Anytime Bi-Objective Optimization with a Hybrid Multi-Objective CMA-ES (HMO-CMA-ES)

COCO/BBOB Workshop
GECCO, July 2016

Ilya Loshchilov
Research Group Machine Learning
for Automated Algorithm Design
University of Freiburg, Germany

Tobias Glasmachers
Institute for Neural Computation
Ruhr-University Bochum, Germany

Outline

- Motivation: Hybrid Algorithm
- Algorithm Components
- HMO-CMA-ES Algorithm
- Results
- Conclusion

Motivation

- Optimization methods have different strengths and weaknesses. There is no single best optimizer for all problems.
- This is because problems are diverse and non-trivial optimizers necessarily have biases.
- The situation is no different in multi-objective optimization.
- Therefore it is tempting to design a hybrid (master) algorithm applying different component algorithms to different problems.
- Such hybrid algorithms bear the potential to perform well on all problems covered by their algorithm portfolio.

Motivation

- Methods differ in how and when the components are selected and executed.
- The proposed approach is analogous to H-CMA-ES for single objective optimization.
- A straightforward strategy it to decide for the best algorithm after an exploration phase.
- Messing up a hard decision can result in bad performance and is therefore fragile. Let's avoid that.

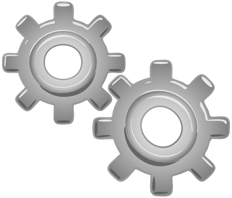
Motivation: Sequential Execution

- Observation: not all problems are equally hard. A natural strategy is to solve easy problems first.
- Say, method A solves some problems in 10^3 iterations and method B solves the remaining problems in 10^6 iterations.
- Strategy: first apply A for 10^3 iterations, then apply B for another 10^6 iterations.
- Rationale: No loss on easy problems, only 0.1% loss on hard problems.
- In addition, B may profit from the warm-start.

Motivation: Parallel Execution

- Now assume that algorithms C and D take roughly the same time to solve the problems they are specialized in.
- Observation: we often get the lion's share of the performance gain in the initial phase of an optimization run, while towards the end we perform fine tuning.
- For optimal anytime behavior C and D should be run in parallel.
- In out setting this means that they should run interleaved. Each algorithm gets every second fitness evaluation.
- We may migrate the best solutions between runs from time to time (island model).

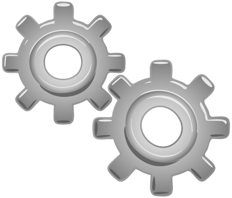
Components



HMO-CMA-ES combines the following established techniques:

- (MO)-CMA-ES.
- Model-based optimization.
- Scalarization vs. "proper" multi-objective selection based on the contributed hypervolume.
- Steady-state selection vs. generational selection.
- Restarts.
- Increasing population size.

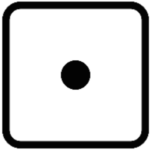
Components



HMO-CMA-ES relies on the following algorithm portfolio:

1. BOBYQA on a scalarized objective function as a warm start.
2. Steady-state MO-CMA-ES in our version with increasing population size (ss-MO-CMA-ES).
3. Our version of CMA-ES with restarts on different scalarized objectives (restart-CMA-ES).
4. Generational MO-CMA-ES in our version with restarts denoted as IPOP-MO-CMA-ES.

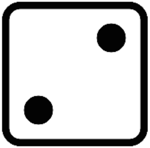
The HMO-CMA-ES Algorithm



Phase 1: function evaluations $0 — 10D$

- Goal: quick approach of the Pareto front, in particular on easy problems.
- Approach: BOBYQA with different linear scalarization functions.

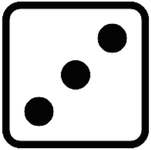
The HMO-CMA-ES Algorithm



Phase 2: function evaluations $10D$ — $1,000D$

- Goal: systematic approach of the Pareto front.
- Approach: ss-MO-CMA-ES with increasing population size and crossover.

The HMO-CMA-ES Algorithm



Phase 3: function evaluations $1,000D$ — $20,000D$

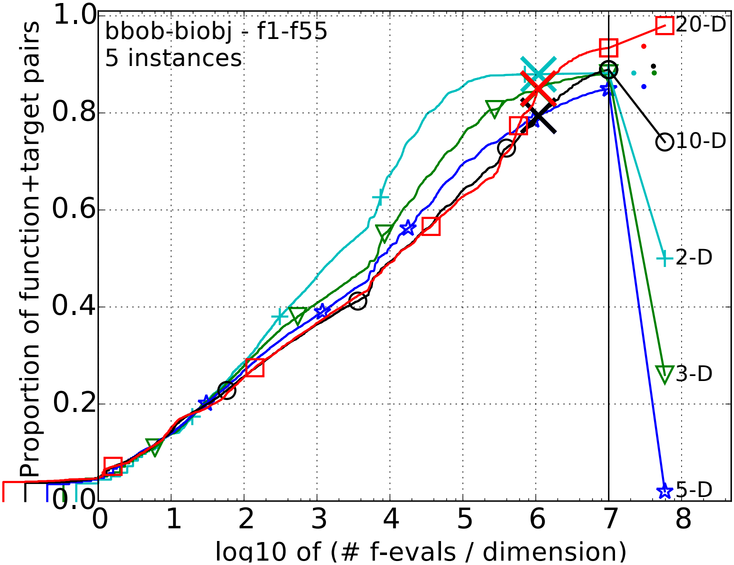
- Goal: keep tuning the front for easy problems, start to explore multi-modal landscapes.
- Approach: ss-MO-CMA-ES with increasing population size and crossover (as before), in parallel with scalarized restart CMA-ES. Optima found by CMA-ES are migrated to ss-MO-CMA-ES.

The HMO-CMA-ES Algorithm

Phase 4: function evaluations $20,000D$ — $1,200,000D$

- Goal: solve multi-modal problems, fine-tune hypervolume coverage.
- Approach: keep both previous algorithms running, and launch IPOP-MO-CMA-ES in addition in parallel. Randomized migration of individuals to ss-MO-CMA-ES.

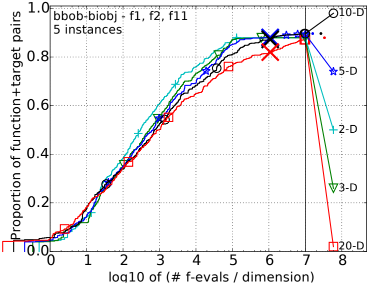
Results: Overall Performance



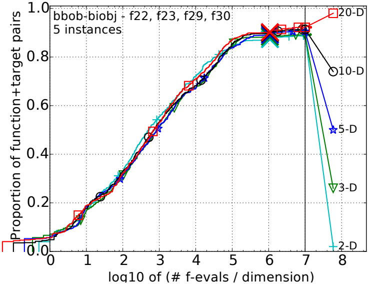
Results: Easy Problems

- We find that **separable**, **moderate** and **ill-conditioned** problems are easy for HMO-CMA-ES.
- They are often "solved" within $10^5 D$ iterations.

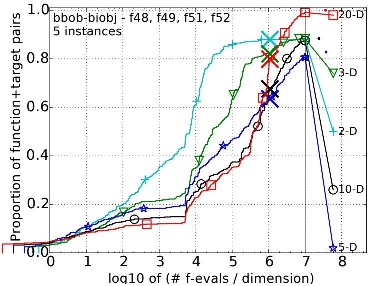
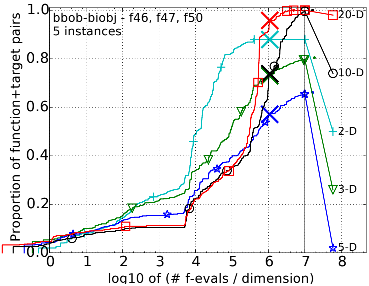
separable/separable



moderate/ill-cond.



- **multi-modal** and **weakly structured** problems are harder for HMO-CMA-ES.
- They are “solved” after $10^6 D$ iterations, if at all.
- Results for $D = 20$ are particularly encouraging.

multi-modal/multi-modal	multi-modal/weakly structured
	

Conclusion

- We have argued for solving unknown and/or diverse black-box problems with hybrid algorithms.
- Both sequential and parallel execution have their roles.
- We use a mix of techniques: different selection schemes, restarts, increasing population size, and model-based optimization as a warm start.
- We build on established components with good invariance properties, mostly variants of (MO)-CMA-ES. We don't try to exploit separability.

Conclusion

- HMO-CMA-ES solves most BBOB problem reasonably well, and many with high precision.
- Multi-modal problem still pose the greatest challenge. It is possible that the incorporation of additional building blocks can help with that.
- The exact composition of the portfolio is based on heuristics, and so is the choice of its parameters (population sizes and increments, details of initialization and warm start, phase lengths, ...).
- The resulting HMO-CMA-ES algorithm is somewhat tailored to BBOB, but we still expect it to perform reasonably well outside the lab.
- HMO-CMA-ES could be further improved with automated tuning tools like smac.

Thank you!
Questions?