

BÁO CÁO HỌC THUẬT

ỨNG DỤNG TRÍ TUỆ NHÂN TẠO (AI) ĐIỀU KHIỂN CÁC ĐỐI TƯỢNG TRONG CÔNG NGHIỆP

Người báo cáo: TS. Đặng Văn Chí
Bộ môn Tự động hóa XN Mở - Dầu khí

Hà nội, ngày 05/2026

Thiết kế, xây dựng và phát triển các thuật toán về AI Controller, Optimization, Identification ...

- 1- Fuzzy Logic (FL)
- 2- Deep Learning & Neural Network (DL-NN)
- 3- Genetic Algorithm (GA)
- 4- Particle Swarm Optimization (PSO)

GIỚI THIỆU VỀ GIẢI THUẬT DI TRUYỀN

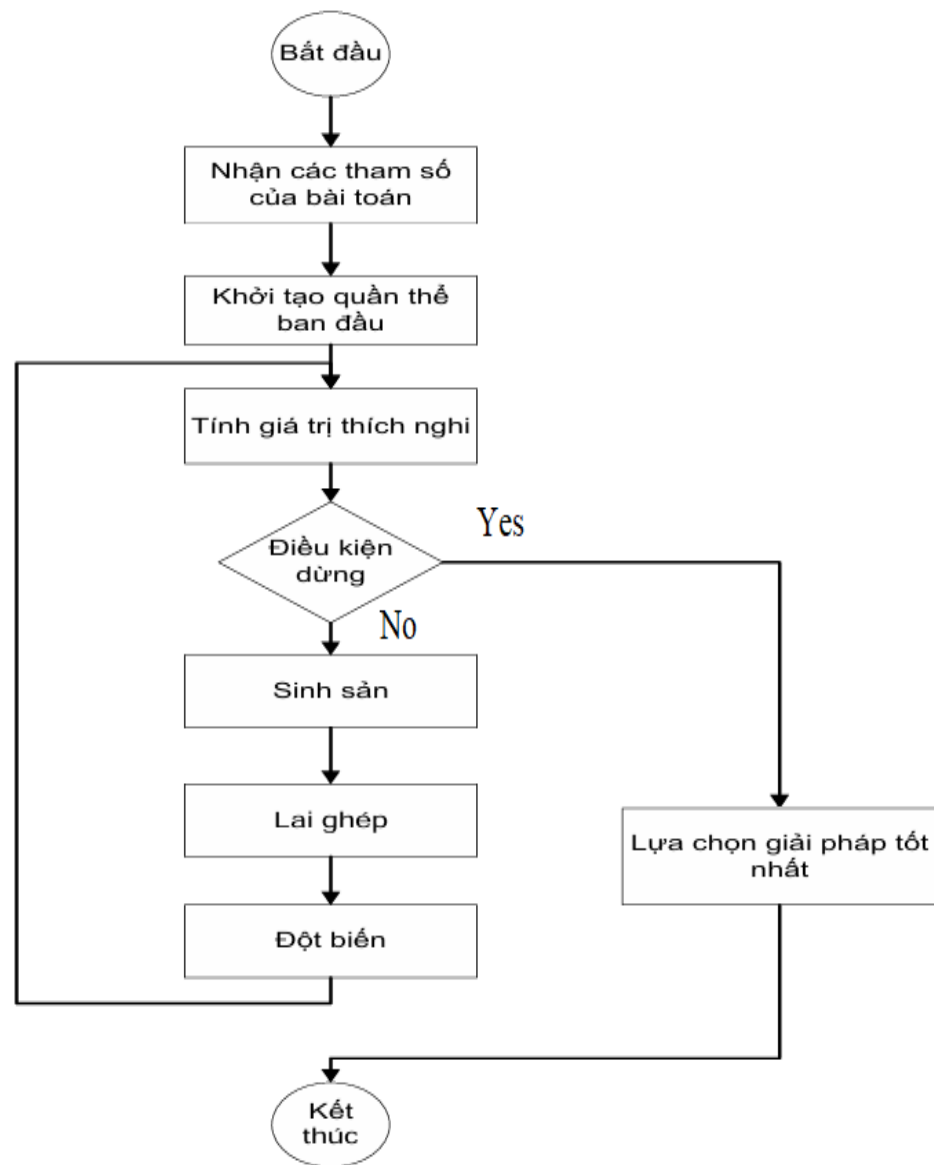
Genetic Algorithms (GA)

- Genetic Algorithms (GA) dựa trên quá trình thích nghi tiến hóa của các quần thể sinh học dựa trên học thuyết của Darwin.
- Nguyên lý cơ bản của GA được tác giả J.H.Holland công bố đầu tiên vào năm 1962. Các nền tảng toán học của GA lần đầu tiên được công bố vào năm 1975. Có thể nói J.H.Holland là người đi tiên phong nghiên cứu trong lĩnh vực GA.
- GA vận dụng các nguyên lý: di truyền, đột biến, chọn lọc tự nhiên và trao đổi chéo. GA dùng một số thuật ngữ của ngành di truyền học như: nhiễm sắc thể (Chromosomes), quần thể (Population), Gene ...
- Ngày nay GA càng trở nên quan trọng, đặc biệt là trong lĩnh vực tối ưu hoá, được ứng dụng nhiều trong thực tiễn, điều khiển công nghiệp ...

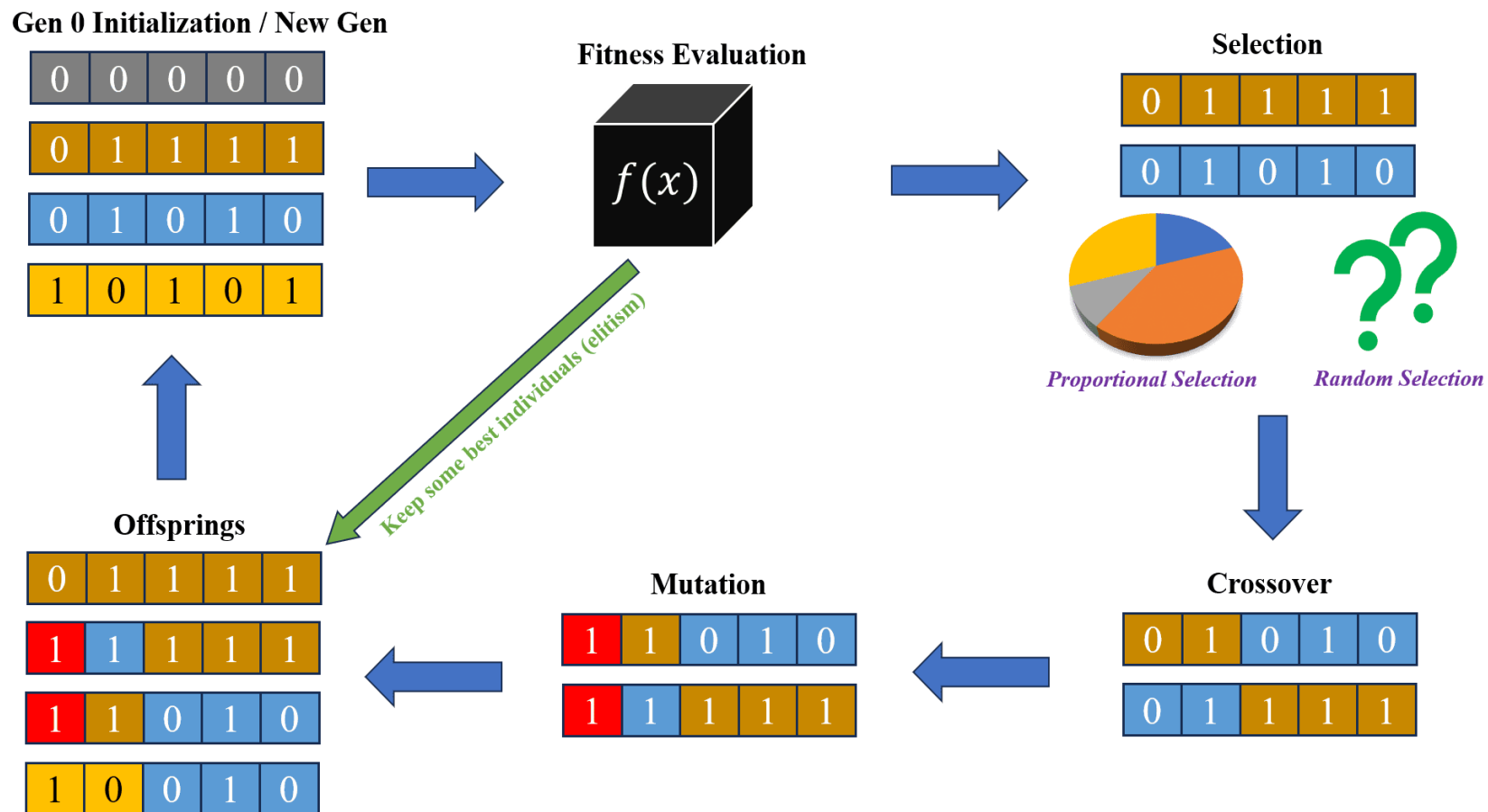
Một số ứng dụng của Giải thuật di truyền - GA

- Học máy (Machine Learning): tối ưu mô hình, nhận dạng hệ thống, điều khiển đối tượng, các bài toán tối ưu kinh doanh – tài chính ...
- Thiết kế mạng nơron (kiến trúc mạng và ma trận trọng số).
- Các bài toán chuyển động quỹ đạo cho Robot.
- Các hệ phi tuyến động học, dự báo, phân tích dữ liệu.
- Lập lịch (Scheduling): lập lịch làm việc, lịch sản xuất, tối ưu ca trực.
- Tối ưu hóa thiết kế: trong kỹ thuật cơ khí, thiết kế mạch điện...
- Cải tiến chương trình LISP (lập trình gen)...

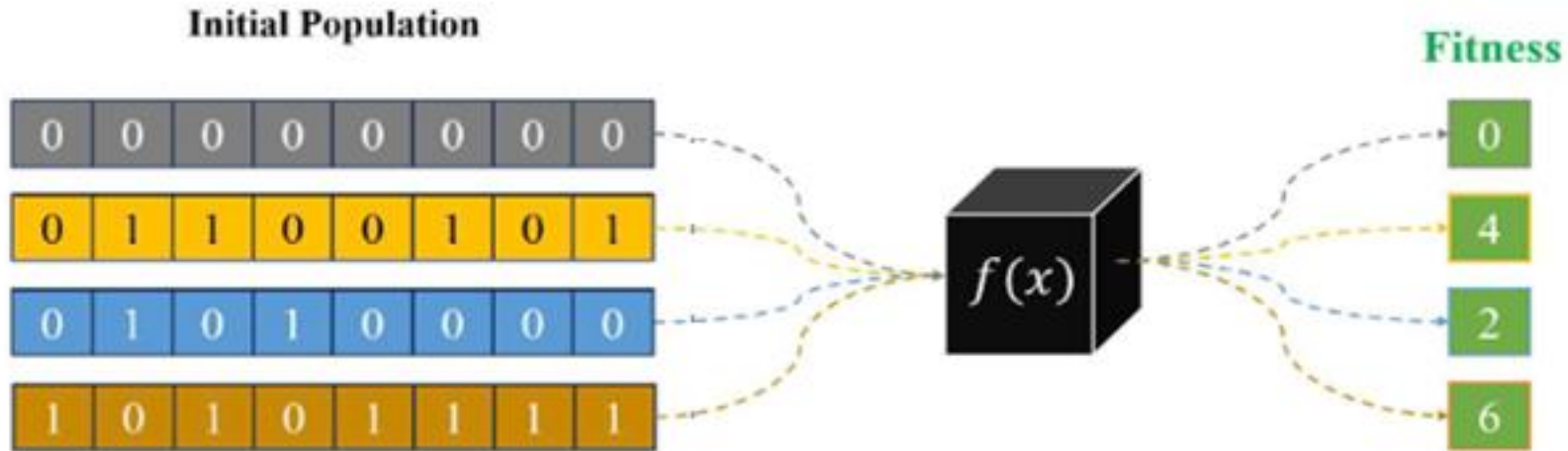
Sơ đồ thực hiện giải thuật di truyền - GA



Lưu đồ tổng quan của giải thuật di truyền GA

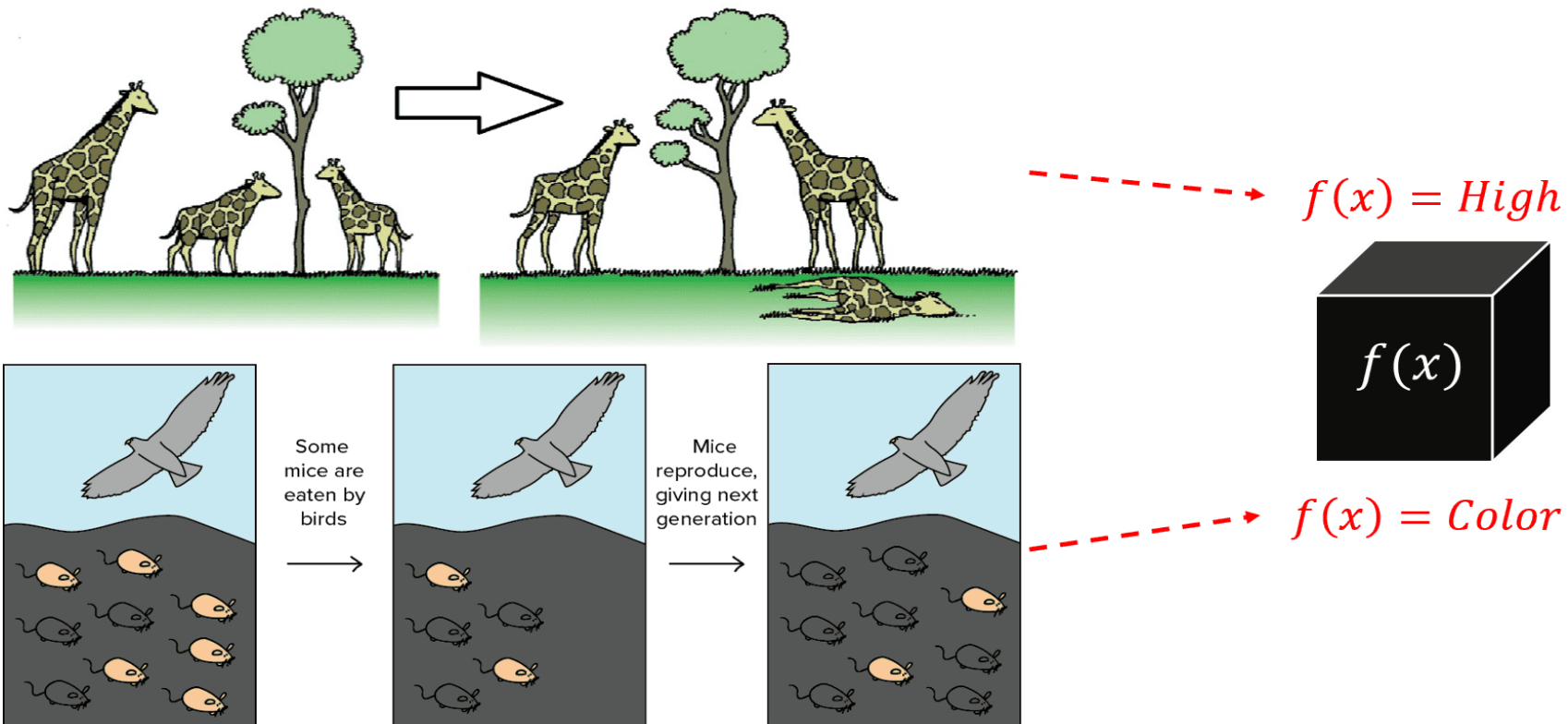


Hàm đánh giá độ thích nghi (Fitness Evaluation)



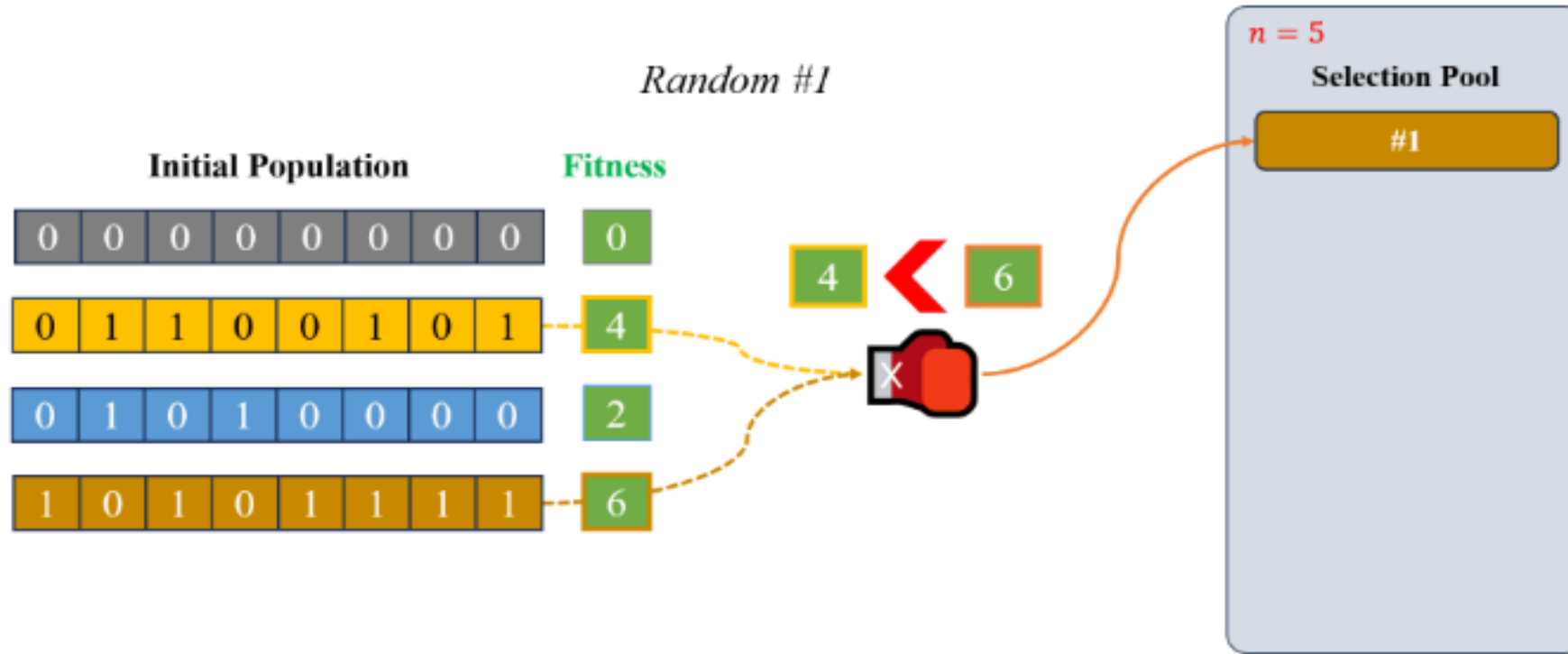
Để biết được bộ GEN nào có thể thích nghi tốt với môi trường, cần một hàm để đánh giá gọi là Fitness Evaluation

Minh họa quá trình chọn lọc tự nhiên



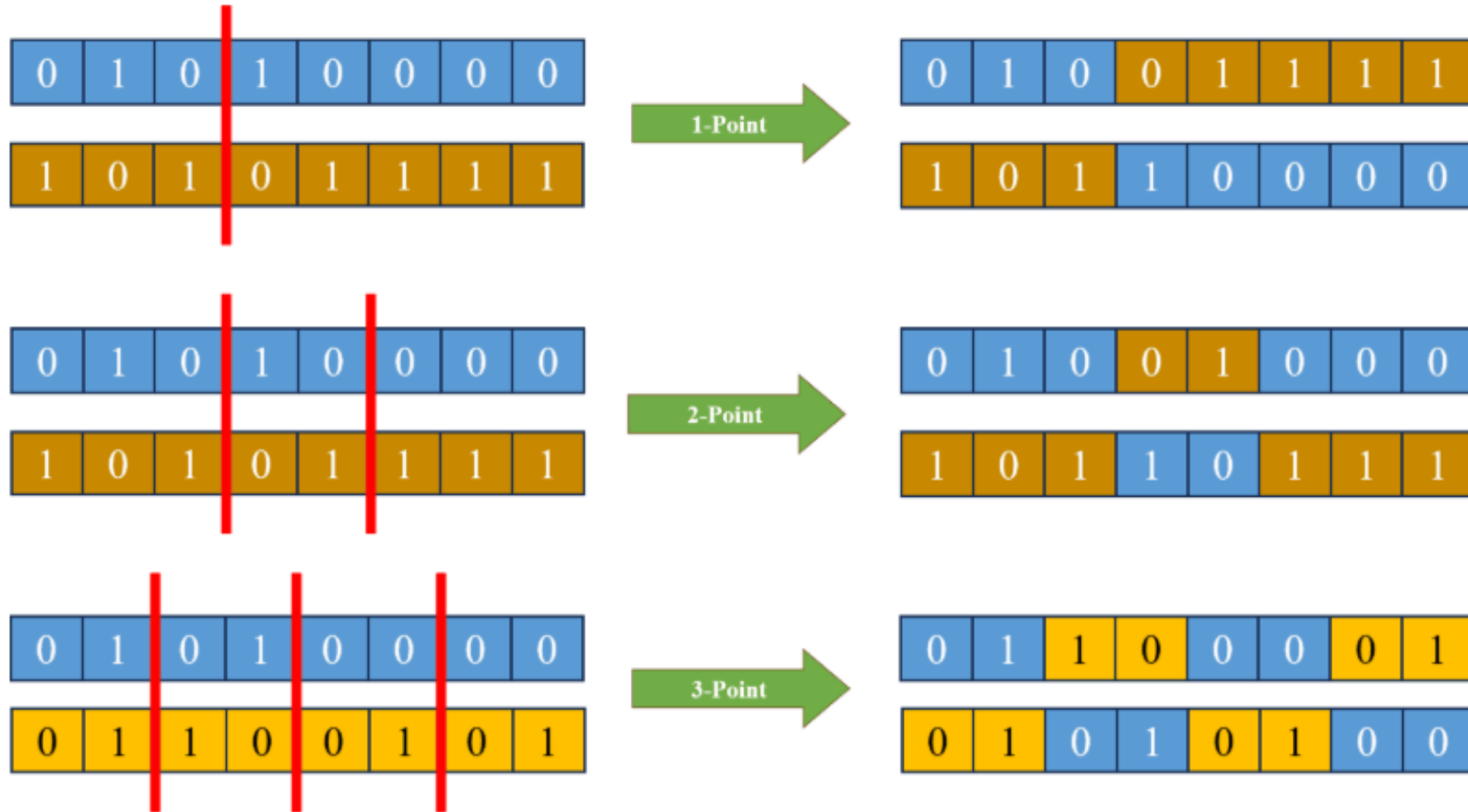
Quá trình chọn lọc cá thể ngẫu nhiên (Selection)

(Fitness(vàng=4) $\leq$ Fitness(cam=6))



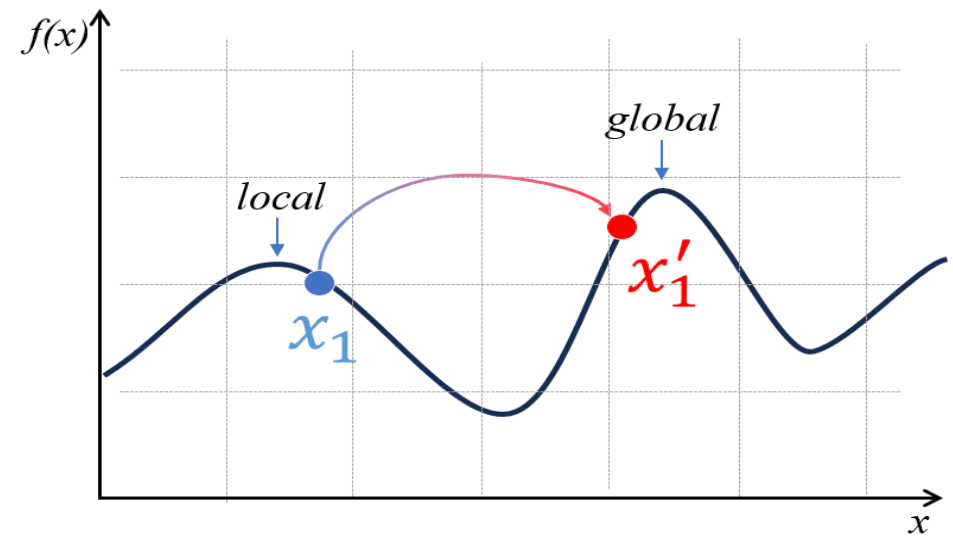
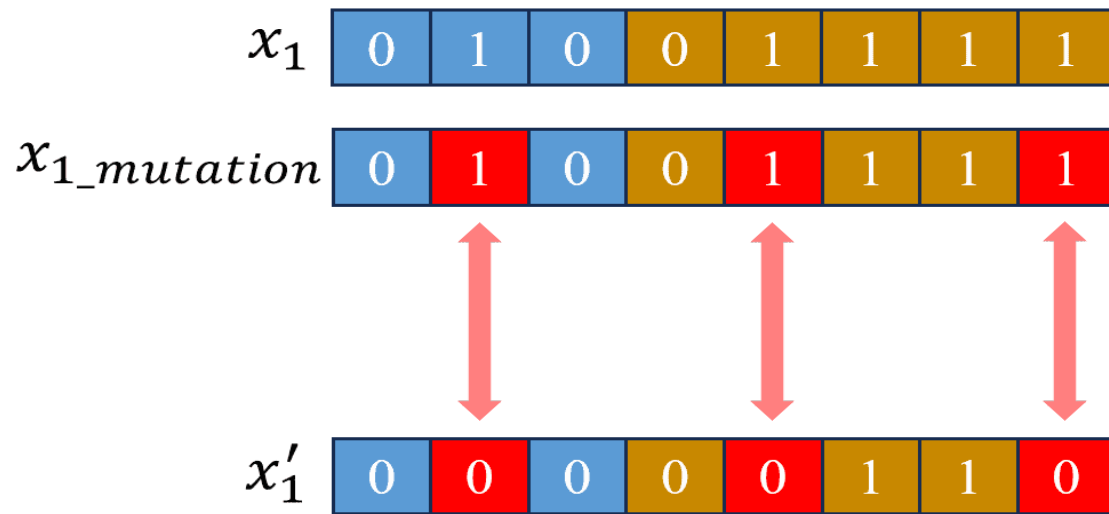
Chọn từng cặp cá thể một cách ngẫu nhiên, cá thể nào có giá trị Fitness lớn hơn (thích nghi tốt hơn) sẽ là cá thể ưu tú và được tham gia vào quá trình lai tạo và đột biến ở các bước tiếp theo.

Quá trình lai ghép (Crossover)



Tiến hành lai tạo các cặp cá thể với nhau để cho ra các cá thể con mới có thể kế thừa các Gen - đặc tính tốt từ cha mẹ

Quá trình đột biến (Mutation)



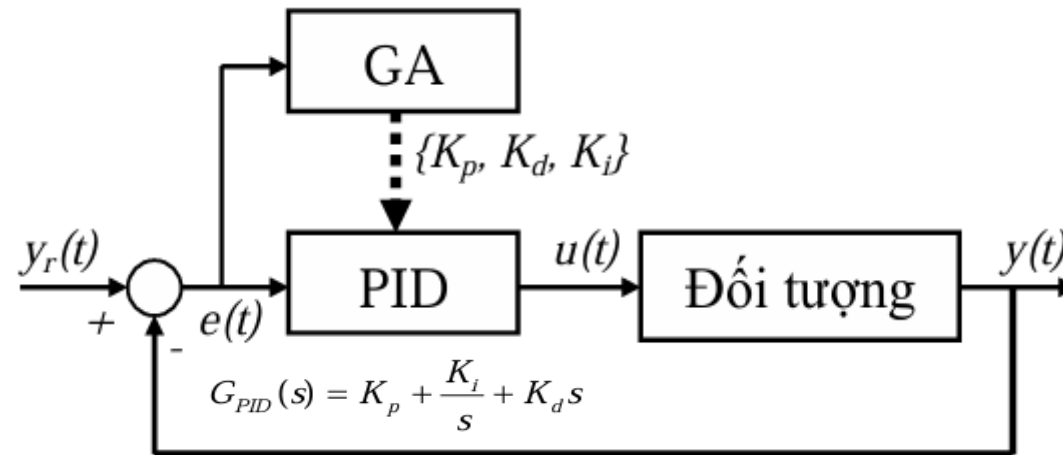
Sự đột biến cho cá thể con mới được tạo ra nhằm tạo sự đa dạng trong quần thể và tránh bị mắc kẹt ở các cực trị cục bộ

Điều kiện dừng (Termination) của GA

Để kết thúc giải thuật GA, quyết định xem có tiếp tục tối ưu nữa không, một số tiêu chí để dừng thuật toán là:

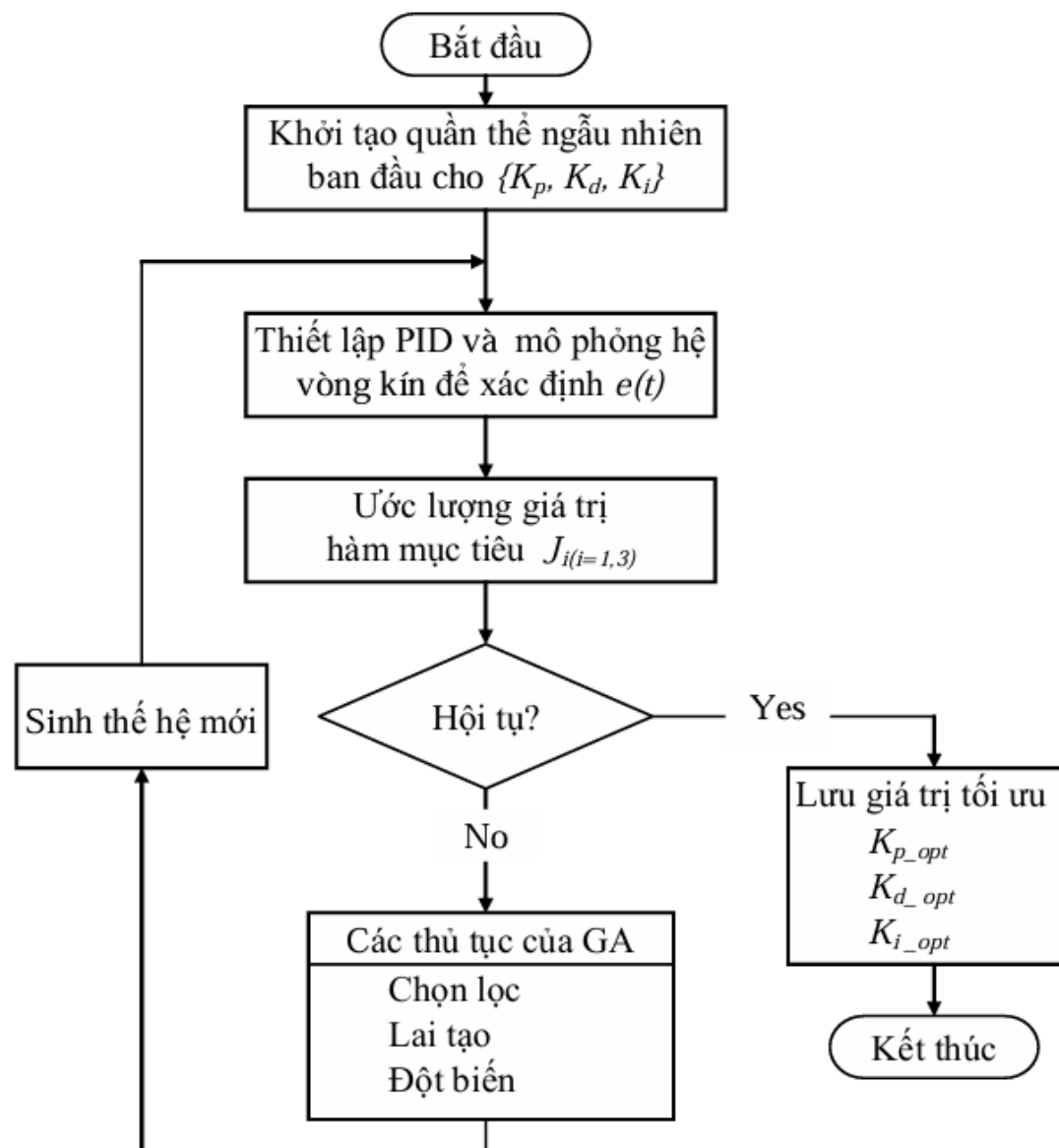
- + **Số thế hệ tối đa:** Dừng khi đã chạy một số lượng thế hệ nhất định.
- + **Đặt ngưỡng cho Fitness:** Dừng khi chất lượng của cá thể tốt nhất trong quần thể đạt đến một ngưỡng nhất định.
- + **Không có cải thiện:** Dừng lại khi không có sự cải thiện nào trong chất lượng của cá thể tốt nhất sau một số thế hệ liên tiếp.

ỨNG DỤNG GA ĐỂ TỐI ƯU HÓA CÁC THAM SỐ BỘ ĐIỀU KHIỂN PID TRONG CÔNG NGHIỆP



Sơ đồ nguyên lý tổng quát bộ điều khiển PID ứng dụng GA

Lưu đồ tiến trình GA xác định thông số bộ điều khiển PID



Hàm mục tiêu

Trong điều khiển vòng kín, $e(t)$ là sai lệch giữa tín hiệu đặt $y_r(t)$ và đáp ứng $y(t)$ thì:

$$e(t) = y_r(t) - y(t)$$

Các hàm mục tiêu của quá trình tinh chỉnh bộ điều khiển được định nghĩa như sau:

$$IAE: J_1 = \int_0^{\infty} |e(t)| dt$$

$$IATE: J_2 = \int_0^{\infty} t |e(t)| dt$$

$$MSE: J_3 = \frac{1}{N} \sum_{i=1}^N e_i^2(t)$$

Nhiệm vụ của giải thuật GA là tìm kiếm các giá trị $\{K_{p_opt}, K_{d_opt}, K_{i_opt}\}$ tối ưu của bộ điều khiển PID, mà ở đó các hàm J_i ($i=1,3$) đạt giá trị cực tiểu

Code m-file/Matlab thực hiện thuật toán GA

1. Hàm đánh giá (fitness_pid.m)

```
function cost = fitness_pid(params)
    % Trích xuất các tham số từ biến đầu vào của GA
    Kp = params(1);
    Ki = params(2);
    Kd = params(3);

    % Định nghĩa hàm truyền của đối tượng (Plant)
    % Ví dụ:  $G(s) = 1 / (s^2 + 10s + 20)$ 
    s = tf('s');
    G = 1 / (s^2 + 10*s + 20);

    % Tạo bộ điều khiển PID
    C = pid(Kp, Ki, Kd);

    % Hệ thống vòng kín
    T = feedback(C*G, 1);

    % Kiểm tra tính ổn định (Tránh trường hợp hệ thống mất ổn định)
    if isstable(T)
        [y, t] = step(T);
        error = abs(1 - y); % Sai số so với tín hiệu bước nhảy (step =
1)

        % Tính chỉ số ITAE: Tích phân của (thời gian * trị tuyệt đối sai
số)
        cost = sum(t .* error);
```

2. Script chính (main_ga.m)

```
clear; clc;

% 1. Cấu hình bài toán GA
num_vars = 3; % Số lượng biến cần tìm: Kp, Ki, Kd

% Giới hạn dưới và trên cho [Kp, Ki, Kd]
lb = [0, 0, 0];
ub = [100, 100, 100];

% 2. Cấu hình các tham số của GA
options = optimoptions('ga', ...
    'PopulationSize', 50, ... % Kích thước quần thể
    'MaxGenerations', 25, ... % Số thế hệ tối đa
    'Display', 'iter', ... % Hiển thị quá trình tính toán
    'PlotFcn', @gaplotbestf); % Vẽ biểu đồ quá trình tối ưu

% 3. Chạy Thuật toán Di truyền
fprintf('Đang bắt đầu quá trình tiến hóa... Chờ quá trình chạy!\n');
[best_params, min_cost] = ga(@fitness_pid, num_vars, [], [], [], [],
lb, ub, [], options);
```

Kết quả chạy M-file của thuật toán GA

Command Window

Dang bat dau qua trinh tien hoa - cho mot chut!

Generation	Func-count	Best f(x)	Mean f(x)	Stall Generations
1	100	3.649	2024	0
2	150	3.649	2053	1
3	200	3.645	309.9	0
4	250	3.11	96.39	0
5	300	3.044	61.27	0
6	350	3.044	25.73	1
7	400	3.044	12.64	2
8	450	2.893	4.813	0

Command Window

21	1100	2.395	2.661	0
22	1150	2.395	2.647	1
23	1200	2.381	2.619	0
24	1250	2.369	2.594	0
25	1300	2.331	2.568	0

Optimization terminated: maximum number of generations exceeded.

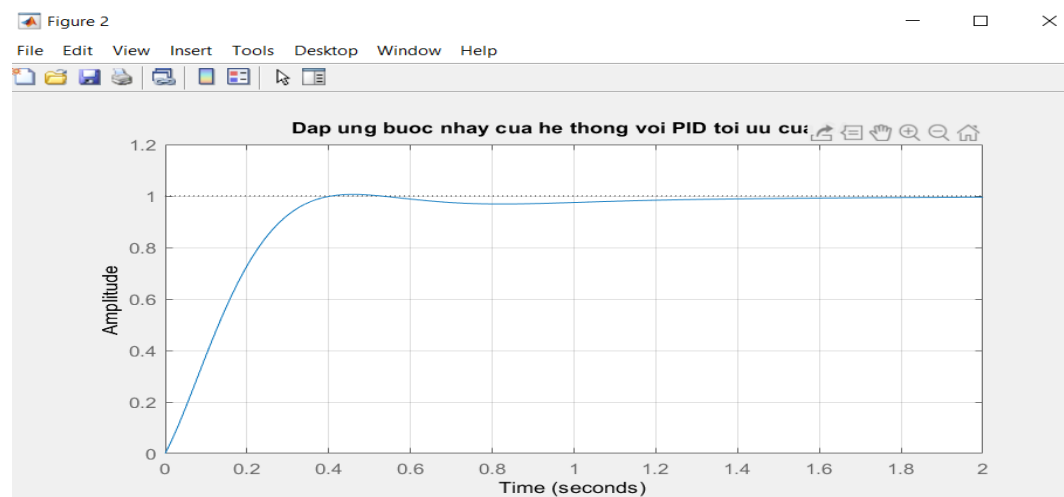
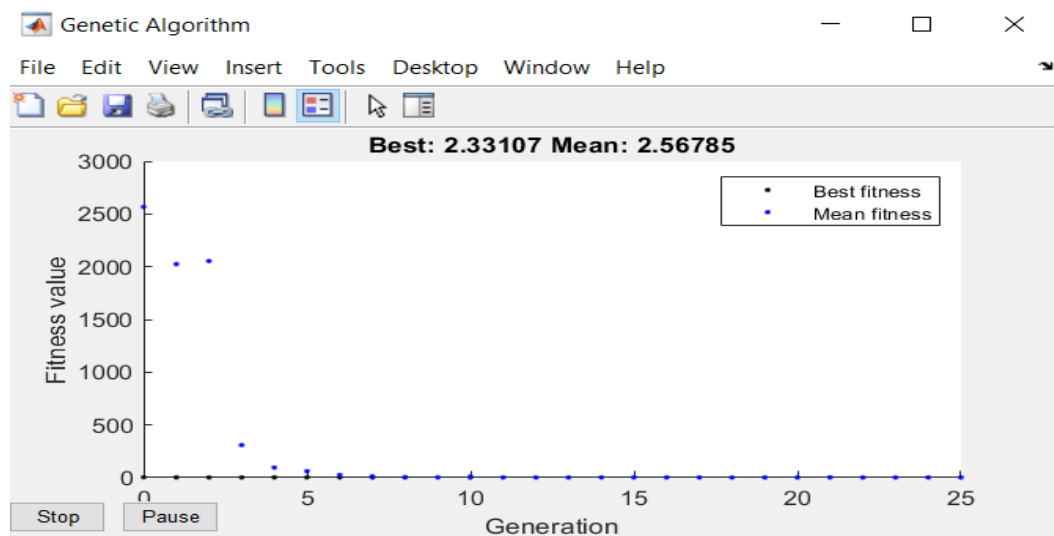
Ket qua toi uu tim duoc:

$K_p = 71.5240$

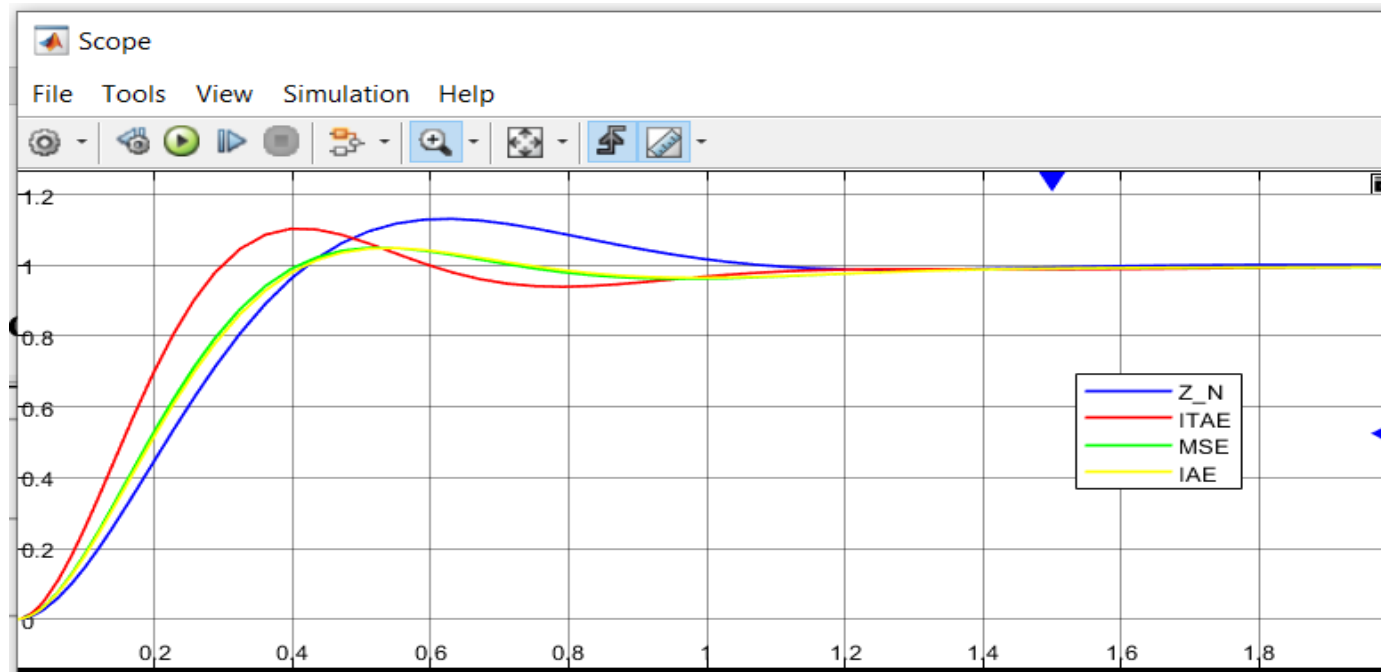
$K_i = 118.6150$

$K_d = 2.9634$

Gia tri ham muc tieu (ITAE) nho nhat: 2.3311



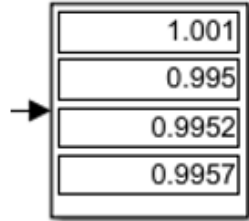
Đáp ứng bước đơn vị với 4 bộ điều khiển



Bảng so sánh bộ điều khiển PID theo các tiêu chí thiết kế

PID	Z_N	GA_ITAE	GA_MSE	GA_IAE
Kp	37.8	71.524	49.9872	48.3834
Ki	114.7	118.615	94.4264	93.6075
Kd	2.438	2.9634	0.0927	0.0185

Bảng so sánh chỉ tiêu chất lượng điều khiển

Tiêu chí đánh giá	$PID_{(Z_N)}$	$PID_{(GA_ITAE)}$	$PID_{(GA_MSE)}$	$PID_{(GA_IAE)}$
Độ quá điều chỉnh (%)	12	10	5	4
Sai lệch tĩnh (%) 	± 0.001	± 0.005	± 0.0048	± 0.0043
Thời gian tăng (s)	0.48	0.28	0.35	0.37
Thời gian xác lập (s)	1.83	1.25	1.41	1.42

KẾT LUẬN

1. Giải thuật GA đã được ứng dụng để tìm kiếm giá trị tối ưu của bộ điều khiển PID, thỏa mãn các hàm mục tiêu IAE, ITAE và MSE.
2. Ưu điểm của phương pháp này là thiết kế được bộ điều khiển PID cận tối ưu mà không cần quan tâm đến mô hình toán của đối tượng.
3. Để có thể ứng dụng được trong thực tế, hệ thống cần được trang bị các Sensor đo tín hiệu đáp ứng. Khi đó, việc thiết kế bộ điều khiển theo thời gian thực trở nên khả thi bằng cách kết nối máy tính vào các thiết bị đo này.

Trân trọng cảm ơn!