New perspectives in stochastic derivative free optimization

Anne Auger

INRIA – Saclay Ile-de-France

http://www.lri.fr/~auger/

Workshop on Advanced Methods and Perspectives in Nonlinear optimization and control

Toulouse, Febuary 3-5

Problem statement: Black-Box-Optimization

Minimize an objective function

$$f : \mathcal{X} \subset \mathbb{R}^d \mapsto \mathbb{R}, \mathbf{x} \mapsto f(\mathbf{x})$$

Black-Box scenario

gradient not available or not useful

Derivative Free Optimization (DFO)

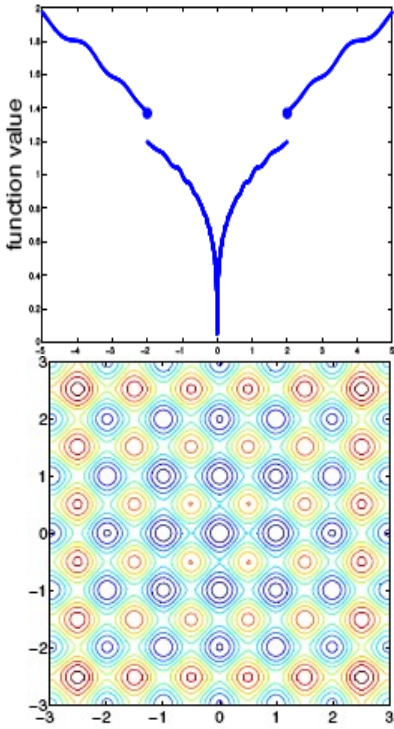
Search cost: number of function evaluations

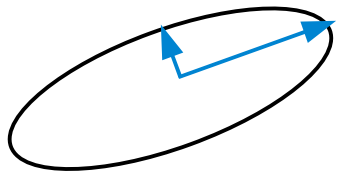
What makes a function difficult to solve?

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

  (considerably) larger than 3
- non-separability

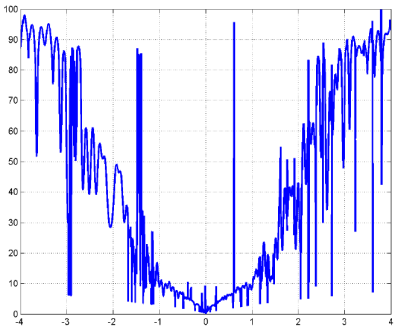
  dependencies between variables
- Ill-conditioning





- ruggedness

  non-smooth, discontinuous, multi-modal, and/or noisy



Stochastic algorithms: try to deal with any of those difficulties

Overview

- Stochastic optimization algorithms
  - Step-size adaptive ESs
  - Covariance Matrix Adaptation ESs (CMA-ES)
- Benchmarking stochastic and deterministic DFOs
- Theoretical convergence results

A stochastic black-box-search template

Initialize distribution parameter θ, set population size λ

While not terminate

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

Everything depends on the definition of P, θ and F_θ

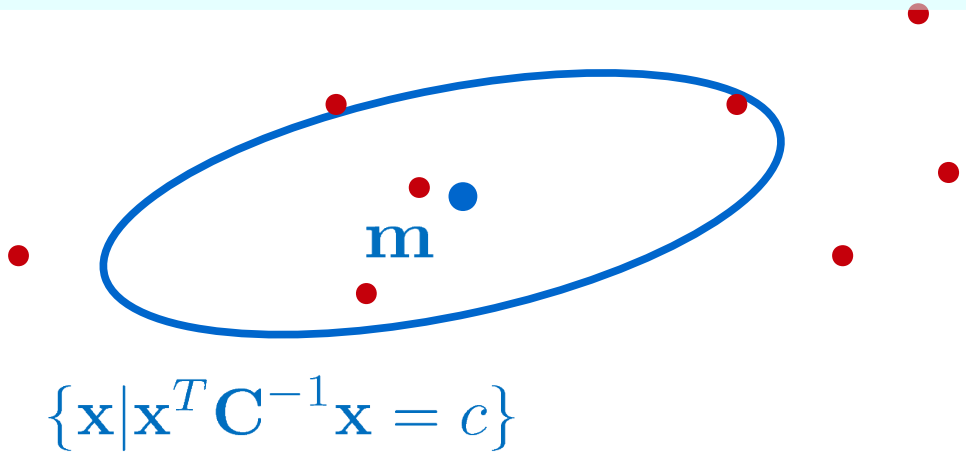
Often, P implicitly defined via operators (mutation, recombination, selection) on population $\mathbf{x}_1, \ldots, \mathbf{x}_\lambda$

Evolution Strategies (ES)

[Rechenberg, Schwefel, 70's]

New search points are normally distributed:

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



$\mathbf{m} \in \mathbb{R}^d$: mean vector, represents favorite solution

$\sigma \in \mathbb{R}_+$: step-size, controls the step-length

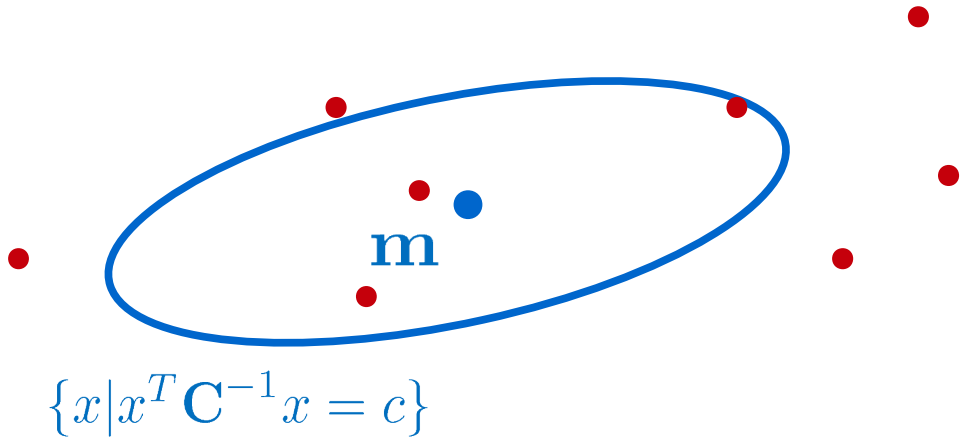
$\mathbf{C} \in \mathbb{R}^{d \times d}$: symmetric positive definite, covariance matrix, determines the shape of the distribution ellipsoid

Evolution Strategies (ES)

[Rechenberg, Schwefel, 70's]

New search points are normally distributed:

$$x_i \sim \mathbf{m} + \sigma \mathcal{N}_i(\mathbf{0}, \mathbf{C}) \text{ for } i = 1, \ldots, \lambda$$



How to update $\mathbf{m}$, σ, $\mathbf{C}$?

ES: typical update of mean value

$(\mu/\mu_w, \lambda)$-ES

n iteration index, X_n: mean value

Sample λ solutions:

$$X_n^i = X_n + \sigma_n N_n^i, \;\; N_n^i \sim \mathcal{N}(\mathbf{0}, \mathbf{C_n})$$

Evaluate and rank:

$$f(X_n^{1:\lambda}) \leq \ldots \leq f(X_n^{\mu:\lambda}) \leq \ldots \leq f(X_n^{\lambda:\lambda})$$

Recombine the μ best:

$$X_{n+1} = \sum_{i=1}^{\mu} w_i X_n^{i:\mu} = X_n + \underbrace{\sum_{i=1}^{\mu} w_i N_n^{i:\lambda}}$$

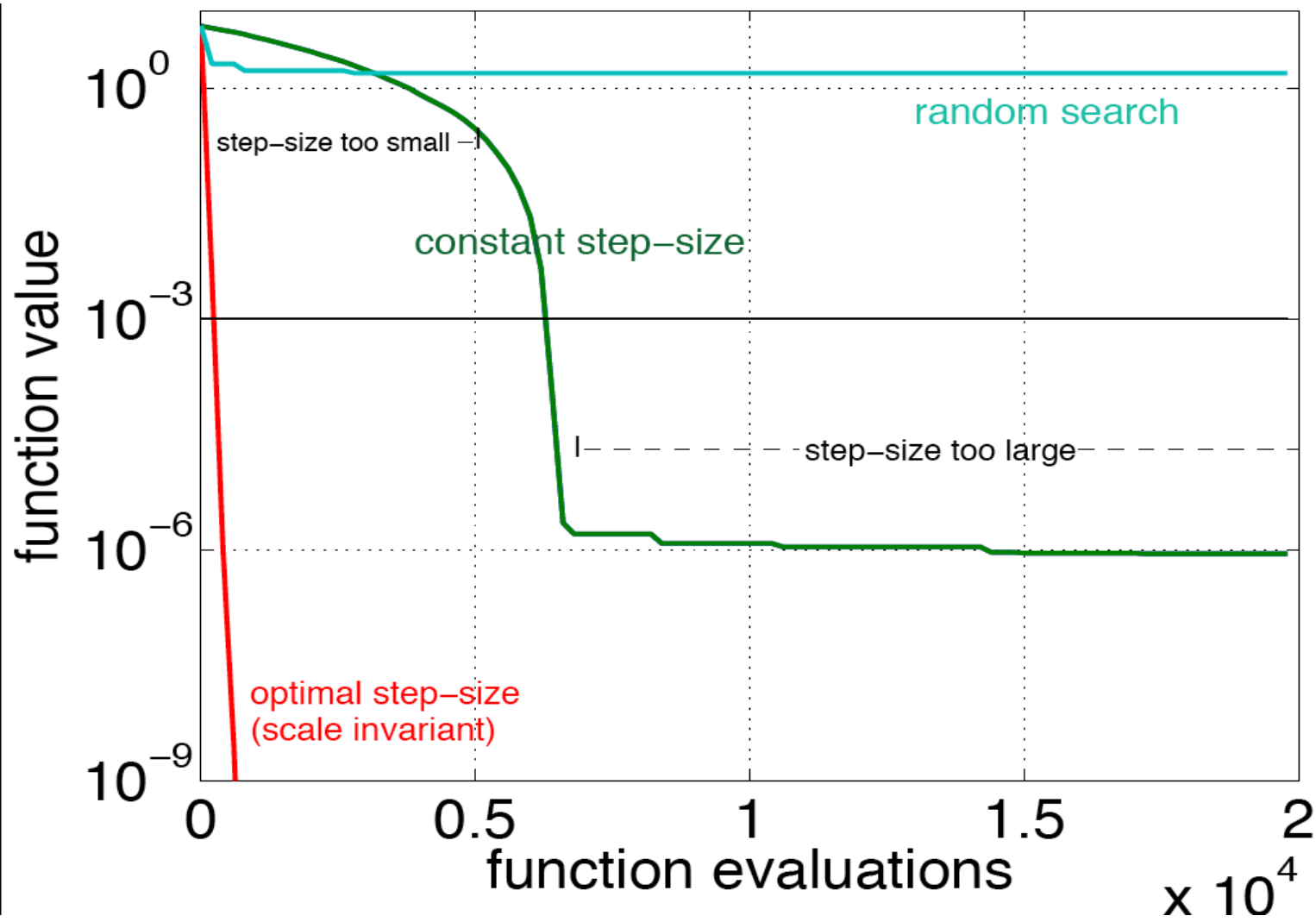
"descent direction"

where $w_1 \geq \ldots \geq w_\mu$ and $\sum_{i=1}^{\mu} w_i = 1$

Update $(\sigma_n, \mathbf{C_n})$

$$(\sigma_{n+1}, \mathbf{C_{n+1}}) = Update(\sigma_n, \mathbf{C_n}, X_n^{1:\lambda}, \ldots, X_n^{\lambda:\lambda})$$

Why step-size control?



$\mathbf{C_n} = \mathbf{I}$

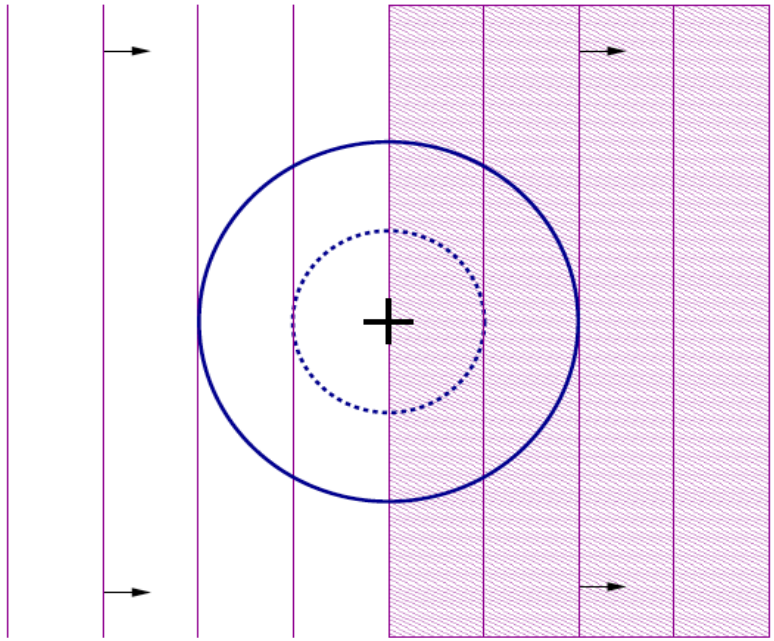
$f(\mathbf{x}) = \sum_{i=1}^{d} x_i^2$

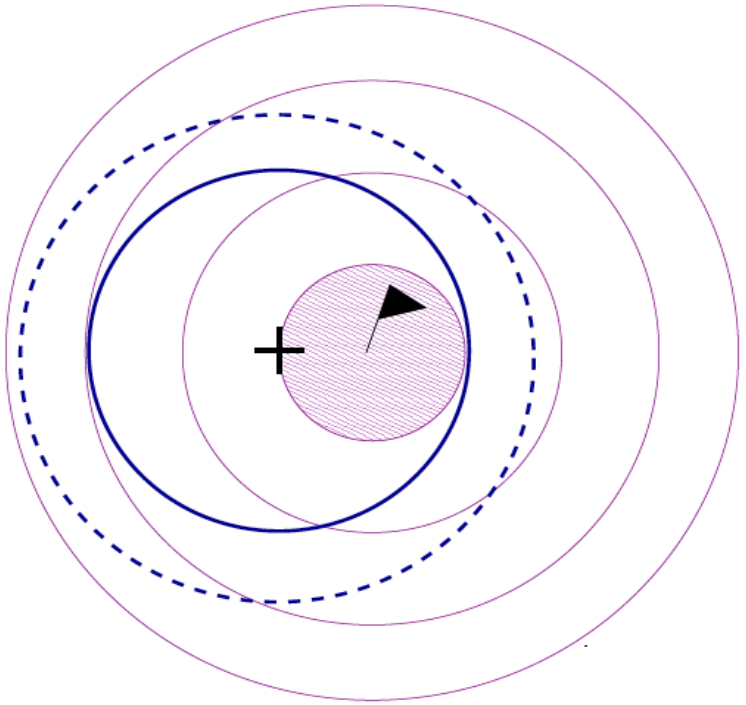
in $[-0.2, 0.8]^d$

for $d = 10$

One-fifth success rule

[Rechenberg 70]





increase σ_n

probability of success (p_s)

$\frac{1}{2}$

$1/5$

decrease σ_n

probability of success (p_s)

"too small"

One-fifth success rule – the equations

$$p_s = \frac{\#\text{ of successful candidate solutions}}{\#\text{ sampled solutions}} \quad \text{(per iteration)}$$

$$\sigma_{n+1} = \sigma_n \exp\left(\frac{1}{3}\frac{p_s - p_{\text{target}}}{1 - p_{\text{target}}}\right)$$

Increase σ_n if $p_s > p_{\text{target}}$

Decrease σ_n if $p_s < p_{\text{target}}$

$(1+1)$-ES

Sample 1 solution: $\tilde{X}_n = X_n + \sigma_n N_n, \; N_n \sim \mathcal{N}(\mathbf{0}, I)$

IF $f(\tilde{X}_n) \leq f(X_n)$

$$p_s = 1$$

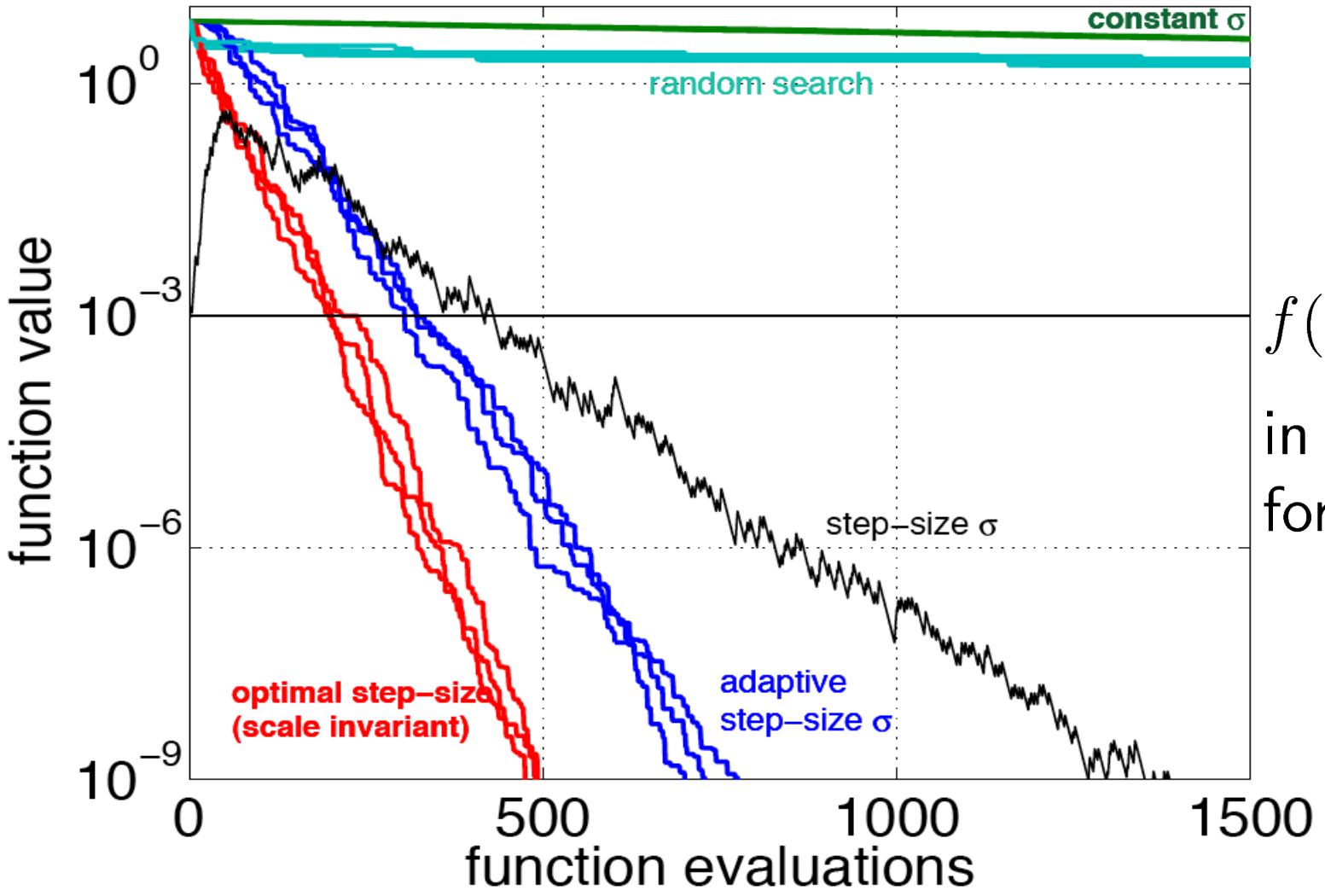
$$X_{n+1} = \tilde{X}_n, \; \sigma_{n+1} = \sigma_n \exp(1/3)$$

ELSE

$$p_s = 0$$

$$X_{n+1} = X_n, \; \sigma_{n+1} = \sigma_n / \exp(1/3)^{1/4}$$

Step-size adaptation → linear convergence



$f(\mathbf{x}) = \sum_{i=1}^{d} x_i^2$

in $[-0.2, 0.8]^d$

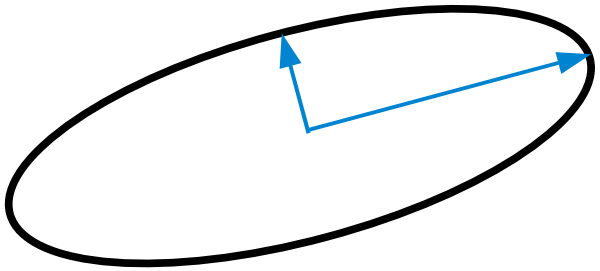
for $d = 10$

Why covariance matrix adaptation?

Ill-conditioned problems:

Consider a convex quadratic function $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T H \mathbf{x}$, H symmetric, positive, definite.

$$cond(f) = cond(H)$$



ill-conditioned problem: $cond(f)$ up to 10^{10}

sampling with isotropic distribution $\mathbf{C_n} = I$ inefficient

What do we want to achieve?

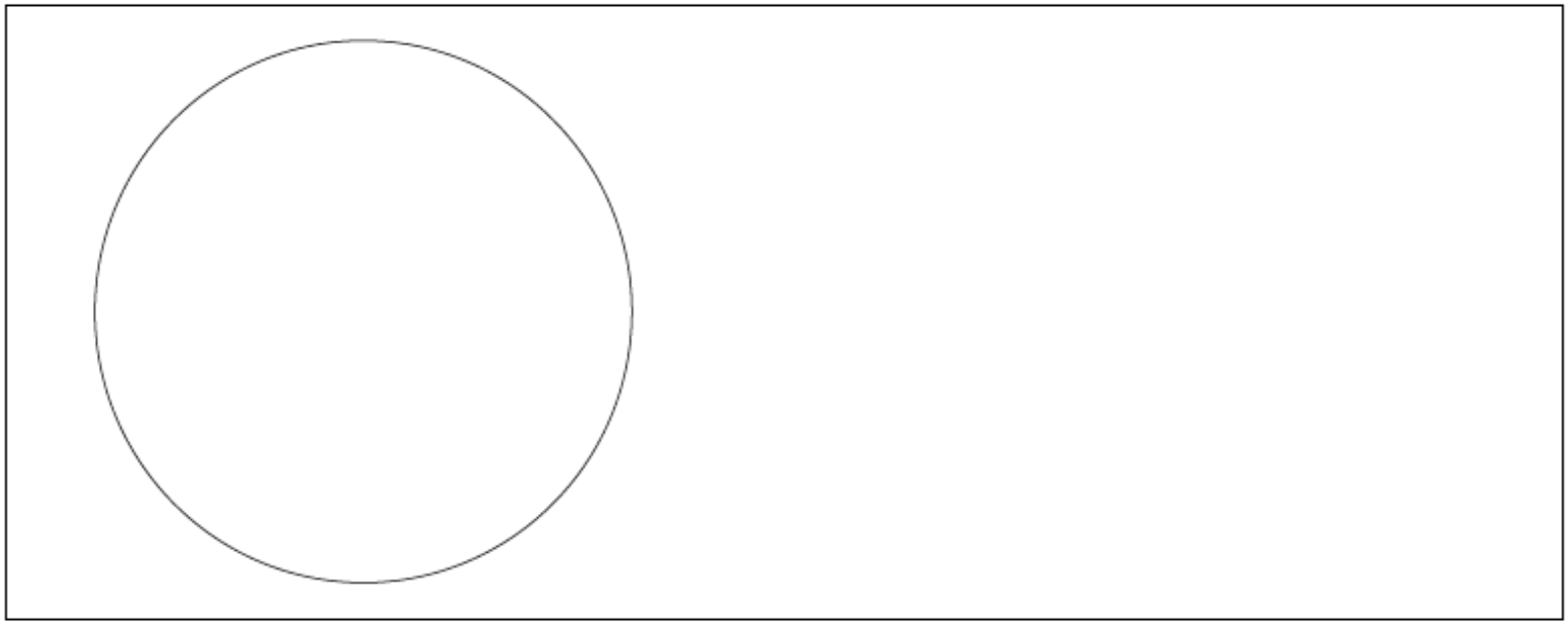
align lines of equal density with level sets of the functions

$$\mathbf{C_n} \propto H^{-1}$$

Rank-one update

[N. Hansen, A Ostermeier 2001]

$$X_1 = X_0 + \sigma_0 \mathbf{y}_w, \quad \mathbf{y}_w = \sum_{i=1}^{\mu} w_i N_0^{i:\lambda}, \quad N_0^i \sim \mathcal{N}_i(\mathbf{0}, \mathbf{C}_0)$$



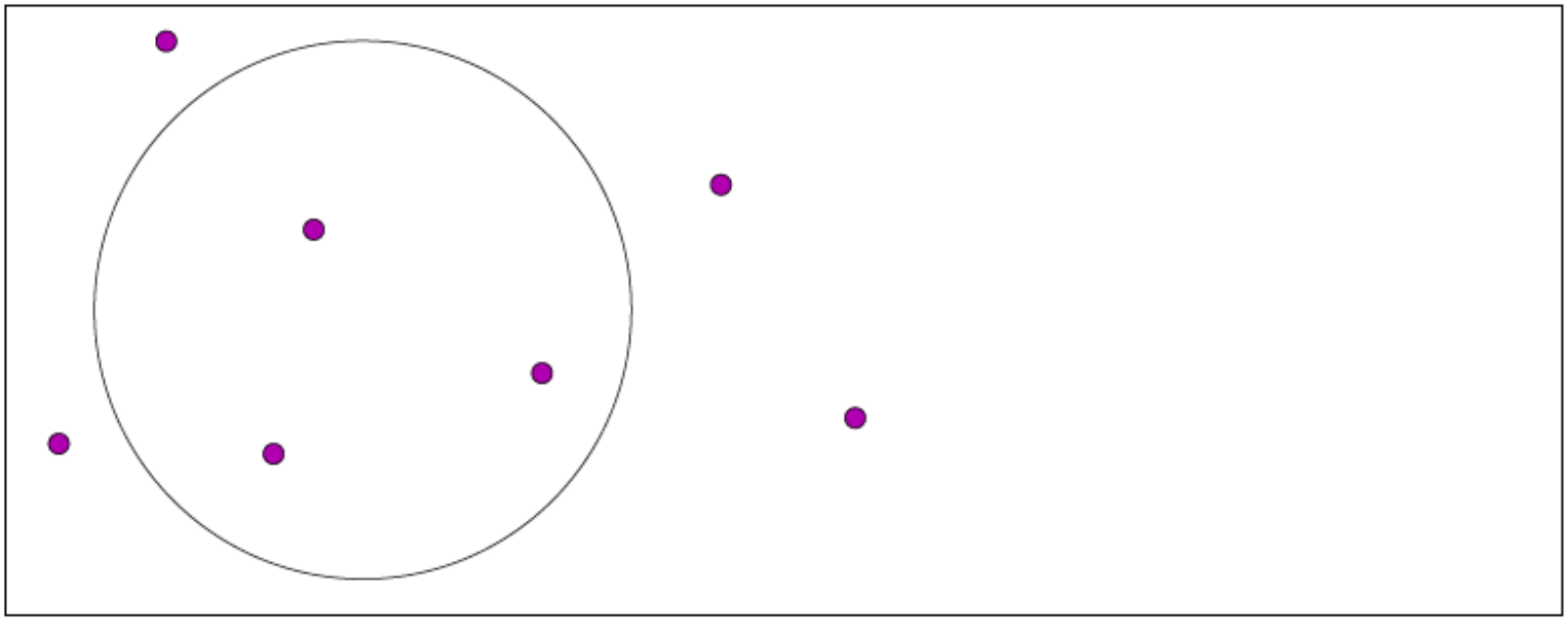
initial distribution, $\mathbf{C}_0 = \mathbf{I}$

...equations

Rank-one update

[N. Hansen, A Ostermeier 2001]

$$X_1 = X_0 + \sigma_0 \boldsymbol{y}_w, \quad \boldsymbol{y}_w = \sum_{i=1}^{\mu} w_i N_0^{i:\lambda}, \quad N_0^i \sim \mathcal{N}_i(\mathbf{0}, \mathbf{C}_0)$$



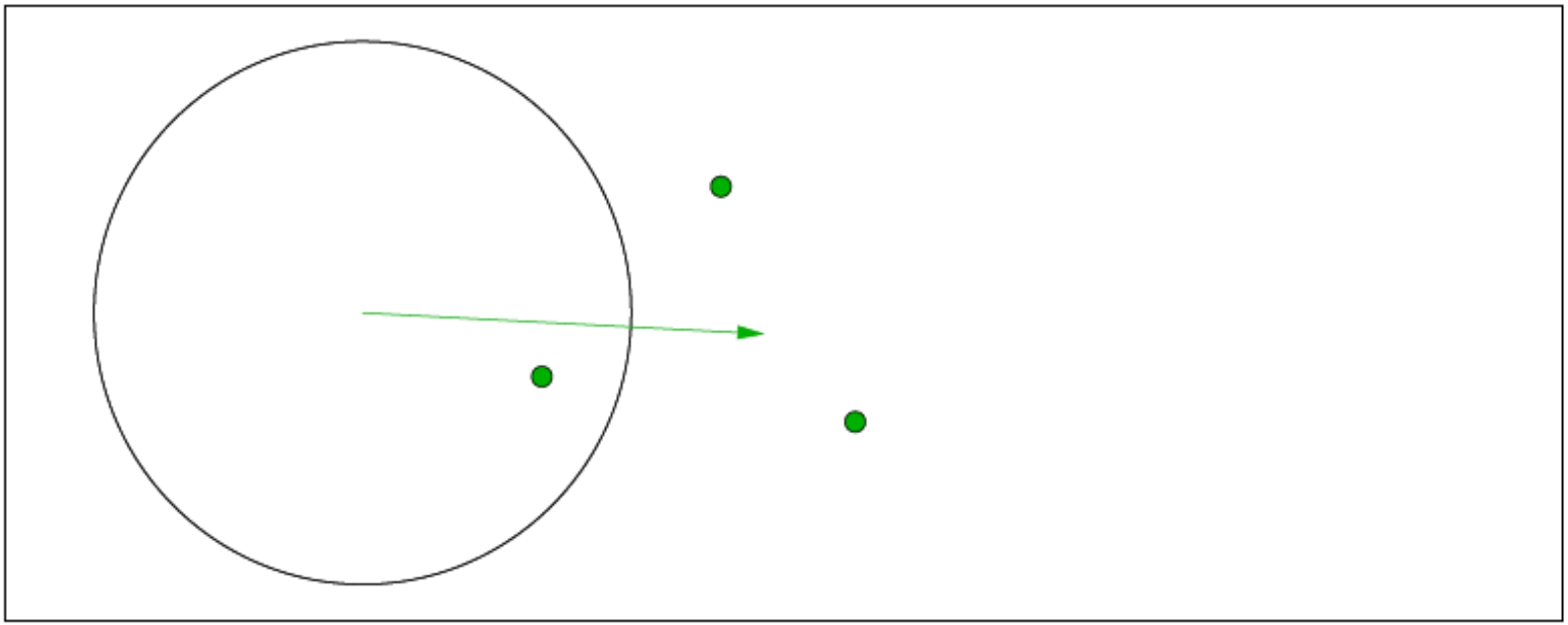
initial distribution, $\mathbf{C}_0 = \mathbf{I}$

...equations

Rank-one update

[N. Hansen, A Ostermeier 2001]

$$X_1 = X_0 + \sigma_0 \boldsymbol{y}_w, \quad \boldsymbol{y}_w = \sum_{i=1}^{\mu} w_i N_0^{i:\lambda}, \quad N_0^i \sim \mathcal{N}_i(\mathbf{0}, \mathbf{C}_0)$$



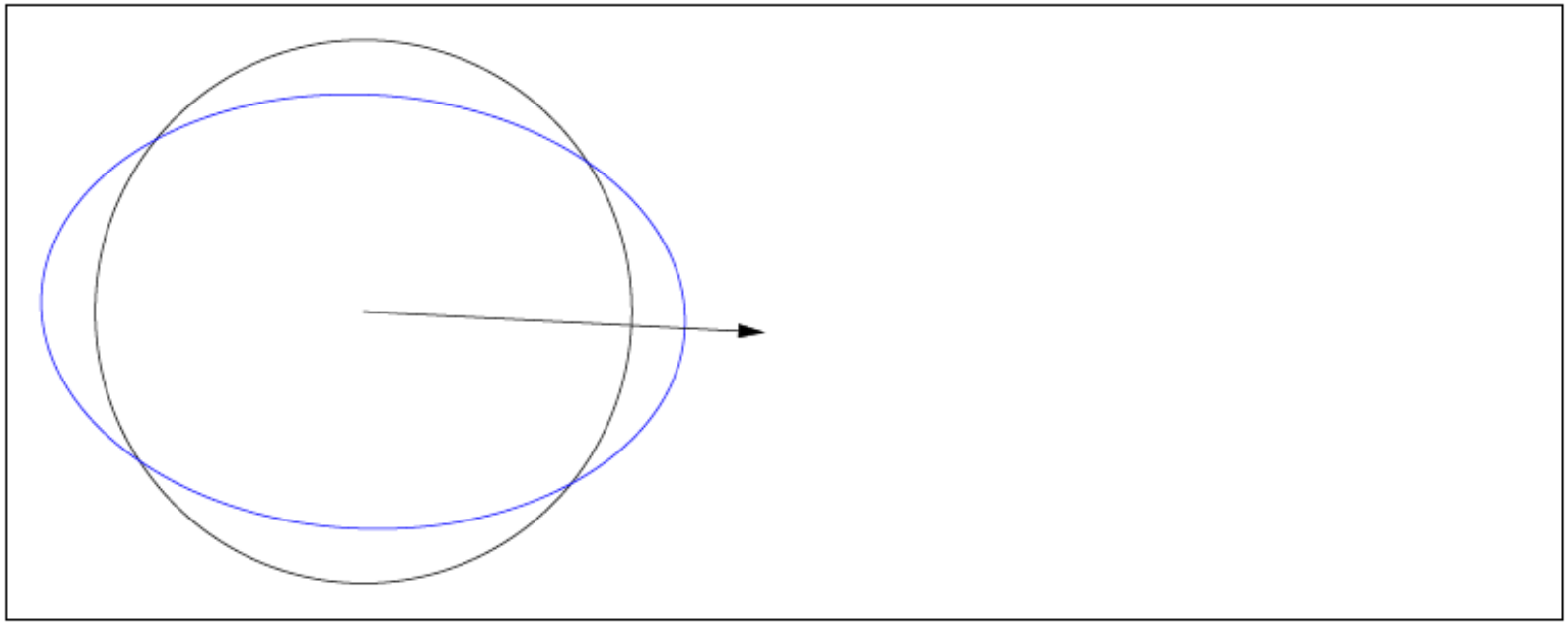
$\boldsymbol{y}_w$, movement of the population mean $X_0 \to X_1$ (disregarding σ_0)

...equations

Rank-one update

[N. Hansen, A Ostermeier 2001]

$$X_1 = X_0 + \sigma_0 \boldsymbol{y}_w, \quad \boldsymbol{y}_w = \sum_{i=1}^{\mu} w_i N_0^{i:\lambda}, \quad N_0^i \sim \mathcal{N}_i(\mathbf{0}, \mathbf{C}_0)$$



mixture of distribution $\mathbf{C}_0$ and step $\boldsymbol{y}_w$,

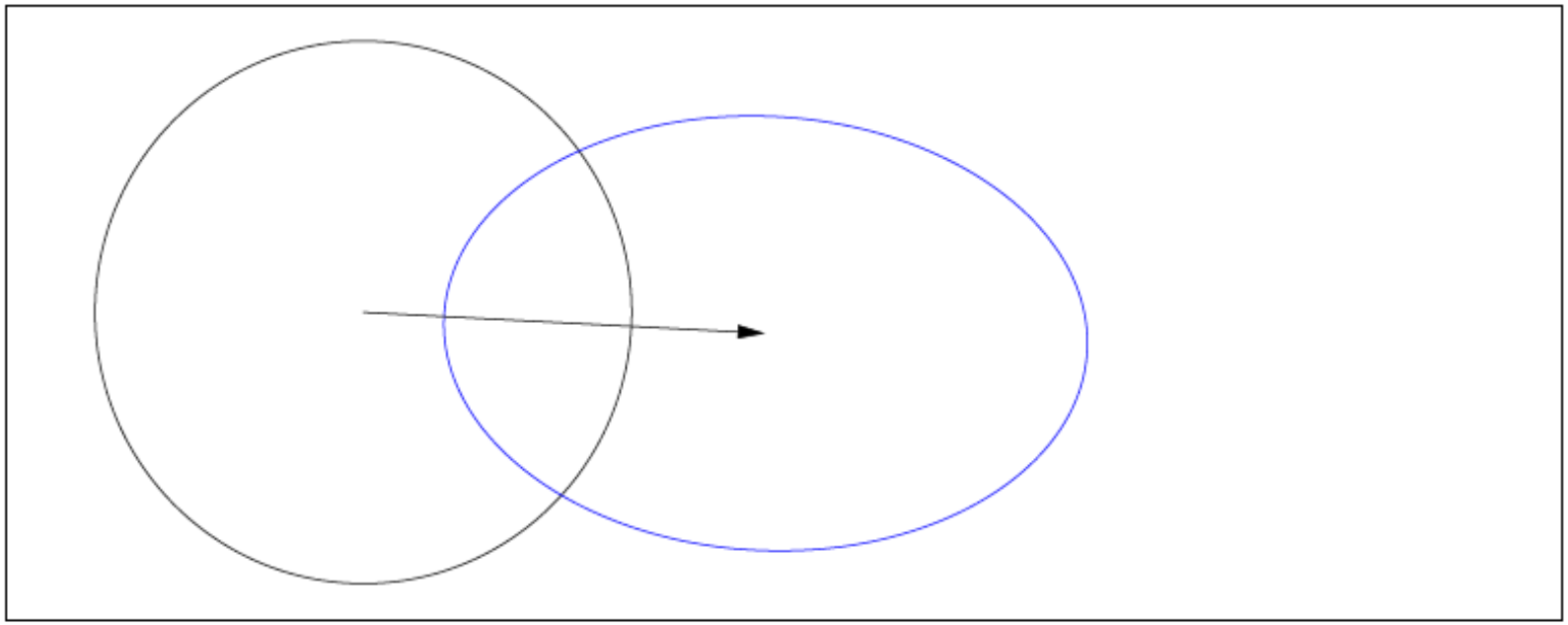
$\mathbf{C}_1 = 0.8 \times \mathbf{C}_0 + 0.2 \times \boldsymbol{y}_w \boldsymbol{y}_w^{\mathrm{T}}$

...equations

Rank-one update

[N. Hansen, A Ostermeier 2001]

$$X_2 = X_1 + \sigma_1 \boldsymbol{y}_w, \quad \boldsymbol{y}_w = \sum_{i=1}^{\mu} w_i N_1^{i:\lambda}, \quad N_1^i \sim \mathcal{N}_i(\mathbf{0}, \mathbf{C}_1)$$



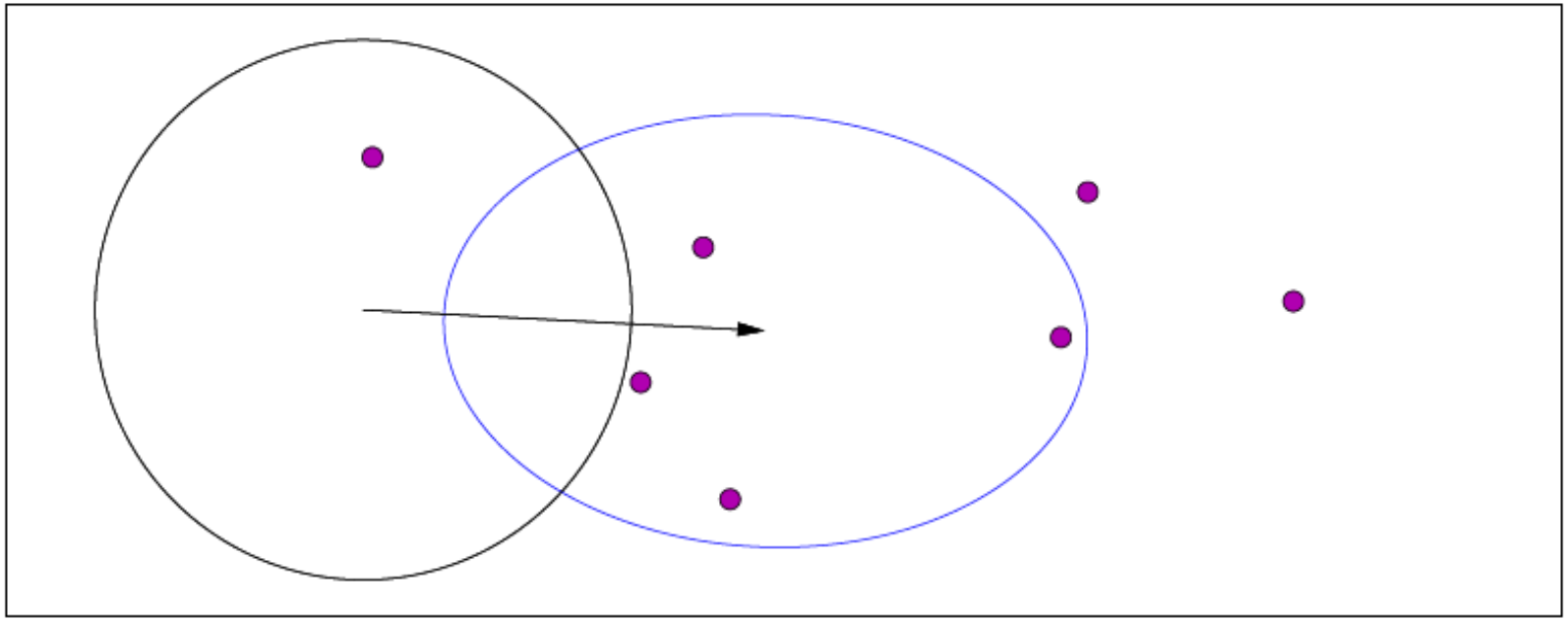
new distribution

...equations

Rank-one update

[N. Hansen, A Ostermeier 2001]

$$X_2 = X_1 + \sigma_1 \boldsymbol{y}_w, \quad \boldsymbol{y}_w = \sum_{i=1}^{\mu} w_i N_1^{i:\lambda}, \quad N_1^i \sim \mathcal{N}_i(\mathbf{0}, \mathbf{C}_1)$$



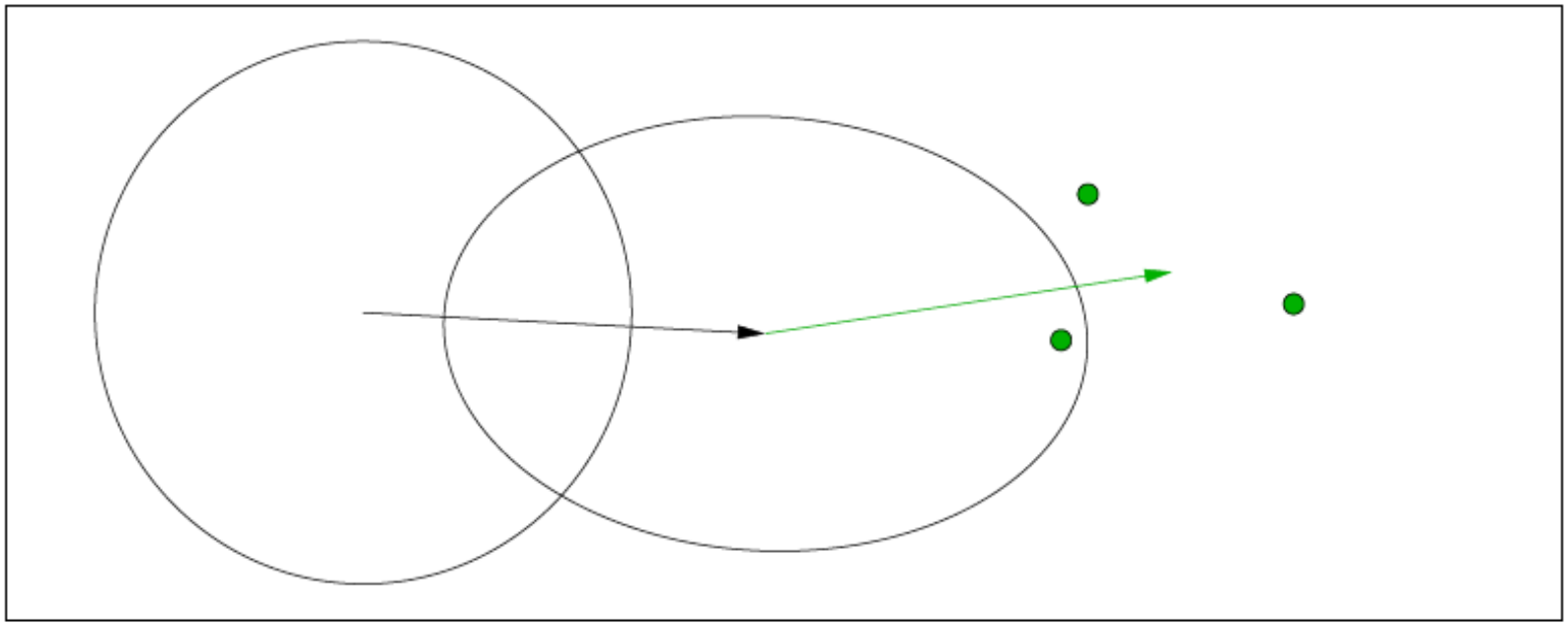
new distribution

...equations

Rank-one update

[N. Hansen, A Ostermeier 2001]

$$X_2 = X_1 + \sigma_1 \boldsymbol{y}_w, \quad \boldsymbol{y}_w = \sum_{i=1}^{\mu} w_i N_1^{i:\lambda}, \quad N_1^i \sim \mathcal{N}_i(\mathbf{0}, \mathbf{C}_1)$$



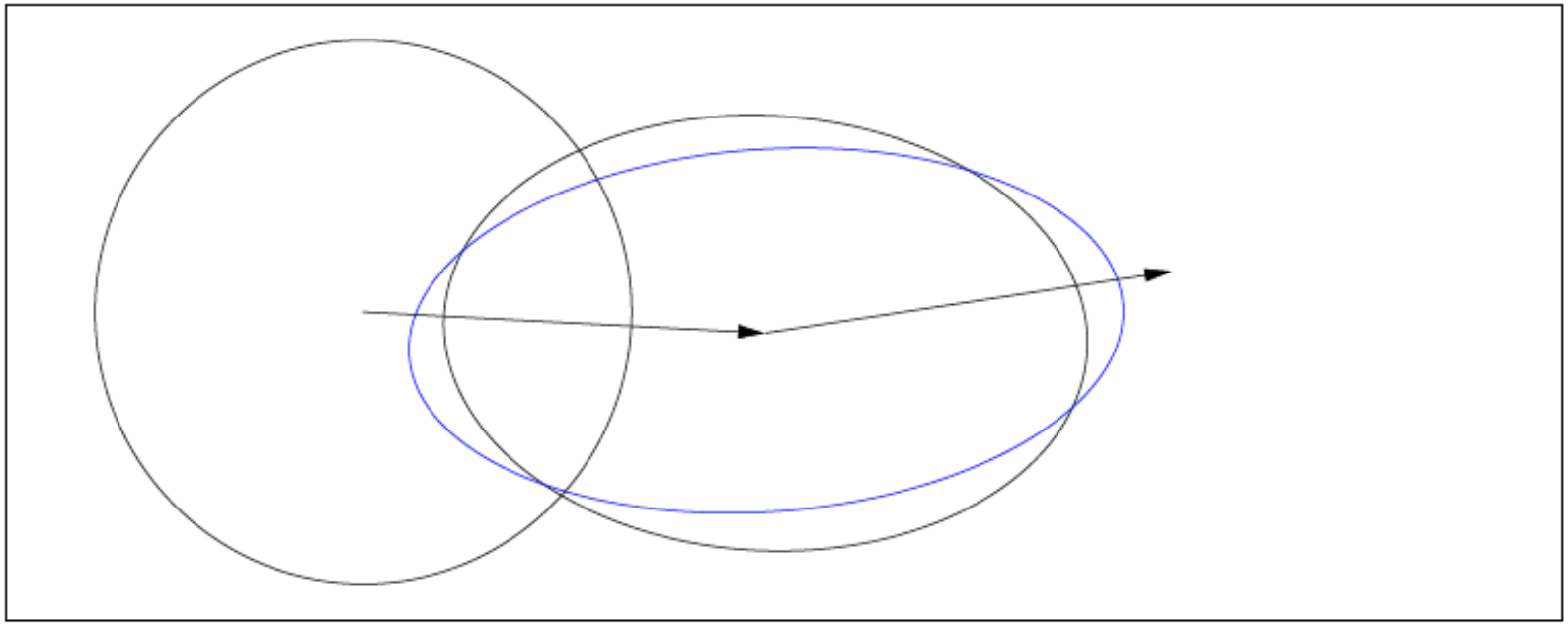
movement of the population mean

...equations

Rank-one update

[N. Hansen, A Ostermeier 2001]

$$X_2 = X_1 + \sigma_1 \boldsymbol{y}_w, \quad \boldsymbol{y}_w = \sum_{i=1}^{\mu} w_i N_1^{i:\lambda}, \quad N_1^i \sim \mathcal{N}_i(\mathbf{0}, \mathbf{C}_1)$$



mixture of distribution $\mathbf{C}_1$ and step $\boldsymbol{y}_w$,

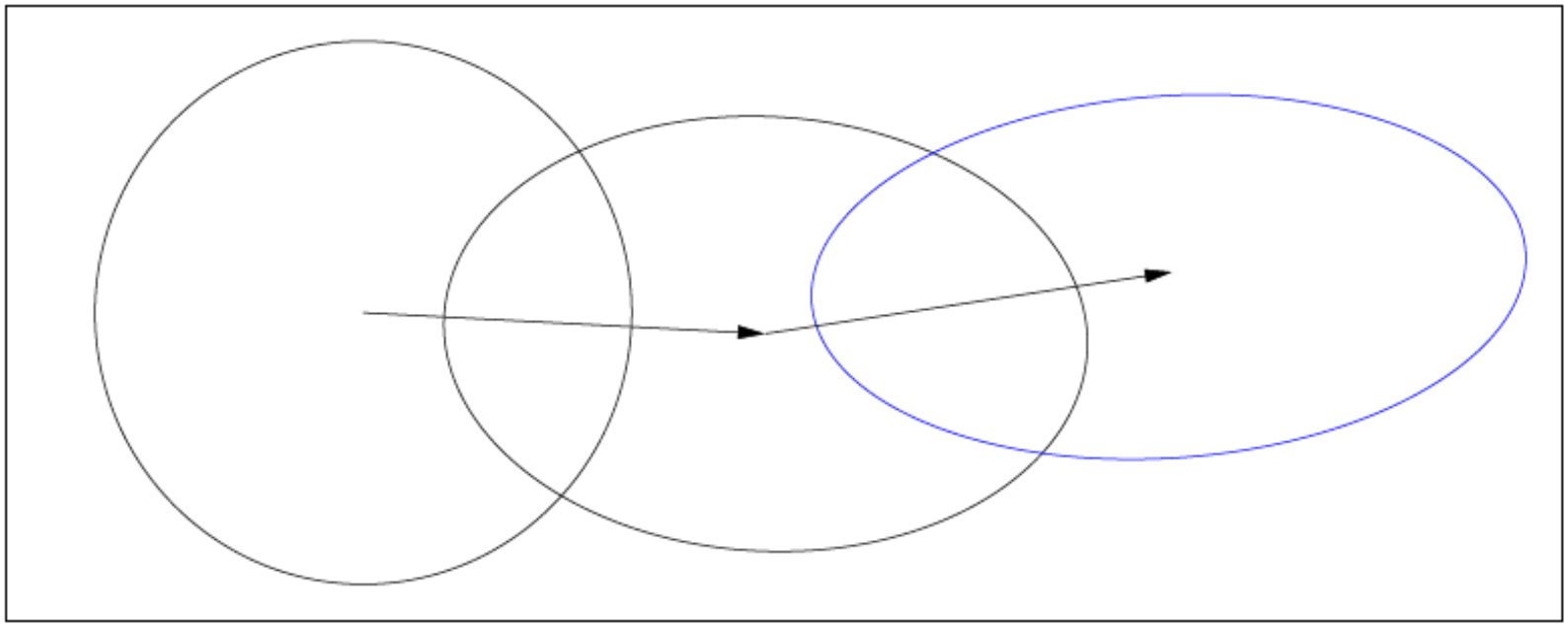
$$\mathbf{C}_2 = 0.8 \times \mathbf{C}_1 + 0.2 \times \boldsymbol{y}_w \boldsymbol{y}_w^{\mathrm{T}}$$

...equations

Rank-one update

[N. Hansen, A Ostermeier 2001]

$$X_2 = X_1 + \sigma_1 \boldsymbol{y}_w, \quad \boldsymbol{y}_w = \sum_{i=1}^{\mu} w_i N_1^{i:\lambda}, \quad N_1^i \sim \mathcal{N}_i(\mathbf{0}, \mathbf{C}_1)$$



new distribution,

$$\mathbf{C}_2 = 0.8 \times \mathbf{C}_1 + 0.2 \times \boldsymbol{y}_w \boldsymbol{y}_w^{\mathrm{T}}$$

the ruling principle: the adaptation increases the likelyhood of successful steps, $\boldsymbol{y}_w$, to appear again

…equations

Rank-one update

Initialize $X_0 \in \mathbb{R}^d$, and $\mathbf{C_0} = I$, learning rate $c_{\text{cov}} \approx 2/d^2$

$$X_n^i = X_n + \sigma_n y^i, \;\; y^i \sim \mathcal{N}(\mathbf{0}, \mathbf{C_n})$$

$$X_{n+1} = X_n + \sigma_n \mathbf{y}_w \text{ where } \mathbf{y}_w = \sum_{i=1}^{\mu} w_i y^{i:\lambda}$$

$$\mathbf{C_{n+1}} = (1 - c_{\text{cov}})\mathbf{C_n} + c_{\text{cov}} \mu_w \underbrace{\mathbf{y}_w \mathbf{y}_w^T}_{\text{rank one}}$$

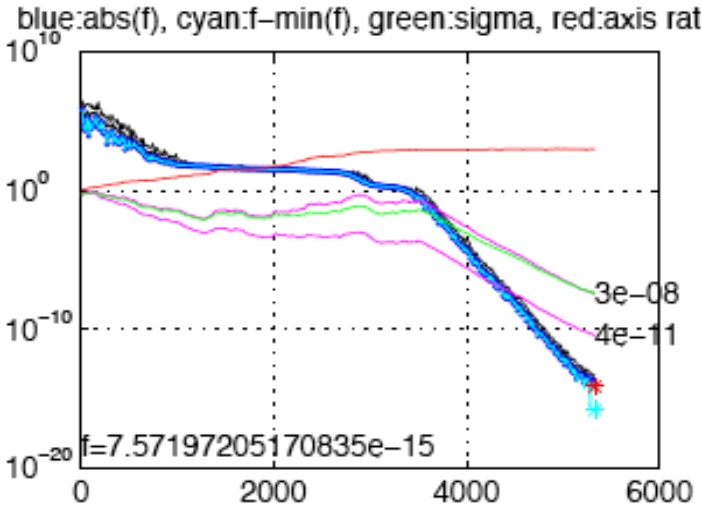
$$\text{where } \mu_w = \frac{1}{\sum_{i=1}^{\mu} w_i^2} \geq 1$$

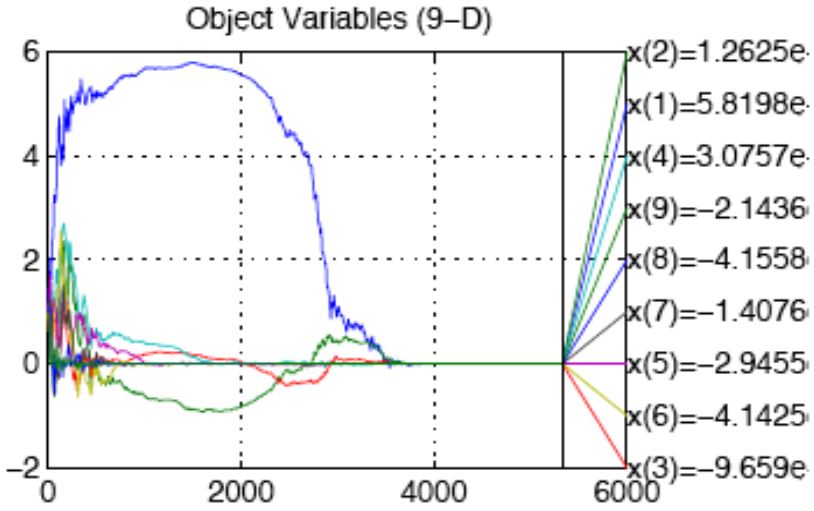
Hansen, N. and A. Ostermeier (2001).
Completely Derandomized Self-Adaptation in Evolution Strategies. Evolutionary Computation,

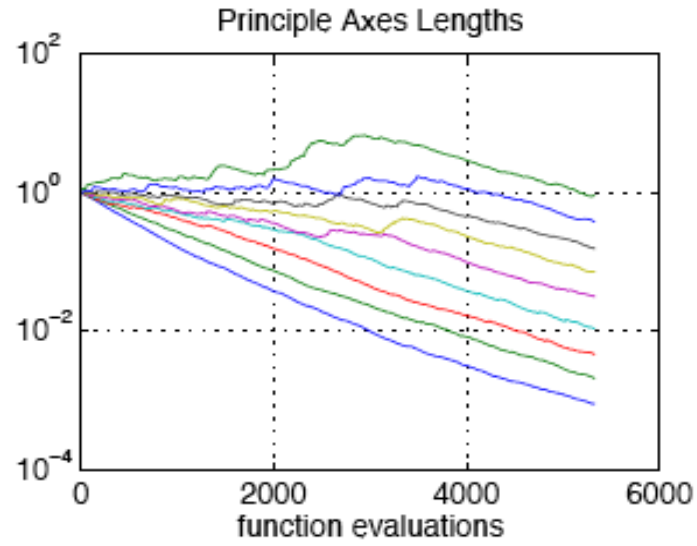
Experimentum crucis (1)

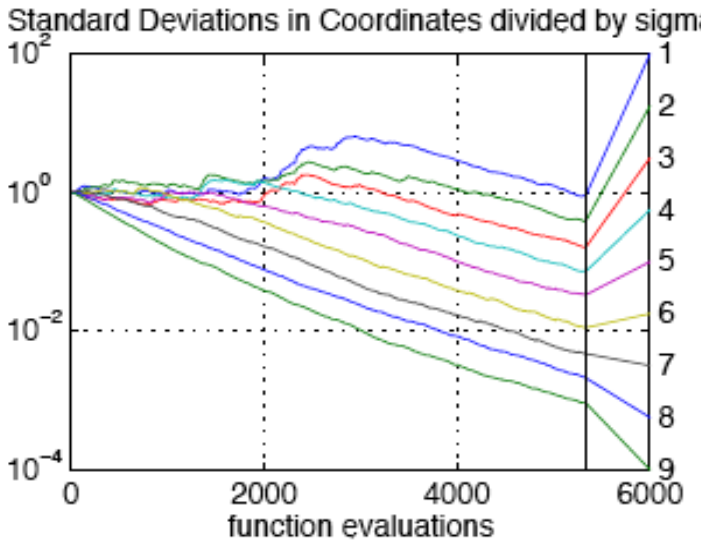
f convex-quadratic, separable

[Code available at www.lri.fr/~hansen/]





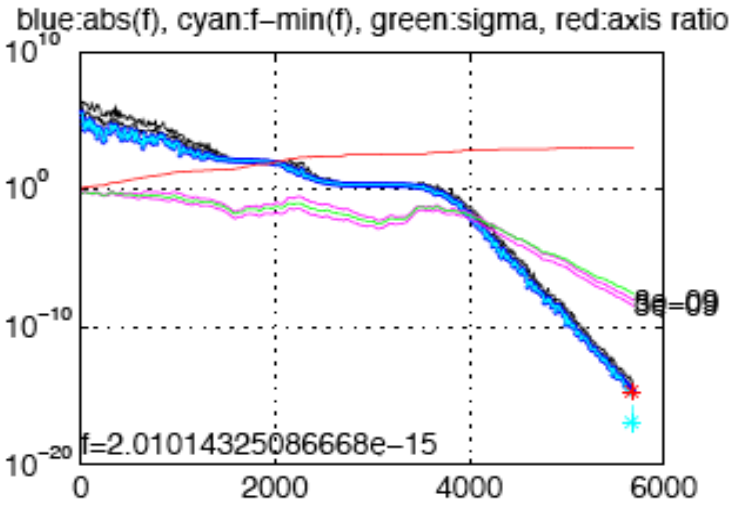


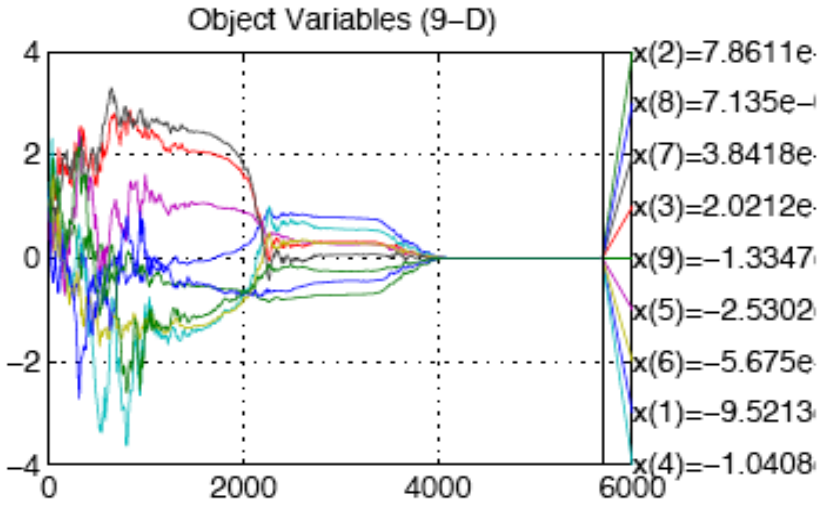


$$f(\mathbf{x}) = \sum_{i=1}^{d} 10^{\alpha \frac{i-1}{d-1}} x_i^2 \, , \alpha = 6$$

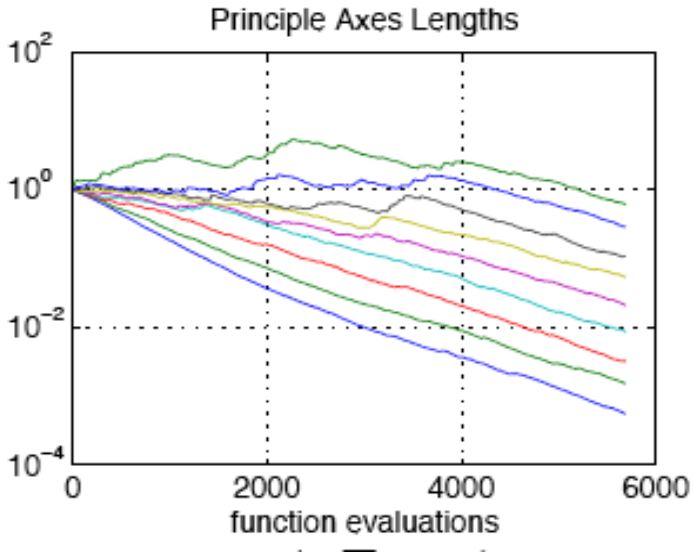
Experimentum crucis (2)

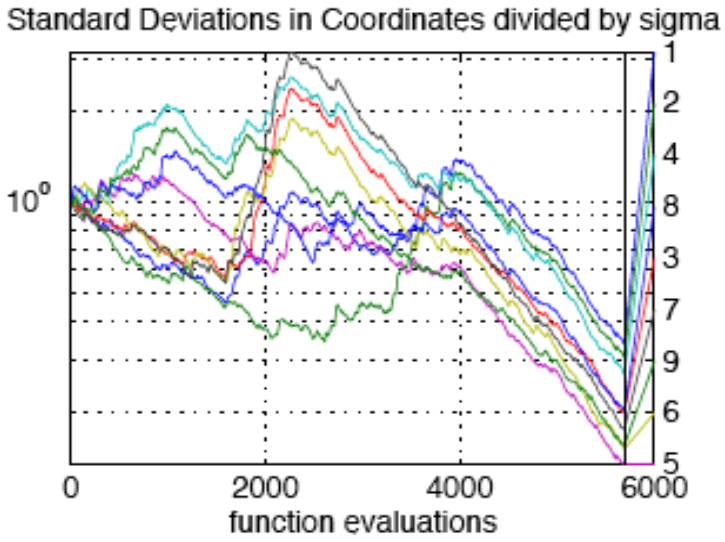
f convex-quadratic, as before but non-separable (rotated)





$\mathbf{C_n} \propto H^{-1}$



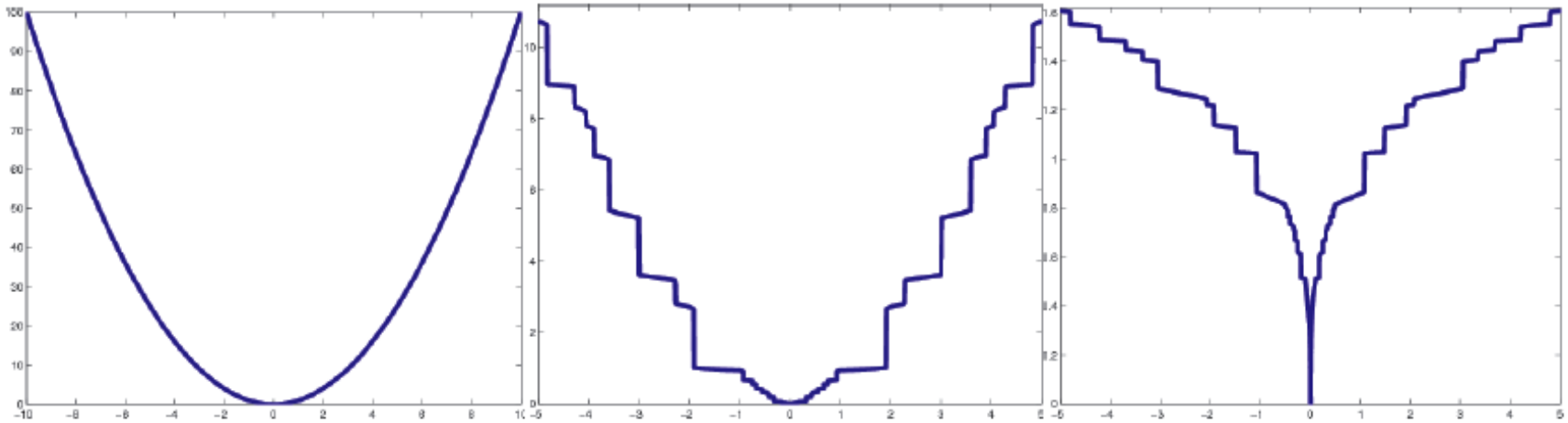


Invariance to Monotonically Increasing Functions

Rank-based algorithms

Update of all parameters uses only the ranks

$$f(\mathbf{x}_{1:\lambda}) \leq f(\mathbf{x}_{2:\lambda}) \leq \ldots \leq f(\mathbf{x}_{\lambda:\lambda})$$



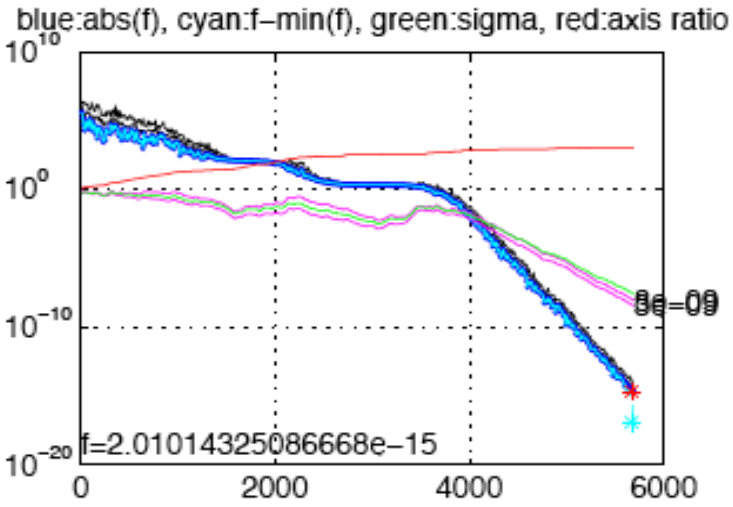
$$g(f(\mathbf{x}_{1:\lambda})) \leq g(f(\mathbf{x}_{2:\lambda})) \leq \ldots \leq g(f(\mathbf{x}_{\lambda:\lambda})) \quad \forall g$$

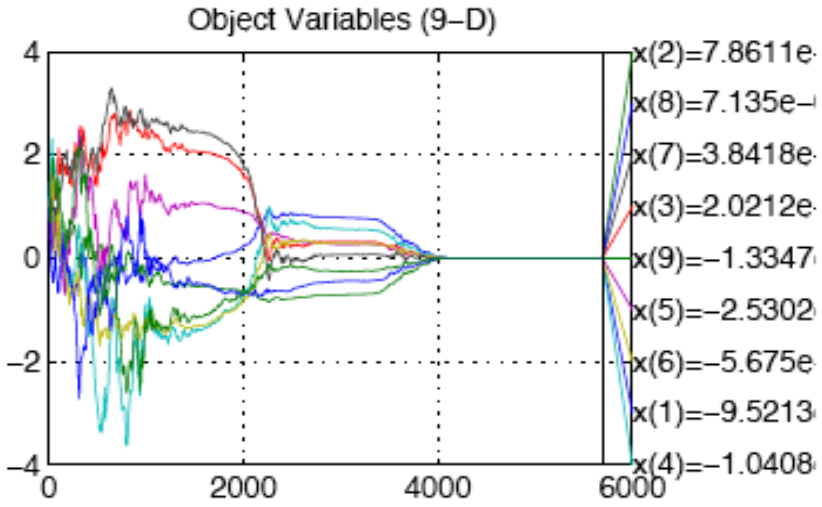
g is strictly monotonically increasing

g preserves ranks

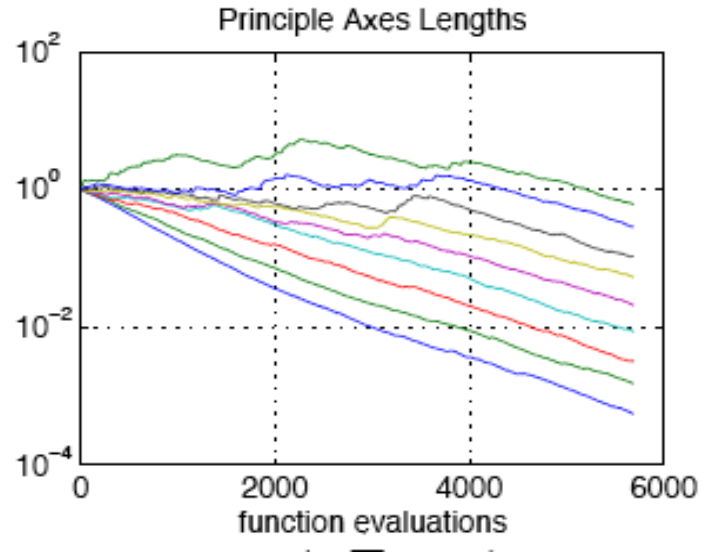
Experimentum crucis (2')

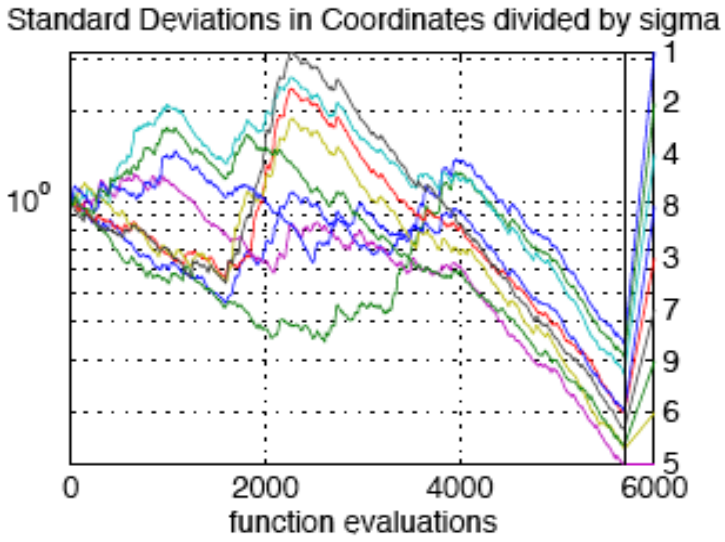
f convex-quadratic, as before but non-separable (rotated)





$\mathbf{C_n} \propto H^{-1}$





$$f(\mathbf{x}) = g(\mathbf{x}^T H \mathbf{x}) \text{ for all } g$$

Overview

- Stochastic optimization algorithms
- Benchmarking stochastic and deterministic DFOs
- Theoretical convergence results

Benchmarking of several DFO

Algorithms tested:

BFGS: gradient estimated by finite differences, Matlab implementation

[Broyden, Fletcher, Goldfarb, Shanno]

NEWUOA (NEW Unconstrained Optimization Algorithm): C-translation by M. Guibert of Fortran Powell code [Powell 2006]

CMA-ES (Covariance Matrix Adaptation ES) [Hansen et al. 2001-2004]

PSO (Particle Swarm Optimization) [Kennedy, Eberhart 1995]

DE (Differential Evolution) [Storn-Price 1997]

Empirical comparisons of several derivative free optimization algorithms, Auger, A. et al., Acte du 9ime colloque national en calcul des structures, Giens, 2009.

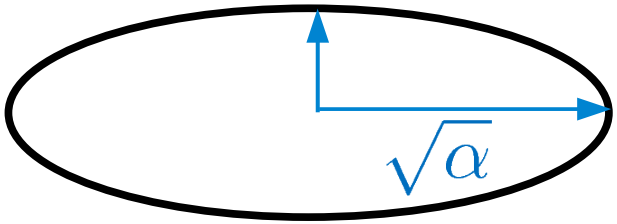
“Optimal” case for deterministic algorithms

Convex quadratic function:

Axis parallel ellipsoid (separable)

$$f(x) = \sum_{i=1}^{d} \alpha^{\frac{i-1}{d-1}} x_i^2$$

Condition number: α



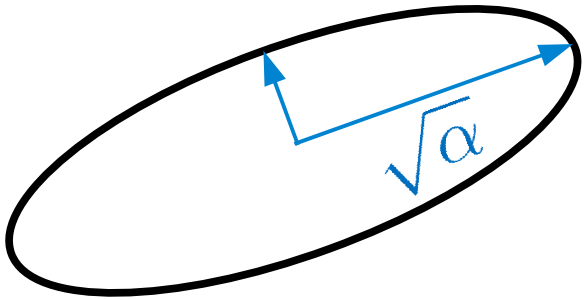
Rotated ellipsoid

$$f(x) = \sum_{i=1}^{d} \alpha^{\frac{i-1}{d-1}} (Rx_i)^2$$

R: orthogonal matrix

(Rotated ellipsoid)$^{1/4}$

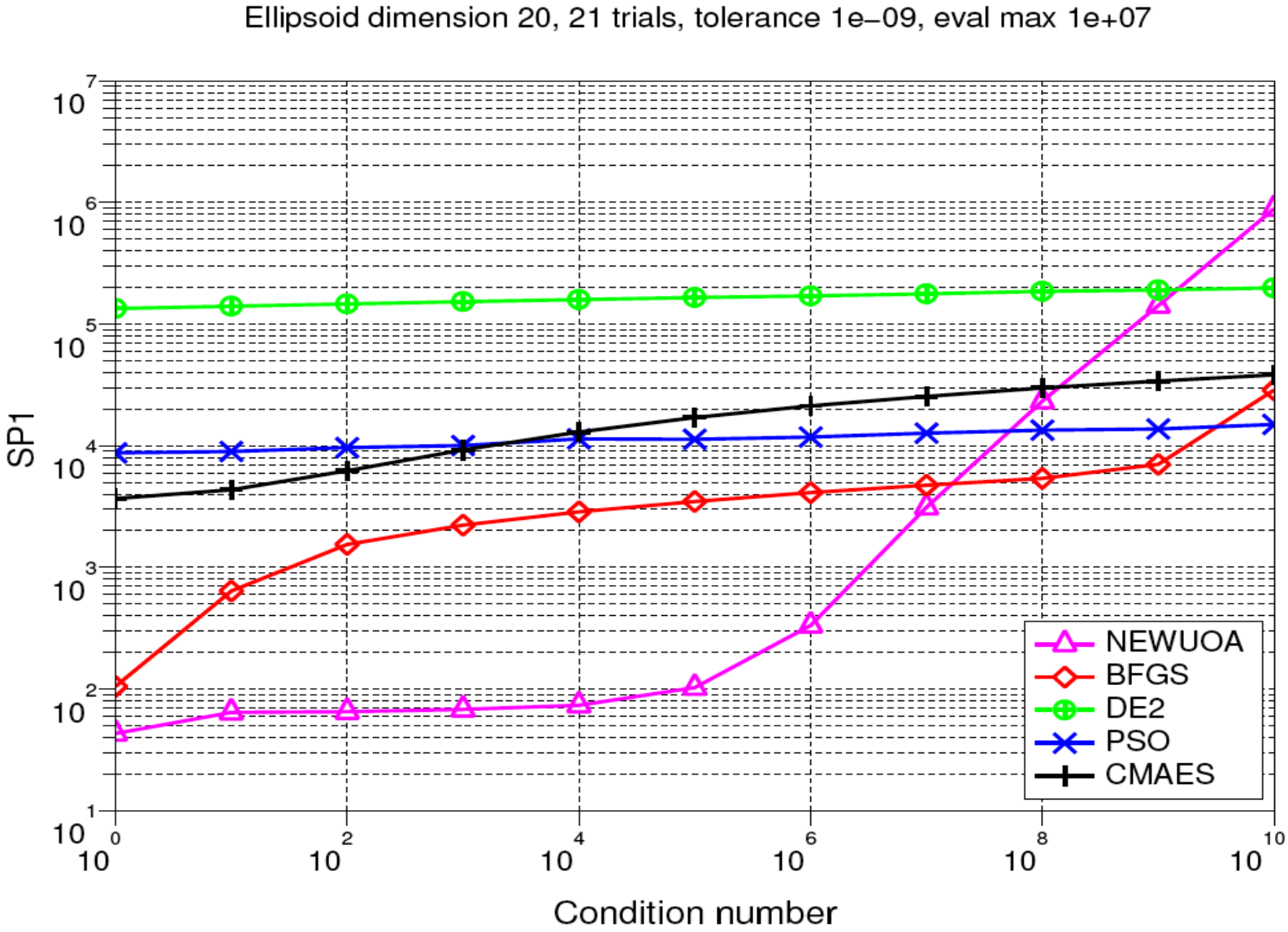
$$f(x) = \left(\sum_{i=1}^{d} \alpha^{\frac{i-1}{d-1}} (Rx_i)^2 \right)^{1/4}$$



Impact of condition number - Ellipsoid

$$\mathrm{SP1} = \#\ \text{Evals to reach } 10^{-9}$$

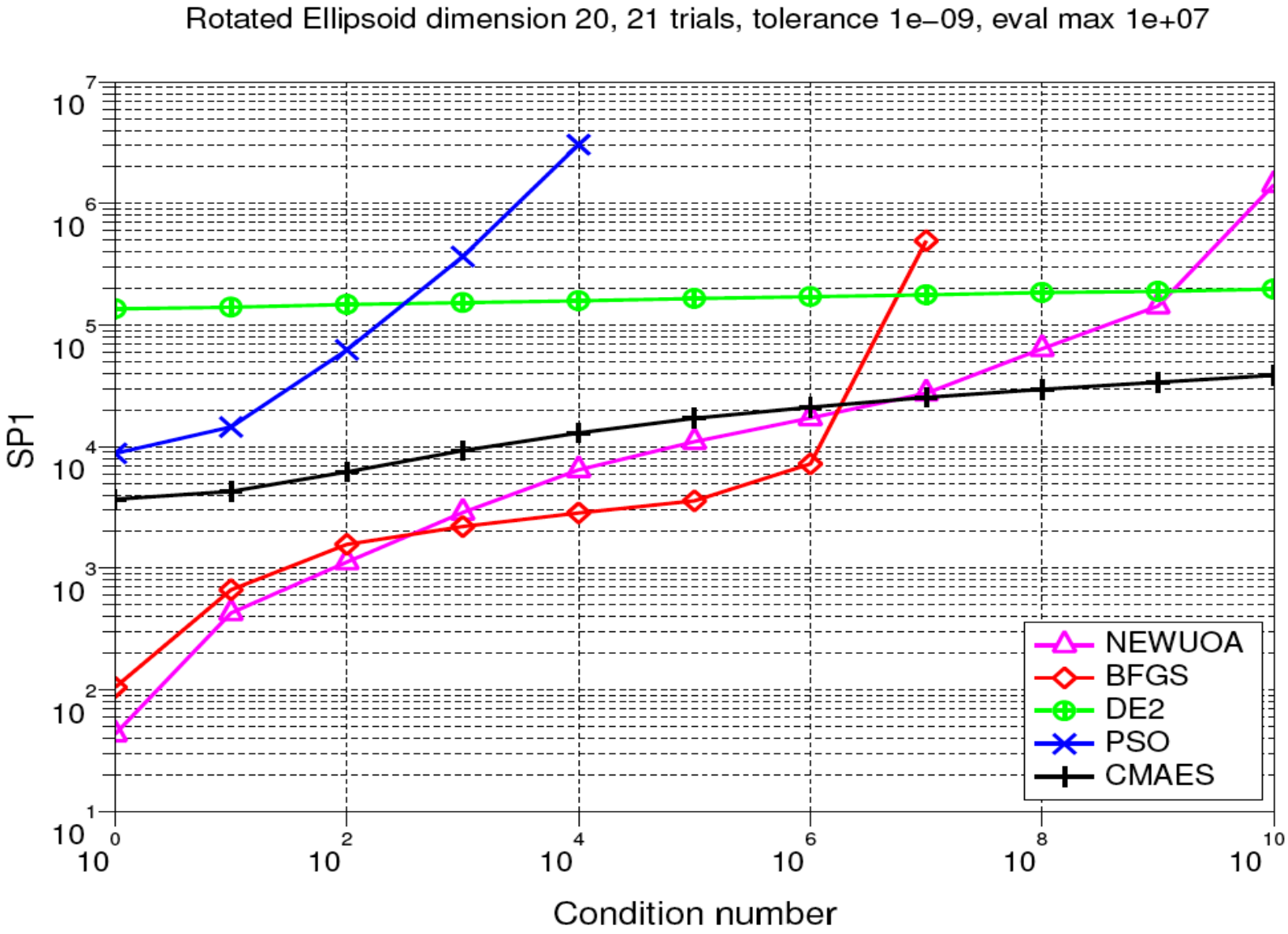
$$f(x) = \sum_{i=1}^{d} \alpha^{\frac{i-1}{d-1}} x_i^2$$



Impact of condition number – Rotated Ellipsoid

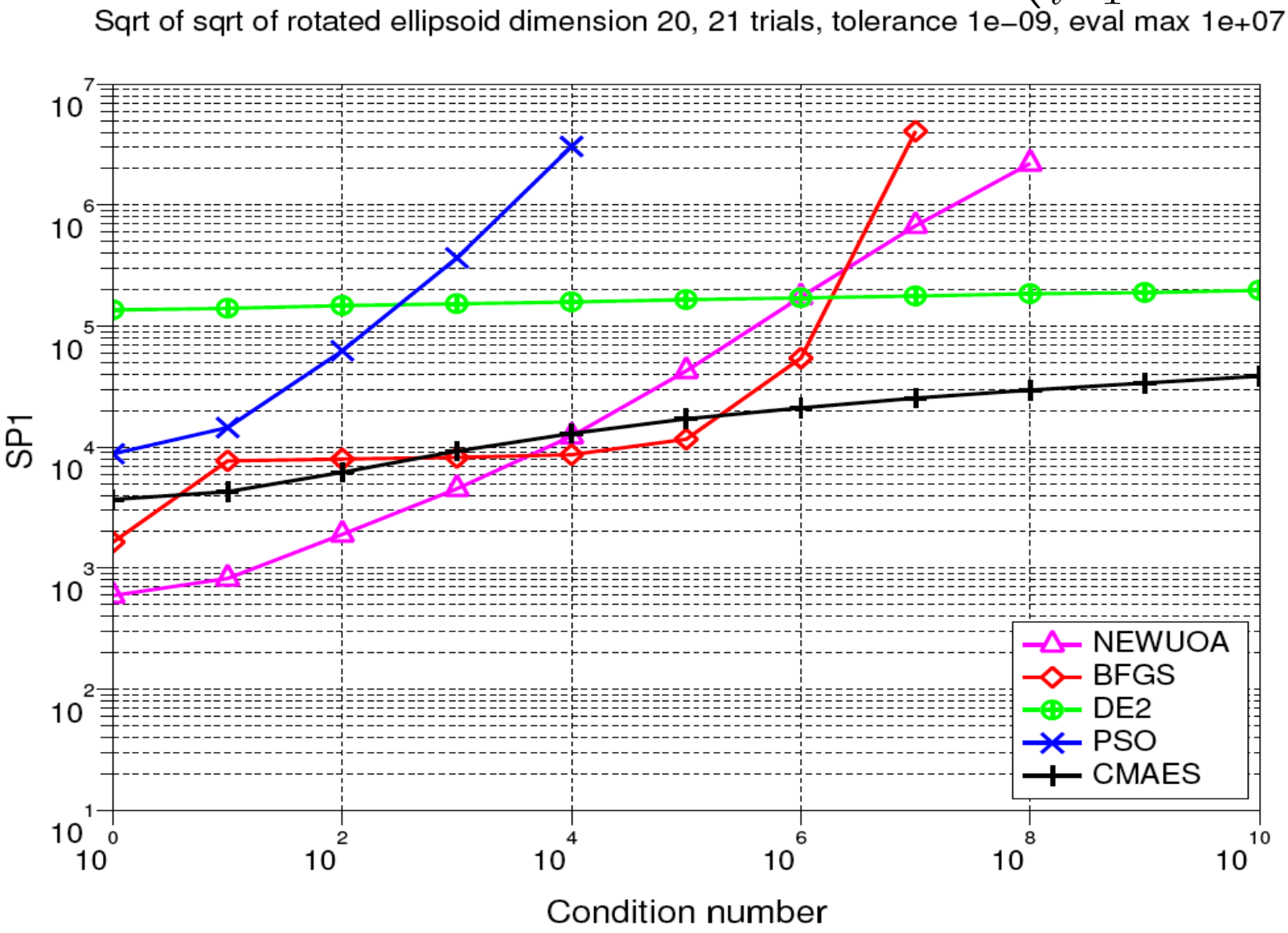
$$\text{SP1} = \#\ \text{Evals to reach } 10^{-9}$$

$$f(x) = \sum_{i=1}^{d} \alpha^{\frac{i-1}{d-1}} (Rx_i)^2$$



Impact of condition number – (Rotated Ellipsoid)¼

$$\text{SP1} = \#\ \text{Evals to reach } 10^{-9} \qquad f(x) = \left(\sum_{i=1}^{d} \alpha^{\frac{i-1}{d-1}} (Rx_i)^2 \right)^{1/4}$$



Black-Box-Optimization-Benchmarking (BBOB)

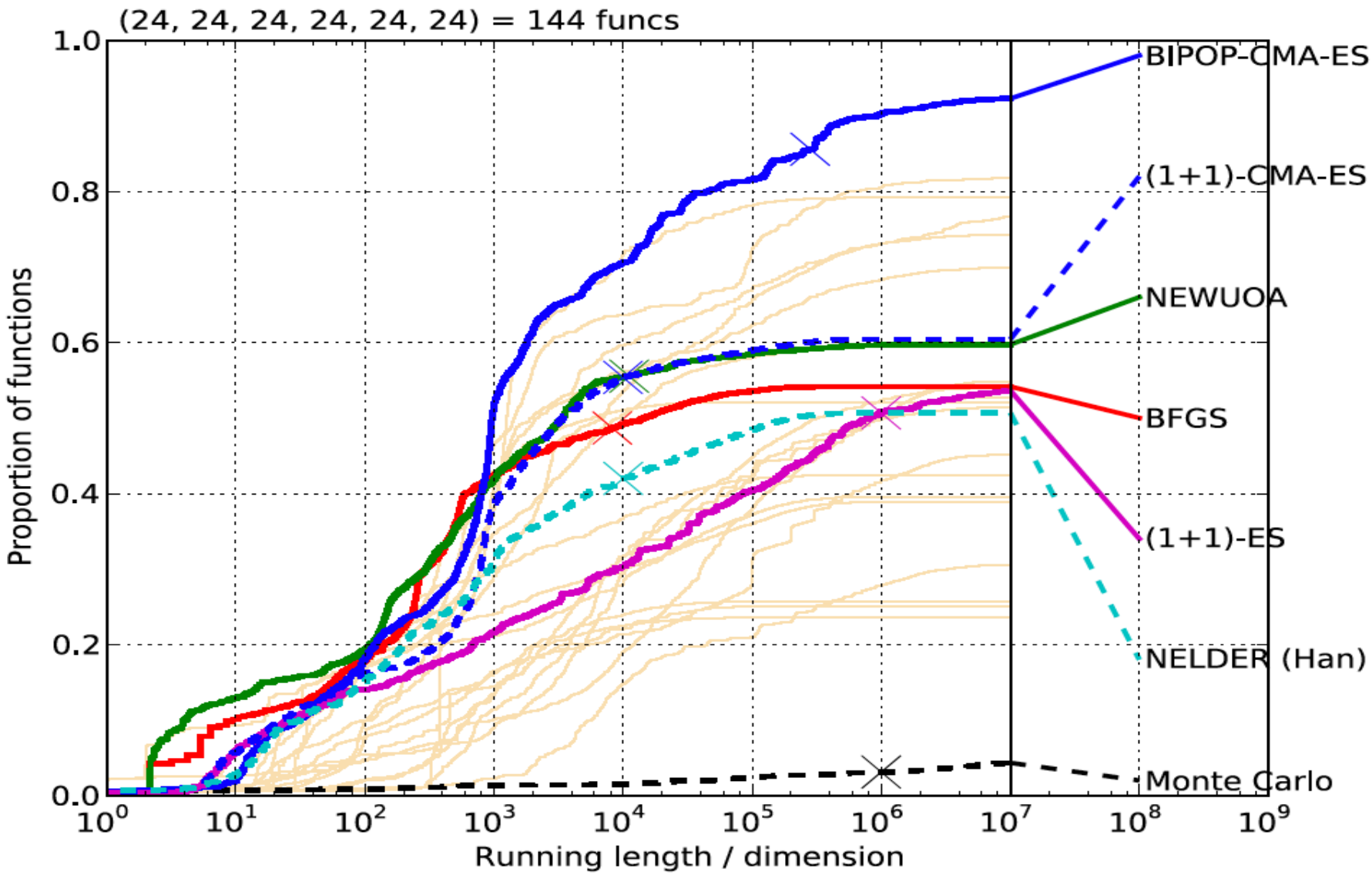
- Black Box Optimization Benchmarking (BBOB) workshop, GECCO-2009, GECCO-2010

  [http://coco.gforge.inria.fr/doku.php?id=bbob-2010]

  - function testbed
    - mainly non-convex and non-separable
    - Scalable with the search space dimension
    - not too easy to solve, but yet comprehensible
  - provide code for data acquisition (only need to plug the optimizers)
  - provide code for post-processing experiments
    - Data presentation yields quantitative assessment, stratified by function properties

Comprehensive comparison of 28 algorithms

Cumulative distribution of running length distribution (aka data profile)

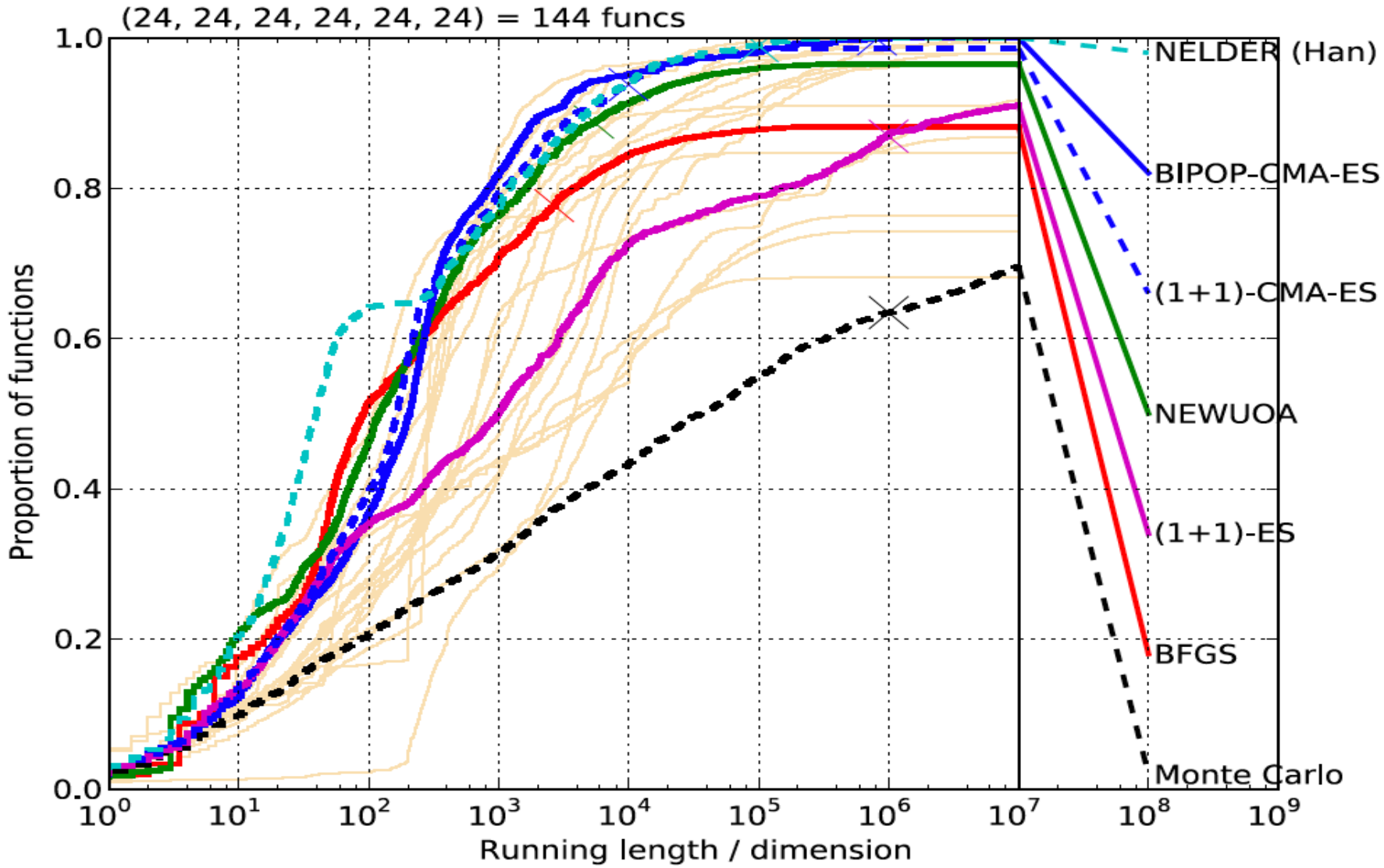


d=20

BBOB non-noisy testbed d=20

Comprehensive comparison of 28 algorithms

Cumulative distribution of running length distribution

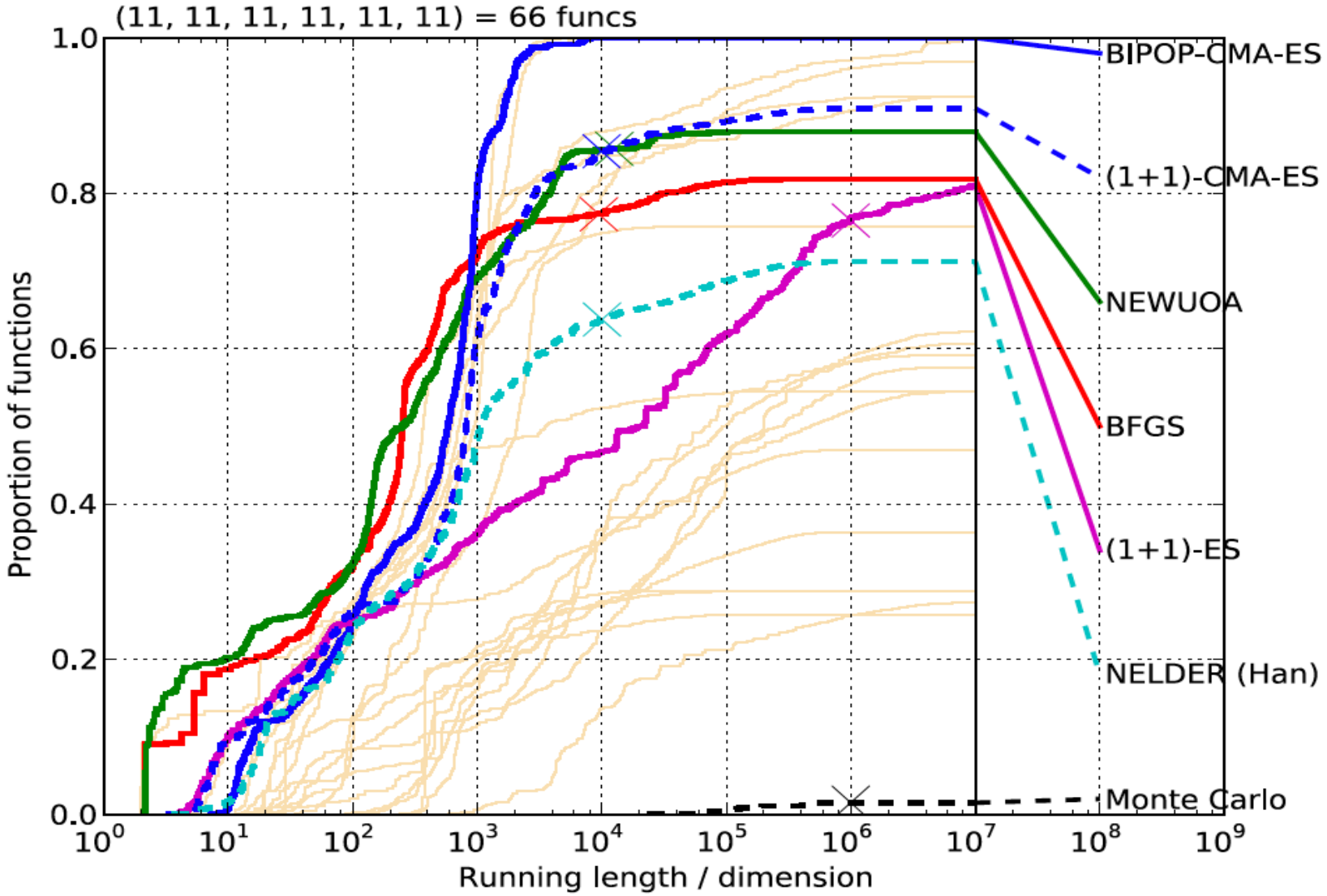


d=2

BBOB non-noisy testbed

Comprehensive comparison of 28 algorithms

Cumulative distribution of running length distribution

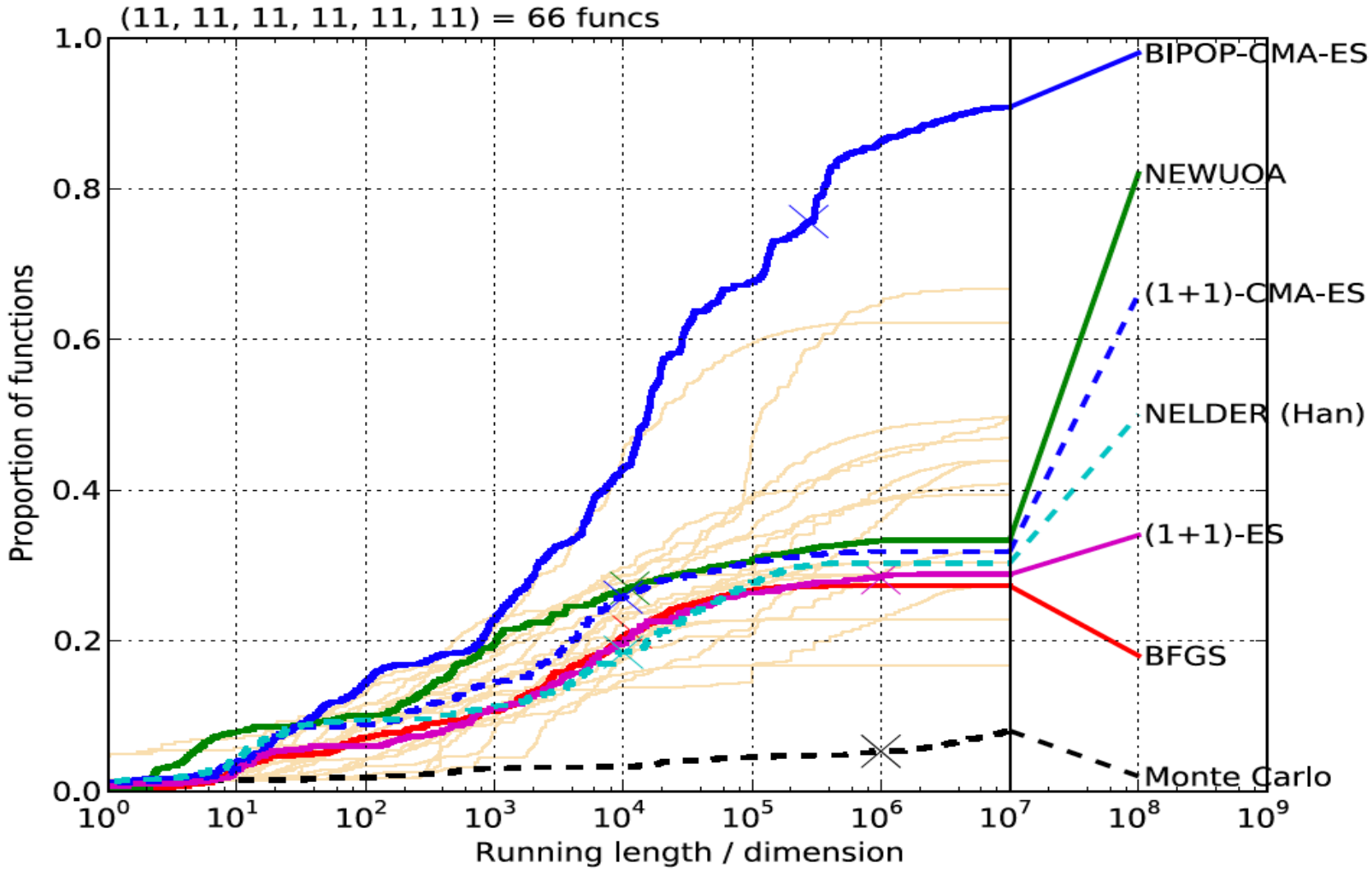


d=20
unimodal functions

BBOB non-noisy testbed

Comprehensive comparison of 28 algorithms

Cumulative distribution of running length distribution

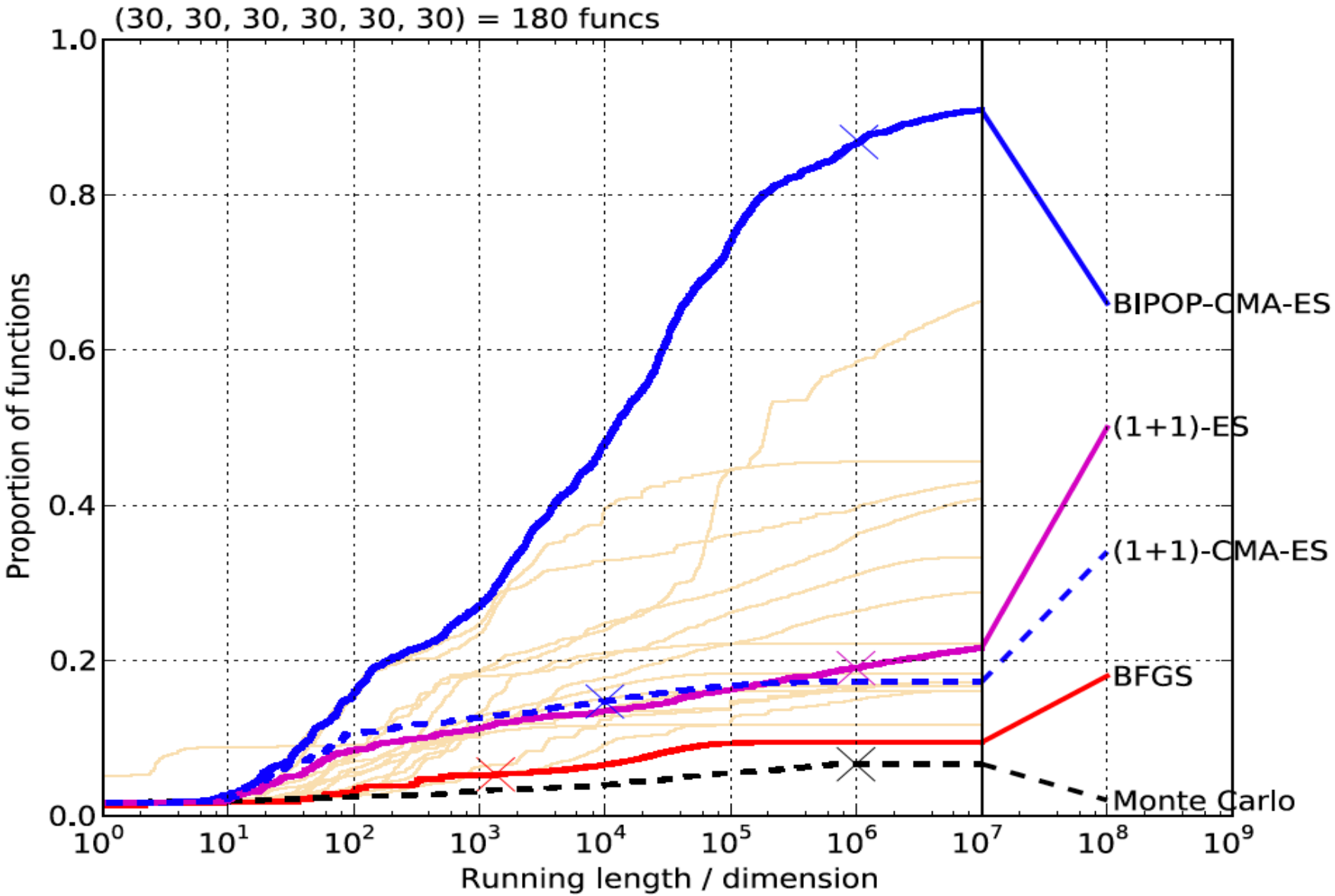


d=20, multi-modal functions

BBOB non-noisy testbed d=20

Comprehensive comparison of 28 algorithms

Cumulative distribution of running length distribution



d=20, noisy functions

Overview

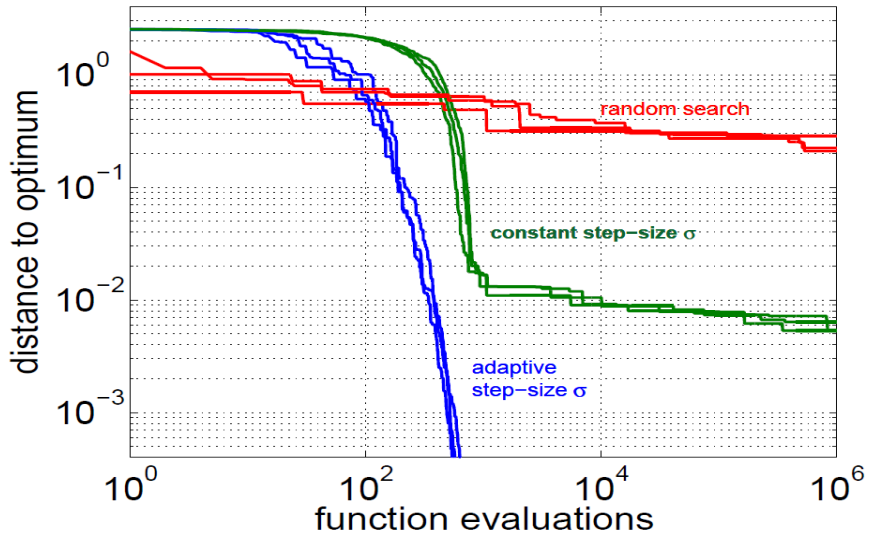
- Stochastic optimization algorithms
- Benchmarking stochastic and deterministic DFOs
- Theoretical convergence results

can be rather meaningless ...

Easy to prove almost sure convergence of Pure Random Search towards global optimum on class of function with very mild assumptions.

Then, easy to make "any" algorithm globally convergent by adding every 10^{100} iterations a step of Pure Random Search

but the expected hitting time $E(\tau_{B(\mathbf{x}^*,\epsilon)}) = \Theta(\frac{1}{\epsilon^d})$



Linear convergence for sequence of RVs

$(z_n)_n$, deterministic sequence, converges to z

Linear convergence: if $\exists \mu \in (0,1)$, $\lim_{n\to\infty} \frac{\|z_{n+1} - z\|}{\|z_n - z\|} = \mu$

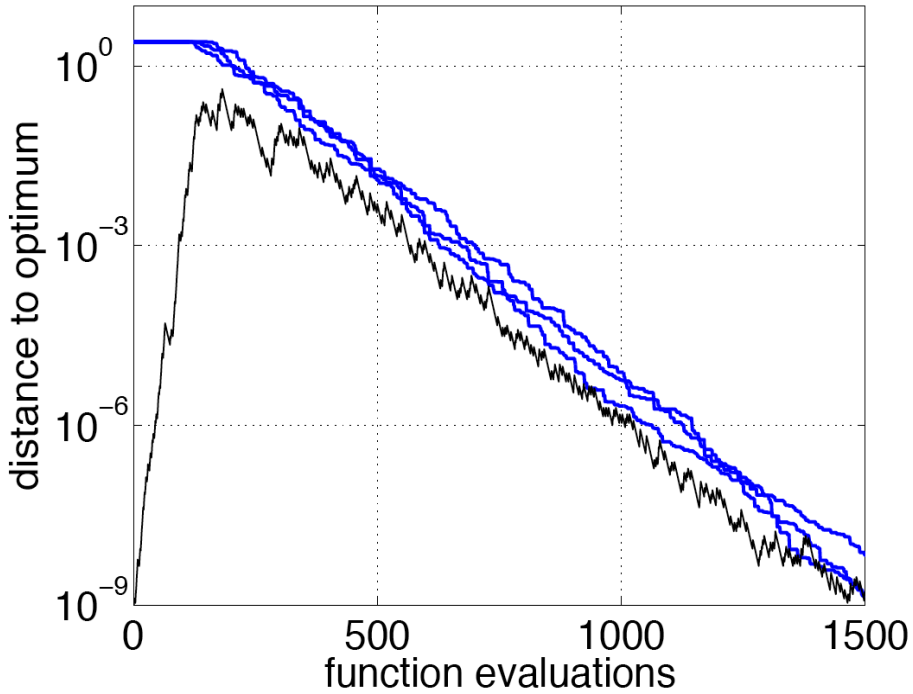
$\Downarrow$ (Cesàro mean result)

$$\lim_{n\to\infty} \frac{1}{n} \ln\left(\frac{\|z_n - z\|}{\|z_0 - z\|}\right) = \ln(\mu)$$

Linear convergence for sequence of RVs

$(z_n)_n$, sequence of random variables, converges to z almost surely (a.s.)
Linear convergence a.s.: if

$$\exists\, c < 0,\ \lim_{n\to\infty} \frac{1}{n} \ln\left(\frac{\|z_n - z\|}{\|z_0 - z\|}\right) = c \text{ a.s.}$$



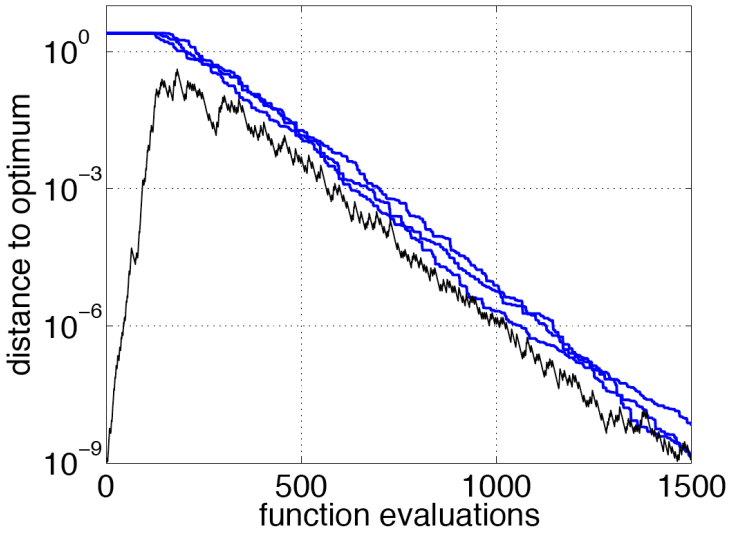
Hitting time growing like $\ln(1/\epsilon)$ when $\epsilon \to 0$

General lower bounds for convergence

Theorem. *The median hitting time of any rank-based search algorithm is lower bounded as*

$$\text{median}(\tau(\epsilon)) \geq \frac{\log_2(M(\epsilon)) - 1}{\log_2 K} \sim d \frac{\log_2(1/\epsilon)}{\log_2 K}$$

O. Teytaud et al (2006-2009)

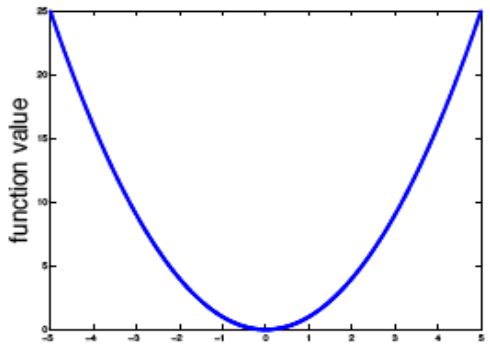


Can we actually prove linear convergence?

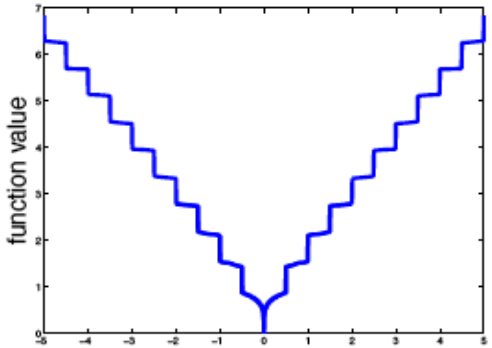
On which class of functions?

Class of functions

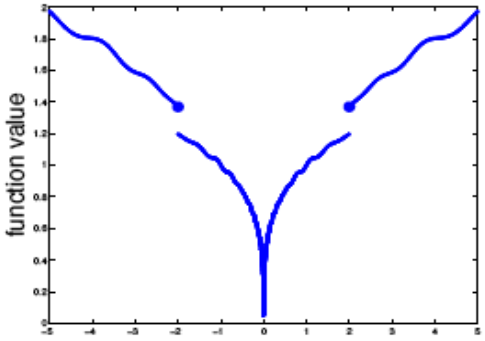
$$f(\mathbf{x}) = g(\|\mathbf{x} - \mathbf{x}^*\|), g : \mathbb{R}^+ \mapsto \mathbb{R} \text{ strictly increasing}$$



$f(x) = x^2$



$f(x) = g_1(x^2)$



$f(x) = g_2(x^2)$

Exploitation of invariance property of rank-based algorithms

Linear convergence – sufficient conditions

Consider the sequence (X_n, σ_n) generated by a step-size adaptive ES

$$\frac{1}{n} \ln \|X_n\| = \frac{1}{n} \ln \|X_0\| + \frac{1}{n} \sum_{k=1}^{n} \ln \underbrace{\frac{\|X_k\|}{\|X_{k-1}\|}}_{\mathcal{F}(\|X_k\|/\sigma_k)}$$

Lemma. *For $f(\mathbf{x}) = g(\|\mathbf{x}\|)$, $Z_n = \|X_n\|/\sigma_n$ is an homegenous Markov chain.*

Theorem. *If $Z_n = \|X_n\|/\sigma_n$ satisfy Law of Large Numbers, then for all X_0, σ_0, there exists c such that*

$$\frac{1}{n} \ln \|X_n\| \to c$$

Law of Large Numbers for Markov Chains

$Z_n = \frac{\|X_n\|}{\sigma_n}$ "stable enough"

- φ-irreducible: Z_n can visit with positive probability all the sets of φ-measure positive
- Harris recurrent: starting from any point, Z_n hits A with $\varphi(A) > 0$ with probability one
- positivity: there exists a unique invariant probability measure ν

$$Z_n \sim \nu \Rightarrow Z_{n+1} \sim \nu$$

Verifying stability conditions

by means of drift conditions

for step-size with self-adaptation

Anne Auger (2005), Convergence results for (1,λ)-SA-ES using the theory of φ-irreducible Markov chains.

for one-fifth success rule with $\sigma_n = \alpha\sigma_n$ for increase and $\sigma_{n+1} = \alpha^{-1/4}\sigma_n$ for decrease

Stability if there exists $\theta > 0$ such that

$$\frac{1}{\alpha^{\theta}} + \alpha^{\theta 1/4} < 2$$

Auger – Hansen in preparation

believe can be extended to larger class of functions, functions with noise and to CMA-ES for proving $\mathbf{C_n} \propto H^{-1}$

- Acknowledgments: Nikolaus Hansen, Raymond Ros

- Forthcoming events:
  - GECCO Workshop Black Box Optimization Benchmarking 2010
    - Deadline to submit your favorite algorithm: March 25th

      [http://coco.gforge.inria.fr/doku.php?id=bbob-2010]
  - Dagstuhl seminar on “Theory of Evolutionary Algorithms” September 2010

    [co-organized with J. Shapiro, D. Whitley, C. Witt]