Surrogate Modeling and Landscape Analysis for Evolutionary Black-box Optimization

Zbyněk Pitra

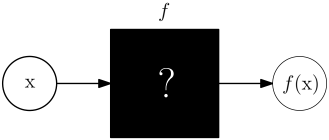
Supervisor: Martin Holeňa

Faculty of Nuclear Sciences and Physical Engineering, Czech Technical University

Prague, Czech Republic

2022

Continuous Black-box Optimization

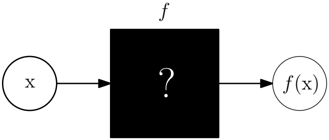


- objective function evaluated empirically or through simulations
- optimization (minimization) is finding such $\mathbf{x}^\star \in \mathbb{R}^n$ that

$$f(\mathbf{x}^\star) = \min_{\forall \mathbf{x} \in \mathbb{R}^n} f(\mathbf{x})$$

- expensive scenario – limited number of evaluations

Continuous Black-box Optimization

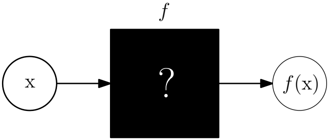


- objective function evaluated empirically or through simulations
- optimization (minimization) is finding such $\mathbf{x}^\star \in \mathbb{R}^n$ that

$$f(\mathbf{x}^\star) = \min_{\forall \mathbf{x} \in \mathbb{R}^n} f(\mathbf{x})$$

- expensive scenario – limited number of evaluations

Continuous Black-box Optimization



- objective function evaluated empirically or through simulations
- optimization (minimization) is finding such $\mathbf{x}^\star \in \mathbb{R}^n$ that

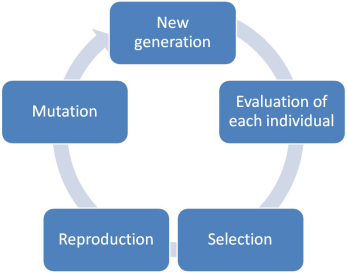
$$f(\mathbf{x}^\star) = \min_{\forall \mathbf{x} \in \mathbb{R}^n} f(\mathbf{x})$$

- expensive scenario – limited number of evaluations

EVOLUTIONARY ALGORITHMS AND SURROGATE MODELING

Evolutionary Algorithms

- escape from local optima
- require many function evaluations

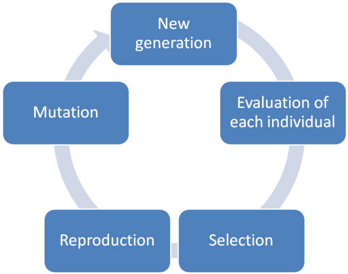


sicara.ai

Evolutionary Algorithms and Surrogate Modeling

Evolutionary Algorithms

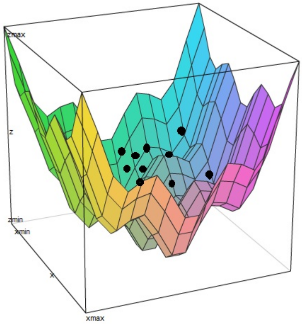
- escape from local optima
- require many function evaluations



sicara.ai

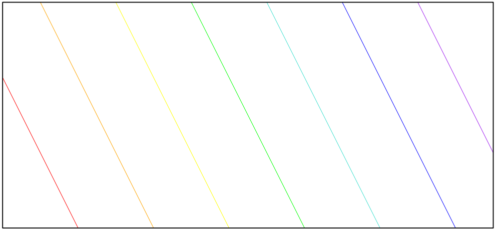
Surrogate Modeling

- approximating regression model
- not expensive
- less accurate



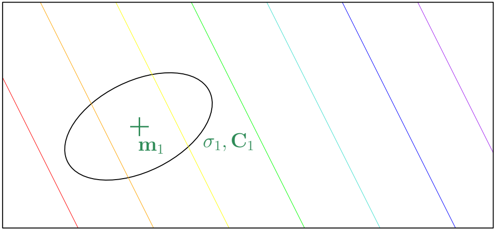
CMA-ES

Input: $\mathbf{m} \in \mathbb{R}^n, \sigma \in \mathbb{R}_+, \lambda \in \mathbb{N}$
Initialize: $\mathbf{C} = \mathbf{I}$ (and several other parameters)
Set the weights $w_1, \ldots w_\lambda$ appropriately



CMA-ES

Input: $\mathbf{m} \in \mathbb{R}^n, \sigma \in \mathbb{R}_+, \lambda \in \mathbb{N}$
Initialize: $\mathbf{C} = \mathbf{I}$ (and several other parameters)
Set the weights $w_1, \dots w_\lambda$ appropriately

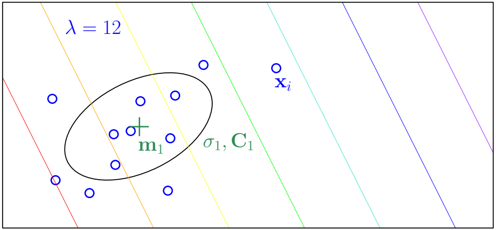


CMA-ES

Input: $\mathbf{m} \in \mathbb{R}^n, \sigma \in \mathbb{R}_+, \lambda \in \mathbb{N}$
Initialize: $\mathbf{C} = \mathbf{I}$ (and several other parameters)
Set the weights $w_1, \ldots w_\lambda$ appropriately

while not terminate

1. $\mathbf{x}_i = \mathbf{m} + \sigma \mathbf{y}_i, \quad \mathbf{y}_i \sim N(\mathbf{0}, \mathbf{C}), \quad$ for $i = 1, \ldots, \lambda$ sampling
2. $\mathbf{m} \leftarrow \sum_{i=1}^{\mu} w_i \, \mathbf{x}_{i:\lambda} = \mathbf{m} + \sigma \mathbf{y}_w$ where $\mathbf{y}_w = \sum_{i=1}^{\mu} w_i \, \mathbf{y}_{i:\lambda}$ update mean
3. update $\mathbf{C}$
4. update step-size σ

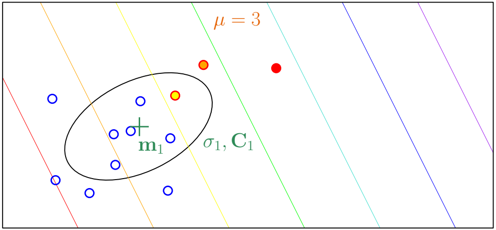


CMA-ES

Input: $\mathbf{m} \in \mathbb{R}^n, \sigma \in \mathbb{R}_+, \lambda \in \mathbb{N}$
Initialize: $\mathbf{C} = \mathbf{I}$ (and several other parameters)
Set the weights $w_1, \ldots w_\lambda$ appropriately

while not terminate

1. $\mathbf{x}_i = \mathbf{m} + \sigma \mathbf{y}_i, \quad \mathbf{y}_i \sim N(\mathbf{0}, \mathbf{C}), \quad$ for $i = 1, \ldots, \lambda$ sampling
2. $\mathbf{m} \leftarrow \sum_{i=1}^{\mu} w_i \, \mathbf{x}_{i:\lambda} = \mathbf{m} + \sigma \mathbf{y}_w$ where $\mathbf{y}_w = \sum_{i=1}^{\mu} w_i \, \mathbf{y}_{i:\lambda}$ update mean
3. update $\mathbf{C}$
4. update step-size σ

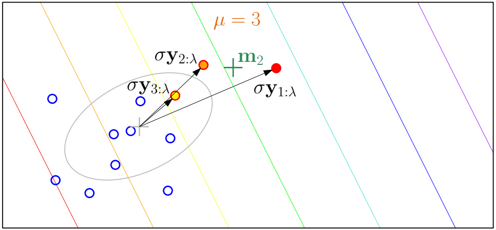


CMA-ES

Input: $\mathbf{m} \in \mathbb{R}^n, \sigma \in \mathbb{R}_+, \lambda \in \mathbb{N}$
Initialize: $\mathbf{C} = \mathbf{I}$ (and several other parameters)
Set the weights $w_1, \ldots w_\lambda$ appropriately

while not terminate

1. $\mathbf{x}_i = \mathbf{m} + \sigma \mathbf{y}_i, \quad \mathbf{y}_i \sim N(\mathbf{0}, \mathbf{C}), \quad$ for $i = 1, \ldots, \lambda$ sampling
2. $\mathbf{m} \leftarrow \sum_{i=1}^{\mu} w_i \mathbf{x}_{i:\lambda} = \mathbf{m} + \sigma \mathbf{y}_w$ where $\mathbf{y}_w = \sum_{i=1}^{\mu} w_i \mathbf{y}_{i:\lambda}$ update mean
3. update $\mathbf{C}$
4. update step-size σ

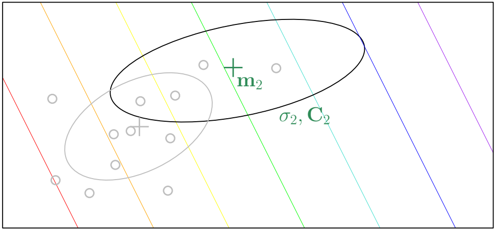


CMA-ES

Input: $\mathbf{m} \in \mathbb{R}^n, \sigma \in \mathbb{R}_+, \lambda \in \mathbb{N}$
Initialize: $\mathbf{C} = \mathbf{I}$ (and several other parameters)
Set the weights $w_1, \ldots w_\lambda$ appropriately

while not terminate

1. $\mathbf{x}_i = \mathbf{m} + \sigma \mathbf{y}_i, \quad \mathbf{y}_i \sim N(\mathbf{0}, \mathbf{C}), \quad$ for $i = 1, \ldots, \lambda$ sampling
2. $\mathbf{m} \leftarrow \sum_{i=1}^{\mu} w_i \, \mathbf{x}_{i:\lambda} = \mathbf{m} + \sigma \mathbf{y}_w \quad$ where $\mathbf{y}_w = \sum_{i=1}^{\mu} w_i \, \mathbf{y}_{i:\lambda}$ update mean
3. update $\mathbf{C}$
4. update step-size σ

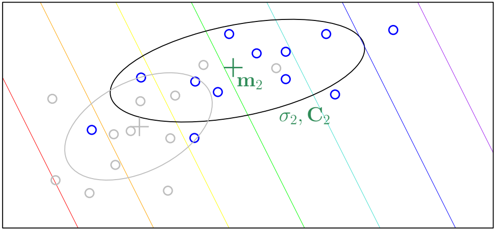


CMA-ES

Input: $\mathbf{m} \in \mathbb{R}^n, \sigma \in \mathbb{R}_+, \lambda \in \mathbb{N}$
Initialize: $\mathbf{C} = \mathbf{I}$ (and several other parameters)
Set the weights $w_1, \ldots w_\lambda$ appropriately

while not terminate

1. $\mathbf{x}_i = \mathbf{m} + \sigma \mathbf{y}_i, \quad \mathbf{y}_i \sim N(\mathbf{0}, \mathbf{C}), \quad$ for $i = 1, \ldots, \lambda$ sampling
2. $\mathbf{m} \leftarrow \sum_{i=1}^{\mu} w_i \, \mathbf{x}_{i:\lambda} = \mathbf{m} + \sigma \mathbf{y}_w$ where $\mathbf{y}_w = \sum_{i=1}^{\mu} w_i \, \mathbf{y}_{i:\lambda}$ update mean
3. update $\mathbf{C}$
4. update step-size σ

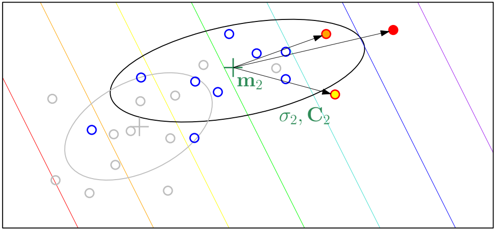


CMA-ES

Input: $\mathbf{m} \in \mathbb{R}^n, \sigma \in \mathbb{R}_+, \lambda \in \mathbb{N}$
Initialize: $\mathbf{C} = \mathbf{I}$ (and several other parameters)
Set the weights $w_1, \ldots w_\lambda$ appropriately

while not terminate

1. $\mathbf{x}_i = \mathbf{m} + \sigma \mathbf{y}_i, \quad \mathbf{y}_i \sim N(\mathbf{0}, \mathbf{C}), \quad$ for $i = 1, \ldots, \lambda$ — sampling
2. $\mathbf{m} \leftarrow \sum_{i=1}^{\mu} w_i \, \mathbf{x}_{i:\lambda} = \mathbf{m} + \sigma \mathbf{y}_w \quad$ where $\mathbf{y}_w = \sum_{i=1}^{\mu} w_i \, \mathbf{y}_{i:\lambda}$ — update mean
3. update $\mathbf{C}$
4. update step-size σ

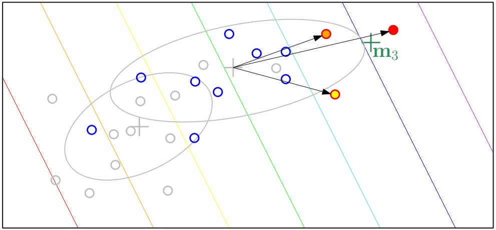


CMA-ES

Input: $\mathbf{m} \in \mathbb{R}^n, \sigma \in \mathbb{R}_+, \lambda \in \mathbb{N}$
Initialize: $\mathbf{C} = \mathbf{I}$ (and several other parameters)
Set the weights $w_1, \ldots w_\lambda$ appropriately

while not terminate

1. $\mathbf{x}_i = \mathbf{m} + \sigma \mathbf{y}_i, \quad \mathbf{y}_i \sim N(\mathbf{0}, \mathbf{C}), \quad$ for $i = 1, \ldots, \lambda$ sampling
2. $\mathbf{m} \leftarrow \sum_{i=1}^{\mu} w_i \, \mathbf{x}_{i:\lambda} = \mathbf{m} + \sigma \mathbf{y}_w \quad$ where $\mathbf{y}_w = \sum_{i=1}^{\mu} w_i \, \mathbf{y}_{i:\lambda}$ update mean
3. update $\mathbf{C}$
4. update step-size σ

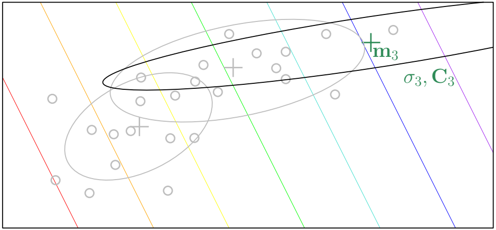


CMA-ES

Input: $\mathbf{m} \in \mathbb{R}^n, \sigma \in \mathbb{R}_+, \lambda \in \mathbb{N}$
Initialize: $\mathbf{C} = \mathbf{I}$ (and several other parameters)
Set the weights $w_1, \ldots w_\lambda$ appropriately

while not terminate

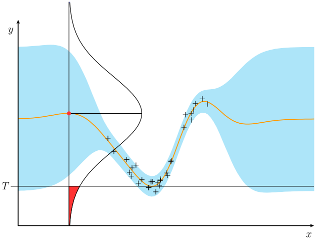
1. $\mathbf{x}_i = \mathbf{m} + \sigma \mathbf{y}_i, \quad \mathbf{y}_i \sim N(\mathbf{0}, \mathbf{C}), \quad \text{for } i = 1, \ldots, \lambda$ sampling
2. $\mathbf{m} \leftarrow \sum_{i=1}^{\mu} w_i \, \mathbf{x}_{i:\lambda} = \mathbf{m} + \sigma \mathbf{y}_w \quad \text{where } \mathbf{y}_w = \sum_{i=1}^{\mu} w_i \, \mathbf{y}_{i:\lambda}$ update mean
3. update $\mathbf{C}$
4. update step-size σ



GAUSSIAN PROCESSES

A collection of random variables, any finite subset of which have a joint Gaussian distribution.

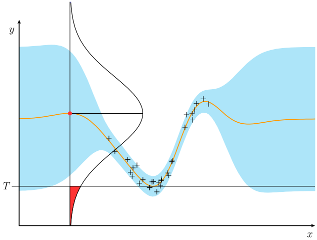
- specified by a **mean function** and a **covariance function**
- prediction in a point given as a **univariate** Gaussian



Gaussian Processes

A collection of random variables, any finite subset of which have a joint Gaussian distribution.

- specified by a **mean function** and a **covariance function**
- prediction in a point given as a **univariate** Gaussian

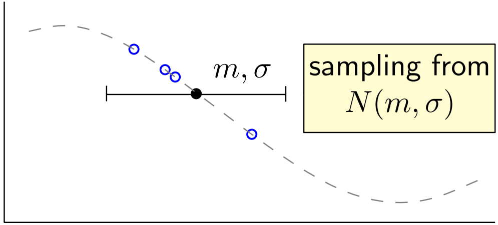


CMA-ES

Input: $\mathbf{m} \in \mathbb{R}^n, \sigma \in \mathbb{R}_+, \lambda \in \mathbb{N}$
Initialize: $\mathbf{C} = \mathbf{I}$ (and several other parameters)
Set the weights $w_1, \ldots w_\lambda$ appropriately

while not terminate

1. $\mathbf{x}_i = \mathbf{m} + \sigma \mathbf{y}_i, \quad \mathbf{y}_i \sim N(\mathbf{0}, \mathbf{C}), \quad \text{for } i = 1, \ldots, \lambda$ {sampling}
2. evaluate $\mathbf{x}_i$ with the original fitness
3. $\mathbf{m} \leftarrow \sum_{i=1}^{\mu} w_i \, \mathbf{x}_{i:\lambda} = \mathbf{m} + \sigma \mathbf{y}_w, \quad \mathbf{y}_w = \sum_{i=1}^{\mu} w_i \, \mathbf{y}_{i:\lambda}$ {update mean}
4. update step-size σ
5. update $\mathbf{C}$

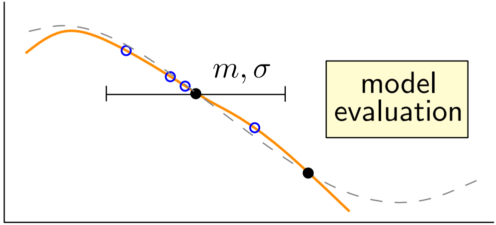


Evolution Control in the CMA-ES

Input: $\mathbf{m} \in \mathbb{R}^n, \sigma \in \mathbb{R}_+, \lambda \in \mathbb{N}$
Initialize: $\mathbf{C} = \mathbf{I}$ (and several other parameters)
Set the weights $w_1, \ldots w_\lambda$ appropriately

while not terminate

1. $\mathbf{x}_i = \mathbf{m} + \sigma \mathbf{y}_i, \quad \mathbf{y}_i \sim N(\mathbf{0}, \mathbf{C}), \quad$ for $i = 1, \ldots, \lambda$ {sampling}
2. $\mathbf{y}_i \leftarrow \text{evolutionControl}(\mathbf{x}_i)$
3. $\mathbf{m} \leftarrow \sum_{i=1}^{\mu} w_i \, \mathbf{x}_{i:\lambda} = \mathbf{m} + \sigma \mathbf{y}_w, \quad \mathbf{y}_w = \sum_{i=1}^{\mu} w_i \, \mathbf{y}_{i:\lambda}$ {update mean}
4. update step-size σ
5. update $\mathbf{C}$

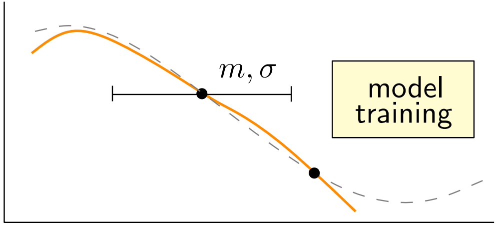


Surrogate CMA-ES

Input: $\mathbf{m} \in \mathbb{R}^n, \sigma \in \mathbb{R}_+, \lambda \in \mathbb{N}$
Initialize: $\mathbf{C} = \mathbf{I}$ (and several other parameters)
Set the weights $w_1, \dots w_\lambda$ appropriately

while not terminate

1. $\mathbf{x}_i = \mathbf{m} + \sigma \mathbf{y}_i, \quad \mathbf{y}_i \sim N(\mathbf{0}, \mathbf{C}), \quad \text{for } i = 1, \dots, \lambda$ {sampling}
2. evaluate $\mathbf{x}_i$ with the original fitness f & build a model $f_{\mathcal{M}}$
3. $\mathbf{m} \leftarrow \sum_{i=1}^{\mu} w_i \, \mathbf{x}_{i:\lambda} = \mathbf{m} + \sigma \mathbf{y}_w, \quad \mathbf{y}_w = \sum_{i=1}^{\mu} w_i \, \mathbf{y}_{i:\lambda}$ {update mean}
4. update step-size σ
5. update $\mathbf{C}$

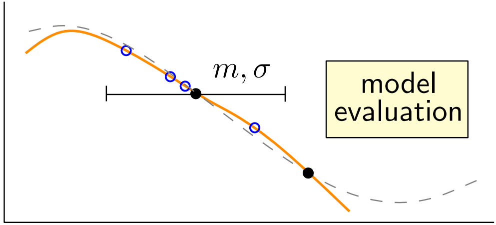


Surrogate CMA-ES

Input: $\mathbf{m} \in \mathbb{R}^n, \sigma \in \mathbb{R}_+, \lambda \in \mathbb{N}$
Initialize: $\mathbf{C} = \mathbf{I}$ (and several other parameters)
Set the weights $w_1, \ldots w_\lambda$ appropriately

while not terminate

1. $\mathbf{x}_i = \mathbf{m} + \sigma \mathbf{y}_i, \quad \mathbf{y}_i \sim N(\mathbf{0}, \mathbf{C}), \quad \text{for } i = 1, \ldots, \lambda$ {sampling}
2. evaluate $\mathbf{x}_i$ with the model $f_{\mathcal{M}}$
3. $\mathbf{m} \leftarrow \sum_{i=1}^{\mu} w_i \, \mathbf{x}_{i:\lambda} = \mathbf{m} + \sigma \mathbf{y}_w, \quad \mathbf{y}_w = \sum_{i=1}^{\mu} w_i \, \mathbf{y}_{i:\lambda}$ {update mean}
4. update step-size σ
5. update $\mathbf{C}$

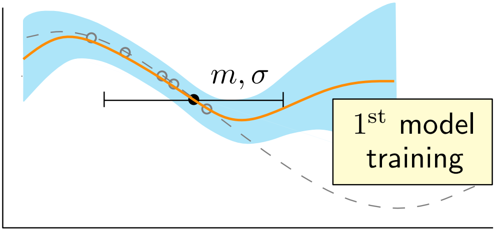


Doubly Trained Surrogate CMA-ES

Input: $\mathbf{m} \in \mathbb{R}^n, \sigma \in \mathbb{R}_+, \lambda \in \mathbb{N}$
Initialize: $\mathbf{C} = \mathbf{I}$ (and several other parameters)
Set the weights $w_1, \ldots w_\lambda$ appropriately

while not terminate

1. $\mathbf{x}_i = \mathbf{m} + \sigma \mathbf{y}_i, \quad \mathbf{y}_i \sim N(\mathbf{0}, \mathbf{C}), \quad$ for $i = 1, \ldots, \lambda$ {sampling}
2. build a model $f_{\mathcal{M}}$
3. $\mathbf{m} \leftarrow \sum_{i=1}^{\mu} w_i \, \mathbf{x}_{i:\lambda} = \mathbf{m} + \sigma \mathbf{y}_w, \quad \mathbf{y}_w = \sum_{i=1}^{\mu} w_i \, \mathbf{y}_{i:\lambda}$ {update mean}
4. update step-size σ
5. update $\mathbf{C}$

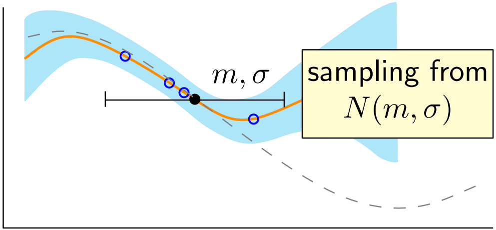


Doubly Trained Surrogate CMA-ES

Input: $\mathbf{m} \in \mathbb{R}^n, \sigma \in \mathbb{R}_+, \lambda \in \mathbb{N}$
Initialize: $\mathbf{C} = \mathbf{I}$ (and several other parameters)
Set the weights $w_1, \ldots w_\lambda$ appropriately

while not terminate

1. $\mathbf{x}_i = \mathbf{m} + \sigma \mathbf{y}_i, \quad \mathbf{y}_i \sim N(\mathbf{0}, \mathbf{C}), \quad$ for $i = 1, \ldots, \lambda$ {sampling}
2. build a model $f_{\mathcal{M}}$
3. $\mathbf{m} \leftarrow \sum_{i=1}^{\mu} w_i \, \mathbf{x}_{i:\lambda} = \mathbf{m} + \sigma \mathbf{y}_w, \quad \mathbf{y}_w = \sum_{i=1}^{\mu} w_i \, \mathbf{y}_{i:\lambda}$ {update mean}
4. update step-size σ
5. update $\mathbf{C}$

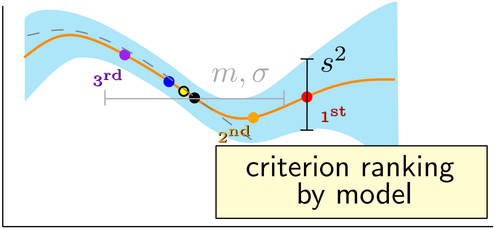


Doubly Trained Surrogate CMA-ES

Input: $\mathbf{m} \in \mathbb{R}^n, \sigma \in \mathbb{R}_+, \lambda \in \mathbb{N}$
Initialize: $\mathbf{C} = \mathbf{I}$ (and several other parameters)
Set the weights $w_1, \dots w_\lambda$ appropriately

while not terminate

1. $\mathbf{x}_i = \mathbf{m} + \sigma \mathbf{y}_i, \quad \mathbf{y}_i \sim N(\mathbf{0}, \mathbf{C}), \quad$ for $i = 1, \dots, \lambda$ {sampling}
2. evaluate $\mathbf{x}_i$ with the model $f_{\mathcal{M}}$
3. $\mathbf{m} \leftarrow \sum_{i=1}^{\mu} w_i \, \mathbf{x}_{i:\lambda} = \mathbf{m} + \sigma \mathbf{y}_w, \quad \mathbf{y}_w = \sum_{i=1}^{\mu} w_i \, \mathbf{y}_{i:\lambda}$ {update mean}
4. update step-size σ
5. update $\mathbf{C}$

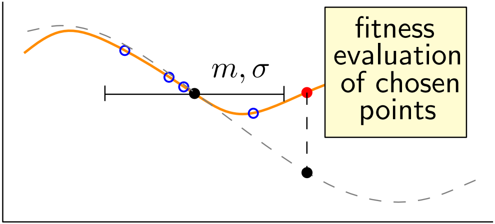


Doubly Trained Surrogate CMA-ES

Input: $\mathbf{m} \in \mathbb{R}^n, \sigma \in \mathbb{R}_+, \lambda \in \mathbb{N}$
Initialize: $\mathbf{C} = \mathbf{I}$ (and several other parameters)
Set the weights $w_1, \ldots w_\lambda$ appropriately

while not terminate

1. $\mathbf{x}_i = \mathbf{m} + \sigma \mathbf{y}_i, \quad \mathbf{y}_i \sim N(\mathbf{0}, \mathbf{C}), \quad \text{for } i = 1, \ldots, \lambda$ {sampling}
2. evaluate $\mathbf{x}_i$ with the original fitness f & build a model $f_{\mathcal{M}}$
3. $\mathbf{m} \leftarrow \sum_{i=1}^{\mu} w_i \, \mathbf{x}_{i:\lambda} = \mathbf{m} + \sigma \mathbf{y}_w, \quad \mathbf{y}_w = \sum_{i=1}^{\mu} w_i \, \mathbf{y}_{i:\lambda}$ {update mean}
4. update step-size σ
5. update $\mathbf{C}$

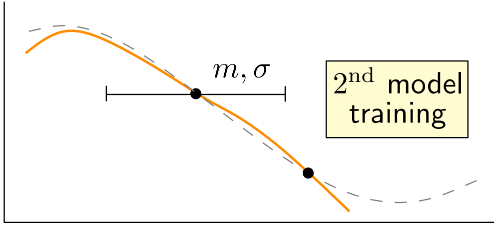


Doubly Trained Surrogate CMA-ES

Input: $\mathbf{m} \in \mathbb{R}^n, \sigma \in \mathbb{R}_+, \lambda \in \mathbb{N}$
Initialize: $\mathbf{C} = \mathbf{I}$ (and several other parameters)
Set the weights $w_1, \ldots w_\lambda$ appropriately

while not terminate

1. $\mathbf{x}_i = \mathbf{m} + \sigma \mathbf{y}_i, \quad \mathbf{y}_i \sim N(\mathbf{0}, \mathbf{C}), \quad$ for $i = 1, \ldots, \lambda$ {sampling}
2. evaluate $\mathbf{x}_i$ with the original fitness f & build a model $f_{\mathcal{M}}$
3. $\mathbf{m} \leftarrow \sum_{i=1}^{\mu} w_i \, \mathbf{x}_{i:\lambda} = \mathbf{m} + \sigma \mathbf{y}_w, \quad \mathbf{y}_w = \sum_{i=1}^{\mu} w_i \, \mathbf{y}_{i:\lambda}$ {update mean}
4. update step-size σ
5. update $\mathbf{C}$

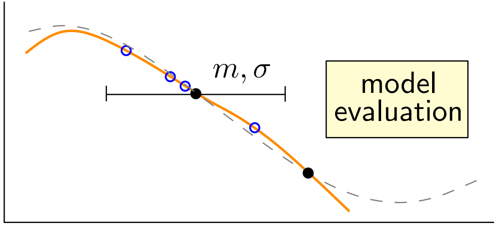


Doubly Trained Surrogate CMA-ES

Input: $\mathbf{m} \in \mathbb{R}^n, \sigma \in \mathbb{R}_+, \lambda \in \mathbb{N}$
Initialize: $\mathbf{C} = \mathbf{I}$ (and several other parameters)
Set the weights $w_1, \ldots w_\lambda$ appropriately

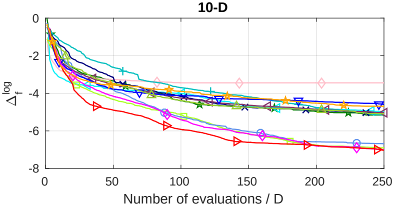
while not terminate

1. $\mathbf{x}_i = \mathbf{m} + \sigma \mathbf{y}_i, \quad \mathbf{y}_i \sim N(\mathbf{0}, \mathbf{C}), \quad$ for $i = 1, \ldots, \lambda$ {sampling}
2. evaluate $\mathbf{x}_i$ with the model $f_{\mathcal{M}}$
3. $\mathbf{m} \leftarrow \sum_{i=1}^{\mu} w_i \, \mathbf{x}_{i:\lambda} = \mathbf{m} + \sigma \mathbf{y}_w, \quad \mathbf{y}_w = \sum_{i=1}^{\mu} w_i \, \mathbf{y}_{i:\lambda}$ {update mean}
4. update step-size σ
5. update $\mathbf{C}$



DTS-CMA-ES Experimental Validation

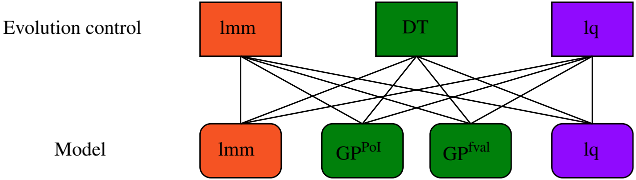
- **COCO testbed**
  - 24 noiseless benchmarks
  - 2, 3, 5, 10, and 20D
  - 15 benchmark transformations (instances)
- 12 surrogate-assisted CMA-ES variants
  - ★ S-CMA-ES (5 gen)
  - ▷ DTS-CMA-ES (0.05/2pop)
  - ◇ adaptive DTS-CMA-ES
- 250 FE/D or 10^{-8} target value



DTS-CMA-ES Variants

- Ordinal GP
  - Lower performance except Attractive sector
- Random forest
  - Overall lower perfomance
  - Improves on multimodal functions with global structure
- Infomation criterion selection (early stage)
  - Lower performance except two multimodal functions
- GP + ANN (early stage)
  - Only linear covariance improvement

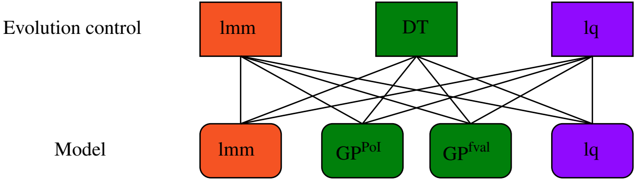
Model vs. Evolution Control



- **Algorithms**
  - lmm-CMA-ES
  - DTS-CMA-ES
  - lq-CMA-ES
- 250 FE/D

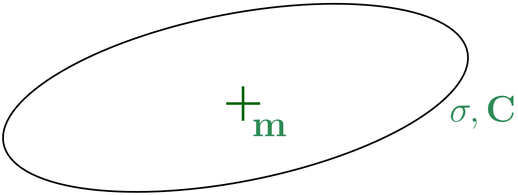
- **Benchmarking**
  - 24 noiseless and 30 noisy benchmarks
  - 5 dimensions and 15 instances
  - Energy wave landscape simulation benchmark (6 dims, 24 settings)

Model vs. Evolution Control



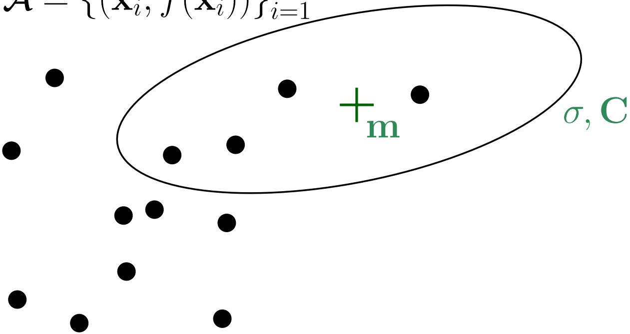
- **Algorithms**
  - lmm-CMA-ES
  - DTS-CMA-ES
  - lq-CMA-ES
- 250 FE/D
- **Results**
  - EC and model significant influence on convergence
  - lq-CMA-ES EC and GP models very successful
- **Benchmarking**
  - 24 noiseless and 30 noisy benchmarks
  - 5 dimensions and 15 instances
  - Energy wave landscape simulation benchmark (6 dims, 24 settings)

Model Training in the CMA-ES

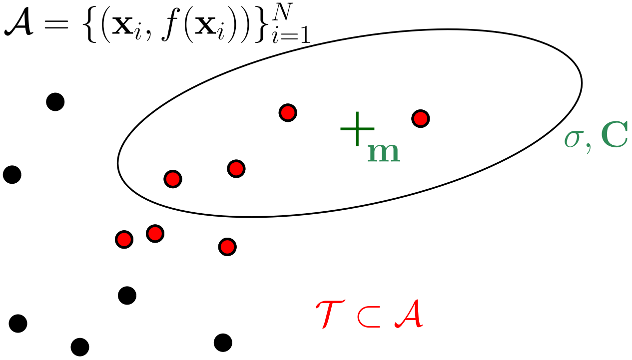


Model Training in the CMA-ES

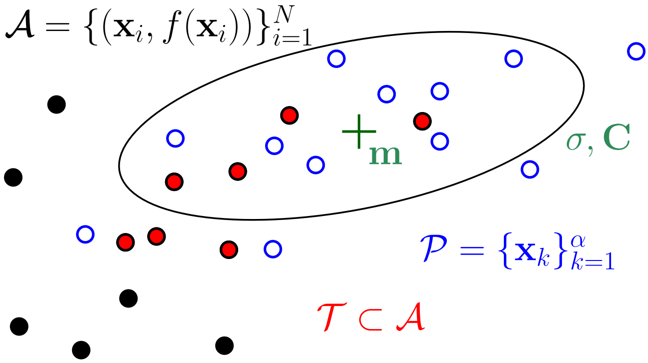
$$\mathcal{A} = \{(\mathbf{x}_i, f(\mathbf{x}_i))\}_{i=1}^{N}$$



Model Training in the CMA-ES



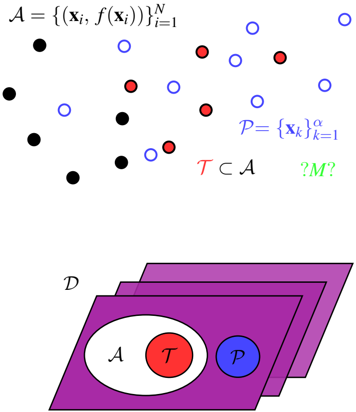
Model Training in the CMA-ES



Experimental settings

Dataset

- Snapshots from 100 artificial runs of the DTS-CMA-ES
  - 24 noiseless benchmark functions
  - 5 dimensions
  - 5 instances
  - 8 covariance functions
  - 100 generations
- 48 mil. data



Experimental settings

Dataset - Sample Sets

$$\mathcal{S} = \left\{ (\mathbf{x}_i, y_i) \in \mathbb{R}^D \times \mathbb{R} \cup \{\circ\} \,\middle|\, i = 1, \ldots, N \right\}$$

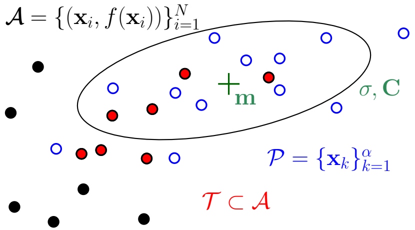
Archive $\mathcal{A}$, $\mathcal{A}^\top$

Training set $\mathcal{T}$, $\mathcal{T}^\top$

Archive + Population set $\mathcal{A}_\mathcal{P}$, $\mathcal{A}_\mathcal{P}{}^\top$

Training + Population set $\mathcal{T}_\mathcal{P}$, $\mathcal{T}_\mathcal{P}{}^\top$

$^\top$ set in CMA-ES basis



EXPERIMENTAL SETTINGS

DATASET - TRAINING SET SELECTION METHODS (TSS)

TSS full

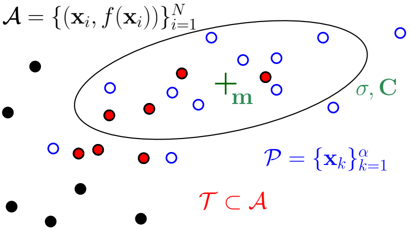
- $\mathcal{A} = \mathcal{T}$

TSS knn

- Unification of k-nn points to $\mathcal{P}$

TSS nearest

- Unification of k-nn points to $\mathcal{P}$
- k is maximal such that $|\mathcal{T}| \leq N_{\max}$
- Mahalanobis distance to $\mathbf{m}$ $\leq r_{\max}$



EXPERIMENTAL SETTINGS

FEATURE SPACE ∼ EXPLORATORY LANDSCAPE FEATURES

$$\varphi : \bigcup_{N \in \mathbb{N}} \mathbb{R}^{N,D} \times (\mathbb{R} \cup \{\circ\})^{N,1} \mapsto \mathbb{R} \cup \{\pm\infty, \bullet\}$$

- Distribution
- Levelset
- Meta-Model
- Nearest better clustering (NBC)
- Dispersion
- Information content
- *Dimension*
- *Number of observations*

New CMA-ES features

Experimental settings

Feature space ∼ Exploratory landscape features

$$\varphi : \bigcup_{N \in \mathbb{N}} \mathbb{R}^{N,D} \times (\mathbb{R} \cup \{\circ\})^{N,1} \mapsto \mathbb{R} \cup \{\pm\infty, \bullet\}$$

- Distribution
- Levelset
- Meta-Model
- Nearest better clustering (NBC)
- Dispersion
- Information content
- *Dimension*
- *Number of observations*

New CMA-ES features

Experimental settings

CMA-ES features

- Generation number $\varphi_g = g$
- Step-size $\varphi_\sigma = \sigma$
- Number of restarts $\varphi_{\text{restart}} = n_{\text{r}}$
- CMA mean distance $\varphi_{d(\mathbf{m})} = \sqrt{(\mathbf{m} - \mu_{\mathbf{X}})^\top \mathbf{C_X}^{-1} (\mathbf{m} - \mu_{\mathbf{X}})}$
- $\mathbf{C}$ evolution path length square $\varphi_{\mathbf{p}_c} = \|\mathbf{p}_c\|^2$
- σ evolution path ratio $\varphi_{\mathbf{p}_\sigma} = \frac{\|\mathbf{p}_\sigma\|}{E\|\mathrm{N}(\mathbf{0},\mathbf{I})\|}$
- CMA similarity likelihood
  $\varphi_{\mathcal{L}} = -\frac{N}{2}\left(D \log 2\pi\sigma^2 + \log\det \mathbf{C}\right) - \frac{1}{2}\sum_{\mathbf{x}\in\mathbf{X}} \left(\frac{\mathbf{x}-\mathbf{m}}{\sigma}\right)^\top \mathbf{C}^{-1} \left(\frac{\mathbf{x}-\mathbf{m}}{\sigma}\right)$

Experimental settings

CMA-ES features

- Generation number $\varphi_g = g$
- Step-size $\varphi_\sigma = \sigma$
- Number of restarts $\varphi_{\text{restart}} = n_{\text{r}}$
- CMA mean distance $\varphi_{d(\mathbf{m})} = \sqrt{(\mathbf{m} - \mu_\mathbf{X})^\top \mathbf{C_X}^{-1} (\mathbf{m} - \mu_\mathbf{X})}$
- $\mathbf{C}$ evolution path length square $\varphi_{\mathbf{p}_c} = \|\mathbf{p}_c\|^2$
- σ evolution path ratio $\varphi_{\mathbf{p}_\sigma} = \frac{\|\mathbf{p}_\sigma\|}{E\|\mathrm{N}(\mathbf{0},\mathbf{I})\|}$
- CMA similarity likelihood
  $\varphi_\mathcal{L} = -\frac{N}{2}\left(D \log 2\pi\sigma^2 + \log\det \mathbf{C}\right) - \frac{1}{2}\sum_{\mathbf{x}\in\mathbf{X}} \left(\frac{\mathbf{x}-\mathbf{m}}{\sigma}\right)^\top \mathbf{C}^{-1} \left(\frac{\mathbf{x}-\mathbf{m}}{\sigma}\right)$

Experimental settings

CMA-ES features

- Generation number $\varphi_g = g$
- Step-size $\varphi_\sigma = \sigma$
- Number of restarts $\varphi_{\text{restart}} = n_{\text{r}}$
- CMA mean distance $\varphi_{d(\mathbf{m})} = \sqrt{(\mathbf{m} - \mu_{\mathbf{X}})^\top \mathbf{C_X}^{-1} (\mathbf{m} - \mu_{\mathbf{X}})}$
- $\mathbf{C}$ evolution path length square $\varphi_{\mathbf{p}_c} = \|\mathbf{p}_c\|^2$
- σ evolution path ratio $\varphi_{\mathbf{p}_\sigma} = \frac{\|\mathbf{p}_\sigma\|}{E\|\mathrm{N}(\mathbf{0},\mathbf{I})\|}$
- CMA similarity likelihood
  $\varphi_{\mathcal{L}} = -\frac{N}{2}\left(D \log 2\pi\sigma^2 + \log\det \mathbf{C}\right) - \frac{1}{2}\sum_{\mathbf{x}\in\mathbf{X}} \left(\frac{\mathbf{x}-\mathbf{m}}{\sigma}\right)^\top \mathbf{C}^{-1} \left(\frac{\mathbf{x}-\mathbf{m}}{\sigma}\right)$

EXPERIMENTAL SETTINGS

CMA-ES FEATURES

- Generation number $\varphi_g = g$
- Step-size $\varphi_\sigma = \sigma$
- Number of restarts $\varphi_{\text{restart}} = n_{\text{r}}$
- CMA mean distance $\varphi_{d(\mathbf{m})} = \sqrt{(\mathbf{m} - \mu_{\mathbf{X}})^\top \mathbf{C_X}^{-1} (\mathbf{m} - \mu_{\mathbf{X}})}$
- $\mathbf{C}$ evolution path length square $\varphi_{\mathbf{p}_c} = \|\mathbf{p}_c\|^2$
- σ evolution path ratio $\varphi_{\mathbf{p}_\sigma} = \frac{\|\mathbf{p}_\sigma\|}{E\|\mathrm{N}(\mathbf{0},\mathbf{I})\|}$
- CMA similarity likelihood
  $\varphi_{\mathcal{L}} = -\frac{N}{2}\left(D \log 2\pi\sigma^2 + \log\det \mathbf{C}\right) - \frac{1}{2}\sum_{\mathbf{x}\in\mathbf{X}} \left(\frac{\mathbf{x}-\mathbf{m}}{\sigma}\right)^\top \mathbf{C}^{-1} \left(\frac{\mathbf{x}-\mathbf{m}}{\sigma}\right)$

EXPERIMENTAL SETTINGS

CMA-ES FEATURES

- Generation number $\varphi_g = g$
- Step-size $\varphi_\sigma = \sigma$
- Number of restarts $\varphi_{\text{restart}} = n_{\text{r}}$
- CMA mean distance $\varphi_{d(\mathbf{m})} = \sqrt{(\mathbf{m} - \mu_{\mathbf{X}})^\top \mathbf{C}_{\mathbf{X}}{}^{-1}(\mathbf{m} - \mu_{\mathbf{X}})}$
- **C** evolution path length square $\varphi_{\mathbf{p}_c} = \|\mathbf{p}_c\|^2$
- σ evolution path ratio $\varphi_{\mathbf{p}_\sigma} = \frac{\|\mathbf{p}_\sigma\|}{E\|\mathrm{N}(\mathbf{0},\mathbf{I})\|}$
- CMA similarity likelihood
  $\varphi_{\mathcal{L}} = -\frac{N}{2}\left(D \log 2\pi\sigma^2 + \log \det \mathbf{C}\right) - \frac{1}{2}\sum_{\mathbf{x}\in\mathbf{X}} \left(\frac{\mathbf{x}-\mathbf{m}}{\sigma}\right)^\top \mathbf{C}^{-1}\left(\frac{\mathbf{x}-\mathbf{m}}{\sigma}\right)$

Experimental settings

CMA-ES features

- Generation number $\varphi_g = g$
- Step-size $\varphi_\sigma = \sigma$
- Number of restarts $\varphi_{\text{restart}} = n_{\text{r}}$
- CMA mean distance $\varphi_{d(\mathbf{m})} = \sqrt{(\mathbf{m} - \mu_{\mathbf{X}})^\top \mathbf{C_X}^{-1} (\mathbf{m} - \mu_{\mathbf{X}})}$
- **C** evolution path length square $\varphi_{\mathbf{p}_c} = \|\mathbf{p}_c\|^2$
- σ evolution path ratio $\varphi_{\mathbf{p}_\sigma} = \frac{\|\mathbf{p}_\sigma\|}{E\|\mathsf{N}(\mathbf{0},\mathbf{I})\|}$
- CMA similarity likelihood
  $\varphi_{\mathcal{L}} = -\frac{N}{2}\left(D \log 2\pi\sigma^2 + \log\det \mathbf{C}\right) - \frac{1}{2}\sum_{\mathbf{x}\in\mathbf{X}} \left(\frac{\mathbf{x}-\mathbf{m}}{\sigma}\right)^\top \mathbf{C}^{-1} \left(\frac{\mathbf{x}-\mathbf{m}}{\sigma}\right)$

Experimental settings

CMA-ES features

- Generation number $\varphi_g = g$
- Step-size $\varphi_\sigma = \sigma$
- Number of restarts $\varphi_{\text{restart}} = n_{\text{r}}$
- CMA mean distance $\varphi_{d(\mathbf{m})} = \sqrt{(\mathbf{m} - \mu_{\mathbf{X}})^\top \mathbf{C}_{\mathbf{X}}{}^{-1} (\mathbf{m} - \mu_{\mathbf{X}})}$
- $\mathbf{C}$ evolution path length square $\varphi_{\mathbf{p}_c} = \|\mathbf{p}_c\|^2$
- σ evolution path ratio $\varphi_{\mathbf{p}_\sigma} = \frac{\|\mathbf{p}_\sigma\|}{E\|\mathsf{N}(\mathbf{0},\mathbf{I})\|}$
- CMA similarity likelihood
  $\varphi_{\mathcal{L}} = -\frac{N}{2}\left(D \log 2\pi\sigma^2 + \log\det \mathbf{C}\right) - \frac{1}{2}\sum_{\mathbf{x}\in\mathbf{X}} \left(\frac{\mathbf{x}-\mathbf{m}}{\sigma}\right)^\top \mathbf{C}^{-1} \left(\frac{\mathbf{x}-\mathbf{m}}{\sigma}\right)$

Analysis of landscape features

$\bullet$ and $\pm\infty$

Impossibility of calculation $\bullet$

- $\geq 25\%$ of values $= \bullet \rightarrow$ exclude feature
- Minimal number of points for feature calculation $N_\bullet$
  - $< 1\%$ of values $= \bullet$
  - $N_\bullet = 6$ without $\mathcal{P}$
  - $N_\bullet = 13$ with $\mathcal{P}$

Analysis of landscape features

$\bullet$ and $\pm\infty$

Impossibility of calculation $\bullet$

- $\geq 25\%$ of values $= \bullet \rightarrow$ exclude feature
- Minimal number of points for feature calculation $N_\bullet$
  - $< 1\%$ of values $= \bullet$
  - $N_\bullet = 6$ without $\mathcal{P}$
  - $N_\bullet = 13$ with $\mathcal{P}$

Normalization

- Sigmoid scaling to $[0, 1]$
- 0.01 and 0.99 quantiles mapped to 0.01 and 0.99
- Dealing with $\pm\infty$

Analysis of landscape features

Robustness

Proportion of cases for which the difference between the 1^{st} and 100^{th} percentile ≤ 0.05.

threshold	TSS full	TSS nearest	TSS knn
0.5	**125**/195	**244**/384 (119/189)	**188**/366 (63/171)
0.6	**82**/195	**158**/384 (76/189)	**131**/366 (49/171)
0.7	**54**/195	**102**/384 (48/189)	**93**/366 (39/171)
0.8	**43**/195	**80**/384 (37/189)	**73**/366 (30/171)
0.9	**33**/195	**60**/384 (27/189)	**59**/366 (26/171)
0.99	**28**/195	**50**/384 (22/189)	**30**/366 (2/171)

Analysis of landscape features

Dimension dependency and Similarity

Dimension dependency

- Friedman rejected feature medians independence on 0.05 level

Analysis of landscape features

Dimension dependency and Similarity

Dimension dependency

- Friedman rejected feature medians independence on 0.05 level

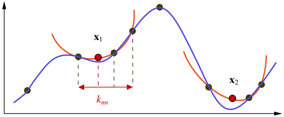
Feature similarity

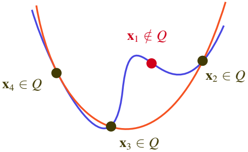
- Agglomerative hierarchical clustering
- Similarity = 1 - Schweizer-Wolf correlation
- Ordering-dependency compensation
  - 5 runs for each TSS method
  - Optimal: 14 clusters
- Feature cluster representatives
  - k-medoids clustering ($k = 14$)
  - Almost identical features for all TSS selected including dimension and number of observations

Experimental settings

Surrogate Models

- GP
  - 8 covariance functions
- RF
  - 5 splitting methods
  - Latin-hypercube design on 100 out of 400 combinations
    - Number of trees $\{2^6, 2^7, 2^8, 2^9, 2^{10}\}$
    - Number of bootstrapped training points $\lceil\{\frac{1}{4}, \frac{1}{2}, \frac{3}{4}, 1\} \cdot N\rceil$
    - Number of subsampled dimensions $\lceil\{\frac{1}{4}, \frac{1}{2}, \frac{3}{4}, 1\} \cdot D\rceil$
  - 2 TSS methods, 2 error measures
- polynomial
  - lmm model
  - lq model

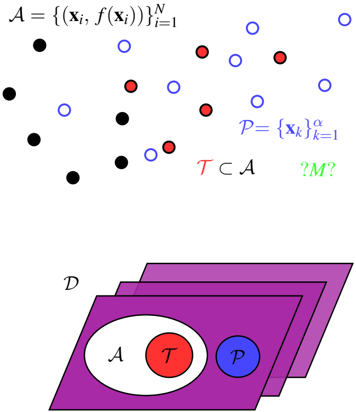




Experimental settings

Dataset

- Snapshots from independent runs of the DTS-CMA-ES
  - 24 noiseless benchmark functions
  - 5 dimensions
  - 5 instances
  - 7 covariance functions
  - 25 generations



Experimental settings

Performance space ∼ MSE & Ranking Difference Error

MSE

- Difference directly from the objective function landscape

Experimental settings

Performance space ∼ MSE & Ranking Difference Error

MSE

- Difference directly from the objective function landscape

RDE

- Difference of ranking of μ best points

$$RDE_{\mu}(\hat{\mathbf{y}}, \mathbf{y}) = \frac{\sum_{i:\rho(i)\leq\mu} |\hat{\rho}(i) - \rho(i)|}{\max_{\pi \in \text{Permutations of } (1,\ldots,\lambda)} \sum_{i:\pi(i)\leq\mu} |\, i - \pi(i)|}$$

λ – population size
$\mathbf{y}, \hat{\mathbf{y}} \in \mathbb{R}^{\lambda}$, $\mu = \lceil \frac{\lambda}{2} \rceil$
$\rho(i)$ – ranks of the i-th element in vector $\mathbf{y}$
$\hat{\rho}(i)$ – ranks of the i-th element in vector $\hat{\mathbf{y}}$

Statistical testing

- MSE and RDE data diversity
  - Friedman test and Tukey's post-hoc test
  - Pairwise — two-sided Wilcoxon signed rank Holm correction
  - Significant differences among wast majority of pairs of 39 model settings
  - GP model settings provided the highest perfomance followed by polynomial models
- Univariate features descriptivity
  - Kolmogorov-Smirnov test
  - Significant differences between features on sample sets with particular best setting and all data
- Multivariate features descriptivity
  - Classification tree per TSS method
  - Equal RDE $\rightarrow$ MSE

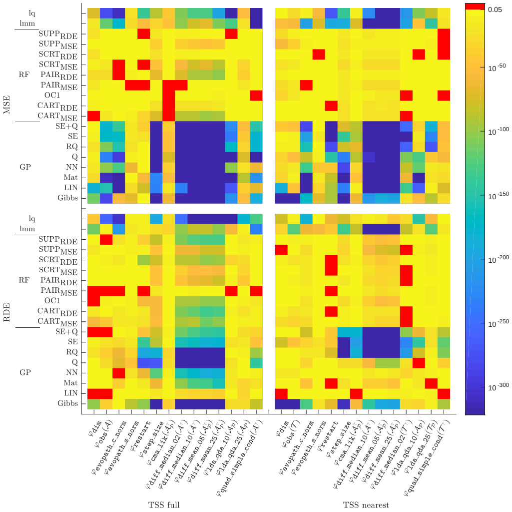
STATISTICAL TESTING

- MSE and RDE data diversity
  - Friedman test and Tukey's post-hoc test
  - Pairwise — two-sided Wilcoxon signed rank Holm correction
  - Significant differences among wast majority of pairs of 39 model settings
  - GP model settings provided the highest perfomance followed by polynomial models
- Univariate features descriptivity
  - Kolmogorov-Smirnov test
  - Significant differences between features on sample sets with particular best setting and all data
- Multivariate features descriptivity
  - Classification tree per TSS method
  - Equal RDE $\rightarrow$ MSE

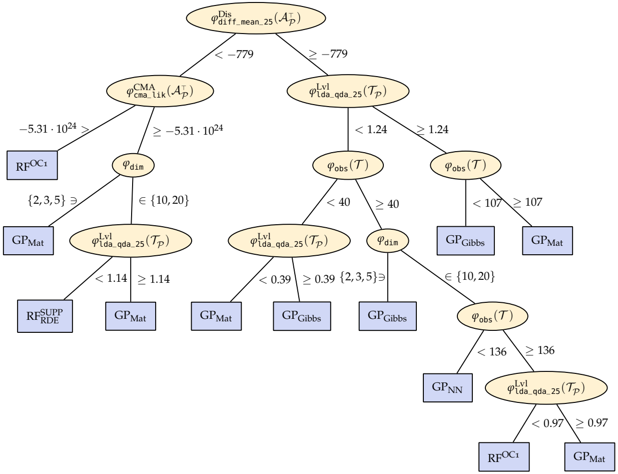
Statistical testing

- MSE and RDE data diversity
  - Friedman test and Tukey's post-hoc test
  - Pairwise — two-sided Wilcoxon signed rank Holm correction
  - Significant differences among wast majority of pairs of 39 model settings
  - GP model settings provided the highest perfomance followed by polynomial models
- Univariate features descriptivity
  - Kolmogorov-Smirnov test
  - Significant differences between features on sample sets with particular best setting and all data
- Multivariate features descriptivity
  - Classification tree per TSS method
  - Equal RDE $\rightarrow$ MSE

Kolmogorov-Smirnov test



Selection ∼ Decision tree



SUMMARY OF RESULTS

- Evolution control
  - Generation EC drops early with GP and RF
  - Doubly trained EC using GP very useful in middle stage
  - EC and SM significantly influence the algorithm's performance
- Landscape analysis
  - Large number of low robust and similar features
  - Significant differences in model settings performance
  - Significant differences in feature distribution
  - CMA-ES based features are useful
- Future research:
  - Surrogate model selection system

Questions?

z.pitra@gmail.com

martin@cs.cas.cz