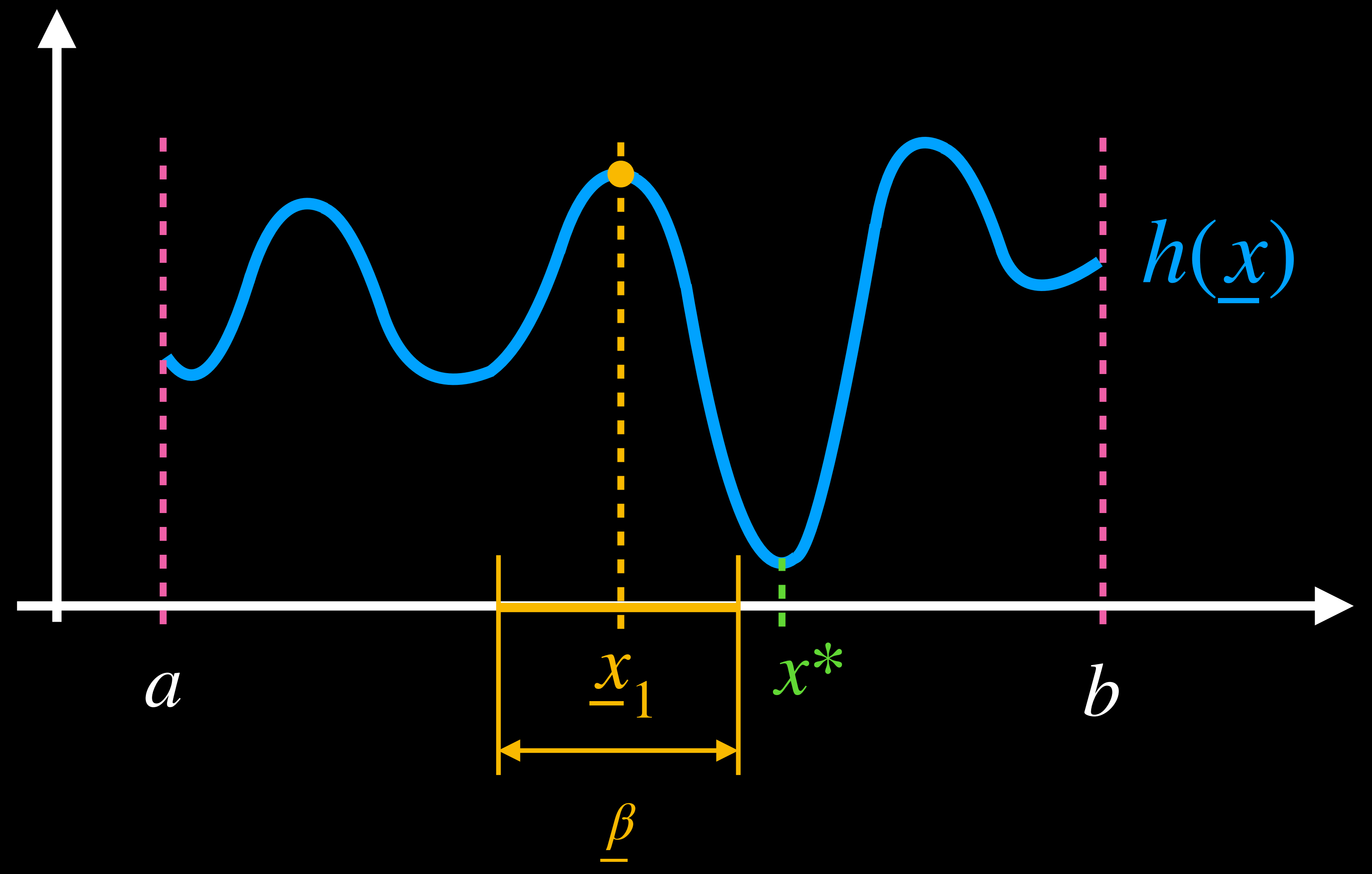
$T_t > 0$ is the "temperature" (parameter for the chance to accept worse points)

(example for a cooling scheme: $T_{t+1} := 0.95\, T_t$)

properties:
- simple
- convergence to a local minimum

algorithm 4: simulated annealing

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$



$$\underline{X}_{t+1} := \begin{cases} \underline{X}_t + \underline{U}_t^\beta & \text{if } U_t < e^{\frac{h(\underline{X}_t) - h(\underline{X}_t + \underline{U}_t^\beta)}{T_t}} \\ \underline{X}_t & \text{else} \end{cases}$$

$$U_t \sim \mathcal{U}(0,1)$$

$$\underline{U}_t^\beta \sim \mathcal{U}(-\tfrac{1}{2}\underline{\beta}, \tfrac{1}{2}\underline{\beta})$$

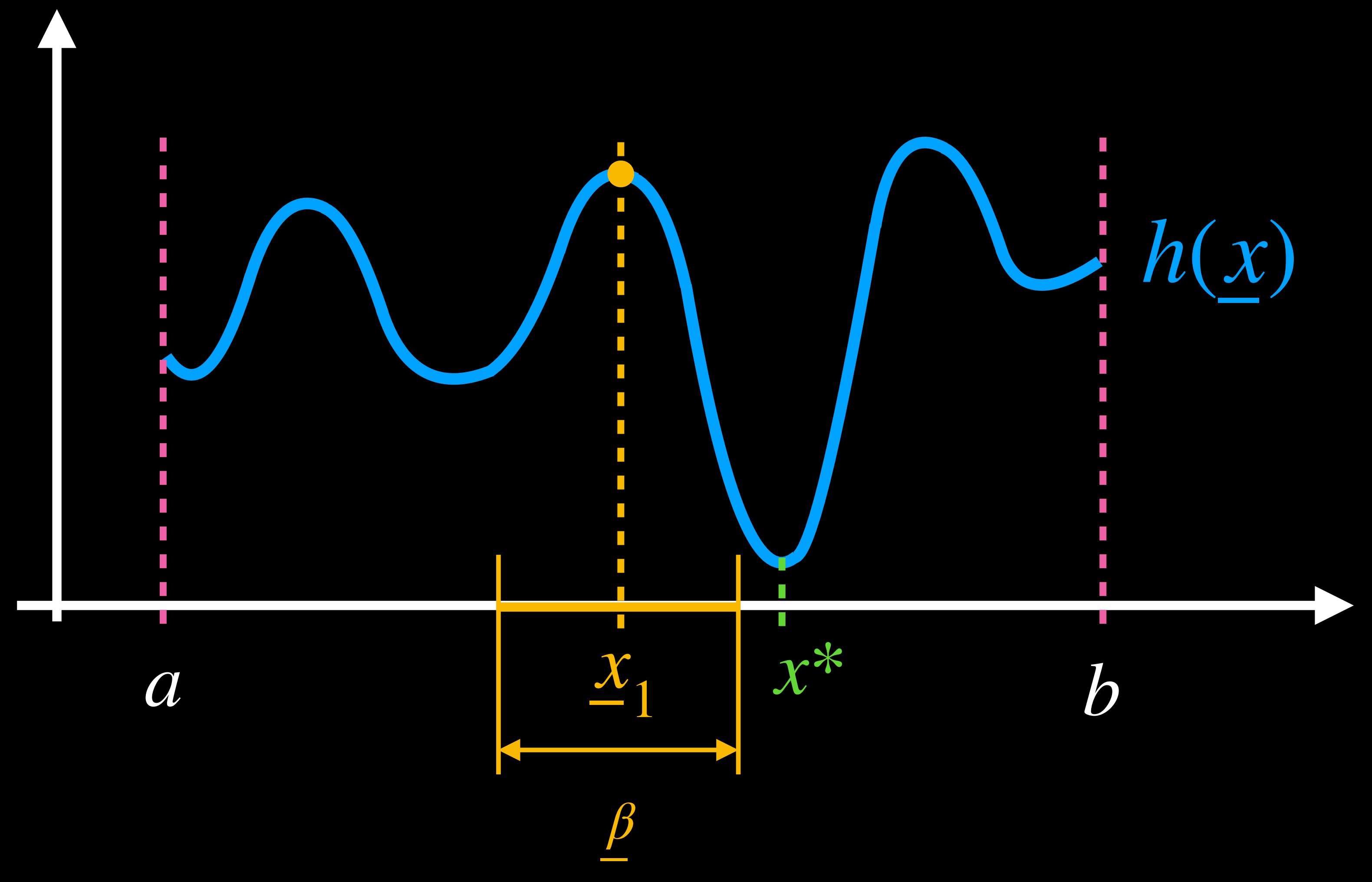
$T_t > 0$ is the "temperature" (parameter for the chance to accept worse points)

(example for a cooling scheme: $T_{t+1} := 0.95\, T_t$)

properties:
- simple
- convergence to a local minimum
- global convergence for the "right" cooling scheme

algorithm 4: simulated annealing

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$



$$\underline{X}_{t+1} := \begin{cases} \underline{X}_t + \underline{U}_t^\beta & \text{if } U_t < e^{\frac{h(\underline{X}_t) - h(\underline{X}_t + \underline{U}_t^\beta)}{T_t}} \\ \underline{X}_t & \text{else} \end{cases}$$

$$U_t \sim \mathcal{U}(0,1)$$

$$\underline{U}_t^\beta \sim \mathcal{U}(-\tfrac{1}{2}\underline{\beta}, \tfrac{1}{2}\underline{\beta})$$

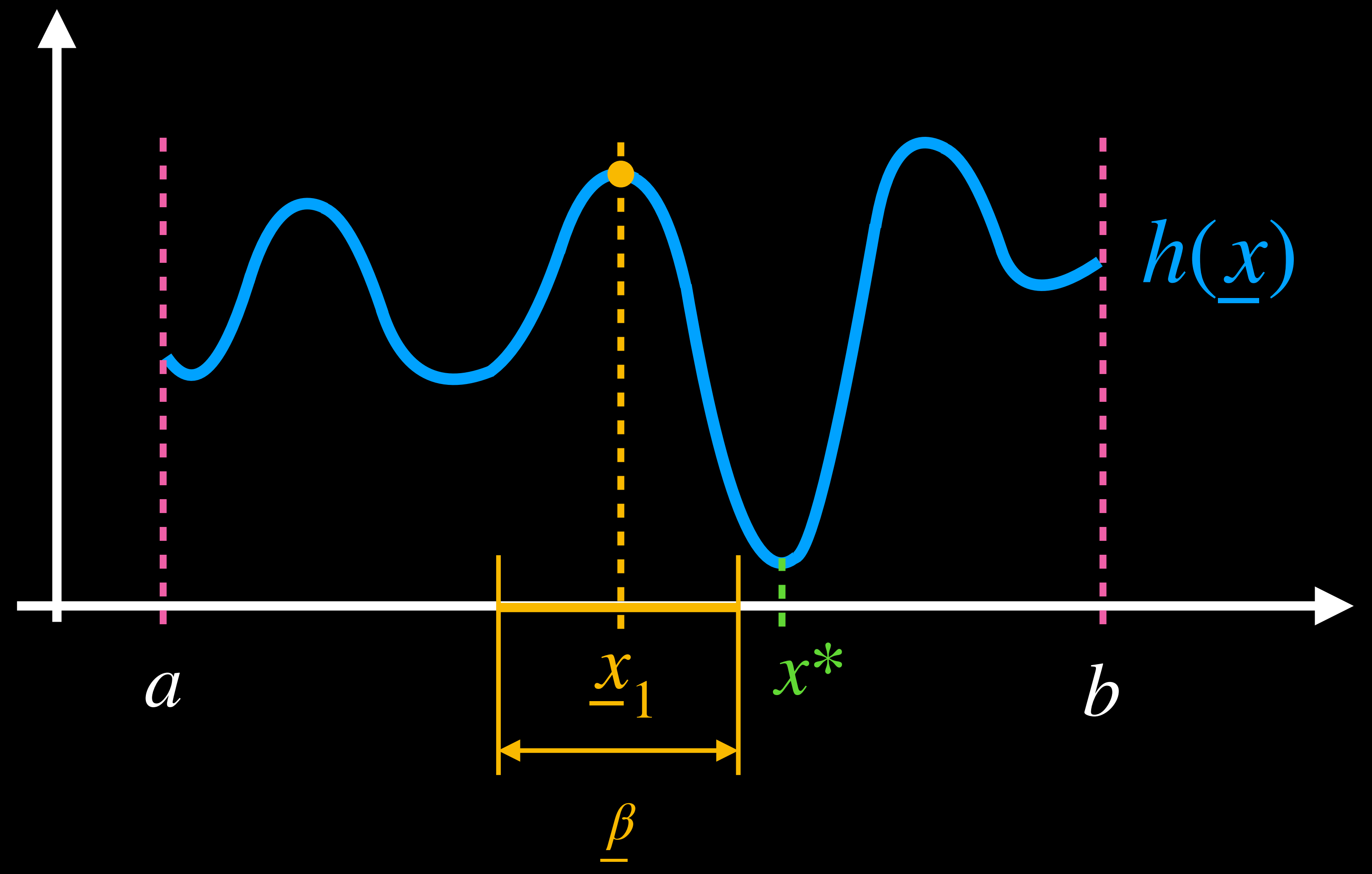
$T_t > 0$ is the "temperature" (parameter for the chance to accept worse points)

(example for a cooling scheme: $T_{t+1} := 0.95\, T_t$)

properties:
- simple
- convergence to a local minimum
- global convergence for the "right" cooling scheme
- can go out of local minima again

algorithm 4: simulated annealing

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \big(h(\underline{x})\big)$$



$$\underline{X}_{t+1} := \begin{cases} \underline{X}_t + \underline{U}_t^\beta & \text{if } U_t < e^{\frac{h(\underline{X}_t) - h(\underline{X}_t + \underline{U}_t^\beta)}{T_t}} \\ \underline{X}_t & \text{else} \end{cases}$$

$$U_t \sim \mathcal{U}(0,1)$$

$$\underline{U}_t^\beta \sim \mathcal{U}(-\tfrac{1}{2}\underline{\beta}, \tfrac{1}{2}\underline{\beta})$$

$T_t > 0$ is the "temperature" (parameter for the chance to accept worse points)

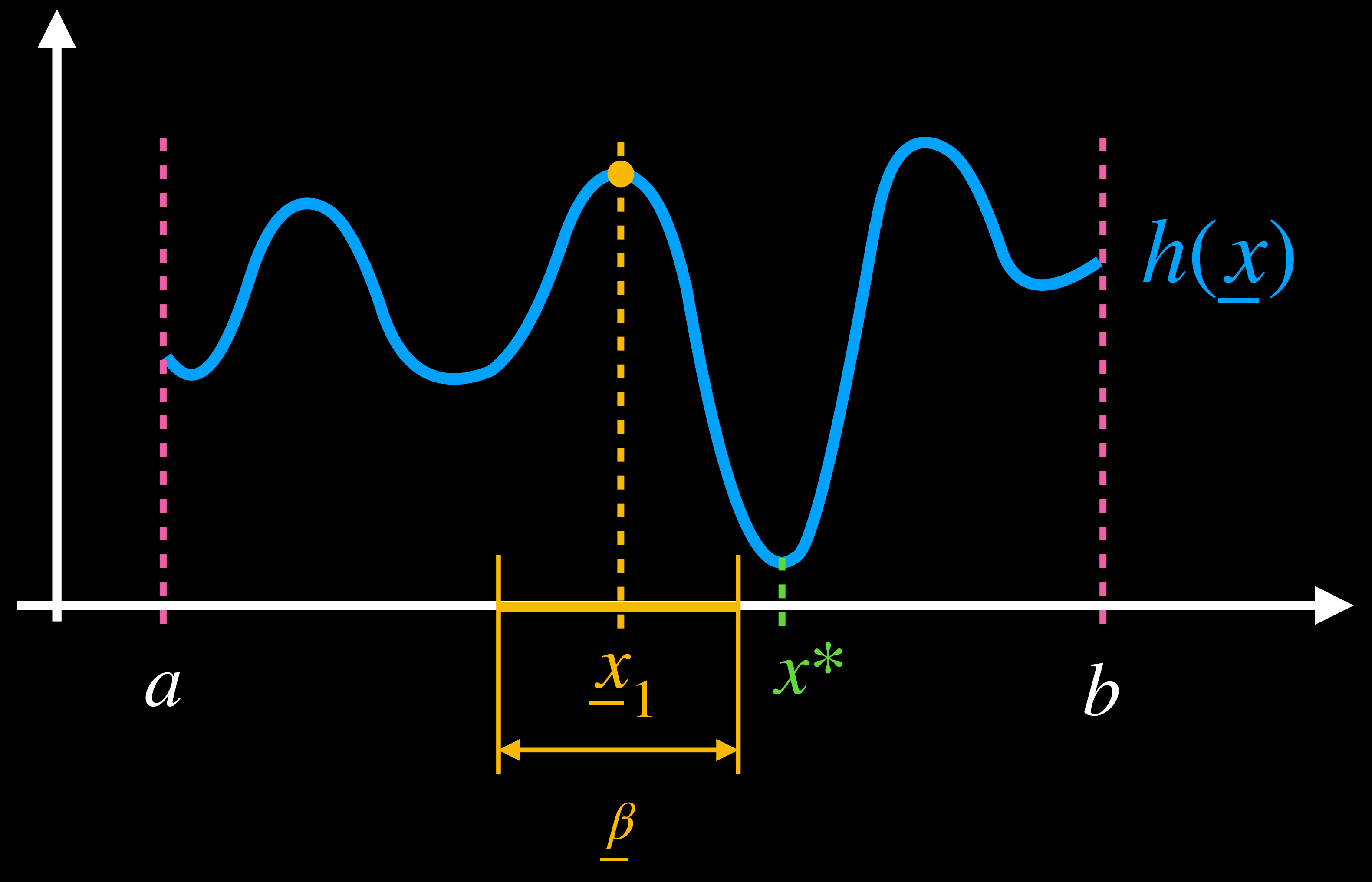
(example for a cooling scheme: $T_{t+1} := 0.95\, T_t$)

properties:
- simple
- convergence to a local minimum
- global convergence for the "right" cooling scheme
- can go out of local minima again

useful if:

algorithm 4: simulated annealing

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$



$$\underline{X}_{t+1} := \begin{cases} \underline{X}_t + \underline{U}_t^\beta & \text{if } U_t < e^{\frac{h(\underline{X}_t) - h(\underline{X}_t + \underline{U}_t^\beta)}{T_t}} \\ \underline{X}_t & \text{else} \end{cases}$$

$$U_t \sim \mathcal{U}(0,1)$$

$$\underline{U}_t^\beta \sim \mathcal{U}(-\tfrac{1}{2}\underline{\beta}, \tfrac{1}{2}\underline{\beta})$$

$T_t > 0$ is the "temperature" (parameter for the chance to accept worse points)

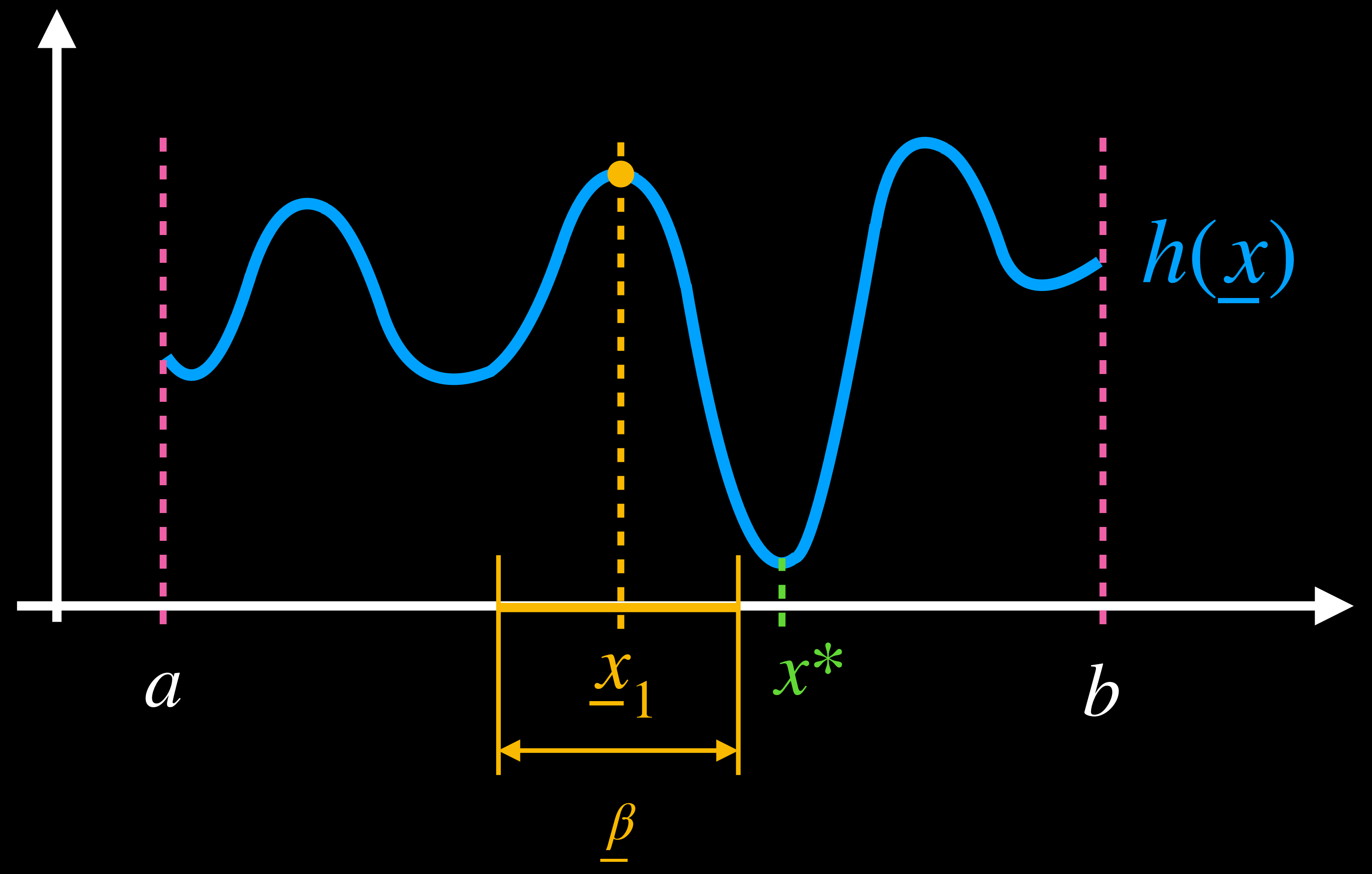
(example for a cooling scheme: $T_{t+1} := 0.95\, T_t$)

properties:
- simple
- convergence to a local minimum
- global convergence for the "right" cooling scheme
- can go out of local minima again

useful if:
- $\mathbb{S}$ is high-dimensional

algorithm 4: simulated annealing

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$



$$\underline{X}_{t+1} := \begin{cases} \underline{X}_t + \underline{U}_t^\beta & \text{if } U_t < e^{\frac{h(\underline{X}_t) - h(\underline{X}_t + \underline{U}_t^\beta)}{T_t}} \\ \underline{X}_t & \text{else} \end{cases}$$

$$U_t \sim \mathcal{U}(0,1)$$

$$\underline{U}_t^\beta \sim \mathcal{U}(-\tfrac{1}{2}\underline{\beta}, \tfrac{1}{2}\underline{\beta})$$

$T_t > 0$ is the "temperature" (parameter for the chance to accept worse points)

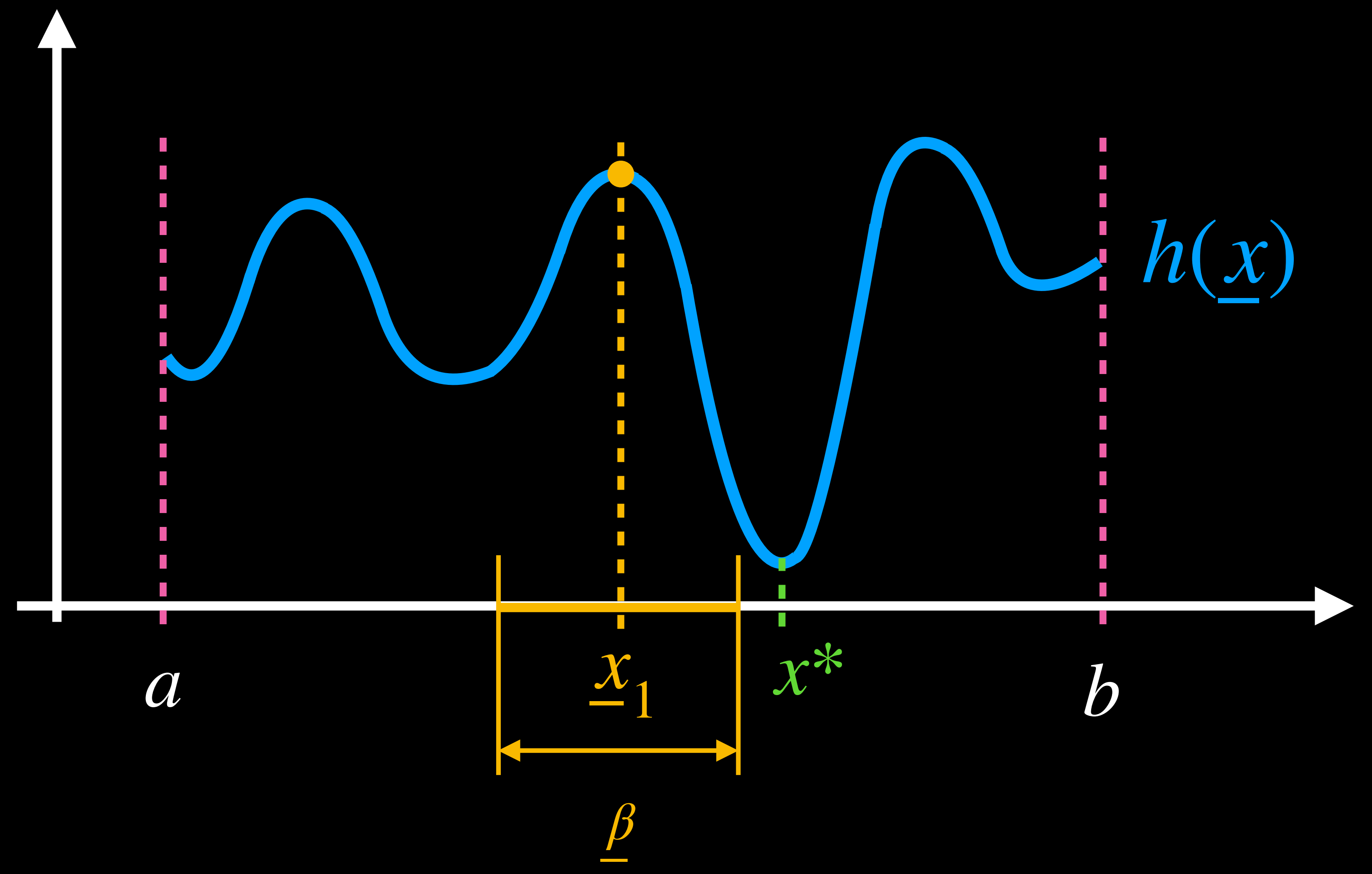
(example for a cooling scheme: $T_{t+1} := 0.95\, T_t$)

properties:
- simple
- convergence to a local minimum
- global convergence for the "right" cooling scheme
- can go out of local minima again

useful if:
- $\mathbb{S}$ is high-dimensional
- finding the global optimum is more important than finding the exact position of a (local/ global) optimum

algorithm 4: simulated annealing

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$



$$\underline{X}_{t+1} := \begin{cases} \underline{X}_t + \underline{U}_t^\beta & \text{if } U_t < e^{\frac{h(\underline{X}_t) - h(\underline{X}_t + \underline{U}_t^\beta)}{T_t}} \\ \underline{X}_t & \text{else} \end{cases}$$

$$U_t \sim \mathcal{U}(0,1)$$

$$\underline{U}_t^\beta \sim \mathcal{U}(-\tfrac{1}{2}\underline{\beta}, \tfrac{1}{2}\underline{\beta})$$

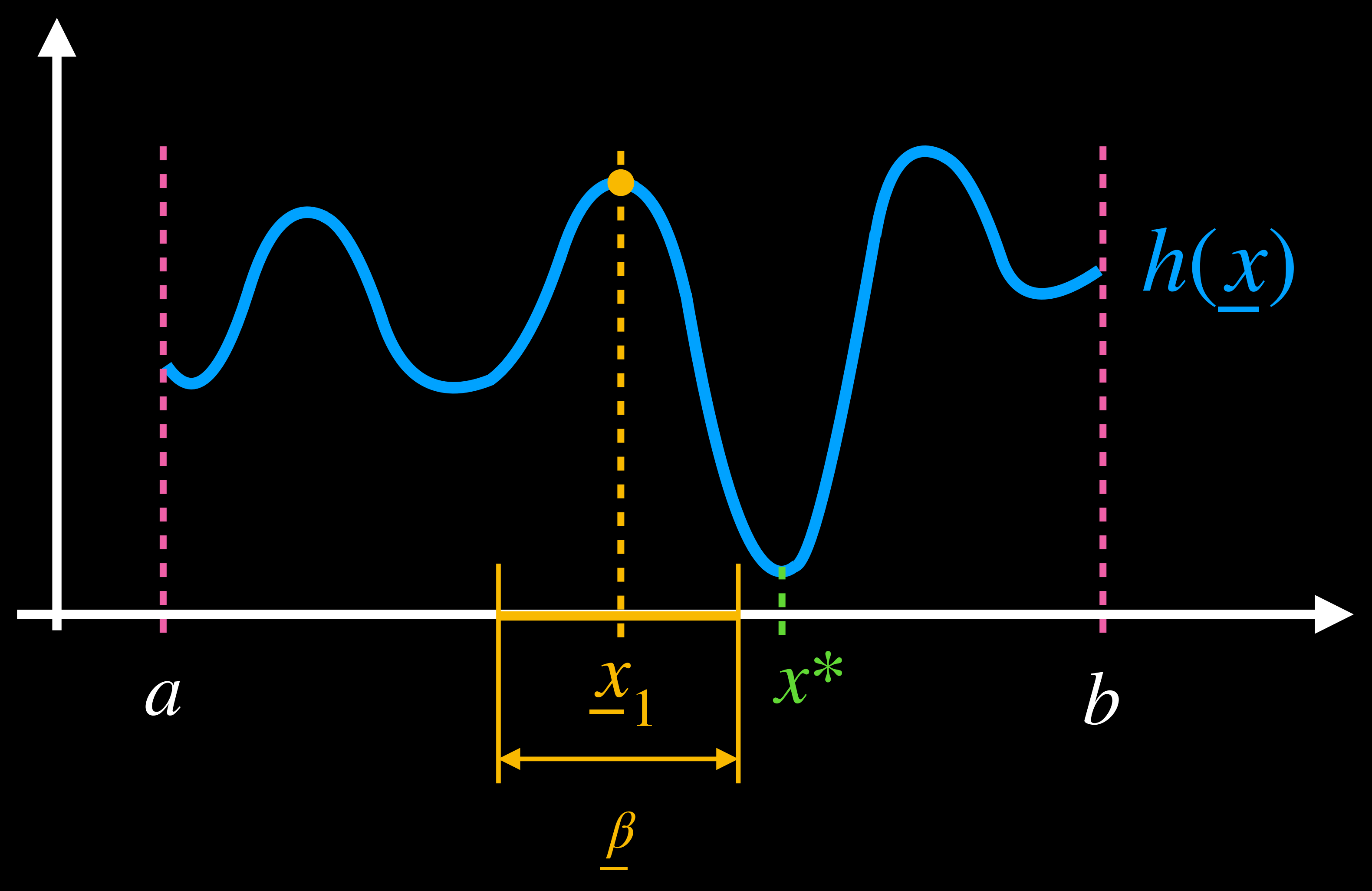
$T_t > 0$ is the "temperature" (parameter for the chance to accept worse points)

(example for a cooling scheme: $T_{t+1} := 0.95\, T_t$)

properties:
- simple
- convergence to a local minimum
- global convergence for the "right" cooling scheme
- can go out of local minima again

useful if:
- $\mathbb{S}$ is high-dimensional
- finding the global optimum is more important than finding the exact position of a (local/ global) optimum
- h has a lot of local minima

algorithm 4: simulated annealing



$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

$$\underline{X}_{t+1} := \begin{cases} \underline{X}_t + \underline{U}_t^\beta & \text{if } U_t < e^{\frac{h(\underline{X}_t) - h(\underline{X}_t + \underline{U}_t^\beta)}{T_t}} \\ \underline{X}_t & \text{else} \end{cases}$$

$$U_t \sim \mathcal{U}(0,1)$$

$$\underline{U}_t^\beta \sim \mathcal{U}(-\tfrac{1}{2}\underline{\beta}, \tfrac{1}{2}\underline{\beta})$$

$T_t > 0$ is the "temperature" (parameter for the chance to accept worse points)

(example for a cooling scheme: $T_{t+1} := 0.95\, T_t$)

properties:
- simple
- convergence to a local minimum
- global convergence for the "right" cooling scheme
- can go out of local minima again

useful if:
- $\mathbb{S}$ is high-dimensional
- finding the global optimum is more important than finding the exact position of a (local/ global) optimum
- h has a lot of local minima
- h is discrete

algorithm 5: evolutionary algorithms

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \arg\min_{\underline{x} \in \mathbb{S}} \left(h(\underline{x}) \right)$$

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

(1+1)-ES:

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

(1+1)-ES: (ES = Evolution Strategy)

algorithm 5: evolutionary algorithms

$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$

(1+1)-ES: (ES = Evolution Strategy)

$\underline{Y}_t$

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t$$

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

algorithm 5: evolutionary algorithms

$\underline{x}^* := \underset{\underline{x} \in \mathbb{S}}{\arg\min} \left(h(\underline{x}) \right)$

$\underline{Y}_t$ child

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{\underline{0}}, \underline{\underline{\Sigma}}\right)$$

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

$\underline{Y}_t$ child
$\underline{X}_t$ parent
$\underline{N}_t$ mutation

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \underset{\underline{x} \in \mathbb{S}}{\arg\min} \left(h(\underline{x}) \right)$$

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \underset{\underline{x} \in \mathbb{S}}{\arg\min} \left(h(\underline{x}) \right)$$

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \underset{\underline{x} \in \mathbb{S}}{\arg\min} \left(h(\underline{x}) \right)$$

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1}$$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

(1+λ)-ES:

algorithm 5: evolutionary algorithms

$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

(1+λ)-ES:

$$\underline{Y}_{t,k}$$

algorithm 5: evolutionary algorithms

$\underline{x}^* := \arg\min_{\underline{x} \in \mathbb{S}} \left(h(\underline{x}) \right)$

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min \left(h(\underline{Y}_t), h(\underline{X}_t) \right)$$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k}$$

algorithm 5: evolutionary algorithms

$\underline{x}^* := \underset{\underline{x} \in \mathbb{S}}{\arg\min} \left(h(\underline{x}) \right)$

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min \left(h(\underline{Y}_t), h(\underline{X}_t) \right)$$

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1}$$

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_{t,1}), \ldots, h(\underline{Y}_{t,\lambda}), h(\underline{X}_t)\right)$$

algorithm 5: evolutionary algorithms

$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{\underline{0}}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{\underline{0}}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{\underline{0}}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_{t,1}), \ldots, h(\underline{Y}_{t,\lambda}), h(\underline{X}_t)\right)$$

(μ,λ)-ES & (μ+λ)-ES :

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

(1+1)-ES:

(ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_{t,1}),\ldots,h(\underline{Y}_{t,\lambda}), h(\underline{X}_t)\right)$$

(μ,λ)-ES & (μ+λ)-ES :

$$\left(\underline{Y}'_{t,1},\ldots,\underline{Y}'_{t,\lambda}\right)$$

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_{t,1}),\ldots,h(\underline{Y}_{t,\lambda}), h(\underline{X}_t)\right)$$

(μ,λ)-ES & (μ+λ)-ES :

$$\left(\underline{Y}'_{t,1},\ldots,\underline{Y}'_{t,\lambda}\right) = recombination\left(\underline{X}_{t,1},\ldots,\underline{X}_{t,\mu}\right)$$

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

(1+1)-ES:

(ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_{t,1}), \ldots, h(\underline{Y}_{t,\lambda}), h(\underline{X}_t)\right)$$

(μ,λ)-ES & (μ+λ)-ES :

$$\left(\underline{Y}^{'}_{t,1}, \ldots, \underline{Y}^{'}_{t,\lambda}\right) = recombination\left(\underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu}\right)$$

$$\underline{Y}_{t,k}$$

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \arg\min_{\underline{x} \in \mathbb{S}} \left(h(\underline{x}) \right)$$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

(1+1)-ES:

(ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min \left(h(\underline{Y}_t), h(\underline{X}_t) \right)$$

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min \left(h(\underline{Y}_{t,1}), \ldots, h(\underline{Y}_{t,\lambda}), h(\underline{X}_t) \right)$$

(μ,λ)-ES & (μ+λ)-ES :

$$\left(\underline{Y}'_{t,1}, \ldots, \underline{Y}'_{t,\lambda} \right) = recombination \left(\underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu} \right)$$

$$\underline{Y}_{t,k} = \underline{Y}'_{t,k} + \underline{N}_{t,k}$$

algorithm 5: evolutionary algorithms

$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

($1+\lambda$)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_{t,1}), \ldots, h(\underline{Y}_{t,\lambda}), h(\underline{X}_t)\right)$$

(μ,λ)-ES & ($\mu+\lambda$)-ES :

$$\left(\underline{Y}'_{t,1}, \ldots, \underline{Y}'_{t,\lambda}\right) = recombination\left(\underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu}\right)$$

$$\underline{Y}_{t,k} = \underline{Y}'_{t,k} + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$

$\underline{Y}_t$ child
$\underline{X}_t$ parent
$\underline{N}_t$ mutation
$\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian
$\underline{\underline{\Sigma}}$ co-variance matrix

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_{t,1}), \ldots, h(\underline{Y}_{t,\lambda}), h(\underline{X}_t)\right)$$

(μ,λ)-ES & (μ+λ)-ES :

$$\left(\underline{Y}'_{t,1}, \ldots, \underline{Y}'_{t,\lambda}\right) = recombination\left(\underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu}\right)$$

$$\underline{Y}_{t,k} = \underline{Y}'_{t,k} + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\left(\underline{X}_{t+1,1}, \ldots, \underline{X}_{t+1,\mu}\right)$$

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_{t,1}),\ldots,h(\underline{Y}_{t,\lambda}), h(\underline{X}_t)\right)$$

(μ,λ)-ES & (μ+λ)-ES :

$$\left(\underline{Y}'_{t,1},\ldots,\underline{Y}'_{t,\lambda}\right) = recombination\left(\underline{X}_{t,1},\ldots,\underline{X}_{t,\mu}\right)$$

$$\underline{Y}_{t,k} = \underline{Y}'_{t,k} + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$$

$$\left(\underline{X}_{t+1,1},\ldots,\underline{X}_{t+1,\mu}\right) = selection_{(\mu,\lambda)\text{-ES}}\left(\underline{Y}_{t,1},\ldots,\underline{Y}_{t,\lambda}\right)$$

algorithm 5: evolutionary algorithms

$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

($1+\lambda$)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_{t,1}), \ldots, h(\underline{Y}_{t,\lambda}), h(\underline{X}_t)\right)$$

(μ,λ)-ES & ($\mu+\lambda$)-ES :

$$\left(\underline{Y}'_{t,1}, \ldots, \underline{Y}'_{t,\lambda}\right) = recombination\left(\underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu}\right)$$

$$\underline{Y}_{t,k} = \underline{Y}'_{t,k} + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\left(\underline{X}_{t+1,1}, \ldots, \underline{X}_{t+1,\mu}\right) = selection_{(\mu,\lambda)\text{-ES}}\left(\underline{Y}_{t,1}, \ldots, \underline{Y}_{t,\lambda}\right)$$

$$\left(\underline{X}_{t+1,1}, \ldots, \underline{X}_{t+1,\mu}\right)$$

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$

$\underline{Y}_t$ child
$\underline{X}_t$ parent
$\underline{N}_t$ mutation
$\mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$ multivariate Gaussian
$\underline{\underline{\Sigma}}$ co-variance matrix

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_{t,1}),\ldots,h(\underline{Y}_{t,\lambda}), h(\underline{X}_t)\right)$$

(μ,λ)-ES & (μ+λ)-ES :

$$\left(\underline{Y}'_{t,1},\ldots,\underline{Y}'_{t,\lambda}\right) = recombination\left(\underline{X}_{t,1},\ldots,\underline{X}_{t,\mu}\right)$$

$$\underline{Y}_{t,k} = \underline{Y}'_{t,k} + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$$

$$\left(\underline{X}_{t+1,1},\ldots,\underline{X}_{t+1,\mu}\right) = selection_{(\mu,\lambda)\text{-ES}}\left(\underline{Y}_{t,1},\ldots,\underline{Y}_{t,\lambda}\right)$$

$$\left(\underline{X}_{t+1,1},\ldots,\underline{X}_{t+1,\mu}\right) = selection_{(\mu+\lambda)\text{-ES}}\left(\underline{Y}_{t,1},\ldots,\underline{Y}_{t,\lambda},\underline{X}_{t,1},\ldots,\underline{X}_{t,\mu}\right)$$

algorithm 5: evolutionary algorithms

$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_{t,1}),\ldots,h(\underline{Y}_{t,\lambda}), h(\underline{X}_t)\right)$$

(μ,λ)-ES & (μ+λ)-ES :

$$\boxed{\left(\underline{Y}'_{t,1},\ldots,\underline{Y}'_{t,\lambda}\right) = recombination\left(\underline{X}_{t,1},\ldots,\underline{X}_{t,\mu}\right)}$$ recombination

$$\underline{Y}_{t,k} = \underline{Y}'_{t,k} + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$$

$$\left(\underline{X}_{t+1,1},\ldots,\underline{X}_{t+1,\mu}\right) = selection_{(\mu,\lambda)\text{-ES}}\left(\underline{Y}_{t,1},\ldots,\underline{Y}_{t,\lambda}\right)$$

$$\left(\underline{X}_{t+1,1},\ldots,\underline{X}_{t+1,\mu}\right) = selection_{(\mu+\lambda)\text{-ES}}\left(\underline{Y}_{t,1},\ldots,\underline{Y}_{t,\lambda},\underline{X}_{t,1},\ldots,\underline{X}_{t,\mu}\right)$$

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_{t,1}),\ldots,h(\underline{Y}_{t,\lambda}),h(\underline{X}_t)\right)$$

(μ,λ)-ES & (μ+λ)-ES :

$$\boxed{\left(\underline{Y}'_{t,1},\ldots,\underline{Y}'_{t,\lambda}\right) = recombination\left(\underline{X}_{t,1},\ldots,\underline{X}_{t,\mu}\right)}$$

recombination

$$\boxed{\underline{Y}_{t,k} = \underline{Y}'_{t,k} + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)}$$

mutation

$$\left(\underline{X}_{t+1,1},\ldots,\underline{X}_{t+1,\mu}\right) = selection_{(\mu,\lambda)\text{-ES}}\left(\underline{Y}_{t,1},\ldots,\underline{Y}_{t,\lambda}\right)$$

$$\left(\underline{X}_{t+1,1},\ldots,\underline{X}_{t+1,\mu}\right) = selection_{(\mu+\lambda)\text{-ES}}\left(\underline{Y}_{t,1},\ldots,\underline{Y}_{t,\lambda},\underline{X}_{t,1},\ldots,\underline{X}_{t,\mu}\right)$$

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \arg\min_{\underline{x} \in \mathbb{S}} \left(h(\underline{x}) \right)$$

$\underline{Y}_t$ child
$\underline{X}_t$ parent
$\underline{N}_t$ mutation
$\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian
$\underline{\underline{\Sigma}}$ co-variance matrix

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min \left(h(\underline{Y}_t), h(\underline{X}_t) \right)$$

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min \left(h(\underline{Y}_{t,1}), \ldots, h(\underline{Y}_{t,\lambda}), h(\underline{X}_t) \right)$$

(μ,λ)-ES & (μ+λ)-ES :

$$\left(\underline{Y}'_{t,1}, \ldots, \underline{Y}'_{t,\lambda}\right) = recombination \left(\underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu}\right)$$ recombination

$$\underline{Y}_{t,k} = \underline{Y}'_{t,k} + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$ mutation

$$\left(\underline{X}_{t+1,1}, \ldots, \underline{X}_{t+1,\mu}\right) = selection_{(\mu,\lambda)\text{-ES}} \left(\underline{Y}_{t,1}, \ldots, \underline{Y}_{t,\lambda}\right)$$

$$\left(\underline{X}_{t+1,1}, \ldots, \underline{X}_{t+1,\mu}\right) = selection_{(\mu+\lambda)\text{-ES}} \left(\underline{Y}_{t,1}, \ldots, \underline{Y}_{t,\lambda}, \underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu}\right)$$ selection

algorithm 5: evolutionary algorithms

$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

properties:

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_{t,1}),\ldots,h(\underline{Y}_{t,\lambda}), h(\underline{X}_t)\right)$$

(μ,λ)-ES & (μ+λ)-ES :

$$\boxed{\left(\underline{Y}'_{t,1},\ldots,\underline{Y}'_{t,\lambda}\right) = recombination\left(\underline{X}_{t,1},\ldots,\underline{X}_{t,\mu}\right)}$$

recombination

$$\boxed{\underline{Y}_{t,k} = \underline{Y}'_{t,k} + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)}$$

mutation

$$\left(\underline{X}_{t+1,1},\ldots,\underline{X}_{t+1,\mu}\right) = selection_{(\mu,\lambda)\text{-ES}}\left(\underline{Y}_{t,1},\ldots,\underline{Y}_{t,\lambda}\right)$$

$$\left(\underline{X}_{t+1,1},\ldots,\underline{X}_{t+1,\mu}\right) = selection_{(\mu+\lambda)\text{-ES}}\left(\underline{Y}_{t,1},\ldots,\underline{Y}_{t,\lambda},\underline{X}_{t,1},\ldots,\underline{X}_{t,\mu}\right)$$

selection

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

properties:

- global convergence posible

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_{t,1}), \ldots, h(\underline{Y}_{t,\lambda}), h(\underline{X}_t)\right)$$

(μ,λ)-ES & (μ+λ)-ES :

recombination

$$\boxed{\left(\underline{Y}'_{t,1}, \ldots, \underline{Y}'_{t,\lambda}\right) = recombination\left(\underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu}\right)}$$

mutation

$$\boxed{\underline{Y}_{t,k} = \underline{Y}'_{t,k} + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)}$$

selection

$$\left(\underline{X}_{t+1,1}, \ldots, \underline{X}_{t+1,\mu}\right) = selection_{(\mu,\lambda)\text{-ES}}\left(\underline{Y}_{t,1}, \ldots, \underline{Y}_{t,\lambda}\right)$$

$$\left(\underline{X}_{t+1,1}, \ldots, \underline{X}_{t+1,\mu}\right) = selection_{(\mu+\lambda)\text{-ES}}\left(\underline{Y}_{t,1}, \ldots, \underline{Y}_{t,\lambda}, \underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu}\right)$$

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$

$\underline{Y}_t$ child
$\underline{X}_t$ parent
$\underline{N}_t$ mutation
$\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian
$\underline{\underline{\Sigma}}$ co-variance matrix

properties:
- global convergence posible
- (can go out of local minima)

(1+1)-ES:

(ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_{t,1}), \ldots, h(\underline{Y}_{t,\lambda}), h(\underline{X}_t)\right)$$

(μ,λ)-ES & (μ+λ)-ES :

$$\boxed{\left(\underline{Y}'_{t,1}, \ldots, \underline{Y}'_{t,\lambda}\right) = recombination\left(\underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu}\right)}$$

recombination

$$\boxed{\underline{Y}_{t,k} = \underline{Y}'_{t,k} + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)}$$

mutation

$$\left(\underline{X}_{t+1,1}, \ldots, \underline{X}_{t+1,\mu}\right) = selection_{(\mu,\lambda)\text{-ES}}\left(\underline{Y}_{t,1}, \ldots, \underline{Y}_{t,\lambda}\right)$$

$$\left(\underline{X}_{t+1,1}, \ldots, \underline{X}_{t+1,\mu}\right) = selection_{(\mu+\lambda)\text{-ES}}\left(\underline{Y}_{t,1}, \ldots, \underline{Y}_{t,\lambda}, \underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu}\right)$$

selection

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$

$\underline{Y}_t$ child
$\underline{X}_t$ parent
$\underline{N}_t$ mutation
$\mathcal{N}\left(\underline{\underline{0}},\underline{\underline{\Sigma}}\right)$ multivariate Gaussian
$\underline{\underline{\Sigma}}$ co-variance matrix

properties:
- global convergence posible
- (can go out of local minima)

useful if:

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{\underline{0}},\underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{\underline{0}},\underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_{t,1}),\ldots,h(\underline{Y}_{t,\lambda}), h(\underline{X}_t)\right)$$

(μ,λ)-ES & (μ+λ)-ES :

recombination

$$\boxed{\left(\underline{Y}'_{t,1},\ldots,\underline{Y}'_{t,\lambda}\right) = recombination\left(\underline{X}_{t,1},\ldots,\underline{X}_{t,\mu}\right)}$$

mutation

$$\boxed{\underline{Y}_{t,k} = \underline{Y}'_{t,k} + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{\underline{0}},\underline{\underline{\Sigma}}\right)}$$

selection

$$\left(\underline{X}_{t+1,1},\ldots,\underline{X}_{t+1,\mu}\right) = \underset{(\mu,\lambda)\text{-ES}}{selection}\left(\underline{Y}_{t,1},\ldots,\underline{Y}_{t,\lambda}\right)$$

$$\left(\underline{X}_{t+1,1},\ldots,\underline{X}_{t+1,\mu}\right) = \underset{(\mu+\lambda)\text{-ES}}{selection}\left(\underline{Y}_{t,1},\ldots,\underline{Y}_{t,\lambda},\underline{X}_{t,1},\ldots,\underline{X}_{t,\mu}\right)$$

algorithm 5: evolutionary algorithms

$\underline{x}^* := \arg\min_{\underline{x} \in \mathbb{S}} \left(h(\underline{x}) \right)$

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min \left(h(\underline{Y}_t), h(\underline{X}_t) \right)$$

- $\underline{Y}_t$ child
- $\underline{X}_t$ parent
- $\underline{N}_t$ mutation
- $\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian
- $\underline{\underline{\Sigma}}$ co-variance matrix

properties:
- global convergence posible
- (can go out of local minima)

useful if:
- $\mathbb{S}$ is high-dimensional

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min \left(h(\underline{Y}_{t,1}), \ldots, h(\underline{Y}_{t,\lambda}), h(\underline{X}_t) \right)$$

(μ,λ)-ES & (μ+λ)-ES :

recombination

$$\left(\underline{Y}'_{t,1}, \ldots, \underline{Y}'_{t,\lambda} \right) = recombination \left(\underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu} \right)$$

mutation

$$\underline{Y}_{t,k} = \underline{Y}'_{t,k} + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

selection

$$\left(\underline{X}_{t+1,1}, \ldots, \underline{X}_{t+1,\mu} \right) = selection_{(\mu,\lambda)\text{-ES}} \left(\underline{Y}_{t,1}, \ldots, \underline{Y}_{t,\lambda} \right)$$

$$\left(\underline{X}_{t+1,1}, \ldots, \underline{X}_{t+1,\mu} \right) = selection_{(\mu+\lambda)\text{-ES}} \left(\underline{Y}_{t,1}, \ldots, \underline{Y}_{t,\lambda}, \underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu} \right)$$

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

$\underline{Y}_t$ child
$\underline{X}_t$ parent
$\underline{N}_t$ mutation
$\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian
$\underline{\underline{\Sigma}}$ co-variance matrix

properties:
- global convergence posible
- (can go out of local minima)

useful if:
- $\mathbb{S}$ is high-dimensional
- finding the global optimum is more important than finding the exact position of a (local/ global) optimum

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_{t,1}), \ldots, h(\underline{Y}_{t,\lambda}), h(\underline{X}_t)\right)$$

(μ,λ)-ES & (μ+λ)-ES :

$$\boxed{\left(\underline{Y}'_{t,1}, \ldots, \underline{Y}'_{t,\lambda}\right) = recombination\left(\underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu}\right)}$$

recombination

$$\boxed{\underline{Y}_{t,k} = \underline{Y}'_{t,k} + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)}$$

mutation

$$\left(\underline{X}_{t+1,1}, \ldots, \underline{X}_{t+1,\mu}\right) = \underset{(\mu,\lambda)\text{-ES}}{selection}\left(\underline{Y}_{t,1}, \ldots, \underline{Y}_{t,\lambda}\right)$$

$$\left(\underline{X}_{t+1,1}, \ldots, \underline{X}_{t+1,\mu}\right) = \underset{(\mu+\lambda)\text{-ES}}{selection}\left(\underline{Y}_{t,1}, \ldots, \underline{Y}_{t,\lambda}, \underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu}\right)$$

selection

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_{t,1}), \ldots, h(\underline{Y}_{t,\lambda}), h(\underline{X}_t)\right)$$

properties:
- global convergence posible
- (can go out of local minima)

useful if:
- $\mathbb{S}$ is high-dimensional
- finding the global optimum is more important than finding the exact position of a (local/ global) optimum
- h has a lot of local minima

(μ,λ)-ES & (μ+λ)-ES :

$$\boxed{\left(\underline{Y}'_{t,1}, \ldots, \underline{Y}'_{t,\lambda}\right) = recombination\left(\underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu}\right)}$$

recombination

$$\boxed{\underline{Y}_{t,k} = \underline{Y}'_{t,k} + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)}$$

mutation

$$\left(\underline{X}_{t+1,1}, \ldots, \underline{X}_{t+1,\mu}\right) = \underset{(\mu,\lambda)\text{-ES}}{selection}\left(\underline{Y}_{t,1}, \ldots, \underline{Y}_{t,\lambda}\right)$$

$$\left(\underline{X}_{t+1,1}, \ldots, \underline{X}_{t+1,\mu}\right) = \underset{(\mu+\lambda)\text{-ES}}{selection}\left(\underline{Y}_{t,1}, \ldots, \underline{Y}_{t,\lambda}, \underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu}\right)$$

selection

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

(1+1)-ES:

(ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_{t,1}),\ldots,h(\underline{Y}_{t,\lambda}), h(\underline{X}_t)\right)$$

(μ,λ)-ES & (μ+λ)-ES :

recombination

$$\boxed{\left(\underline{Y}'_{t,1},\ldots,\underline{Y}'_{t,\lambda}\right) = recombination\left(\underline{X}_{t,1},\ldots,\underline{X}_{t,\mu}\right)}$$

mutation

$$\boxed{\underline{Y}_{t,k} = \underline{Y}'_{t,k} + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)}$$

selection

$$\left(\underline{X}_{t+1,1},\ldots,\underline{X}_{t+1,\mu}\right) = selection_{(\mu,\lambda)\text{-ES}}\left(\underline{Y}_{t,1},\ldots,\underline{Y}_{t,\lambda}\right)$$

$$\left(\underline{X}_{t+1,1},\ldots,\underline{X}_{t+1,\mu}\right) = selection_{(\mu+\lambda)\text{-ES}}\left(\underline{Y}_{t,1},\ldots,\underline{Y}_{t,\lambda}, \underline{X}_{t,1},\ldots,\underline{X}_{t,\mu}\right)$$

properties:
- global convergence posible
- (can go out of local minima)

useful if:
- $\mathbb{S}$ is high-dimensional
- finding the global optimum is more important than finding the exact position of a (local/ global) optimum
- h has a lot of local minima
- h is discrete

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

$\underline{Y}_t$ child
$\underline{X}_t$ parent
$\underline{N}_t$ mutation
$\mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$ multivariate Gaussian
$\underline{\underline{\Sigma}}$ co-variance matrix

(1+1)-ES:

(ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_{t,1}), \ldots, h(\underline{Y}_{t,\lambda}), h(\underline{X}_t)\right)$$

(μ,λ)-ES & (μ+λ)-ES :

recombination

$$\boxed{\left(\underline{Y}'_{t,1}, \ldots, \underline{Y}'_{t,\lambda}\right) = recombination\left(\underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu}\right)}$$

mutation

$$\boxed{\underline{Y}_{t,k} = \underline{Y}'_{t,k} + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0},\underline{\underline{\Sigma}}\right)}$$

selection

$$\left(\underline{X}_{t+1,1}, \ldots, \underline{X}_{t+1,\mu}\right) = selection_{(\mu,\lambda)\text{-ES}}\left(\underline{Y}_{t,1}, \ldots, \underline{Y}_{t,\lambda}\right)$$

$$\left(\underline{X}_{t+1,1}, \ldots, \underline{X}_{t+1,\mu}\right) = selection_{(\mu+\lambda)\text{-ES}}\left(\underline{Y}_{t,1}, \ldots, \underline{Y}_{t,\lambda}, \underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu}\right)$$

properties:
- global convergence posible
- (can go out of local minima)

useful if:
- $\mathbb{S}$ is high-dimensional
- finding the global optimum is more important than finding the exact position of a (local/ global) optimum
- h has a lot of local minima
- h is discrete

improvement by
covariance matrix adaptation (CMA)

algorithm 5: evolutionary algorithms

$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$

(1+1)-ES: (ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

$\underline{Y}_t$ child

$\underline{X}_t$ parent

$\underline{N}_t$ mutation

$\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian

$\underline{\underline{\Sigma}}$ co-variance matrix

properties:
- global convergence posible
- (can go out of local minima)

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_{t,1}), \ldots, h(\underline{Y}_{t,\lambda}), h(\underline{X}_t)\right)$$

useful if:
- $\mathbb{S}$ is high-dimensional
- finding the global optimum is more important than finding the exact position of a (local/ global) optimum
- h has a lot of local minima
- h is discrete

(μ,λ)-ES & (μ+λ)-ES :

recombination

$$\boxed{\left(\underline{Y}'_{t,1}, \ldots, \underline{Y}'_{t,\lambda}\right) = recombination\left(\underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu}\right)}$$

mutation

$$\boxed{\underline{Y}_{t,k} = \underline{Y}'_{t,k} + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)}$$

selection

$$\left(\underline{X}_{t+1,1}, \ldots, \underline{X}_{t+1,\mu}\right) = selection_{(\mu,\lambda)\text{-ES}}\left(\underline{Y}_{t,1}, \ldots, \underline{Y}_{t,\lambda}\right)$$

$$\left(\underline{X}_{t+1,1}, \ldots, \underline{X}_{t+1,\mu}\right) = selection_{(\mu+\lambda)\text{-ES}}\left(\underline{Y}_{t,1}, \ldots, \underline{Y}_{t,\lambda}, \underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu}\right)$$

improvement by
covariance matrix adaptation (CMA)

$\underline{\underline{\Sigma}}_t$

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \underset{\underline{x}\in\mathbb{S}}{\arg\min}\left(h(\underline{x})\right)$$

(1+1)-ES:

(ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

- $\underline{Y}_t$ child
- $\underline{X}_t$ parent
- $\underline{N}_t$ mutation
- $\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian
- $\underline{\underline{\Sigma}}$ co-variance matrix

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_{t,1}), \ldots, h(\underline{Y}_{t,\lambda}), h(\underline{X}_t)\right)$$

(μ,λ)-ES & (μ+λ)-ES :

recombination

$$\boxed{\left(\underline{Y}'_{t,1}, \ldots, \underline{Y}'_{t,\lambda}\right) = recombination\left(\underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu}\right)}$$

mutation

$$\boxed{\underline{Y}_{t,k} = \underline{Y}'_{t,k} + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)}$$

selection

$$\left(\underline{X}_{t+1,1}, \ldots, \underline{X}_{t+1,\mu}\right) = selection_{(\mu,\lambda)\text{-ES}}\left(\underline{Y}_{t,1}, \ldots, \underline{Y}_{t,\lambda}\right)$$

$$\left(\underline{X}_{t+1,1}, \ldots, \underline{X}_{t+1,\mu}\right) = selection_{(\mu+\lambda)\text{-ES}}\left(\underline{Y}_{t,1}, \ldots, \underline{Y}_{t,\lambda}, \underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu}\right)$$

properties:
- global convergence posible
- (can go out of local minima)

useful if:
- $\mathbb{S}$ is high-dimensional
- finding the global optimum is more important than finding the exact position of a (local/ global) optimum
- h has a lot of local minima
- h is discrete

improvement by
covariance matrix adaptation (CMA)

$$\underline{\underline{\Sigma}}_t$$

- scaling per dimension (different diagonal entries)

algorithm 5: evolutionary algorithms

$$\underline{x}^* := \arg\min_{\underline{x}\in\mathbb{S}} \left(h(\underline{x})\right)$$

(1+1)-ES:

(ES = Evolution Strategy)

$$\underline{Y}_t = \underline{X}_t + \underline{N}_t \qquad \underline{N}_t \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_t), h(\underline{X}_t)\right)$$

- $\underline{Y}_t$ child
- $\underline{X}_t$ parent
- $\underline{N}_t$ mutation
- $\mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$ multivariate Gaussian
- $\underline{\underline{\Sigma}}$ co-variance matrix

(1+λ)-ES:

$$\underline{Y}_{t,k} = \underline{X}_t + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)$$

$$\underline{X}_{t+1} = \arg\min\left(h(\underline{Y}_{t,1}), \ldots, h(\underline{Y}_{t,\lambda}), h(\underline{X}_t)\right)$$

(μ,λ)-ES & (μ+λ)-ES :

recombination

$$\boxed{\left(\underline{Y}'_{t,1}, \ldots, \underline{Y}'_{t,\lambda}\right) = recombination\left(\underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu}\right)}$$

mutation

$$\boxed{\underline{Y}_{t,k} = \underline{Y}'_{t,k} + \underline{N}_{t,k} \qquad \underline{N}_{t,k} \sim \mathcal{N}\left(\underline{0}, \underline{\underline{\Sigma}}\right)}$$

selection

$$\left(\underline{X}_{t+1,1}, \ldots, \underline{X}_{t+1,\mu}\right) = selection_{(\mu,\lambda)\text{-ES}}\left(\underline{Y}_{t,1}, \ldots, \underline{Y}_{t,\lambda}\right)$$

$$\left(\underline{X}_{t+1,1}, \ldots, \underline{X}_{t+1,\mu}\right) = selection_{(\mu+\lambda)\text{-ES}}\left(\underline{Y}_{t,1}, \ldots, \underline{Y}_{t,\lambda}, \underline{X}_{t,1}, \ldots, \underline{X}_{t,\mu}\right)$$

properties:
- global convergence posible
- (can go out of local minima)

useful if:
- $\mathbb{S}$ is high-dimensional
- finding the global optimum is more important than finding the exact position of a (local/ global) optimum
- h has a lot of local minima
- h is discrete

improvement by
covariance matrix adaptation (CMA)

$\underline{\underline{\Sigma}}_t$

- scaling per dimension (different diagonal entries)
- correlations (off-diagonal entries)

end