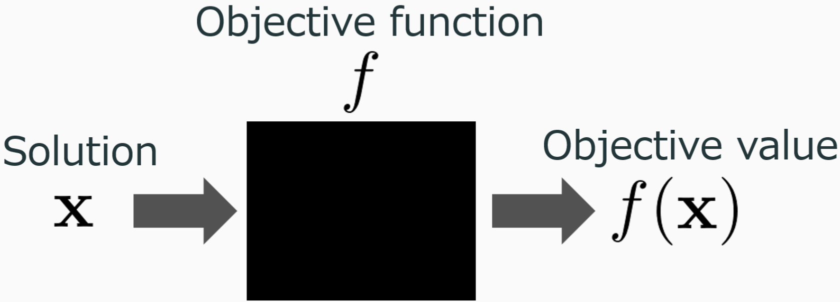
# On Constructing Algorithm Portfolios in Algorithm Selection for Computationally Expensive Black-box Optimization in the Fixed-budget Setting

IAM Workshop 2024 at Melbourne

Takushi Yoshikawa and **Ryoji Tanabe**

Yokohama National University
Yokohama city, Japan

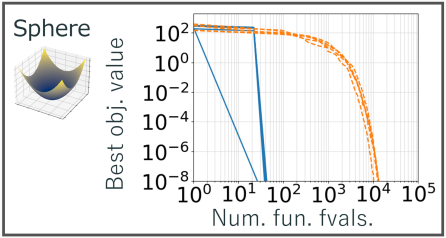
# Black-box continuous optimization



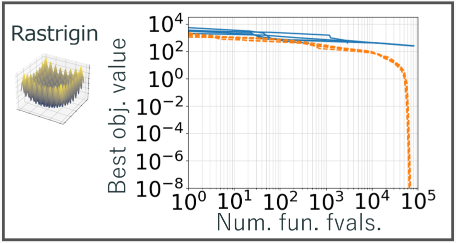
# Algorithm selection problem for BBO

Best optimizer depends on the property of a problem

**BFGS** reaches the optimal solution about 243 times faster than **DE**




Unlike **DE**, **BFGS** cannot reach the optimal solution




- A user needs to select the most promising optimizer
  - Hand-selecting requires tedious trial-and-error

# Feature-based offline algorithm selection for BBO

**Training phase** (on a training problem set)

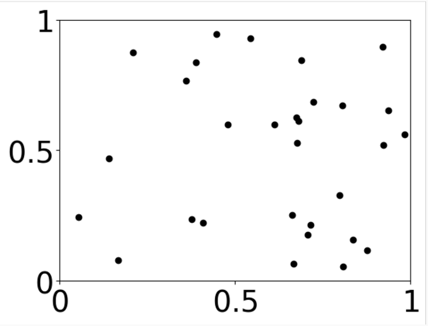
1. Generate a solution set $\mathcal{X}$ and calculate $f(\mathcal{X})$
2. Compute features based on the pair of $\mathcal{X}$ and $f(\mathcal{X})$
3. Train $k$ ML models for $k$ optimizers in a portfolio $\mathcal{A}$
    - $\mathcal{A}$: a set of $k$ candidate optimizers ($k = 4$ in this work)

**Testing phase** (on a target problem)

1. Generate a solution set $\mathcal{X}$ and calculate $f(\mathcal{X})$
2. Compute features based on the pair of $\mathcal{X}$ and $f(\mathcal{X})$
3. Predict the performance of $k$ optimizers by the $k$ ML models, then select the best one

# 1. Generate a solution set $\mathcal{X}$ and calculate $f(\mathcal{X})$

## Generate solutions

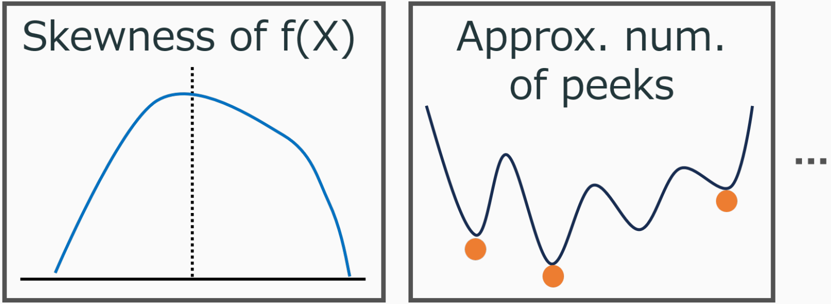



## Calculate objective values

| | f(x) | x1 | x2 |
|---|---|---|---|
| 1 | 37.9 | 0.78 | 0.133 |
| 2 | 70.6 | 0.949 | 0.646 |
| 3 | -34.1 | 0.858 | 0.15 |
| 4 | 56.8 | 0.889 | 0.827 |
| 5 | -78.2 | 0.275 | 0.159 |
| 6 | -35.3 | 0.594 | 0.529 |
| 7 | -12.5 | 0.261 | 0.698 |

# 2. Compute features based on $\mathcal{X}$ and $f(\mathcal{X})$

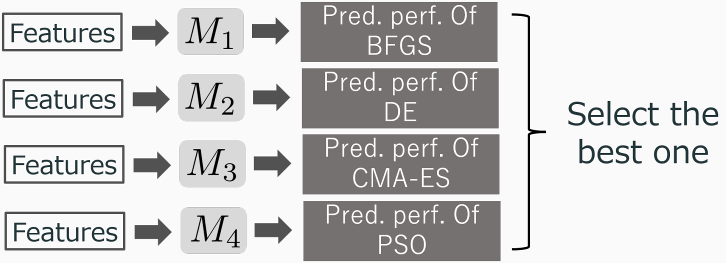
- Exploratory Landscape Analysis (ELA) [Mersmann 11]
  - Input: the pair of $\mathcal{X}$ and $f(\mathcal{X})$
  - Output: a set of numerical features of a problem




Olaf Mersmann, Bernd Bischl, Heike Trautmann, Mike Preuss, Claus Weihs, Günter Rudolph: Exploratory landscape analysis. GECCO 2011: 829-836

# 3. Predict the performance of $k$ optimizers by the $k$ ML models, then select the best one

E.g., a portfolio $\mathcal{A}$ = { BFGS, DE, CMA-ES, PSO }




- The system selects a promising optimizer without:
  - any user interaction
  - actually running the $k$ optimizers on a real-world problem

# Contribution: Suggestion for constructing portfolios

- We focus on computationally expensive optimization
    - Some real-world problems require a long computation time to evaluate a solution $\mathbf{x}$ by expensive computer simulations
    - The max. number of fevals. (`MaxFE`) should be small
- Algorithm selection for compu. expensive opt.
    - has not been studied well
    - A few previous studies did it, but the setup was incorrect
- We point out two issues in existing approaches

# Issue1: Most studies did not correctly count fevals

- `XFE`: fevals used for generating the solution set $\mathcal{X}$
    - A selection system requires `XFE` to calculate $f(\mathcal{X})$
- `OFE`: fevals used for a selected optimizer
    - E.g., the max. fevals of CMA-ES is `OFE`
- `MaxFE`: the max. fevals for the whole system
    - Correct: `MaxFE` = `XFE` + `OFE`
- However, most previous studies ignored `XFE`
    - Incorrect: `MaxFE` = ~~`XFE`~~ + `OFE`
- This leads to an overestimation of the performance
    - Correct: the selected optimizer can use only `OFE`
        - `XFE` should be small because a large `XFE` leads a small `OFE`
    - Incorrect: the selected optimizer can use `MaxFE`
        - `XFE` can be large as possible (`XFE` >> `MaxFE` in some work)

# Our approach for the issue1

$$\texttt{MaxFE} = \texttt{XFE} + \texttt{OFE}$$

# Issue2: Previous studies considered `MaxFE` instead of `OFE` when constructing algorithm portfolios

- `OFE`: fevals used for a selected optimizer
- `MaxFE`: max. fevals for the whole system
- How to construct a portfolio in previous studies
  - Run many optimizers on training problems until `MaxFE`
  - Select $k$ optimizers based on their performance at `MaxFE`
- But, an optimizer can use only `OFE`, not `MaxFE`
  - Suppose: `MaxFE`$= 1\,000$ and `OFE`$= 500$
  - The portfolio consists of good optimizers at $1\,000$ fevals
  - But, they are unlikely to perform well at $500$ fevals
- This gap can make the effectiveness of portfolio poor

# Our approach for the issue2

Construct algorithm portfolios based on the performance of optimizers at `OFE`, not at `MaxFE`

# Experimental setup

- The COCO platform [Hansen 21]
  - The 24 `bbob` functions with $n \in \{2, 3, 5, 10\}$
  - Portfolios were constructed based on the benchmarking data of 244 optimizers in the COCO archive
  - Local search method for subset selection was used
- Settings for algorithm selection systems
  - `flacco` [Kerschke 19] was used for feature computation
  - `MaxFE` was set to $100 \times n$ ($n$: dimension)
    - The first study to set `MaxFE` below $100 \times n$ *actually*
  - Random forest regressor was used

Nikolaus Hansen, Anne Auger, Raymond Ros, Olaf Mersmann, Tea Tušar, Dimo Brockhoff: COCO: a platform for comparing continuous optimizers in a black-box setting. Optim. Methods Softw. 36(1): 114-144 (2021)

Pascal Kerschke and Heike Trautmann. 2019. Comprehensive Feature-Based Landscape Analysis of Continuous and Constrained Optimization Problems Using the R-package flacco. In Applications in Statistical Computing – From Music Data Analysis to Industrial Quality Improvement. Springer, 93-123.

# 6 portfolios constructed in this work ($k = 4$)

- `OFE`: fevals used for a selected optimizer
- `XFE`: fevals used for generating the solution set $\mathcal{X}$
- `MaxFE`: max. fevals for the whole system
  - `MaxFE` = `XFE` + `OFE` = $100n$ ($n$: dimension)
- $\mathcal{A}_{\mathrm{max}}$: based on the perf. at `MaxFE` $= 100n$
  - Traditional, incorrect construction approach
- $\mathcal{A}_{90}$: based on the perf. at `OFE`$= 90n$
  - `XFE` = $100n$ − $90n$ = $10n$
- $\mathcal{A}_{85}$: based on the perf. at `OFE`$= 85n$
  - `XFE` = $100n$ − $85n$ = $15n$
- $\mathcal{A}_{80}$, $\mathcal{A}_{75}$, $\mathcal{A}_{50}$ were constructed in the same way

# 6 portfolios constructed in this work ($k = 4$)

Surrogate-CMA-ES | STEP-based | mathematical | DIRECT-based

| | |
|---|---|
| $\mathcal{A}_{\max}$ | lq-CMA-ES, BIPOP-aCMA-STEP, MLSL, oMads-2N |
| $\mathcal{A}_{90}$ | DTS-CMA-ES_005, BrentSTEPif, DIRECT-REV, CMA-ES-2019 |
| $\mathcal{A}_{85}$ | lq-CMA-ES, lmm-CMA-ES, STEPifeg, fmincon |
| $\mathcal{A}_{80}$ | lq-CMA-ES, lmm-CMA-ES, STEPifeg, fmincon |
| $\mathcal{A}_{75}$ | lq-CMA-ES, lmm-CMA-ES, BrentSTEPif, DIRECT-REV |
| $\mathcal{A}_{50}$ | lq-CMA-ES, lmm-CMA-ES, BrentSTEPrr, oMads-2N |

- Left surrogate-CMA-ES is the single-best solver (SBS)
  - The best optimizer in terms of the average performance
  - Only in $\mathcal{A}_{90}$, the SBS is DTS-CMA-ES_005

# Average rankings of 10 algorithm selection systems

- **The system with $\mathcal{A}_{90}$ performs the best**
  - Importance of using the performance at OFE (not MaxFE)
  - A large OFE allows a long run of an optimizer

| Portfolio | OFE | XFE | MaxFE | $n = 2$ | $n = 3$ | $n = 5$ | $n = 10$ |
|---|---|---|---|---|---|---|---|
| $\mathcal{A}_{\max}$ | $90n$ | $10n$ | $100n$ | 5.50 | 5.65 | 5.73 | 5.58 |
| $\mathcal{A}_{\max}$ | $85n$ | $15n$ | $100n$ | 5.54 | 5.48 | 6.02 | 6.06 |
| $\mathcal{A}_{\max}$ | $80n$ | $20n$ | $100n$ | 6.21 | 5.10 | 5.35 | 6.15 |
| $\mathcal{A}_{\max}$ | $75n$ | $25n$ | $100n$ | 5.50 | 4.85 | 4.60 | 5.94 |
| $\mathcal{A}_{\max}$ | $50n$ | $50n$ | $100n$ | 5.00 | 5.85 | 6.00 | 5.98 |
| $\mathcal{A}_{90}$ | $90n$ | $10n$ | $100n$ | 4.44 | 4.15 | 4.12 | 3.04 |
| $\mathcal{A}_{85}$ | $85n$ | $15n$ | $100n$ | 4.38 | 4.52 | 5.37 | 4.04 |
| $\mathcal{A}_{80}$ | $80n$ | $20n$ | $100n$ | 5.29 | 5.65 | 5.79 | 4.67 |
| $\mathcal{A}_{75}$ | $75n$ | $25n$ | $100n$ | 5.21 | 5.81 | 4.73 | 5.29 |
| $\mathcal{A}_{50}$ | $50n$ | $50n$ | $100n$ | 7.94 | 7.94 | 7.27 | 8.25 |

# Conclusion: This work focused on portfolios in algorithm selection for comput. expensive BBO

- `XFE`: fevals used for generating the solution set $\mathcal{X}$
- `OFE`: fevals used for a selected optimizer
- `MaxFE`: the max. fevals for the whole system

- **Two take home messages**:
  - `MaxFE` should be `XFE` + `OFE`
  - Portfolios should be constructed based on the performance of optimizers for `OFE`, not `MaxFE`
- **Future work: Improving selection systems**
  - The performance of the present system is not good