BỘ GIÁO DỤC & ĐÀO TẠO

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH

KHOA ĐIỆN – ĐIỆN TỬ

BỘ MÔN TỰ ĐỘNG ĐIỀU KHIỂN

-----------------⸙∆⸙-----------------



BÁO CÁO

NHẬN DẠNG VÀ ĐIỀU KHIỂN HỆ THỐNG

**GVHD:** PSG.TS Vũ Văn Phong

**SVTH:** **MSSV:**

Nguyễn Văn Pháp 21151303

Vũ Tiến Phát 21151309

**Tp. Hồ Chí Minh, tháng 6 năm 2024**

# BÀI TOÁN 1: MODELING

MÔ HÌNH ROBOT 2 BÁNH TỰ CÂN BẰNG SỬ DỤNG BỘ ĐIỀU KHIỂN LQR

## Xây dựng mô hình toán học

* *Lý thuyết về LQR (Linear – quadratic regulator)*

Đối tượng tuyến tính mô tả bởi phương trình trạng thái:

Ta phải thiết kế bộ điều khiển phản hồi trạng thái K sao cho hàm mục tiêu J là nhỏ nhất.

Xác định hàm chỉ tiêu chất lượng:

Trong đó Q và R là các ma trận trọng số.

Quy tắc điều khiển phản hồi để giảm thiểu giá trị của hàm J là:

K được xác định là :

Khi S là hằng số thì 𝑆 = 0 ta có phương trình Riccati như sau:

* *Sơ đồ và hệ quy chiếu xe 2 bánh tự cân bằng*

A diagram of a wheel with wheels and arrows

Description automatically generated with medium confidence

Hình 1. Mô hình xe 2 bánh tự cân bằng trong mặt phẳng

Bảng 1. Ký hiệu và ý nghĩa của các đại lượng

|  |  |  |
| --- | --- | --- |
| Ký hiệu | Đơn vị | Ý nghĩa |
| m | Kg | Khối lượng của bánh xe |
| M | Kg | Khối lượng của Robot |
| R | m | Bán kính bánh xe |
| W | m | Chiều ngang của Robot |
| D | m | Chiều rộng của Robot |
| H | m | Chiều cao của Robot |
| L | m | Khoảng cách từ trọng tâm Robot đến trục bánh xe |
| fw |  | Hệ số ma sát giữa bánh xe và mặt phẳng di chuyển |
| fm |  | Hệ số ma sát giữa Robot và động cơ DC |
| Jm | kgm2 | Moment quán tính động cơ DC |
| Rm | Ω | Điện trở động cơ DC |
| Kb | Vs/rad | Hệ số EMF của động cơ DC |
| Kt | Nm/A | Moment xoắn của động cơ DC |
| N |  | Tỉ số giảm tốc |
| G | m/s2 | Gia tốc trọng trường |
| *θ* | rad | Góc trung bình của bánh trái và bánh phải |
| *θl,r* | rad | Góc của bánh trái và bánh phải |
| *ψ* | rad | Góc nghiêng của phần thân Robot |
| *ϕ* | rad | Góc xoay của Robot |
| xl, yl, zl | m | Tọa độ bánh trái |
| xr, yr, zr | m | Tọa độ bánh phải |
| xm, ym, zm | m | Tọa độ trung bình |
| Fθ, Fψ, Fϕ | Nm | Moment phát động theo các phương khác nhau |
| Fl, r | Nm | Moment phát động của động cơ bánh trái, phải |
| il, ir | A | Dòng điện động cơ bánh trái, phải |
| vl, vr | V | Điện áp động cơ bánh trái, phải |

Sử dụng phương pháp Euler – Lagrange để xây dựng mô hình động học. Giả sử tại thời điểm t = 0, Robot di chuyển theo chiều dương trục x, ta có các phương trình sau:

Tọa độ trung bình của Robot trong hệ quy chiếu:

Và

Tọa độ bánh trái trong hệ quy chiếu:

Tọa độ bánh phải trong hệ quy chiếu:

Tọa độ tâm đối xứng giữa hai động cơ trong hệ quy chiếu:

Phương trình động năng của chuyển động tịnh tiến:

Phương trình động năng của chuyển động quay:

Với là động năng quay của phần ứng động cơ trái

là động năng quay của phần ứng động cơ phải

Phương trình thế năng:

Phương trình Lagrange:

Lấy đạo hàm L theo các biến θ, ψ, ϕ ta được:

Momen động lực do động lực DC sinh ra:

Và:

Sử dụng phương pháp PWM để điều khiển động cơ nên chuyển từ dòng điện sang điện áp động cơ:

Xem điện cảm phần ứng tương đối nhỏ (gần bằng 0), có thể bỏ qua, suy ra:

Từ đó, các moment lực sinh ra:

Với và

Thu được phương trình động lực học mô tả chuyển động của robot như sau:

* *Tuyến tính hóa hệ thống*

Giả sử ta đặt các biến trạng thái như sau:

x1 = *θ*, x2 = , x3 = là góc quay và vận tốc góc quay bánh xe

x4 = *ψ* , x5 = , x6 = là góc nghiêng và vận tốc nghiêng của thân Robot

x7 = *ϕ* , x8 = , x9 = là góc xoay và vận tốc xoay của Robot

Với

Nếu chọn điểm làm việc là:

,

Ta có thể tuyến tính hóa hệ thống về dạng:

Với:

Tìm ma trận A như sau:

Tìm ma trận B như sau:

Lúc này ta có ma trận trọng số như sau:

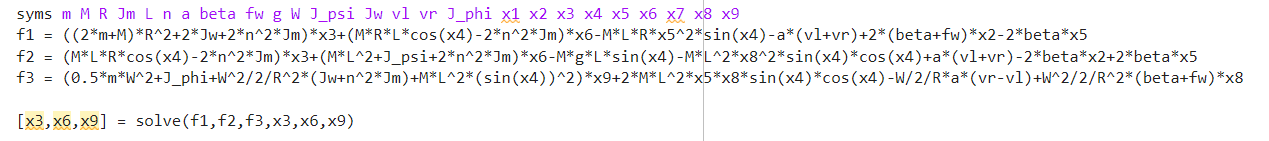
và

Với các thống số , để tinh chỉnh cho bộ điều khiển LQR. Trong đó tham số được coi là trọng số tối ưu tương ứng cho 6 biến trạng thái . Với mô hình hệ xe hai bánh ta có ma trận Q là ma trận 6x6 (tương ứng với 6 biến trạng thái) và R là 2x2 (tương ứng với 2 biến ngõ vào). Sau khi chọn được các tham số điều khiển tương ứng, chúng ta có thể xây dựng được tham số phản hồi K với tín hiệu điều khiển .

Thông số K được tính toán dựa vào phương trình Riccati. Tuy nhiên Matlab đã hỗ trợ việc tính toán thủ công bằng hàm LQR(A,B,Q,R) khi các ma trận A, B, Q, R đã được tìm ra.

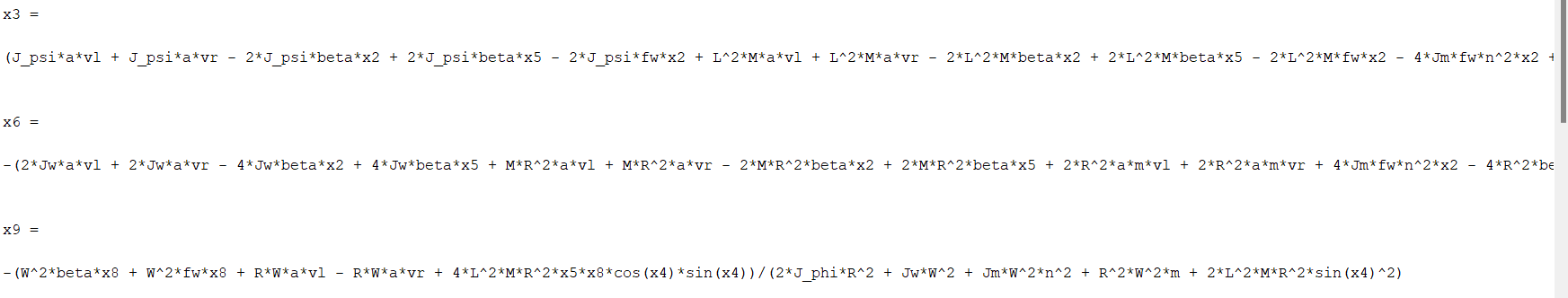
## Mô phỏng Matlab

Từ phương trình (1.26), (1.27), (1.28) chuyển hết sang 1 vế, ta được:



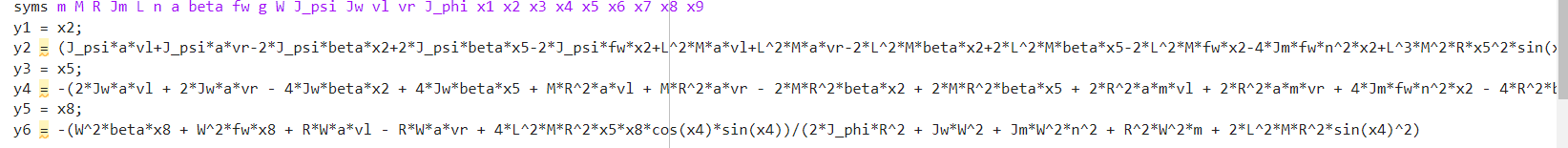
Hình 2. Tính , ,

Sau khi dùng Matlab giải phương trình ta được x3, x6, x9 tương ứng với , , :



Hình 3. Kết quả , ,

Từ công thức (1.29), ta đặt



Hình 4. Đặt các biến trạng thái

Tiếp theo, khai báo các thông số của mô hình:

A screenshot of a computer

Description automatically generated

Hình 5. Thông số mô hình

A group of black text

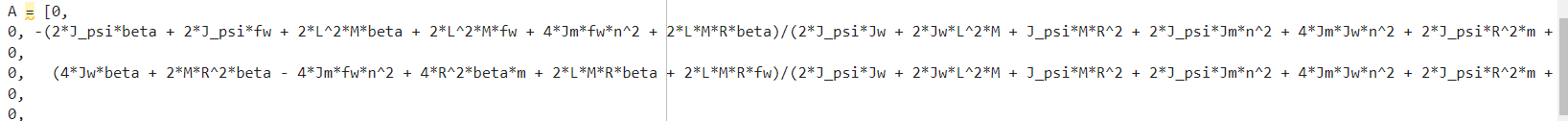
Description automatically generated

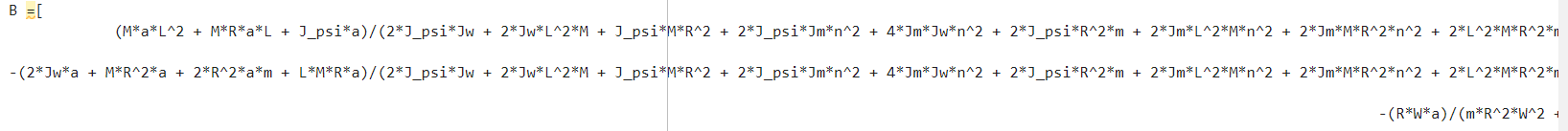
A screenshot of a computer code

Description automatically generated

Hình 6. Công thức tính ma trận A và B

Tính ra được ma trận A và B theo các biến:





Hình 7. Ma trận A và B theo các biến

Thế số vào, ta được:

A screenshot of a computer

Description automatically generated

Hình 8. Kết quả ma trận A và B

Chọn Q và R như sau:

A white background with black text

Description automatically generated

Hình 9. Chọn ma trận Q và R

Tính toán ma trận K theo hàm LQR:



Hình 10. Hàm LQR tính ma trận K

A black numbers and a white background

Description automatically generated

Hình 11. Kết quả ma trận K

Mô phỏng mô hình xe 2 bánh trong Simulink như sau:

A diagram of a computer

Description automatically generated

Hình 12. Mô phỏng xe 2 bánh tự cân bằng

A diagram of a computer

Description automatically generated

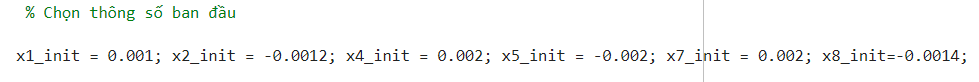
Hình 13. Khối Xe\_2\_Banh\_Tu\_Can\_Bang

A screenshot of a computer

Description automatically generated

Hình 14. Matlab function

Đặt các giá trị ban đầu như sau:



Hình 15. Thông số ban đầu

Sau khi chạy mô phỏng Simulink trong Matlab, kết quả thu được:

A graph with lines and a red line

Description automatically generated

Hình 16. Kết quả mô phỏng

**Nhận xét:** Tín hiệu đáp ứng của hệ thống tương đối tốt, các biến trạng thái của xe cân bằng tại 0, không có sai số xác lập, thời gian đạt xác lập nhanh, có xuất hiện vọt lố nhưng không đáng kể.

# BÀI TOÁN 2: INDENTIFICATION

## Xây dựng mô hình toán học

Ở bài tập này, chúng em chọn mô hình xe con lắc ngược.

A diagram of a machine

Description automatically generated

Hình 17. Hệ trục tọa độ cho hệ xe con lắc ngược

Bảng 2. Ký hiệu các đại lượng vật lý của hệ xe con lắc ngược

|  |  |  |
| --- | --- | --- |
| Ký hiệu | Đơn vị | Ý nghĩa |
| *θ* | rad | Góc nghiêng của con lắc |
| x | m | Vị trí của xe trên trục x |
| g | m/s2 | Gia tốc trọng trường |
| F | N | Lực tác dụng lên xe |
| M | kg | Khối lượng của xe |
| m | kg | Khối lượng của thanh con lắc |
|  | m | Chiều dài con lắc |
| I | kg/m2 | Moment quán tính của con lắc |

Sử dụng phương pháp Euler – Lagrange để xây dựng mô hình toán học. Ta có phương trình Euler – Lagrange như sau:

Với

T là tổng các thành phần động năng của hệ

U là tổng các thành phần thế năng của hệ

L là nhân tử Lagrangian

là tọa độ tổng quát

là tổng ngoại lực tác động lên hệ

Trong đó ta có:

Chọn mốc thế năng tại vị trí y = 0 nên thế năng của xe luôn luôn bằng 0. Do đó, thế năng của hệ chính là thế năng của con lắc:

Động năng của xe là:

Động năng của thanh con lắc:

Vị trí cuối của thanh con lắc chiều lên hệ trục tọa độ:

Vận tốc của con lắc trên hệ trục tọa độ là đạo hàm của vị trí:

Bình phương vận tốc trung bình của thanh con lắc:

Suy ra động năng của thanh con lắc:

Vậy động năng của hệ là:

Hàm Euler – Lagrange có dạng như sau:

Ta có:

Thay vào hệ phương trinhd Euler – Lagrange ta được hệ xe con lắc ngược như sau:

Lần lượt rút và ra, ta được:

## Mô phỏng Matlab để lấy dữ liệu

A diagram of a circuit

Description automatically generated

Hình 18. Mô phỏng Simulink mô hình xe con lắc ngược để lấy dữ liệu

A diagram of a computer

Description automatically generated

Hình 19. Khối Xe\_con\_lac\_nguoc

* Tín hiệu lực cấp vào mô hình xe con lắc ngược là khối Step

A screenshot of a computer program

Description automatically generated

Hình 20. Tín hiệu cấp vào mô hình

* Giả sử thông số hệ thống cần lấy dữ liệu như sau:

A number of numbers and letters

Description automatically generated with medium confidence

Hình 21. Thông số mô hình

Sau khi chạy mô phỏng, ta thu được bộ dữ liệu.

Để tiện cho việc tìm mô hình toán sử dụng tool Indentification của Matlab, ta dùng lệnh sau để chuyền dữ liệu vào cùng một struct:



Hình 22. Chuyển dữ liệu vào struct out

A screenshot of a computer program

Description automatically generated

Hình 23. Dữ liệu thu được trong struct out

## Sử dụng tool Identification của Matlab để tìm mô hình toán

Đầu tiên, mở cửa sổ System Identification bằng lệnh “ident” nhập trong Command Window.

A screenshot of a computer

Description automatically generated

Hình 24. Lựa chọn dữ liệu theo miền thời gian

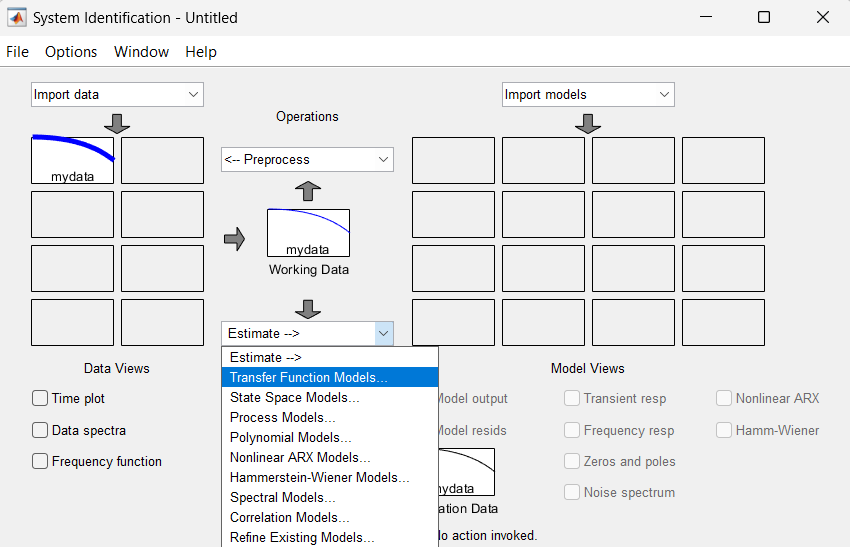
* Sau đó, nhập dữ liệu thu thập được vào input và output

A screenshot of a computer

Description automatically generated

Hình 25. Nhập dữ liệu vào Import Data

* Tiếp theo lựa chọn Transfer Function Models:



Hình 26. Lựa chọn ước lượng hàm truyền

A screenshot of a computer

Description automatically generated

Hình 27. Chọn 5 poles 3 zeros

* Kết quả thu được

A screenshot of a computer

Description automatically generated

Hình 28. Kết quả thu được

A number and a line

Description automatically generated with medium confidence

Hình 29. Hàm truyền thu được

* Ước lượng phù hợp 99.78% so với dữ liệu:

A black text on a white background

Description automatically generated

Hình 30. Độ chính xác của ước lượng

* Tiến hành mô phỏng so sánh giữa mô hình nhận dạng và mô hình gốc:

A diagram of a rectangular object

Description automatically generated

Hình 31. Mô phỏng so sánh

Sau khi chạy mô phỏng, ta thu được kết quả sau:

A graph with a line

Description automatically generated

Hình 32. Kết quả đầu ra giữa mô hình gốc và mô hình nhận dạng

A graph with a line

Description automatically generated

Hình 33. Sai số giữa mô hình gốc và mô hình nhận dạng

**Nhận xét:** Mô hình gốc và mô hình nhận dạng tuy có hơi lệch nhau nhưng nhìn chung sai số ở mức có thể chấp nhận được (~ 1%).

## Sử dụng Parameter Estimation của Matlab để nhận dạng thông số cho hệ thống

A number of numbers and symbols

Description automatically generated with medium confidence

Hình 34. Thông số tự cho để ước lượng

* Mô phỏng ước lượng giống với mô phỏng để lấy số liệu, chỉ thay các thông số động cơ bằng các thông số tự cho để ước lượng.

A diagram of a block diagram

Description automatically generated

Hình 35. Mô phỏng nhận dạng thông số

A diagram of a computer program

Description automatically generated with medium confidence

Hình 36. Khối Subsystem

* Chọn công cụ Parameter Estimator để tiến hành nhận dạng thông số M, m, l

A screenshot of a computer

Description automatically generated

Hình 37. Select Parameters

Copy dữ liệu thu thập được bỏ vào đây:

A screenshot of a computer

Description automatically generated

Hình 38. Select Experiments

* Vẽ biểu đồ sau đó bắt đầu nhận dạng, kết quả sau khi chạy xong như sau:

A screenshot of a computer

Description automatically generated

Hình 39. Kết quả nhận dạng

Các thông số nhận dạng được như sau:

M = 2.1534

l = 0.30319

m = -0.91927

Thực hiện so sánh giữa mô hình gốc và mô hình nhận dạng:

A diagram of a computer program

Description automatically generated with medium confidence

Hình 40. Mô phỏng nhận dạng

A graph with a curve

Description automatically generated

Hình 41. Sai số giữa mô hình gốc và mô hình nhận dạng

A line graph with dots

Description automatically generated

Hình 42. Đồ thị đầu ra giữa thông số mô hình và thông số ước lượng

**Nhận xét:** Tuy thông số nhận dạng được khá lệch so với thông số để lấy số liệu, nhưng tổng quan sai số giữa 2 mô hình là chấp nhận được, rơi vào khoảng ±4 rad.

## Thiết kế bộ điều khiển trượt cho hệ xe con lắc

**\_Bước 1: Chọn mặt trượt**



Trong đó: ; ; ; 

Với:



 và là các hằng số dương

Chọn giá trị 

**\_Bước 2: Tính đạo hàm của mặt trượt:**

****

**\_Bước 3: Biểu thức bộ điều khiển trượt gồm 2 thành phần:**

****

Trong đó: ****được tính khi cho và 



Và:



Thay bộ điều khiển trượt vào công thức đạo hàm mặt trượt ta có:

=

=

Chọn giá trị ; giả thuyết không có nhiễu nên =0

Ta tìm được luật của bộ điều khiển trượt là:

U ==

Trong đó: với được chọn sao cho

Chọn =