Per Instance Algorithm Configuration of CMA-ES with Limited Budget

Nacim Belkhir, Johann Dréo, Pierre Savéant, Marc Schoenauer

COSEAL workshop, September 20, 2017

Context

Continuous Black Box optimization with small budget

- Optimize $\mathcal{F} : \mathbb{R}^d \mapsto \mathbb{R}$
- No prior knowledge on the objective function $\mathcal{F}$
- But algorithm selection / configuration is problem-dependent

$\longrightarrow$ PIAC – Per Instance Algorithm Configuration

Problem Features

- Characterize the objective function
- Used for instance-based Algorithm Selection[1] / Algorithm Configuration[2]
- Computed from a sample of objective function values $(x_i, \mathcal{F}(x_i))_{i=1,\dots n})$
- ☹ But usually requires a large sample $n \geq 500 \times d$

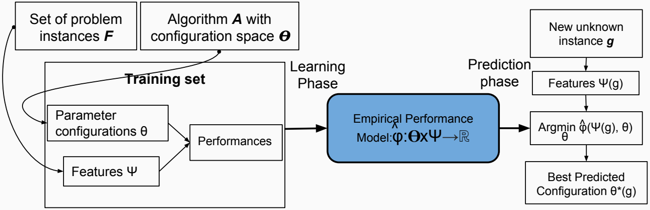
How to deal with the small budget constraint?

[1] Mersmann et al. 2011; Bischl et al. 2012; Kadioglu et al. 2010; Abell, Malitsky, and Tierney 2012.

[2] Muñoz, Kirley, and Halgamuge 2012; Bossek et al. 2015; Belkhir et al. 2016a.

Per Instance Algorithm Configuration

Per Instance Algorithm Configuration



Algorithm Runtime Prediction for offline parameter setting based on problem features

- Heavily relies on problem features.
- Empirical Performance Model (EPM) used for algorithm selection / configuration . Leyton-Brown et al. 2003; Hutter et al. 2014
- Successfuly applied in combinatorial domains Hutter et al. 2006 and continuous domains Abell, Malitsky, and Tierney 2012; Muñoz, Kirley, and Halgamuge 2012

Limitation: features might be expensive to compute

Continuous Problem Features

Continuous Problem Features

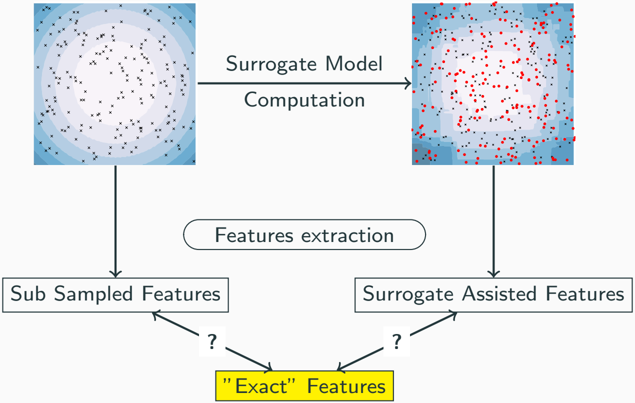
Feature Classes

- Y-**Distribution (3)**, **Meta model (9)**, **Level Set** (18) Mersmann et al. 2011
- **Information Content (5)** Munoz, Kirley, Halgamuge, et al. 2015
- **Dispersion** (16) Lunacek and Whitley 2006
- Curvature (14), Convexity (4), Local Search (7) Mersmann et al. 2011
  . Require additional evaluations

- Help to charaterize Fitness Landscape properties (multimodality, separability, levelset, plateaus, search space homogeneity,...)
- Require large samples to be accurately computed ($\geq 500 \times d$)
- Publicly available R package Pascal Kerschke, http://github.com/flacco
- Soon available as Python package ...

Sub-sampled and Surrogate-Assisted Features

- Proposed in Belkhir et al. 2016b
- Use small samples: $\leq 100 \times d$



Experimental Setting

Experimental Setting

Target Algorithm

- BIPOP-CMA-ES
  - restarts
  - two populations
  - doubling trick for large population
- Parameters of the Covariance Matrix Adaptation c_1, c_c, c_μ

Training Phase

- BBOB testbench[3]:
  - 24 test functions with known optimum and known properties (non-convex, multi-modal, separable)
  - $d \in \{2, 4, 5, 8, 10, 16, 20, 32, 40, 64\}$
  - Overvall budget: $10^3 \times d$ and target precision $\Delta_f = 10^{-6}$
- **Features**
  Y-Distribution, Meta model, Level Set, Information Content, Dispersion
- **Empirical Performance Model**
  - Random Forest regression (10 trees, maximal depth 200, scikit-learn implementation)

[3] Hansen et al. 2010.

Experimental Setting (2)

Validation Set

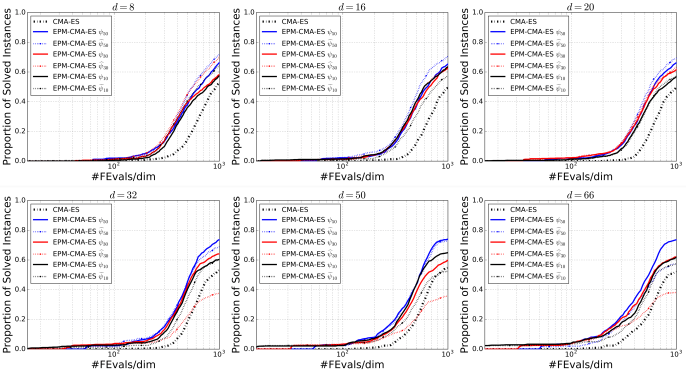
- **not** BBOB
- 21 test functions with known optimum and known properties (non-convex, multi-modal, separable)
- $d \in \{2, 4, 8, 10, 16, 20, 32, 40, 50, 66, 100\}$
- 15 independent runs
- Overvall budget: $10^3 \times d$ and target precision $\Delta_f = 10^{-6}$

Comparing

- BIPOP-CMA-ES default parameter setting
- PIAC approaches
  - Different sample sizes $k \times d$ with $k \in \{10, 30, 50\}$
  - Sub-sampled features vs surrogate-assisted features (Random Forest again)
- Alternative restart strategy for BIPOP-CMA-ES
  - Recomputing the features at each restart

Results

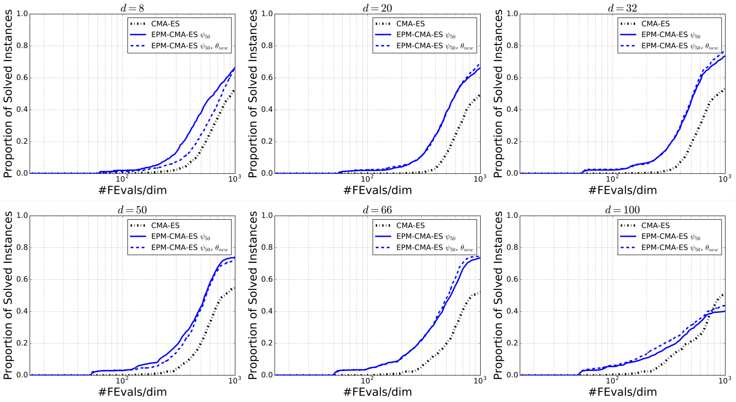
Sub-Sampled vs Surrogate-assisted Features



ECDF comparing EPM-CMA-ES with ψ_k or $\widehat{\psi_k}$ ($k \in \{\mathbf{10}, \mathbf{30}, \mathbf{50}\}$), and default CMA-ES

Discussion

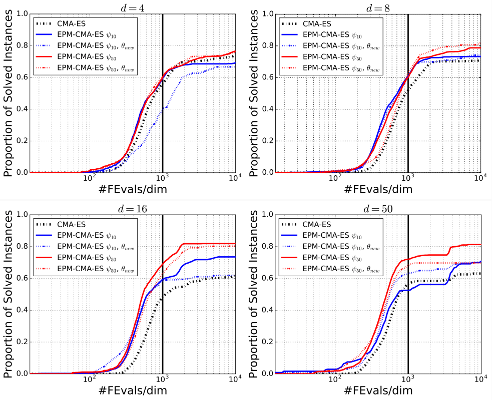
- even $k = 10$ improves on standard CMA-ES
- Best performances for $k = 50$ no surprise
- for $d > 20$, Sub-sampled features outperform Surrogate-assisted features



ECDF comparing ψ_{50} without and with the alternative restart strategy (θ_{new}), and the default CMA-ES

Discussion

- Alternative strategy is not beneficial



Typical Results of ECDF of EPM-CMA-ES compared to CMA-ES beyond the $10^3 \times d$ initial budget limit

Discussion

- EPM-CMA-ES with sub-sampled features behaves like CMA-ES

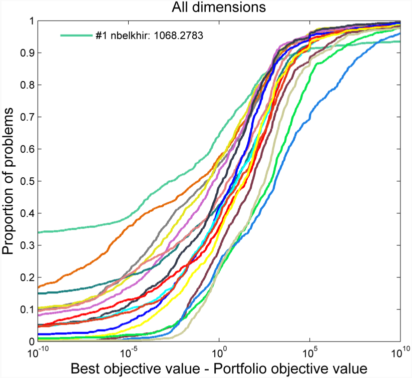
BBComp GECCO2017 One-objective track :-)

Algorithm

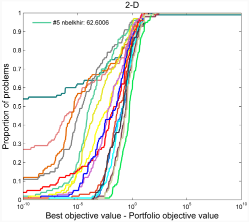
- $d < 10$: EPM-Algorithm Selection (1+1)-CMA-ES, BIPOP-CMA-ES, restart NM, LBFGS, DE
- $d \geq 10$: This talk, $k = 50$
- EPMs trained on full BBOB
- 10% budget for final Nelder-Mead

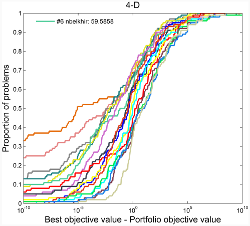
Results

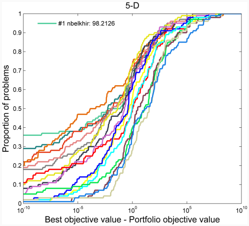
- Overall rank 1/17
- 1st in dim 5, 10, 20, 32 and 40
- 2nd in dim 8
- poor in dim 2, 4
- failed to solve half instance in dim. 64

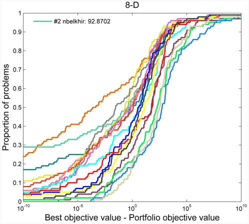


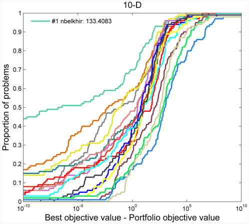
BBComp - detailed results 1/2



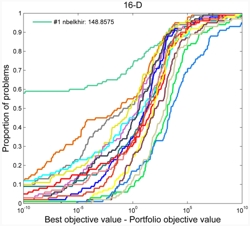


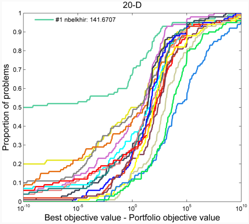


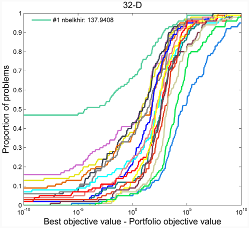


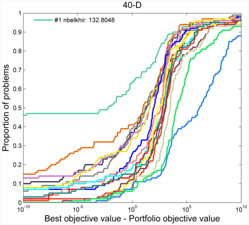


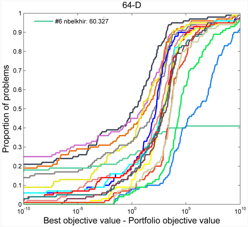
BBComp - detailed results 2/2











Conclusion and Further Works

Conclusion

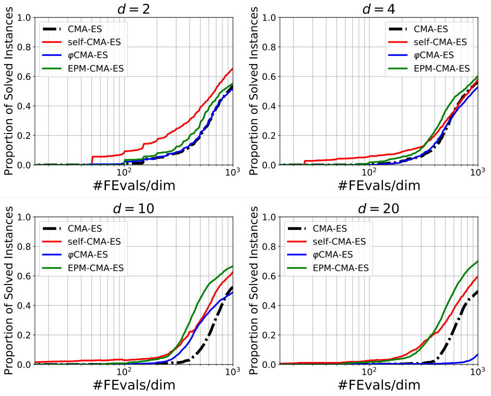
- Empirically validated PIAC for CMA-ES with limited budget
  - Trained on BBOB
  - Validated on other testbed e.g., BBComp
- Sub-sampled features behave reasonably well

Further work

- Rank based regression for EPM learning $\longrightarrow$ did not improve as expected
- Full Algorithm Selection and Configuration as sketched for BBComp
- Online parameter control

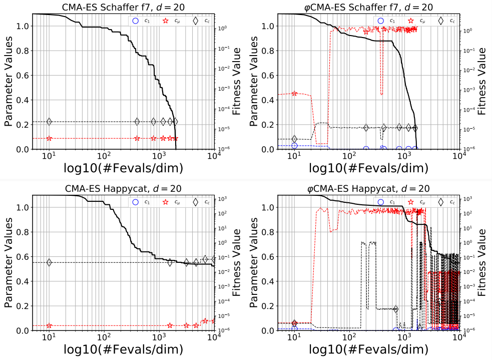
Toward Online Per Instance Parameter Control

- Embedding the EPM into a parameter control mechanism
- Features computation with the current population at each iteration



ECDF comparing φCMA-ES to the original version of CMA-ES, EPM-CMA-ES, and self-CMA-ES

Toward Online Per Instance Parameter Control (2)



Example of a run of median performance of CMA-ES (left) and φCMA-ES (right). The parameter values and the best f_{min} values are displayed.

Next

- sliding sample set, and/or weighted parameters update
- select (best) samples, ... and a lot more

References

- Abell, Tinus, Yuri Malitsky, and Kevin Tierney (2012). *Fitness landscape based features for exploiting black-box optimization problem structure*. Tech. rep. TR-2012-163. IT University of Copenhagen.
- Belkhir, Nacim et al. (2016a). "Feature Based Algorithm Configuration: A Case Study with Differential Evolution". In: *International Conference on Parallel Problem Solving from Nature*. Springer, pp. 156–166.
- – (2016b). "Surrogate assisted feature computation for continuous problems". In: *International Conference on Learning and Intelligent Optimization*. Springer, pp. 17–31.
- Bischl, Bernd et al. (2012). "Algorithm selection based on exploratory landscape analysis and cost-sensitive learning". In: *Proceedings of the 14th annual conference on Genetic and evolutionary computation*. ACM, pp. 313–320.
- Bossek, Jakob et al. (2015). "Learning Feature-Parameter Mappings for Parameter Tuning via the Profile Expected Improvement". In: *Proceedings of the 2015 on Genetic and Evolutionary Computation Conference*. ACM, pp. 1319–1326.
- Hansen, Nikolaus et al. (2010). *Real-Parameter Black-Box Optimization Benchmarking 2010: Experimental Setup*. Tech. rep. RR-7215. INRIA.

Hutter, Frank et al. (2006). "Performance prediction and automated tuning of randomized and parametric algorithms". In: *International Conference on Principles and Practice of Constraint Programming*. Springer, pp. 213–228.

Hutter, Frank et al. (2014). "Algorithm runtime prediction: Methods & evaluation". In: *Artificial Intelligence* 206, pp. 79–111.

Kadioglu, Serdar et al. (2010). "ISAC - Instance-Specific Algorithm Configuration". In: *ECAI*. Vol. 215, pp. 751–756.

Leyton-Brown, Kevin et al. (2003). "A portfolio approach to algorithm selection". In: *IJCAI*. Vol. 1543, p. 2003.

Lunacek, Monte and Darrell Whitley (2006). "The dispersion metric and the CMA evolution strategy". In: *Proc. 8th GECCO*. ACM, pp. 477–484.

Mersmann, Olaf et al. (2011). "Exploratory landscape analysis". In: *Proc. 13th GECCO*. ACM, pp. 829–836.

Munoz, Mario, Michael Kirley, Saman K Halgamuge, et al. (2015). "Exploratory landscape analysis of continuous space optimization problems using information content". In: *Evolutionary Computation, IEEE Transactions on* 19.1, pp. 74–87.

Muñoz, Mario A, Michael Kirley, and Saman K Halgamuge (2012). "A meta-learning prediction model of algorithm performance for continuous optimization problems". In: *PPSN XII*. Springer, pp. 226–235.