Benchmarking as a Stepping Stone to Dynamic Algorithm Selection

Diederick Vermetten | Joint Lectures on Evolutionary Algorithms (JoLEA)

Universiteit Leiden
The Netherlands

Background

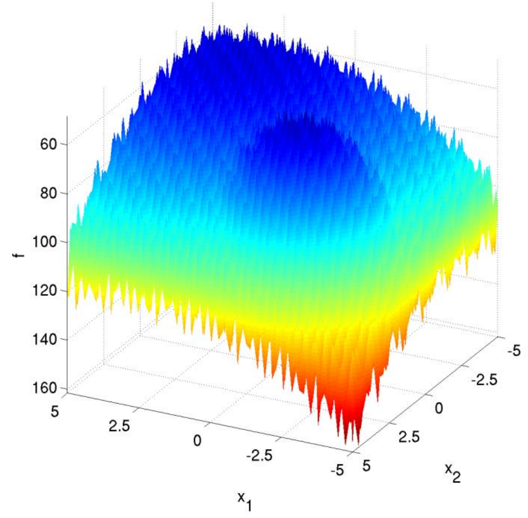
- Phd Candidate at LIACS in the naco group
- Main interests are benchmarking and dynamic algorithm selection / configuration
- Focus on continuous, single-objective, noiseless optimization

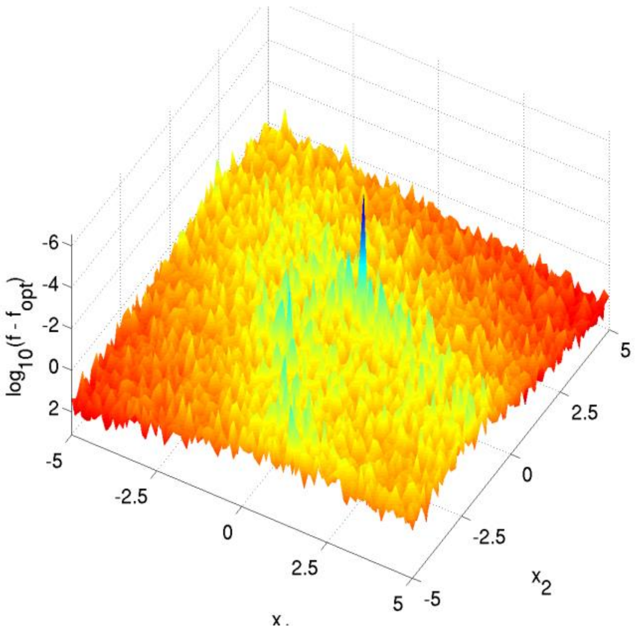
Universiteit Leiden
The Netherlands

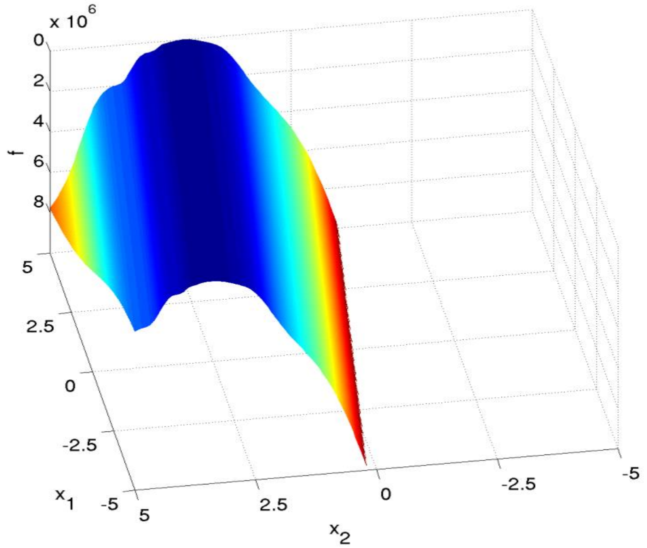
SORBONNE UNIVERSITÉ

Reasons for Benchmarking

- Potential goal: 'better' algorithms
- But what does 'better' mean?
- Algorithm behaviour is more than a single number
- Goal: benchmark to gain insight into behaviour

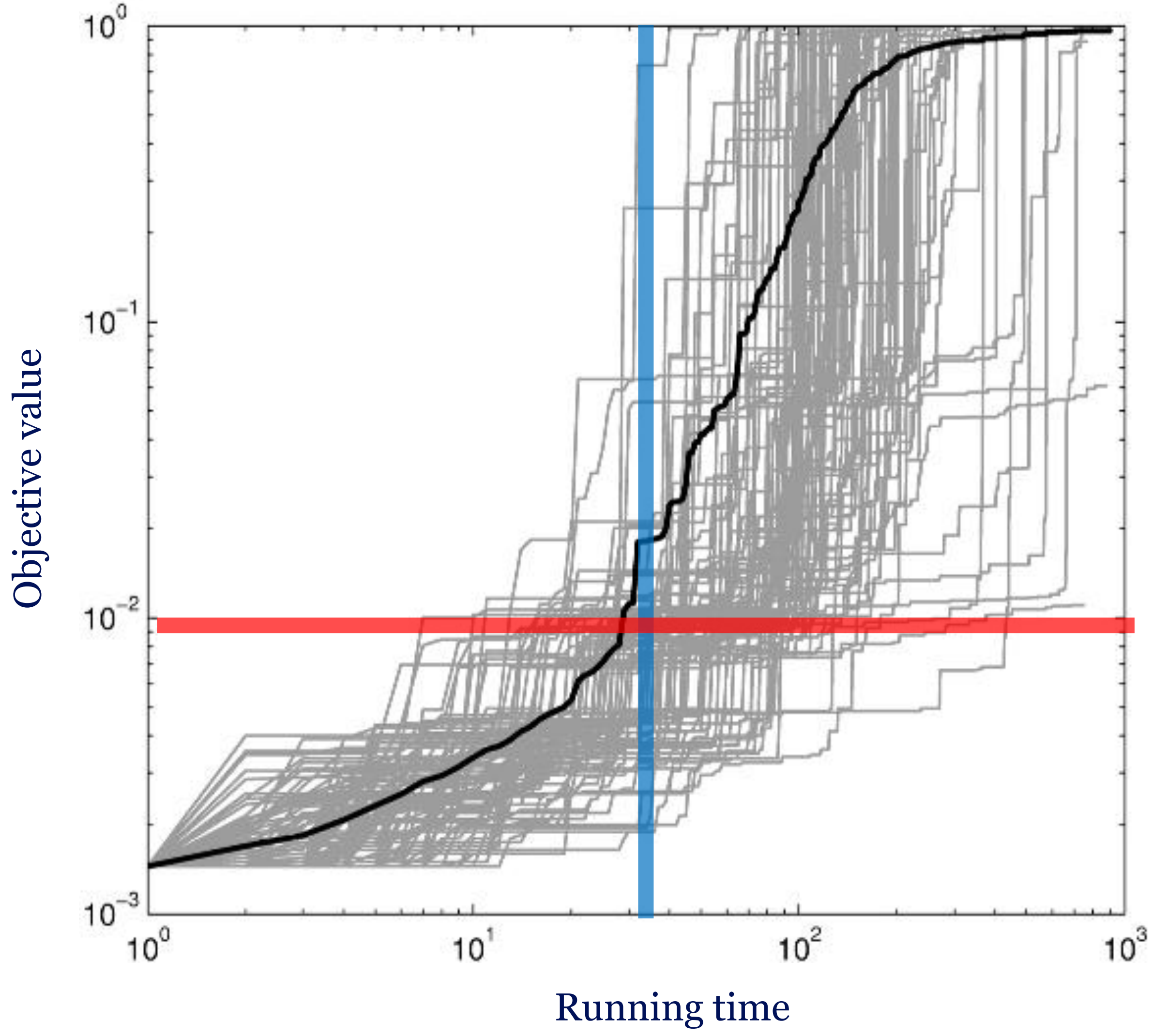






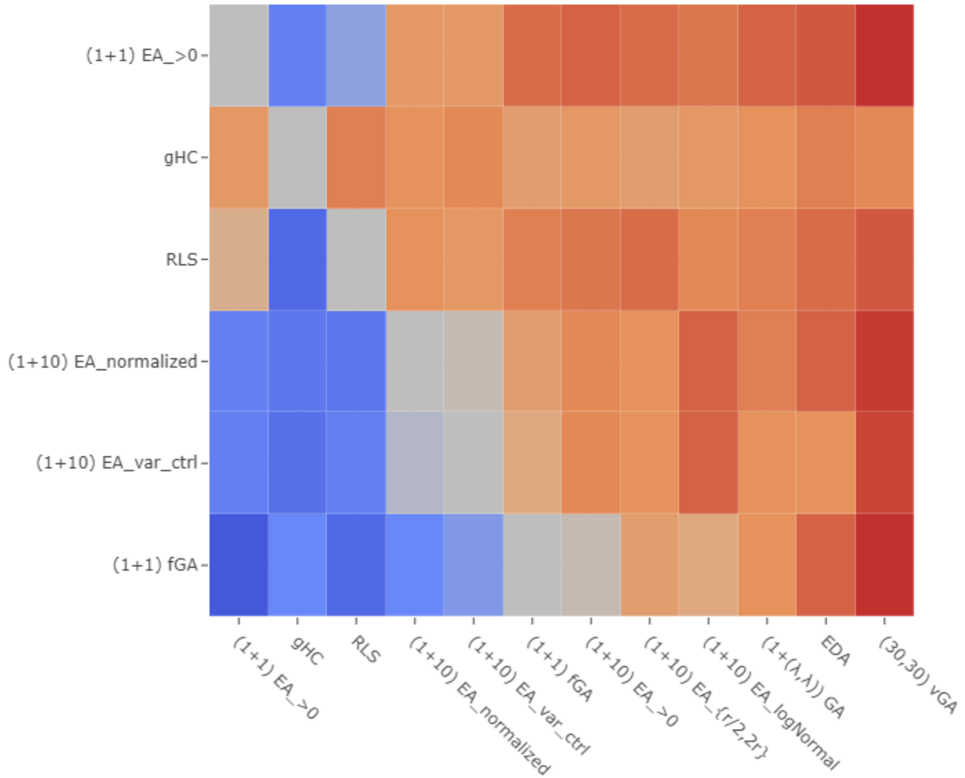
Benchmarking Perspectives

- One algorithm on one function
- Inherently stochastic
- Performance is based on perspective
- Information is lost when considering only one number / distribution

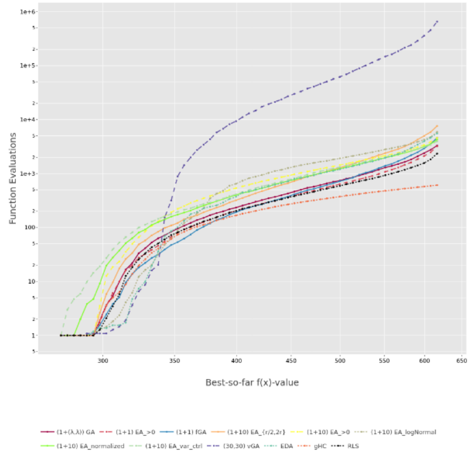


Benchmarking: IOHprofiler

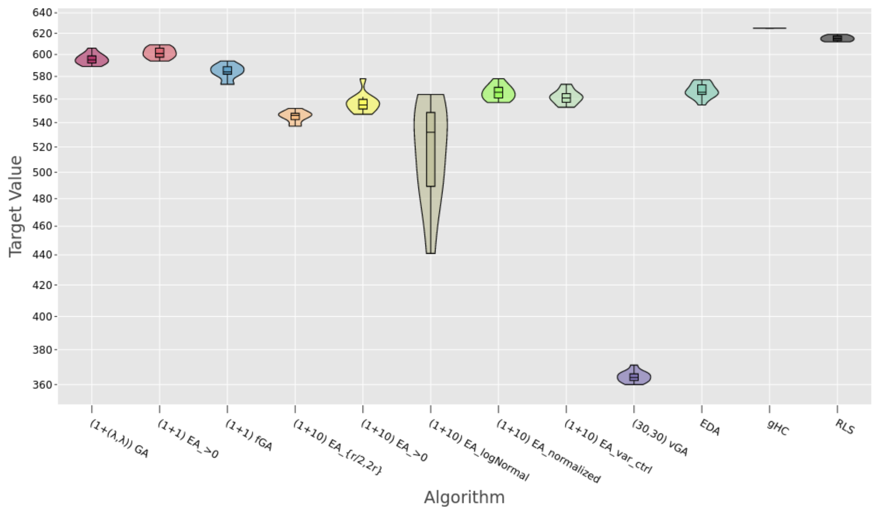
- The performance trajectory contains information on algorithm behaviour
- Tracking this allows for a wide variety of post-processing to be applied
- Can be done in an interactive manner



Expected Runtime (ERT)



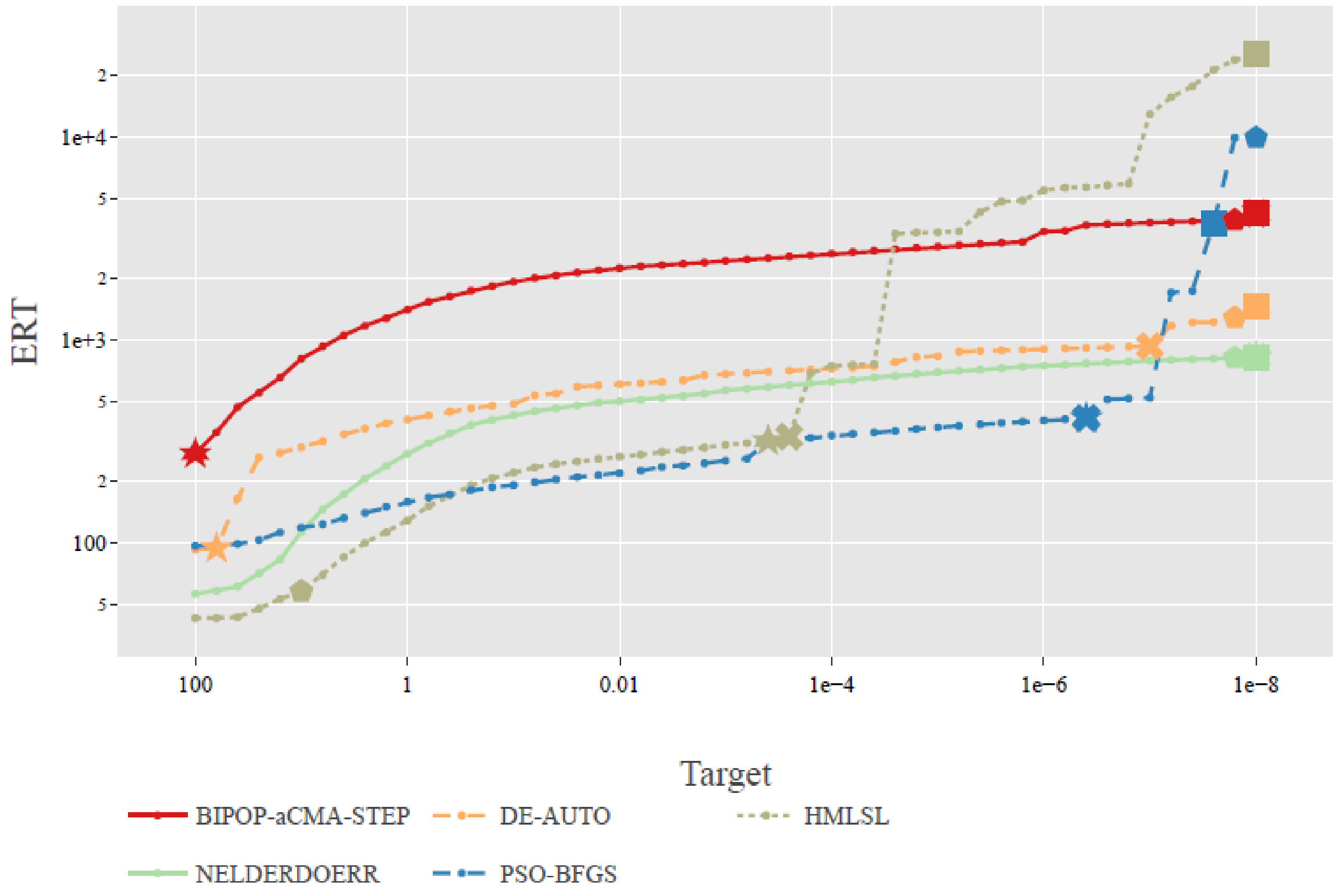
Fixed-Budget Probability Density



[1]

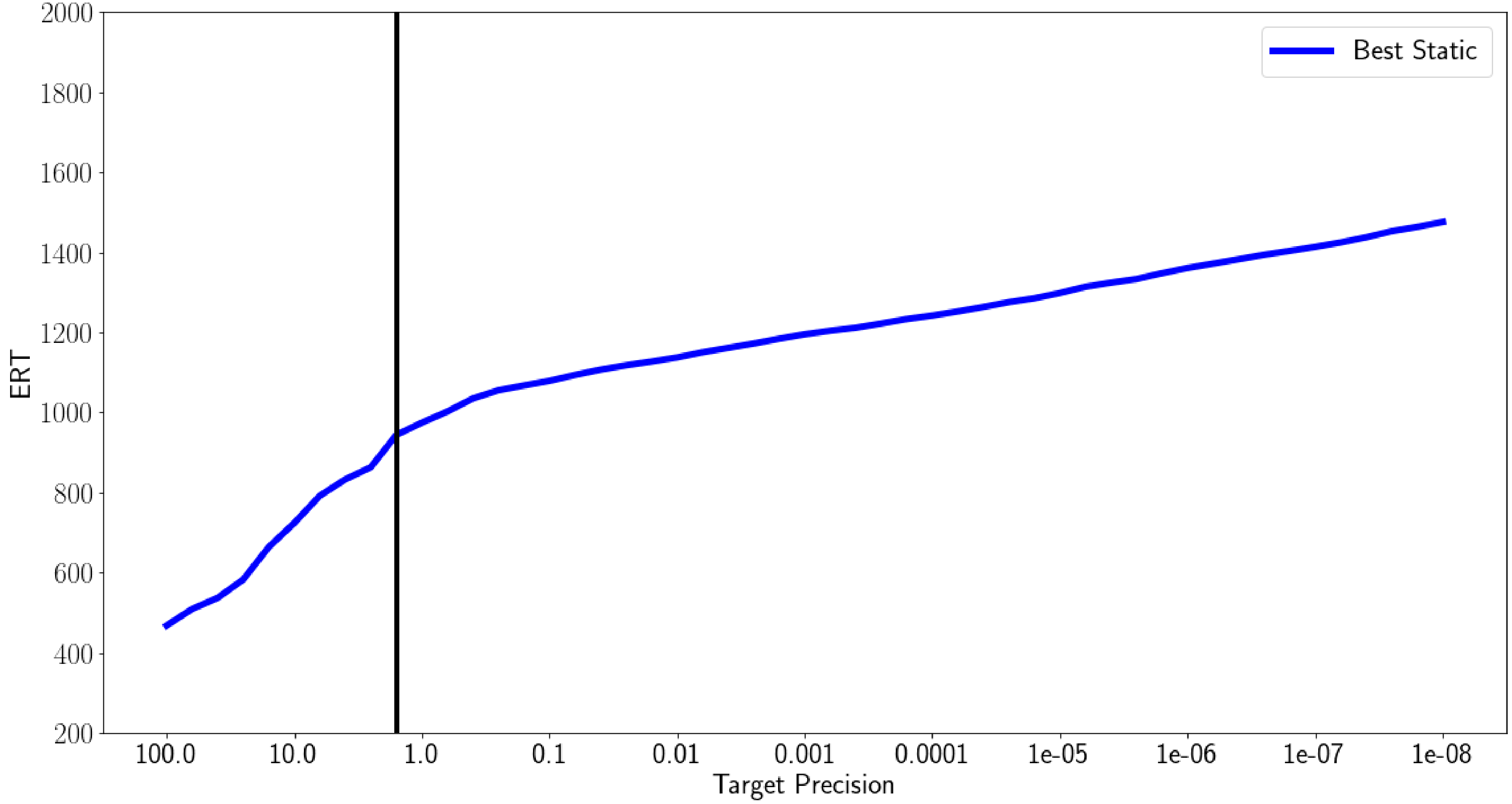
Benchmarking to Dynamic Algorithm Selection

- Looking at performance trajectories gives insight to when an algorithm performs well
- We even see large differences within functions
- We might be able to exploit this by switching between them



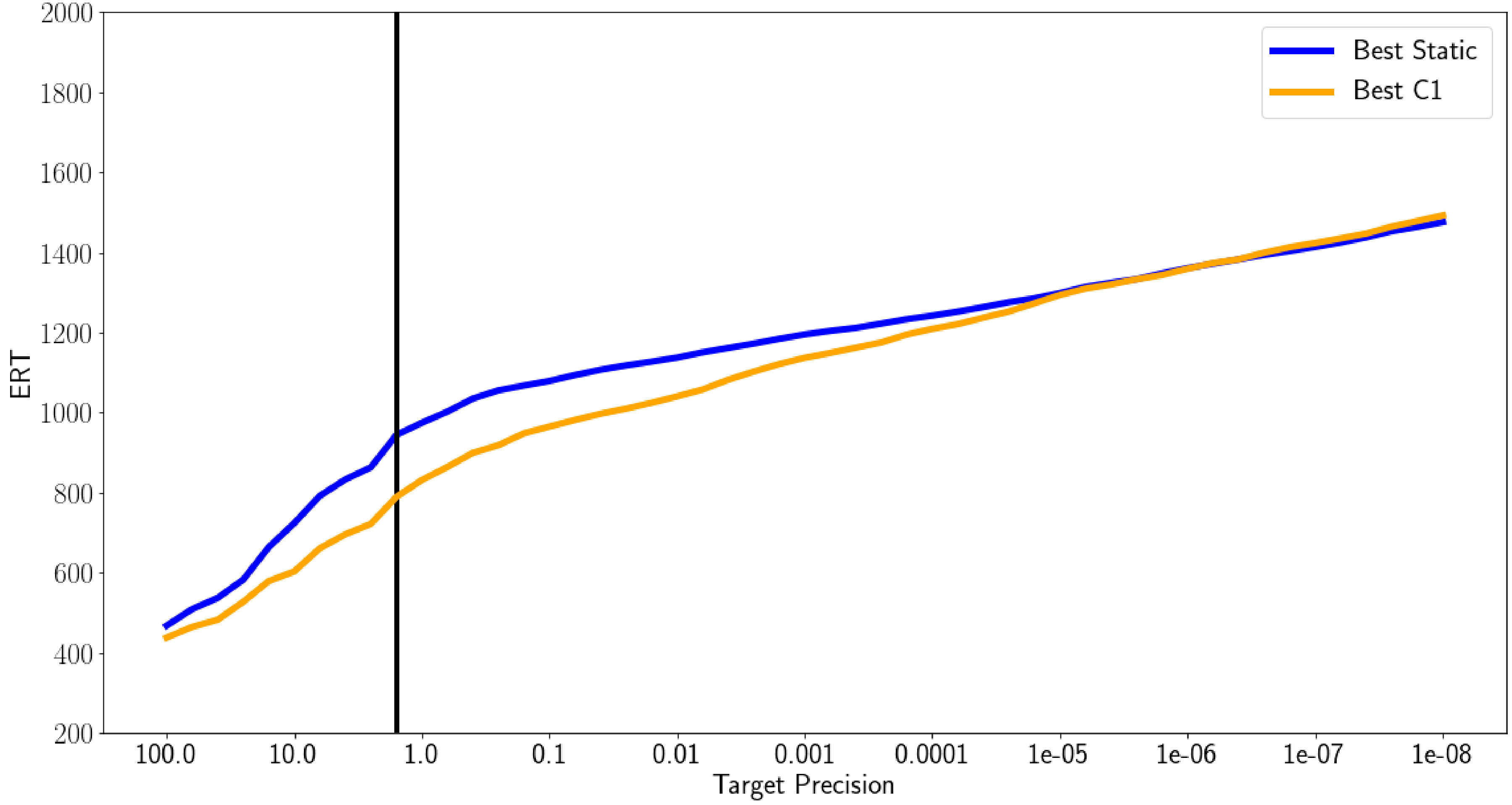
[2]

Idea Behind Dynamic Algorithm Selection



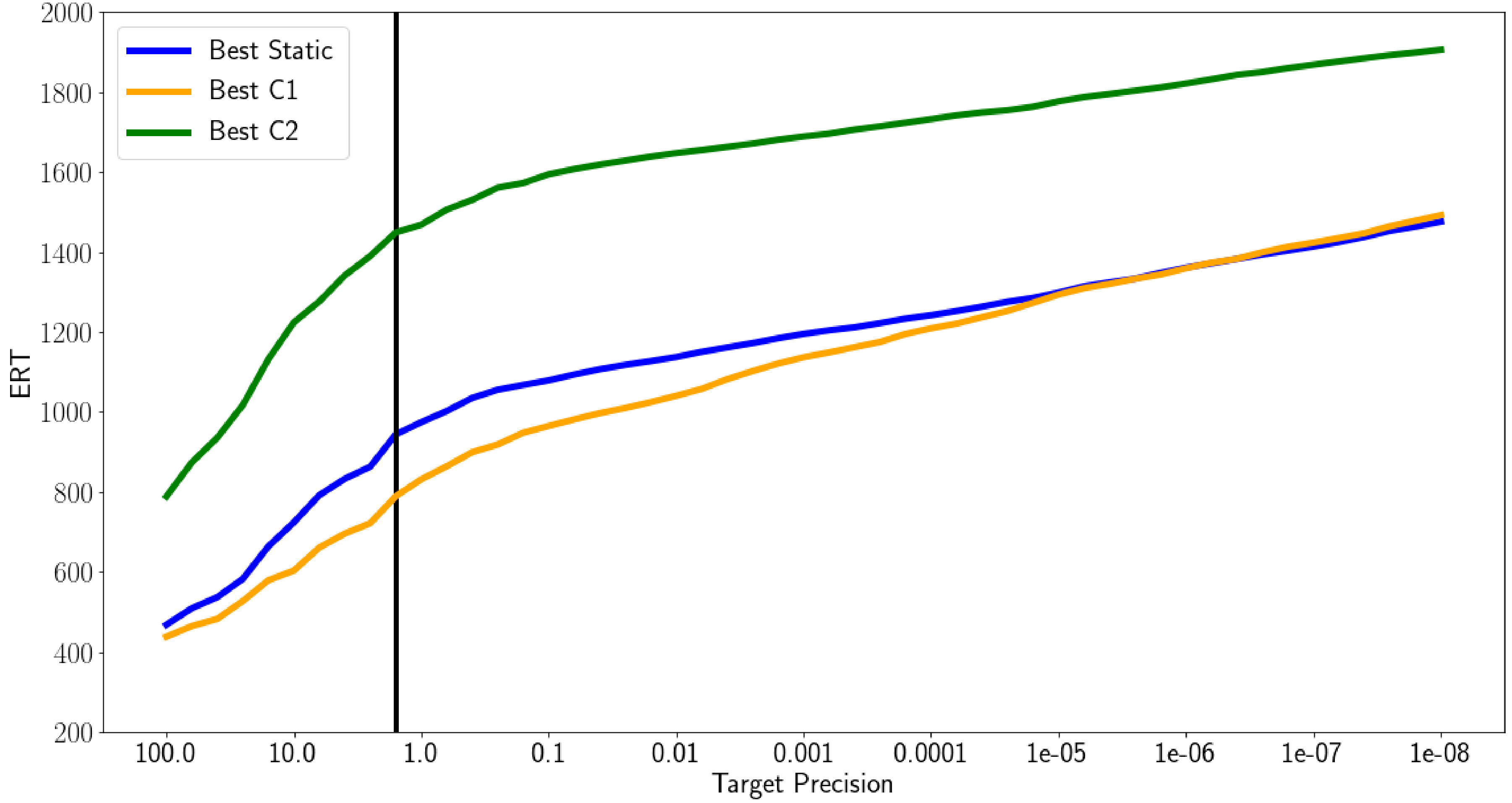
[3]

Idea Behind Dynamic Algorithm Selection



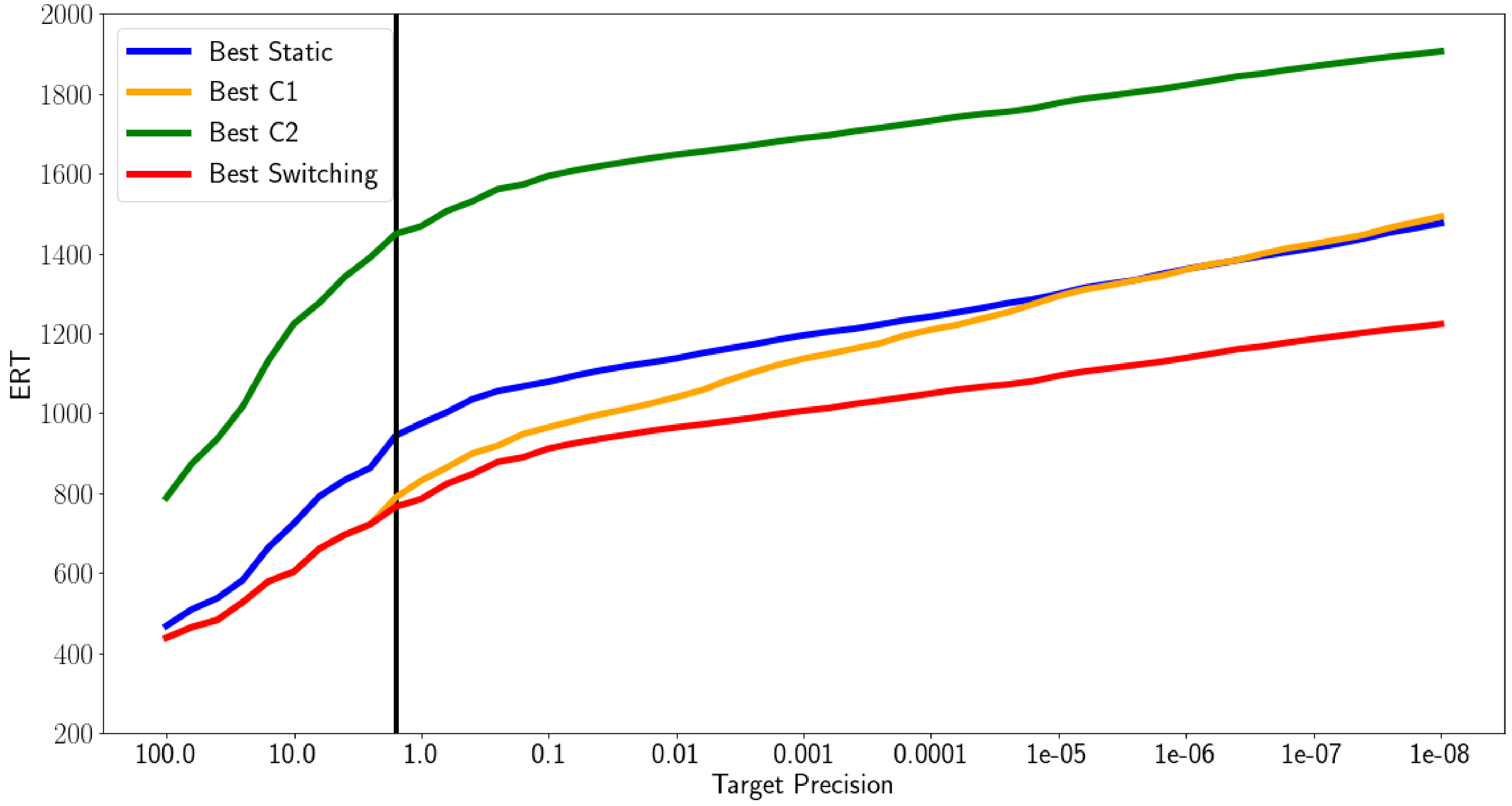
[3]

Idea Behind Dynamic Algorithm Selection



[3]

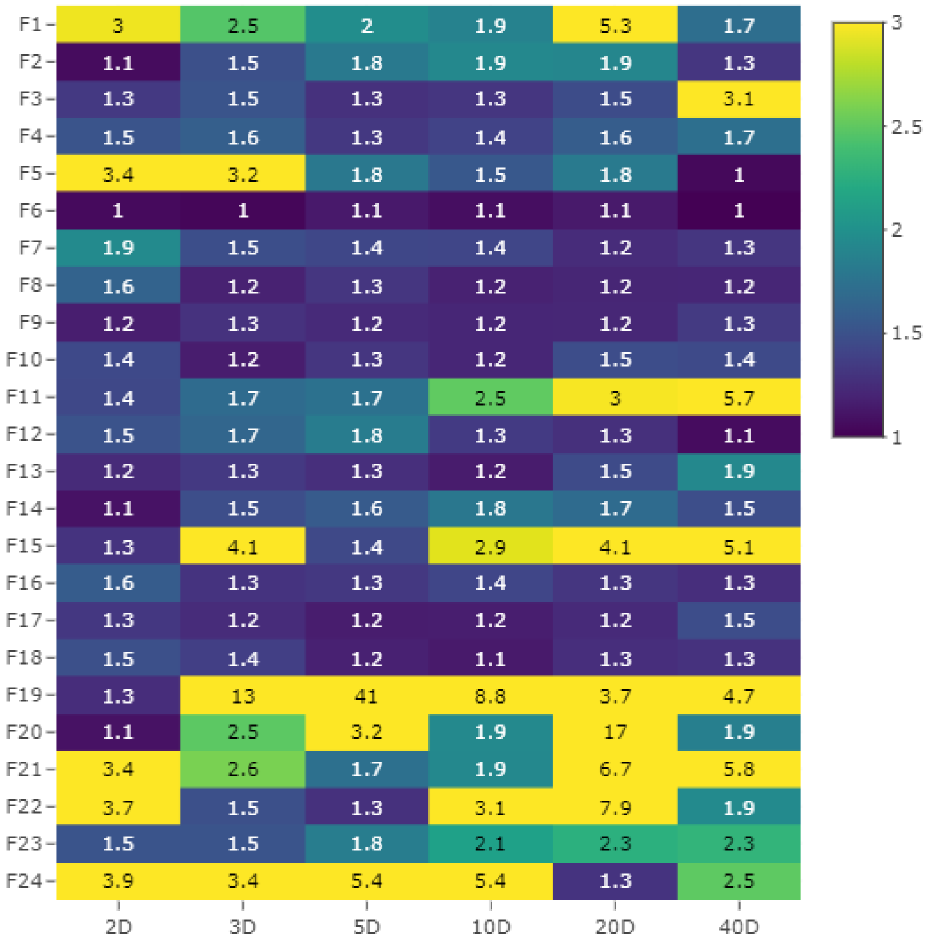
Idea Behind Dynamic Algorithm Selection



[3]

Potential of Dynamic Algorithm Selection

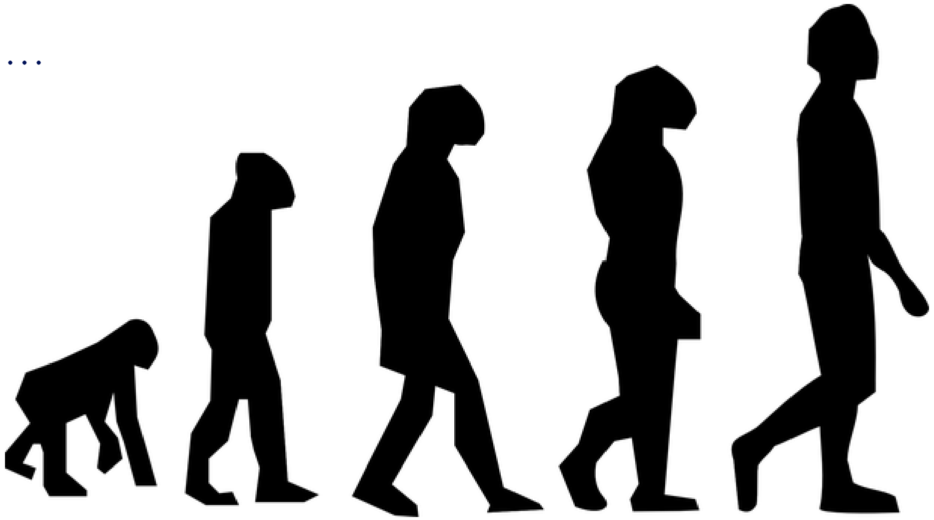
- Use available benchmark data from COCO, containing more than 100 algorithms
- Identify the potential of switching between any of them
- By ‘glueing’ together performance after reaching each target
- Can be used to identify smaller algorithm portfolios with the largest potential

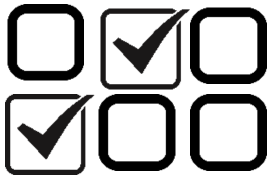


[2]

Usecase: CMA-ES

- Alternative: restrict to one algorithm family
- Allows for easier switching
- Modular CMA-ES framework:
    - Elitism
    - Active update
    - (B)IPOP
    - …

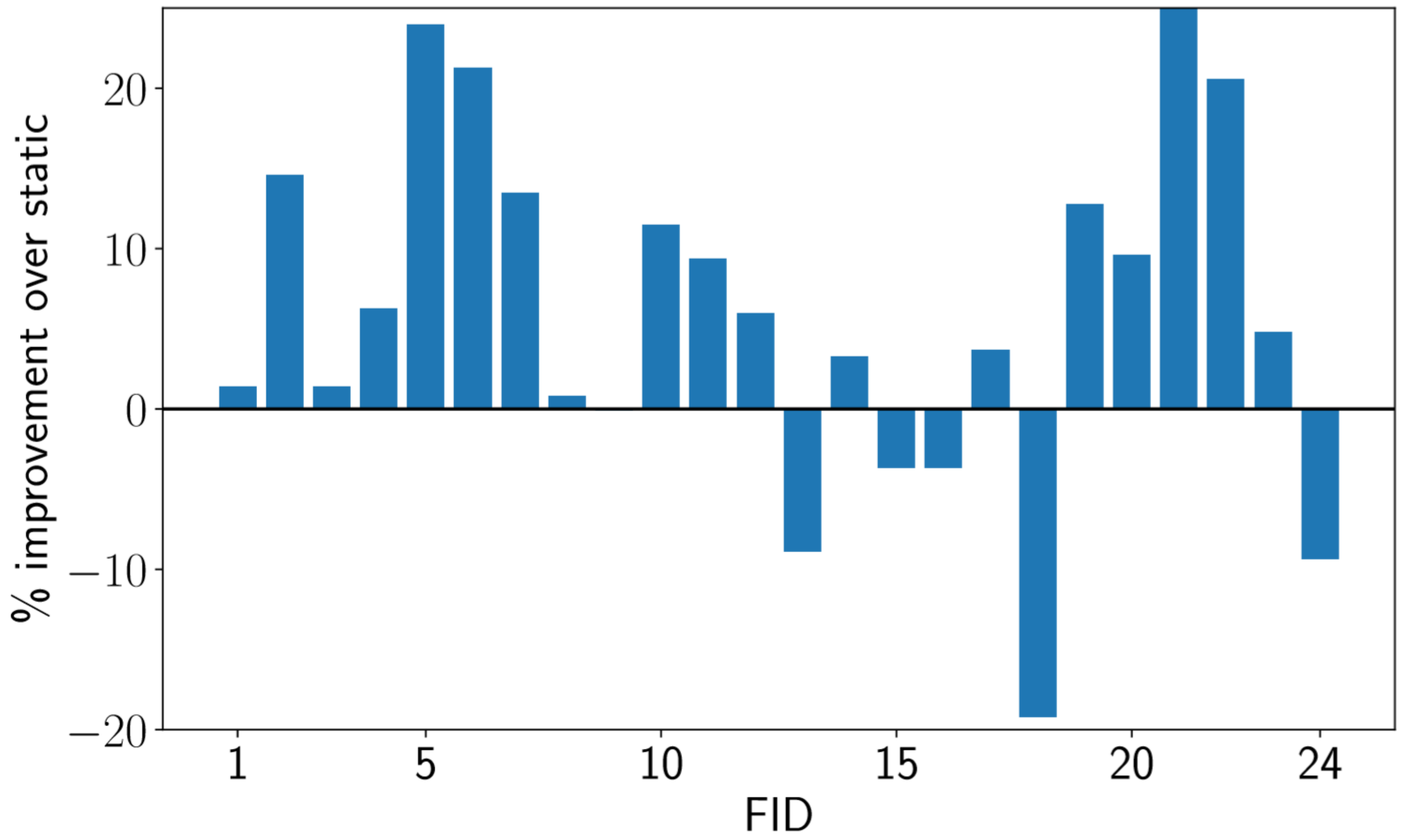




[4,5]

Switching between CMA-ES Variants

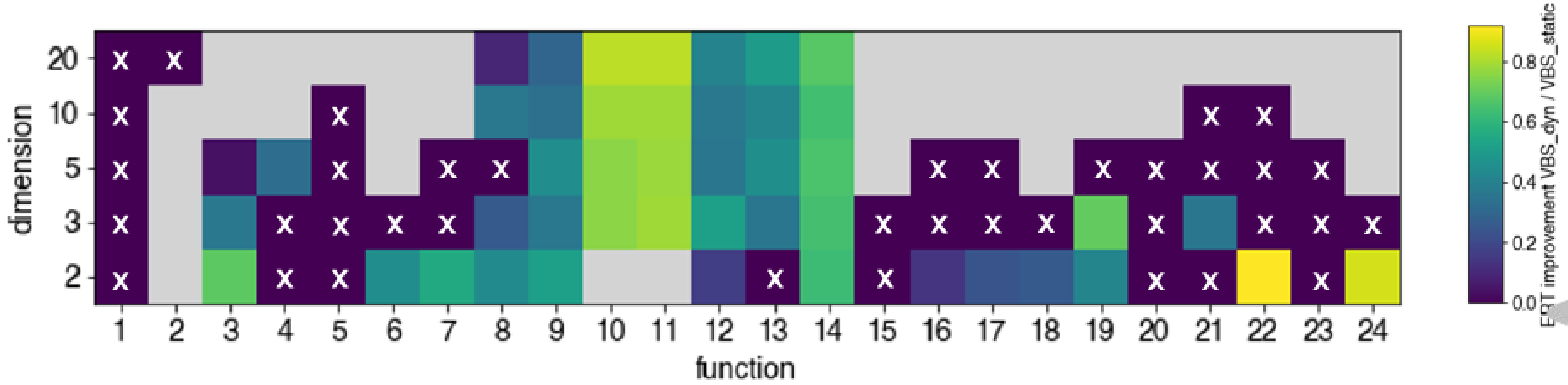
- Perform the switch identified by performance data
- Take top 50 combinations, rerun 250 times to reduce noise
- Compare to best static variant
- Only modify parameters, preserve internal state



[6]

Realizing Dynamic Algorithm Selection

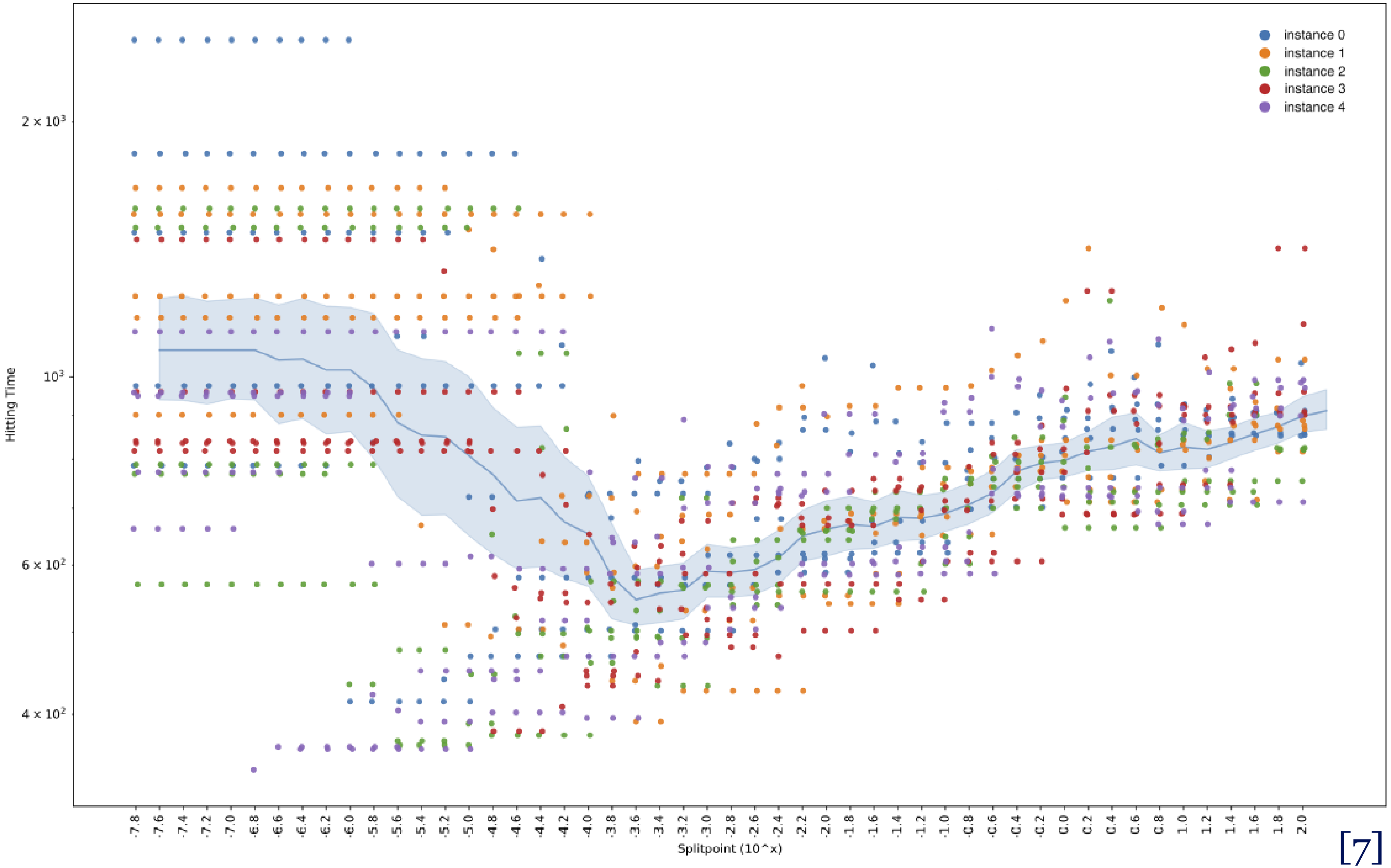
- Switching from one CMA-ES to another is ‘doable’
- But, the principle can be used between any algorithms
- Requires some warm-starting procedure
- But, data shows that there is a lot of potential in this approach!
- Create small portfolio of algorithms to test



[7]

Towards Landscape-Aware Switching

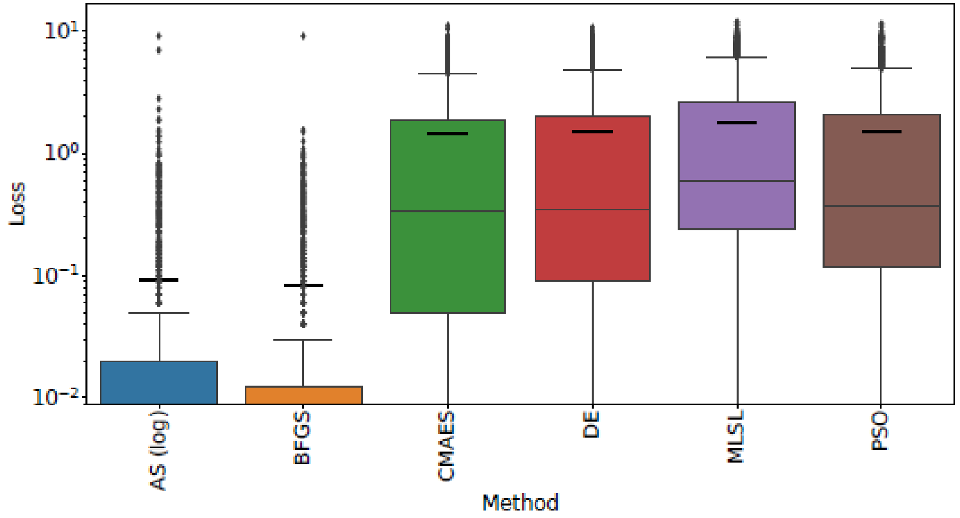
- Benefit from switching seems to change per run
- If detected, can see more benefits
- ELA features could be beneficial



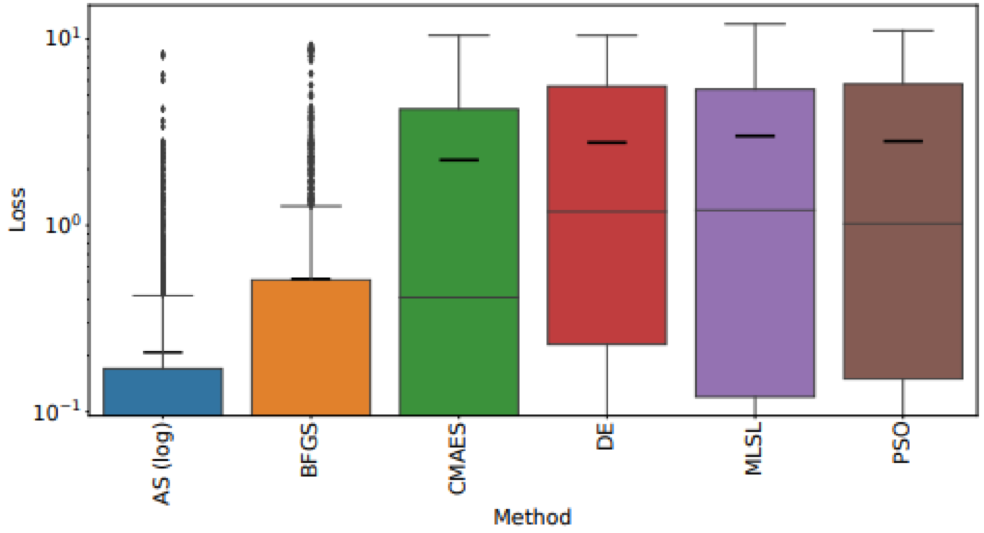
[7]

Per-run Dynamic Algorithm Selection

- Start with CMA-ES, switch to another algorithm in portfolio
- Based on ELA features collected in 150 evaluations
- Basic warm-starting mechanisms



(a) Budget 100

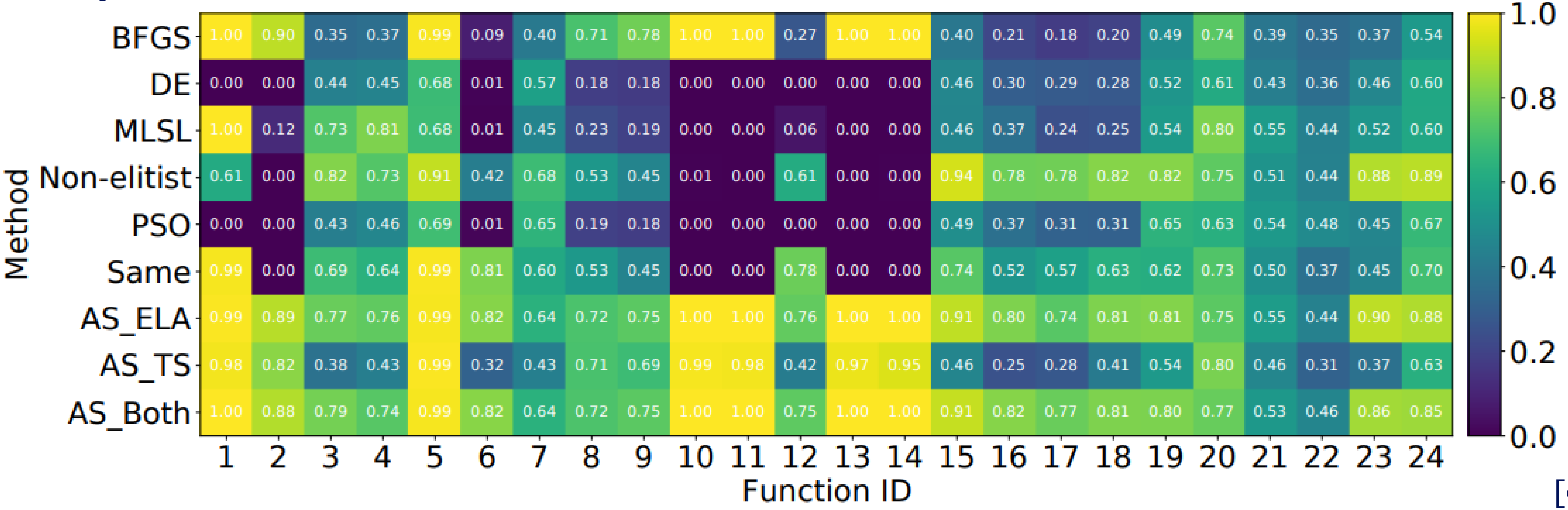


(b) Budget 900

[8]

Per-run Dynamic Algorithm Selection

- Extended portfolio to 6 algorithms, slightly improved warm-starting
- Consider performance ratios (relative difference in fixed-budget performance between method and the best choice per run)
- 3 Algorithm Selectors: ELA-based, TS-based (internal CMA-ES state features) and combined



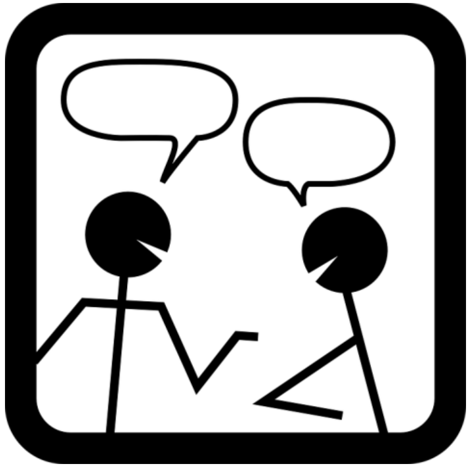
[9]

Challenges

- Need to investigate better warm-starting mechanisms
- Find a good way to identify promising dynamic algorithms
- Deal with the inherent stochasticity of algorithms
- Choose the point at which to perform the switch dynamically

Summary

- Benchmarking is more than just a number
- Data collected gives insights to build new ideas
- Dynamic algorithm selection has potential
- But, many open challenges



d.l.vermetten@liacs.leidenuniv.nl

References

1. Hao Wang, Diederick Vermetten, Furong Ye, Carola Doerr, Thomas Bäck:
   IOHanalyzer: Detailed Performance Analyses for Iterative Optimization Heuristics. ACM Trans. Evol. Learn. Optim. 2(1): 3:1-3:29 (2022)

2. Diederick Vermetten, Hao Wang, Thomas Bäck, Carola Doerr:
   Towards dynamic algorithm selection for numerical black-box optimization: investigating BBOB as a use case. GECCO 2020: 654-662

3. Sander van Rijn, Carola Doerr, Thomas Bäck:
   Towards an Adaptive CMA-ES Configurator. PPSN (1) 2018: 54-65

4. Sander van Rijn, Hao Wang, Matthijs van Leeuwen, Thomas Bäck:
   Evolving the structure of Evolution Strategies. SSCI 2016: 1-8

5. Jacob de Nobel, Diederick Vermetten, Hao Wang, Carola Doerr, Thomas Bäck:
   Tuning as a means of assessing the benefits of new ideas in interplay with existing algorithmic modules. GECCO Companion 2021: 1375-1384

6. Diederick Vermetten, Sander van Rijn, Thomas Bäck, Carola Doerr:
   Online selection of CMA-ES variants. GECCO 2019: 951-959

7. Dominik Schröder, Diederick Vermetten, Hao Wang, Carola Doerr, Thomas Bäck:
   Chaining of Numerical Black-box Algorithms: Warm-Starting and Switching Points. CoRR abs/2204.06539 (2022)

8. Anja Jankovic, Diederick Vermetten, Ana Kostovska, Jacob de Nobel, Tome Eftimov, Carola Doerr:
   Trajectory-based Algorithm Selection with Warm-starting. CoRR abs/2204.06397 (2022) (to appear at CEC 2022)

9. Ana Kostovska, Anja Jankovic, Diederick Vermetten, Jacob de Nobel, Hao Wang, Tome Eftimov, Carola Doerr:
   Per-run Algorithm Selection with Warm-starting using Trajectory-based Features. CoRR abs/2204.09483 (2022) (to appear at PPSN 2022)