Advanced Optimization

Lecture 3: Randomized Algorithms for Continuous Problems

Master AIC

Université Paris-Saclay, Orsay, France

Anne Auger
INRIA Saclay – Ile-de-France

Dimo Brockhoff
INRIA Saclay – Ile-de-France

Problem Statement

Continuous Domain Search/Optimization

- Task: minimize an objective function (*fitness* function, *loss* function) in continuous domain

$$f : \mathcal{X} \subseteq \mathbb{R}^n \to \mathbb{R}, \qquad \boldsymbol{x} \mapsto f(\boldsymbol{x})$$

- Black Box scenario (direct search scenario)

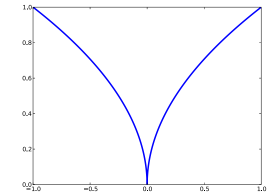
  - gradients are not available or not useful
  - problem domain specific knowledge is used only within the black box, e.g. within an appropriate encoding
- Search costs: number of function evaluations

What Makes a Function Difficult to Solve?

Why stochastic search?

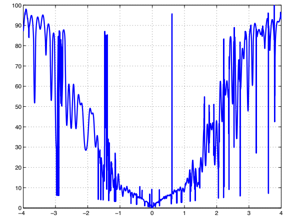
- non-linear, non-quadratic, non-convex

  on linear and quadratic functions much better search policies are available



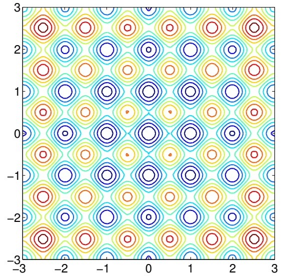
- ruggedness

  non-smooth, discontinuous, multimodal, and/or noisy function



- dimensionality (size of search space)

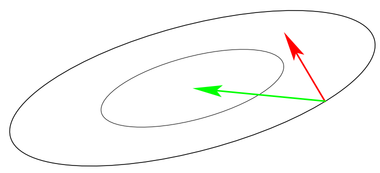
  (considerably) larger than three



- non-separability

  dependencies between the objective variables

- ill-conditioning



gradient direction Newton directic

Curse of Dimensionality

The term *Curse of dimensionality* (Richard Bellman) refers to problems caused by the rapid increase in volume associated with adding extra dimensions to a (mathematical) space.

Example: Consider placing 100 points onto a real interval, say $[0, 1]$. To get similar coverage, in terms of distance between adjacent points, of the 10-dimensional space $[0, 1]^{10}$ would require $100^{10} = 10^{20}$ points. A 100 points appear now as isolated points in a vast empty space.

Consequence: a search policy (e.g. exhaustive search) that is valuable in small dimensions might be useless in moderate or large dimensional search spaces.

Separable Problems

Definition (Separable Problem)

A function f is separable if

$$\arg\min_{(x_1,\ldots,x_n)} f(x_1,\ldots,x_n) = \left(\arg\min_{x_1} f(x_1,\ldots),\ldots,\arg\min_{x_n} f(\ldots,x_n)\right)$$

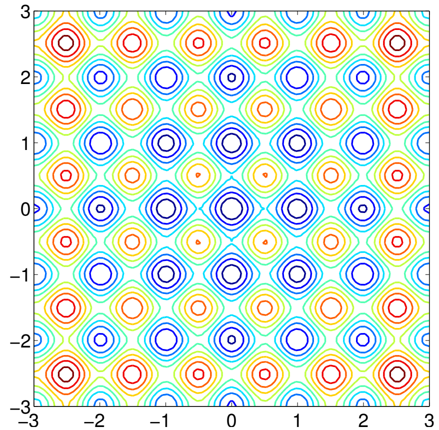
$\Rightarrow$ it follows that f can be optimized in a sequence of n independent 1-D optimization processes

Example: Additively decomposable functions

$$f(x_1,\ldots,x_n) = \sum_{i=1}^{n} f_i(x_i)$$

Rastrigin function

$f(\boldsymbol{x}) = 10n + \sum_{i=1}^{n}(x_i^2 - 10\cos(2\pi x_i))$



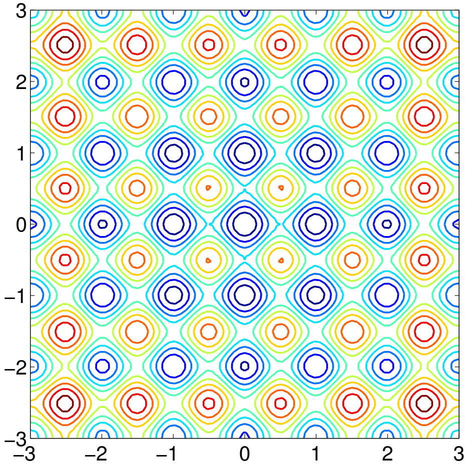
Non-Separable Problems

Building a non-separable problem from a separable one [1,2]

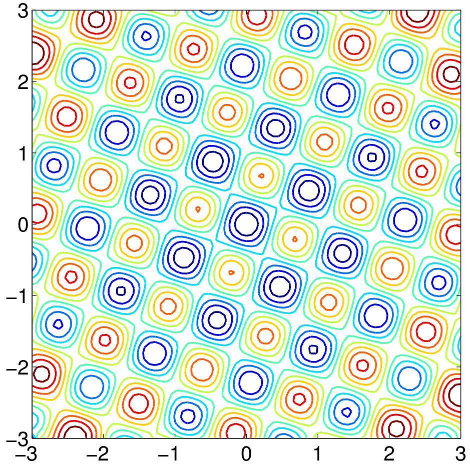
Rotating the coordinate system

- $f : \boldsymbol{x} \mapsto f(\boldsymbol{x})$ separable
- $f : \boldsymbol{x} \mapsto f(\mathbf{R}\boldsymbol{x})$ non-separable

R rotation matrix



$\xrightarrow{\mathbf{R}}$



[1] Hansen, Ostermeier, Gawelczyk (1995). On the adaptation of arbitrary normal mutation distributions in evolution strategies: The generating set adaptation. Sixth ICGA, pp. 57-64, Morgan Kaufmann

[2] Salomon (1996). "Reevaluating Genetic Algorithm Performance under Coordinate Rotation of Benchmark Functions; A survey of some theoretical and practical aspects of genetic algorithms." BioSystems, 39(3):263-278

Ill-Conditioned Problems

- If f is convex quadratic, $f : \boldsymbol{x} \mapsto \frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{H}\boldsymbol{x} = \frac{1}{2}\sum_i h_{i,i}\, x_i^2 + \frac{1}{2}\sum_{i\neq j} h_{i,j}\, x_i x_j$, with $\boldsymbol{H}$ positive, definite, symmetric matrix

  H* is the Hessian matrix of *f

- ill-conditioned means a high condition number of Hessian Matrix $\boldsymbol{H}$

$$\mathrm{cond}(\boldsymbol{H}) = \frac{\lambda_{\max}(\boldsymbol{H})}{\lambda_{\min}(\boldsymbol{H})}$$

Example / exercice

The level-sets of a function are defined as

$$\mathcal{L}_c = \{x \in \mathbb{R}^n | f(x) = c\},\ c \in \mathbb{R}\,.$$

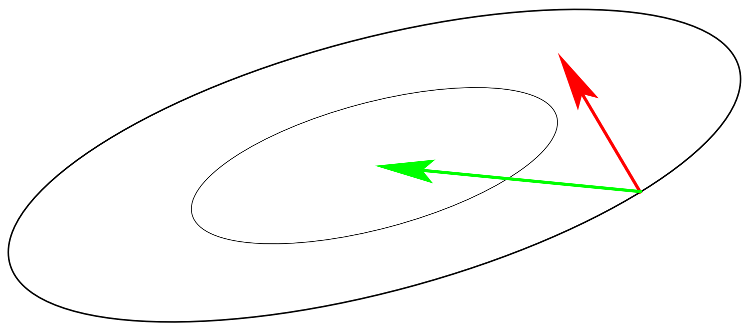
Consider the objective function $f(\boldsymbol{x}) = \frac{1}{2}(x_1^2 + 9x_2^2)$

1. Plot the level sets of f.
2. Compute the condition number of the Hessian matrix of f, relate it to the axis ratio of the level sets of f.
3. Generalize 1. and 2. to a general convex-quadratic function.

Ill-conditionned Problems

consider the curvature of the level sets of a function

ill-conditioned means "squeezed" lines of equal function value (high curvatures)



gradient direction $-f'(\boldsymbol{x})^{\mathrm{T}}$

Newton direction $-\boldsymbol{H}^{-1}f'(\boldsymbol{x})^{\mathrm{T}}$

Condition number equals nine here. Condition numbers up to 10^{10} are not unusual in real world problems.

Stochastic Search

A black box search template to minimize $f : \mathbb{R}^n \to \mathbb{R}$

Initialize distribution parameters $\boldsymbol{\theta}$, set population size $\lambda \in \mathbb{N}$

While not terminate

1. Sample distribution $P(\boldsymbol{x}|\boldsymbol{\theta}) \to \boldsymbol{x}_1, \ldots, \boldsymbol{x}_\lambda \in \mathbb{R}^n$
2. Evaluate $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_\lambda$ on f
3. Update parameters $\boldsymbol{\theta} \leftarrow F_\theta(\boldsymbol{\theta}, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_\lambda, f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_\lambda))$

Everything depends on the definition of P and F_θ

In Evolutionary Algorithms the distribution P is often implicitly defined via operators on a population, in particular, selection, recombination and mutation

Natural template for *Estimation of Distribution Algorithms*

A Simple Example: The Pure Random Search

Also an Ineffective Example

The Pure Random Search

- Sample uniformly at random a solution
- Return the best solution ever found

Exercice

See the exercice on the document "Exercices - class 1".

Non-adaptive Algorithm

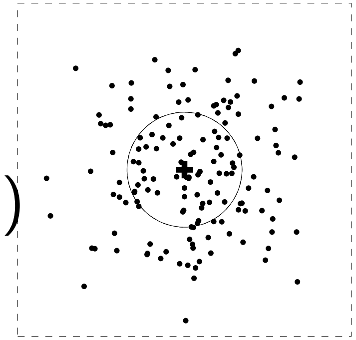
For the pure random search $P(\boldsymbol{x}|\boldsymbol{\theta})$ is independent of $\boldsymbol{\theta}$ (i.e. no $\boldsymbol{\theta}$ to be adapted): the algorithm is "blind"

In this class: present algorithms that are "much better" than that

Evolution Strategies

New search points are sampled normally distributed

$$\boldsymbol{x}_i = \boldsymbol{m} + \sigma\, \boldsymbol{y}_i \qquad \text{for } i = 1, \ldots, \lambda \text{ with } \boldsymbol{y}_i \text{ i.i.d.} \sim \mathcal{N}(\mathbf{0}, \mathbf{C})$$



as perturbations of $\boldsymbol{m}$, where $\boldsymbol{x}_i, \boldsymbol{m} \in \mathbb{R}^n$, $\sigma \in \mathbb{R}_+$, $\mathbf{C} \in \mathbb{R}^{n \times n}$

where

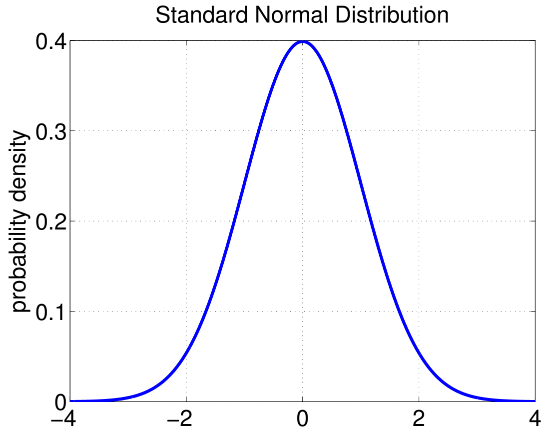
- the mean vector $\boldsymbol{m} \in \mathbb{R}^n$ represents the favorite solution
- the so-called step-size $\sigma \in \mathbb{R}_+$ controls the *step length*
- the covariance matrix $\mathbf{C} \in \mathbb{R}^{n \times n}$ determines the shape of the distribution ellipsoid

here, all new points are sampled with the same parameters

The question remains how to update $\boldsymbol{m}$, $\mathbf{C}$, and σ.

Normal Distribution

1-D case



probability density of the 1-D standard normal distribution $\mathcal{N}(0,1)$

(expected (mean) value, variance) = (0,1)

$$p(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right)$$

General case

- Normal distribution $\mathcal{N}(\boldsymbol{m}, \sigma^2)$

  (expected value, variance) $= (\boldsymbol{m}, \sigma^2)$

  density: $p_{\boldsymbol{m},\sigma}(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\boldsymbol{m})^2}{2\sigma^2}\right)$

- A normal distribution is entirely determined by its mean value and variance
- The family of normal distributions is closed under linear transformations: if X is normally distributed then a linear transformation $aX + b$ is also normally distributed
- **Exercice:** Show that $\boldsymbol{m} + \sigma\mathcal{N}(0,1) = \mathcal{N}(\boldsymbol{m}, \sigma^2)$

Normal Distribution

General case

A random variable following a 1-D normal distribution is determined by its mean value $\boldsymbol{m}$ and variance σ^2.

In the n-dimensional case it is determined by its mean vector and covariance matrix

Covariance Matrix

If the entries in a vector $\boldsymbol{X} = (X_1, \ldots, X_n)^T$ are random variables, each with finite variance, then the covariance matrix Σ is the matrix whose (i, j) entries are the covariance of (X_i, X_j)

$$\Sigma_{ij} = \mathrm{cov}(X_i, X_j) = \mathrm{E}\left[(X_i - \mu_i)(X_j - \mu_j)\right]$$

where $\mu_i = \mathrm{E}(X_i)$. Considering the expectation of a matrix as the expectation of each entry, we have

$$\Sigma = \mathrm{E}[(X - \mu)(X - \mu)^T]$$

Σ is symmetric, positive definite

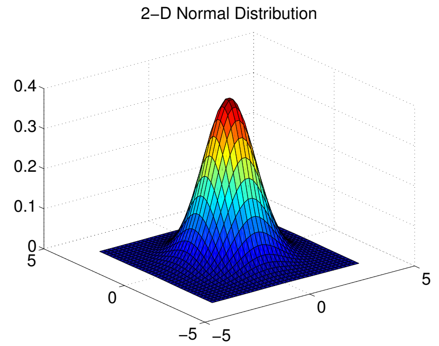
The Multi-Variate (n-Dimensional) Normal Distribution

Any multi-variate normal distribution $\mathcal{N}(\boldsymbol{m}, \mathbf{C})$ is uniquely determined by its mean value $\boldsymbol{m} \in \mathbb{R}^n$ and its symmetric positive definite $n \times n$ covariance matrix $\mathbf{C}$.

$$\text{density: } p_{\mathcal{N}(\boldsymbol{m},\mathbf{C})}(\boldsymbol{x}) = \frac{1}{(2\pi)^{n/2}|\mathbf{C}|^{1/2}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{m})^{\mathrm{T}}\mathbf{C}^{-1}(\boldsymbol{x}-\boldsymbol{m})\right),$$

The mean value $\boldsymbol{m}$

- determines the displacement (translation)
- value with the largest density (modal value)
- the distribution is symmetric about the distribution mean



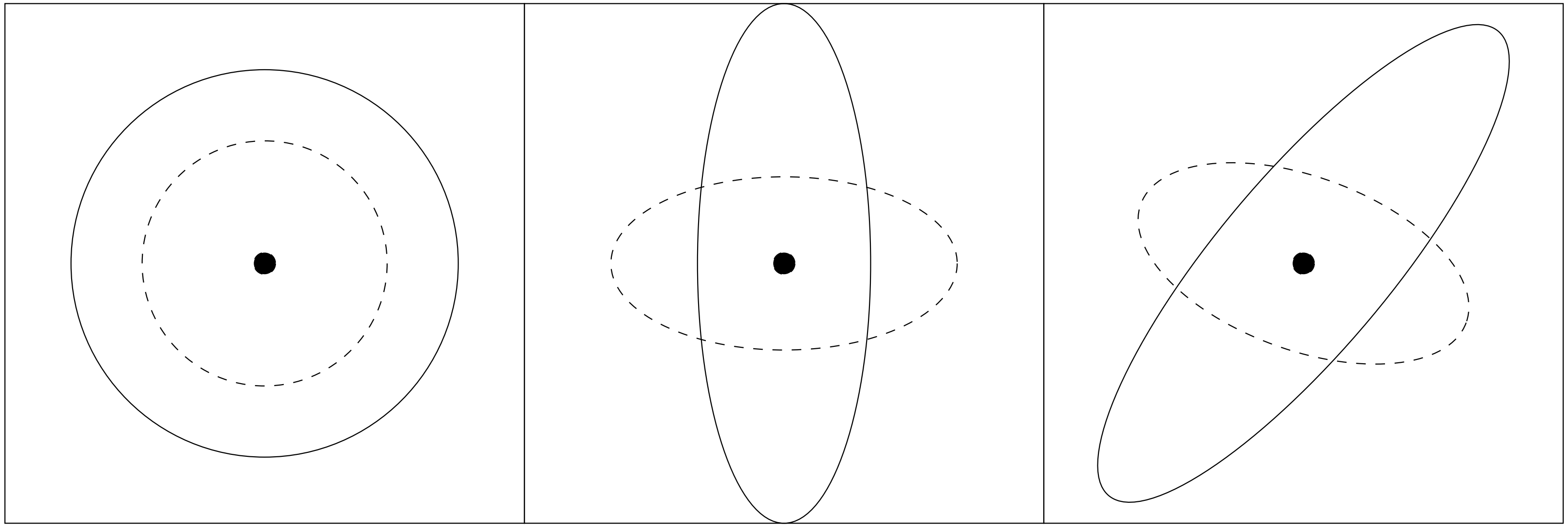
$$\mathcal{N}(\boldsymbol{m}, \mathbf{C}) = \boldsymbol{m} + \mathcal{N}(0, \mathbf{C})$$

The covariance matrix $\mathbf{C}$

- determines the shape
- geometrical interpretation: any covariance matrix can be uniquely identified with the iso-density ellipsoid $\{\boldsymbol{x} \in \mathbb{R}^n \mid (\boldsymbol{x}-\boldsymbol{m})^{\mathrm{T}}\mathbf{C}^{-1}(\boldsymbol{x}-\boldsymbol{m}) = 1\}$

...any covariance matrix can be uniquely identified with the iso-density ellipsoid $\{\boldsymbol{x} \in \mathbb{R}^n \mid (\boldsymbol{x} - \boldsymbol{m})^{\mathrm{T}} \mathbf{C}^{-1} (\boldsymbol{x} - \boldsymbol{m}) = 1\}$

Lines of Equal Density



$\mathcal{N}(\boldsymbol{m}, \sigma^2 \mathbf{I}) \sim \boldsymbol{m} + \sigma \mathcal{N}(\mathbf{0}, \mathbf{I})$
one degree of freedom σ
components are independent standard normally distributed

$\mathcal{N}(\boldsymbol{m}, \mathbf{D}^2) \sim \boldsymbol{m} + \mathbf{D}\,\mathcal{N}(\mathbf{0}, \mathbf{I})$
n degrees of freedom
components are independent, scaled

$\mathcal{N}(\boldsymbol{m}, \mathbf{C}) \sim \boldsymbol{m} + \mathbf{C}^{\frac{1}{2}} \mathcal{N}(\mathbf{0}, \mathbf{I})$
$(n^2 + n)/2$ degrees of freedom
components are correlated

where $\mathbf{I}$ is the identity matrix (isotropic case) and $\mathbf{D}$ is a diagonal matrix (reasonable for separable problems) and $\mathbf{A} \times \mathcal{N}(\mathbf{0}, \mathbf{I}) \sim \mathcal{N}(\mathbf{0}, \mathbf{A}\mathbf{A}^{\mathrm{T}})$ holds for all $\mathbf{A}$.

Adapting the mean …

Evolution Strategies (ES)

Simple Update for Mean Vector

Let μ: # parents, λ: # offspring

Plus (elitist) and comma (non-elitist) selection

$(\mu + \lambda)$-ES: selection in $\{\text{parents}\} \cup \{\text{offspring}\}$
(μ, λ)-ES: selection in $\{\text{offspring}\}$

ES algorithms emerged in the community of bio-inspired methods where a parallel between optimization and evolution of species as described by Darwin served in the origin as inspiration for the methods. Nowadays this parallel is mainly visible through the terminology used: candidate solutions are parents or offspring, the objective function is a fitness function, ...

$(1 + 1)$-ES

Sample one offspring from parent $\boldsymbol{m}$

$$\boldsymbol{x} = \boldsymbol{m} + \sigma \mathcal{N}(\mathbf{0}, \mathbf{C})$$

If $\boldsymbol{x}$ better than $\boldsymbol{m}$ select

$$\boldsymbol{m} \leftarrow \boldsymbol{x}$$

The $(\mu/\mu, \lambda)$-ES - Update of the mean vector

Non-elitist selection and intermediate (weighted) recombination

Given the i-th solution point $\boldsymbol{x}_i = \boldsymbol{m} + \sigma \underbrace{\boldsymbol{y}_i}_{\sim\mathcal{N}(\mathbf{0},\mathbf{C})}$

Let $\boldsymbol{x}_{i:\lambda}$ the i-th ranked solution point, such that $f(\boldsymbol{x}_{1:\lambda}) \leq \cdots \leq f(\boldsymbol{x}_{\lambda:\lambda})$.

Notation: we denote $\boldsymbol{y}_{i:\lambda}$ the vector such that $\boldsymbol{x}_{i:\lambda} = \boldsymbol{m} + \sigma \boldsymbol{y}_{i:\lambda}$

Exercice: realize that $\boldsymbol{y}_{i:\lambda}$ is generally not distributed as $\mathcal{N}(\mathbf{0}, \mathbf{C})$

The new mean reads

$$\boldsymbol{m} \leftarrow \sum_{i=1}^{\mu} w_i \, \boldsymbol{x}_{i:\lambda} \;=\; \boldsymbol{m} + \sigma \underbrace{\sum_{i=1}^{\mu} w_i \, \boldsymbol{y}_{i:\lambda}}_{=:\, \boldsymbol{y}_w}$$

where

$$w_1 \geq \cdots \geq w_\mu > 0, \quad \sum_{i=1}^{\mu} w_i = 1, \quad \frac{1}{\sum_{i=1}^{\mu} {w_i}^2} =: \mu_w \approx \frac{\lambda}{4}$$

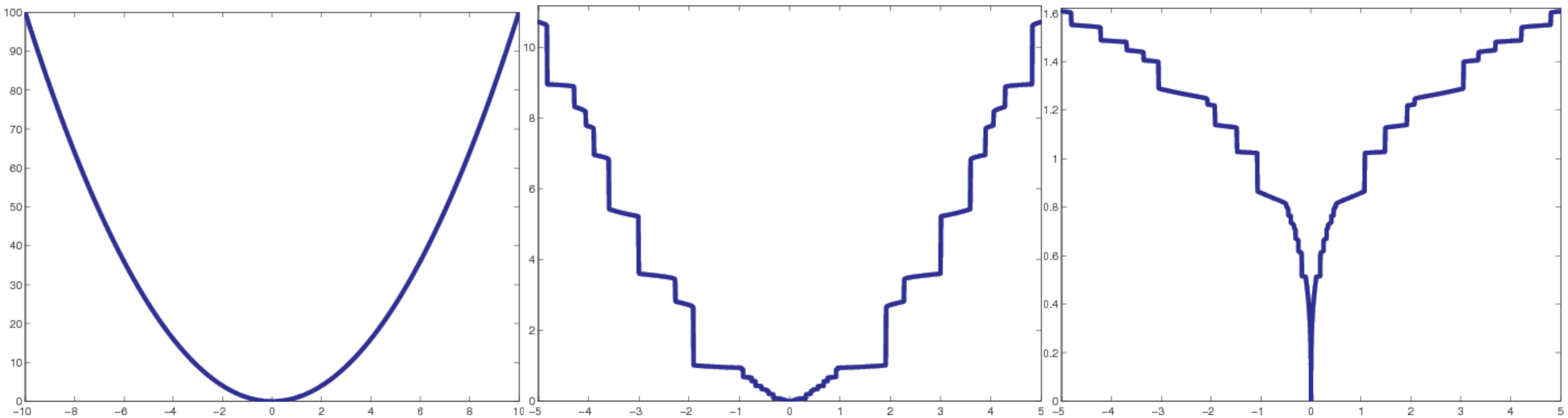
The best μ points are selected from the new solutions (non-elitistic) and weighted intermediate recombination is applied.

Invariance Under Monotonically Increasing Functions

Rank-based algorithms

Update of all parameters uses only the ranks

$$f(x_{1:\lambda}) \leq f(x_{2:\lambda}) \leq \ldots \leq f(x_{\lambda:\lambda})$$

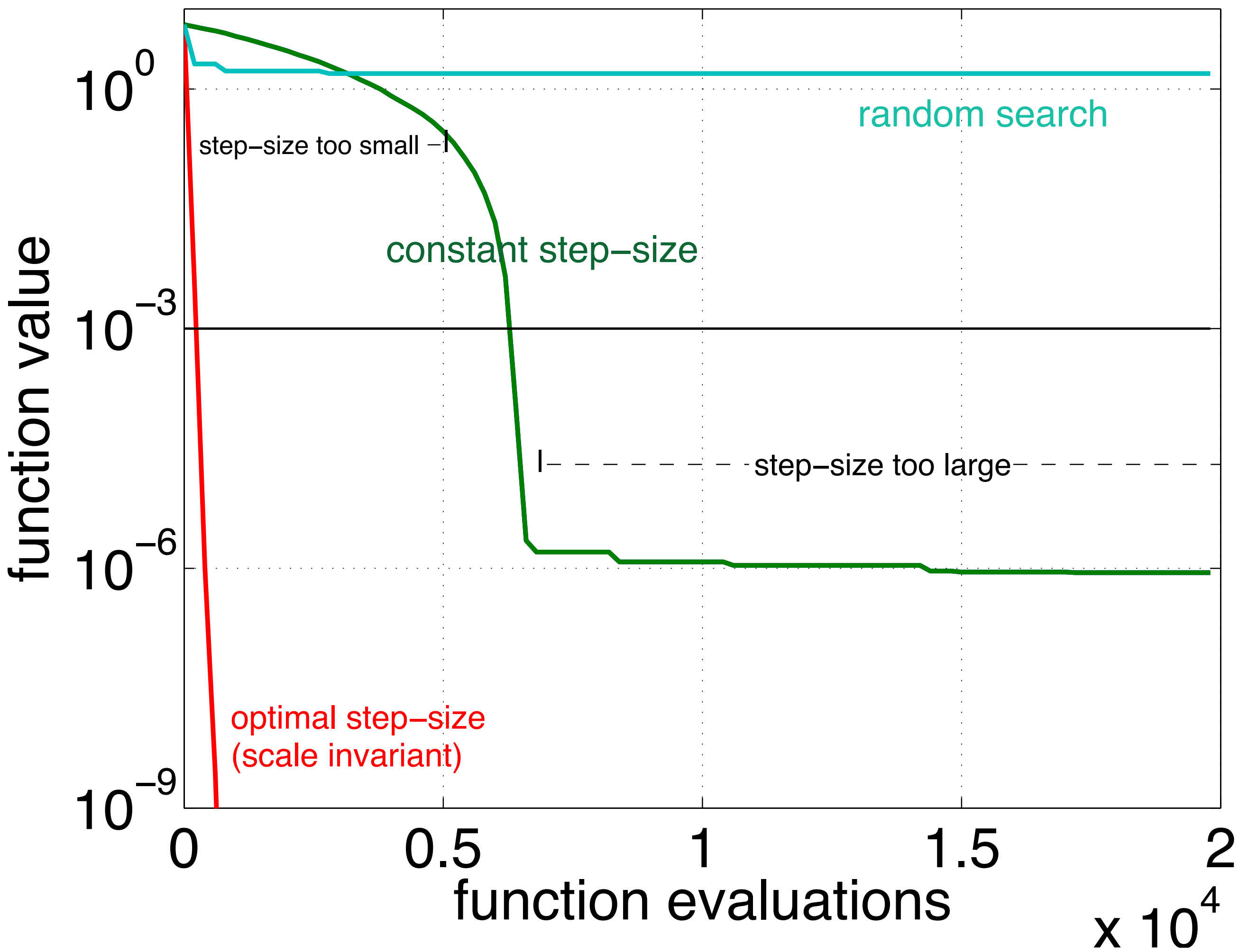


$$g(f(x_{1:\lambda})) \leq g(f(x_{2:\lambda})) \leq \ldots \leq g(f(x_{\lambda:\lambda})) \quad \forall g$$

g is strictly monotonically increasing
g preserves ranks

Adapting the step-size …

Why Step-Size Control?



(1+1)-ES
(red & green)

$$f(\boldsymbol{x}) = \sum_{i=1}^{n} x_i^2$$

in $[-2.2, 0.8]^n$
for $n = 10$

Methods for Step-Size Control

- 1/5-th success rule[a][b], often applied with "+"-selection

  increase step-size if more than 20% of the new solutions are successful, decrease otherwise

- σ-self-adaptation[c], applied with ","-selection

  mutation is applied to the step-size and the better one, according to the objective function value, is selected

  simplified "global" self-adaptation

- path length control[d] (Cumulative Step-size Adaptation, CSA)[e], applied with ","-selection

---

[a] Rechenberg 1973, *Evolutionsstrategie, Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*, Frommann-Holzboog

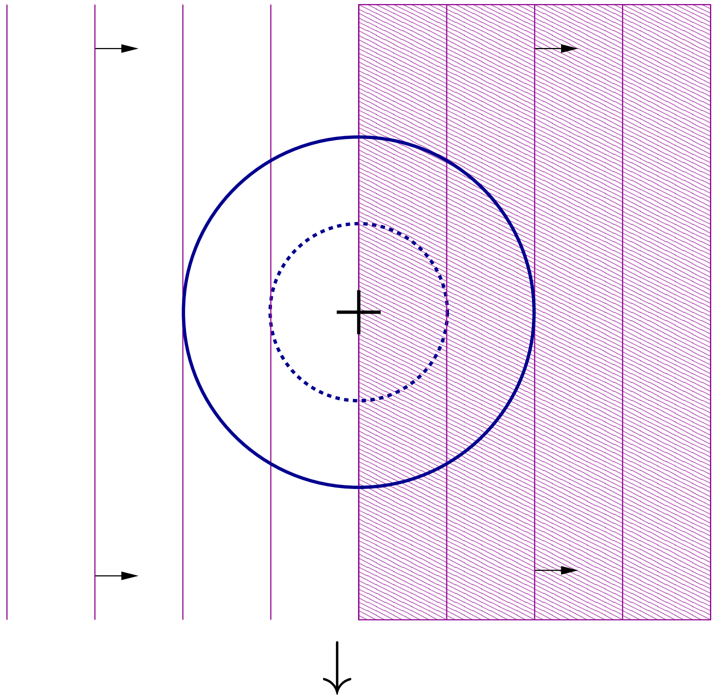
[b] Schumer and Steiglitz 1968. Adaptive step size random search. *IEEE TAC*

[c] Schwefel 1981, *Numerical Optimization of Computer Models*, Wiley

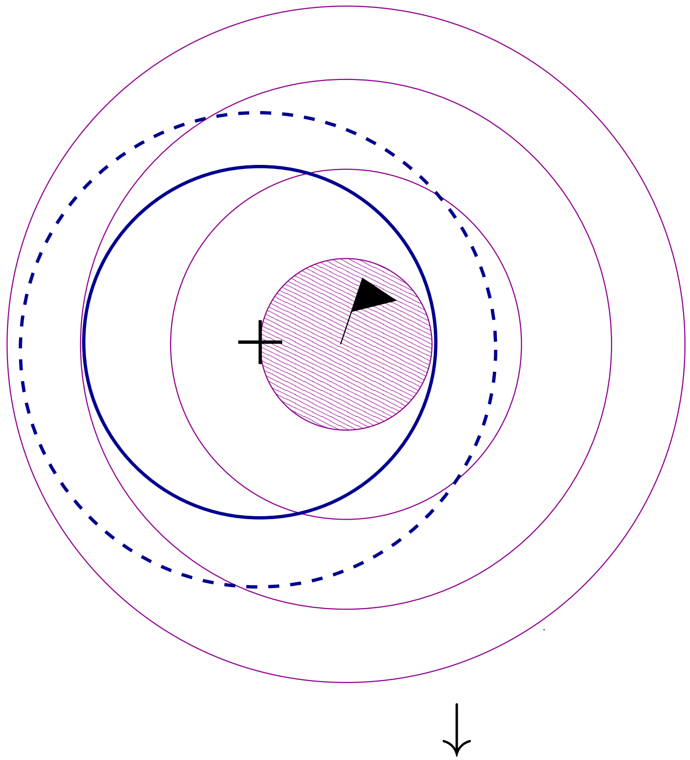
[d] Hansen & Ostermeier 2001, Completely Derandomized Self-Adaptation in Evolution Strategies, *Evol. Comput. 9(2)*

[e] Ostermeier *et al* 1994, Step-size adaptation based on non-local use of selection information, *PPSN IV*

One-fifth success rule

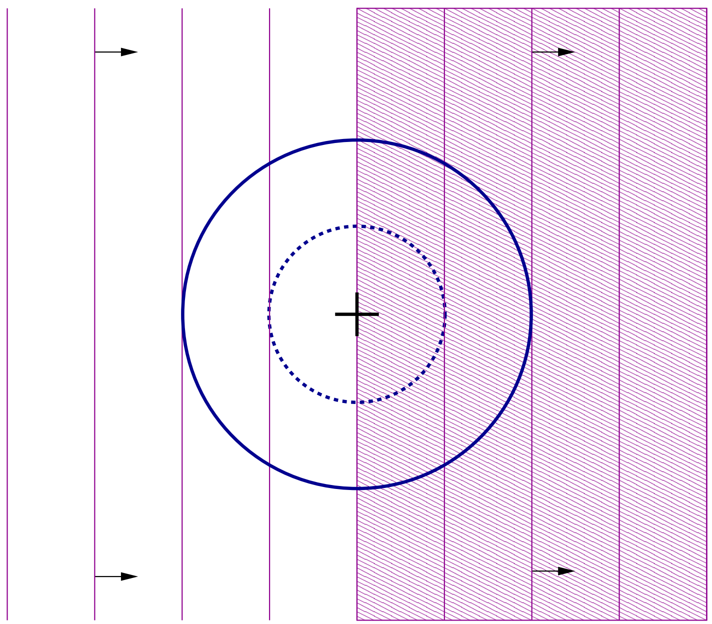


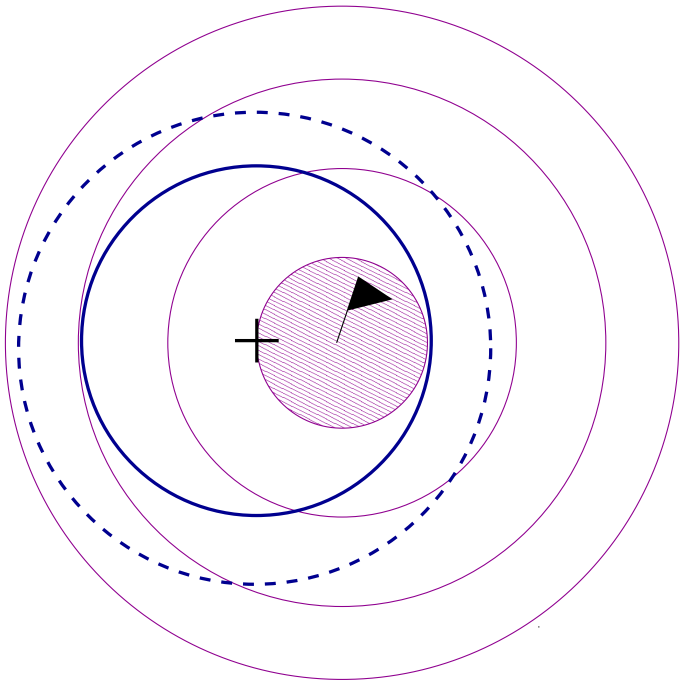
increase σ



decrease σ

One-fifth success rule





Probability of success (p_s)

Probability of success (p_s)

1/2

1/5

“too small”

One-fifth success rule

p_s: # of successful offspring / # offspring (per iteration)

$$\sigma \leftarrow \sigma \times \exp\left(\frac{1}{3} \times \frac{p_s - p_{\text{target}}}{1 - p_{\text{target}}}\right)$$

Increase σ if $p_s > p_{\text{target}}$
Decrease σ if $p_s < p_{\text{target}}$

$(1+1)$-ES

$p_{target} = 1/5$

IF *offspring better parent*

$p_s = 1$, $\sigma \leftarrow \sigma \times \exp(1/3)$

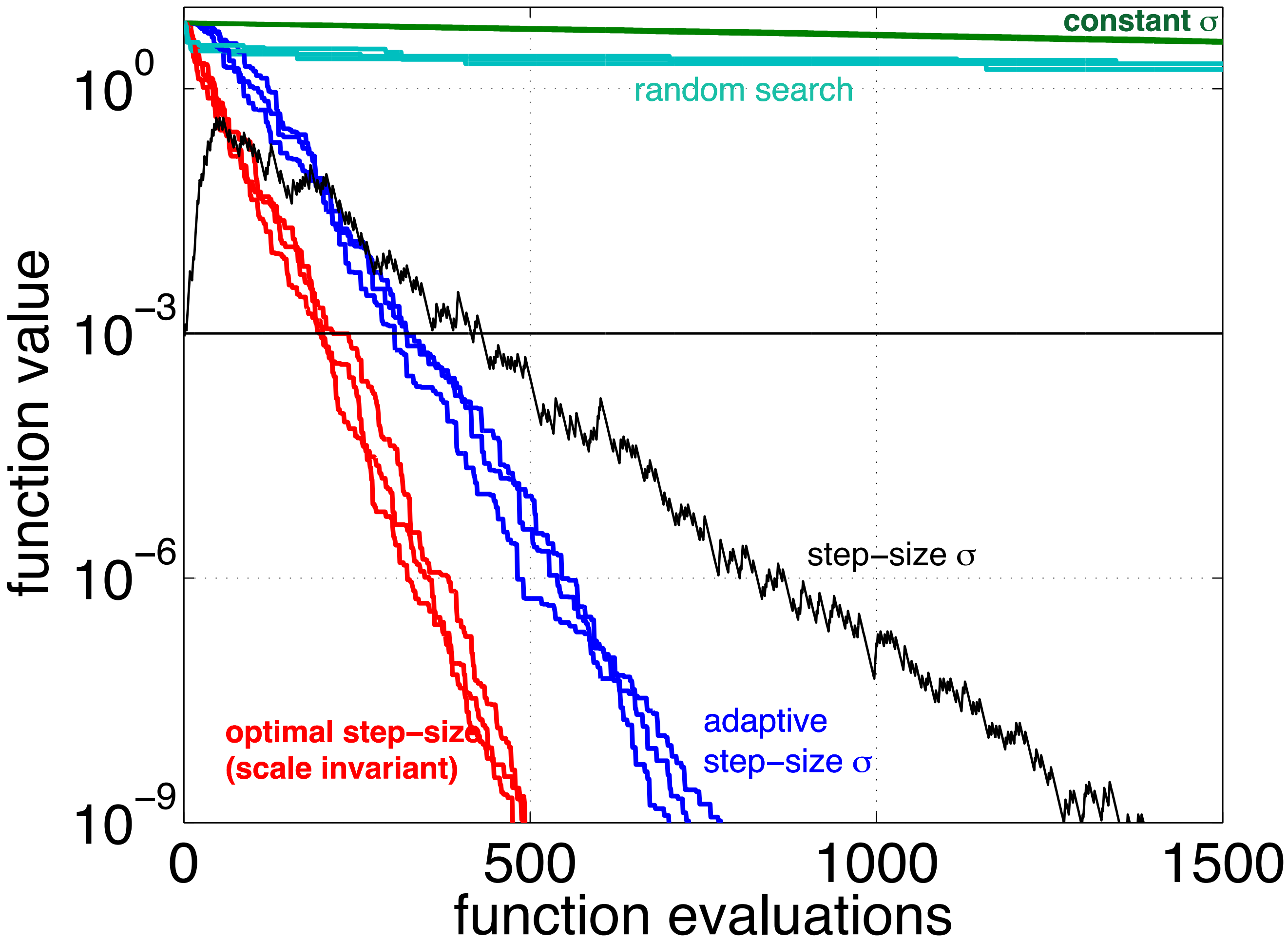
ELSE

$p_s = 0$, $\sigma \leftarrow \sigma / \exp(1/3)^{1/4}$

Step-size adaptation

What is achieved

$(1+1)$-ES with one-fifth success rule (blue)

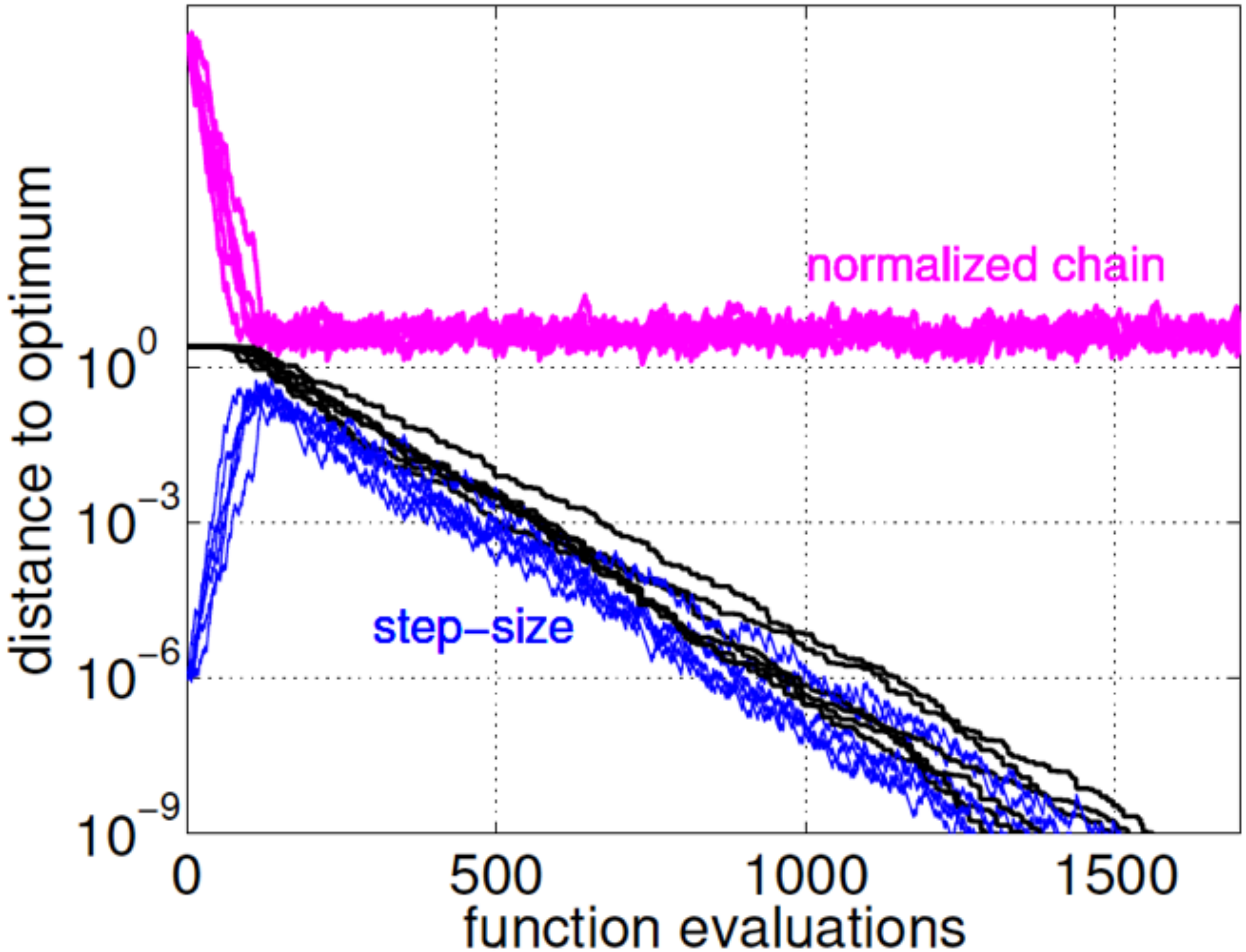


$$f(\boldsymbol{x}) = \sum_{i=1}^{n} x_i^2$$

in $[-0.2, 0.8]^n$
for $n = 10$

Linear convergence

What do we achieve?

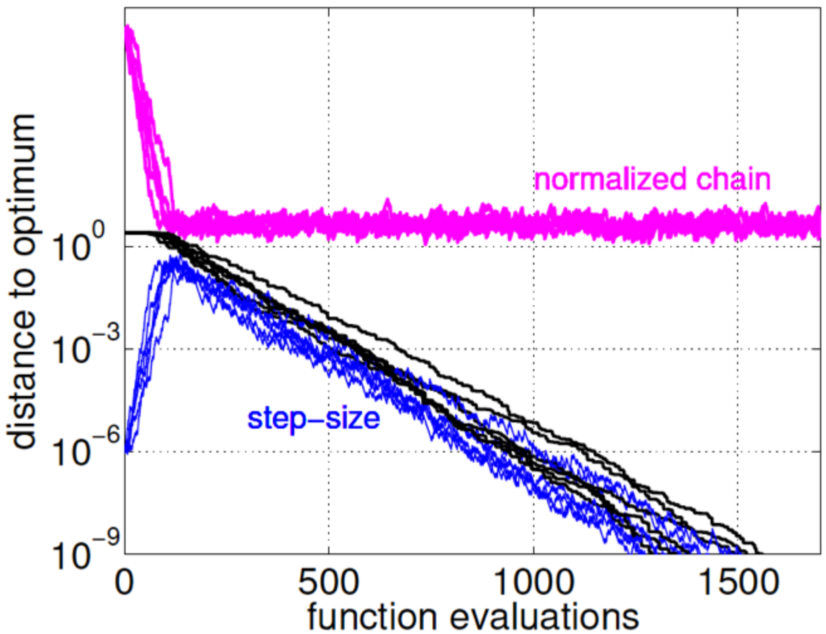


(1+1)-ES

$$f(\boldsymbol{x}) = \sum_{i=1}^{n} x_i^2$$

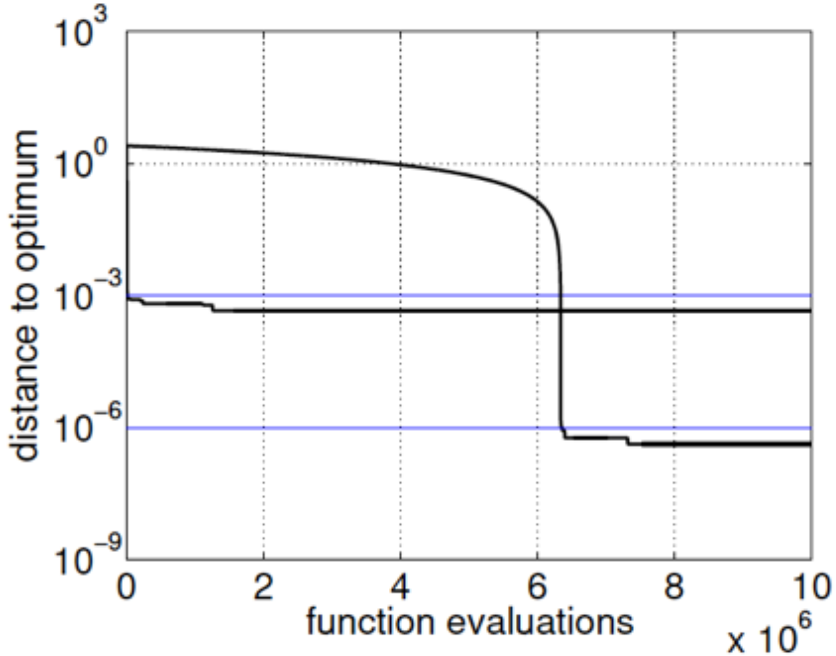
for $n = 10$

Adaptive versus Constant Step-size



adaptive step-size

constant step-size



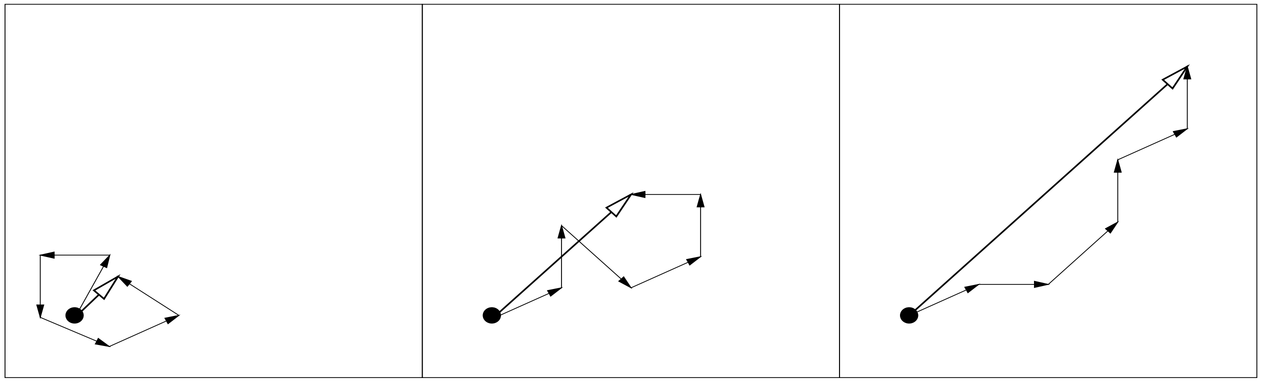
Path Length Control (CSA)

The Concept of Cumulative Step-Size Adaptation

$$\begin{aligned} \boldsymbol{x}_i &= \boldsymbol{m} + \sigma\, \boldsymbol{y}_i \\ \boldsymbol{m} &\leftarrow \boldsymbol{m} + \sigma\, \boldsymbol{y}_w \end{aligned}$$

Measure the length of the *evolution path*

the pathway of the mean vector $\boldsymbol{m}$ in the iteration sequence



$\Downarrow$ decrease σ | $\Downarrow$ increase σ

Path Length Control (CSA)

The Equations

Sampling of solutions, notations as on slide "The $(\mu/\mu, \lambda)$-ES - Update of the mean vector" with $\mathbf{C}$ equal to the identity.

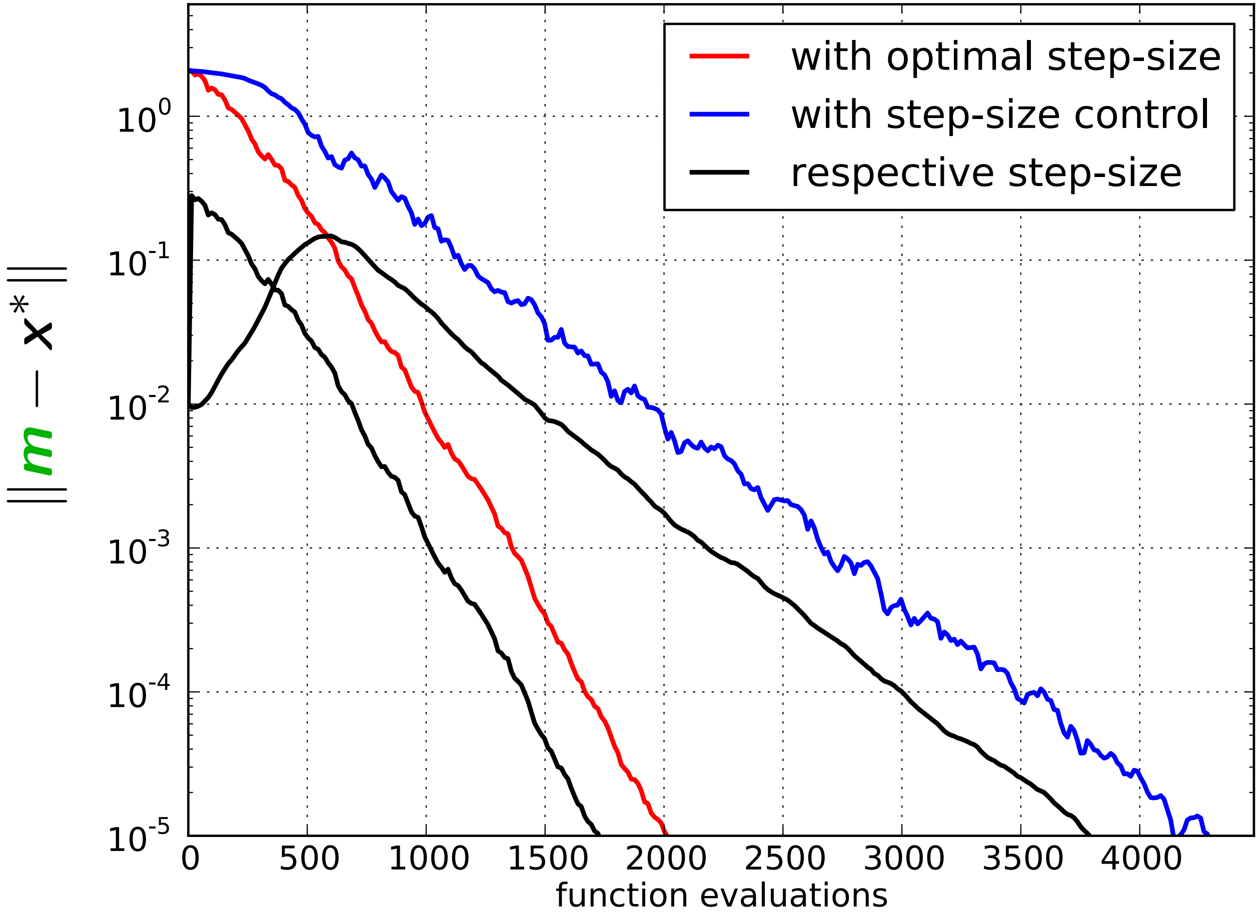
Initialize $\boldsymbol{m} \in \mathbb{R}^n$, $\sigma \in \mathbb{R}_+$, evolution path $\boldsymbol{p}_\sigma = \mathbf{0}$,
set $c_\sigma \approx 4/n$, $d_\sigma \approx 1$.

$$\begin{aligned}
\boldsymbol{m} &\leftarrow \boldsymbol{m} + \sigma \boldsymbol{y}_w \quad \text{where } \boldsymbol{y}_w = \textstyle\sum_{i=1}^{\mu} w_i \, \boldsymbol{y}_{i:\lambda} && \text{update mean} \\
\boldsymbol{p}_\sigma &\leftarrow (1 - c_\sigma)\, \boldsymbol{p}_\sigma + \underbrace{\sqrt{1 - (1 - c_\sigma)^2}}_{\text{accounts for } 1-c_\sigma} \quad \underbrace{\sqrt{\mu_w}}_{\text{accounts for } w_i} \quad \boldsymbol{y}_w \\
\sigma &\leftarrow \sigma \times \underbrace{\exp\left(\frac{c_\sigma}{d_\sigma}\left(\frac{\|\boldsymbol{p}_\sigma\|}{\mathsf{E}\|\mathcal{N}(\mathbf{0}, \mathbf{I})\|} - 1\right)\right)}_{>1 \iff \|\boldsymbol{p}_\sigma\| \text{ is greater than its expectation}} && \text{update step-size}
\end{aligned}$$

Step-size adaptation

What is achieved

$(5/5, 10)$-CSA-ES, default parameters

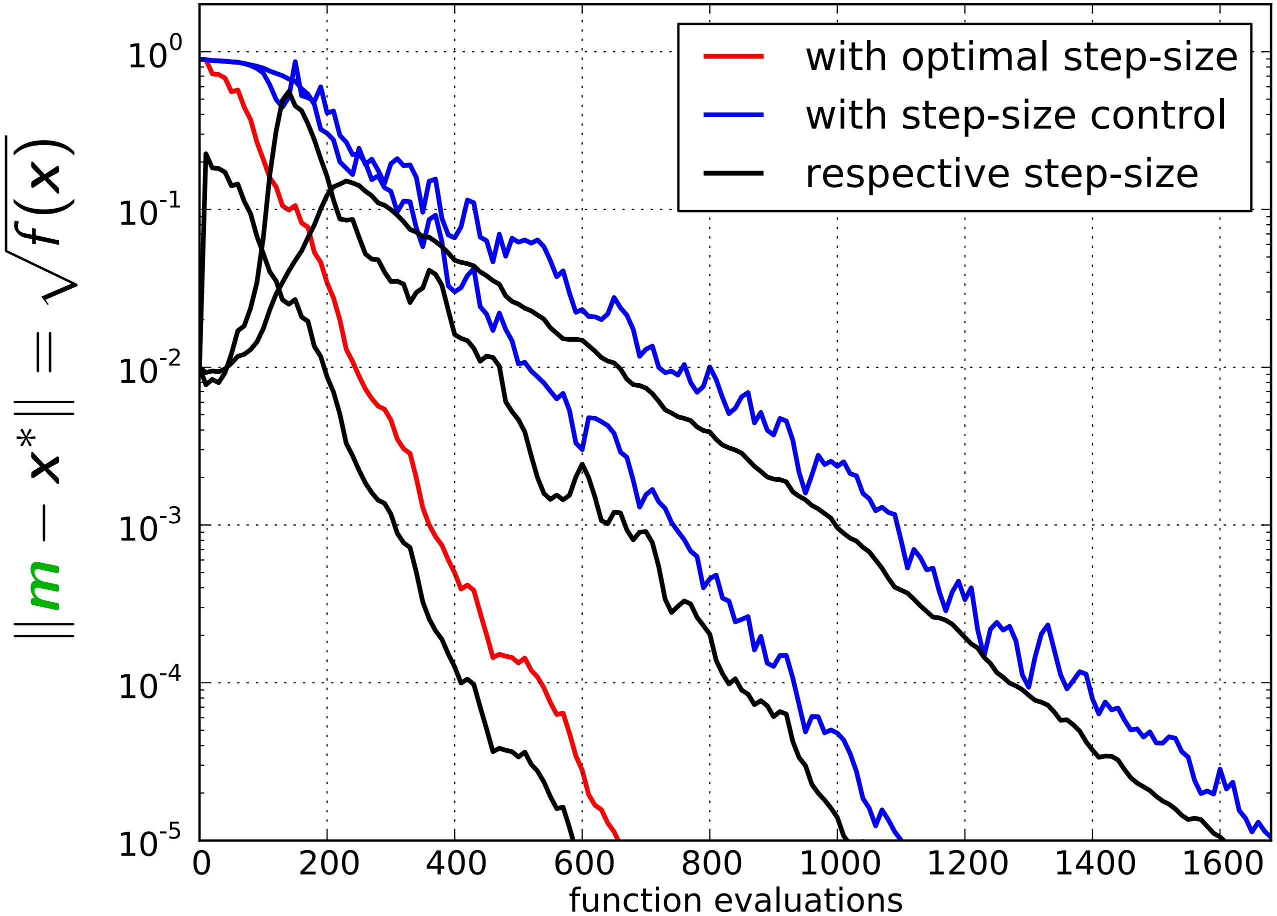


$$f(\boldsymbol{x}) = \sum_{i=1}^{n} x_i^2$$

in $[-0.2, 0.8]^n$
for $n = 30$

Why Step-Size Control?

$(5/5_w, 10)$-ES



$$f(\boldsymbol{x}) = \sum_{i=1}^{n} x_i^2$$

for $n = 10$

and

$\boldsymbol{x}^0 \in [-0.2, 0.8]^n$

comparing optimal versus default damping parameter d_σ:

$\frac{1700}{1100} \approx 1.5$

On linear convergence …

Hitting Time versus Convergence

Finite hitting time for all epsilon

$$T_\epsilon = \inf\{t \in \mathbb{N}, \mathbf{X}_t \in B(\mathbf{x}^\star, \epsilon)\}$$

$$T_\epsilon < \infty \text{ for all } \epsilon > 0$$

$\Longleftrightarrow$ under some regularity conditions on the algorithm and the function e.g.) (1+1)-ES on a spherical function

Convergence towards the optimum

$$\lim_{t \to \infty} \mathbf{X}_t = \mathbf{x}^\star$$

$$\Longleftrightarrow \quad \forall \epsilon > 0, \; \exists T_\epsilon < \infty \text{ such that } \|\mathbf{X}_t - \mathbf{x}^\star\| < \epsilon \text{ for all } t \geq T_\epsilon$$

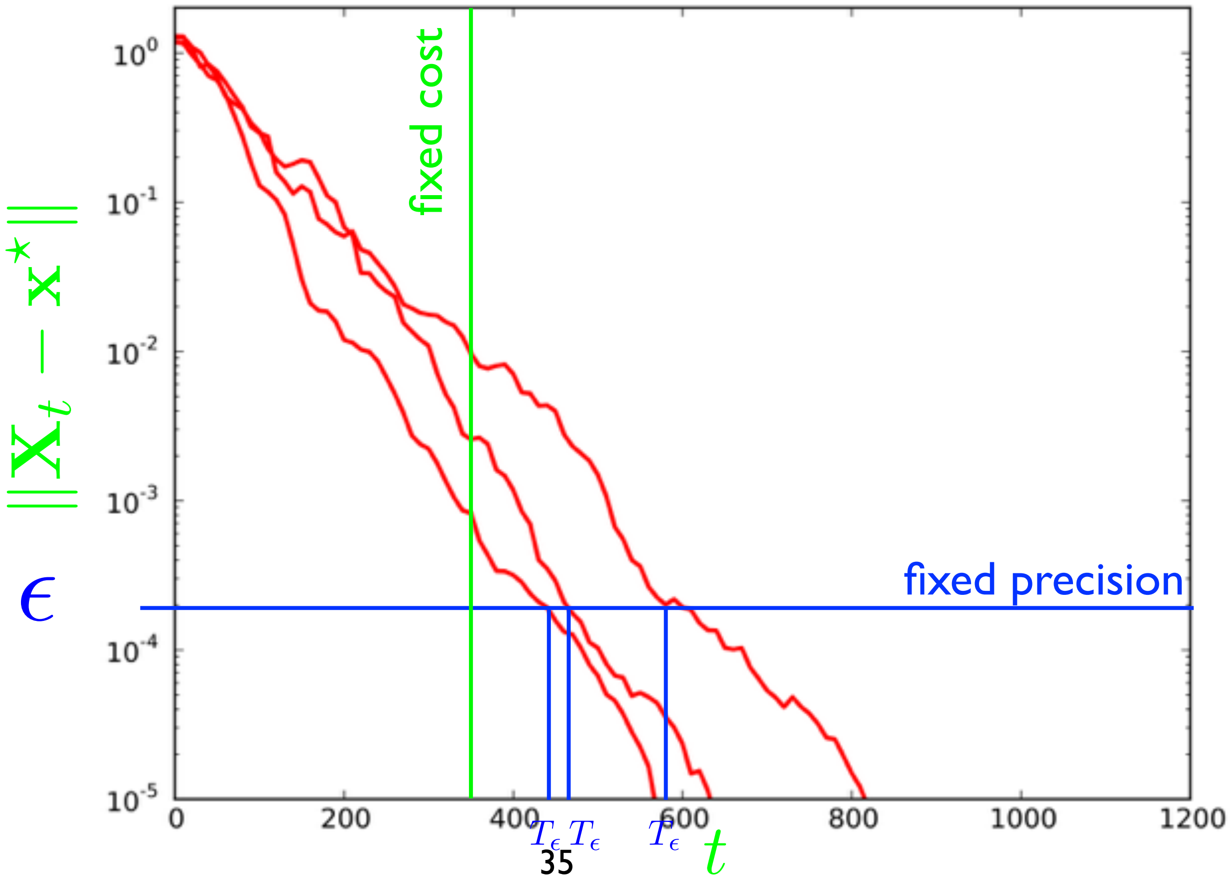
translate that an algorithm approximates the optimum with arbitrary precision

Hitting Time versus Convergence

two side of a coin, measuring

the hitting time T_ϵ given a fixed precision ϵ

the precision $\|\mathbf{X}_t - \mathbf{x}^\star\|$ (or ϵ) given the iteration number t



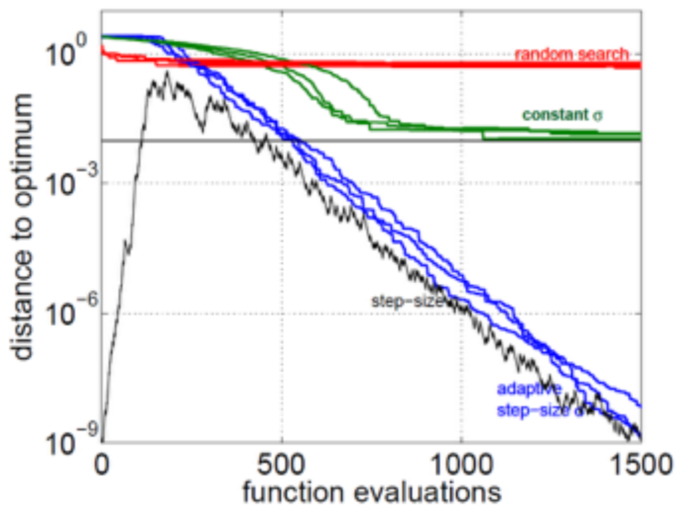
On Convergence alone …

A theoretical convergence result is a "guarantee" that the algorithm will approach the solution in infinite time

$$\lim_{t \to \infty} \mathbf{X}_t = \mathbf{x}^\star$$

often the first/only question investigated about an optimization algorithm

But a convergence result alone is pretty meaningless in practice as it does not tell how fast the algorithm converges



need to quantify how fast the optimum is approached

Quantifying How Fast the Optimum is Approached

For a fixed dimension

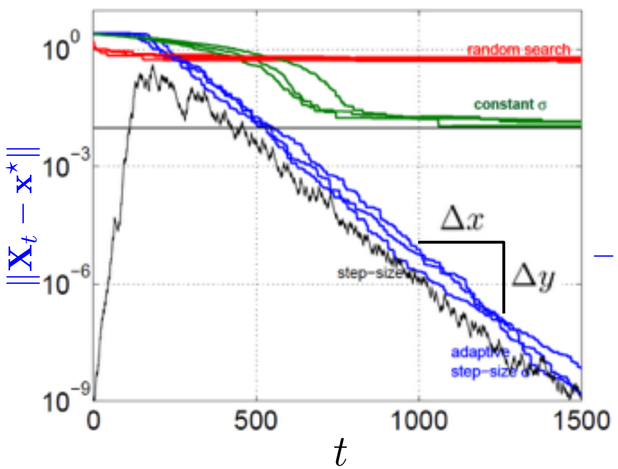
convergence speed of $\mathbf{X}_t$ towards $\mathbf{x}^\star$ ⟷ dependency in ϵ of T_ϵ find $\epsilon \mapsto \tau(\epsilon, n)$

Scaling wrt the dimension

dependency of convergence rate wrt n ⟷ find $n \mapsto \tau(\epsilon, n)$

Compromises to obtain such results:
asymptotic in n, in epsilon / t

Linear Convergence



$$-\frac{c}{n} = \frac{\Delta y}{\Delta x}$$

$$\frac{\|\mathbf{X}_{t+1} - \mathbf{x}^\star\|}{\|\mathbf{X}_t - \mathbf{x}^\star\|} \approx \exp\left(-\frac{c}{n}\right)$$

$$\log \frac{\|\mathbf{X}_{t+1} - \mathbf{x}^\star\|}{\|\mathbf{X}_t - \mathbf{x}^\star\|} \approx -\frac{c}{n}$$

$$\log \frac{\|\mathbf{X}_t - \mathbf{x}^\star\|}{\|\mathbf{X}_0 - \mathbf{x}^\star\|} \approx -\frac{c}{n} t$$

Different formal statements (not exactly equivalent)

almost surely

$$\lim_{t\to\infty} \frac{1}{t} \log \frac{\|\mathbf{X}_t - \mathbf{x}^\star\|}{\|\mathbf{X}_0 - \mathbf{x}^\star\|} = -\frac{c}{n}$$

in expectation

$$\frac{\mathbb{E}\left[\|\mathbf{X}_{t+1} - \mathbf{x}^\star\|\right]}{\mathbb{E}\left[\|\mathbf{X}_t - \mathbf{x}^\star\|\right]} = \exp\left(-\frac{c}{n}\right)$$

$$\mathbb{E} \log \frac{\|\mathbf{X}_{t+1} - \mathbf{x}^\star\|}{\|\mathbf{X}_t - \mathbf{x}^\star\|} = -\frac{c}{n}$$

Connection with Hitting Time formulation

$$T_\epsilon \approx \frac{n}{c} \log \frac{\epsilon_0}{\epsilon}$$

Convergence Rates - Hitting time - Wrap up

	Rate of convergence	Hitting time scaling
Pure Random Search **(1+1)-ES constant step-size**	$\frac{1}{t} \log \frac{\|\mathbf{X}_t - \mathbf{x}^\star\|}{\|\mathbf{X}_0 - \mathbf{x}^\star\|} \approx -\frac{1}{n} \frac{\log(t)}{t}$	$\left(\frac{\epsilon_0}{\epsilon}\right)^n$
Linear Convergence (fixed n) **+** **Linear dependence wrt n**	$\mathbb{E}\left[\|\mathbf{X}_t - \mathbf{x}^\star\|\right] = \exp\left(-\frac{c}{n}\right)^t \mathbb{E}\left[\|\mathbf{X}_0 - \mathbf{x}^\star\|\right]$ $\lim_{t\to\infty} \frac{1}{t} \log \frac{\|\mathbf{X}_t - \mathbf{x}^\star\|}{\|\mathbf{X}_0 - \mathbf{x}^\star\|} = -\frac{c}{n}$	$\frac{n}{c} \log \frac{\epsilon_0}{\epsilon}$