HS-ES-DE: HS-ES Followed by L-SHADE-EpSin for Real Parameter Single Objective Optimization

Authors: Jianjun Jiang, Yong Wan, Chengjun Li, Chuchuan Cen, Yikai Shao, Qinxue Meng, Jianjun Jiang

Date: 2022-12-07T00:00:00+00:00

Abstract

For real parameter single objective optimization, Differential Evolution (DE) and Covariance Matrix Adaptation Evolution Strategy (CMA-ES) both perform powerfully. Nevertheless, in the field of real parameter single objective optimization, it is impossible for a given algorithm to perform well in all fitness landscapes. Practice has proved that ensemble of different algorithms may lead to improvement in solution. In this paper, based on two famous population-based metaheuristics - LSHADE-EpSin and HS-ES, we propose ensemble with successively executed constituent algorithms - HS-ES-DE. In our algorithm, HS-ES is replaced by L-SHADE-EpSin after stagnation is detected. Beside our HS-ES-DE, 12 population-based metaheuristics are involved in our experiments in which three benchmark test suites are employed. Experimental results show that our algorithm is very competitive.

Full Text

Preamble

HS-ES-DE: HS-ES Followed by L-SHADE-EpSin for Real Parameter Single Objective Optimization

Jianjun Jianga, Yong Wanga,b, Chengjun Lia,b,∗, Chuchuan Cena, Yikai Shaoa, Qinxue Mengc

aSchool of Computer Science, China University of Geosciences, Wuhan 430074, P.R. China
bHubei Key Laboratory of Intelligent Geo-Information Processing, China University of Geosciences, Wuhan 430074, P.R. China
cSchool of Software, University of Technology Sydney, Sydney 2000, Australia

Abstract

For real parameter single objective optimization, Differential Evolution (DE) and Covariance Matrix Adaptation Evolution Strategy (CMA-ES) both exhibit powerful performance. Nevertheless, in this domain, no single algorithm can perform well across all fitness landscapes. Empirical evidence demonstrates that ensemble methods combining different algorithms can lead to improved solutions. In this paper, we propose HS-ES-DE, an ensemble of two renowned population-based metaheuristics—L-SHADE-EpSin and HS-ES—with successively executed constituent algorithms. In our approach, HS-ES is replaced by L-SHADE-EpSin after stagnation is detected.

In our experiments, we evaluate HS-ES-DE alongside 12 population-based metaheuristics across three benchmark test suites. The results demonstrate that our algorithm is highly competitive.

Keywords: Real parameter single objective optimization, population-based metaheuristic, ensemble, stagnation

1. Introduction

Real parameter single objective optimization aims to find the optimal decision vector in solution space to minimize (or maximize) an objective function, representing a significant research focus in artificial intelligence. Numerous population-based metaheuristics have been proposed in this field. The Congress on Evolutionary Computation (CEC), a premier venue in evolutionary computation, has held at least six competitions annually on real parameter single objective optimization among population-based metaheuristics.

The CEC competitions are highly influential in this research domain and shape future research directions. We briefly introduce the winners of each competition as follows:

- **NBIPOP-aCMA-ES** [10], the 2013 winner, is a variant of Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [6] with a restart scheme comprising two regimes.
- **L-SHADE** [17], the 2014 winner, is a variant of Differential Evolution (DE) [15] extended from SHADE [16] with Linear Population Size Reduction (LPSR).
- **UMOEAs-II** [5], a joint winner in 2016, is an ensemble of DE and CMA-ES.
- **L-SHADE-EpSin** [2], the other joint winner in 2016, is a variant of L-SHADE with adaptive parameter settings and local search based on Gaussian walks.
- **EBOwithCMAR** [8], the 2017 winner, is a self-adaptive butterfly optimizer with success history-based adaptation, LPSR, and a covariance matrix adapted retreat phase.

- **HS-ES** [22], the 2018 winner, is controlled by CMA-ES with a restart scheme and an improved version of univariate sampling.
- **IMODE** [12], the 2020 winner, is a DE algorithm with three DE mutation strategies and sequential quadratic programming.

Among these seven winners, NBIPOP-aCMA-ES and HS-ES are based on CMA-ES, while L-SHADE, L-SHADE-EpSin, and IMODE are based on DE. UMOEAs-II incorporates both CMA-ES and DE. Further comparison in [14] reveals that UMOEA-II, L-SHADE-EpSin, and HS-ES are top performers among the six winners from the previous five competitions, indicating that the leading algorithms are based on DE and/or CMA-ES. Thus, the CEC competitions demonstrate that both CMA-ES and DE outperform other approaches in real parameter single objective optimization.

DE and CMA-ES were initially proposed in [15] and [7], respectively. In DE, operators such as mutation, crossover, and selection are applied to individuals (target vectors). The evolutionary framework produces trial vectors for the next generation through mutation first, then crossover, which are specified as trial vector generation strategies. During selection, either a target vector or its trial vector is chosen based on fitness comparison to become a target vector for the next generation.

In CMA-ES, the individual center is key to generating the next generation. Specifically, the individual center is initialized using the subset of individuals with better fitness. New individuals are then generated by sampling from a Gaussian distribution based on this center, after which the individual center for the next generation is determined.

The motivation for this paper stems from the observation that CMA-ES is prone to stagnation—a state where insufficient diversity among individuals prevents effective search. Consequently, both NBIPOP-aCMA-ES and HS-ES employ restart mechanisms to enhance CMA-ES. Furthermore, HS-ES applies an improved version of univariate sampling after the CMA-ES step for additional enhancement. However, compared to restart and univariate sampling, it may be more effective to have another powerful population-based metaheuristic take over the search after CMA-ES stagnation. Based on this idea, we can create an ensemble with successively executed constituent algorithms.

In this paper, we propose HS-ES-DE, an ensemble of HS-ES and L-SHADE-EpSin, both of which are CEC competition winners. Our algorithm begins with HS-ES, applying a scheme to detect stagnation. Stagnation is confirmed when the improvement ratio of average fitness from the previous generation to the current one falls below a threshold for a specified number of consecutive generations. Once stagnation is detected, HS-ES hands over the evolutionary process to L-SHADE-EpSin. For this transition, both population size and diversity are adjusted through our scheme. The initial population size for L-SHADE-EpSin ($NP_{DE_{max}}$) is calculated based on the remaining number of function evaluations before handover. Then, in the final generation of the CMA-ES step, $NP_{DE_{max}}$

individuals are produced. Subsequently, mutation and crossover from the original DE version [15] are executed for several generations, during which offspring produced by the trial vector generation strategy are directly selected without fitness evaluation to improve diversity.

We evaluate our proposed algorithm on the CEC 2014, 2017, and 2020 benchmark test suites for real parameter single objective optimization. In the first experiment using the CEC 2014 suite, we compare HS-ES-DE with L-SHADE-EpSin, HS-ES, and seven other population-based metaheuristics at dimensionalities of 30, 50, and 100. In the second experiment using the CEC 2017 suite, we compare HS-ES-DE with the same algorithms at dimensionality 30. In the third experiment using the CEC 2020 suite, we compare our algorithm with the top three algorithms from the 2020 competition at dimensionality 20. The experimental results demonstrate that our algorithm outperforms other population-based metaheuristics for real parameter single objective optimization.

The remainder of this paper is organized as follows: Section 2 reviews related work, Section 3 details our proposed algorithm, Section 4 presents and discusses experimental results, and Section 5 concludes the paper.

2. Related Work

This section reviews recent research on population-based metaheuristics for real parameter single objective optimization, focusing on ensemble approaches covering both ensemble DE algorithms and ensemble population-based metaheuristics.

Wu [20] proposed MPEDE, which simultaneously employs three mutation strategies. MPEDE's population consists of three equal-sized indicator subpopulations and one larger reward subpopulation, with each mutation strategy controlling one indicator subpopulation. The best-performing mutation strategy is determined every certain number of generations based on the ratio between fitness improvement and consumed function evaluations, and the reward subpopulation is then dynamically allocated to this strategy.

Mostafa et al. [1] presented sTDE-dR, which clusters the population into multiple tribes and utilizes an ensemble of different mutation and crossover strategies. Both lifecycle and participation ratio for the next generation are determined by a success-based scheme. Each tribe employs a different adaptive scheme for controlling scaling factor and crossover rate, and the mean success of each tribe is used to calculate the participation ratio for the next generation. Population size is dynamically reduced during execution.

Sallam et al. [13] proposed a DE algorithm that considers both fitness landscape characteristics and operator performance history to dynamically select the most suitable operator. Cui et al. [3] presented AMECoDEs, which employs two elites-guided trial vector generation strategies for each individual to produce two candidate solutions, with only the better one participating in selection. Wu et al. [21] proposed EDEV, which consists of three highly efficient DE variants—JADE [23], CoDE [19], and EPSDE. Similar to MPEDE, EDEV uses three indicator subpopulations and a reward subpopulation with a comparable competition mechanism.

Zhang et al. [24] suggested the MLCC framework, which implements a parallel structure where the entire population is simultaneously monitored by multiple DE algorithms assigned to different layers. Individuals can store, utilize, and update their evolution information across different layers, with MLCC-SI representing the state-of-the-art in MLCC-based DE algorithms.

Elsayed et al. [5] proposed UMOEAs-II, which employs a DE algorithm with multiple operator combinations and CMA-ES, where each constituent algorithm controls a subpopulation and the two subpopulations share information at intervals. Liu et al. [9] presented CETMS, based on tissue membrane systems and CMA-ES, where the tissue-like membrane system network consists of several cells, each with its own multiset of objects, and CMA-ES serves as reaction rules to evolve objects in different cells. Mohamed et al. [11] suggested L-SHADE-SPACMA, where individuals are controlled by either L-SHADE-SPA or a modified version of CMA-ES, with the better performer controlling more individuals.

3. Methodology

This section first presents details of HS-ES and L-SHADE-EpSin, then introduces our proposed HS-ES-DE algorithm based on our analysis of these two algorithms. The pseudocode for all three algorithms is provided.

3.1. The Involved Population-Based Metaheuristics

3.1.1. HS-ES The pseudocode for HS-ES is shown in Algorithm 1. Essentially, HS-ES is an ensemble of an improved version of univariate sampling [22] and CMA-ES [6]. The constituent algorithms are not explained in detail here; readers should refer to [22] and [6] for specifics.

3.1.2. L-SHADE-EpSin The pseudocode for L-SHADE-EpSin is given in Algorithm 2, followed by equations requiring further explanation. However, mutation, crossover, and selection—which are inherited from JADE [23]—are not detailed here.

In the first stage of L-SHADE-EpSin, the non-adaptive sinusoidal decreasing adjustment for each target vector in the gth generation is:

$$F_{i,g} = \left\lfloor \sin(2\pi \cdot f_{req} \cdot g + \pi) \cdot \frac{G_{max} - g}{G_{max}} \right\rfloor$$

where i ranges from 1 to NP, g is the current generation number, f_{req} represents the frequency of the sinusoidal function, and G_{max} is the maximum generation, computed based on the maximum number of function evaluations $MaxFES$ and the LPSR settings.

Meanwhile, the adaptive sinusoidal increasing adjustment for each target vector in the gth generation is:

$$F_{i,g} = \left\lfloor \sin\left(2\pi \cdot f_{req_{i,g}} \cdot g\right) \cdot \frac{g}{G_{max}} \right\rfloor$$

where $f_{req_{i,g}}$ is computed based on Equation 3:

$$f_{req_{i,g}} = \text{randc}(\mu_{f_{req_{ri}}}, 0.1)$$

In Equation 3, $\mu_{f_{req_{ri}}}$ is chosen randomly from external memory $M_{f_{req}}$ which stores successful mean frequencies in history.

In the second stage:

$$F_{i,g} = \text{randc}(\mu_{F_{ri}}, 0.1)$$

where $\mu_{F_{ri}}$ is chosen randomly from external memory M_F which stores the successful mean of F in history.

Throughout the process:

$$Cr_{i,g} = \text{randn}(\mu_{Cr_{ri}}, 0.1)$$

where $\mu_{Cr_{ri}}$ is chosen randomly from external memory M_{Cr} which stores the successful mean of Cr in history.

For update:

$$\mu_{f_{req_{ri,g+1}}} = \text{mean}_L(S_{f_{req}}), \quad \mu_{F_{ri,g+1}} = \text{mean}_L(S_F), \quad \mu_{Cr_{ri,g+1}} = \text{mean}_L(S_{Cr})$$

Here, $\text{mean}_L(S_{f_{req}})$, $\text{mean}_L(S_F)$, and $\text{mean}_L(S_{Cr})$ are computed based on:

$$\text{mean}_L(S) = \frac{\sum_{i=1}^{|S|} w_i \cdot S_i^2}{\sum_{i=1}^{|S|} w_i \cdot S_i}, \quad w_i = \frac{\Delta f_i}{\sum_{j=1}^{|S|} \Delta f_j}, \quad \Delta f_i = |f(u_{i,g}) - f(x_{i,g})|$$

For the LPSR:

$$NP = \text{Round}\left[\frac{NP_{min} - NP_{max}}{MaxFES} \cdot FES + NP_{max}\right]$$

For the Gaussian Walks, ten randomly selected target vectors are reinitialized. Then, in the following G_{ls} generations:

$$y_m = \text{Gaussian}(\mu_b, \sigma) + (\epsilon \cdot \mathbf{x}_{mbest} - \hat{\epsilon} \cdot x_m)$$

where $m \in [1, 10]$, ϵ and $\hat{\epsilon}$ are uniform random numbers in $[0, 1]$, and $\mathbf{x}_{mbest}$ is the best individual among the reinitialized ten. Here:

$$\sigma = \left|\frac{\log(G)}{G}\right| \cdot \|\mathbf{x}_m - \mathbf{x}_{mbest}\|$$

3.2. Our Algorithm

The first step of HS-ES involving univariate sampling can be viewed as an initialization method superior to random initialization. In the second step, CMA-ES may run multiple times. However, the termination criterion for each CMA-ES execution—lack of progress on the best solution in just one generation—often terminates execution long before actual stagnation occurs, as it cannot reflect stagnation in most cases. Thus, in HS-ES, CMA-ES is employed repeatedly merely to obtain approximate solutions, after which univariate sampling aims for further improvement based on the foundation established by CMA-ES.

We believe the initialization step is reasonable. However, it may be preferable to execute CMA-ES until stagnation actually occurs, given its power. To utilize the remaining function evaluations after stagnation, another population-based metaheuristic could take over the population. We select L-SHADE-EpSin for this purpose for two reasons: First, L-SHADE-EpSin can maintain search capability longer than many other metaheuristics, as demonstrated in our experiments. Second, as a DE algorithm with LPSR, L-SHADE-EpSin consumes far fewer function evaluations in later generations than in earlier ones.

Considering these factors, we propose HS-ES-DE, shown in Algorithm 3. The improving rate of average fitness (IR) in Step 12 is calculated as:

$$IR = \left|\frac{f_p - f_c}{f_p}\right|$$

where f_p denotes the average fitness in the previous generation and f_c denotes the average fitness in the current generation. Step 12 employs a strict criterion to detect stagnation.

To produce the appropriate number of individuals for L-SHADE-EpSin, CMA-ES is executed for one additional generation without fitness evaluation after stagnation is confirmed. Individuals are then diversified using DE with direct offspring selection, also without fitness evaluation. Thus, our population processing scheme for handover requires no fitness evaluations.

4. Experiments

We compare HS-ES-DE with 12 population-based metaheuristics using the CEC 2014, 2017, and 2020 benchmark test suites. Functions in these suites can be divided into four types as shown in Table 1 .

For CEC 2014 and 2017 comparisons, we select L-SHADE-EpSin, UMOEAs-II, MPEDE, ETI-JADE [4], HS-ES, AMECoDEs, EDEV, MLCC-SI, and NDE [18] as peers, including top performers from CEC competitions and recent state-of-the-art algorithms. Notably, the maximum number of function evaluations was significantly increased in the CEC 2020 competition. The top three algorithms under the new rules are IMODE, AGSK, and j2020, which serve as peers for CEC 2020 comparisons.

4.1. Experimental Settings

Algorithm parameters are listed in Table 2 , where D denotes dimensionality. For HS-ES-DE, only the listed parameters require consideration; other parameters from HS-ES and L-SHADE-EpSin remain unchanged.

For CEC 2014 and 2017 suites, the maximum number of function evaluations $MaxFES$ is set to $D \cdot 10000$ (i.e., 3.0×10^5, 5.0×10^5, and 1.0×10^6 for $D = 30, 50, 100$, respectively) following previous CEC competition rules. For CEC 2020, $MaxFES$ is set to 2.0×10^5 and 1.0×10^7 for $D = 20$, representing previous and 2020 competition rules, respectively. Each algorithm is executed 30 times per case.

4.2. Comparison Based on the CEC 2014 Suite

We compare HS-ES-DE with nine metaheuristics on CEC 2014 functions at dimensionalities 30, 50, and 100, with results shown in Tables 3-5 .

At $D = 30$, HS-ES-DE performs significantly better than L-SHADE-EpSin, UMOEAs-II, MPEDE, ETI-JADE, HS-ES, AMECoDEs, EDEV, MLCC-SI, and NDE on 6, 12, 20, 21, 18, 19, 21, 17, and 19 functions, respectively, while being inferior on 9, 12, 4, 4, 4, 5, 3, 5, and 7 functions. Although losing to L-SHADE-EpSin and UMOEAs-II, HS-ES-DE outperforms all other peers.

Detailed analysis reveals: For unimodal functions (F1-F3), HS-ES-DE outperforms most peers except L-SHADE-EpSin. For 13 simple multimodal functions (F4-F16), it surpasses all competitors. For six hybrid functions (F17-F22), it performs better than ETI-JADE, AMECoDEs, and EDEV but worse than L-SHADE-EpSin, UMOEAs-II, HS-ES, MLCC-SI, and NDE, while being comparable to MPEDE. For eight composition functions (F23-F30), it outperforms most peers except L-SHADE-EpSin.

At $D = 50$, HS-ES-DE outperforms all peers across all function types, demonstrating superior performance in most categories. At $D = 100$, it again outperforms all peers except L-SHADE-EpSin, showing particular strength in simple multimodal and composition functions.

The Friedman test results are shown in Table 6 . HS-ES-DE ranks third at $D = 30$, second at $D = 50$, and first at $D = 100$.

Convergence graphs for $D = 100$ are shown in Figure 1 [Figure 1: see original paper] for 14 functions. Some algorithms like UMOEAs-II and HS-ES may show deteriorating best fitness during execution because CMA-ES-based algorithms do not necessarily preserve the best-ever solution. This phenomenon also appears in the early stages of HS-ES-DE, which begins with HS-ES. However, both L-SHADE-EpSin and HS-ES-DE demonstrate solution improvement even in the final stage, with sudden decreases visible in the terminal convergence graph segments.

4.3. Comparison Based on the CEC 2017 Suite

Table 7 shows results for CEC 2017 functions at $D = 30$. HS-ES-DE outperforms L-SHADE-EpSin, UMOEAs-II, MPEDE, ETI-JADE, HS-ES, AMECoDEs, EDEV, MLCC-SI, and NDE on 11, 10, 15, 20, 13, 19, 17, 16, and 17 functions, respectively, while being inferior on 10, 12, 9, 3, 7, 6, 7, 6, and 8 functions. Although losing to UMOEAs-II, it outperforms all other peers.

For unimodal functions, HS-ES-DE performs better than most peers except L-SHADE-EpSin. For seven simple multimodal functions, it outperforms most competitors except HS-ES. For ten hybrid functions, it surpasses ETI-JADE and EDEV but performs worse than several others while being comparable to AMECoDEs. For eight composition functions, it outperforms most peers except UMOEAs-II.

The Friedman test in Table 8 shows HS-ES-DE ranks third.

4.4. Comparison Based on the CEC 2020 Suite

We compare HS-ES-DE with the top three 2020 competition algorithms—IMODE, AGSK, and j2020—using both old and new $MaxFES$ rules. Table 9 shows results for CEC 2020 functions at $D = 20$.

With $MaxFES = 2.0 \times 10^5$ (old rule), HS-ES-DE outperforms IMODE, AGSK, and j2020 on 8, 5, and 6 functions, respectively, while being inferior on 2, 1, and 2 functions. It demonstrates superior performance across all peers, particularly on unimodal and simple multimodal functions.

With $MaxFES = 1.0 \times 10^7$ (new rule), HS-ES-DE outperforms each peer on only one function while being inferior on 8, 6, and 8 functions, respectively, indicating worse performance than all peers under long-budget conditions.

The Friedman test in Table 10 confirms HS-ES-DE' s competitiveness under the old rule but poor ranking under the new rule.

4.5. Result Analysis

Under the old $MaxFES$ rule, HS-ES-DE proves highly competitive against all peers. However, under the new rule, it performs significantly worse than IMODE, AGSK, and j2020. The old rule evaluates solving ability over a short course, while the new rule assesses long-course performance. In summary, HS-ES-DE demonstrates strong solving ability for short optimization runs.

5. Conclusion

Population-based metaheuristics have been extensively applied to real parameter single objective optimization, with CEC competitions driving advancement in this field. Based on L-SHADE-EpSin and HS-ES—two competition winners —we propose HS-ES-DE, an ensemble with successively executed constituents. HS-ES runs first, and upon stagnation detection, L-SHADE-EpSin takes over after specific population processing. Experimental results demonstrate that our ensemble is competitive among population-based metaheuristics for real parameter single objective optimization.

Since timely handover yields improvement, alternating handovers warrant investigation. Future work will enhance HS-ES-DE with multiple handover cycles, allowing L-SHADE-EpSin and HS-ES to alternate control. This will require more complex population processing but may yield better performance suited for both short and long optimization courses.

References

[1] Mostafa Z Ali, Noor H Awad, Ponnuthurai Nagaratnam Suganthan, and Robert G Reynolds. An adaptive multipopulation differential evolution with dynamic population reduction. *IEEE Trans. on Cybernetics*, 47(9):2768–2779, 2017.

[2] Noor H Awad, Mostafa Z Ali, Ponnuthurai N Suganthan, and Robert G Reynolds. An ensemble sinusoidal parameter adaptation incorporated with L-SHADE for solving CEC2014 benchmark problems. In *Proc of CEC*, pages 2958–2965. IEEE, 2016.

[3] Laizhong Cui, Genghui Li, Zexuan Zhu, Qiuzhen Lin, Ka-Chun Wong, Jianyong Chen, Nan Lu, and Jian Lu. Adaptive multiple-elites-guided composite differential evolution algorithm with a shift mechanism. *Information Sciences*, 422:122–143, 2018.

[4] Wei Du, Sunney Yung Sun Leung, Yang Tang, and Athanasios V Vasilakos. Differential evolution with event-triggered impulsive control. *IEEE Trans. on Cybernetics*, 47(1):244–257, 2017.

[5] Saber Elsayed, Noha Hamza, and Ruhul Sarker. Testing united multi-operator evolutionary algorithms-II on single objective optimization problems. In *Proc of CEC*, pages 2966–2973. IEEE, 2016.

[6] Nikolaus Hansen, Sibylle D Müller, and Petros Koumoutsakos. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES). *Evolutionary computation*, 11(1):1–18, 2003.

[7] Nikolaus Hansen and Andreas Ostermeier. Completely derandomized self-adaptation in evolution strategies. *Evolutionary computation*, 9(2):159–195, 2001.

[8] Abhishek Kumar, Rakesh Kumar Misra, and Devender Singh. Improving the local search capability of effective butterfly optimizer using covariance matrix adapted retreat phase. In *Proc of CEC*, pages 1835–1842. IEEE, 2017.

[9] Chuang Liu and Linan Fan. A hybrid evolutionary algorithm based on tissue membrane systems and cma-es for solving numerical optimization problems. *Knowledge-Based Systems*, 105:38–47, 2016.

[10] Ilya Loshchilov. CMA-ES with restarts for solving CEC 2013 benchmark problems. In *Proc of CEC*, pages 369–376. IEEE, 2013.

[11] Ali W Mohamed, Anas A Hadi, Anas M Fattouh, and Kamal M Jambi. Lshade with semi-parameter adaptation hybrid with cma-es for solving cec 2017 benchmark problems. In *2017 IEEE Congress on evolutionary computation (CEC)*, pages 145–152. IEEE, 2017.

[12] Karam M Sallam, Saber M Elsayed, Ripon K Chakrabortty, and Michael J Ryan. Improved multi-operator differential evolution algorithm for solving unconstrained problems. In *2020 IEEE Congress on Evolutionary Computation (CEC)*, pages 1–8. IEEE, 2020.

[13] Karam M Sallam, Saber M Elsayed, Ruhul A Sarker, and Daryl L Essam. Landscape-based adaptive operator selection mechanism for differential evolution. *Information Sciences*, 418:383–404, 2017.

[14] Urban Škvorc, Tome Eftimov, and Peter Korošec. CEC real-parameter optimization competitions: Progress from 2013 to 2018. In *Proc of CEC*, pages 3126–3133. IEEE, 2019.

[15] Rainer Storn and Kenneth Price. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11(4):341–359, 1997.

[16] Ryoji Tanabe and Alex Fukunaga. Success-history based parameter adaptation for differential evolution. In *Proc of CEC*, pages 71–78. IEEE, 2013.

[17] Ryoji Tanabe and Alex S Fukunaga. Improving the search performance of SHADE using linear population size reduction. In *Proc of CEC*, pages 1658–1665. IEEE, 2014.

[18] Mengnan Tian and Xingbao Gao. Differential evolution with neighborhood-based adaptive evolution mechanism for numerical optimization. *Information Sciences*, 478:422–448, 2019.

[19] Yong Wang, Zixing Cai, and Qingfu Zhang. Differential evolution with composite trial vector generation strategies and control parameters. *IEEE Trans. on Evo. Comput.*, 15(1):55–66, 2011.

[20] Guohua Wu, Rammohan Mallipeddi, Ponnuthurai Nagaratnam Suganthan, Rui Wang, and Huangke Chen. Differential evolution with multi-population based ensemble of mutation strategies. *Information Sciences*, 329:329–345, 2016.

[21] Guohua Wu, Xin Shen, Haifeng Li, Huangke Chen, Anping Lin, and PN Suganthan. Ensemble of differential evolution variants. *Information Sciences*, 423:172–186, 2018.

[22] Geng Zhang and Yuhui Shi. Hybrid sampling evolution strategy for solving single objective bound constrained problems. In *Proc of CEC*, pages 1-7. IEEE, 2018.

[23] Jingqiao Zhang and Arthur C Sanderson. JADE: adaptive differential evolution with optional external archive. *IEEE Trans. on Evo. Comput.*, 13(5):945–958, 2009.

[24] Sheng Xin Zhang, Li Ming Zheng, Kit Sang Tang, Shao Yong Zheng, and Wing Shing Chan. Multi-layer competitive-cooperative framework for performance enhancement of differential evolution. *Information Sciences*, 482:86–104, 2019.

Note: Figure translations are in progress. See original paper for figures.