

Trong bài tập này, chúng ta cài đặt Differential Evolution (DE) và Cross Entropy Method (CEM) phiên bản cải tiến để tối ưu hóa các hàm mục tiêu sau:

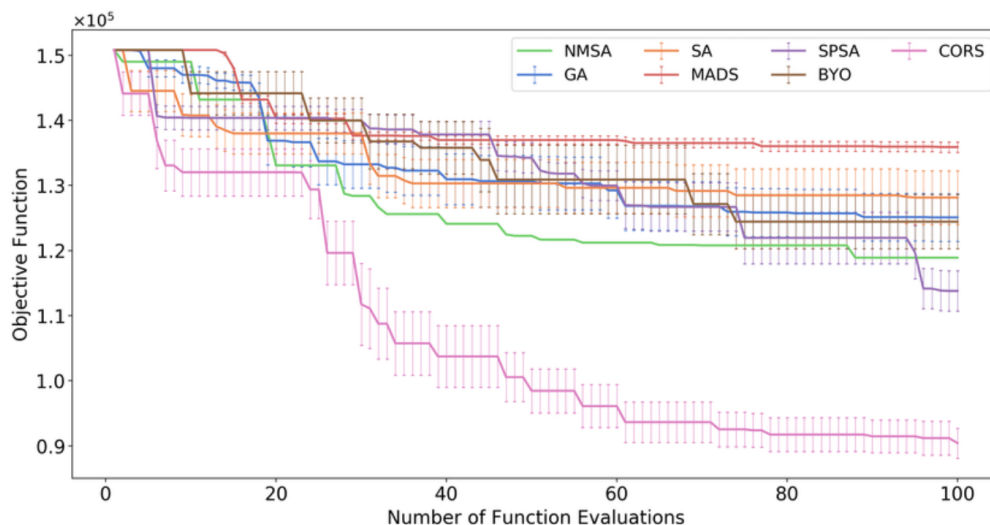
1. f1: Sphere (d = 2 biến và 10 biến).
2. f2: Zakharov (d = 2 biến và 10 biến).
3. f3: Rosenbrock (d = 2 biến và 10 biến)
4. f4: Michalewicz (d = 2 biến và 10 biến).
5. f5: Ackley (d = 2 biến và 10 biến).

Thông tin về các hàm này: công thức hàm mục tiêu (objective function), miền giá trị (search domain), cực trị toàn cục (global minima), các bạn tham khảo trong link sau đây:

<https://www.sfu.ca/~ssurjano/optimization.html>

Các bạn thực hiện thí nghiệm như sau:

- Kích thước quần thể N (hoặc λ) = 32, 64, 128, 256, 512, 1024
- Với mỗi trường hợp (f,d,N), ta cần chạy thực nghiệm 10 lần, dùng các random seed là từ MSSV \rightarrow MSSV+9.
- Mỗi lần chạy thực nghiệm thì thuật toán dừng lại ngay sau khi 100,000 lần (đối với d=2) hoặc 1,000,000 lần (đối với d=10) hàm đánh giá fitness evaluation được gọi thực hiện. Các bạn lưu lại thông tin về **lời giải tốt nhất $best_i$** và **giá trị hàm mục tiêu $fitnes(best_i)$** mà DE và CEM tìm ra **tại mỗi thế hệ thứ i** và **số lần gọi hàm đánh giá** (num_of_evaluations) đã sử dụng tính từ đầu lần chạy thực nghiệm cho tới cuối thế hệ thứ i này.
- **Với mỗi tổ hợp (f,d)**, hãy vẽ 1 đồ thị hội tụ (convergence graph): trục tung **giá trị hàm mục tiêu $fitnes(best_i)$** đã lưu như trên, trục hoành là **số lần gọi hàm đánh giá** (từ 0 tới 100,000 hoặc 1,000,000) giống như hình minh họa sau đây. Lưu ý: chỉ cần chọn 2 trường hợp N=128 và 1024 của DE và CEM để vẽ cho dễ nhìn. Đường thẳng ở giữa là giá trị trung bình tính trên 10 lần chạy thực nghiệm, và error bars là ± 1 std. Có thể vẽ error bar bằng fill_between.



- Các bạn lập một bảng thống kê cho kết quả sau cùng như sau **cho mỗi tổ hợp (f,d)**.

Popsize N / λ	DE	CEM
32	Mean (std)	Mean (std)
64	Mean (std)	Mean (std)
128	Mean (std)	Mean (std)
256	Mean (std)	Mean (std)
512	Mean (std)	Mean (std)
1024	Mean (std)	Mean (std)

Mean: Trung bình objective value của lời giải tốt nhất trên 10 lần chạy ứng với tổ hợp (f,d) tương ứng.

std: Độ lệch chuẩn của objective value của lời giải tốt nhất trên 10 lần chạy ứng với tổ hợp (f,d) tương ứng.

Cần kiểm tra ý nghĩa thống kê t-test các kết quả và tô đậm kết quả tốt hơn (có giá trị mean nhỏ hơn) trong mỗi trường hợp.

- Đối với các trường hợp d=2, các bạn tìm hiểu cách làm file gif tạo animation cho chuyển động của quần thể qua các thế hệ từ đầu tới thế hệ cuối cùng (cho DE và CEM) với contour của hàm mục tiêu cho lượt chạy có random seed là MSSV và popsize=32. Nhớ đánh dấu vị trí của lời giải tối ưu (global optimum) thực sự của mỗi hàm mục tiêu để xem khi sử dụng hết 100,000 lần gọi hàm đánh giá thì quần thể còn cách lời giải tối ưu bao xa. Mỗi trường hợp chỉ cần tạo file gif cho 1 seed. **Đặt tên file gif cho đúng với các trường hợp. Cho các file gif vào một thư mục drive và nộp đường link đến thư mục drive này).**

Nêu các kết luận ngắn gọn về hiệu năng (performance) của DE và CEM dựa trên kết quả thực nghiệm.

Trong các log files cần chứa thông tin về random seed của lần chạy, lời giải tốt nhất DE và CEM tìm được trong lần chạy đó là gì, giá trị hàm mục tiêu (objective value) là bao nhiêu?

Các bạn để log files, file báo cáo và source code vào 1 thư mục, zip lại và nộp file zip vào link nộp trên trang courses.