Preferences, Invariances, Optimization

Ilya Loshchilov, Marc Schoenauer, Michèle Sebag

TAO, CNRS — INRIA — Université Paris-Sud

Dagstuhl, March 2014

Position

One goal of

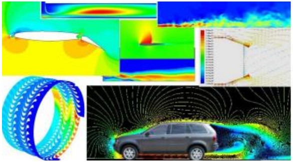
- Machine learning: optimal decision making
- Preference learning: optimization

This talk: black box optimization

When using preference learning ?

- when dealing with the user in the loop Herdy et al., 96
- when dealing with computationally expensive criteria

Herdy et al. 96



Surrogate models

Position

One goal of

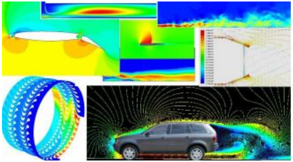
- Machine learning: optimal decision making
- Preference learning: multi-objective optimization

This talk: black box multi-objective optimization

When using preference learning ?

- when dealing with the user in the loop Herdy et al., 96
- when dealing with computationally expensive criteria

Herdy et al. 96



Surrogate models

Optimizing coffee taste

Features

- Search space $X \subset {\mathbb{R}^+}^d$
  (recipe x: 33% arabica, 25% robusta, etc)
- A non-computable objective
- Expert can (by tasting) emit preferences $x \prec x'$.

Interactive optimization see also Viappiani et al. 11

1. Alg. generates two or more candidates $x, x', x``, ..$
2. Expert emits preferences
3. goto 1.

Issues

- Asking as few questions as possible $\neq$ active ranking
- Modelling the expert's taste surrogate model
- Enforce the exploration vs exploitation trade-off

Expensive black-box optimization

Notations

- Search space: $X \subset \mathbb{R}^d$
- Computable objective $\mathcal{F}$: $X \mapsto \mathbb{R}$
- Not well behaved (non convex, non differentiable, etc).

Evolutionary optimization

1. Alg. generates candidate solutions (population) $x_1, \ldots x_\lambda$
2. Compute $\mathcal{F}(x_i)$ and rank x_i accordingly
3. goto 1.

Issues

- Computational cost number of $\mathcal{F}$ computations
- Learn $\widehat{\mathcal{F}}$ surrogate model
- When to use $\mathcal{F}$ and when $\widehat{\mathcal{F}}$? when to refresh $\widehat{\mathcal{F}}$?

Overview

Motivations

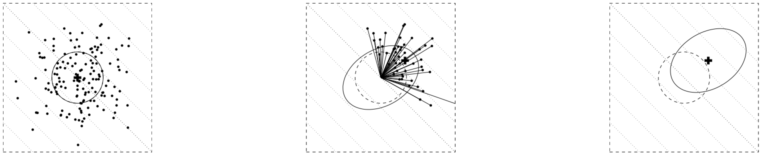
Black-box optimization...

... with surrogate models

Multi-objective optimization

Covariance-Matrix Adaptation (CMA-ES)
Rank-μ Update

$$\boldsymbol{x}_i = \boldsymbol{m} + \sigma\, \boldsymbol{y}_i, \qquad \boldsymbol{y}_i \sim \mathcal{N}_i(\mathbf{0}, \mathbf{C}),$$
$$\boldsymbol{m} \leftarrow \boldsymbol{m} + \sigma \boldsymbol{y}_w \qquad \boldsymbol{y}_w = \sum_{i=1}^{\mu} w_i\, \boldsymbol{y}_{i:\lambda}$$



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

$\mathbf{C}_\mu = \frac{1}{\mu} \sum \boldsymbol{y}_{i:\lambda} \boldsymbol{y}_{i:\lambda}^{\mathrm{T}}$
$\mathbf{C} \leftarrow (1-1) \times \mathbf{C} + 1 \times \mathbf{C}_\mu$

$\boldsymbol{m}_{\text{new}} \leftarrow \boldsymbol{m} + \frac{1}{\mu} \sum \boldsymbol{y}_{i:\lambda}$

sampling of $\lambda = 150$ solutions where $\mathbf{C} = \mathbf{I}$ and $\sigma = 1$

calculating $\mathbf{C}$ from $\mu = 50$ points, $w_1 = \cdots = w_\mu = \frac{1}{\mu}$

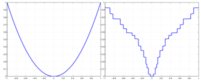
new distribution

Remark: the old (sample) distribution shape has a great influence on the new distribution $\longrightarrow$ iterations needed

- Source codes available:
  `https://www.lri.fr/~hansen/cmaes_inmatlab.html`

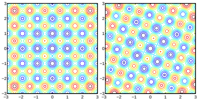
Invariance: Guarantees for Generalization

Invariance properties of CMA-ES



- Invariance to order preserving transformations in function space

  like all comparison-based algorithms



- Translation and rotation invariance

  to *affine transformations* of the search space

CMA-ES is almost parameterless

- Tuning on a small set of functions — Hansen & Ostermeier 2001
- Except: population size for multi-modal functions

  More: IPOP-CMA-ES Auger & Hansen, 05
  and BIPOP-CMA-ES Hansen, 09

Information-Geometric Optimization

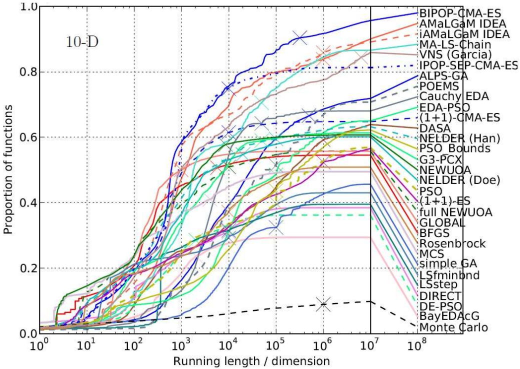
Yann Ollivier et al. 2012

BBOB – Black-Box Optimization Benchmarking

- ACM-GECCO workshops: 2009, 2010, 2012
- Set of 25 benchmark functions, dimensions 2 to 40
- With known difficulties (non-separability, #local optima, condition number...)
- Noisy and non-noisy versions

Competitors include

- BFGS (Matlab version),
- Fletcher-Powell,
- DFO (Derivative-Free Optimization, Powell 04)
- Differential Evolution
- Particle Swarm Optimization
- and others.



Fraction of runs reaching specified accuracy vs number of $\mathcal{F}$ computation.

Overview

Motivations

Black-box optimization...

... with surrogate models

Multi-objective optimization

Surrogate Models for CMA-ES

Exploiting first evaluated solutions as training set

$$\mathcal{E} = \{(x_i, \mathcal{F}(x_i))\}$$

Using Ranking-SVM

- Builds $\widehat{\mathcal{F}}$ using Ranking-SVM

$$\mathbf{x_i} \succ \mathbf{x_j} \text{ iff } \mathcal{F}(\mathbf{x_i}) < \mathcal{F}(\mathbf{x_j})$$

- Kernel and parameters problem-dependent

T. Runarsson (2006). "Ordinal Regression in Evolutionary Computation"

- ACM: Use C from CMA-ES as Gaussian kernel

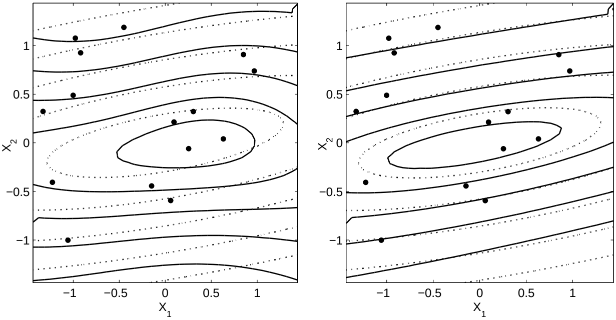
I. Loschilov et al. (2010). "Comparison-based optimizers need comparison-based surrogates”

I. Loschilov et al. (2012). "Self-Adaptive Surrogate-Assisted CMA-ES”

About Model Learning

Non-separable Ellipsoid problem

$$K(x_i, x_j) = e^{-\frac{(x_i - x_j)^t (x_i - x_j)}{2\sigma^2}}; \quad K_C(x_i, x_j) = e^{-\frac{(x_i - x_j)^t C_\mu^{-1} (x_i - x_j)}{2\sigma^2}}$$



Invariance to affine transformations of the search space.

The devil is in the hyper-parameters

SVM Learning

- Number of training points: $N_{training} = 30\sqrt{d}$ for all problems, except Rosenbrock and Rastrigin, where $N_{training} = 70\sqrt{d}$
- Number of iterations: $N_{iter} = 50000\sqrt{d}$
- Kernel function: RBF function with σ equal to the average distance of the training points
- The cost of constraint violation: $C_i = 10^6 (N_{training} - i)^{2.0}$

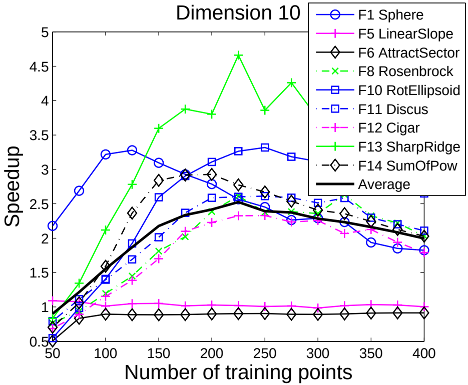
Offspring Selection

- Number of test points: $N_{test} = 500$
- Number of evaluated offsprings: $\lambda' = \frac{\lambda}{3}$
- Offspring selection pressure parameters: $\sigma^2_{sel0} = 2\sigma^2_{sel1} = 0.8$

Sensitivity analysis

The speed-up of ACM-ES is very sensitive

- w.r.t. number of training points.



- w.r.t. lifelength of the surrogate model

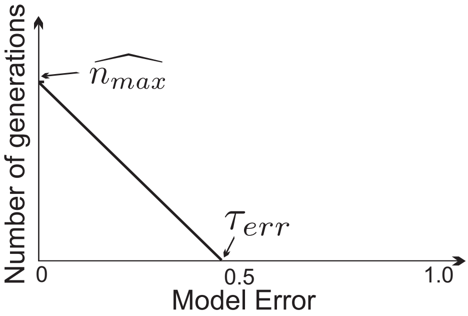
Self-adaptation of $\widehat{\mathcal{F}}$ lifelength

Principle: iterated preference learning

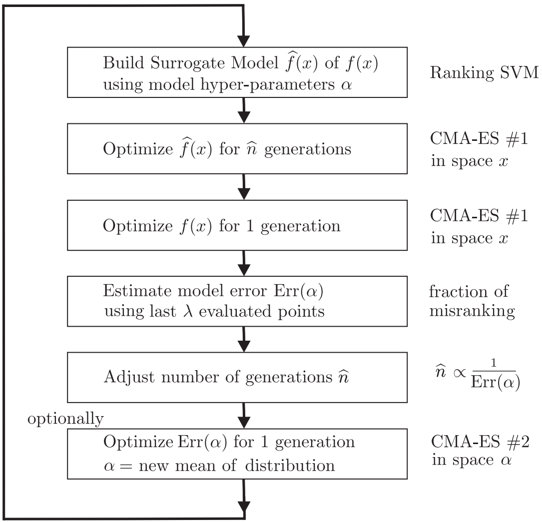
- After n generations, gather new examples $\{x_i, \mathcal{F}(x_i)\}$
- Evaluate rank loss of old $\widehat{\mathcal{F}}$
- Low error: $\widehat{\mathcal{F}}$ could have been used for more generations
- High error: $\widehat{\mathcal{F}}$ should have been relearned earlier.

Self-adaptation

$$n = g(\text{ rank loss}(\widehat{\mathcal{F}}))$$

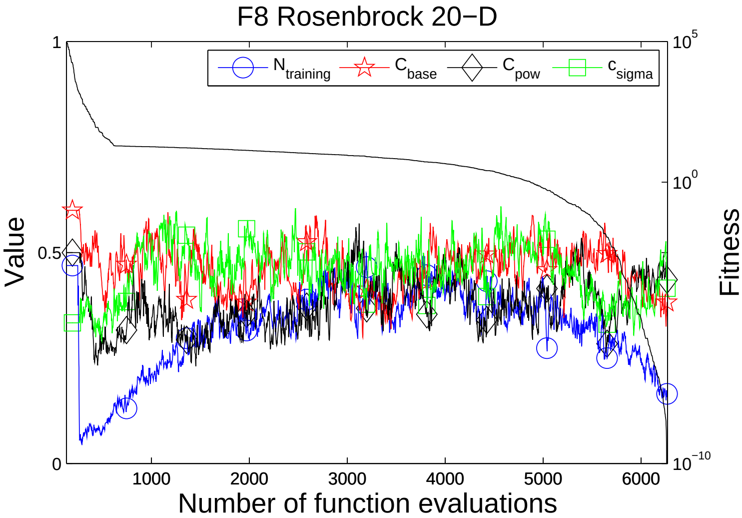


ACM-ES algorithm



Surrogate-assisted CMA-ES with online adaptation of model hyper-parametets.

Online adaptation of model hyper-parameters

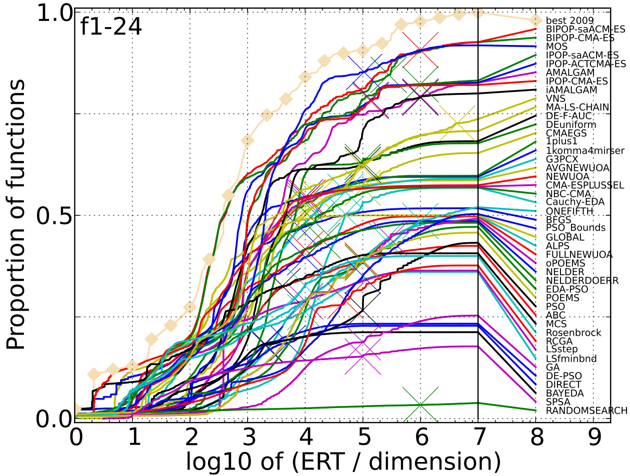


Online-adaptation of hyper-parameters:

improves on optimally tuned hyper-parameters

Results on black-box optimization competition (BBOB)

BIPOP-^{s*}aACM and IPOP-^{s*}aACM (with restarts) on 24 noiseless 20 dimensional functions



ACM-XX significantly improves on XX (BIPOP-CMA, IPOP-CMA)

progress on the top of advanced CMA-ES variants.

Overview

Motivations

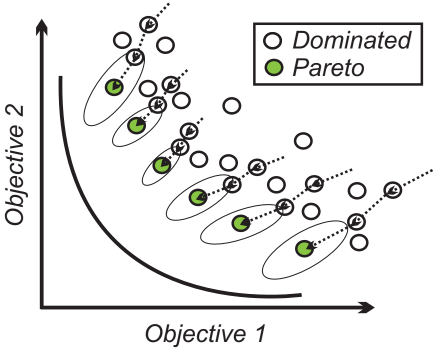
Black-box optimization...

... with surrogate models

Multi-objective optimization

Multi-objective CMA-ES (MO-CMA-ES)

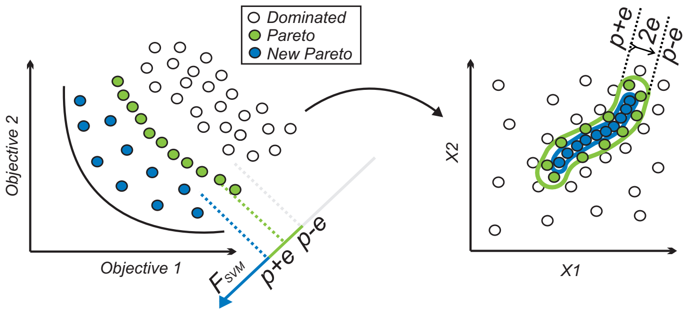
- MO-CMA-ES = μ_{mo} independent (1+1)-CMA-ES.
- Each (1+1)-CMA samples new offspring. The size of the temporary population is $2\mu_{mo}$.
- Only μ_{mo} best solutions should be chosen for new population after the hypervolume-based non-dominated sorting.
- Update of CMA individuals takes place.



A Multi-Objective Surrogate Model

Rationale

- Rationale: find a unique function $F(x)$ that defines the aggregated quality of the solution x in multi-objective case.
- Idea originally proposed using a mixture of One-Class SVM and regression-SVM[1]

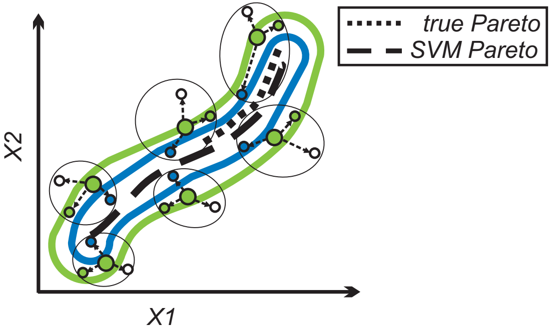


[1] I. Loshchilov, M. Schoenauer, M. Sebag (GECCO 2010). "A Mono Surrogate for Multiobjective Optimization"

Unsing the Surrogate Model

Filtering

- Generate N_{inform} pre-children
- For each pre-children A and the nearest parent B calculate $Gain(A, B) = F_{svm}(A) - F_{svm}(B)$
- New children is the point with the maximum value of $Gain$

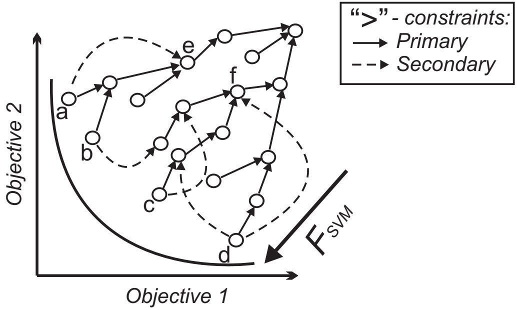


Dominance-Based Surrogate

Using Rank-SVM

Which ordered pairs?

- Considering **all** possible $\succ$ relations may be too expensive.
- Primary constraints: x and its nearest dominated point
- Secondary constraints: any 2 points not belonging to the same front (according to non-dominated sorting)



All primary constraints, and a limited number of secondary constraints

Dominance-Based Surrogate (2)

Construction of the surrogate model

- Initialize archive Ω_{active} as the set of **Primary constraints**, and $\Omega_{passive}$ as the set of **Secondary constraints**.
- Learn the model for $1000\,|\Omega_{active}|$ iterations.
- Add the most violated passive contraint from $\Omega_{passive}$ to Ω_{active} and optimize the model for $10\,|\Omega_{active}|$ iterations.
- Repeat the last step 0.1$|\Omega_{active}|$ times.

Experimental Validation

Parameters

Surrogate Models

- ASM - aggregated surrogate model based on One-Class SVM and Regression SVM
- RASM - proposed Rank-based SVM

SVM Learning

- Number of training points: at most $N_{training} = 1000$ points
- Number of iterations: $1000 \left|\Omega_{active}\right|$ + $\left|\Omega_{active}\right|^2 \approx 2N^2_{training}$
- Kernel function: RBF function with σ equal to the average distance of the training points
- The cost of constraint violation: $C = 1000$

Offspring Selection

- Number of pre-children: $p = 2$ and $p = 10$

Experimental Validation

Comparative Results

ΔHtarget	1	0.1	0.01	1e-3	1e-4	1	0.1	0.01	1e-3	1e-4
			ZDT1					ZDT2		
Best	1100	3000	5300	7800	38800	1400	4200	6600	8500	32700
S-NSGA-II	1.6	2	2	2.3	1.1	1.8	1.7	1.8	2.3	1.2
ASM-NSGA p=2	1.2	1.5	1.4	1.5	1.5	1.2	1.2	1.2	1.4	**1**
ASM-NSGA p=10	**1**	**1**	**1**	**1**	.	**1**	**1**	**1**	**1**	.
RASM-NSGA p=2	1.2	1.4	1.4	1.6	**1**	1.3	1.2	1.2	1.5	**1**
RASM-NSGA p=10	**1**	1.1	1.1	1.5	.	1.1	**1**	**1**	1.2	.
MO-CMA-ES	16.5	14.4	12.3	11.3	.	14.7	10.7	10	10.1	.
ASM-MO-CMA p=2	6.8	8.5	8.3	8	.	5.9	8.2	7.7	7.5	.
ASM-MO-CMA p=10	6.9	10.1	10.4	12.1	.	5	.	.	.	.
RASM-MO-CMA p=2	5.1	7.7	7.6	7.4	.	5.2	.	.	.	.
RASM-MO-CMA p=10	3.6	4.3	4.9	7.2	.	3.2	.	.	.	.
			IHR1					IHR2		
Best	500	2000	35300	41200	50300	1700	7000	12900	52900	.
S-NSGA-II	1.6	1.5	.	.	.	1.1	3.2	6.2	.	.
ASM-NSGA p=2	1.2	1.3	.	.	.	**1**	3.9	4.9	.	.
ASM-NSGA p=10	**1**	1.5	.	.	.	1.4	6.4	4.6	.	.
RASM-NSGA p=2	1.2	1.2	.	.	.	1.5	.	.	.	.
RASM-NSGA p=10	**1**	**1**	.	.	.	1.2	5.1	4.8	.	.
MO-CMA-ES	8.2	6.5	1.1	1.2	1.2	5.8	2.7	2.1	**1**	.
ASM-MO-CMA p=2	4.6	2.9	**1**	**1**	**1**	3.1	1.6	1.4	1.1	.
ASM-MO-CMA p=10	9.2	6.1	1.3	1.2	.	5.9	2.6	2.4	.	.
RASM-MO-CMA p=2	2.6	2.3	2.4	2.1	.	2.2	**1**	**1**	.	.
RASM-MO-CMA p=10	1.8	1.9	.	.	.	.	.	.	.	.

ASM and Rank-based ASM applied on top of NSGA-II (with hypervolume secondary criterion) and MO-CMA-ES, on ZDT and IHR functions.

N = How many more true evaluations than best performer

Discussion 1. Preferences $\rightarrow$ Optimization

Preference learning for robust black-box optimization

- ACM-ES: speed-up ($\times 2$, $\times 4$) on the state of the art on uni-modal problems.
- Invariant to rank-preserving and orthogonal transformations of the search space
- The cost of speed-up is $O(d^3)$
- Source codes available: `https://www.lri.fr/~ilya/`

Lessons learned

- Preference learning repeatedly used in the optimization platform
- Hyper-parameter adjustment is critical
- ML assessment criterion: not the average case: the **worst** case
  a critical hyper-parameter: the stopping criterion.
  Ill-conditioned Gram matrices are encountered with probability 1.

Discussion 2. Optimization $\rightarrow$ Preferences ?

Eliciting preferences ?

- Subjectiveness is an issue (phrasing of questions, choice of units)
- Designing a questionaire: an optimization problem ?
- Which criterion: stability ?

Designing aggregation operators / voting rules ?

- A learning or an optimization problem ?

References

Black box optimization

Arnold, L., Auger, A., Hansen, N., and Ollivier, Y. Information-Geometric Optimization Algorithms: A Unifying Picture via Invariance Principles. ArXiv e-prints, 2011.

Hansen, N., Ostermeier, A. Completely derandomized self-adaptation in evolution strategies. Evolutionary computation 9 (2), 159-195, 2001.

Surrogate models for optimization

Loshchilov, I., Schoenauer, M., Sebag, M. Self-adaptive surrogate-assisted covariance matrix adaptation evolution strategy. GECCO 2012, ACM Press: 321-328, 2012.

—, A mono surrogate for multiobjective optimization. GECCO 2010, ACM Press: 471-478.

Related

Viappiani, P., Boutilier, C. Optimal Bayesian Recommendation Sets and Myopically Optimal Choice Query Sets. NIPS 2010: 2352-2360

Hoos, H. H. Programming by optimization. Commun. ACM 55(2): 70-80.