

Đại học Quốc gia thành phố Hồ Chí Minh  
Trường Đại học Công nghệ Thông tin

Khoa Khoa học Máy tính

---



# **MẠNG NEURAL VÀ THUẬT GIẢI DI TRUYỀN**

## **BÁO CÁO BÀI TẬP SỐ 2**

Differential Evolution và Cross Entropy Method

---

Huỳnh Anh Dũng – 22520278  
Lớp: CS410.P21

---

Ngày thực hiện: ngày 03 tháng 4 năm 2025  
Giảng viên hướng dẫn: thầy Lương Ngọc Hoàng

# Mục lục

<b>1</b>	<b>Giới thiệu</b>	<b>2</b>
<b>2</b>	<b>Kết quả thực nghiệm</b>	<b>3</b>
2.1	Thực nghiệm với hàm benchmark Sphere . . . . .	3
2.2	Thực nghiệm với hàm benchmark Rosenbrock . . . . .	4
2.3	Thực nghiệm với hàm benchmark Rastrigin . . . . .	5
2.4	Thực nghiệm với hàm benchmark Griewank . . . . .	6
2.5	Thực nghiệm với hàm benchmark Ackley . . . . .	6
<b>3</b>	<b>Kết luận</b>	<b>8</b>
<b>4</b>	<b>Minh họa</b>	<b>9</b>

# 1. Giới thiệu

Bài báo cáo thực nghiệm bài toán tối ưu hóa trên hai thuật toán *Differential Evolution* (DE) và *Cross-Entropy Method* (CEM) nhằm đánh giá tốc độ hội tụ, độ chính xác của lời giải cuối và tính ổn định của giải thuật khi thử nghiệm trên nhiều hàm benchmark khác nhau.

Bộ hàm benchmark sẽ được sử dụng bao gồm:

- Hàm *Sphere*: hàm lồi có công thức đơn giản, thường dùng làm tiêu chí tiên quyết của bài toán tối ưu hóa.
- Hàm *Griewank*: hàm benchmark có công thức phức tạp tạo nhiều điểm cực tiểu lân cận, gây khó khăn cho việc tìm ra lời giải tối ưu toàn cục.
- Hàm *Rosenbrock*: hàm phi lồi với đoạn lũng hẹp và cong dẫn đến điểm cực tiểu toàn cục.
- Hàm *Rastrigin*: hàm benchmark có tính đa cực cao nhưng có một lượng lớn các cực tiểu lân cận nằm theo phân bố đều.
- Hàm *Ackley*: hàm có nhiều cực tiểu lân cận phân bố đều ở vùng ngoài cùng với một cực tiểu toàn cục tại trung tâm, dễ khiến lời giải của các giải thuật tối ưu mắc kẹt ở một trong những cực tiểu lân cận.

Với mỗi bộ tham số thực nghiệm (bao gồm kích thước quần thể, số lượng tham số, hàm benchmark), sinh viên sẽ chạy với 10 random seed [22520278; 22520278 + 10) để đảm bảo tính khách quan của thực nghiệm. Giải pháp tối ưu toàn cục và không gian tìm kiếm (với số lượng tham số bằng 2) với mỗi hàm được tham khảo từ [đường dẫn](#).

Bảng 1.1: Miền tìm kiếm được gợi ý và giải pháp tối ưu toàn cục của các hàm benchmark

Hàm	Giải pháp	Miền tìm kiếm
Sphere	(0; 0)	$[-5.12; 5.12]$
Griewank	(0; 0)	$[-600; 600]$
Rosenbrock	(1; 1)	$[-5; 10]$
Rastrigin	(0; 0)	$[-5.12; 5.12]$
Ackley	(0; 0)	$[-32.768; 32.768]$

Trong quá trình thực nghiệm, một số siêu tham số của mô hình đã được sinh viên cài đặt như sau:

- Với DE: các siêu tham số `scale_factor` và `crossover_probabilities` được cài đặt lần lượt là 0.7 và 0.5.
- Với CEM: siêu tham số `elites_size` được cài đặt bằng  $\frac{1}{2}$  của `population_size`.

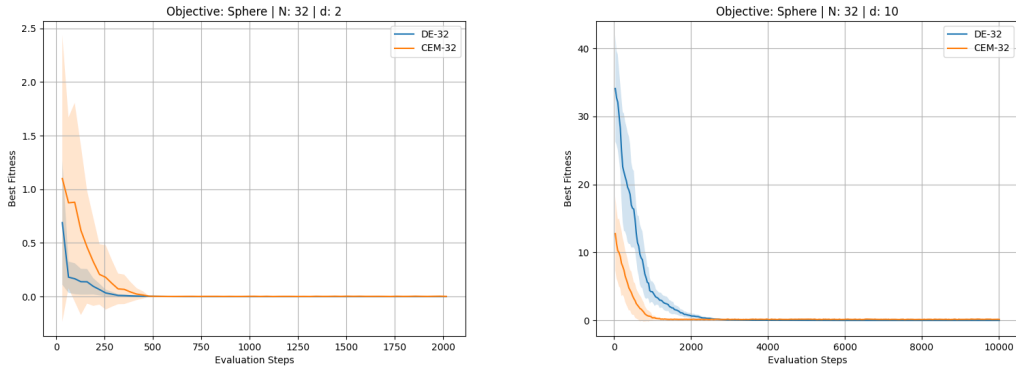
## 2. Kết quả thực nghiệm

\*Các kết quả thực nghiệm bên dưới đều được làm tròn đến chữ số thập phân thứ 2, p-value được làm tròn đến chữ số thập phân thứ 3.

\*\*Nếu giá trị p-value dưới ngưỡng 0.05 thì kết quả t-test sẽ được coi là có tầm quan trọng.

### 2.1 Thực nghiệm với hàm benchmark Sphere

Hình 2.1: Biểu đồ hội tụ của hai thuật toán với hàm Sphere với  $N = 32 \wedge d \in \{2; 10\}$



Bảng 2.1: Bảng so sánh giải pháp tốt nhất với hàm benchmark Sphere

N	DE	CEM	p-value
$d = 2$			
8	0.0 ± 0.0	0.0 ± 0.0	<b>0.001</b>
16	0.0 ± 0.0	0.0 ± 0.0	<b>0.004</b>
32	0.0 ± 0.0	0.0 ± 0.0	<b>0.007</b>
64	0.0 ± 0.0	0.0 ± 0.0	<b>0.002</b>
128	0.0 ± 0.0	0.0 ± 0.0	0.52
$d = 10$			
8	<b>0.0 ± 0.0</b>	0.29 ± 0.11	<b>0.0</b>
16	<b>0.0 ± 0.0</b>	0.22 ± 0.08	<b>0.0</b>
32	<b>0.0 ± 0.0</b>	0.15 ± 0.04	<b>0.0</b>
64	<b>0.0 ± 0.0</b>	0.15 ± 0.04	<b>0.0</b>
128	0.17 ± 0.05	<b>0.13 ± 0.03</b>	0.053

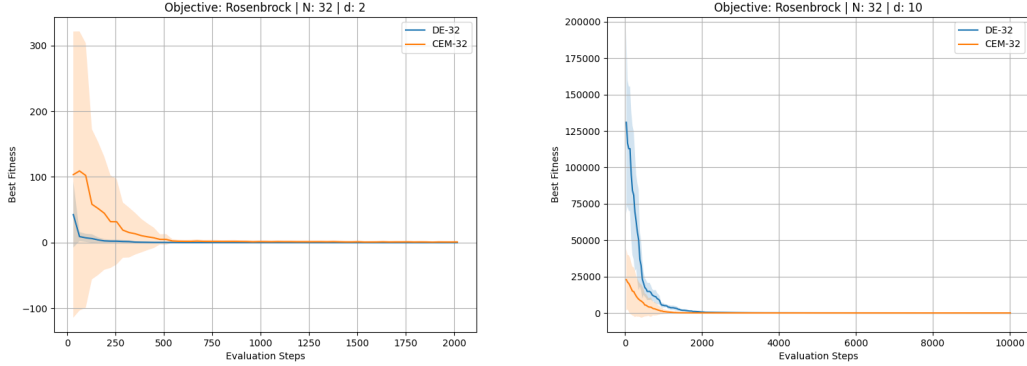
Về tốc độ hội tụ, hình 2.1 cho thấy DE và CEM có tốc độ hội tụ với hàm Sphere gần ngang nhau khi đồ thị cả hai đều có dạng giảm mạnh ở những bước đầu và ổn định sớm ở gần giá trị cực tiểu toàn cục cho đến hết thực nghiệm.

Về độ ổn định, DE thể hiện vượt trội hơn hẳn khi vùng biểu thị phương sai trên hình 2.1 đều nhỏ hơn hoặc bằng khi so với CEM.

Về độ chính xác của giải pháp cuối cùng, cả hai không có sự khác biệt quá lớn nhưng DE vẫn nhỉnh hơn CEM khi đa phần các kết quả thực nghiệm của DE đều tốt hơn hoặc bằng CEM, đặc biệt khi số chiều tăng lên 10.

## 2.2 Thực nghiệm với hàm benchmark Rosenbrock

Hình 2.2: Biểu đồ hội tụ của hai thuật toán với hàm Rosenbrock với  $N = 32 \wedge d \in \{2; 10\}$



Bảng 2.2: Bảng so sánh giải pháp tốt nhất với hàm benchmark Rosenbrock

N	DE	CEM	p-value
$d = 2$			
8	<b>0.05 <math>\pm</math> 0.11</b>	1.24 $\pm$ 2.14	0.114
16	<b>0.0 <math>\pm</math> 0.01</b>	0.61 $\pm$ 0.6	<b>0.007</b>
32	<b>0.0 <math>\pm</math> 0.0</b>	0.97 $\pm$ 1.67	0.1
64	<b>0.03 <math>\pm</math> 0.02</b>	0.83 $\pm$ 1.38	0.095
128	<b>0.11 <math>\pm</math> 0.15</b>	1.21 $\pm$ 1.81	0.088
$d = 10$			
8	<b>12.56 <math>\pm</math> 20.75</b>	35.23 $\pm$ 10.99	<b>0.01</b>
16	<b>4.52 <math>\pm</math> 0.68</b>	25.56 $\pm$ 5.96	<b>0.0</b>
32	<b>7.73 <math>\pm</math> 2.63</b>	21.86 $\pm$ 3.16	<b>0.0</b>
64	41.01 $\pm$ 13.7	<b>22.68 <math>\pm</math> 3.1</b>	<b>0.001</b>
128	244.99 $\pm$ 76.76	<b>25.0 <math>\pm</math> 16.87</b>	<b>0.0</b>

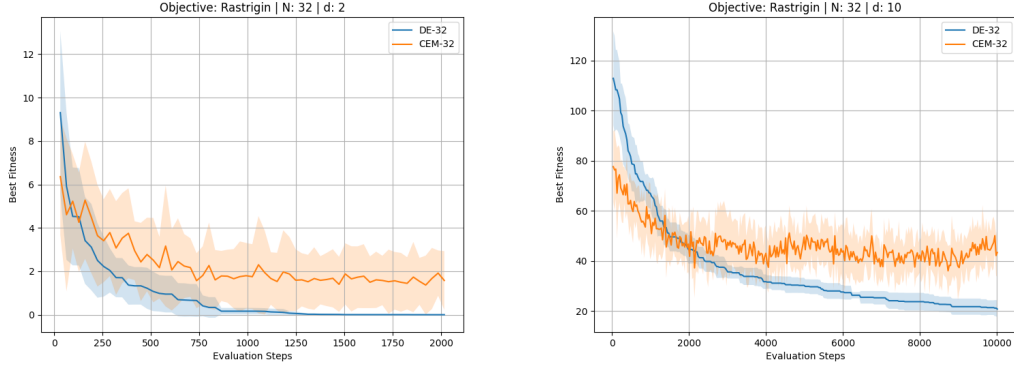
Tốc độ hội tụ của hai thuật toán theo hình 2.2 là gần như nhau, cả hai đều hội tụ sớm và ổn định tại điểm cực tiểu toàn cục.

Tuy vậy, độ ổn định của DE ở bài toán benchmark này lại vượt trội hơn hẳn so với CEM, khi mà vùng tô màu phương sai theo hình 2.2 của CEM rất rộng còn với DE lại hẹp đến gần như không có.

Độ chính xác của giải pháp cuối cùng của DE so với CEM nhỉnh hơn đôi phần khi số chiều là 2 nhưng đa phần đều không được ủng hộ khi được kiểm định với giả thuyết thống kê. Ngược lại, kết quả của thực nghiệm với số chiều bằng 10 đều được giả thuyết thống kê ủng hộ và cho thấy giải pháp của DE sẽ chính xác hơn nhưng chỉ với N nhỏ hơn 64; với N bằng 64 hoặc 128, DE không được hiệu quả bằng CEM.

## 2.3 Thực nghiệm với hàm benchmark Rastrigin

Hình 2.3: Biểu đồ hội tụ của hai thuật toán với hàm Rastrigin với  $N = 32 \wedge d \in \{2; 10\}$



Bảng 2.3: Bảng so sánh thực nghiệm sử dụng hàm benchmark Rastrigin

N	DE	CEM	p-value
$d = 2$			
8	<b><math>0.4 \pm 0.49</math></b>	$4.57 \pm 1.94$	<b>0.0</b>
16	<b><math>0.1 \pm 0.3</math></b>	$3.67 \pm 1.84$	<b>0.0</b>
32	<b><math>0.0 \pm 0.0</math></b>	$1.58 \pm 1.35$	<b>0.003</b>
64	<b><math>0.19 \pm 0.16</math></b>	$1.74 \pm 1.57$	<b>0.008</b>
128	<b><math>0.46 \pm 0.39</math></b>	$1.28 \pm 0.75$	<b>0.009</b>
$d = 10$			
8	<b><math>4.44 \pm 2.2</math></b>	$64.76 \pm 15.16$	<b>0.0</b>
16	<b><math>9.68 \pm 2.99</math></b>	$48.11 \pm 8.71$	<b>0.0</b>
32	<b><math>20.84 \pm 3.46</math></b>	$43.39 \pm 9.86$	<b>0.0</b>
64	<b><math>26.78 \pm 4.14</math></b>	$40.77 \pm 11.63$	<b>0.003</b>
128	<b><math>37.62 \pm 4.87</math></b>	$43.31 \pm 7.98$	0.084

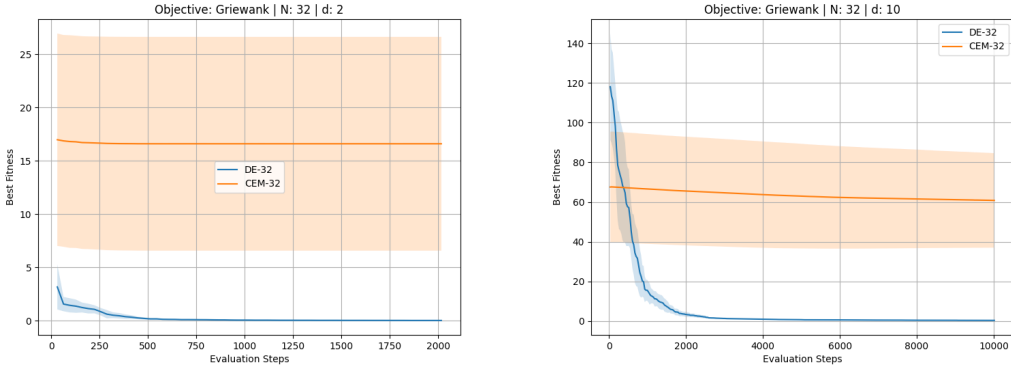
Tốc độ hội tụ của DE trong bài benchmark Griewank thực sự vượt trội hơn so với CEM vì CEM theo như hình 2.3 vẫn có biên độ dao động khá lớn mặc dù đã đến cuối thời gian thực nghiệm, nên sinh viên nhận định rằng CEM vẫn chưa đạt được mức độ hội tụ tối ưu trong khi DE đã ổn định từ những bước sớm hơn với  $d = 2$  và tiến gần hơn một cách ổn định đến giá trị cực tiểu toàn cục ở cuối thực nghiệm.

Độ ổn định của DE cũng đáng kể hơn khi đồ thị của CEM cho thấy vùng biểu thị phương sai của DE hẹp hơn CEM nhiều.

Tính chính xác của giải pháp cuối cùng vẫn vượt trội hơn về phần DE với mọi điều kiện thử nghiệm và hầu hết đều được ủng hộ bởi kiểm định giả thuyết thống kê.

## 2.4 Thực nghiệm với hàm benchmark Griewank

Hình 2.4: Biểu đồ hội tụ của hai thuật toán với hàm Griewank với  $N = 32 \wedge d \in \{2; 10\}$



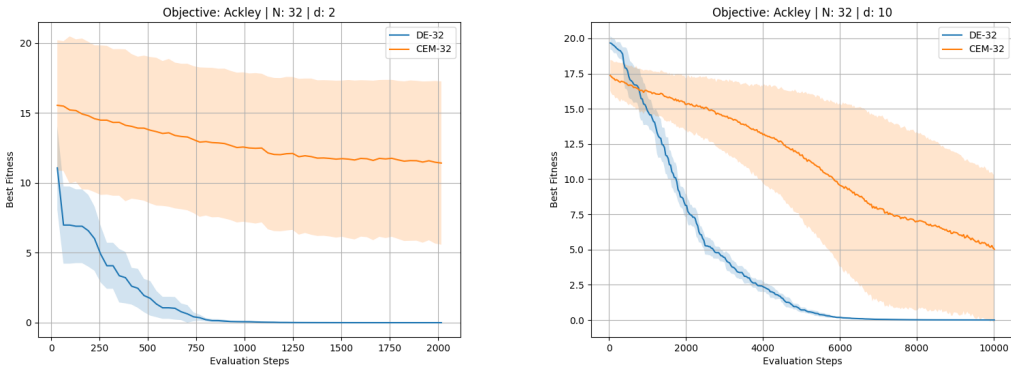
Bảng 2.4: Bảng so sánh thực nghiệm sử dụng hàm benchmark Griewank

N	DE	CEM	p-value
$d = 2$			
8	$0.01 \pm 0.01$	$16.59 \pm 10.02$	<b>0.0</b>
16	$0.0 \pm 0.0$	$16.64 \pm 10.01$	<b>0.0</b>
32	$0.02 \pm 0.01$	$16.59 \pm 10.03$	<b>0.0</b>
64	$0.04 \pm 0.02$	$16.59 \pm 10.03$	<b>0.0</b>
128	$0.07 \pm 0.05$	$16.59 \pm 10.03$	<b>0.0</b>
$d = 10$			
8	$0.06 \pm 0.05$	$47.62 \pm 14.49$	<b>0.0</b>
16	$0.21 \pm 0.08$	$54.59 \pm 20.05$	<b>0.0</b>
32	$0.35 \pm 0.08$	$60.81 \pm 23.81$	<b>0.0</b>
64	$0.62 \pm 0.08$	$63.53 \pm 26.67$	<b>0.0</b>
128	$1.57 \pm 0.19$	$64.67 \pm 27.38$	<b>0.0</b>

Với bài benchmark Griewank, CEM gần như không có hiệu quả. DE vẫn ổn định nếu không muốn nói là rất phù hợp cho bài toán này.

## 2.5 Thực nghiệm với hàm benchmark Ackley

Hình 2.5: Biểu đồ hội tụ của hai thuật toán với hàm Ackley với  $N = 32 \wedge d \in \{2; 10\}$



Bảng 2.5: Bảng so sánh thực nghiệm sử dụng hàm benchmark Ackley

N	DE	CEM	p-value
$d = 2$			
8	<b><math>0.26 \pm 0.77</math></b>	$14.43 \pm 5.39$	<b>0.0</b>
16	<b><math>0.0 \pm 0.0</math></b>	$13.55 \pm 6.77$	<b>0.0</b>
32	<b><math>0.0 \pm 0.0</math></b>	$11.41 \pm 5.83$	<b>0.0</b>
64	<b><math>0.03 \pm 0.01</math></b>	$10.91 \pm 5.45$	<b>0.0</b>
128	<b><math>0.86 \pm 0.51</math></b>	$13.13 \pm 4.94$	<b>0.0</b>
$d = 10$			
8	<b><math>0.37 \pm 0.81</math></b>	$4.17 \pm 5.03$	<b>0.038</b>
16	<b><math>0.0 \pm 0.0</math></b>	$3.06 \pm 5.24$	0.097
32	<b><math>0.0 \pm 0.0</math></b>	$5.03 \pm 5.22$	<b>0.01</b>
64	<b><math>0.77 \pm 0.18</math></b>	$7.54 \pm 4.71$	<b>0.0</b>
128	<b><math>4.85 \pm 0.5</math></b>	$13.02 \pm 2.58$	<b>0.0</b>

Kết quả của bài benchmark Ackley tương tự như với Griewank, khác biệt ở điểm, với Ackley, CEM vẫn có xu hướng cải thiện tốt hơn tuy vẫn không thể hội tụ và với số chiều bằng 10 thì CEM mặc dù càng bất ổn định nhưng có xu hướng cho ra kết quả tốt hơn so với thực nghiệm với số chiều bằng 2.



### 3. Kết luận

Dựa trên các kết quả thực nghiệm, báo cáo cho thấy DE thường vượt trội hơn CEM trong việc giải quyết các bài toán tối ưu hóa trên các hàm benchmark được khảo sát. DE thể hiện khả năng hội tụ nhanh hơn, đạt độ chính xác cao hơn và duy trì tính ổn định tốt hơn, đặc biệt trên các hàm phức tạp và khi tăng số chiều của bài toán. Trong khi CEM gặp nhiều khó khăn, đặc biệt trên các hàm đa cực trị, và hiệu suất của nó thường kém ổn định hơn so với DE. Kết quả kiểm định thống kê cũng củng cố nhận định về sự khác biệt có ý nghĩa giữa hiệu suất của hai thuật toán, nghiêng về lợi thế của DE trong hầu hết các trường hợp. Tóm lại, Differential Evolution là một phương pháp tối ưu hóa hiệu quả và mạnh mẽ hơn Cross-Entropy Method đối với các bài toán đã được thử nghiệm.

## 4. Minh họa

- Tập minh họa GIF: [Đường dẫn](#).