

Hướng dẫn Phân tích Hệ Phương trình Vi phân và Vẽ Trường Hướng bằng Maple

Mã nguồn Maple đầy đủ

```
restart;
with(LinearAlgebra);
with(DEtools);

% Buoc 1: Khai bao he phuong trinh
f1 := x*(180-3*x-y);
f2 := y*(100-x-2*y);

% Buoc 2: Tim diem can bang (giai f1=0, f2=0)
solve({f1 = 0, f2 = 0}, {x, y})

% Buoc 3: Tinh ma tran Jacobi
J := Matrix([[diff(f1, x), diff(f1, y)], [diff(f2, x), diff(f2, y)]])

% Buoc 4: Liet ke diem can bang
eq_points := [[0, 0], [60, 0], [0, 50], [52, 24]]

% Buoc 5: Phan tich on dinh tai moi diem can bang
for pt in eq_points do
    % Thay gia tri diem vao ma tran Jacobi
    J_pt := eval(J, [x = pt[1], y = pt[2]]);

    % Tinh gia tri rieng
    eig := Eigenvalues(J_pt);

    % Hien thi ket qua
    printf("Tai ~&dstrok; i m ~(%f,%f):\n", pt[1], pt[2]);
    print(J_pt);
    print(eig)
end do

% Buoc 6: Ve truong huong
with(plots);

% Khai bao he phuong trinh vi phan day du
sys := [diff(x(t), t) = x(t)*(180-3*x(t)-y(t)),
        diff(y(t), t) = y(t)*(100-x(t)-2*y(t))]

% Cac dieu kien ban dau de ve nhieu nghiem
ics := [[x(0) = 10, y(0) = 10],
        [x(0) = 30, y(0) = 20],
        [x(0) = 70, y(0) = 10],
        [x(0) = 20, y(0) = 40]]

% Ve truong huong va cac nghiem
phase := DEplot(sys, [x(t), y(t)],
    t = 0 .. 20,
    x = 0 .. 80,
    y = 0 .. 60,
    arrows = THIN,
    dirfield = 500,
    ics,
    % Thoi gian mo phong
    % Mien hien thi theo x
    % Mien hien thi theo y
    % Mui ten manh
    % Mat do mui ten
    % Dieu kien ban dau
```

```

color = blue,           % Mau duong cong
scene = [x(t), y(t)],   % Mat phang pha
title = "Tr ñng ñ h ñ c a ñ h ñ x' ñ= ñ x(180-3x-y), ñ y' ñ= ñ y(100-x-2y)"

% Buoc 7: Ve diem can bang
eq_points_plot := pointplot([[0, 0], [60, 0], [0, 50], [52, 24]],
                             symbol = solidcircle, % Hinh tron dac
                             color = red,          % Mau do
                             symbolsize = 18);      % Kich thuoc lon

% Buoc 8: Hien thi ket qua
display([phase, eq_points_plot])

```

Listing 1: Code Maple phân tích hệ phương trình vi phân

Giải thích từng phần

1. Khởi tạo và định nghĩa hệ phương trình

- `restart`: Xóa bộ nhớ Maple
- `with(LinearAlgebra)`: Nạp gói đại số tuyến tính
- `with(DEtools)`: Nạp gói công cụ phương trình vi phân
- `f1, f2`: Định nghĩa vế phải của hệ phương trình:

$$\begin{cases} \frac{dx}{dt} = f_1(x, y) = x(180 - 3x - y) \\ \frac{dy}{dt} = f_2(x, y) = y(100 - x - 2y) \end{cases}$$

2. Tìm điểm cân bằng

Giải hệ phương trình:

$$\begin{cases} f_1(x, y) = 0 \\ f_2(x, y) = 0 \end{cases}$$

để tìm các điểm cân bằng của hệ.

3. Ma trận Jacobi

Tính ma trận Jacobi:

$$J(x, y) = \begin{bmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} \end{bmatrix} = \begin{bmatrix} 180 - 6x - y & -x \\ -y & 100 - x - 4y \end{bmatrix}$$

4. Điểm cân bằng

Kết quả giải hệ cho 4 điểm cân bằng:

$$(0, 0), \quad (60, 0), \quad (0, 50), \quad (52, 24)$$

5. Phân tích ổn định

Tại mỗi điểm cân bằng (x^*, y^*) :

1. Tính $J(x^*, y^*)$
2. Tìm giá trị riêng λ_1, λ_2
3. Kết luận tính ổn định:

- Nếu $\text{Re}(\lambda_i) < 0$: Ổn định tiệm cận
- Nếu $\text{Re}(\lambda_i) > 0$: Không ổn định
- Nếu $\text{Re}(\lambda_i) = 0$: Cần xét tiếp

6. Vẽ trường hướng

- **sys**: Khai báo hệ phương trình vi phân đầy đủ
- **ics**: Các điều kiện ban đầu khác nhau
- **DEplot**: Vẽ trường hướng và các nghiệm số
 - **t** = 0..20: Thời gian mô phỏng
 - **x** = 0..80, **y** = 0..60: Miền hiển thị
 - **arrows** = THIN: Mũi tên mảnh
 - **dirfield** = 500: Mật độ mũi tên
 - **scene** = [x(t), y(t)]: Mặt phẳng pha

7. Vẽ điểm cân bằng

Sử dụng **pointplot** để đánh dấu các điểm cân bằng bằng các chấm đỏ.

8. Hiển thị kết quả

Gộp trường hướng và điểm cân bằng vào một hình vẽ bằng **display**.

Hướng dẫn thay đổi cho hệ khác

Khi đề bài thay đổi, cần chỉnh sửa các phần sau:

Phần cần thay	Mô tả
Biểu thức f1 , f2	Thay bằng biểu thức mới của hệ phương trình
Điểm cân bằng eq_points	Thay bằng kết quả từ solve cho hệ mới
Hệ phương trình sys	Cập nhật theo f1 , f2 mới
Miền vẽ x = a..b , y = c..d	Điều chỉnh cho phù hợp với miền nghiệm
Tiêu đề title	Sửa cho đúng với hệ mới
Điều kiện ban đầu ics	Có thể giữ nguyên hoặc thay đổi tùy ý

Các điểm cân bằng và tính ổn định

Dựa vào kết quả chạy code, ta có phân tích ổn định sau:

Điểm cân bằng	Giá trị riêng	Tính ổn định	Loại điểm
(0, 0)	180, 100	Nguồn (không ổn định)	Node không ổn định
(60, 0)	-180, 40	Yên ngựa (không ổn định)	Saddle point
(0, 50)	30, -100	Yên ngựa (không ổn định)	Saddle point
(52, 24)	-120, -26	Bồn (ổn định tiệm cận)	Node ổn định

Chú ý: Điểm cân bằng ổn định duy nhất là (52, 24), các nghiệm sẽ hội tụ về điểm này từ nhiều điều kiện ban đầu khác nhau.