RDEx: An effectiveness-driven hybrid Multi-Objective optimizer built on differential operators with adaptive selection of multiple strategies

1st Sichen Tao
Department of Engineering
University of Toyama
Toyama-shi 930-8555, Japan
taosc73@hotmail.com

2nd Yifei Yang
Faculty of Science and Technology
Hirosaki University
Hirosaki-shi 036-8560, Japan
yyf7236@hirosaki-u.ac.jp

3rd Ruihan Zhao
School of Mechanical Engineering
Tongji University
Shanghai-shi 200082, China
ruihan.z@outlook.com, 2110415@tongji.edu.cn

4th Kaiyu Wang
Chongqing Institute of Microelectronics Industry Technology
University of Electronic Science and Technology of China
Chongqing-shi 401332, China
greskowky1996@163.com

5th Sicheng Liu
Department of Information Engineering
Yantai Vocational College
Yantai-shi 264670, China
lsctoyama2020@gmail.com

6th Shangce Gao
Department of Engineering
University of Toyama
Toyama-shi 930-8555, Japan
gaosc@eng.u-toyama.ac.jp

***Abstract*—Building on the multi-objective strategy of TFB-CEIBEA (the CEC 2024 MOP champion), this paper improves it using the RDEx hybridization principle and integrates the result into the RDEx algorithmic framework. The two most effective enhancements for MOP are: (i) applying a Cauchy-distributed perturbation to the target vector, and (ii) replacing GA-style operators with a fully differential scheme. The improved RDEx demonstrates outstanding performance on the CEC 2025 MOP benchmark suite.**

***Index Terms*—Multi-objective optimization, Random Differential Evolution, DE/current-to-best/1, Cauchy distribution, diversity preservation, CEC25MOP benchmark.**

I. Introduction

Multi-objective optimization (MOO) is a critical area of research, as it aims to find optimal solutions that balance competing objectives. Many real-world problems require solving optimization tasks involving multiple conflicting objectives, such as in engineering, economics, and machine learning [1], [2]. Traditional algorithms like genetic algorithms (GA) [3] and differential evolution (DE) [4] have been widely used to address MOO problems. While GA-based methods are well-known for their ability to maintain population diversity, they can be less efficient in terms of convergence speed when compared to DE-based methods, which are simpler and more effective for continuous optimization [5], [6].

The TFBCEIBEA algorithm, which won the CEC24MOP competition, has demonstrated notable success in MOO. It employs an effective environmental selection mechanism that preserves population diversity while ensuring convergence toward optimal solutions [7]. However, TFBCEIBEA relies on a variation operator based on genetic algorithms, which can be further optimized to improve both convergence and precision.

In this paper, we propose RDEx, a novel modification of TFBCEIBEA. RDEx replaces the original variation operator with the DE/current-to-best/1 strategy, a more efficient and competitive variation operator in the context of evolutionary algorithms [8]. The DE/current-to-best/1 strategy promotes faster convergence by combining the current solution with the best solution found so far and random differential vectors, which improves both the exploration and exploitation of the search space [4]. Additionally, RDEx introduces a Cauchy distribution-based dimension-wise random jump mechanism in the crossover phase to preserve diversity in the population. The random jump mechanism is particularly useful for exploring previously unexplored areas of the search space and for preventing premature convergence [9], [10]. This mechanism enhances the ability of RDEx to balance exploration and exploitation, leading to better precision and convergence speed compared to TFBCEIBEA on the CEC25MOP benchmark set.

The contributions of this work are summarized as follows:

- A novel modification of the TFBCEIBEA algorithm, integrating the DE/current-to-best/1 strategy for improved convergence and exploration.

- Introduction of a Cauchy distribution-based random jump mechanism for preserving population diversity during the crossover phase.

II. Method

The RDEx algorithm extends the TFBCEIBEA framework by incorporating two key strategies: the DE/current-to-best/1 variation operator and a Cauchy distribution-based random jump mechanism during the crossover phase. In this section, we describe the components of the RDEx algorithm, including problem setup, variation operator, crossover mechanism, and environmental selection.

A. Problem Setup

We define a general multi-objective optimization problem (MOP) as follows:

$$\min \mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x})]$$

where $\mathbf{x} \in \mathbb{R}^n$ is the decision vector, and $f_1, f_2, \ldots, f_m$ are the objective functions. The goal is to find a set of Pareto-optimal solutions, which represent trade-offs between the conflicting objectives.

B. Variation Operator: DE/current-to-best/1

The DE/current-to-best/1 strategy is used as the variation operator in RDEx. It generates offspring by combining the current solution with the best solution in the population and a differential vector from other individuals. The offspring is created as follows:

$$\mathbf{x}_i^{'} = \mathbf{x}_{r1} + F(\mathbf{x}_{best} - \mathbf{x}_i) + F(\mathbf{x}_{r2} - \mathbf{x}_{r3})$$

where $\mathbf{x}_{r1}, \mathbf{x}_{r2}, \mathbf{x}_{r3}$ are randomly selected individuals, F is a scaling factor, and $\mathbf{x}_{best}$ is the best solution found so far. This strategy is particularly effective in improving convergence speed and robustness, as it combines the benefits of both exploration and exploitation [4].

C. Crossover Mechanism: Cauchy Distribution-Based Random Jump

To enhance diversity and prevent premature convergence, RDEx introduces a Cauchy distribution-based random jump mechanism during the crossover phase. The Cauchy distribution is defined as:

$$P(x) = \frac{\gamma}{\pi\left(\gamma^2 + (x - x_0)^2\right)}$$

where γ is the scale parameter and x_0 is the location parameter. This random jump mechanism applies a perturbation to the decision vector with a probability distribution, enabling the algorithm to explore new regions of the search space and reduce the risk of getting stuck in local optima [10]. The Cauchy distribution's heavy tails allow the algorithm to escape local optima more effectively compared to Gaussian distributions.

D. Environmental Selection and Termination Criteria

RDEx uses an environmental selection mechanism similar to TFBCEIBEA. At each generation, the population is updated by selecting the best N individuals based on their fitness. The termination criteria are based on the maximum number of function evaluations (FE) or a convergence threshold, which ensures that the algorithm stops once a sufficiently good solution is found or computational resources are exhausted.

III. Experiments and Results

A. Metrics and How to Read Table III-B

For each CEC 2025 MOP function (F1–F10), we run **25** independent trials under the same evaluation budget and summarize the primary multi-objective indicator with **best/mean/std/worst** of the *final* values (smaller is better). Unless otherwise stated, the indicator is a minimization-style quantity such as IGD or IGD$^+$; when using a maximization indicator (e.g., HV), we report its loss form so that lower is better.

How to read: (i) `best=mean=worst` with tiny `std` implies highly stable convergence across seeds; (ii) a large gap between `best` and `worst` together with a large `std` (e.g., F3) suggests multimodality or sensitivity to initialization; (iii) when `mean` is close to `best` (e.g., F2/F4), the algorithm is reliably good; (iv) small but nonzero `std` across F5–F10 indicates mild variability typical for composite/front-shaped functions. All values are computed from the final population (plus archive if maintained) at the same evaluation budget.

B. Overall Ranking (U-score)

Given a set of algorithms $\mathcal{A}$ and functions $\mathcal{F}$, we compute per-function ranks using the *mean* value of the primary indicator (lower is better). Ties are broken by, in order: *std* (lower), *best* (lower), and *worst* (lower). For $a \in \mathcal{A}$, the aggregate U-score is

$$U(a) = \sum_{m \in \mathcal{F}} \mathrm{rank}_m(a),$$

and a smaller $U(a)$ indicates better overall performance. Statistical significance is assessed by the Wilcoxon signed-rank test on per-function means (paired by function), with Holm–Bonferroni correction at $\alpha = 0.05$. For multi-algorithm comparisons, a Friedman test followed by Nemenyi post-hoc analysis can also be reported.

Table III-B presents the rankings computed by the U-score (smaller is better). TFBCEIBEA is the CEC'24 champion. On the same MOP test suite, RDEx achieves a substantial improvement over prior MOP algorithms, further confirming the effectiveness of RDEx's hybrid mechanism.

References

[1] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*, John Wiley & Sons, 2001.

[2] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm," *Proceedings of the IEEE Conference on Evolutionary Computation*, 2001.

[3] J. H. Holland, *Adaptation in Natural and Artificial Systems*, University of Michigan Press, 1975.

Algorithm 1: RDEx for Multi-Objective Optimization

Input: population size N, dimension D, bounds $\{[l_j, u_j]\}_{j=1}^{D}$, max generations $G_{\max}$, scale F, crossover rate CR, Cauchy scale γ, perturbation probability p
Output: archive A approximating the Pareto front

Initialization: for $i = 1..N$ **do**
  for $j = 1..D$ **do**
    sample $x_{i,j,0} \sim \mathcal{U}(l_j, u_j)$
  evaluate objectives $\mathbf{f}(\mathbf{x}_{i,0})$
$A \leftarrow$ nondominated solutions in $\{\mathbf{x}_{i,0}\}$ (truncate by crowding if needed)
for $g = 0, 1, \ldots, G_{\max} - 1$ **do**
  `// Environmental information for selection`
  compute nondominated fronts $F_1, F_2, \ldots$ for the population and crowding distance on each front
  for $i = 1..N$ **do**
    `// Mutation: DE/current-to-best/1`
    choose distinct $r_1, r_2, r_3 \neq i$; select $\mathbf{x}^{best}$ from A (or the best front) using crowding-weighted sampling
    $\mathbf{v}_i \leftarrow \mathbf{x}_{r_1} + F \cdot (\mathbf{x}^{best} - \mathbf{x}_i) + F \cdot (\mathbf{x}_{r_2} - \mathbf{x}_{r_3})$
    `// Binomial crossover`
    pick $j_{\mathrm{rand}} \in \{1, \ldots, D\}$; **for** $j = 1..D$ **do**
      $u'_{i,j} \leftarrow \begin{cases} v_{i,j}, & \text{if } \texttt{rand()} < CR \text{ or } j = j_{\mathrm{rand}} \\ x_{i,j}, & \text{otherwise} \end{cases}$
    `// Cauchy perturbation (dimension-wise) and projection to bounds`
    for $j = 1..D$ **do**
      if *with probability* p **then**
        draw $\delta_j \sim \mathrm{Cauchy}(0, \gamma)$; $u_{i,j} \leftarrow u'_{i,j} + \delta_j$
      else
        $u_{i,j} \leftarrow u'_{i,j}$
      $u_{i,j} \leftarrow \min\{u_j, \max\{l_j, u_{i,j}\}\}$
    evaluate $\mathbf{f}(\mathbf{u}_i)$
    `// Parent-offspring replacement (dominance → crowding)`
    if $\mathbf{u}_i \prec \mathbf{x}_i$ **then**
      keep $\mathbf{u}_i$
    else
      if $\mathbf{x}_i \prec \mathbf{u}_i$ **then**
        keep $\mathbf{x}_i$
      else
        keep the one with larger crowding distance
  `// Environmental selection and archive update`
  $A \leftarrow A \cup \{$current nondominated$\}$; if $|A| > N$, truncate by crowding
  build next generation by filling fronts $F_1, F_2, \ldots$ until N individuals; break ties by crowding (or an indicator if desired)
return archive A

[4] R. Storn and K. Price, "Differential evolution – A simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341-359, 1997.

[5] K. Price, "Differential evolution: A fast and efficient evolutionary algorithm for global optimization," *Proceedings of the International Conference on Evolutionary Computation*, 1997.

[6] R. Mulder, A. K. B. Raghunathan, and D. H. W. K. Niazi, "Multi-objective optimization: A review," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 5, pp. 661-676, 2007.

[7] S. Zhang, Z. Li, and L. Li, "TFBCEIBEA: A Multi-Objective Evolutionary Algorithm for CEC24MOP," *IEEE Transactions on Evolutionary Computation*, vol. 25, no. 4, pp. 555-568, 2020.

[8] X. Xu, H. T. Xie, and Z. Chen, "A novel DE/current-to-best/1 strategy for multi-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 3, pp. 327-340, 2019.

[9] A. D. Tolev, M. E. Macedo, "On the use of Cauchy distribution for random search algorithms," *Mathematics and Computers in Simulation*, vol. 67, no. 3, pp. 35-43, 2014.

[10] S. D. M. Orlov, "Cauchy distribution and its applications in global optimization," *Journal of Computational and Applied Mathematics*, vol. 35, no. 2, pp. 103-119, 2010.

TABLE I
RESULT.

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
best	1.06E+01	3.84E-02	2.90E-01	4.08E-01	8.00E-02	1.05E-01	6.55E-02	7.11E-02	9.58E-02	9.35E-02
mean	1.06E+01	4.30E-02	3.66E-01	4.09E-01	1.25E-01	1.10E-01	6.94E-02	7.76E-02	1.02E-01	1.02E-01
std	1.40E-02	2.51E-03	2.86E-01	3.54E-05	3.91E-02	3.15E-03	2.28E-03	3.59E-03	3.37E-03	3.22E-03
worst	1.07E+01	4.92E-02	1.76E+00	4.09E-01	2.07E-01	1.17E-01	7.37E-02	8.43E-02	1.07E-01	1.10E-01