Covariance Matrix Adaptation (CMA) Evolution Strategy

Nikolaus Hansen

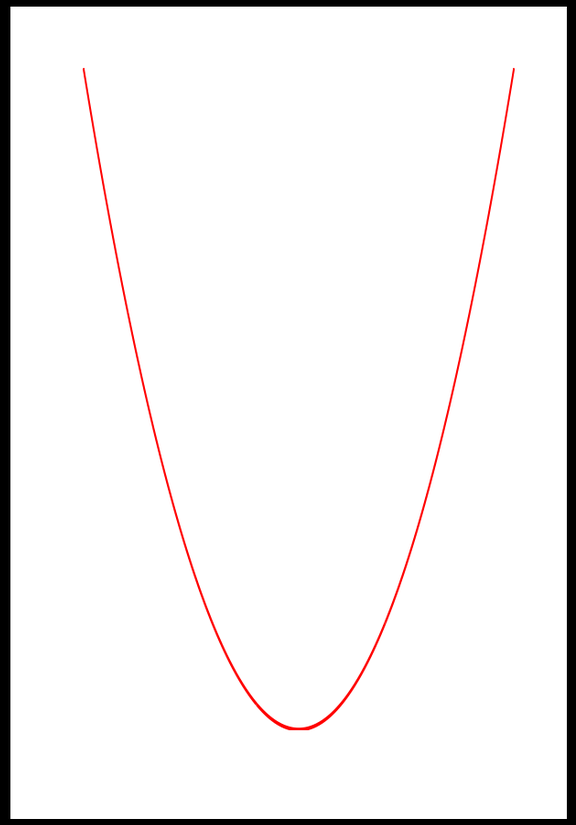
Motivation::

Optimization refers to finding the **best element** from a set of available alternatives

"element"	$\mathbf{x} \in S \subseteq \mathbb{R}^n$
"best"	$f : S \subseteq \mathbb{R}^n \longrightarrow \mathbb{R}$
"finding the best"	$\min_{\mathbf{x} \subseteq S} f(\mathbf{x})$

Motivation::

Cost function “well behaved” *(analytical definition, differentiable, convex, separable, etc)*

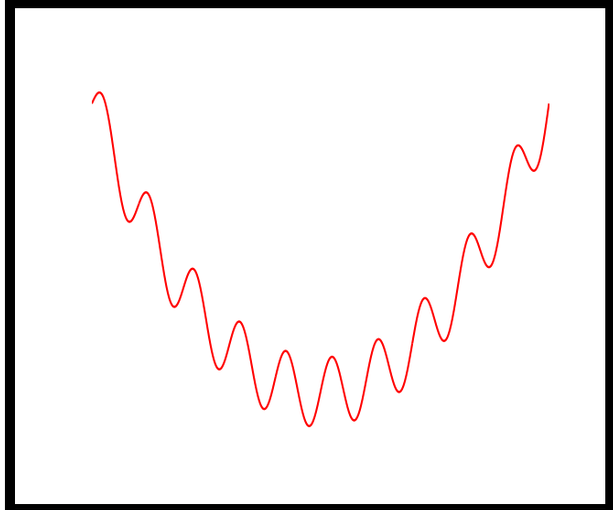


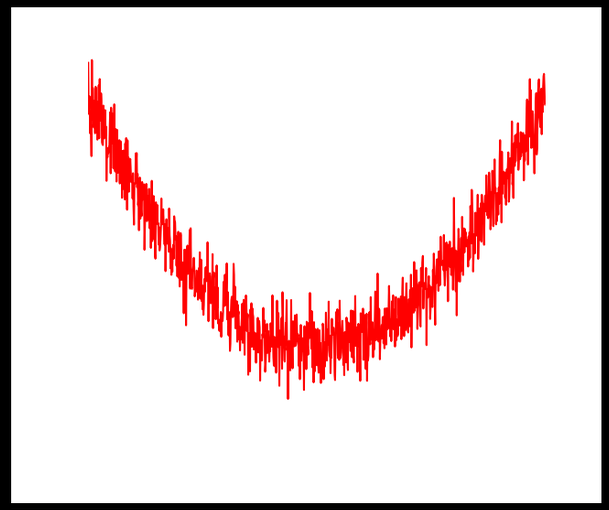
- analytically
- gradient descent methods
- convex programming
- quadratic programming
- nonlinear programming
- etc

http://www.flickr.com/photos/narnonic/

Motivation::

Real world problems *(no derivatives, non-convex, non-separable, non-linear, noisy, etc)*

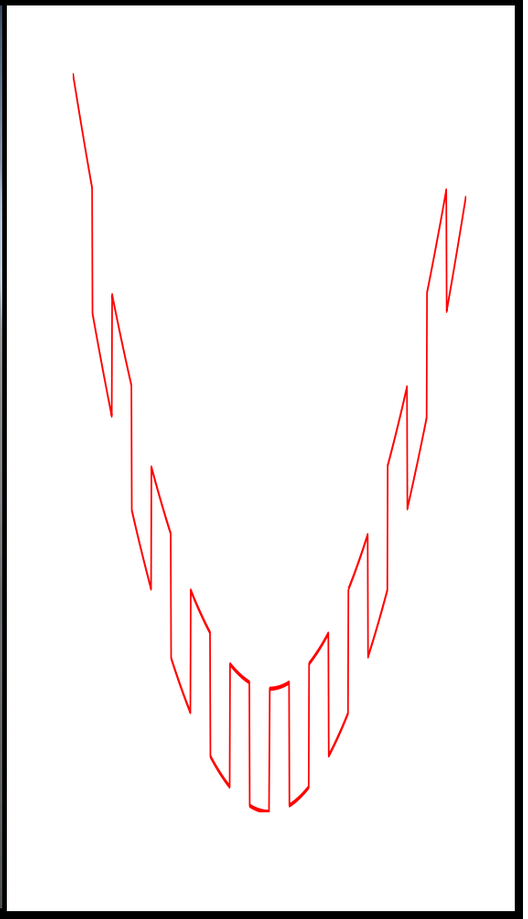




http://www.flickr.com/photos/pattylagera/

http://www.flickr.com/photos/70148269@N00/

http://www.flickr.com/photos/castaspella/



Motivation::

Goal: cope with a real-world black box scenario

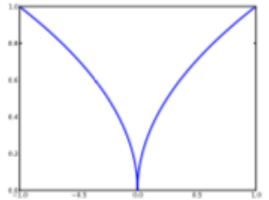
- moderate dimensionality (~10÷100)
- non-linear
- non-convex
- non-separable
- multimodal
- noisy
- discontinuous
- ill-conditioned

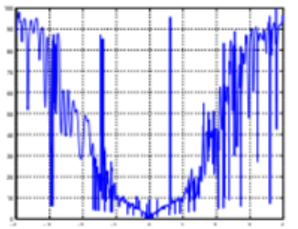
Approach: stochastic search, evolutionary strategies

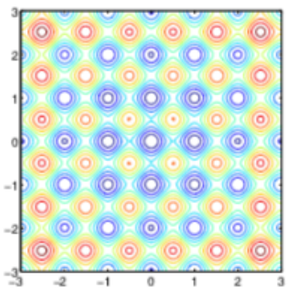
What Makes a Function Difficult to Solve?

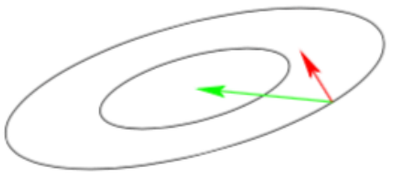
Why stochastic search?

- 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





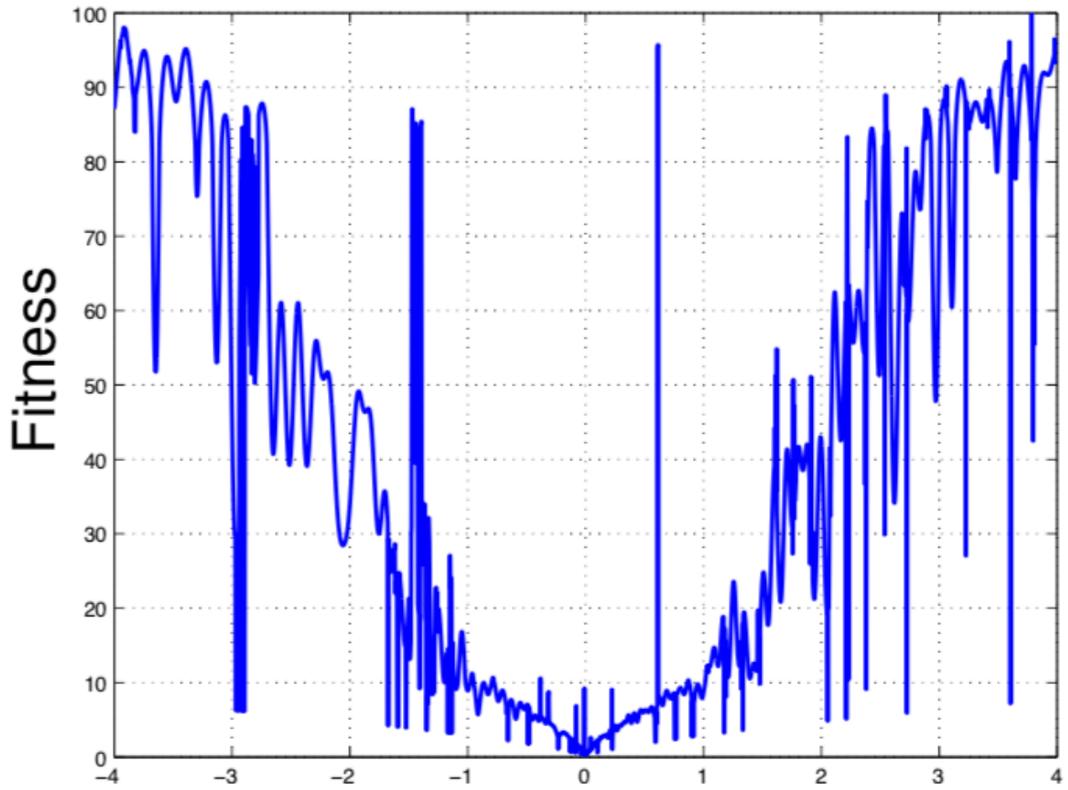




gradient direction Newton direction

Ruggedness

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



cut from a 5-D example, (easily) solvable with evolution strategies

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 $[0, 1]$. Now consider the 10-dimensional space $[0, 1]^{10}$. To get **similar coverage** in terms of distance between adjacent points would require $100^{10} = 10^{20}$ points. A 100 points appear now as isolated points in a vast empty space.

Remark: **distance measures** break down in higher dimensionalities (the central limit theorem kicks in)

Consequence: a **search policy** that is valuable in small dimensions **might be useless** in moderate or large dimensional search spaces. Example: exhaustive search.

Separable Problems

Definition (Separable Problem)

A function f is separable if

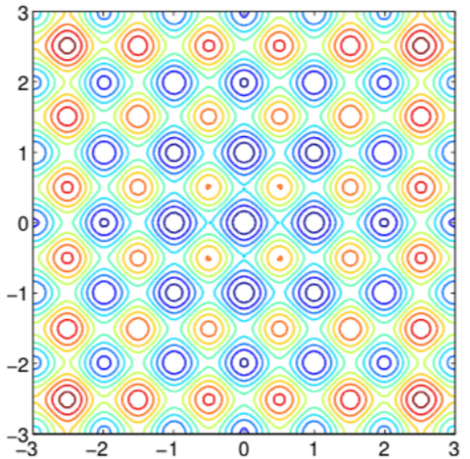
$$\arg\min_{(x_1,\ldots,x_n)} f(x_1,\ldots,x_n) = \left(\arg\min_{x_1} f(x_1,\ldots), \ldots, \arg\min_{x_n} f(\ldots,x_n)\right)$$

$\Rightarrow$ 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



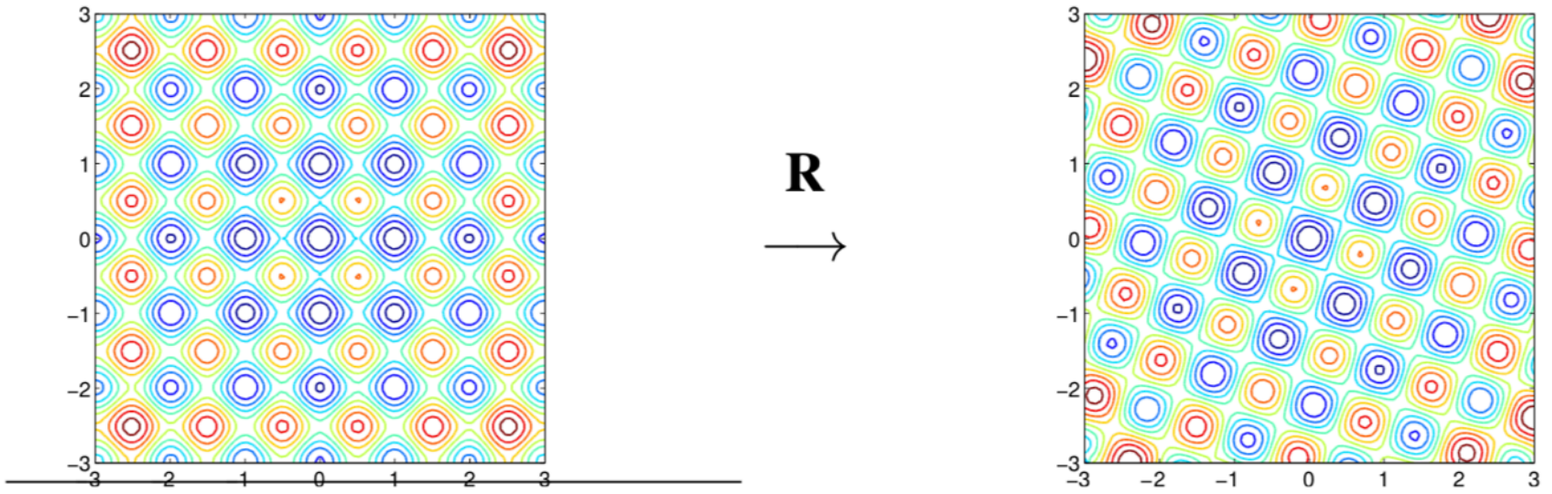
Non-Separable Problems

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**

$\mathbf{R}$ rotation matrix



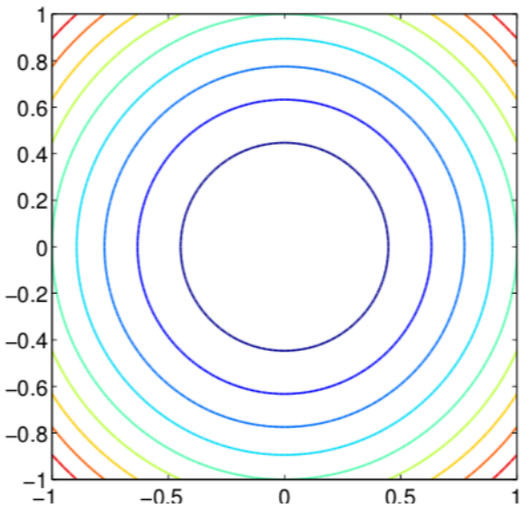
[1] Hansen, Ostermeier, Gawelczyk (1995). On the adaptation of arbitrary normal mutation distributions in evolution strategies: The generating set adaptation. Sixth ICGA, pp. 57-64, Morgan Kaufmann

[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 quadratic, $f : \boldsymbol{x} \mapsto \boldsymbol{x}^{\mathrm{T}} \boldsymbol{H} \boldsymbol{x}$, ill-conditioned means a high condition number of Hessian Matrix $\boldsymbol{H}$

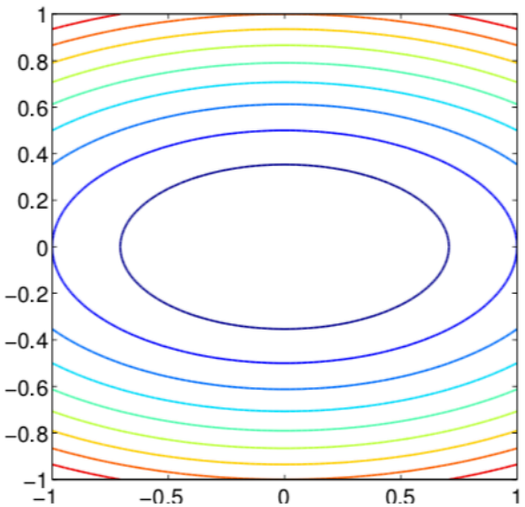
ill-conditioned means "squeezed" lines of equal function value



Increased

$\longrightarrow$

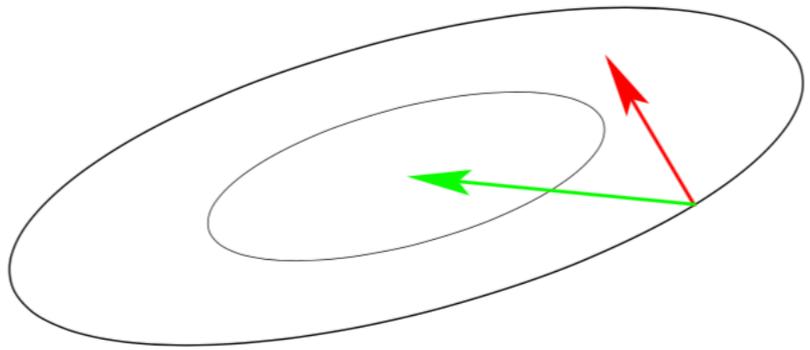
condition number



consider the curvature of iso-fitness lines

The Benefit of Second Order Information

Consider the convex quadratic function $f(\boldsymbol{x}) = \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}^*)^T \boldsymbol{H}(\boldsymbol{x} - \boldsymbol{x}^*)$



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

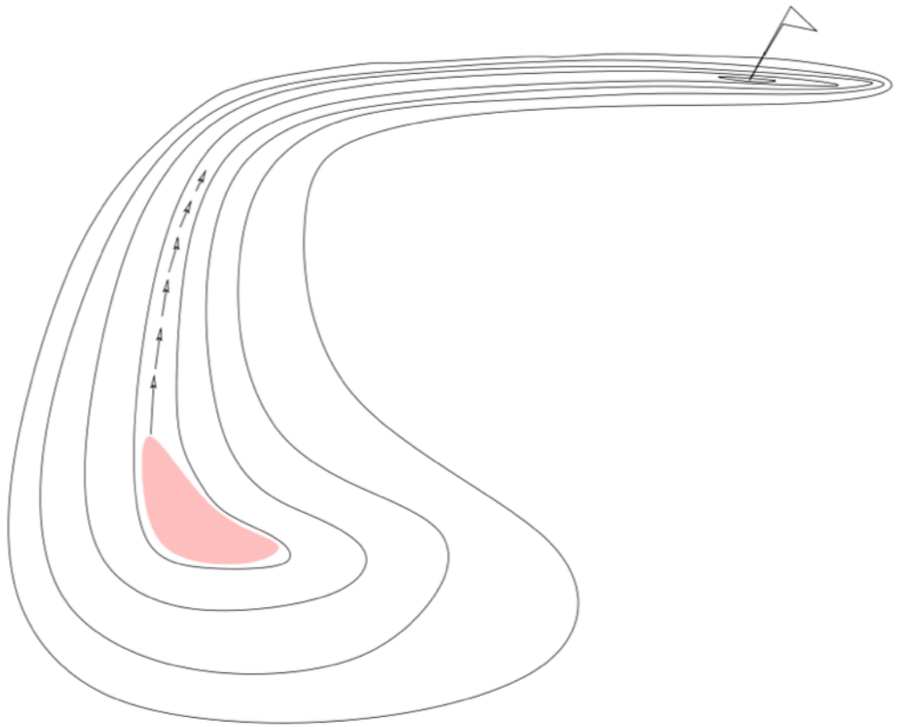
Newton direction $-\boldsymbol{H}^{-1}f'(\boldsymbol{x})^{\mathrm{T}}$

Condition number equals nine here. Condition numbers between 100 and even 10^6 can be observed in real world problems.

If $\boldsymbol{H} \approx \mathbf{I}$ (small condition number of $\boldsymbol{H}$) first order information (e.g. the gradient) is sufficient. Otherwise second order information (estimation of $\boldsymbol{H}^{-1}$) is required.

Ill-Conditioned Problems

Example: A Narrow Ridge



Volume oriented search ends up in the pink area.
To approach the optimum an ill-conditioned problem needs to be solved (e.g. by following the narrow bent ridge).[3]

[3] Whitley, Lunacek, Knight 2004. Ruffled by Ridges: How Evolutionary Algorithms Can Fail, *GECCO*

What Makes a Function Difficult to Solve?

. . . and what can be done

The Problem	Possible Approaches
Dimensionality	exploiting the problem structure separability, locality/neighborhood, encoding
Ill-conditioning	second order approach changes the neighborhood metric
Ruggedness	**non-local** policy, large sampling width (step-size) as large as possible while preserving a reasonable convergence speed
	population-based method, stochastic, non-elitistic
	recombination operator serves as repair mechanism
	restarts

. . . metaphors

Stochastic search & CMA::

Black box template to minimize $f : \mathbb{R}^n \longrightarrow \mathbb{R}$

set variable strategy parameters θ

set population size $\lambda \in \mathbb{N}$

loop

1) sample distribution $P(\mathbf{x}|\theta) \longrightarrow \mathbf{x}_1, \ldots, \mathbf{x}_\lambda \in \mathbb{R}^n$

2) evaluate $f(\mathbf{x}_1), \ldots, f(\mathbf{x}_\lambda)$

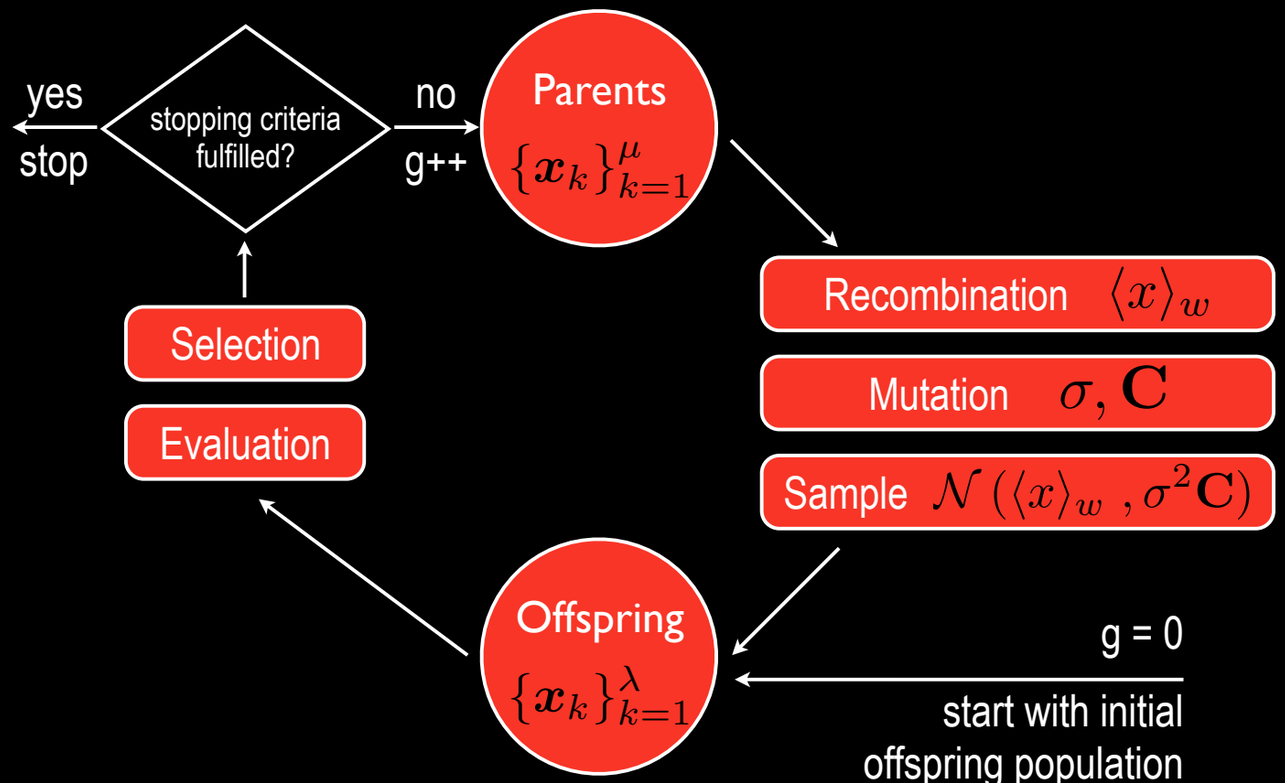
3) update $\theta \longleftarrow F_\theta(\theta, \mathbf{x}_1, \ldots, \mathbf{x}_\lambda, f(\mathbf{x}_1), \ldots, f(\mathbf{x}_\lambda))$

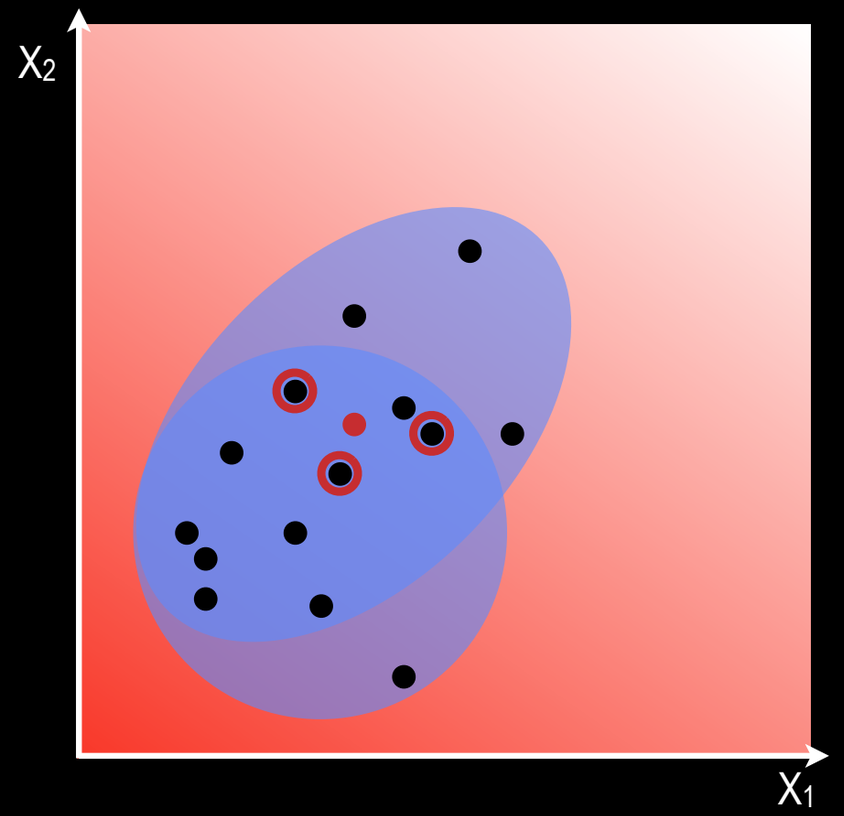
CMA-ES

- P is a multivariate normal distribution $\mathcal{N}(\mathbf{m}, \sigma^2\mathbf{C})$
- $\theta = \{\mathbf{m}, \sigma, \mathbf{C}\} \in \mathbb{R}^n \times \mathbb{R}^{n\times n} \times \mathbb{R}_+$
- $F_\theta = F_\theta(\theta, \mathbf{x}_{1:\lambda}, \ldots, \mathbf{x}_{\mu:\lambda})$ where $\mathbf{x}_{i:\lambda}$ is the i-th best point and $\mu \leq \lambda$

Covariance Matrix Adaptation - ES::

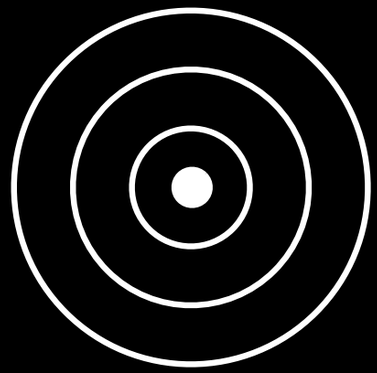
- **black box** approach (no $\nabla\Phi$, Φ = cost function)
- **iterative** methods operating with **populations** of candidate solutions



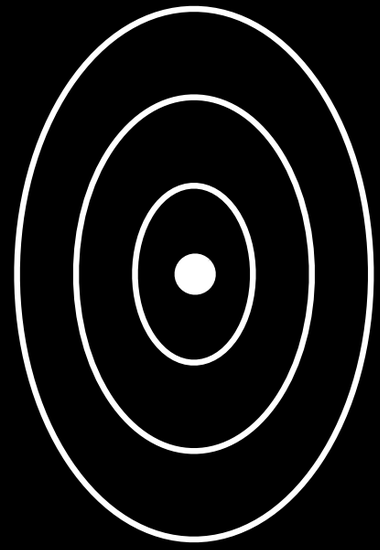


[1] Hansen, Muller, Koumoutsakos. **Evolutionary Computation** (2003)

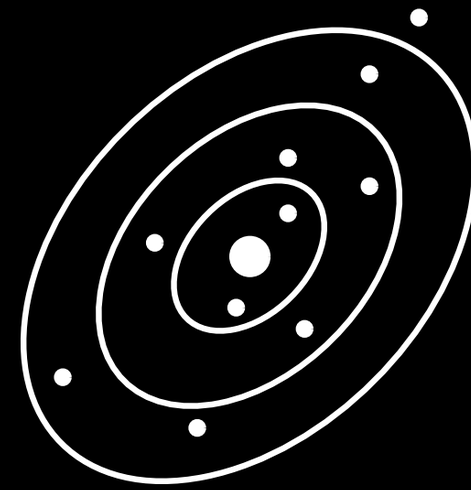
Sampling multivariate normal distribution::



$$\mathcal{N}(\mathbf{m}, \sigma^2\mathbf{I}) \sim \mathbf{m} + \sigma\mathcal{N}(\mathbf{0}, \mathbf{I})$$



$$\mathcal{N}(\mathbf{m}, \sigma^2\mathbf{D}) \sim \mathbf{m} + \sigma\mathbf{D}^{\frac{1}{2}}\mathcal{N}(\mathbf{0}, \mathbf{I})$$

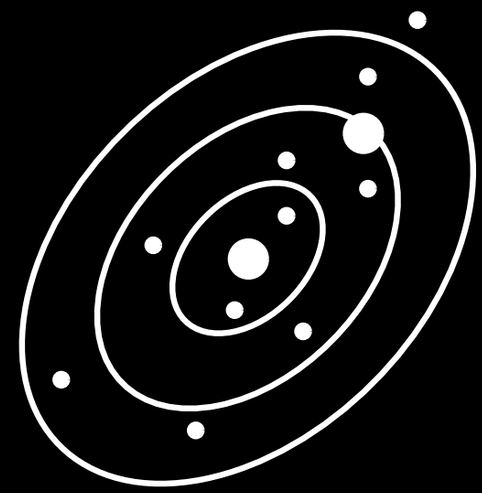


$$\mathcal{N}(\mathbf{m}, \sigma^2\mathbf{C}) \sim \mathbf{m} + \sigma\mathbf{C}^{\frac{1}{2}}\mathcal{N}(\mathbf{0}, \mathbf{I})$$

Recombination:: Update *m*::

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

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

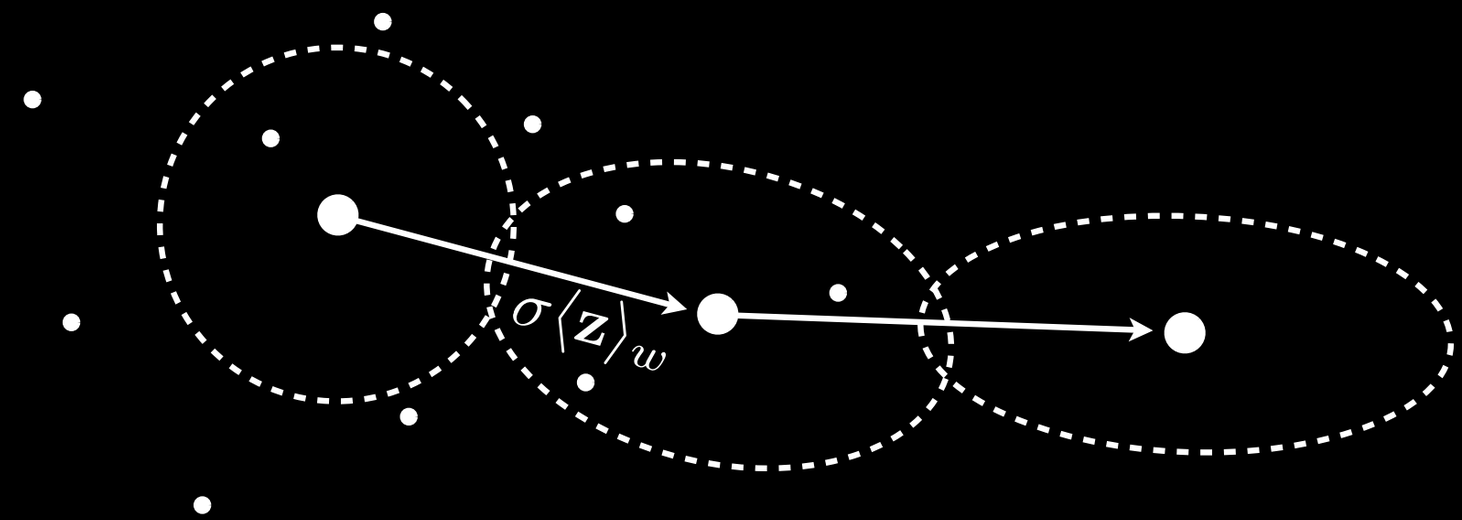


$$\mathbf{m} \longleftarrow \sum_{i=1}^{\mu} w_i \mathbf{x}_{i:\lambda} = \mathbf{m} + \sigma \sum_{i=1}^{\mu} w_i \mathbf{z}_{i:\lambda} = \mathbf{m} + \sigma\langle\mathbf{z}\rangle_w$$

$$\sum_{i=1}^{\mu} w_i = 1 \qquad w_1 \geq \cdots \geq w_\mu \geq 1$$

Mutation:: Update C:: Rank-one update

Mutate **C** to increase the probability of successful steps to appear again



$$\mathbf{C} \longleftarrow (1 - LR_{cov})\mathbf{C} + LR_{cov}\langle \mathbf{z} \rangle_w \langle \mathbf{z} \rangle_w^T$$

covariance matrix adaptation

- learns all **pairwise dependencies** between variables

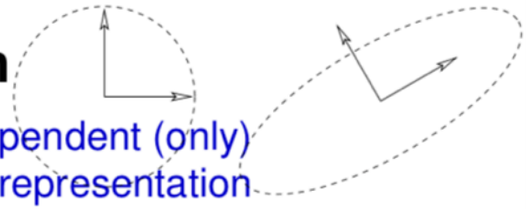
  off-diagonal entries in the covariance matrix reflect the dependencies

- conducts a **principle component analysis** (PCA) of steps y_w, sequentially in time and space

  eigenvectors of the covariance matrix $\mathbf{C}$ are the principle components / the principle axes of the mutation ellipsoid

- learns a new **rotated problem representation**

  components are independent (only) in the new representation



- learns a **new** (Mahalanobis) **metric**

  variable metric method

- approximates the **inverse Hessian** on quadratic functions

  transformation into the sphere function

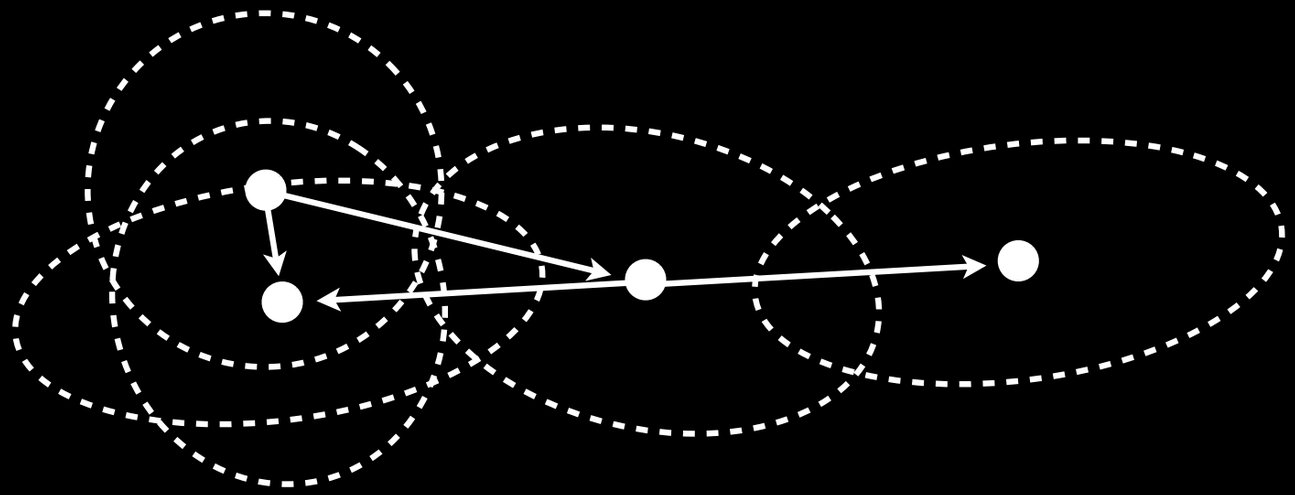
- for $\mu = 1$: conducts a **natural gradient ascent** on the distribution $\mathcal{N}$

  entirely independent of the given coordinate system

Mahalanobis metric: It is a multi-dimensional generalization of the idea of measuring how many standard deviations away P is from the mean of D. This distance is zero if P is at the mean of D, and grows as P moves away from the mean along each principal component axis.

Mutation:: Update C:: Evolution path

$$\langle \mathbf{z} \rangle_w \langle \mathbf{z} \rangle_w^T = -\langle \mathbf{z} \rangle_w (-\langle \mathbf{z} \rangle_w)^T$$



$$\mathbf{p}_c \longleftarrow (1 - LR_c)\mathbf{p}_c + LR_c \langle \mathbf{z} \rangle_w$$

Recap::

$$\mathbf{x}_i = m + \sigma \mathbf{z}_i \qquad \mathbf{z}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{C})$$

$$\mathbf{m} \longleftarrow \mathbf{m} + \sigma \langle \mathbf{z} \rangle_w \qquad \langle \mathbf{z} \rangle_w = \sum_{i=1}^{\mu} w_i \mathbf{z}_{i:\lambda}$$

$$\mathbf{p}_c \longleftarrow (1 - c_c)\mathbf{p}_c + \sqrt{1 - (1 - c_c)^2} \sqrt{\frac{1}{\sum_{i=1}^{\mu} w_i^2}} \langle \mathbf{z} \rangle_w$$

$$\mathbf{C} \longleftarrow (1 - c_{cov})\mathbf{C} + c_{cov} \mathbf{p}_c \mathbf{p}_c^T$$

Recap:: Rank-one + cumulation

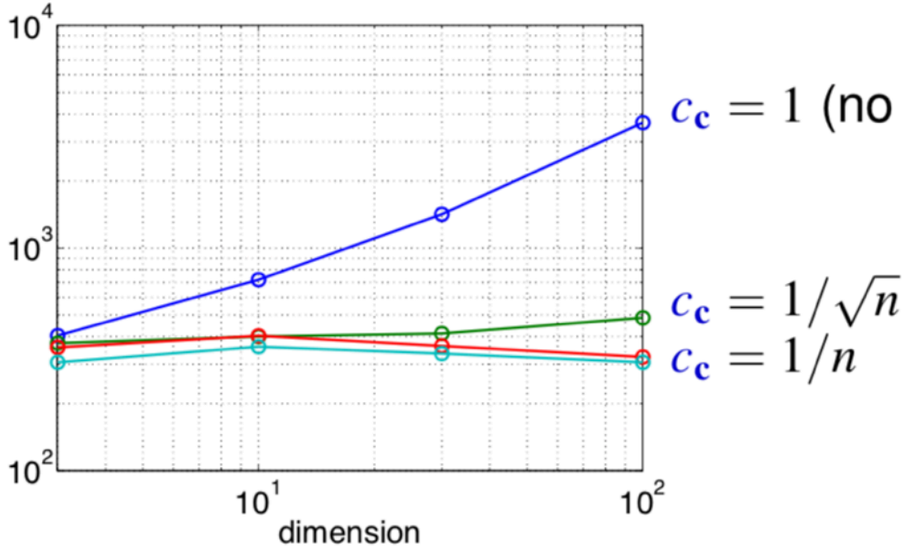
Evolution path + rank-one update reduces the number of function evaluations from $O(n^2)$ to $O(n)$

model complexity is $\mathbf{n^2}$ *but an important part of it can be learnt in* $\mathbf{n}$ *function evaluations*

Using an **evolution path** for the **rank-one update** of the covariance matrix reduces the number of function evaluations to adapt to a straight ridge **from about** $\mathcal{O}(n^2)$ **to** $\mathcal{O}(n)$.[a]

[a] Hansen & Auger 2013. Principled design of continuous stochastic search: From theory to practice.

Number of f-evaluations divided by dimension on the cigar function $f(\boldsymbol{x}) = x_1^2 + 10^6 \sum_{i=2}^{n} x_i^2$



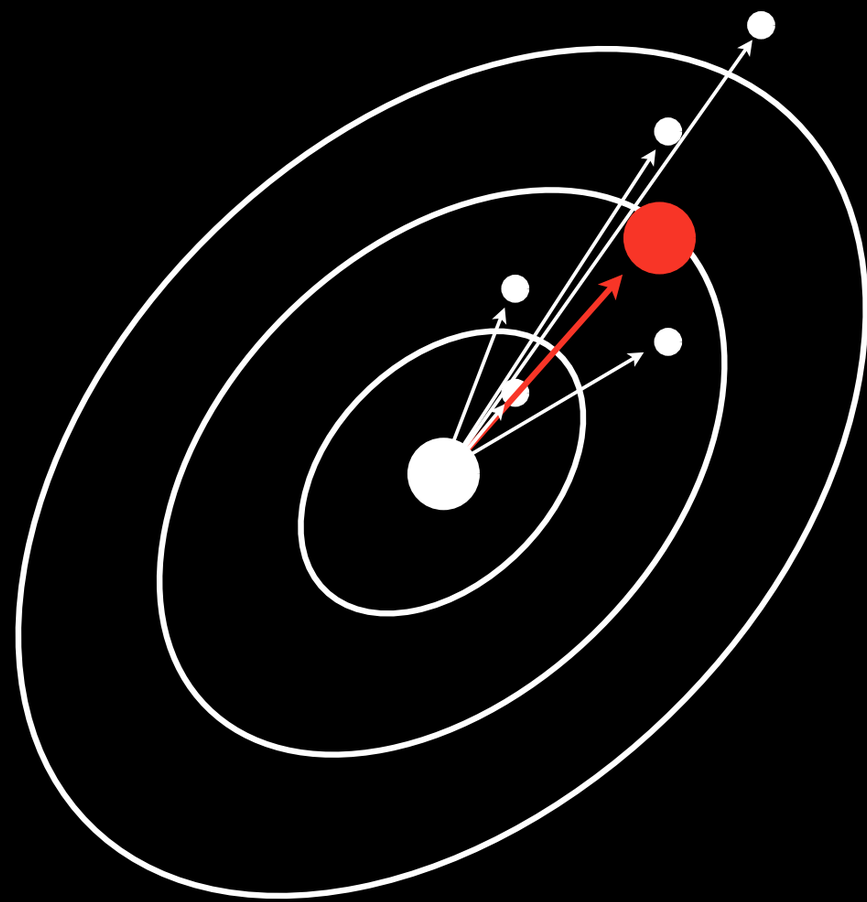
The overall model complexity is n^2 but important parts of the model can be learned in time of order n

Mutation:: Update C:: Rank-μ update

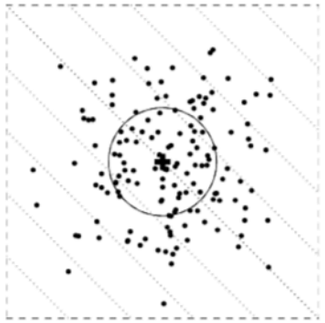
There is more information out there

$$\mathbf{Z} = \sum_{i=1}^{\mu} w_i \mathbf{z}_{i:\lambda} \mathbf{z}_{i:\lambda}^T$$

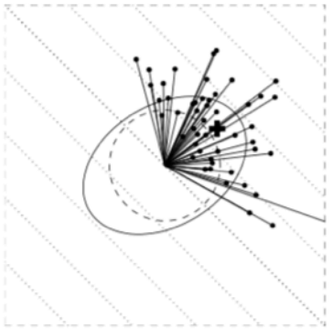
$$\mathbf{C} \longleftarrow (1 - c_{cov})\mathbf{C} + c_{cov}\mathbf{Z}$$



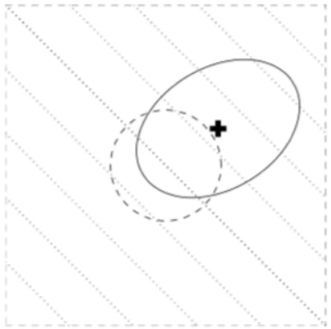
Rank-μ CMA versus Estimation of Multivariate Normal Algorithm EMNA$_{\text{global}}$[11]



$x_i = m_{\text{old}} + y_i, \quad y_i \sim \mathcal{N}(\mathbf{0}, \mathbf{C})$

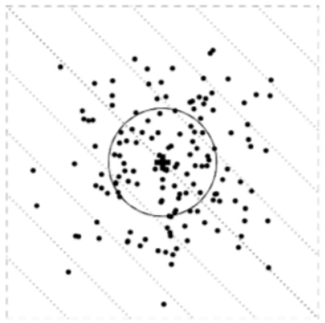


$\mathbf{C} \leftarrow \frac{1}{\mu} \sum (x_{i:\lambda} - m_{\text{old}})(x_{i:\lambda} - m_{\text{old}})^{\mathrm{T}}$

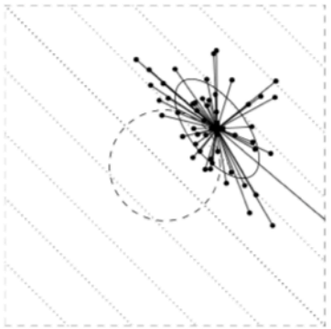


$m_{\text{new}} = m_{\text{old}} + \frac{1}{\mu} \sum y_{i:\lambda}$

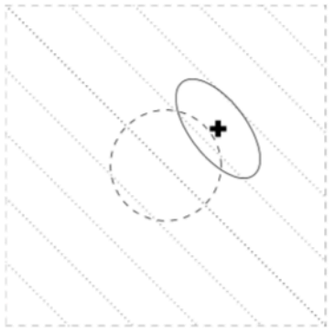
rank-μ CMA conducts a PCA **of steps**



$x_i = m_{\text{old}} + y_i, \quad y_i \sim \mathcal{N}(\mathbf{0}, \mathbf{C})$



$\mathbf{C} \leftarrow \frac{1}{\mu} \sum (x_{i:\lambda} - m_{\text{new}})(x_{i:\lambda} - m_{\text{new}})^{\mathrm{T}}$



$m_{\text{new}} = m_{\text{old}} + \frac{1}{\mu} \sum y_{i:\lambda}$

EMNA$_{\text{global}}$ conducts a PCA **of points**

sampling of $\lambda = 150$ solutions (dots)

calculating $\mathbf{C}$ from $\mu = 50$ solutions

new distribution

m_{new} is the minimizer for the variances when calculating $\mathbf{C}$

---

[11] Hansen, N. (2006). The CMA Evolution Strategy: A Comparing Review. In J.A. Lozano, P. Larranga, I. Inza and E. Bengoetxea (Eds.). Towards a new evolutionary computation. Advances in estimation of distribution algorithms. pp. 75-102

Recap::

$$\mathbf{x}_i = m + \sigma \mathbf{z}_i \qquad \mathbf{z}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{C})$$

$$\mathbf{m} \longleftarrow \mathbf{m} + \sigma \langle \mathbf{z} \rangle_w \qquad \langle \mathbf{z} \rangle_w = \sum_{i=1}^{\mu} w_i \mathbf{z}_{i:\lambda}$$

$$\mathbf{p}_c \longleftarrow (1 - c_c)\mathbf{p}_c + \sqrt{1 - (1 - c_c)^2} \sqrt{\frac{1}{\sum_{i=1}^{\mu} w_i^2}} \langle \mathbf{z} \rangle_w$$

$$\mathbf{C} \longleftarrow (1 - c_{cov})\mathbf{C}$$

$$+ c_{cov} \frac{1}{\mu_{cov}} \mathbf{p}_c \mathbf{p}_c^T \qquad \mu_{cov} = \sqrt{\frac{1}{\sum_{i=1}^{\mu} w_i^2}}$$

$$+ c_{cov} \left(1 - \frac{1}{\mu_{cov}}\right) \mathbf{Z} \qquad \mathbf{Z} = \sum_{i=1}^{\mu} w_i \mathbf{z}_{i:\lambda} \mathbf{z}_{i:\lambda}^T$$

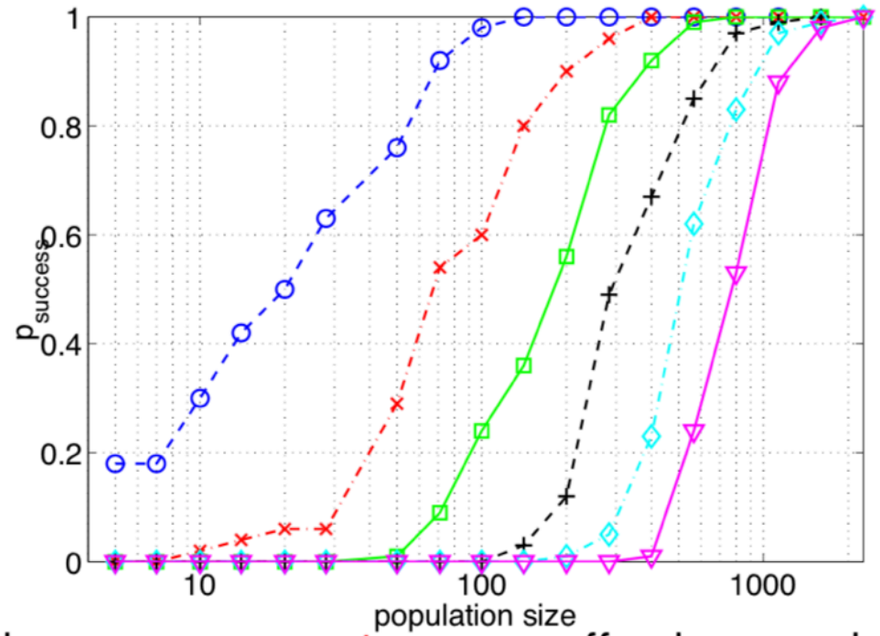
Recap:: Rank-μ update

Rank-μ increases possible learning rates

Rank-μ reduces the number of **generations** roughly from $O(n^2)$ to $O(n)$

Population Size on Multi-Modal Functions

Success Probability to Find the Global Optimum



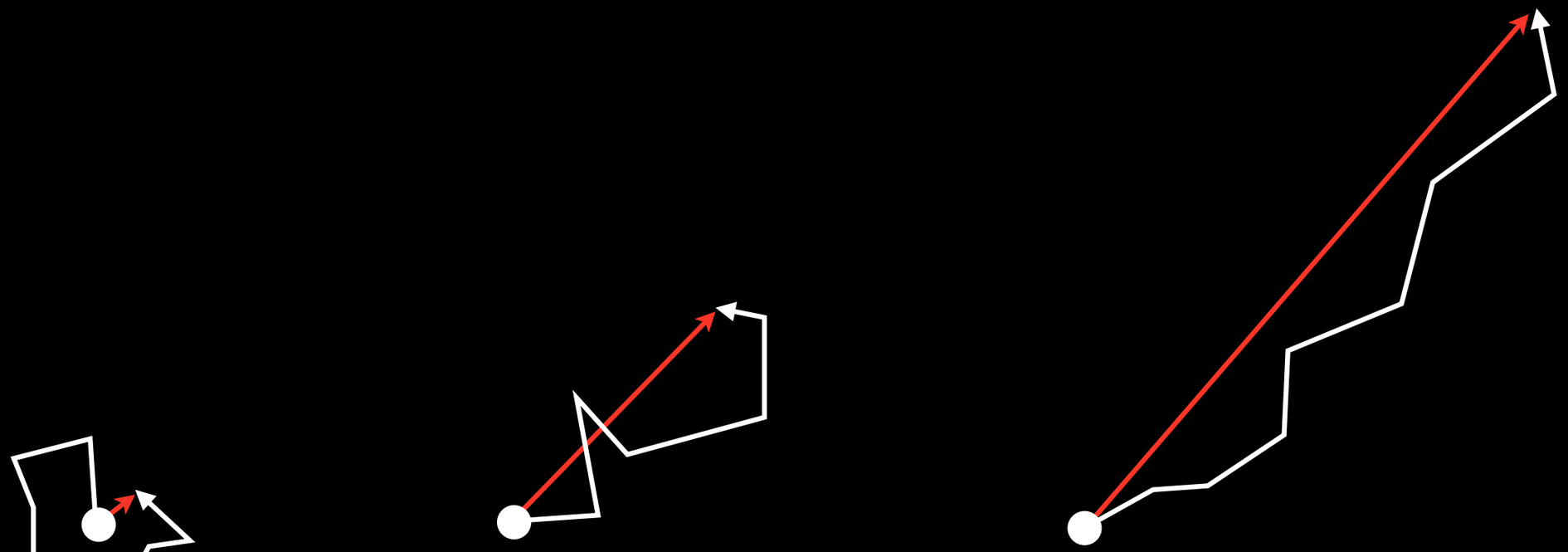
$n = 2$ ('– – ◯ – –')
$n = 5$ ('– · – × – · –')
$n = 10$ ('—□—')
$n = 20$ ('– – + – –')
$n = 40$ ('– · –◇ – ·–')
$n = 80$ ('—▽—')

Shown: success rate versus offspring population size on the highly multi-modal Rastrigins function[7]

On multi-modal functions increasing the population size can sharply increase the success probability to find the global optimum

[7] Hansen & Kern 2004. Evaluating the CMA Evolution Strategy on Multimodal Test Functions. PPSN VIII, Springer-Verlag, pp. 282-291.

Mutation:: Update σ:: Evolution path



$\sigma \downarrow$

random selection

$E||\mathcal{N}(\mathbf{0}, \mathbf{I})||$

$\sigma \uparrow$

Mutation:: Update σ:: Evolution path

$$\mathbf{p}_\sigma \longleftarrow (1-c_\sigma)\mathbf{p}_\sigma + \sqrt{1-(1-c_\sigma)^2}\sqrt{\frac{1}{\sum_{i=1}^{\mu} w_i^2}}\mathbf{C}^{-\frac{1}{2}}\langle \mathbf{z} \rangle_w$$

$$\sigma \longleftarrow \sigma e^{\left(\frac{1}{d_\sigma}\left(\frac{||p_\sigma||}{E||\mathcal{N}(\mathbf{0},\mathbf{I})||}-1\right)\right)}$$

$$> 1 \Longleftrightarrow ||p_\sigma|| > E||\mathcal{N}(\mathbf{0},\mathbf{I})||$$

Recap::

$$\mathbf{x}_i = m + \sigma \mathbf{z}_i \qquad \mathbf{z}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{C})$$

$$\mathbf{m} \longleftarrow \mathbf{m} + \sigma \langle \mathbf{z} \rangle_w \qquad \langle \mathbf{z} \rangle_w = \sum_{i=1} w_i \mathbf{z}_{i:\lambda}$$

$$\mathbf{p}_c \longleftarrow (1 - c_c)\mathbf{p}_c + \sqrt{1 - (1 - c_c)^2} \sqrt{\frac{1}{\sum_{i=1}^{\mu} w_i^2}} \langle \mathbf{z} \rangle_w$$

$$\mathbf{Z} = \sum_{i=1}^{\mu} w_i \mathbf{z}_{i:\lambda} \mathbf{z}_{i:\lambda}^T$$

$$\mathbf{C} \longleftarrow (1 - c_{cov})\mathbf{C} \quad + c_{cov} \frac{1}{\mu_{cov}} \mathbf{p}_c \mathbf{p}_c^T \quad + c_{cov} \left(1 - \frac{1}{\mu_{cov}}\right) \mathbf{Z}$$

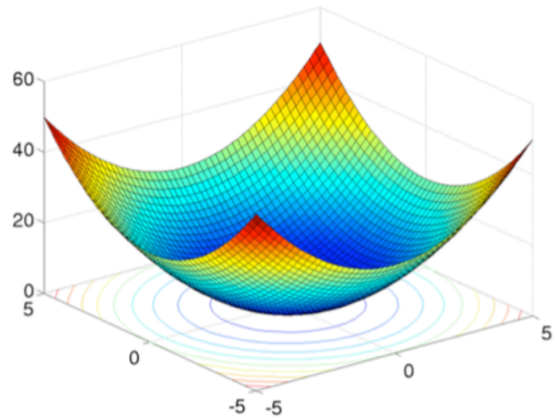
$$\mu_{cov} = \sqrt{\frac{1}{\sum_{i=1}^{\mu} w_i^2}}$$

$$\mathbf{p}_\sigma \longleftarrow (1 - c_\sigma)\mathbf{p}_\sigma + \sqrt{1 - (1 - c_\sigma)^2} \sqrt{\frac{1}{\sum_{i=1}^{\mu} w_i^2}} \mathbf{C}^{-\frac{1}{2}} \langle \mathbf{z} \rangle_w$$

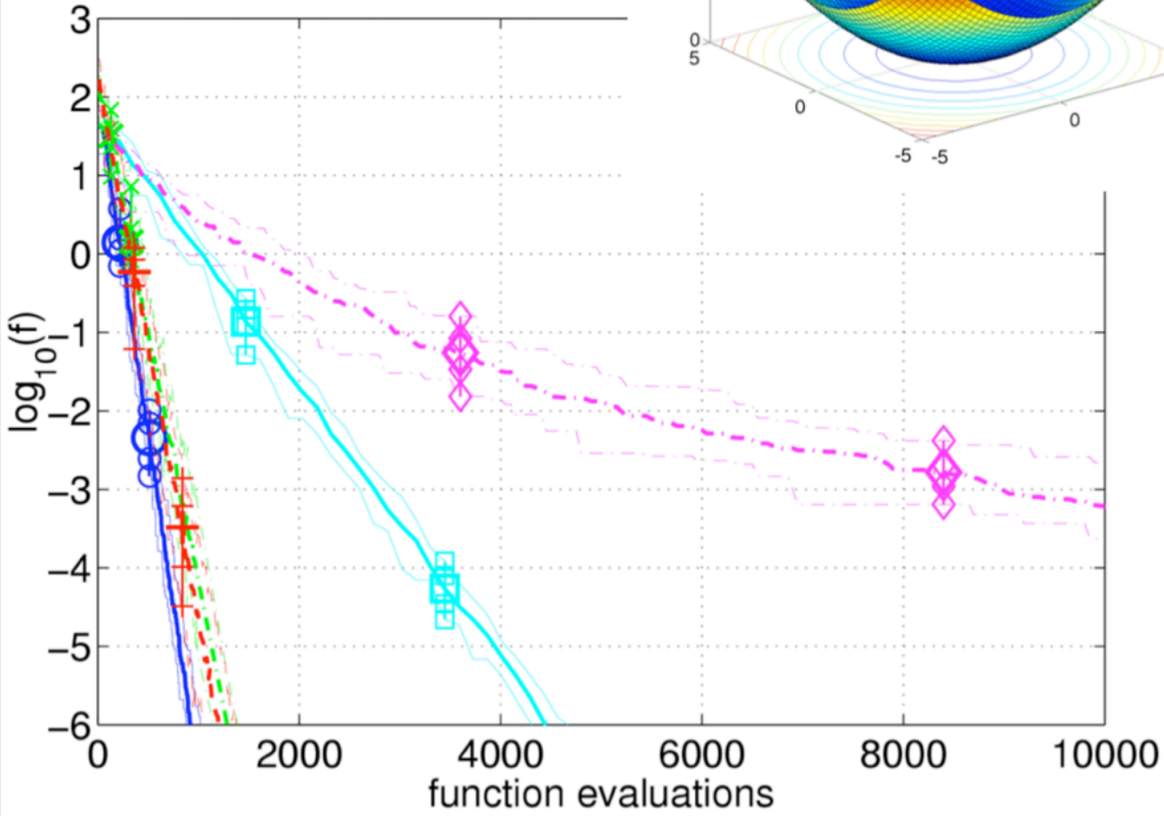
$$\sigma \longleftarrow \sigma e^{\left(\frac{1}{d_\sigma} \left(\frac{||p_\sigma||}{E||\mathcal{N}(\mathbf{0}, \mathbf{I})||} - 1\right)\right)}$$

Sphere function, 10D

$$f_{Sphere}(\mathbf{x}) = \sum_{i=1}^{n} x_i^2$$



- Separable
- Unimodal
- Well scaled

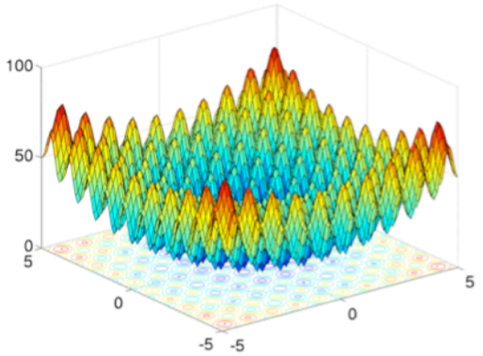


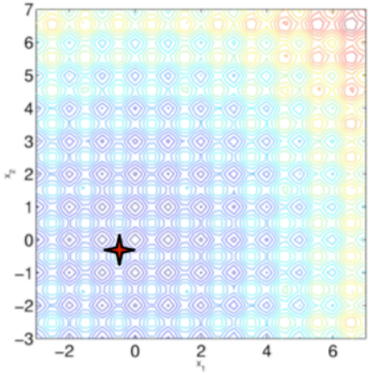
- (1+1)-ES
- (5,10)-CSA-ES
- (5,10)-CMA-ES
- IDEA, N_{base} = 200
- MBOA, N_{base} = 100

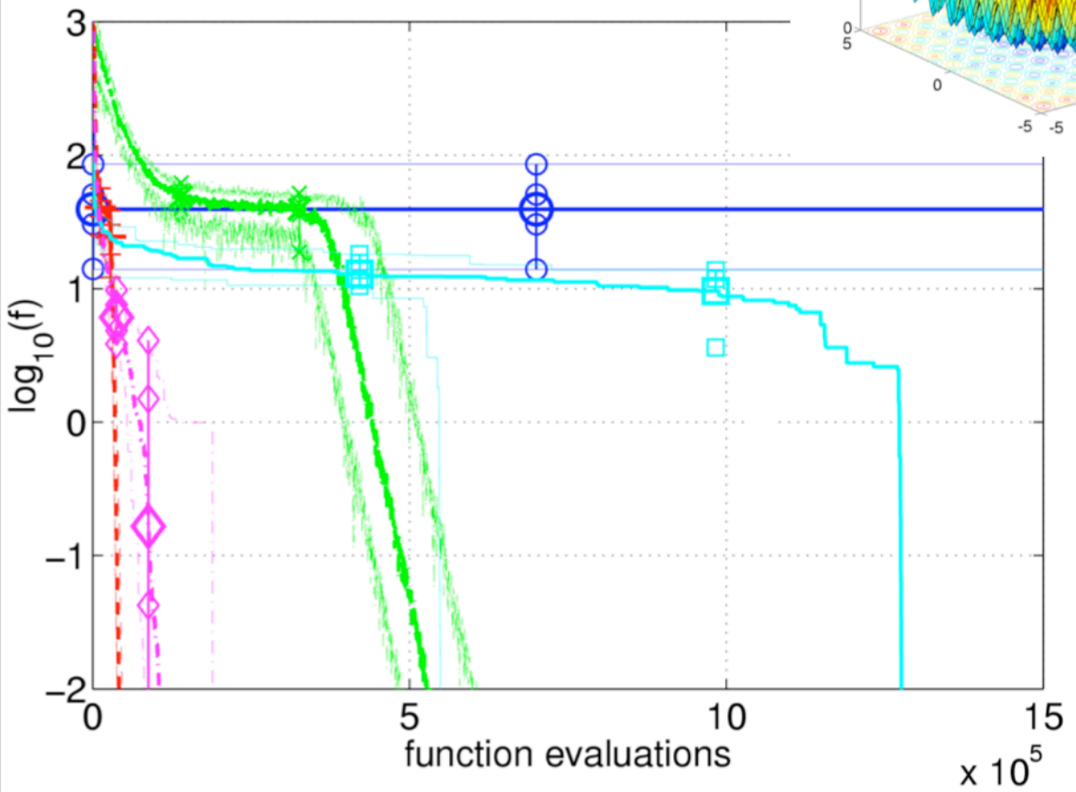
Rastrigin function, 10D

$$f_{Rastrigin}(\mathbf{x}) = 10n + \sum_{i=1}^{n} \left(x_i^2 - 10\cos(2\pi \times x_i) \right)$$

- Multimodal - Separable



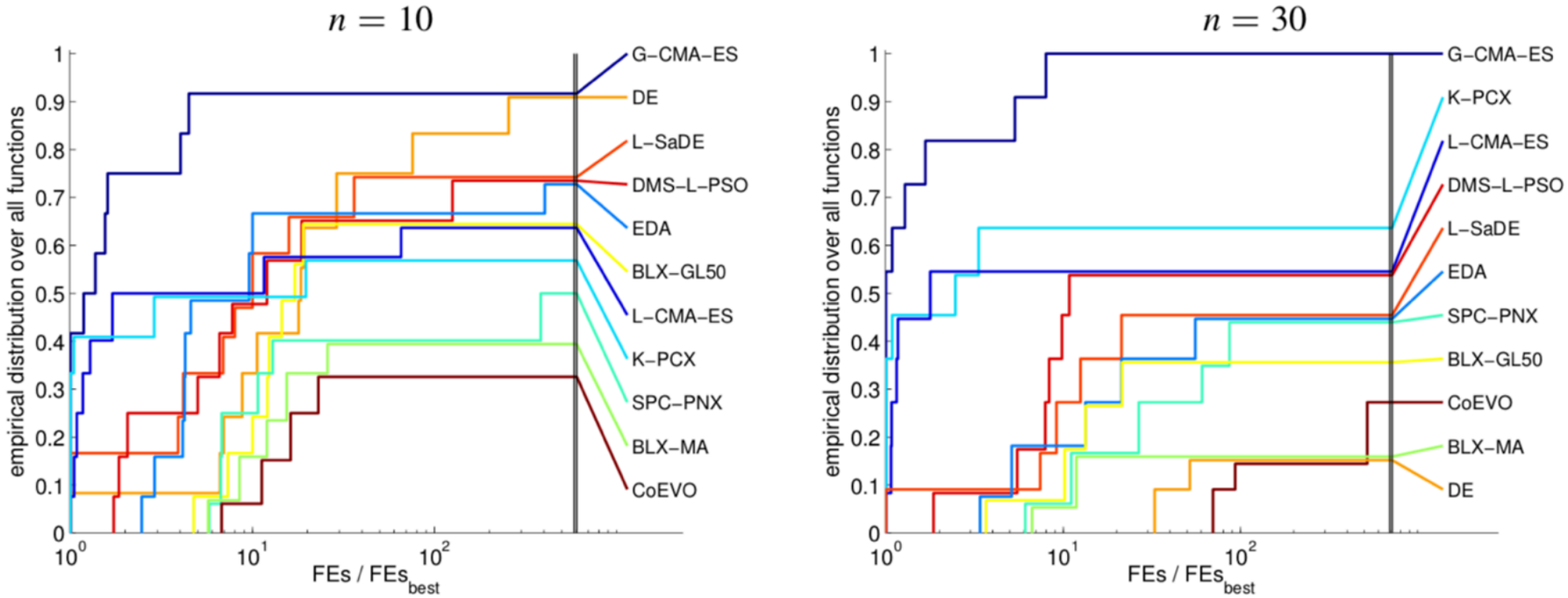




- (1+1)-ES
- (200,400)-CSA-ES
- (400,800)-CMA-ES
- IDEA, N_{base} = 400
- MBOA, N_{base} = 200

Summarized Results

Empirical Distribution of Normalized Success Performance



$\texttt{FEs} = \texttt{mean}(\#\mathit{fevals}) \times \frac{\#\text{all runs (25)}}{\#\text{successful runs}}$, where $\#\mathit{fevals}$ includes only successful runs.

Shown: **empirical distribution function** of the Success Performance $\texttt{FEs}$ divided by $\texttt{FEs}$ of the best algorithm on the respective function.

Results of all functions are used where at least one algorithm was successful at least once, i.e. where the target function value was reached in at least one experiment (out of 11×25 experiments).

Small values for $\texttt{FEs}$ and therefore large (cumulative frequency) values in the graphs are preferable.

CMA internal parameters::

Lots of pain, testing and benchmarking

Just believe

Recap::

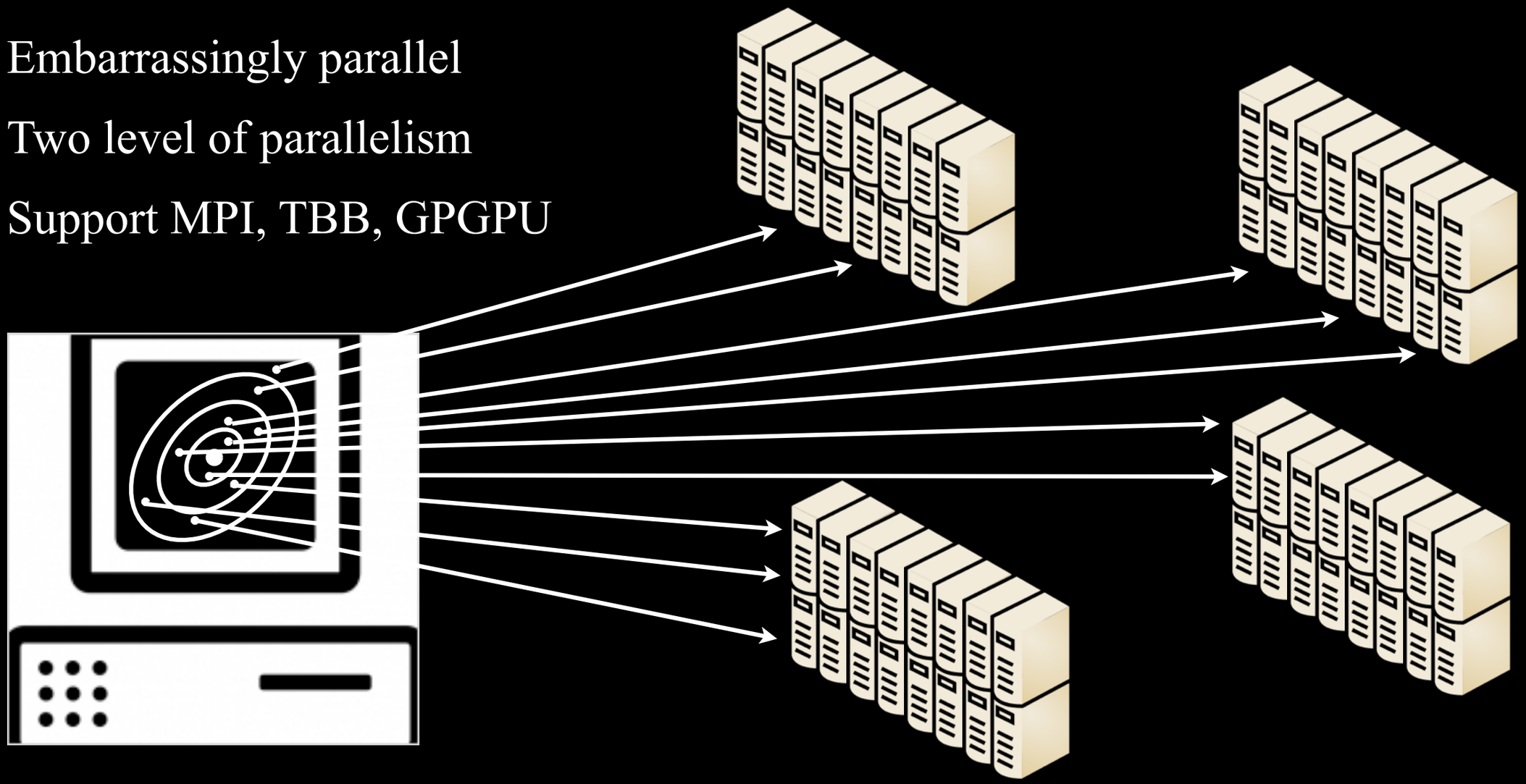
Overall control over on normal multivariate scaling

Prevents premature convergence

On quadratic functions step length is close to optimal

Multihost CMA-ES::

- Embarrassingly parallel
- Two level of parallelism
- Support MPI, TBB, GPGPU



Conclusions::

- Multivariate normal distribution sampling
- Rank based selection
- Covariance matrix adaptation to maximize probability of repeating successful steps
- Step length control
- Allows massively parallel computation

CMA-ES is the algorithm of choice for moderate dimensional non-linear, non-convex, non-separable, multimodal, noisy and discontinuous real valued functions

References::

- We credit Nikolaus Hansen for many of the slides used in the CMA discussion, and much of the material is based on his publications and software page
- Hansen, Nikolaus, and Andreas Ostermeier. "Completely derandomized self-adaptation in evolution strategies." Evolutionary computation 9.2 (2001): 159-195.
- Hansen, Nikolaus. "The CMA evolution strategy: a comparing review." Towards a new evolutionary computation. Springer, Berlin, Heidelberg, 2006. 75-102.
- Hansen, Nikolaus. "The CMA evolution strategy: A tutorial." arXiv preprint arXiv: 1604.00772 (2016).