 **ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC BÁCH KHOA**

**\_\_\_\_\_\_\_\_\_­­­­­­­­­­­­\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**



**Bài tập lớn**

Thiết kế hệ thống cơ điện tử

Đề tài: Thiết kế bộ điều khiển và mô phỏng con lắc ngược

Lớp: L01

Giảng Viên: Võ Tường Quân

Danh sách thành viên:

|  |  |  |
| --- | --- | --- |
| STT | Họ và tên | MSSV |
| 1 | Đồng Minh Quốc | 1914857 |
| 2 | Lê Dương Can | 1910853 |
| 3 | Phạm Cao Tâm | 1915029 |
| 4 | Nguyễn Duy Nhân | 1914437 |

*Thành phố Hồ Chí Minh - 2022*

Mục Lục

Chương 1: Giới thiệu 1

* 1. Bảng phân công nhiệm vụ 1
  2. Giới thiệu đề tài 1
  3. Mục đích thiết kế 2
  4. Khả năng ứng dụng 3

Chương 2: Cơ sở lý thuyết 3

2.1 Phân tích động học con lắc ngược 3

2.2 Tính ổn định và điều khiển 7

Chương 3: Thiết kế bộ điều khiển PID 9

3.1 Giới thiệu về bộ điều khiển PID 9

3.2 Các phương pháp tìm hệ số Kp, Ki, Kd 15

Chương 4: Mô phỏng bằng Matlab 17

4.1 Mô phỏng đáp ứng của hệ 17

4.2 Mô phỏng bằng Simulink 21

Tài liệu tham khảo 24

Chương 1: Giới thiệu

**1.1 Bảng phân công nhiêm vụ**

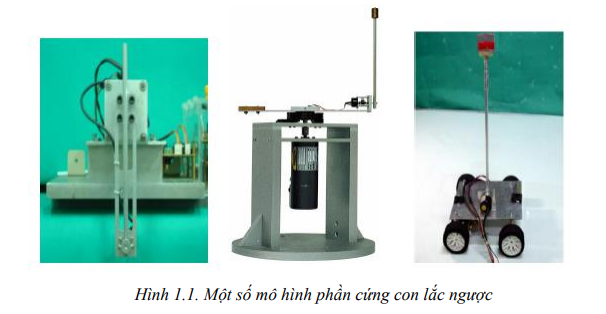
|  |  |
| --- | --- |
| Họ và tên | Nhiệm vụ |
| Nguyễn Duy Nhân | + Giới thiệu đề tài và bộ điều khiển |
| Đồng Minh Quốc | + Viết phương trình toán của con lắc  + Xét tính ổn định và điều khiển được |
| Phạm Cao Tâm | + Mô phỏng đáp ứng của hệ |
| Lê Dương Can | + Mô phỏng vị trí, vận tốc của con lắc và xe |

**1.2 Giới thiệu về đề tài**

Hệ thống con lắc ngược là một hệ thống điều khiển kinh điển, nó được sử dụng trong giảng dạy và nghiên cứu ở hầu hết các trường đại học trên khắp thế giới. Hệ thống con lắc ngược là mô hình phù hợp để kiểm tra các thuật toán điều khiển hệ phi tuyến cao trở lại ổn định. Đây là một hệ thống SIMO (Single Input Multi Output) điển hình vì chỉ gồm một ngõ vào là lực tác động cho động cơ mà phải điều khiển cả vị trí và góc lệch con lắc ngược sao cho thẳng đứng. Ngoài ra, phương trình toán học được đề cập đến của con lắc ngược mang tính chất phi tuyến điển hình. Vì thế, đây là một mô hình nghiên cứu lý tưởng cho các phòng thí nghiệm điều khiển tự động. Các giải thuật hay phương pháp điều khiển được nghiên cứu trên mô hình con lắc ngược nhằm tìm ra các giải pháp tốt nhất trong các ứng dụng điều khiển thiết bị tự động trong thực tế: điều khiển tốc độ động cơ, giảm tổn hao công suất, điều khiển vị trí, điều khiển nhiệt độ, điều khiển cân bằng hệ thống, …

Đây là một mô hình rất điển hình cần phải có ở bất kỳ một trường Đại học, Cao đẳng nào theo hướng chuyên ngành kỹ thuật tại Việt Nam, nhất là ngành điều khiển tự động hóa, điện công nghiệp, cơ điện tử… Việc xây dựng mô hình sẽ giúp ích cho công tác giảng dạy trực quan hơn, dễ dàng kiểm chứng với các giải thuật điều khiển trên lý thuyết, là cơ sở nghiên cứu khoa học cho cả giảng viên và sinh viên tại trường.

Mô hình con lắc ngược là một mô hình kinh điển và là một mô hình phức tạp có độ phi tuyến cao trong lĩnh vực điều khiển tự động hóa. Để xây dựng và điều khiển hệ con lắc ngược tự cân bằng đòi hỏi người điều khiển phải có nhiều kiến thức về cơ khí lẫn điều khiển hệ thống. Với mô hình này sẽ giúp người điều khiển kiểm chứng được nhiều cơ sở lý thuyết và các thuật toán khác nhau trong điều khiển tự động. Hệ thống con lắc ngược đang được nghiên cứu hiện nay gồm một số loại như sau: con lắc ngược đơn, con lắc ngược quay, hệ xe con lắc ngược, con lắc ngược 2, 3 bậc tự do, ….



**1.3 Mục đích thiết kế**

Mục tiêu chung: Thiết kế và điều khiển hệ con lắc ngược tự cân bằng phục vụ trong giảng dạy đại học và nghiên cứu khoa học

Mục tiêu cụ thể:

+ Thiết kế mô hình con lắc ngược

+ Sử dụng các giải thuật điều khiển khác nhau để điều khiển hệ thống

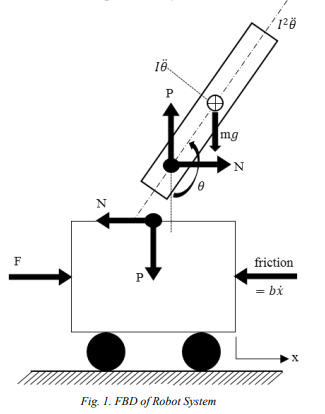
+ Điều khiển cân bằng hệ con lắc ngược ứng dụng vào 1 số đề tài cụ thể

**1.4 Khả năng ứng dụng**

Con lắc ngược là cơ sở để tạo ra các hệ thống tự cân bằng như:  
xe hai bánh tự cân bằng, tháp vô tuyến, giàn khoan, công trình biển…

Chương 2: Cơ sở lý thuyết

**2.1 Phân tích động học con lắc ngược**



Tổng lực theo phương ngang:

 (1)

Lực ly tâm theo phương ngang:

 (2)

Phản lực N:

 (3)

Thay (2) và (3) vào (1):

 (4)

Tổng lực dọc:

 (5)

Tổng momen tại tâm con lắc:

 (6)

Suy ra:

 (7)

Phương trình của cơ hệ theo t với phi là góc lệch so với biên độ con lắc



Tìm hàm truyền của hệ bằng phép biến đổi laplace



Hàm truyền của góc con lắc:



Hàm truyền vị trí con lắc:



Phương trình viết dưới dạng không gian trạng thái (state space)

= Ax + Bu

y = Cx + Du

Trong đó

* A là ma trận của hệ thống,
* B là ma trận điều khiển
* u là vector điều khiển
* C là ma đầu ra
* D là ma trận truyền thẳng
* x là vector trạng thái hệ
* y là vector đầu ra

= + u

y = + u

Thông số của hệ

M=0.5 kg

m=0.2 kg

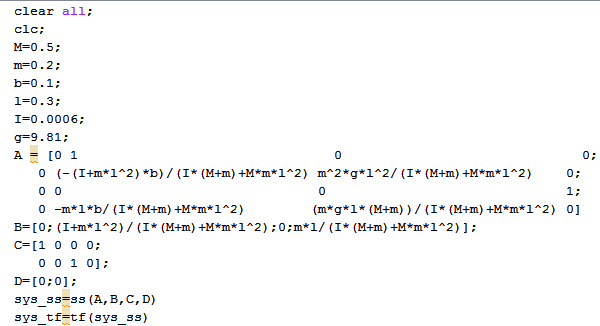
b=0.1 N/m/sec

I= 0.006 kg.m^2

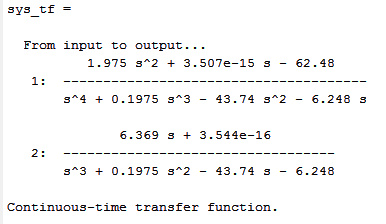
g=9.8 m/s^2

l=0.3 m

Tính hàm truyền của hệ bằng matlab



Chạy code ta tìm được hàm truyền theo miền s

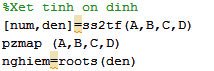


**2.2 Tính ổn định và điều khiển**

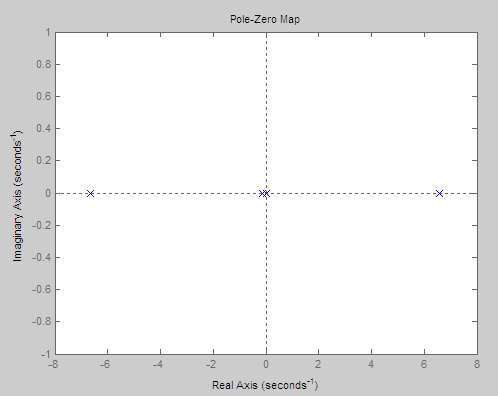
**-** Tính ổn định

Hệ được cho là ổn định khi nghiệm của đa thức dưới mẫu được cho bởi hàm truyền thỏa những phương pháp như poles – zeros, bảng routh, roots locus

Code matlab



Kết quả



Nhìn đồ thị ta thấy có một điểm tọa độ x dương nên hệ không ổn định.

- Tính điều khiển được

Tính điều khiển được là khả năng tác động vào hệ thống để đạt được trạng thái mong muốn bằng cách sử dụng một tín hiệu điều khiển thích hợp.

Đối với hệ thống tuyến tính – bất biến theo thời gian (LTI) tính chất điều khiển được liên quan đến ma trận hệ thống A, ma trận nói lên tính chất của hệ thống và ma trận điều khiển B, ma trận liên quan đến việc bố trí các động cơ dẫn động.

Theo tiêu chuẩn Kalman là điều khiển được nếu ma trận:

C = [B, AB, A2B, …, An-1B] có hạng bằng n, rank(C)=n

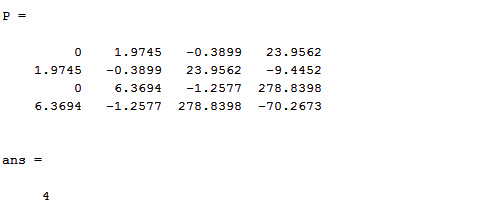
Code matlab

%Xet tinh dieu khien

P = [B A\*B A^2\*B A^3\*B]

rank(P)

Kết quả

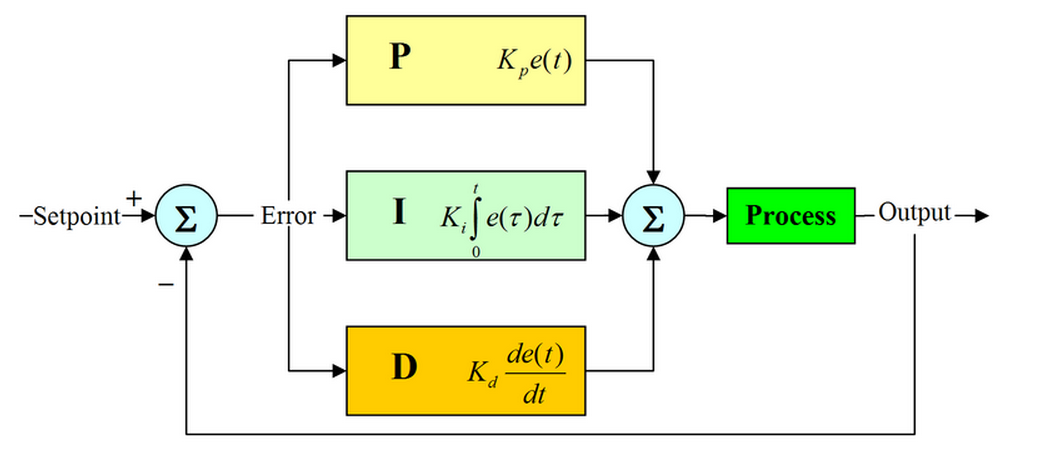


Từ đây ta suy ra hệ điều khiển được.

Chương 3: Thiết kế bộ điều khiển PID

**3.1 Giới thiệu bộ điều khiển PID**

Bộ điều khiển PID là một bộ điều khiển vi tích phân tỉ lệ (bộ điều khiển PID- Proportional Integral Derivative) là một [cơ chế phản hồi](http://vi.wikipedia.org/w/index.php?title=C%C6%A1_ch%E1%BA%BF_ph%E1%BA%A3n_h%E1%BB%93i&action=edit&redlink=1) [vòng điều khiển](http://vi.wikipedia.org/w/index.php?title=V%C3%B2ng_%C4%91i%E1%BB%81u_khi%E1%BB%83n&action=edit&redlink=1) ([bộ điều khiển](http://vi.wikipedia.org/w/index.php?title=B%E1%BB%99_%C4%91i%E1%BB%81u_khi%E1%BB%83n_(l%C3%BD_thuy%E1%BA%BFt_%C4%91i%E1%BB%81u_khi%E1%BB%83n_t%E1%BB%B1_%C4%91%E1%BB%99ng)&action=edit&redlink=1)) tổng quát được sử dụng rộng rãi trong các [hệ thống điều khiển](http://vi.wikipedia.org/w/index.php?title=H%E1%BB%87_th%E1%BB%91ng_%C4%91i%E1%BB%81u_khi%E1%BB%83n&action=edit&redlink=1) công nghiệp – bộ điều khiển PID được sử dụng phổ biến nhất trong số các bộ điều khiển phản hồi. Một bộ điều khiển PID tính toán một giá trị "sai số" là hiệu số giữa giá trị đo [thông số biến đổi](http://vi.wikipedia.org/w/index.php?title=Th%C3%B4ng_s%E1%BB%91_bi%E1%BA%BFn_%C4%91%E1%BB%95i&action=edit&redlink=1) và [giá trị đặt](http://vi.wikipedia.org/w/index.php?title=Gi%C3%A1_tr%E1%BB%8B_%C4%91%E1%BA%B7t_(h%E1%BB%87_th%E1%BB%91ng_%C4%91i%E1%BB%81u_khi%E1%BB%83n)&action=edit&redlink=1) mong muốn. Bộ điều khiển sẽ thực hiện giảm tối đa sai số bằng cách điều chỉnh giá trị điều khiển đầu vào.



*Hình 1: Sơ đồ khối của bộ điều khiển PID*

[Giải thuật](http://vi.wikipedia.org/wiki/Thu%E1%BA%ADt_to%C3%A1n) tính toán bộ điều khiển PID bao gồm 3 thông số riêng biệt, do đó đôi khi nó còn được gọi là điều khiển ba khâu: các giá trị [tỉ lệ](http://vi.wikipedia.org/w/index.php?title=T%E1%BB%89_l%E1%BB%87(to%C3%A1n_h%E1%BB%8Dc)&action=edit&redlink=1), [tích phân](http://vi.wikipedia.org/wiki/T%C3%ADch_ph%C3%A2n) và [đạo hàm](http://vi.wikipedia.org/wiki/%C4%90%E1%BA%A1o_h%C3%A0m_v%C3%A0_vi_ph%C3%A2n_c%E1%BB%A7a_h%C3%A0m_s%E1%BB%91), viết tắt là P, I, và D. Giá trị tỉ lệ xác định tác động của sai số hiện tại, giá trị tích phân xác định tác động của tổng các sai số quá khứ, và giá trị vi phân xác định tác động của tốc độ biến đổi sai số. Tổng chập của ba tác động này dùng để điều chỉnh quá trình thông qua một phần tử điều khiển như vị trí của van điều khiển hay bộ nguồn của phần tử gia nhiệt. Nhờ vậy, những giá trị này có thể làm sáng tỏ về quan hệ thời gian: P phụ thuộc vào sai số hiện tại, I phụ thuộc vào tích lũy các sai số quá khứ, và D dự đoán các sai số tương lai, dựa vào tốc độ thay đổi hiện tại.

Bằng cách điều chỉnh 3 hằng số trong giải thuật của bộ điều khiển PID, bộ điều khiển có thể dùng trong những thiết kế có yêu cầu đặc biệt. Đáp ứng của bộ điều khiển có thể được mô tả dưới dạng độ nhạy sai số của bộ điều khiển, giá trị mà bộ điều khiển [vọt lố](http://vi.wikipedia.org/w/index.php?title=V%E1%BB%8Dt_l%E1%BB%91_(t%C3%ADn_hi%E1%BB%87u)&action=edit&redlink=1) điểm đặt và giá trị dao động của hệ thống. Lưu ý là công dụng của giải thuật PID trong điều khiển không đảm bảo tính tối ưu hoặc ổn định cho hệ thống.

Vài ứng dụng có thể yêu cầu chỉ sử dụng một hoặc hai khâu tùy theo hệ thống. Điều này đạt được bằng cách thiết đặt đội lợi của các đầu ra không mong muốn về 0. Một bộ điều khiển PID sẽ được gọi là bộ điều khiển PI, PD, P hoặc I nếu vắng mặt các tác động bị khuyết. [Bộ điều khiển PI](http://vi.wikipedia.org/w/index.php?title=B%E1%BB%99_%C4%91i%E1%BB%81u_khi%E1%BB%83n_PI&action=edit&redlink=1) khá phổ biến, do đáp ứng vi phân khá nhạy đối với các nhiễu đo lường, trái lại nếu thiếu giá trị tích phân có thể khiến hệ thống không đạt được giá trị mong muốn.

Sơ đồ điều khiển PID được đặt tên theo ba khâu hiệu chỉnh của nó, tổng của ba khâu này tạo thành bởi các biến điều khiển (MV). Ta có:

MV(t) = Pout + Iout+ Dout

Trong đó:

Khâu tỉ lệ (đôi khi còn được gọi là độ lợi) làm thay đổi giá trị đầu ra, tỉ lệ với giá trị sai số hiện tại. Đáp ứng tỉ lệ có thể được điều chỉnh bằng cách nhân sai số đó với một hằng số Kp, được gọi là độ lợi tỉ lệ.

**Khâu tỉ lệ được cho bởi:**

Pout = Kpe(t)

Trong đó:

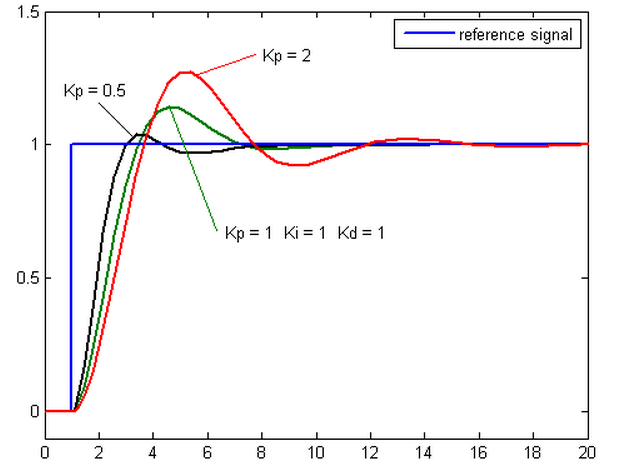
Pout: Thừa số tỉ lệ của đầu ra

Kp: Độ lợi tỉ lệ, thông số điều chỉnh

e: Sai số

t: Thời gian hay thời gian tức thời (hiện tại)

Độ lợi của khâu tỉ lệ lớn là do thay đổi lớn ở đầu ra mà sai số thay đổi nhỏ. Nếu độ lợi của khâu tỉ lệ quá cao, hệ thống sẽ không ổn định (xem phần [điều chỉnh vòng](http://vi.wikipedia.org/wiki/B%E1%BB%99_%C4%91i%E1%BB%81u_khi%E1%BB%83n_PID#.C4.91i.E1.BB.81u_ch.E1.BB.89nh_v.C3.B2ng)). Ngược lại, độ lợi nhỏ là do đáp ứng đầu ra nhỏ trong khi sai số đầu vào lớn, và làm cho bộ điều khiển kém nhạy, hoặc đáp ứng chậm. Nếu độ lợi của khâu tỉ lệ quá thấp, tác động điều khiển có thể sẽ quá bé khi đáp ứng với các nhiễu của hệ thống.



*Hình 2: Sự thay đổi khi điều chỉnh Kp*

**Khâu tích phân**

Phân phối của khâu tích phân (đôi khi còn gọi là reset) tỉ lệ thuận với cả biên độ sai số lẫn quảng thời gian xảy ra sai số. Tổng sai số tức thời theo thời gian (tích phân sai số) cho ta tích lũy bù đã được hiệu chỉnh trước đó. Tích lũy sai số sau đó được nhân với độ lợi tích phân và cộng với tín hiệu đầu ra của bộ điều khiển. Biên độ phân phối của khâu tích phân trên tất cả tác động điều chỉnh được xác định bởi độ lợi tích phân, Ki.

Thừa số tích phân được cho bởi:

Trong đó:

Iout: thừa số tích phân của đầu ra

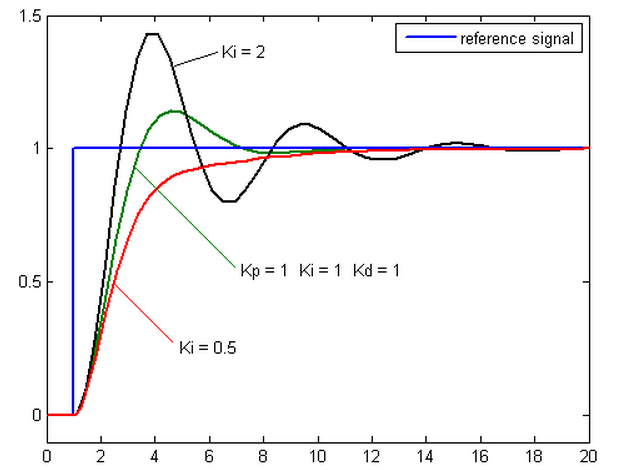
Ki: độ lợi tích phân, 1 thông số điều chỉnh

e: sai số

t: thời gian hoặc thời gian tức thời (hiện tại)

T: một biến tích phân trung gian

Khâu tích phân (khi cộng thêm khâu tỉ lệ) sẽ tăng tốc chuyển động của quá trình tới điểm đặt và khử số dư sai số ổn định với một tỉ lệ chỉ phụ thuộc vào bộ điều khiển. Tuy nhiên, vì khâu tích phân là đáp ứng của sai số tích lũy trong quá khứ, nó có thể khiến giá trị hiện tại vọt lố qua giá trị đặt (ngang qua điểm đặt và tạo ra một độ lệch với các hướng khác). Để tìm hiểu thêm các đặc điểm của việc điều chỉnh độ lợi tích phân và độ ổn của bộ điều khiển, xin xem phần [điều chỉnh vòng lặp](http://vi.wikipedia.org/wiki/B%E1%BB%99_%C4%91i%E1%BB%81u_khi%E1%BB%83n_PID#.C4.91i.E1.BB.81u_ch.E1.BB.89nh_v.C3.B2ng_l.E1.BA.B7p).



*Hình 3: Sự thay đổi khi điều chỉnh Ki*

**Khâu vi phân:**

Tốc độ thay đổi của sai số qua trình được tính toán bằng cách xác định độ dốc của sai số theo thời gian (tức là đạo hàm bậc một theo thời gian) và nhân tốc độ này với độ lợi tỉ lệ Kd. Biên độ của phân phối khâu vi phân (đôi khi được gọi là tốc độ) trên tất cả các hành vi điều khiển được giới hạn bởi độ lợi vi phân, Kd.

Thừa số vi phân được cho bởi:

Trong đó:

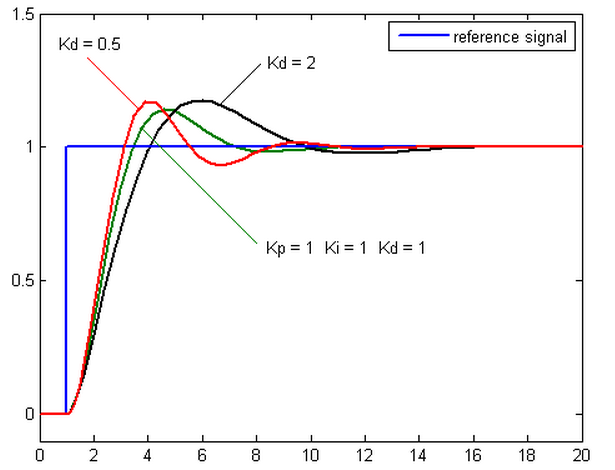
Dout: thừa số vi phân của đầu ra

Kd: Độ lợi vi phân, một thông số điều chỉnh

e: Sai số

t: thời gian hoặc thời gian tức thời (hiện tại)

Khâu vi phân làm chậm tốc độ thay đổi của đầu ra bộ điều khiển và đặc tính này là đang chú ý nhất để đạt tới điểm đặt của bộ điều khiển. Từ đó, điều khiển vi phân được sử dụng để làm giảm biên độ vọt lố được tạo ra bởi thành phần tích phân và tăng cường độ ổn định của bộ điều khiển hỗn hợp. Tuy nhiên, phép vi phân của một tín hiệu sẽ khuếch đại nhiễu và do đó khâu này sẽ nhạy hơn đối với nhiễu trong sai số, và có thể khiến quá trình trở nên không ổn định nếu nhiễu và độ lợi vi phân đủ lớn. Do đó một xấp xỉ của bộ vi sai với băng thông giới hạn thường được sử dụng hơn. Chẳng hạn như mạch bù sớm pha.



*Hình 4: Sự thay đổi khi điều chỉnh Kd*

**Tóm tắt về bộ PID:**

Khâu tỉ lệ, tích phân, vi phân được cộng lại với nhau để tính toán đầu ra của bộ điều khiển PID. Định nghĩa rằng u(t) là đầu ra của bộ điều khiển, biểu thức cuối cùng của giải thuật PID là:

trong đó các thông số điều chỉnh là:

* Độ lợi tỉ lệ, Kp

+ Giá trị càng lớn thì đáp ứng càng nhanh do đó sai số càng lớn, bù khâu tỉ lệ càng lớn. Một giá gị độ lợi tỉ lệ quá lớn sẽ dấn đến quá trình mất ổn định và dao động.

* Độ lợi tích phân, Ki

+ Giá trị càng lớn kéo theo sai số ổn định bị khử càng nhanh. Đổi lại là độ vọt lố càng lớn: bất kỳ sai số âm nào được tích phân trong suốt đáp ứng quá độ phải được triệt tiêu tích phân bằng sai số dương trước khi tiến tới trạng thái ổn định.

* Độ lợi vi phân, Kd

+ Giá trị càng lớn càng giảm độ vọt lố, nhưng lại làm chậm đáp ứng quá độ và có thể dẫn đến mất ổn định do khuếch đại nhiễu tín hiệu trong phép vi phân sai số.

**3.2 Các phương pháp tìm hệ số Kp, Ki, Kd**

Có nhiều phương pháp khác nhau để điều chỉnh vòng lặp PID. Những phương pháp hữu hiệu nhất thường bao gồm những triển khai của vài dạng mô hình xử lý, sau đó chọn P, I, và D dựa trên các thông số của mô hình động học. Các phương pháp điều chỉnh thủ công tương đối không hiệu quả lắm, đặc biệt nếu vòng lặp có thời gian đáp ứng được tính bằng phút hoặc lâu hơn.

Lựa chọn phương pháp thích hợp sẽ phụ thuộc phần lớn vào việc có hay không vòng lặp có thể điều chỉnh "offline", và đáp ứng thời gian của hệ thống. Nếu hệ thống có thể thực hiện offline, phương pháp điều chỉnh tốt nhất thường bao gồm bắt hệ thống thay đổi đầu vào từng bước, tín hiệu đo lường đầu ra là một hàm thời gian, sử dụng đáp ứng này để xác định các thông số điều khiển.

|  |  |  |
| --- | --- | --- |
| Lựa chọn phương pháp điều chỉnh | | |
|  | Ưu điểm | Khuyết điểm |
| Điều chỉnh thủ công | Không cần hiểu biết về toán. Phương pháp online. | Yêu cầu nhân viên có kinh nghiệm. |
| Ziegler–Nichols | Phương pháp chứng minh.  Phương pháp online. | Làm rối loạn quá trình, một số thử nghiệm và lỗi, phải điều chỉnh nhiều lần |
| Các công cụ phần mềm | Điều chỉnh chắc chắn. Phương pháp online hoặc offline. Có thể bao gồm phân tích các van và cảm biến. Cho phép mô phỏng trước khi tải xuống để thực thi. | Giá cả cao, và phải huấn luyện. |
| Cohen-Coon | Xử lý các mô hình tốt. | Yêu cầu kiến thức toán học. Phương pháp offline. Chỉ tốt đối với các quá trình bậc một. |

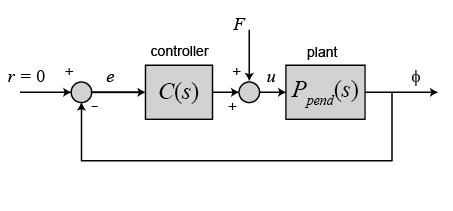
**Phương pháp nhóm chọn điều chỉnh thủ công:**

Nếu hệ thống phải duy trì trạng thái online, một phương pháp điều chỉnh là thiết đặt giá trị đầu tiên của Ki và Kd bằng không. Tăng dần Kp cho đến khi đầu ra của vòng điều khiển dao động, sau đó Kp có thể được đặt tới xấp xỉ một nửa giá trị đó để đạt được đáp ứng "1/4 giá trị suy giảm biên độ". Sau đó tăng Ki đến giá trị phù hợp sao cho đủ thời gian xử lý. Tuy nhiên, Ki quá lớn sẽ gây mất ổn định. Cuối cùng, tăng Kd, nếu cần thiết, cho đến khi vòng điều khiển nhanh có thể chấp nhận được nhanh chóng lấy lại được giá trị đặt sau khi bị nhiễu. Tuy nhiên, Kd quá lớn sẽ gây đáp ứng dư và vọt lố.Một điều chỉnh cấp tốc của vòng điều khiển PID thường hơi quá lố một ít khi tiến tới điểm đặt nhanh chóng; tuy nhiên, vài hệ thống không chấp nhận xảy ra vọt lố, trong trường hợp đó, ta cần một hệ thống vòng kín [giảm lố](http://vi.wikipedia.org/w/index.php?title=Gi%E1%BA%A3m_l%E1%BB%91&action=edit&redlink=1), thiết đặt một giá trị Kp nhỏ hơn một nữa giá trị Kp gây ra dao động.

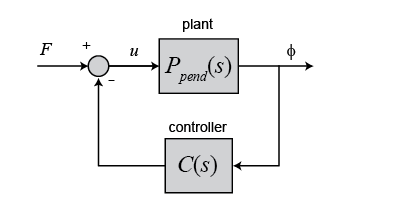
Chương 4 Mô phỏng bằng Matlab

**4.1 Mô phỏng đáp ứng của hệ**

Sơ đồ khối của hệ khi điều khiển vị trí

Rút gọn ta được



Khi đó hàm truyền biến đổi thành 

Đáp ứng xung của vị trí con lắc ngược khi điều khiển bằng PID

Code matlab:

s = tf('s');

q = (M+m) \*(I+m\*l^2) - (m\*l) ^2;

P\_pend = (m\*l\*s/q)/ (s^3 + (b\*(I + m\*l^2)) \*s^2/q - ((M + m) \*m\*g\*l) \* s/q - b\*m\*g\*l/q)

Kp = 100;

Ki = 1;

Kd = 20;

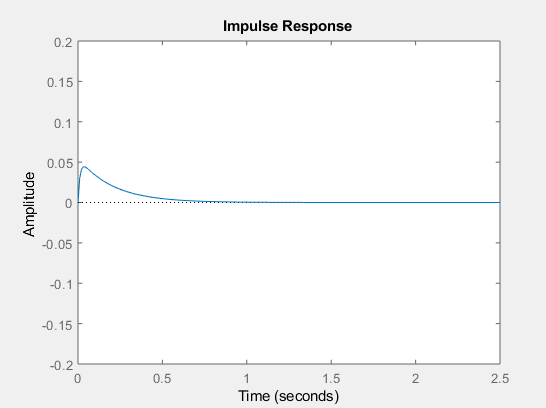
C = pid (Kp, Ki, Kd);

T = feedback (P\_pend, C);

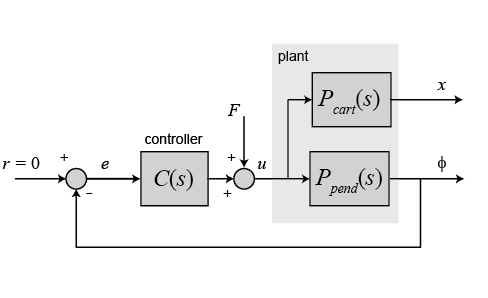
t=0:0.01:10;

impulse (T, t)

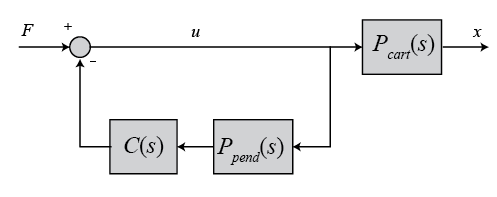
axis ([0, 2.5, -0.2, 0.2]);



Sơ đồ khối của hệ khi điều khiển vị trí và góc con lắc



Sắp xếp lại ta được sơ đồ khối sau



 Khi đó hàm truyền sẽ bằng



Đáp ứng của vị trí xe khi điều khiển bằng PID

Code matlab

P\_pend = (m\*l\*s/q)/ (s^3 + (b\*(I + m\*l^2)) \*s^2/q - ((M + m) \*m\*g\*l) \*s/q - b\*m\*g\*l/q);

Kp = 100;

Ki = 1;

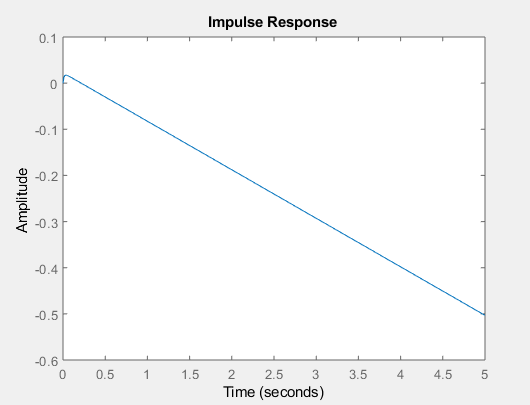
Kd = 20;

P\_cart = (((I+m\*l^2)/ q) \*s^2 - (m\*g\*l/q))/ (s^4 + (b\*(I + m\*l^2)) \*s^3/q - ((M + m) \*m\*g\*l) \*s^2/q - b\*m\*g\*l\*s/q);

T2 = feedback (1, P\_pend\*C) \*P\_cart;

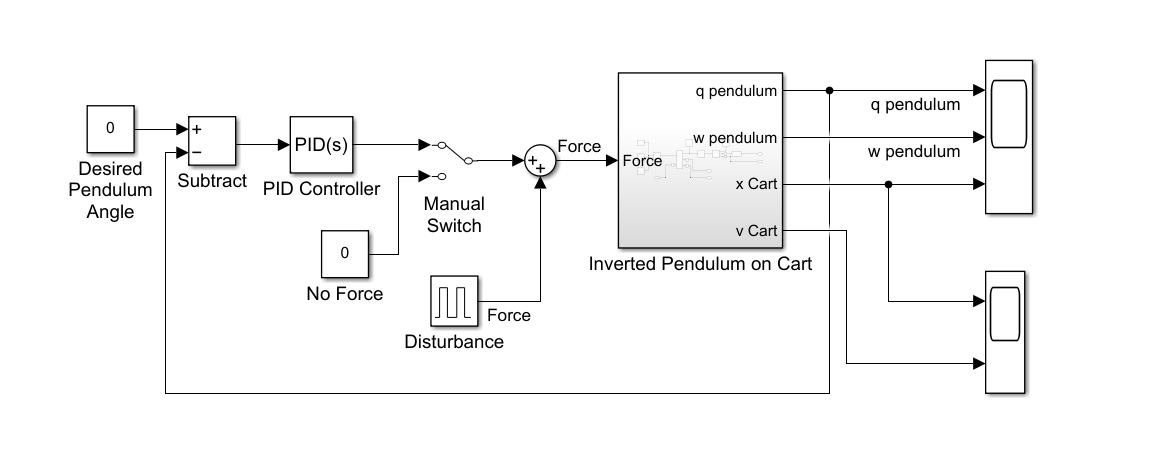
t = 0:0.01:5;

impulse (T2, t);

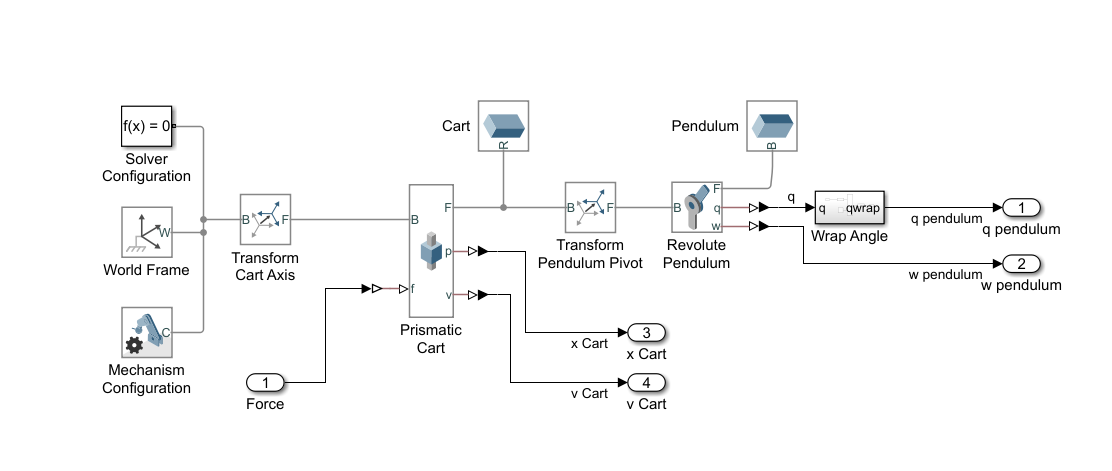


**4.2 Mô phỏng bằng simulink**

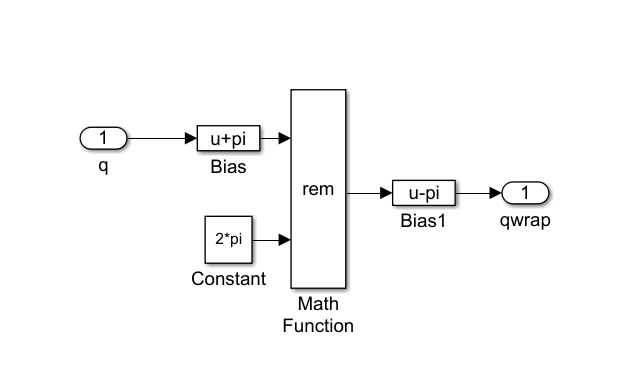
Sơ đồ khối tổng thể



Khối Inverted Pendulum on Cart

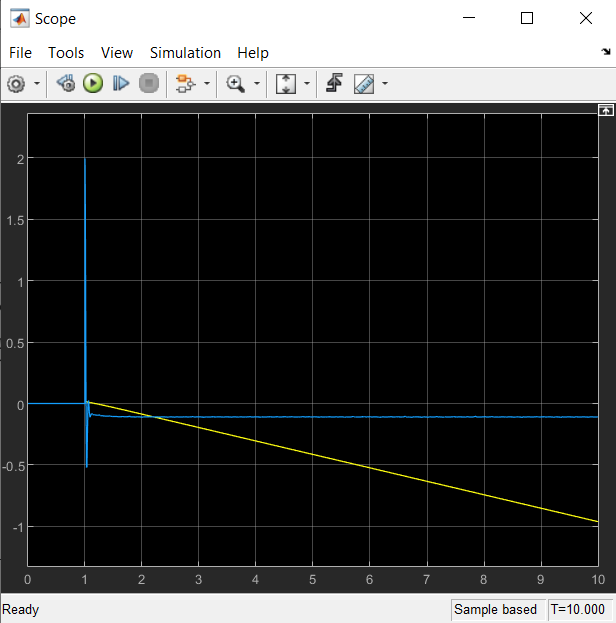


Khối Wrap Angle

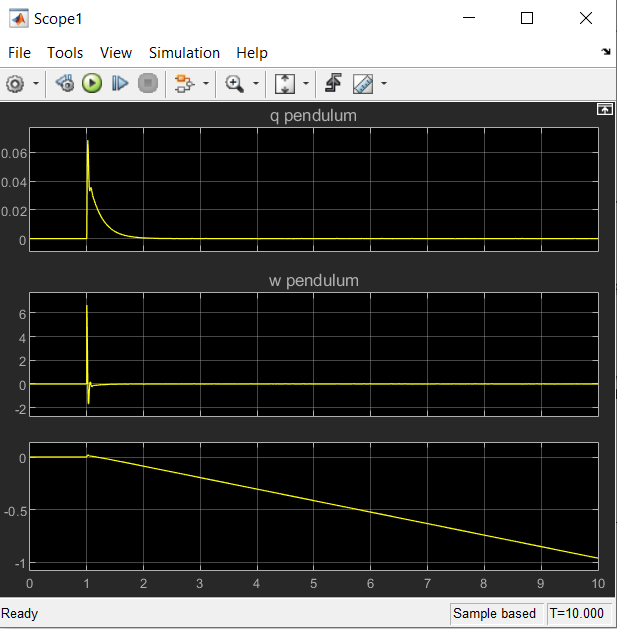


Kết quả chạy mô phỏng:

1. Vận tốc góc và góc con lắc



1. Góc con lắc, vận tốc góc con lắc, vận tốc xe



**Tài liệu tham khảo**

**Trang web:**

<https://scribd.com>

<https://ctms.engin.umich.edu/CTMS/index.php?example=InvertedPendulum&section=ControlPID&fbclid=IwAR3hK-X9_N0I0Pf-Muo2_qwjxYyaqBL_Pd6vaQX95El2aMBu3pbm9DwMwnU>