Analyzing Adaptive Parameter Landscapes in Parameter Adaptation Methods for Differential Evolution

Ryoji Tanabe

Yokohama National University
Yokohama, Japan

Differential Evolution (DE) [Storn 97]: A simple black-box optimizer

DE is sensitive to the setting of two parameters: F and C

- Scale factor F controls the magnitude of the differential mutation
- Crossover rate C controls the number of inherited variables from $\boldsymbol{x}$

Adaptive DE algorithms

- E.g., jDE [Brest 06], JADE [Zhang 09], SHADE [Tanabe 13]
- adaptively adjust F and C values

Poor understanding of parameter adaptation mechanisms in DE

- Its working principle is unclear
- Only a few previous studies tried to analyze adaptive DEs
    - E.g., [Zielinski 08, Drozdik 15, Tanabe 16, Tanabe 17, Tanabe 20]

Difficulty comes from the unclarity of the adaptive parameter space

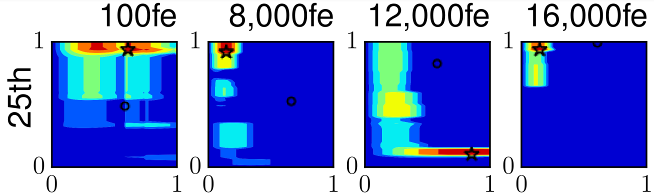
- Is it possible to make the adaptive parameter space "visualizable"?

Contributions

This work

1. proposes a concept called adaptive parameter landscapes
2. proposes a method of analyzing adaptive parameter landscapes
3. provides insightful knowledge on parameter adaptation in DE

Visualization of dynamically changing parameter landscapes



Fitness landscape analysis

Fitness landscape [Pitzer 12]

$$\mathcal{L}_{\text{fitness}} = (\mathbb{X}, f, D)$$

- $\mathbb{X}$: solution space, f: objective function, D: distance function

No explanation needed in GECCO

Parameter landscape analysis

Parameter landscape [Harrison 19]

$$\mathcal{L}_{\text{parameter}} = (\Theta, M, D),$$

- Θ: parameter space, M: performance metric, D: distance function
- $\mathcal{L}_{\text{parameter}}$ is an $\mathcal{L}_{\text{fitness}}$ of a parameter tuning problem

Example: parameter tuning of $F \in [0, 1]$ and $C \in [0, 1]$ in DE

- to find the best pair $(F, C) \in \Theta$ on a training problem set $\boldsymbol{I}$
- $\Theta = [0, 1] \times [0, 1]$, M: ERT on $\boldsymbol{I}$, D: Euclidean distance

Parameter landscapes coined in [Yuan 12] are also known as

- performance landscapes [Yuan 07], meta-fitness landscapes [Pedersen 10], utility landscapes [Eiben 11], ERT landscapes [Belkhir 16], parameter configuration landscapes [Harrison 19], and algorithm configuration landscapes [Pushak 18]
- No consistency in the terminology in the EC community

Proposed concept: adaptive parameter landscapes

Parameter landscape [Harrison 19]

$$\mathcal{L}_{\text{parameter}} = (\Theta, M, D)$$

Adaptive parameter landscape

$$\mathcal{L}_{\text{adaptive}} = (\Theta_i^t, M, D)$$

- Θ_i^t: dynamic parameter space of $\boldsymbol{x}_i^t$, M: performance metric, D: distance function
- $\mathcal{L}_{\text{adaptive}}$ is an $\mathcal{L}_{\text{parameter}}$ of the i-th individual at iteration t ($\boldsymbol{x}_i^t$)
- While $\mathcal{L}_{\text{parameter}}$ is static, $\mathcal{L}_{\text{adaptive}}$ is dynamic

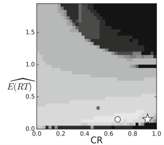
Example: parameter adaptation of $F \in [0,1]$ and $C \in [0,1]$ in DE

- to find the best pair $(F_i^t, C_i^t) \in \Theta_i^t$ for $\boldsymbol{x}_i^t$
- $\Theta_i^t = [0,1] \times [0,1]$, M: G1 (explained later), D: Euclidean distance

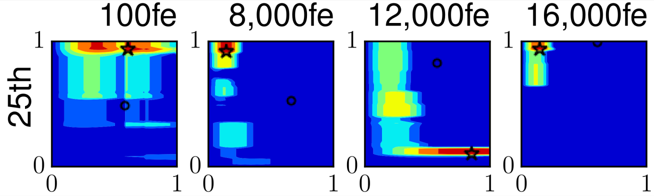
Parameter landscapes vs. adaptive parameter landscapes (F and C in DE)

Parameter landscape $\mathcal{L}_{\text{parameter}} = (\Theta, M, D)$ is STATIC

Parameter landscape of DE [Belkhir 16]

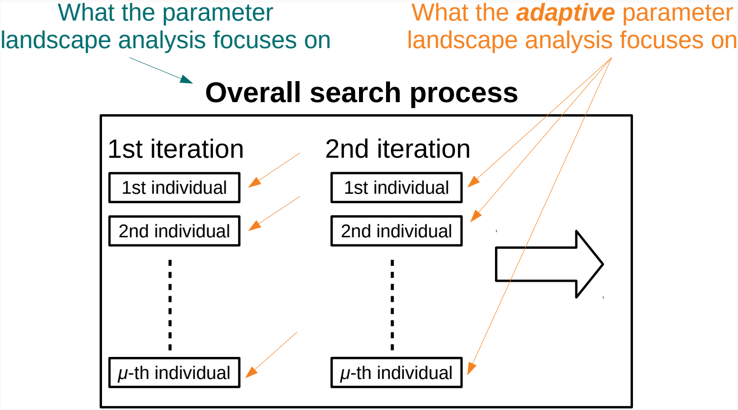


Adaptive parameter landscape $\mathcal{L}_{\text{adaptive}} = (\Theta_i^t, M, D)$ is DYNAMIC



N. Belkhir, J. Dréo, P. Savéant, M. Schoenauer: Feature Based Algorithm Configuration: A Case Study with Differential Evolution. PPSN 2016: 156-166

Parameter landscapes vs. adaptive parameter landscapes (continued)



NOT proposed: 1-step-lookahead greedy improvement metric (G1)

Nomenclature

- $\boldsymbol{x}_i^t$: the i-th individual in the population at iteration t
- $\boldsymbol{u}_i^t$: the i-th child generated by $\boldsymbol{x}_i^t$ at iteration t
- F_i^t: the scale factor value used for generating $\boldsymbol{u}_i^t$
- C_i^t: the crossover rate value used for generating $\boldsymbol{u}_i^t$

G1 measures how much F_i^t and C_i^t improve the fitness value of x_i^t

$$\mathrm{G1}(F_i^t, C_i^t) = \begin{cases} |f(\boldsymbol{x}_i^t) - f(\boldsymbol{u}_i^t)| & \text{if } f(\boldsymbol{u}_i^t) < f(\boldsymbol{x}_i^t) \\ 0 & \text{otherwise} \end{cases}$$

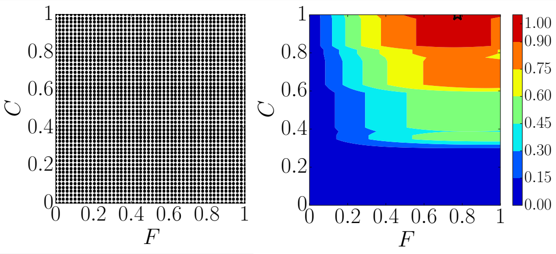
- If $f(\boldsymbol{x}_i^t) = 10$ and $f(\boldsymbol{u}_i^t) = 3$, $\mathrm{G1}(F_i^t, C_i^t) = 7$
- If $f(\boldsymbol{x}_i^t) = 10$ and $f(\boldsymbol{u}_i^t) = 30$, $\mathrm{G1}(F_i^t, C_i^t) = 0$
- Just for the sake of simplicity, we use the term "G1"
- G1 can be replaced with G2, G3, and G1 + novelty

Proposed method for analyzing adaptive parameter landscapes

For the i-th individual at iteration t (x_i^t)

1. Generate $50 \times 50 = 2\,500$ pairs of F and C in a grid manner
2. Generate $2\,500$ children by using the $2\,500$ pairs of F and C
    - Same random numbers are used for generating the $2\,500$ children
3. Evaluate the objective function values of the $2\,500$ children
    - Extra $2\,500$ function evaluations are *not counted*
4. Calculate the $\mathrm{G1}$ values of the $2\,500$ pairs

(a) 50×50 pairs **(b)** Contour map of $\mathcal{L}_{\mathrm{adaptive}}$



Properties of the proposed method

Proposed method is totally independent from the procedure of DE

- Proposed method is just a logger, not an optimizer
- $2\,500$ children are used only for the analysis, not for the search
- Behavior of DE with/without the proposed method is the same

Suppression strategy of the randomness in DE

- 1 child for the actual search and $2\,500$ children for the G1 calculation are generated using the same parents and crossover mask
- Stochastic nature of DE can be suppressed

Cheat in not counting the $2\,500$ extra function evaluations

- This cheat is no problem at all for the analysis
- We are not interested in solving any real-world problem

Experimental setup

Settings for adaptive DEs

- jDE [Brest 06], JADE [Zhang 09], SHADE [Tanabe 13]
- Default hyperparameter settings
- Population size $\mu = 100$, no restart
- Current-to-$p$best/1 [Zhang 09], binomial crossover [Storn 97]

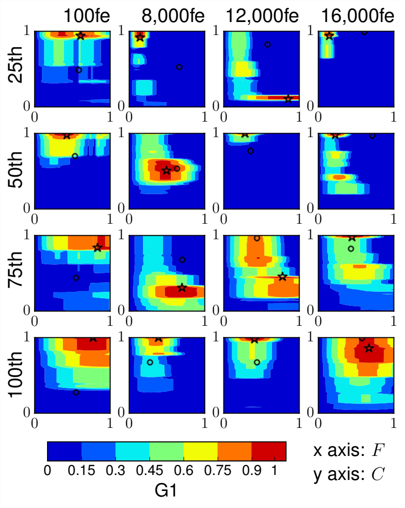
Settings for test functions

- 24 BBOB noiseless functions [Hansen 09] in COCO [Hansen 16]
- Dimensionality $n \in \{2, 3, 5, 10, 20, 40\}$
- Maximum number of evaluations $= 10\,000 \times n$
- Number of runs $= 15$ (results of a single run are shown)

Source code is available:

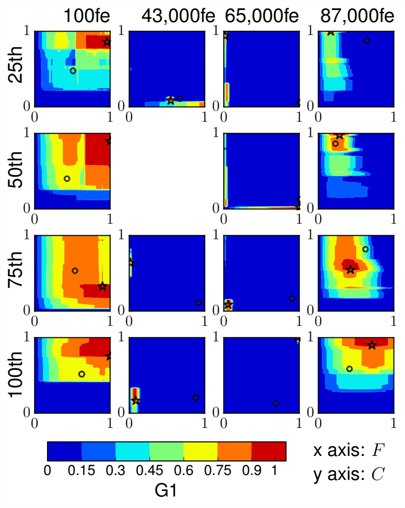
- `https://github.com/ryojitanabe/APL`

Contour maps of adaptive parameter landscapes in SHADE on the 20-dimensional Sphere function (f_1)



- 100 individuals were sorted
- Pairs for the best (1st) individual are seldom successful, so omitted it
- SHADE found $\boldsymbol{x}^*$ at $\approx 16\,000$ fe
- $\circ$: the pair generated by SHADE
- $\star$: the best pair regarding G1
- Shape of $\mathcal{L}_{\mathrm{adaptive}}$ is different depending on:
  - the rank of each individual
  - the search progress
- $\circ$ and $\star$ are far from each other
- $\mathcal{L}_{\mathrm{adaptive}}$ is unimodal/multimodal?

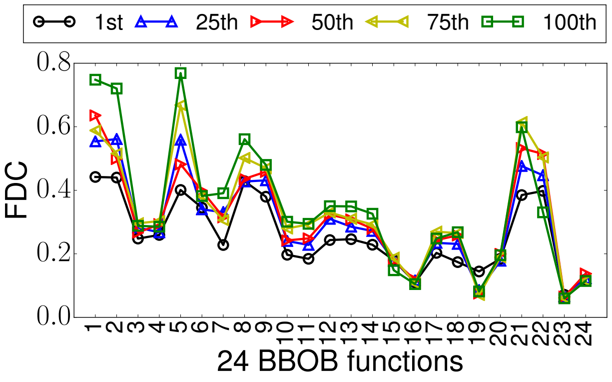
Contour maps of adaptive parameter landscapes in SHADE on the 20-dimensional Rastrigin function (f_3)



- Generating a successful pair on f_3 is more difficult than that on f_1
- $\mathcal{L}_{\mathrm{adaptive}}$ at 100 fe looks easy
  - Area with G1 > 0 is large
- $\mathcal{L}_{\mathrm{adaptive}}$ at $43\,000$ fe looks hard
  - Area with G1 > 0 is very small like needle-in-haystack land.
  - When all $2\,500$ pairs obtain $\mathrm{G1} = 0$, $\mathcal{L}_{\mathrm{adaptive}}$ is not shown
- $\mathcal{L}_{\mathrm{adaptive}}$ at $87\,000$ fe looks easy
  - Area with G1 > 0 values becomes large again
  - SHADE found $\boldsymbol{x}^*$ at $\approx 87\,000$ fe
  - Population has converged to $\boldsymbol{x}^*$
  - Generating better children is easy

Average FDC values of adaptive parameter landscapes in SHADE ($n = 20$)

- Results of FDC [Jones 95] and Dispersion [Lunacek 06] are similar
- FDC value is different depending on the function
- $\mathcal{L}_{\text{adaptive}}$ of individuals with similar ranks have similar FDC values
  - E.g., FDC values of the 75th and 100th individuals are similar
- Global structures of $\mathcal{L}_{\text{adaptive}}$ can correlate with the rank of indiv.



Conclusion

This work

- proposed the concept called adaptive parameter landscapes $\mathcal{L}_{\mathrm{adaptive}}$
- proposed the method of analyzing adaptive parameter landscapes
- provided insightful knowledge on parameter adaptation in DE

Our observations

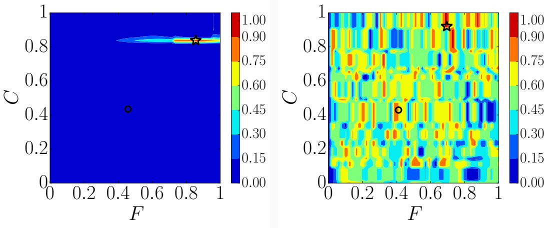
- $\mathcal{L}_{\mathrm{adaptive}}$ is different depending on the search progress
- $\mathcal{L}_{\mathrm{adaptive}}$ is influenced by the properties of a problem
- Global structures of $\mathcal{L}_{\mathrm{adaptive}}$ can correlate with the rank of indiv.
- ADEs generally generate a pair of F and C far from the best pair

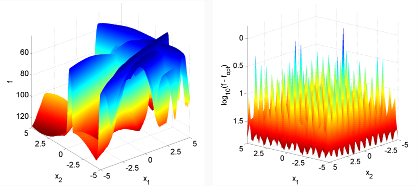
Future work

- analyze other adaptive evolutionary algorithms, e.g., GA and ES
- use other performance metric, e.g., G2 and G1+novelty

Contour maps of adaptive parameter landscapes in SHADE on the 20-dimensional Gallagher and Katsuura functions (f_{22} and f_{23})

- Results of the 100-th individual at 100 evaluations
- Shape of adaptive parameter landscapes is significantly influenced by the global structures of fitness landscapes

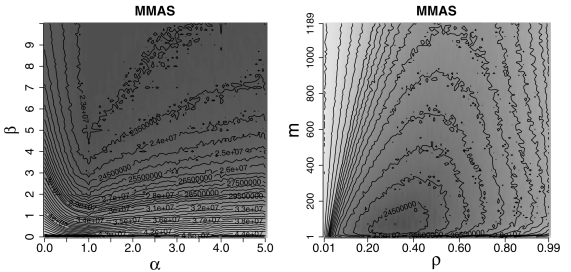




[Hansen 09]

Parameter landscape analysis

Parameter landscapes of $\mathcal{MM}$AS [Yuan 12]



Motivation

- A better understanding of $\mathcal{L}_{\text{parameter}}$ can lead to a better understanding of the corresponding optimizer
  - making the optimizer more efficient
- Knowledge on $\mathcal{L}_{\text{parameter}}$ are useful for designing a parameter tuner

Z. Yuan and M. A. M. de Oca, M. Birattari, T. Stützle: Continuous optimization algorithms for tuning real and integer parameters of swarm intelligence algorithms. Swarm Intelligence 6(1): 49-75 (2012)

Basic DE with almost any parameter adaptation method

input: $\mathbb{X} \subseteq \mathbb{R}^n$, $f : \mathbb{R}^n \to \mathbb{R}$, population size μ, some hyperparameters

$t \leftarrow 1$, initialize $\boldsymbol{P} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_\mu\}$ randomly;

Initialize internal parameters for adaptation of F and C;

while The termination criteria are not met **do**

- **for** $i \in \{1, ..., \mu\}$ **do**
  - Generate F_i and C_i;
  - Randomly select r_1, r_2, r_3 from $\{1, ..., \mu\} \setminus \{i\}$ s.t. $r_1 \neq r_2 \neq r_3$;
  - Mutant vector $\boldsymbol{v}_i \leftarrow \boldsymbol{x}_{r_1} + F_i \; (\boldsymbol{x}_{r_2} - \boldsymbol{x}_{r_3})$;
  - Child $\boldsymbol{u}_i = (u_{i,1}, ..., u_{i,n})^\top$, randomly select j_{rand} form $\{1, ..., n\}$;
  - **for** $j \in \{1, ..., n\}$ **do**
    - **if** $\text{rand}[0, 1] \leq C_i$ *or* $j = j_{\text{rand}}$ **then** $u_{i,j} \leftarrow v_{i,j}$;
    - **else** $u_{i,j} \leftarrow x_{i,j}$;
- **for** $i \in \{1, ..., \mu\}$ **do**
  - **if** $f(\boldsymbol{u}_i) \leq f(\boldsymbol{x}_i)$ **then** $\boldsymbol{x}_i \leftarrow \boldsymbol{u}_i$;
- Update internal parameters for adaptation of F and C;
- $t \leftarrow t + 1$;

Basic DE [Storn 97]

input: $\mathbb{X} \subseteq \mathbb{R}^n$, $f : \mathbb{R}^n \to \mathbb{R}$, population size μ, scale factor F, crossover rate C

$t \leftarrow 1$, initialize $\boldsymbol{P} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_\mu\}$ randomly;

while The termination criteria are not met **do**

- **for** $i \in \{1, ..., \mu\}$ **do**
  - Randomly select r_1, r_2, r_3 from $\{1, ..., \mu\} \setminus \{i\}$ s.t. $r_1 \neq r_2 \neq r_3$;
  - Mutant vector $\boldsymbol{v}_i \leftarrow \boldsymbol{x}_{r_1} + F\,(\boldsymbol{x}_{r_2} - \boldsymbol{x}_{r_3})$;
  - Child $\boldsymbol{u}_i = (u_{i,1}, ..., u_{i,n})^\top$, randomly select j_{rand} form $\{1, ..., n\}$;
  - **for** $j \in \{1, ..., n\}$ **do**
    - **if** $\text{rand}[0, 1] \leq C$ *or* $j = j_{\text{rand}}$ **then** $u_{i,j} \leftarrow v_{i,j}$;
    - **else** $u_{i,j} \leftarrow x_{i,j}$;
- **for** $i \in \{1, ..., \mu\}$ **do**
  - **if** $f(\boldsymbol{u}_i) \leq f(\boldsymbol{x}_i)$ **then** $\boldsymbol{x}_i \leftarrow \boldsymbol{u}_i$;
- $t \leftarrow t + 1$;

Basic DE with the parameter adaptation method in JADE [Zhang 09]

input: $\mathbb{X} \subseteq \mathbb{R}^n$, $f : \mathbb{R}^n \to \mathbb{R}$, population size μ, adaptation rate $\alpha = 0.1$

$t \leftarrow 1$, initialize $\boldsymbol{P} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_\mu\}$ randomly;

Initialize internal parameters $m_F \leftarrow 0.5$ and $m_C \leftarrow 0.5$;

while The termination criteria are not met **do**

- **for** $i \in \{1, ..., \mu\}$ **do**
  - $F_i \sim \mathrm{CauchyDist}(m_F, 0.1)$ and $C_i \sim \mathrm{NormalDist}(m_C, 0.1)$;
  - Randomly select r_1, r_2, r_3 from $\{1, ..., \mu\} \setminus \{i\}$ s.t. $r_1 \neq r_2 \neq r_3$;
  - Mutant vector $\boldsymbol{v}_i \leftarrow \boldsymbol{x}_{r_1} + F_i \, (\boldsymbol{x}_{r_2} - \boldsymbol{x}_{r_3})$;
  - Child $\boldsymbol{u}_i = (u_{i,1}, ..., u_{i,n})^\top$, randomly select j_{rand} form $\{1, ..., n\}$;
  - **for** $j \in \{1, ..., n\}$ **do**
    - **if** $\mathrm{rand}[0, 1] \leq C_i$ *or* $j = j_{\mathrm{rand}}$ **then** $u_{i,j} \leftarrow v_{i,j}$;
    - **else** $u_{i,j} \leftarrow x_{i,j}$;
- **for** $i \in \{1, ..., \mu\}$ **do**
  - **if** $f(\boldsymbol{u}_i) \leq f(\boldsymbol{x}_i)$ **then** $\boldsymbol{x}_i \leftarrow \boldsymbol{u}_i$;
- $m_F \leftarrow (1 - \alpha) m_F + \alpha \, \mathrm{Lmean}(\boldsymbol{S}_F)$ and $m_C \leftarrow (1 - \alpha) m_C + \alpha \, \mathrm{mean}(\boldsymbol{S}_C)$;
- $t \leftarrow t + 1$;

Behavior of the internal parameters m_F and m_C in JADE

(c) Rastrigin $(n = 10)$ **(d)** Rosenbrock $(n = 10)$