

2.7. Xác định tham số PID sử dụng trí tuệ nhân tạo

Ưu khuyết điểm của điều khiển thông thường

* Ưu điểm:

- Có tính hệ thống, cơ sở toán học rõ ràng, chặt chẽ.
- Đảm bảo hệ thống ổn định và bền vững (về lý thuyết)

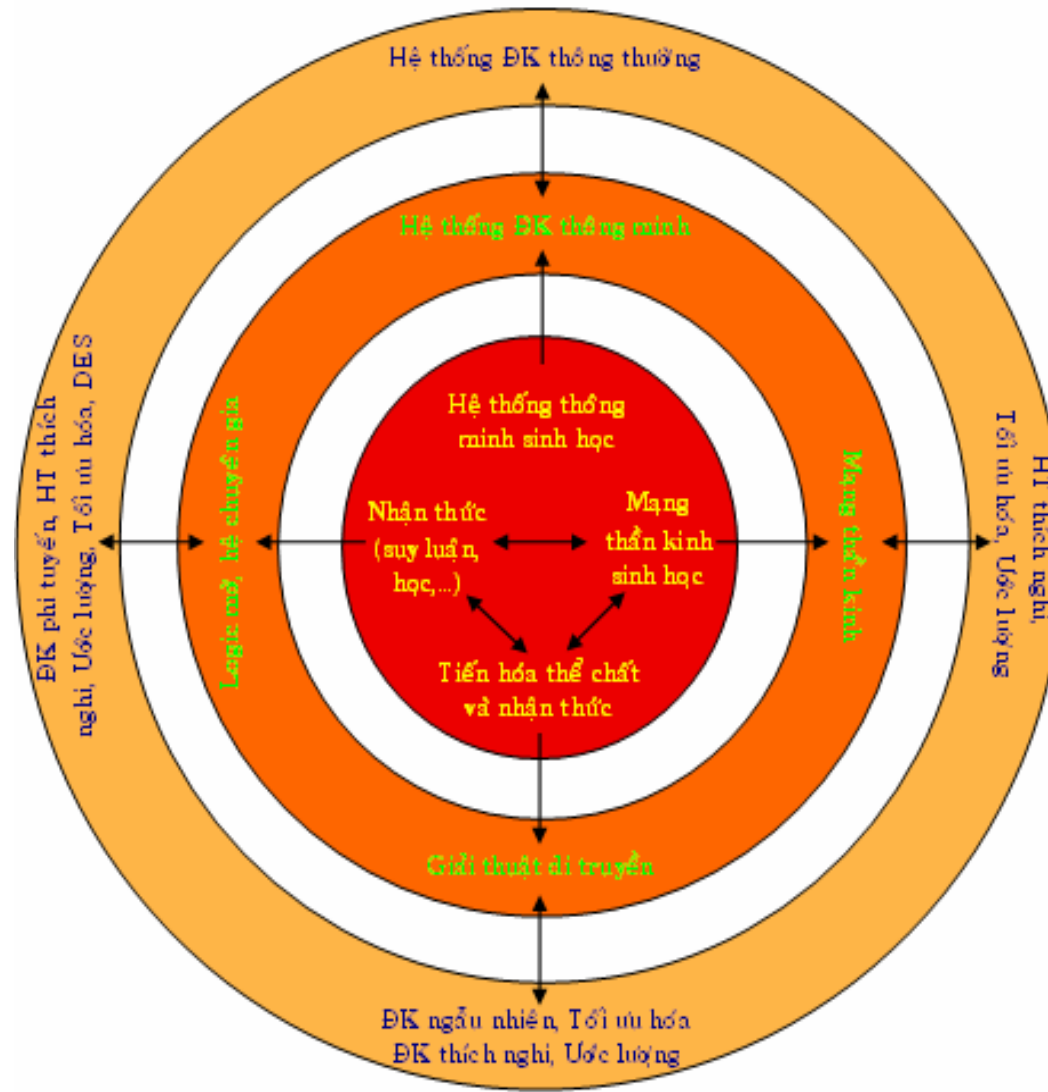
* Khuyết điểm:

- Cần mô hình toán học của đối tượng để thiết kế được bộ điều khiển
- Cần hiểu biết sâu về kỹ thuật điều khiển mới thiết kế được bộ điều khiển
- Thường không hiệu quả khi điều khiển hệ phi tuyến
- Không sử dụng được kinh nghiệm của con người (trong nhiều trường hợp kinh nghiệm của con người đóng vai trò quan trọng)

Tại sao phải điều khiển thông minh

- Yêu cầu đạt được chất lượng điều khiển ngày càng tăng cao.
 - Yêu cầu điều khiển các hệ thống động phức tạp ngày càng tăng.
 - Yêu cầu điều khiển trong điều kiện gia tăng các yếu tố bất định.
- Các yêu cầu trên không những không thể đáp ứng được trọn vẹn nếu dùng lý thuyết điều khiển thông thường sẵn có. Đây chính là động lực cho ra đời lý thuyết điều khiển mới: Lý thuyết điều khiển thông minh

Các phương pháp điều khiển thông minh



Giải thuật di truyền - GA

Thuật toán này có các bước cơ bản sau:

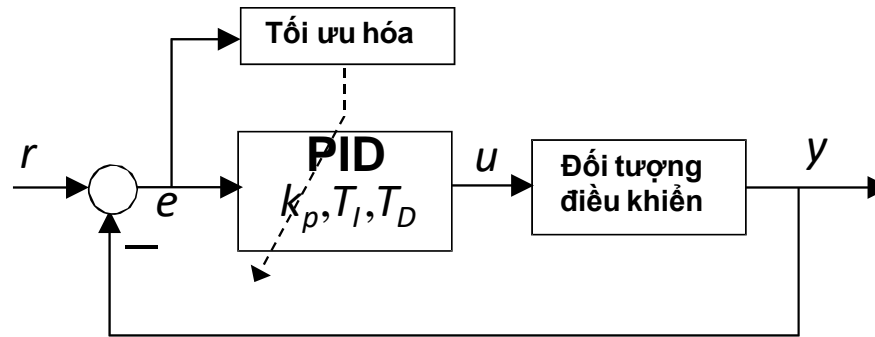
1. Chọn ngẫu nhiên $p_i(0), i = 1, 2, \dots, N$ phần tử ban đầu, gọi là cá thể khởi tạo, và ký hiệu tập các cá thể đó là thế hệ khởi tạo (initial generation) $G(0)$, gán $k=0$.
2. Gán cho mỗi cá thể $p_i(k)$, một chỉ số xác suất π_i được tính từ giá trị hàm mục tiêu tại đó: _

$$\pi_i = 1 - J(p_i) / \bar{J} \quad \text{Với } \bar{J} = \frac{1}{N} \sum_{j=1}^N J(p_j)$$

3. Nếu điều kiện kết thúc thuật toán được thỏa mãn thì chọn cá thể có π_i lớn nhất làm nghiệm. Ngược lại thì chuyển sang bước 4.
4. Sao chép $G(k)$ vào tập trung gian $I(k)$ theo tỷ lệ xác suất của từng cá thể.
5. Tiến hành việc lai ghép từng cặp cá thể được chọn ngẫu nhiên trong $I(k)$ bằng cách nếu cặp cá thể của cặp đó có xác suất lớn $\pi_i \geq \pi_c$ thì hoán đổi nửa dưới trong dãy nhị phân biểu diễn giá trị của cặp hai cá thể đó. Những cặp có xác suất nhỏ $\pi_i \leq \pi_m$ thì đổi giá trị bit 0,1 trong dãy nhị phân của cá thể đó (đột biến). Hai giá trị π_c, π_m được cho trước.
6. Sao chép $I(k)$ vào $G(k+1)$. Gán $k=k+1$ và quay về 2.

Chọn tham số tối ưu cho bộ điều khiển PID

Nguyên tắc chung



Xác định được vector tham số $p = (k_p, T_I, T_D)^T$ cho bộ điều khiển PID để hệ kín bám ổn định theo được tín hiệu mẫu. Nguyên tắc xác định này là phải cực tiểu được sai lệch bám:

$$J(p) = \int_0^T e^2(p, t) dt \xrightarrow{p \in P} \min \text{ hoặc } J(p) = \int_0^T |e(p, t)| dt \xrightarrow{p \in P} \min$$

Giá trị của sai lệch bám $e(p, t)$, bên cạnh việc điều khiển phụ thuộc vào đặc tính động học của đối tượng, còn phụ thuộc vào bộ điều khiển. Khó khăn chính là nằm ở việc xác định được công thức tường minh cho hàm mục tiêu $J(p)$.

Chọn tham số tối ưu cho bộ điều khiển PID (tiếp)

Ví dụ 1: Ứng dụng GA chọn tham số PI

Đối tượng điều khiển giả định có hàm truyền:

$$s(s) = \frac{b_0s + b_1}{a_0s^2 + a_1s + a_2} = \frac{s + 5}{s^2 + 3s + 5}$$

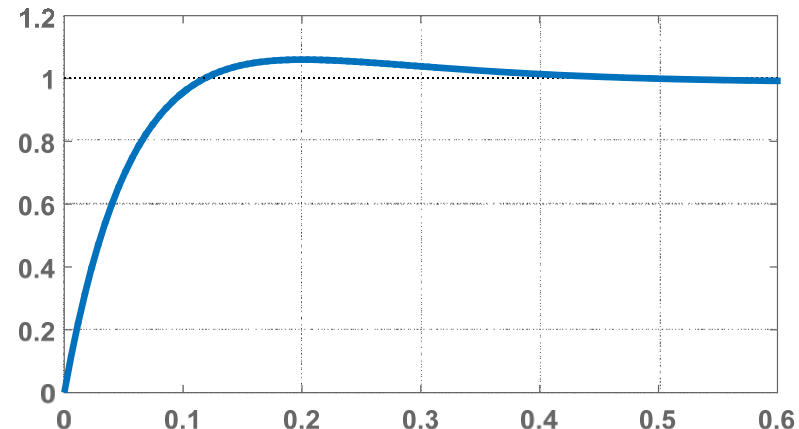
Bộ điều khiển là PI với hàm truyền: $c(s) = k_p + \frac{k_i}{s} = \frac{k_p s + k_i}{s}$

Sử dụng GA để xác định tham số tối ưu $0 \leq k_p, k_i$ được kết quả sau:

Tham số PI: $k_p = 16.3451, k_i = 26.4433$

– Giá trị hàm mục tiêu: $J_{\min} = 0.0077$

– Đồ thị hàm quá độ hệ kín:



Chọn tham số tối ưu cho bộ điều khiển PID (tiếp)

Ví dụ 1: Ứng dụng GA chọn tham số PI (tiếp)

runPI_GA.m

```
clc;  
[x fval] = ga(@PI_GA,2,-diag([1  
1]),zeros(2,1)); kp=x(1);ki=x(2);  
b0=1; b1=5; a0=1; a1=3; a2=5;  
S = tf([b0 b1],[a0 a1 a2]); C = tf([kp  
ki],[1 0]); G = feedback(S*C,1);  
step(G);
```

PI_GA.m

```
function fitness =  
PI_GA(x) kp=x(1);  
ki=x(2);  
b0=1; b1=5; a0=1;a1=3; a2=5;  
S=tf([b0 b1],[a0 a1 a2]); C=tf([kp  
ki],[1 0]); G=feedback(S*C,1); [y  
t]=step(G);  
n=length(y); dt=t(end)/(n-1);  
fitness = 0; for j=1:n-1;  
    fitness = fitness + dt*abs(1-y(j,1));  
end
```


Chọn tham số tối ưu cho bộ điều khiển PID (tiếp)

Ví dụ 2: Ứng dụng GA chọn tham số PID

Đối tượng điều khiển giả định có hàm truyền:

$$S(s) = \frac{b_0s + b_1}{a_0s^3 + a_1s^2 + a_2s + a_3} = \frac{s + 5}{s^3 + 3s^2 + 2s + 1}$$

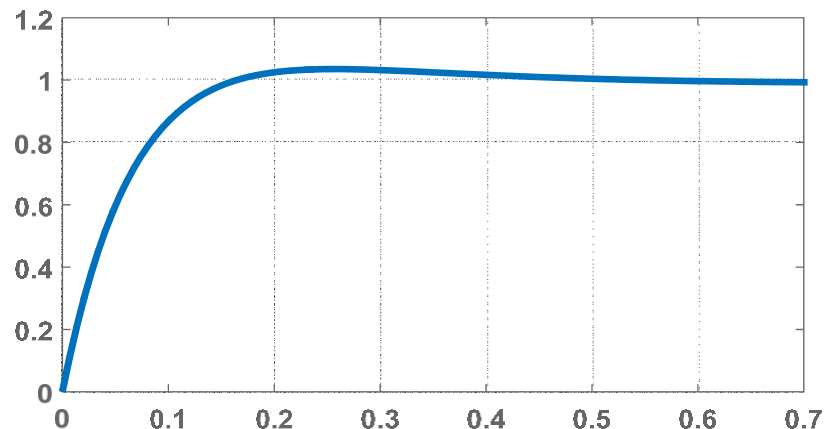
Bộ điều khiển là PI với hàm truyền: $C(s) = k_p + \frac{k_i}{s} + k_d s = \frac{k_d s^2 + k_p s + k_i}{s}$

Sử dụng GA để xác định tham số tối ưu $0 \leq k_p, k_i, k_d$ được kết quả sau:

– PID: $k_p = 0.0002, k_i = 9.2378, k_d = 17.0572$

– Hàm mục tiêu: $J_{\min} = 0.0047$

– Hàm quá độ hệ kín:



Chọn tham số tối ưu cho bộ điều khiển PID (tiếp)

Ví dụ 2: Ứng dụng GA chọn tham số PID (tiếp)

runPID_GA_mod.m

```
clc;
[x fval] = ga(@PID_GA_mod,3,-diag([1 1 1]),zeros(3,1));
kp=x(1);ki=x(2);kd=x(3);
b0=1; b1=5; a0=1;a1=3; a2=2; a3=1;
S=tf([b0 b1],[a0 a1 a2 a3]);
C = tf([kd kp ki],[1 0]); G =
feedback(S*C,1); step(G);
```

PID_GA_mod.m

```
function fitness =
PID_GA_mod(x) kp=x(1);
ki=x(2); kd=x(3);
b0=1; b1=5; a0=1;a1=3; a2=2; a3=1;
S=tf([b0 b1],[a0 a1 a2 a3]); C=tf([kd
kp ki],[1 0]); G=feedback(S*C,1); [y
t]=step(G);
n=length(y); dt=t(end)/(n-1);
fitness = 0; for j=1:n-1;
    fitness = fitness + dt*abs(1-
    y(j,1));
end
```