Algorytm CMA-ES i jego wybrane rozszerzenia

Mateusz Zaborski
M.Zaborski@mini.pw.edu.pl

Faculty of Mathematics and Information Science
Warsaw University of Technology

16.12.2020

Table of Contest

Black Box optimization

Black Box optimization problem

- Goal - minimize an objective function (or fitness function or cost function)

$$f : \mathbb{R}^n \to \mathbb{R}$$

- Single-objective optimization
- Black Box scenario
  - Function values of evaluated search points are the only accessible information
  - Gradients are not available
- Search cost - number of function evaluations

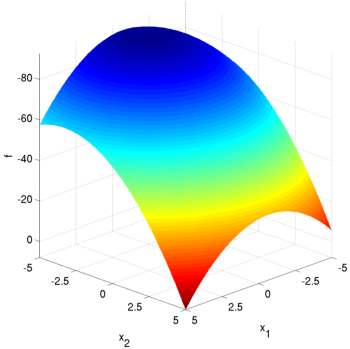


Figure 1: Sphere 2D function from COCO BBOB benchmark [8]

Black Box optimization

Applications

Applications:

- Aerodynamics - shape optimization (simulation-based)
- Economics - e.g. portfolio selection
- Operations research
- Model selection
- Hyperparameter tuning
- etc.

Black Box optimization

Performance measures

Performance measures:

- **Number of function evaluations (per target)**
  - Especially for time-consuming evaluations
- Computation time
- Final target reached
- Memory consumption
- Performance for various functions classes

Black Box optimization

Problems

Problems:

- Infinite number of solutions in a continuous domain
- Multidimensional problems are difficult for grid search
- The goal is to find a global (not local) optimum
- Unknown function shape
  - Non-linear, non-quadratic
  - Discontinuities, sharp ridges
  - Non-separability
  - Ill-conditioning

Black Box optimization problem

Examples

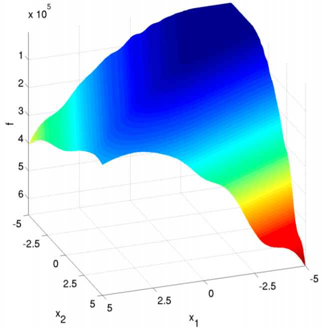


Figure 2: Attractive Sector Function (2D)[8]

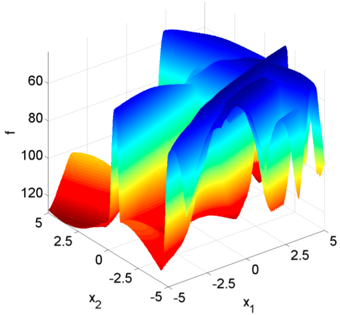


Figure 3: Gallagher's Gaussian 21-hi Peaks Function (2D)[8]

Black Box optimization problem

Examples

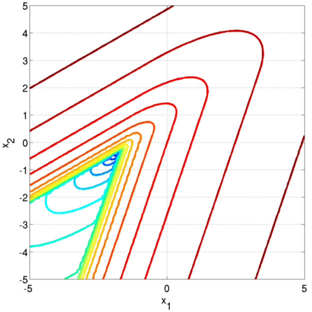


Figure 4: Attractive Sector Function (2D)[8]

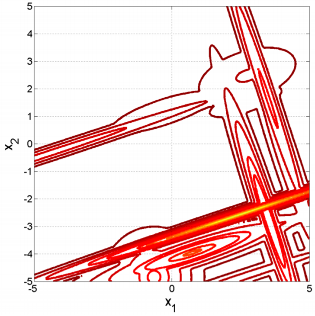


Figure 5: Gallagher's Gaussian 21-hi Peaks Function (2D)[8]

Optimization techniques

Optimization techniques:

- Random / Monte Carlo
- Grid search
- Simulated annealing
- Bayesian optimization
- Swarm-based algorithms (e.g. PSO)
- Evolutionary algorithms (e.g. DE, CMA-ES)
- Surrogate optimization
- etc.

Benchmarking

COCO BBOB benchmark

- Platform for optimizer comparition
- 20–100 functions (24 noiselessall)
- 5–15 instances for each function
- 2, 3, 5, 10, 20, 40 dimensions
- Up to 100 targets per instance

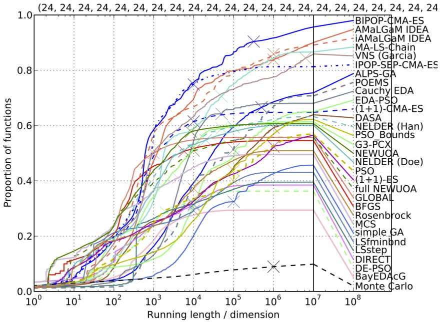


Figure 6: Example empirical runtime distributions from COCO BBOB benchmark [8]

CMA-ES algorithm

Covariance Matrix Adaptation Evolution Strategy (CMA-ES)

Completely Derandomized Self-Adaptation in Evolution Strategies

Nikolaus Hansen and Andreas Ostermeier

In *Evolutionary Computation*, 9(2), pp. 159-195 (2001)

Figure 7: Covariance Matrix Adaptation Evolution Strategy (CMA-ES) idea[9]

CMA-ES

Algorithm idea

Initialize distribution parameters $\boldsymbol{\theta}^{(0)}$
For generation $g = 0, 1, 2, \ldots$
  Sample λ independent points from distribution $P\left(\boldsymbol{x}|\boldsymbol{\theta}^{(g)}\right) \to \boldsymbol{x}_1, \ldots, \boldsymbol{x}_\lambda$
  Evaluate the sample $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_\lambda$ on f
  Update parameters $\boldsymbol{\theta}^{(g+1)} = F_\theta(\boldsymbol{\theta}^{(g)}, (\boldsymbol{x}_1, f(\boldsymbol{x}_1)), \ldots, (\boldsymbol{x}_\lambda, f(\boldsymbol{x}_\lambda)))$
  break, if termination criterion met

Figure 8: CMA-ES as randomized black box search algorithm[6]

CMA-ES

Multivariate normal distribution

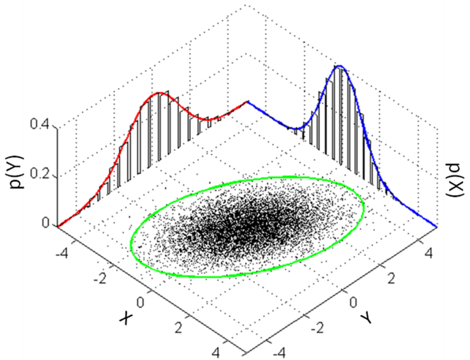


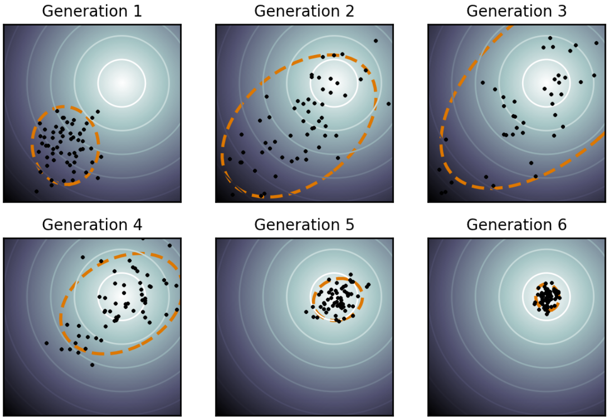
Figure 9: Source: [4]

$$m = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 3/5 \\ 3/5 & 2 \end{bmatrix}$$

CMA-ES

Algorithm idea [3]



CMA-ES

Sampling[6]

$$x_k^{(g+1)} \sim m^{(g)} + \sigma^{(g)} \mathcal{N}\left(0, C^{(g)}\right) \qquad \text{for } k = 1, \ldots, \lambda$$

$\mathcal{N}(0, C^{(g)})$ is a multivariate normal distribution with zero mean and covariance matrix $C^{(g)}$, see Sect. 0.2. It holds $m^{(g)} + \sigma^{(g)} \mathcal{N}(0, C^{(g)}) \sim \mathcal{N}\left(m^{(g)}, (\sigma^{(g)})^2 C^{(g)}\right)$.

$x_k^{(g+1)} \in \mathbb{R}^n$, k-th offspring (individual, search point) from generation $g + 1$.

$m^{(g)} \in \mathbb{R}^n$, mean value of the search distribution at generation g.

$\sigma^{(g)} \in \mathbb{R}_{>0}$, "overall" standard deviation, step-size, at generation g.

$C^{(g)} \in \mathbb{R}^{n \times n}$, covariance matrix at generation g. Up to the scalar factor ${\sigma^{(g)}}^2$, $C^{(g)}$ is the covariance matrix of the search distribution.

$\lambda \geq 2$, population size, sample size, number of offspring.

CMA-ES

Adaptive components

Adaptive components:

- Mean
- Covariance matrix
  - rank-*one*-update (evolution path p_c)
  - rank-μ-update
- Step-size
- Population size (in *IPOP-CMA-ES*[1])

CMA-ES

Mean update[6]

$$\boldsymbol{m}^{(g+1)} = \sum_{i=1}^{\mu} w_i \, \boldsymbol{x}_{i:\lambda}^{(g+1)}$$

$$\sum_{i=1}^{\mu} w_i = 1, \qquad w_1 \geq w_2 \geq \cdots \geq w_\mu > 0$$

$$\boldsymbol{m}^{(g+1)} = \boldsymbol{m}^{(g)} + c_{\mathrm{m}} \sum_{i=1}^{\mu} w_i \left(\boldsymbol{x}_{i:\lambda}^{(g+1)} - \boldsymbol{m}^{(g)} \right)$$

Usually $c_m = 1$

CMA-ES

Covariance matrix update

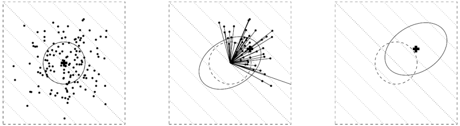


Figure 10: Rank-μ-update idea[5]

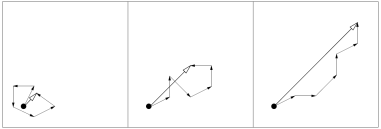


Figure 11: Rank-*one*-update (evolution path) idea[5]

CMA-ES

Covariance matrix update[6]

$$\boldsymbol{C}^{(g+1)} = \underbrace{(1 - c_1 - c_\mu \textstyle\sum w_j)}_{\text{can be close or equal to } 0} \boldsymbol{C}^{(g)} + c_1 \underbrace{\boldsymbol{p}_c^{(g+1)} \boldsymbol{p}_c^{(g+1)^\mathsf{T}}}_{\text{rank-one update}} + \underbrace{c_\mu \sum_{i=1}^{\lambda} w_i \, \boldsymbol{y}_{i:\lambda}^{(g+1)} \left(\boldsymbol{y}_{i:\lambda}^{(g+1)}\right)^\mathsf{T}}_{\text{rank-}\mu\text{ update}}$$

$$\boldsymbol{p}_c^{(g+1)} = (1 - c_c)\boldsymbol{p}_c^{(g)} + \sqrt{c_c(2 - c_c)\mu_{\text{eff}}} \, \frac{\boldsymbol{m}^{(g+1)} - \boldsymbol{m}^{(g)}}{\sigma^{(g)}}$$

$$c_1 \approx 2/n^2$$

$$c_\mu \approx \min(\mu_{\text{eff}}/n^2, 1 - c_1)$$

$$\boldsymbol{y}_{i:\lambda}^{(g+1)} = (\boldsymbol{x}_{i:\lambda}^{(g+1)} - \boldsymbol{m}^{(g)})/\sigma^{(g)}$$

$$\textstyle\sum w_j = \sum_{i=1}^{\lambda} w_i \approx -c_1/c_\mu$$

CMA-ES

Step size

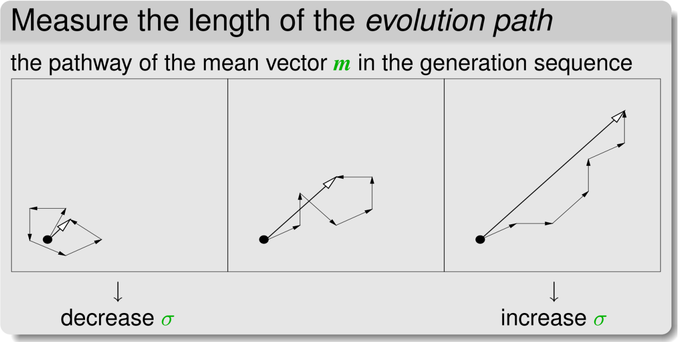


Figure 12: Step size adaptation idea[2]

CMA-ES

Conjugate evolution path[6]

$$p_\sigma^{(g+1)} = (1 - c_\sigma) p_\sigma^{(g)} + \sqrt{c_\sigma (2 - c_\sigma) \mu_{\text{eff}}} \, {C^{(g)}}^{-\frac{1}{2}} \frac{m^{(g+1)} - m^{(g)}}{\sigma^{(g)}} \tag{31}$$

where

$p_\sigma^{(g)} \in \mathbb{R}^n$ is the conjugate evolution path at generation g.

$c_\sigma < 1$. Again, $1/c_\sigma$ is the backward time horizon of the evolution path (compare (20)). For small μ_{eff}, a time horizon between $\sqrt{n}$ and n is reasonable.

$\sqrt{c_\sigma (2 - c_\sigma) \mu_{\text{eff}}}$ is a normalization constant, see (24).

${C^{(g)}}^{-\frac{1}{2}} \overset{\text{def}}{=} B^{(g)} {D^{(g)}}^{-1} {B^{(g)}}^{\mathsf{T}}$, where $C^{(g)} = B^{(g)} \left(D^{(g)}\right)^2 {B^{(g)}}^{\mathsf{T}}$ is an eigendecomposition of $C^{(g)}$, where $B^{(g)}$ is an orthonormal basis of eigenvectors, and the diagonal elements of the diagonal matrix $D^{(g)}$ are square roots of the corresponding positive eigenvalues (cf. Sect. 0.1).

CMA-ES

Step size[6]

$$\sigma^{(g+1)} = \sigma^{(g)} \exp\left(\frac{c_\sigma}{d_\sigma}\left(\frac{\|\boldsymbol{p}_\sigma^{(g+1)}\|}{\mathsf{E}\|\mathcal{N}(\mathbf{0},\mathbf{I})\|} - 1\right)\right)$$

Final sampling rule:

$$\boldsymbol{x}_k^{(g+1)} \sim \boldsymbol{m}^{(g)} + \sigma^{(g)}\mathcal{N}\left(\mathbf{0}, \boldsymbol{C}^{(g)}\right) \qquad \text{for } k = 1, \ldots, \lambda$$

CMA-ES extensions

CMA-ES extensions

Main ideas:

- Parameter modification
- Covariance matrix adaptation modification (e.g. including worst samples)
- **Restarts**
- **Increasing population size**
- **Surrogate models**

IPOP-CMA-ES[1]

Restart criteria:

- Stop if the range of the best objective function values of the last $10 + \lceil 30n/ \rceil$ generations is zero (`equalfunvalhist`), or the range of these function values and all function values of the recent generation is below `Tolfun`$= 10^{-12}$.
- Stop if the standard deviation of the normal distribution is smaller than `TolX` in all coordinates and $\vec{p}_c$ (the evolution path from Eq. 2 in [3]) is smaller than `TolX` in all components. We set `TolX`$= 10^{-12}\ ^{(0)}$.
- Stop if adding a 0.1-standard deviation vector in a principal axis direction of $\boldsymbol{C}^{(g)}$ does not change $\langle \vec{x} \rangle_{\mathrm{W}}^{(g)}$ (`noeffectaxis`)[3]
- Stop if adding 0.2-standard deviation in each coordinate does change $\langle \vec{x} \rangle_{\mathrm{W}}^{(g)}$ (`noeffectcoord`).
- Stop if the condition number of the covariance matrix exceeds 10^{14} (`conditioncov`).

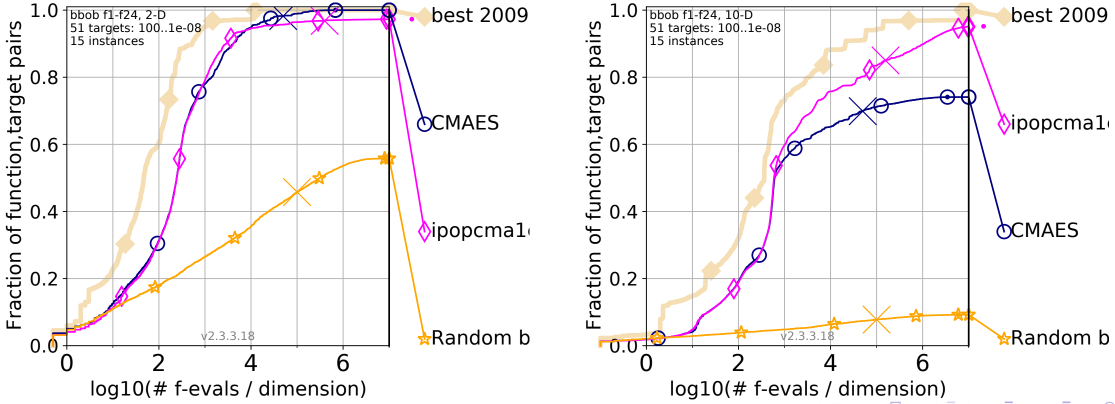
A Restart CMA Evolution Strategy With Increasing Population Size

Anne Auger
CoLab Computational Laboratory,
ETH Zürich, Switzerland
Anne.Auger@inf.ethz.ch

Nikolaus Hansen
CSE Lab,
ETH Zürich, Switzerland
Nikolaus.Hansen@inf.ethz.ch

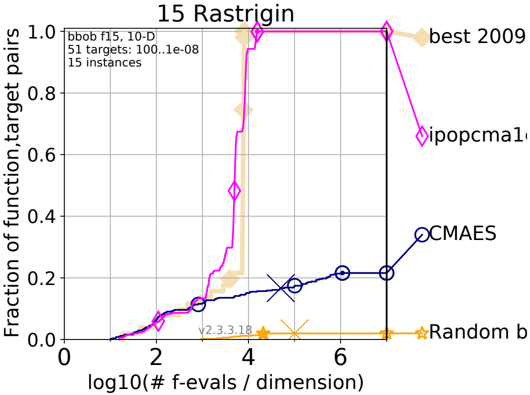
IPOP-CMA-ES[1]

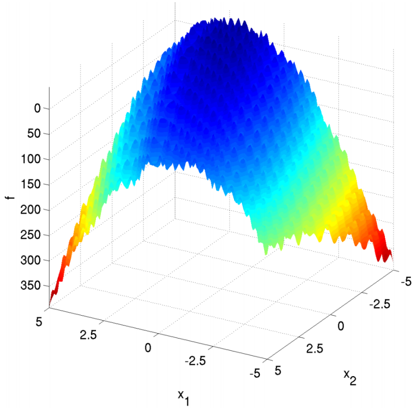
Comparison



IPOP-CMA-ES[1]

Comparison





LQ-CMA-ES[7]

Key concept:

- Linear / quadratic surrogate model added to the CMA-ES
- The purpose of the model is to reduce the search cost (true evaluations)
- Weighted linear regression is used to obtain coefficients
- Rank correlation is used as meta-model quality measure
- Model optimum is injected to the population in the next iteration

A Global Surrogate Assisted CMA-ES

Nikolaus Hansen
Inria & Ecole polytechnique
Palaiseau, France
forename.lastname@inria.fr

LQ-CMA-ES[7]

Algorithm 1 Determine Population Surrogate Values

Require: A population $X = \boldsymbol{x}_1, \ldots, \boldsymbol{x}_\lambda$, a model $\mathcal{M}$ with a data queue of at most $\max(\lambda, 2\mathrm{df}_{\max})$ pairs $(\boldsymbol{y}_i, f(\boldsymbol{y}_i))$, and a fitness function f

1: $k \leftarrow \lfloor 1 + \max(\lambda \times 2\%, 3/0.75 - |\mathcal{M}|) \rfloor$ # incrementing evaluations
2: **while** $|X| > 0$ **do** # while not all elements are added to $\mathcal{M}$
3: drop the $k - (\lambda - |X|)$ $\mathcal{M}$-best elements from X into $\mathcal{M}$
4: sort the newest $\min(k, \lambda)$ elements in $\mathcal{M}$ w.r.t. f # last = best
5: $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_j \leftarrow$ the last $\max(15, \min(1.2k, 0.75\lambda))$ elements in $\mathcal{M}$
6: **if** Kendall-$\tau([\mathcal{M}(\boldsymbol{y}_i)]_i, [f(\boldsymbol{y}_i)]_i) \geq 0.85$ **then**
7: **break while**
8: $k \leftarrow \lceil 1.5k \rceil$
9: **if** $|X| > 0$ **then**
10: **return** $\mathcal{M}(\boldsymbol{x}_1), \ldots, \mathcal{M}(\boldsymbol{x}_\lambda)$ all offset by
11: $\min_{\boldsymbol{x} \in \text{last } k \text{ elements of } \mathcal{M}}(f(\boldsymbol{x})) - \min_{i=1\ldots\lambda}(\mathcal{M}(\boldsymbol{x}_i))$
12: **else**
13: **return** $f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_\lambda)$

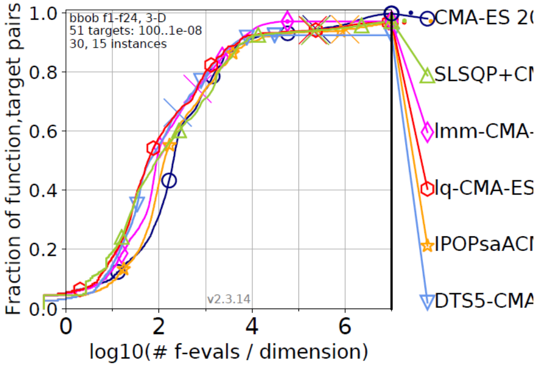
$$z_{\text{lin}} : \boldsymbol{x} \mapsto [1, x_1, x_2, \ldots, x_n]^\top$$

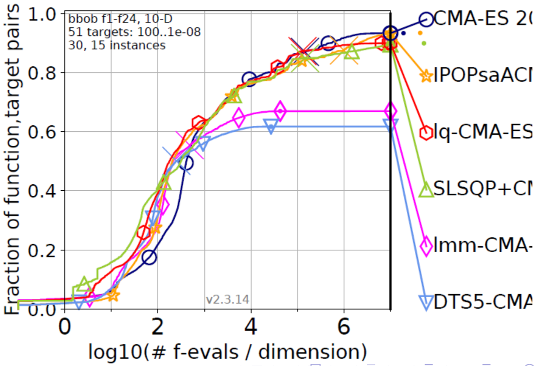
$$z_{\text{quad}} : \boldsymbol{x} \mapsto [z_{\text{lin}}(\boldsymbol{x})^\top, x_1^2, \ldots, x_n^2]^\top$$

$$z_{\text{full}} : \boldsymbol{x} \mapsto [z_{\text{quad}}(x)^\top, x_1x_2, x_1x_3, \ldots, x_1x_n, x_2x_3, \ldots x_2x_n, x_3x_4, \ldots, x_{n-1}x_n]^\top$$

LQ-CMA-ES[7]

Comparison





Experiments

CMA-ES + Quadratic model + rotation + R^2

Concept

- CMA-ES-based with simple restart criterion and population doubling
- Add FIFO archive evaluated samples (size: $5 * \lambda$)
- Extend (conditionally) population with $\lambda + 1$ offspring
- R^2 criterion
- $\lambda + 1$ offspring is designated by the hyper-parabola peak
- Transform archive population using B^T matrix from SVD decomposition of covariance matrix C

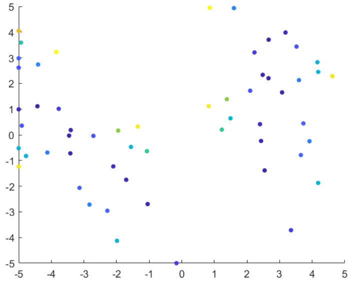
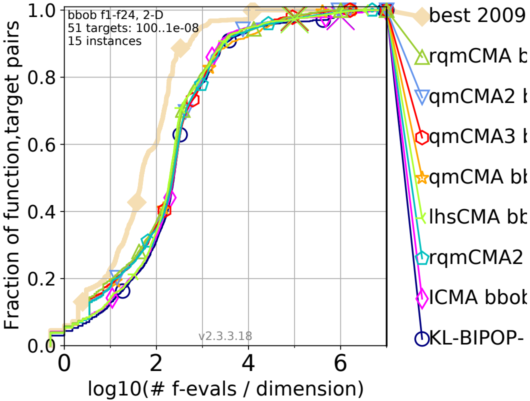
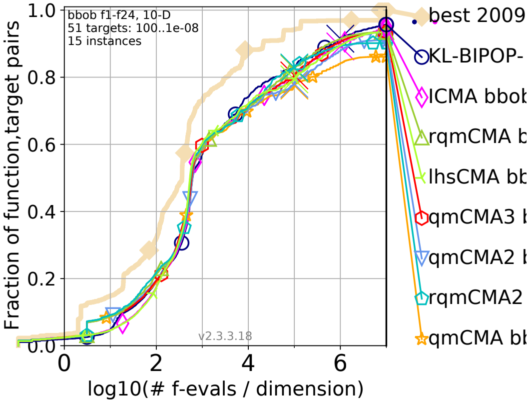


Figure 13: Step Ellipsoidal example

CMA-ES + Quadratic model + rotation + R^2

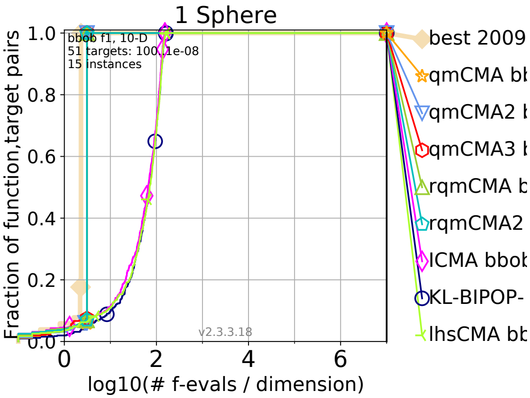
Comparison

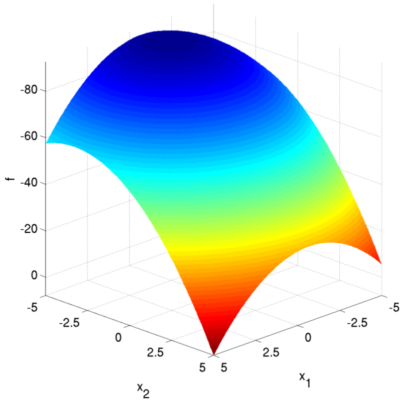




CMA-ES + Quadratic model + rotation + R^2

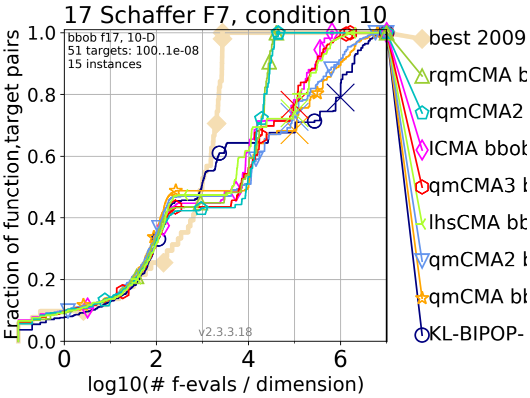
Comparison

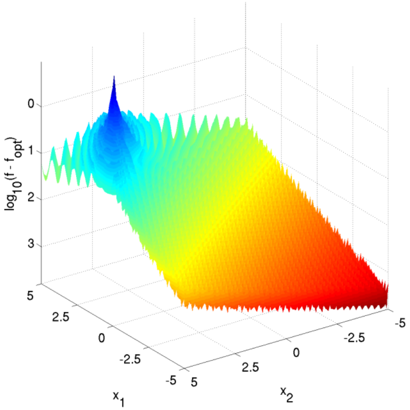




CMA-ES + Quadratic model + rotation + R^2

Comparison





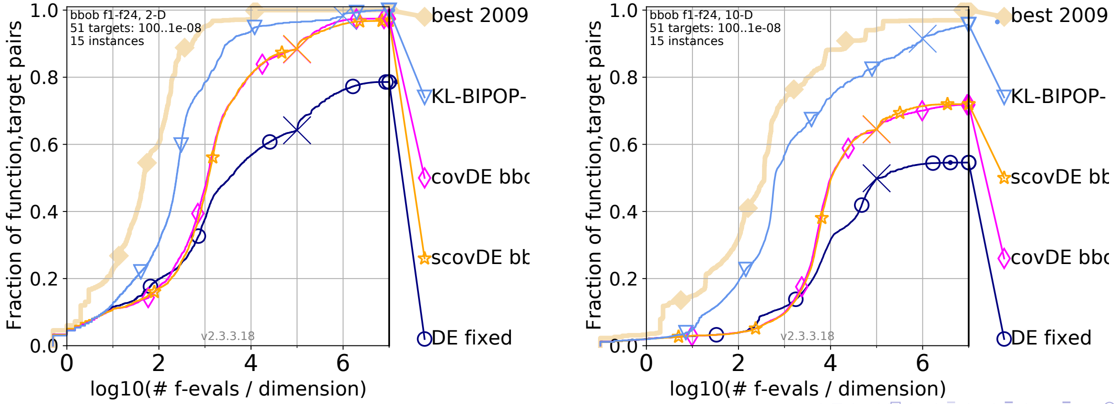
DE + rotation

Concept:

- Differential Evolution based with simple restart criterion
- Covariance matrix C calculated after each iteration using λ offspring (current population)
- Covariance matrix C arithmetic smoothing is possible $(0.2 * C^g)$
- Covariance matrix C SVD to get transformation (rotation) component
- Transformation is done before mutation
- Reverse-transformation is done before selection

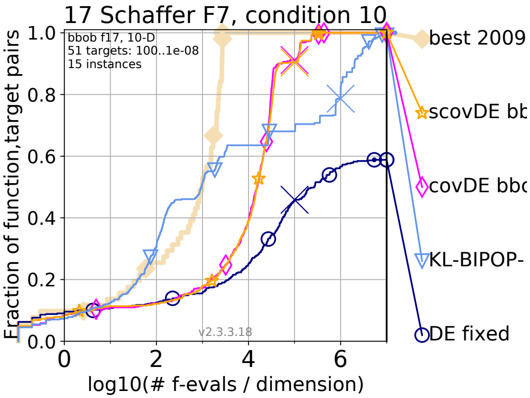
DE + rotation

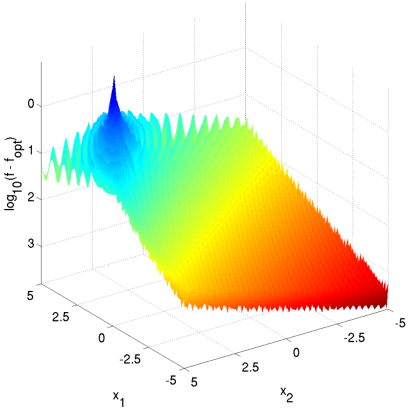
Comparison



DE + rotation

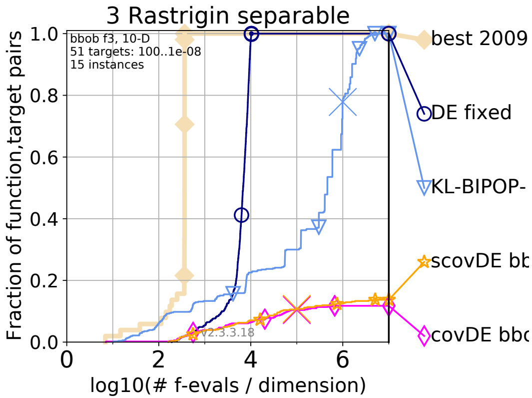
Comparison

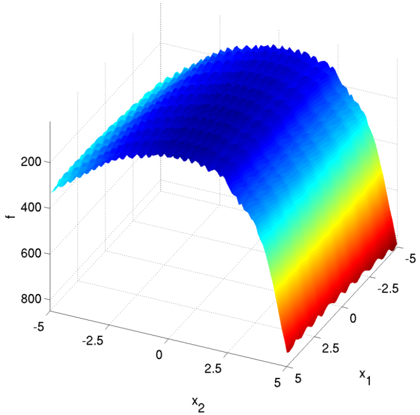




DE + rotation

Comparison





Summary

- CMA-ES is a powerful optimization algorithm
- CMA-ES behaves well on strong global structure functions
- CMA-ES can designate optimum of rotated hyper-parabola by its normal sampling procedure
- Rotation concept can be applied into Differential Evolution
- State-of-the-art version of DE such as L-SHADE should enhanced with rotation component

Bibliography

- Auger, A., Hansen, N.: A restart cma evolution strategy with increasing population size. In: 2005 IEEE congress on evolutionary computation. vol. 2, pp. 1769–1776. IEEE (2005)
- Auger, A., Hansen, N.: Tutorial cma-es: evolution strategies and covariance matrix adaptation. In: Proceedings of the 14th annual conference companion on Genetic and evolutionary computation. pp. 827–848 (2012)
- Commons, W.: Concept of directional optimization in cma-es algorithm (2020), https://commons.wikimedia.org/wiki/File:Concept_of_directional_optimization_in_CMA-ES_algorithm.png
- Commons, W.: Illustration of a multivariate gaussian distribution and its marginals. (2020), https://en.wikipedia.org/wiki/Multivariate_normal_distribution#/media/File:MultivariateNormal.png
- Hansen, N.: The cma evolution strategy: a comparing review. In: Towards a new evolutionary computation, pp. 75–102. Springer (2006)
- Hansen, N.: The cma evolution strategy: A tutorial. arXiv preprint arXiv:1604.00772 (2016)
- Hansen, N.: A global surrogate assisted cma-es. In: Proceedings of the Genetic and Evolutionary Computation Conference. pp. 664–672 (2019)
- Hansen, N., Brockhoff, D., Mersmann, O., Tusar, T., Tusar, D., ElHara, O.A., Sampaio, P.R., Atamna, A., Varelas, K., Batu, U., Nguyen, D.M., Matzner, F., Auger, A.: COmparing Continuous Optimizers: numbbo/COCO on Github (2019). https://doi.org/10.5281/zenodo.2594848, https://doi.org/10.5281/zenodo.2594848
- Hansen, N., Ostermeier, A.: Completely derandomized self-adaptation in evolution strategies. Evolutionary computation **9**(2), 159–195 (2001)