

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA TOÁN - CƠ - TIN HỌC

Đỗ Văn Chung

Tính toán chuẩn H_∞
thông qua công thức không gian trạng thái

LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC

Ngành Toán Tin ứng dụng
(Chương trình đào tạo: Chuẩn)

Hà Nội - 2021

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA TOÁN - CƠ - TIN HỌC

Đỗ Văn Chung

Tính toán chuẩn H_∞ của hệ thống điều khiển
~~thông qua công thức không gian trạng thái~~
Bằng phương pháp miền tần số

LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC

Ngành Toán Tin ứng dụng
(Chương trình đào tạo: Chuẩn)

Cán bộ hướng dẫn: TS Hà Phi

Hà Nội - 2021

Lời cảm ơn

Luận văn này được hoàn thành dưới sự hướng dẫn của TS Hà Phi.

Tôi xin chân thành cảm ơn TS Hà Phi đã hướng dẫn và cho những lời khuyên quý báu để tôi có thể hoàn thành bài luận văn này. Tôi cũng chân thành cảm ơn các thầy cô trong Khoa Toán - Cơ - Tin học đã cung cấp và truyền đạt cho tôi những kiến thức và kinh nghiệm quý giá trong quá trình học tập.

Do thời gian và trình độ còn hạn chế nên luận văn của tôi không thể tránh khỏi những thiếu sót. Tôi rất kính mong nhận được sự góp ý của thầy cô và bạn bè để bài luận văn này được hoàn thiện hơn.

Tôi xin chân thành cảm ơn.

Hà Nội, ngày 24 tháng 12 năm 2020,

Sinh viên: Đỗ Văn Chung

Lớp K62 Toán - Tin ứng dụng.

Mục lục

1	Giới thiệu	3
2	Kiến thức chuẩn bị	5
2.1	Lập công thức không gian trạng thái	5
2.2	Nghiem không gian-trạng thái	6
2.3	Hàm truyền	7
2.4	Khả năng điều khiển và khả năng quan sát	8
2.5	Tính ổn định và khả năng phát hiện	10
2.6	Chuẩn H_∞	12
3	Thuật toán: Miền tần số	13
3.1	Tính chuẩn H_∞	13
3.2	Thuật toán phân đôi	13
3.3	Thuật toán hai bước	15
4	Kết luận	17

~~Tính toán chuẩn H_∞~~ ~~thông qua công thức không gian trạng thái~~

~~Đỗ Văn Chung K62 Toán Tin ứng dụng~~

1. Giới thiệu

Một trong những đòi hỏi đặt ra trong lý thuyết điều khiển chính là việc thiết kế hệ thống điều khiển, giúp cho máy móc có một hiệu suất làm việc hợp lý dưới nhiều điều kiện đầu vào và các yếu tố gây nhiễu khác nhau. Lý thuyết điều khiển H_∞ được sử dụng để có thể giảm các lỗi mô hình hóa và các nhiễu không xác định trong một hệ thống, đồng thời cung cấp khả năng tối ưu hóa một cách định lượng được đối với một bài toán quy mô lớn có nhiều biến tham gia. Trong thực tế hầu hết các ~~phương pháp của vấn đề sử dụng điều khiển H_∞~~ thực ra chỉ là các bộ điều khiển dưới mức tối ưu. Nghĩa là hàm chuyển của bộ điều khiển đạt đến một giới hạn định trước với chuẩn H_∞ . Trong ~~lập công thức cho bài toán, kết quả sẽ có chuẩn mạnh~~ đối với một giới hạn cho trước. Tuy nhiên, nếu ta tìm thấy chuẩn H_∞ , thì ta sẽ có câu trả lời cho sự tồn tại của bộ điều khiển trong một ~~hiệu động nhất định~~ đối với hệ thống.

George Zames, [23] đưa ra lý thuyết điều khiển H_∞ bằng cách xây dựng việc giảm độ nhạy như một vấn đề trong tối ưu với toán tử chuẩn, cụ thể là chuẩn H_∞ [15]. Tại đó H_∞ là không gian của mọi ~~các hàm giá trị ma trận giải tích~~, bị giới hạn trong nửa ~~phẳng phức bên phải~~. Công thức này hoàn toàn dựa trên miền tần số. Zames gợi ý rằng việc sử dụng chuẩn H_∞ làm thước đo hiệu suất sẽ đáp ứng tốt hơn nhu cầu ứng dụng so với thiết kế điều khiển Gaussian bậc hai tuyến tính phổ biến hơn [5]. Từ đó, ta có thiết kế của điều khiển tối ưu H_∞ nhằm mục đích tìm ra một bộ điều khiển có thể ổn định được hệ thống trong khi giảm thiểu tác động của nhiễu gây ra. Chuẩn H_∞ hiện được sử dụng để đánh giá ~~một cách số lượng~~ độ nhạy, độ mạnh và hiệu suất của bộ điều khiển của hệ thống phản hồi vòng kín.

Ngay sau đó, John Doyle đã phát triển các công cụ để kiểm tra độ ổn định chắc trong nỗ lực đánh giá độ không đảm bảo của mô hình và đạt được mục tiêu điều khiển H_∞ , giảm thiểu ảnh hưởng của những nhiễu động này lên các đầu ra được điều chỉnh [18]. Vào năm 1984, Doyle sử dụng các phương pháp trong không gian trạng thái, và đưa ra giải pháp đầu tiên cho bài toán điều khiển tối ưu H_∞ đa biến tổng quát [6].

Trong khi các công thức miền tần số này giải quyết các vấn đề về độ vững chắc, các công cụ được phát triển đều được cho là phức tạp và cung cấp cái nhìn hạn chế về bản chất của các giải pháp ổn định. Các công thức không gian trạng thái ban đầu được phát triển bởi Doyle, Francis và Glover vào giữa những năm 1980, nhưng chỉ thành công ở một mức độ nào đó, chúng là các giải pháp cho các công thức ban đầu này có các giải pháp bậc cao, xem [8] và [11].

Vào cuối những năm 1980, công trình nghiên cứu của Khargonekar, Petersen, Rotea và Zhou, [13], [14], và [20], đã trình bày giải pháp cho vấn đề phản hồi trạng thái H_∞ dưới dạng ma trận phản hồi liên tục và cung cấp công thức để tìm ma trận khuếch đại này như là đáp án cho một phương

trình Riccati. Hơn nữa, họ đã thiết lập mối quan hệ giữa lý thuyết trò chơi tuyến tính bậc hai và điều khiển tối ưu H_∞ và đưa ra giải pháp cho vấn đề phản hồi trạng thái bằng cách giải một phương trình Riccati đại số.

Năm 1989, Doyle, Glover, Khargonekar và Francis, [7], đã trình bày giải pháp không gian trạng thái tổng quát đầu tiên cho bài toán điều khiển H_∞ trong bài báo kinh điển *State-space solutions to standard H_2 and H_∞ control problems*. Bài báo này đưa ra các kết quả cần và điều kiện đủ để tồn tại một bộ điều khiển có thể chấp nhận được các nghiệm của phương trình Riccati đại số và điều kiện ghép liên quan đến điều kiện của bán kính phổ. Do đó, công thức tạo ra các thuật toán sử dụng tiêu chí lồi cũng như các phương pháp giống như phương pháp Newton, xem [9], [10], và [21].

Nghiên cứu này tập trung vào tập hợp các thuật toán được sử dụng cho mục đích điều khiển H_∞ , đặc biệt là cho bài toán phản hồi trạng thái, [3], [4] và các vấn đề phản hồi đầu ra. Sau đó, thuật toán được trình bày bởi Lin, Wang và Xu [16].

Phần thảo luận chính được tìm thấy trong Chương 4, nơi ta thảo luận và thực hiện thuật toán của Lin, Wang và Xu [16] được đề xuất để tính chuẩn H_∞ của trạng thái và các vấn đề điều khiển phản hồi đầu ra. Thuật toán sử dụng công thức không gian trạng thái như là một công cụ miền tần số để tìm ra nguyên tố r mà chuẩn H_∞ của ma trận hàm chuyển nhỏ hơn r . Bài toán tìm chuẩn H_∞ được chia thành ba bài toán con. Hai giải pháp đầu tiên chủ yếu dựa vào các thuật toán được thảo luận trong Chương 3 và giải pháp thứ ba dựa vào bản chất hình học của bán kính quang phổ của tích của hai nghiệm cho Phương trình Riccati Đại số liên quan đến hai bài toán con đầu tiên. Phần kết luận được trình bày trong Chương 5.

Phương pháp miền tần số là một trong hai phương pháp chính để nghiên cứu chuẩn H_∞ của hệ thống điều khiển, chính vì vậy trong luận văn này chúng ta sẽ đi tìm hiểu hai thuật toán kinh điển đầu tiên, được trình bày trong các bài báo [3, 4] để tính toán chuẩn này. Các thuật toán này vẫn còn được tham khảo và trích dẫn cho đến nay (2021).

2. Kiến thức chuẩn bị

2.1 Lập công thức không gian trạng thái

Ta xét hệ ~~động~~ ~~không gian~~ ~~bất biến~~ liên tục **sau:** **vứt hết : thể này đi**

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t), \quad x(t_0) = x_0 \\ y(t) &= Cx(t) + Du(t).\end{aligned}\tag{2.1.1}$$

trong đó A là ma trận cỡ $n \times n$, B cỡ $n \times m$, C là $r \times n$ và D cỡ $r \times m$. Vector x biểu diễn trạng thái, u là vector điều khiển và y là đầu ra ~~đã được tính toán~~. Ta minh họa bằng sơ đồ khối sau

Ví dụ 2.1. Để ~~có thể diễn tả~~ **minh họa** rõ hơn việc lập công thức không gian trạng thái, ta xét hệ 2 vật nặng được nối với nhau bởi một lò xo và giảm chấn. Lực u tác động vào vật M_1 , làm thay đổi vị trí z của vật M_2 . Đầu vào của hệ này là lực tác động, đầu ra chính là ~~sự thay đổi~~ **đầu ra** vị trí của vật M_2 . Ở đây thứ ta cần quan tâm chính là việc ta có thể điều khiển được ~~sự thay đổi~~ vị trí của M_2 . Áp dụng định luật II Newton, ta có:

$$M_1 \ddot{w} = -b(\dot{w} - \dot{z}) - k(w - z) + u\tag{2.1.2}$$

$$M_2 \ddot{z} = b(\dot{w} - \dot{z}) + k(w - z).\tag{2.1.3}$$

Ở đây, b là hệ số tắt dần, w và z liên tục là sự dịch chuyển M_1 và M_2 , k là ~~hằng số~~ **độ cứng** của lò xo. Ta đưa hệ đạo hàm bậc 2 này về bậc 1 bằng cách: **Đặt $x_1 = w$, $\dot{x}_1 = \dot{w} = x_2$, $x_3 = z$, $\dot{x}_3 = \dot{z} = x_4$** . Khi đó,

Chỉnh lại hết
 \dot{x}_1
thành
 \dot{x}_1
và tương tự với x khác

$$\dot{x}_2 = -\frac{b}{m_1}(x_2 - x_4) - \frac{k}{m_1}(x_1 - x_3) + u,\tag{2.1.4}$$

$$\dot{x}_4 = \frac{b}{m_2}(x_2 - x_4) + \frac{k}{m_2}(x_1 - x_3) + u,\tag{2.1.5}$$

$$y = x_3\tag{2.1.6}$$

Viết ~~(2.1.4)-(2.1.6)~~ **lại** hệ dưới dạng ma trận: **bỏ hết mấy cái : đi**

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k}{m_1} & -\frac{b}{m_1} & \frac{k}{m_1} & \frac{b}{m_1} \\ 0 & 0 & 0 & 1 \\ \frac{k}{m_2} & \frac{b}{m_2} & -\frac{k}{m_2} & -\frac{b}{m_2} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m_1} \\ 0 \\ 0 \end{bmatrix} \cdot u,\tag{2.1.7}$$

$$y = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}.\tag{2.1.8}$$

Ta đưa hệ lò xo-giảm chấn này về dạng (2.1.1), ta **có:** $A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k}{m_1} & -\frac{b}{m_1} & \frac{k}{m_1} & \frac{b}{m_1} \\ 0 & 0 & 0 & 1 \\ \frac{k}{m_2} & \frac{b}{m_2} & -\frac{k}{m_2} & -\frac{b}{m_2} \end{bmatrix}$ $B = \begin{bmatrix} 0 \\ \frac{1}{m_1} \\ 0 \\ 0 \end{bmatrix}$ **sửa như thầy comment trên**

những chỗ nào người ta
xuống dòng thì mình cũng
phải xuống

$$\begin{bmatrix} 0 \\ \frac{1}{m_1} \\ 0 \\ 0 \end{bmatrix} \quad C = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix} \quad D = 0.$$

2.2 Nghiệm không gian-trạng thái

Nếu ta đã biết vector u , hệ (2.1.1) có thể giải được ~~nhẹ nhàng~~. Trước khi đi vào cụ thể, ta cần biết một vài định nghĩa [5] . ~~thiếu chấm~~

Định nghĩa 2.2. Cho ma trận A cỡ $n \times n$, khi đó ta định nghĩa **hàm mũ ma trận** là

$$e^{At} = \sum_{k=0}^{\infty} \frac{(At)^k}{k!} \quad \text{lại thiếu chấm}$$

Hai điều lưu ý về hàm mũ ma trận:

$$1, e^{A0} = A^0 = I$$

$$2, \frac{d}{dt} e^{At} = \sum_{k=0}^{\infty} \frac{d}{dt} \frac{A^k t^k}{k!} = \sum_{k=1}^{\infty} \frac{A^k t^{k-1}}{(k-1)!} = \sum_{k=0}^{\infty} A \frac{A^k t^k}{k!} = A e^{At}.$$

~~Để dàng tìm được nghiệm đối với phương trình ban đầu~~
Ta biết nghiệm của bài toán giá trị ban đầu

$$\dot{x}(t) = Ax(t), \quad x(t_0) = x_0, \quad (2.2.1)$$

là $x(t) = e^{A(t-t_0)} x_0$. Trở lại với phương trình tổng quát

$$\dot{x}(t) = Ax(t) + Bu(t), \quad x(t_0) = x_0 \quad (2.2.2)$$

$$y(t) = Cx(t) + Du(t). \quad (2.2.3)$$

ta sẽ tìm nghiệm qua biến đổi Laplace.

Định nghĩa 2.3 (Biến đổi Laplace). Hàm $f(x)$, $x \in [0, \infty)$ có biến đổi Laplace là tích phân $L(f(x))(s) = \hat{f}(s) = \int_0^{\infty} f(x) e^{-sx} dx$ với giá trị $s \in \mathbb{C}$ mà tích phân ~~định~~ ^{có} nghĩa. ~~đúng~~ \mathcal{L}

Hàm Laplace ngược của \hat{f} là $L^{-1}\hat{f} = f$. Ta biến đổi Laplace (2.2.2) tìm nghiệm \hat{x} là

$$\hat{x}(s) = (sI - A)^{-1} x_0 + (sI - A)^{-1} B \hat{u}(s) \quad (2.2.4)$$

Áp dụng hàm Laplace ngược (2.2.4) cho ta nghiệm

$$\begin{aligned} x(t) &= L^{-1}((sI - A)^{-1} x_0) + L^{-1}((sI - A)^{-1} B \hat{u}(s)) \\ &= e^{A(t-t_0)} x_0 + \int_{t_0}^t e^{A(t-s)} B u(s) ds. \end{aligned} \quad (2.2.5)$$

Dịch lại, **ầu 0** chấp nhận được

~~Lấy (2.2.5) trừ đi (2.2.3)~~ ta tìm được $y(t) = C e^{A(t-t_0)} x_0 + \int_{t_0}^t C e^{A(t-s)} B u(s) ds + Du(t)$

Định nghĩa 2.4. Ma trận $e^{A(t-t_0)}$ được gọi là ma trận chuyển trạng thái.

~~Nếu ta chưa biết trước được vector u thì ta có thể sử dụng nhiều phương pháp khác nhau để cho ra kết quả như ý muốn.~~

2.3 Hàm truyền

Hàm chuyển sẽ cho ta thấy mối quan hệ mật thiết giữa đầu vào và đầu ra của hệ thống và sự ảnh hưởng của nhiễu đối với đầu ra như thế nào. Để tìm ra hàm chuyển của hệ (2.1.1)

$$\dot{x}(t) = Ax(t) + Bu(t), \quad x(0) = x_0 \quad (2.3.1)$$

$$y(t) = Cx(t) + Du(t)$$

ta sử dụng biến đổi Laplace như (2.2.4) thu được nghiệm

$$\hat{x}(s) = (sI - A)^{-1}x_0 + (sI - A)^{-1}B\hat{u}(s) \quad (2.3.2)$$

$$\hat{y}(s) = C\hat{x}(s) + D\hat{u}(s). \quad (2.3.3)$$

Dịch sai như trên

Lấy (2.3.2) trừ cho (2.3.3) và đặt $G(s) = \frac{\hat{y}(s)}{\hat{u}(s)}$. Phương trình (2.3.3) trở thành

$$\hat{y}(s) = G(s)\hat{u}(s)$$

hay

$$G(s) = \frac{\hat{y}(s)}{\hat{u}(s)} \quad (2.3.4)$$

Hàm $G(s)$ phản ánh tỷ số giữa đầu ra hệ thống $\hat{y}(s)$ và đầu vào hệ thống $\hat{u}(s)$ và được gọi là hàm truyền ma trận hoặc đơn giản chỉ là ma trận truyền. Thành phần G_{ij} trong ma trận biểu diễn hàm truyền từ đầu vào thứ j sang đầu ra thứ i . Hàm truyền này đại diện cho các thuộc tính của hệ thống và không đại diện cho độ lớn hoặc bản chất của các yếu tố đầu vào. Nó cũng không phục vụ cho việc cung cấp bất kỳ thông tin nào về cấu trúc của hệ thống.

Một ưu điểm chính của việc sử dụng hàm truyền là chúng cho phép sử dụng một hệ thống phức tạp trong miền thời gian và biểu diễn nó như một hàm của một biến duy nhất trong miền tần số. Thật không may, cách tiếp cận này chỉ hoạt động đối với các hệ thống đầu ra - đầu vào duy nhất (SISO). Điều này có nghĩa là, đối với nhiều hệ thống đầu vào - nhiều đầu ra, thì sẽ có nhiều hàm truyền được sử dụng để biểu diễn mối quan hệ đầu vào - đầu ra. Cần lưu ý rằng các chức năng chuyển giao không duy nhất, nhiều hệ thống có thể chia sẻ cùng một hàm chuyển, không liên quan đến độ lớn của hệ thống hoặc bản chất của đầu vào. ta làm việc với đơn

Định nghĩa 2.5. Các điểm p mà tại đó hàm truyền $G(p) = \infty$ được gọi là các cực của hệ.

Nếu $G(\infty)$ là một ma trận hằng thì hàm truyền được gọi là **chính** và nếu $G(\infty) = 0$, hàm truyền được gọi là **riêng**. Chúng ta hãy sử dụng hệ thống giảm chấn khối lượng lò xo từ Ví dụ (2.1.1) để minh họa cách tìm hàm truyền của một hệ đã cho được mô tả bằng biểu diễn không gian trạng thái.

Ví dụ 2.6. Từ ví dụ (2.1.1) ta có $A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k}{m_1} & -\frac{b}{m_1} & \frac{k}{m_1} & \frac{b}{m_1} \\ 0 & 0 & 0 & 1 \\ \frac{k}{m_2} & \frac{b}{m_2} & -\frac{k}{m_2} & -\frac{b}{m_2} \end{bmatrix}$ $B = \begin{bmatrix} 0 \\ \frac{1}{m_1} \\ 0 \\ 0 \end{bmatrix}$

$$C = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix} \quad D = 0.$$

Suy ra

$$G(s) = C(sI - A)^{-1}B + D = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix} \left(\begin{bmatrix} s & 0 & 0 & 0 \\ 0 & s & 0 & 0 \\ 0 & 0 & s & 0 \\ 0 & 0 & 0 & s \end{bmatrix} - \begin{bmatrix} 0 & 1 & 0 & 0 \\ -4 & -2 & 4 & 2 \\ 0 & 0 & 0 & 1 \\ 4 & 2 & -4 & -2 \end{bmatrix} \right)^{-1} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}.$$

Giản ước đi ta tìm ra

$$G(s) = \frac{s+2}{s^4+3s^3+6s^2} \quad \text{Chấm đầu}$$

Định nghĩa 2.7. Trong không gian thời gian - trạng thái biểu diễn ở (2.3.1), ma trận

$$G(i\omega) = c(i\omega I - A)^{-1}B + D \quad (2.3.5)$$

được gọi là **ma trận trả tần số**, $\omega \in \mathbb{R}$ chính tần số.
phản hồi tần số

2.4 Khả năng điều khiển và khả năng quan sát được

Định nghĩa 2.8. Hệ (2.1.1)

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t) + Du(t) \end{aligned}$$

được gọi là có thể điều khiển được nếu hệ thống đạt được bất kỳ trạng thái $x_1 = x(t_1)$, trong thời gian $\leq t_1$, từ bất kỳ trạng thái ban đầu $x(0)$ bằng cách chọn điều khiển $u(t)$, $0 \leq t \leq t_1$ phù hợp tương ứng.

Dịch lại Khả năng kiểm soát của hệ thống (2.1.1), được gọi là khả năng kiểm soát của cặp (A, B) . Định lý sau cung cấp các điều kiện có thể kiểm chứng được về việc một hệ thống có thể điều khiển được hay không.

Định lý 2.9. Cho $A \in \mathbb{R}^{n \times n}$ và $B \in \mathbb{R}^{n \times m} (m \leq n)$. Các điều sau đây là tương đương:

1. Hệ (2.1.1) có thể điều khiển được.
2. Ma trận điều khiển $n \times nm$ là $C_M = [B, AB, A^2B, \dots, A^{n-1}B]$ có hạng đầy đủ. đủ hạng dòng
3. Ma trận

$$W_C = \int_0^{t_1} e^{At} B B^T e^{A^T t} dt$$

không suy biến với bất kỳ $t_1 > 0$.

4. Nếu (λ, x) là 1 cặp giá trị riêng và vector riêng tương ứng của A^T , thì $x^T B \neq 0$.
5. $\text{Rank}(A - \lambda I, B) = n$ với mọi giá trị riêng λ của A .
6. Các giá trị riêng của $A - BK$ có thể được gán tùy ý bằng một lựa chọn thích hợp của K .

Định nghĩa 2.10. Hệ thời gian liên tục (2.1.1) gọi là ~~có thể quan sát được~~ nếu tồn tại $t_1 > 0$ sao cho trạng thái ban đầu $x(t_0)$ xác định duy nhất nếu biết $u(t)$ và $y(t)$ với mọi $0 \leq t \leq t_1$.

~~Khả năng quan sát~~ của hệ thống (2.1.1) được gọi là ~~khả năng quan sát~~ của cặp (A, C) . Đôi ngấu có nghĩa là (A, C) có thể quan sát được nếu (A^T, C^T) có thể kiểm soát được. Do tính hai mặt của khả năng quan sát và khả năng điều khiển, khả năng quan sát có các đặc điểm tương tự như tính điều khiển được trong Định lý (2.9). ~~ngoặc chỉ dùng cho phương trình thời~~ Phải xem bản gốc chứ

Định lý 2.11. Những điều sau đây là tương đương:

1. Hệ (2.1.1) có thể quan sát được.
2. Ma trận của khả năng quan sát $O_M = [CCA \dots CA^{n-1}]^T$ có hạng đủ.
3. Ma trận

$$W_O = \int_0^{t_0} e^{A^T t} C^T C e^{A t} dt$$

không suy biến với mọi t_1 dương.

4. Ma trận $\begin{bmatrix} \lambda I - A \\ C \end{bmatrix}$ có hạng bằng n với mọi giá trị riêng của A .

5. Nếu (λ, y) là cặp riêng của A , thì $Cy \neq 0$. xem ở trên

6. Các giá trị riêng cho $A - LC$ có thể được gán tùy ý bằng một lựa chọn thích hợp của L .

Khả năng điều khiển và khả năng quan sát đóng vai trò quan trọng trong sự tồn tại của các nghiệm xác định dương và ~~bán~~ xác định dương cho phương trình Lyapunov.

Định nghĩa 2.12. Cho A là một ma trận ổn định, thì

$$C_G = \int_0^{t_1} e^{A t} B B^T e^{A^T t} dt \quad (2.4.1)$$

gọi là ~~khả năng điều khiển Gram~~. Gramian điều khiển

Định nghĩa 2.13. Cho A là một ma trận ổn định, thì

$$O_G = \int_0^\infty e^{A^T t} C^T C e^{A t} dt \quad (2.4.2)$$

được gọi là ~~khả năng quan sát Gram~~. Gramian quan sát

Định lý 2.14. Cho A là I ma trận ổn định. Khả năng điều khiển Gram C_G thỏa mãn phương trình Lyapunov

$$A C_G + C_G A^T = -B B^T \quad (2.4.3)$$

và là đối xứng, xác định dương

~~và xác định dương đối xứng~~ khi và chỉ khi (A, B) có thể điều khiển được.

Định lý 2.15. Cho A là 1 ma trận ổn định. Khả năng quan sát Gram O_G thỏa mãn phương trình Lyapunov

$$O_G A + A^T O_G = -C^T C \quad (2.4.4)$$

sửa nt

~~và xác định dương đối xứng~~ khi và chỉ khi (A, B) có thể quan sát được.

dịch lại chỗ
bôi vàng

Sau đây là một ví dụ minh họa tính toán của cả 2 giá trị C_G và O_G .

Ví dụ 2.16. Trở lại hệ (2.1.1), với

$$A = \begin{bmatrix} -4 & -8 & -12 \\ 0 & -8 & -4 \\ 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, C = [1 \quad 1 \quad 1], D = 0.$$

Ma trận A ổn định và sử dụng lệnh Matlab **lyap** để giải các phương trình Lyapunov trong Định lý (2.14) và (2.15), chúng ta tìm được

$$C_G = \begin{bmatrix} 0.1458 & 0.0208 & 0.0208 \\ 0.0208 & 0.0833 & 0.0833 \\ 0.0208 & 0.0833 & 0.0833 \end{bmatrix} = O_G$$

Ma trận này suy biến, nên (A, B) không điều khiển được và (C, D) không quan sát được.

vì sao biết

Tính ổn định hóa được và khả năng phát hiện được

2.5 ~~Tính ổn định và khả năng phát hiện~~

Một bộ điều khiển thường được thiết kế để ổn định hóa hệ thống tuyến tính. Điều này được thực hiện bằng cách chọn một điều khiển u thích hợp sao cho ma trận vòng kín $A - BK$ ổn định, trong đó K là ma trận phản hồi mà $u(t) = Kx(t)$. Trong phần này, chúng ta xác định tính ổn định của một hệ thống và trình bày các đặc điểm của hệ thống có thể được ổn định thông qua điều khiển phản hồi.

Đầu tiên chúng ta chọn hệ chưa điều khiển

$$\dot{x}(t) = Ax(t), \quad x(0) = x_0. \quad (2.5.1)$$

Định nghĩa 2.17. Trạng thái cân bằng của hệ (2.5.1), là vectơ x_e thỏa mãn

$$Ax_e = 0 \quad (2.5.2)$$

Nếu ma trận A không suy biến, phương trình trên chỉ có nghiệm tầm thường $x_e = 0$.

Định nghĩa 2.18. Gọi x_e là trạng thái cân bằng của (2.5.1), thì x_e được gọi là

1. Ổn định nếu với mỗi $\epsilon > 0$ tồn tại $\delta > 0$ sao cho $\|x(t) - x_e\| < \epsilon, \forall t \geq 0$ nếu $\|x(t_0) - x_e\| < \delta$.
2. ~~Cân~~ ổn định ^{tiệm cận} nếu trạng thái cân bằng ổn định và $\exists \delta > 0$ sao cho $\|x(t_0) - x_e\| < \delta$ ~~hay~~ dẫn đến $\lim_{t \rightarrow \infty} x(t) = x_e$.

Điều này có nghĩa là mọi nghiệm đúng của hệ nếu bắt đầu đủ gần điểm cân bằng **cân ổn định**, thì không chỉ giữ nguyên tại điểm đó, mà sẽ tiến gần đến điểm cân bằng khi thời gian tăng. Nên nếu $x_e = 0$ và A không suy biến, hệ không **kiểm soát** ở trên **tiệm cận cân bằng** nếu $x(t) \rightarrow 0$ khi $t \rightarrow \infty$. Các giá trị riêng của ma trận A trong (2.5.1) ~~mô tả~~ ^{đặc trưng cho} độ ổn định của hệ thống.

Định lý 2.19. Hệ thống (2.5.1) **tiệm cận ổn định** khi và chỉ khi mọi giá trị riêng của A có phần thực âm ~~tuyệt đối~~ ^{thực sự}.

Định lý 2.20. Ma trận A được gọi **là ma trận ổn định** nếu mọi giá trị riêng của A có phần thực âm ~~tuyệt đối~~ ^{thực sự}.

Cũng như với ~~Về~~ tính điều khiển, nghiệm của phương trình Lyapunov cũng ^{đặc trưng} ~~biểu thị~~ cho tính ổn định của (2.5.1).

Định lý 2.21. *Hệ*

$$\dot{x}(t) = Ax(t)$$

tiệm cận ổn định nếu và chỉ nếu, với ma trận M đối xứng dương bất kỳ, thì phương trình sau có nghiệm duy nhất

$$XA + A^T X = -M.$$

Xét hệ tuyến tính (2.1.1),

$$\dot{x}(t) = Ax(t) + Bu(t),$$

$$y(t) = Cx(t) + Du(t).$$

Điều đó có nghĩa là

Ta xét việc tìm nghiệm điều khiển u để đưa hệ sang một trạng thái mong muốn. Hay tìm 1 ma trận phản hồi K sao cho $A - BK$ ổn định. Không phải mọi hệ có thể đưa về một trạng thái nhất định, nên điều quan trọng nhất của phần này là tìm ra điều kiện cần và đủ sao cho tồn tại một ma trận phản hồi ổn định.

Hơn nữa giả sử ta có thể tìm được mọi thông tin của trạng thái $x(t)$ ở mọi thời gian t bất kỳ. Sử dụng điều khiển tuyến tính, ta suy ra

$$u(t) = -Kx(t) \quad (2.5.3)$$

Thay $u(t) = -Kx(t)$ cho (2.1.1) ta được là hệ khép kín:

$$\dot{x}(t) = (A - BK)x(t), \quad (2.5.4)$$

$$y(t) = (C - DK)x(t). \quad (2.5.5)$$

Ta nhắc lại hệ không kiểm soát (2.5.1) ổn định khi ma trận A ổn định. Vậy nên việc làm ổn định hệ (2.5.4) quy về việc tìm ma trận K sao cho $(A - BK)$ là ổn định. Nếu $\exists K$, thì việc tìm K được gọi là *việc ổn định ma trận phản hồi* và ta đã ổn định được hệ thống. Cấp ma trận (A, B) được gọi là **cấp ma trận ổn định**. hóa được.

Những định lý sau đây cho ta điều kiện cần và đủ để đánh giá việc có thể ổn định được ma trận cho 1 hệ đang xét. một ma trận phản hồi ổn định hóa tồn tại.

Định lý 2.22. Trong việc ổn định, những khẳng định sau là tương đương:

1. Cấp ma trận (A, B) ổn định. là ổn định hóa được
2. $\text{Rank}(A - \lambda I, B) = n, \forall \text{Re}(\lambda) \geq 0$.
3. $\forall \lambda$ và $x \neq 0$, nếu $x^* A = \lambda x^*$ và $\text{Re}(\lambda) \geq 0$ thì $x^* B \neq 0$.

Hệ quả 2.23. Nếu cấp (A, B) là điều khiển được, thì nó cũng ổn định hóa được.

Lưu ý khả năng điều khiển suy ra khả năng ổn định chỉ là chiều suy ra mà không suy ngược lại. Cũng như khả năng quan sát đi cùng với khả năng điều khiển, khả năng ổn định của hệ thống cũng đi cùng với khả năng phát hiện.

dịch lại câu này

là đối ngẫu của tính

Định nghĩa 2.24. Hệ được gọi là ~~có thể~~ phát hiện nếu tồn tại ma trận L sao cho cặp ma trận (A, C) , $A - LC$ là ổn định. ~~hóa được~~

Định lý 2.25. Các điều sau đây là tương đương:

1. (A, C) ~~có thể~~ là phát hiện. ~~được~~
2. Ma trận $\begin{bmatrix} \lambda I - A \\ C \end{bmatrix}$ có hạng đầy đủ, $\forall \operatorname{Re}(\lambda) \geq 0$.
3. $\forall \lambda$ và $x \neq 0$ sao cho $Ax = \lambda x$ và $\operatorname{Re}(\lambda) \geq 0$, thì $Cx \neq 0$.
4. Cặp (A^T, C^T) ~~có thể~~ là ổn định. ~~hóa~~

Có thể thấy rằng một hệ thống có thể điều khiển được thì các giá trị riêng của vòng lặp kín có thể được gán tùy ý (bài toán gán giá trị riêng). Vấn đề với việc chỉ định các giá trị riêng tùy ý đó là không có phương pháp luận để tìm một bộ điều khiển tối ưu sao cho các đặc điểm thiết kế hệ thống có thể đáp ứng một cách hiệu quả nhất có thể.

Điều này dẫn đến nhiều phương pháp điều khiển tối ưu như điều khiển tối ưu tuyến tính bậc hai. Phương pháp này tối thiểu hóa hàm ~~mất mát~~ bậc 2 đã được xác định trước. Tuy nhiên, lưu ý [7] cho ta sự phát triển song song của điều khiển H_∞ và điều khiển H_2 tối ưu hay còn được gọi là điều khiển ~~Gauss bậc 2 tuyến tính~~. ~~dạng toàn phương tuyến tính Gauss~~.

2.6 Chuẩn H_∞

Trước tiên, hãy lưu ý rằng hàm truyền của một hệ thống ~~ổn định~~ bất kỳ được mô hình hóa bởi phương trình vi phân thời gian tuyến tính ~~hệ số hằng là hàm~~ ~~bất biến thông thường có giá trị hữu tỉ và có các hệ số thực như trong [17] và [1].~~ Vì vậy, các hàm truyền của bất kỳ hệ thống nào như vậy đều có các cực nằm hoàn toàn trong nửa mặt phẳng bên trái và ~~được phân tích~~ ~~là giải tích~~ trong nửa mặt phẳng bên phải. Một không gian của các ~~hàm~~ ~~chức năng~~ như vậy được gọi là không gian Hardy, ký hiệu là H_∞ . Đối với hệ đa biến bất kỳ, hàm truyền $G(i\omega)$ ~~Là một~~ ma trận. Các giá trị ~~suy biến~~ ~~kí di~~ của ma trận A , $\sigma_j(A)$ được định nghĩa là:

$$\sigma_j(A) = \lambda_j(AA^T).$$

trong đó $\lambda_j(E)$ là giá trị riêng thứ j của ma trận **E.Chuẩn** Euclid của ma trận là: **Chính tả viết sai lung tung**

$$\|A\| = \max_i \sigma_i(A)$$

Suy ra:

$$\|G(i\omega)\| = \max_i \sigma_i(G(i\omega)) = \sigma_{\max}(G(i\omega)).$$

Định nghĩa 2.26. Chuẩn H_∞ của **hàm chuyển $G(s)$** được định nghĩa là

$$\|G\|_\infty = \sup_{\omega \in \mathbb{R}} \sigma_{\max}(G(i\omega)), \quad (2.6.1)$$

~~mà sup là cận trên đúng của mọi giá trị thực của tần số ω~~

Chuẩn này cung cấp một cách khác để ~~mô tả~~ ~~phân tích~~ tính ổn định của hệ thống, cụ thể là nếu hệ thống ~~chuyển hàm~~ $G(s) \in H_\infty$ thì hệ thống ổn định. Như được mô tả trong [15] cho trường hợp khi G là hữu tỉ, $G \in H_\infty$ khi và chỉ khi G không có cực trong nửa mặt phẳng đóng bên phải.

CHƯƠNG

3. Thuật toán: Miền tần số

3.1 Tính chuẩn H_∞

Trong phần này, tôi trình bày hai thuật toán để tìm chuẩn H_∞ của một hệ thống điều khiển được trình bày phương trình vi phân thường. Trong khi các thuật toán sử dụng các công cụ miền tần số, chúng cũng quan trọng trong việc hiểu thuật toán không gian trạng thái được trình bày ở phần sau. Xấp xỉ chuẩn H_∞ của một hệ thống yêu cầu tính toán supremum của tần số đáp ứng trên tất cả tần số, do đó việc sử dụng quy trình lặp lại là bắt buộc. Kết nối giữa chuẩn H_∞ của một hàm truyền ổn định (tức là không có cực trong nửa mặt phẳng bên phải) và liên hệ của nó với ma trận Hamilton của nó đóng một vai trò quan trọng trong thuật toán cho trường hợp phản hồi trạng thái được trình bày sau đây. Xét hệ

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t) + Du(t).\end{aligned}\tag{3.1.1}$$

Định nghĩa 3.1. Ta định nghĩa ma trận Hamilton M_r là

$$M_r = \begin{bmatrix} A + BR^{-1}D^TC & BR^{-1}B^T \\ -C^T(I + DR^{-1}D^T)C & -(A + BR^{-1}D^TC)^T \end{bmatrix}\tag{3.1.2}$$

với $R = r^2I - D^TD$. Đặt R vào trong \$ \$ cho nó in nghiêng

Định lý 3.2. Xét $G(s)$ là hàm truyền ổn định và $r > 0$. Khi đó $\|G\|_\infty < r$ nếu và chỉ nếu $\sigma_{\max}(D) < r$ và M_r không có giá trị riêng thuần ảo.

Định lý này cho ta biết nếu $r > r^* = \|G(s)\|_\infty$ thì M_r sẽ không có giá trị riêng thuần ảo nào. Ngược lại nếu $r < r^*$ thì M_r sẽ có ít nhất một giá trị riêng thuần ảo.

3.2 Thuật toán phân đôi

Dịch thiếu cả 1 đoạn dài

Để bắt đầu thuật toán, ta cần tính giới hạn trên và giới hạn dưới của khoảng chứa r . Ta có thể sử dụng $r_{lb} = 0$ và đặt r_{ub} đủ lớn và ngay lập tức tiến hành thuật toán phân đôi. Tuy nhiên, để có được giới hạn chặt hơn, người ta có thể sử dụng các giá trị đơn Hankel của hệ.

Định nghĩa 3.3. Các giá trị đơn Hankel là căn bậc hai của các giá trị riêng của ma trận $C_G O_G$, với C_G và O_G lần lượt là giá trị Grammian của khả năng kiểm soát và khả năng quan sát, được định nghĩa trong (2.14) và (2.15). sửa lại như chương 2

Kí hiệu các giá trị Hankel là σH_i , được sắp xếp thứ tự giảm dần với σH_1 là giá trị lớn nhất.

lại dịch thiếu Công thức Enns-Glover trong bước 1 của thuật toán dưới đây có thể được tính bằng công thức sau:

$$\begin{aligned}r_{lb} &= \max(\sigma_{\max}(D), \sqrt{\text{Trace}(O_G C_G)/n}) \\ r_{ub} &= \sigma_{\max}(D) + 2\sqrt{n \text{Trace}(C_G O_G)}\end{aligned}\tag{3.2.1}$$

Thuật toán 3.4. Xét hệ (3.1.1) với các hệ số A, B, C, D , trong đó A ổn định và sai số $\epsilon > 0$

Đánh lại thuật toán theo mẫu cho đẹp

<https://nhdien.wordpress.com/2011/09/22/algorithms-sty-bi%E1%BB%83u-di%E1%BB%85n-thu%E1%BA%ADt-toan/>

Bước 1: Tính toán các cận trên và cận dưới r_{ub} và r_{lb}

$$r_{lb} = \max\{\sigma_{\max}(D), \sigma H_1\} \quad (3.2.2)$$

$$r_{ub} = \sigma_{\max}(D) + 2 \sum_{j=1}^n \sigma H_j$$

Bước 2: Đặt $r = (r_{lb} + r_{ub})/2$

Nếu $2(r_{ub} - r_{lb}) < \epsilon$, dừng thuật toán.

Bước 3: Tính M_r

Bước 4: Nếu M_r có giá trị riêng thuần ảo, đặt $r_{lb} = r$. Ngược lại đặt $r_{ub} = r$.

Lại ăn bớt dịch

Ta thấy thuật toán này cần nhân 2 ma trận cỡ $n \times n$ và tính giá trị riêng của kết quả phép nhân. Và thuật toán phụ thuộc quá nhiều vào việc tính chính xác giá trị riêng.

Ví dụ 3.5. Xét hệ (3.1.1) với các hệ số

$$A = \begin{bmatrix} -4 & -8 & -12 \\ 0 & -8 & 0 \\ 0 & 0 & -16 \end{bmatrix}, B = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, C = [1 \quad 1 \quad 1], D = 0$$

Đặt $\epsilon = 10^{-6}$.

Ta tính O_G và C_G bằng lệnh *lyap* trong MATLAB thu được:

$$O_G = \begin{bmatrix} 0.1250 & 0 & 0.1250 \\ 0 & 0.0625 & 0 \\ 0.1250 & 0 & 0.1250 \end{bmatrix}, C_G = \begin{bmatrix} 0.1250 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Sử dụng công thức (3.2.1) cho ra

$$r_{lb} = 0.0722, r_{ub} = 0.433.$$

suy ra $r = 0.2526$ ở bước lặp đầu tiên và

$$M_r = \begin{bmatrix} -4.0000 & -8.0000 & 12.0000 & 15.6375 & 0 & 0 \\ 0 & -8.0000 & 0 & 0 & 0 & 0 \\ 0 & 0 & -16.0000 & 0 & 0 & 0 \\ -1.0000 & -1.0000 & -1.0000 & 4.0000 & 0 & 0 \\ -1.0000 & -1.0000 & -1.0000 & 8.0000 & 8.0000 & 0 \\ -1.0000 & -1.0000 & -1.0000 & -12.0000 & 0 & 16.0000 \end{bmatrix}$$

Các giá trị riêng của M_r là 8.0000, 16.0000, -0.5714, 0.5714, -8.0000, -16.0000. Vì không có giá trị nào là số thuần ảo nên ta đặt $r = r_{ub}$ và tiếp tục vòng lặp. Sau 20 bước lặp, ta tìm được $r_{lb} = 0.2499$ và $r_{ub} = 0.2500$, thỏa mãn điều kiện dừng. Ta tính được $\|G(s)\|_{\infty} \approx .2500$.

3.3 Thuật toán hai bước

Ngay sau khi thuật toán phân đôi được trình bày, một thuật toán hai bước đã được đề xuất bởi Bruinsma và Steinbuch [4], trong đó ta chỉ cần tính cận dưới. Cần lưu ý rằng một thuật toán tương tự đã được Boyd và Balakrishnan [3] đưa ra cùng thời điểm. Điều cần lưu ý là tính toán cận dưới không yêu cầu tìm O_G hoặc C_G , do đó tính toán ít tốn kém hơn. Thuật toán của Bruinsma và Steinbuch cũng dựa vào **Định lý (3.2)** cũng như định lý sau.

Định lý 3.6. Giả sử $r > \sigma_{\max}(D)$ và $\omega \in \mathbb{R}$. Khi đó $\det(M_r - \omega I) = 0$ khi và chỉ khi $\sigma_n(G(\omega i)) = r$ với ~~hầu hết số n~~ một số n nào đó.

Hệ quả của định lý này cho ta biết ~~nếu~~ ωi là ~~1~~ giá trị riêng của M_r khi và chỉ khi r là ~~một~~ giá trị ~~ki~~ ~~di~~ ~~suy~~ ~~biến~~ của $G(\omega i)$ với $\omega \in \mathbb{R}$. **Dịch thiếu đoạn "Trong Hình 3.1 ..."**

Thuật toán của Bruinsma và Steinbuch được xây dựng dựa trên ý tưởng của thuật toán phân đôi, nhưng sử dụng **Định lý (3.6)** để tìm kiếm $\|G(s)\|_\infty$ bằng cách sử dụng nhiều giá trị của ω . Thuật toán phân đôi chỉ tìm kiếm ở một tần số mỗi lần lặp, trong khi thuật toán hai bước tìm kiếm nhiều tần số mỗi lần lặp. Đây là ưu điểm chính của việc sử dụng thuật toán này. **Lại dịch sót câu "The authors claim ..."**

Để bắt đầu thuật toán, ta đặt

$$r_{lb} = \max \{ \sigma_{\max} G(0), \sigma_{\max} G(\omega_p i), \sigma_{\max}(D) \} \quad (3.3.1)$$

để tìm cận dưới với $\omega_p =$

$|\lambda_{\max}|$ và ω_i là cực của hàm chuyển $G(s)$, được chọn theo tiêu chí sau: **dịch vừa sai vừa thiếu câu này**

$$\left| \frac{\operatorname{Im}(\lambda_i)}{\operatorname{Re}(\lambda_i)} \frac{1}{\lambda_i} \right|. \quad (3.3.2)$$

Nếu $G(s)$ có các cực hoàn toàn là số thực, thì λ_i là cực làm cực đại hóa $\left| \frac{1}{\lambda_i} \right|$.

Với ma trận A là ổn định Vừa dịch vừa ăn bót à Dịch lại từ đây và đừng có ăn bót nữa

Thuật toán 3.7. Cho ma trận ~~Với~~ hệ (3.1.1) với các hệ số ~~ma trận~~ A, B, C, D trong đó A ổn định.

Bước 1.1 : Tính toán hàm chuyển $G(s)$.

Tôi 0 đọc vì dịch sai và ăn bót quá nhiều

Bước 1.2 : Tìm cực của $G(s)$

Nếu các cực hoàn toàn là số thực thì đặt $\omega_p = \max |\lambda_i|$. Ngược lại đặt $\omega_p = \lambda_i$, với λ_i cực đại hóa (3.3.2).

Bước 1.3 : Tính r_{lb} bằng (3.6).

Bước 2.1 : Tính $r = (1 + 2\epsilon)r_{lb}$

Bước 2.2 : Tính M_r bằng (3.1.2).

Sắp xếp các giá trị riêng thuần ảo của M_r theo thứ tự giảm dần ω_i, ω_k .

Nếu M_r không có giá trị riêng thuần ảo nào, đặt $r_{ub} = r$ và chuyển sang bước 3.

Bước 2.3 : Từ $i = 1$ đến $k - 1$

(i) Tính $m_i = \frac{1}{2}(\omega_i + \omega_{i+1})$.

(ii) Tính $\omega_{\max}(G(m_i i)) = \operatorname{svd}_i$.

Đặt $r_{lb} = \max(\operatorname{svd}_i)$.

Bước 3 Tính $\|G\|_\infty = \frac{1}{2}(r_{lb} + r_{ub})$.

Bằng cách tính σ_{max} với $\forall i$, thuật toán hai bước tìm kiếm ở một khoảng rộng hơn nhiều với thuật toán phân đôi. Sau khi tìm $\sigma_{max}(G(m_i))$ với mọi i , ta đặt giá trị này bằng r_{lb} và bắt đầu bước lặp tiếp theo.

Ví dụ 3.8. Có $A = \begin{bmatrix} -1.0000 + 1.0000i & -1.0000 & -1.0000 \\ -1.0000 & -2.0000 + 1.0000i & -1.0000 \\ -1.0000 & -1.0000 & -2.0000 - 1.0000i \end{bmatrix}$. Các cực của

$G(s)$ là các giá trị riêng của A . Các giá trị đó là $-3.3589 + 0.2858i$, $-0.3628 + 0.9534i$, $-1.2784 - 0.2392i$. Vậy ma trận A ổn định.

Tiếp theo ta thấy $-0.3628 + 0.9534i$ cực đại hóa (3.3.2) và sử dụng (3.6) ta tính được $r_{lb} = 0.9907$.

Ta tính ma trận Hamilton

$$M_r = \begin{bmatrix} -1+i & -1 & -1 & 1.0188 & 0 & 0 \\ -1 & -2+i & -1 & 0 & 0 & 0 \\ -1 & -1 & -2-i & 0 & 0 & 0 \\ -1 & -1 & -1 & 1+i & 1 & 1 \\ -1 & -1 & -1 & 1 & 2+i & 1 \\ -1 & -1 & -1 & 1 & 1 & 2-i \end{bmatrix}$$

Tính các giá trị riêng của M_r ta được:

$$-3.1927 + 0.2164i,$$

$$3.1927 + 0.2164i,$$

$$1.32240.1563i,$$

$$1.32240.1563i,$$

$$0.0000 + 0.8597i,$$

$$0.0000 + 1.0201i$$

Ta tính và sắp xếp các giá trị riêng thuần ảo, $\omega_1 = 0.0000 + 0.8597i$ và $\omega_2 = 0.0000 + 1.0201i$.

Tính trung bình của 2 giá trị trên ta thu được $m_1 = 0.0000 + 0.9399i$.

Giá trị đơn lớn nhất của $G(m_1i) = 1.0121 = r_{lb}$.

Ta tiếp tục chạy vòng lặp đến vòng số 4 ta đạt được $\|G\|_\infty = 1.0121$.

Nhận xét. Ta thấy rằng một vấn đề chính nảy sinh trong việc triển khai thuật toán này là sự cần thiết của việc giải giá trị riêng chính xác, phụ thuộc vào việc các giá trị riêng thuần ảo hay không.

4. Kết luận

Điều hấp dẫn nhất trong bài báo của Lin, Wang và Xu là việc họ sử dụng các công thức miền tần số trong công thức không gian trạng thái. Công thức không gian trạng thái ban đầu được phát triển trong nỗ lực chuyển từ cách tiếp cận miền tần số sang điều khiển H_∞ . Tuy nhiên, Lin Wang và Xu đã biến đổi một số điều kiện cần và đủ của không gian trạng thái như là một tuyên bố về các giá trị suy biến.

Một vấn đề chính nảy sinh từ việc biến đổi này là việc tính toán chính xác các giá trị riêng. Cần lưu ý rằng đối với mục đích của nghiên cứu này, các hàm MatLab được sử dụng để tính toán tất cả các giá trị riêng và giá trị suy biến. Tuy nhiên, điều này đã không xảy ra ở [16] nơi mà các bộ giải riêng chính xác được sử dụng.

Việc tính toán chính xác các giá trị riêng là điều tối quan trọng ở đây. Lý do cho điều này là quá trình ra quyết định vốn có trong thuật toán được trình bày trong Chương 4. Tính chất đối xứng của ma trận Hamilton là yếu tố then chốt trong việc sử dụng công thức không gian trạng thái. Nhiều bộ giải riêng không có khả năng bảo toàn tính chất này và kết quả là độ chính xác trong tính toán của chúng bị mất đi. Đối lại, vì thuật toán ở đây dựa vào việc trả lời loại có hoặc không, xóa toàn bộ khoảng thời gian khỏi một tìm kiếm, dương hoặc âm sai có thể gây ra kết quả sai. Ví dụ, giả sử thuật toán hội tụ đến một giá trị sao cho nó chỉ ra một bộ điều khiển tồn tại một nhiễu lớn hơn so với thực tế nó có thể đáp ứng được. Người ta có thể lãng phí tài nguyên khi triển khai một điều khiển chỉ để thấy rằng nó không hoạt động khi được áp dụng.

Các kế hoạch nghiên cứu trong tương lai bao gồm triển khai các bộ giải riêng chính xác hơn và áp dụng thuật toán này cho một mô hình quy mô lớn từ kiểu thiết lập ứng dụng. Sẽ tốt nếu ta xem liệu thuật toán này có thể cạnh tranh được với các phương pháp khác hiện đang được sử dụng để ước tính chuẩn H_∞ của một hệ thống.

Luận văn đi tìm hiểu hai phương pháp tính toán chuẩn H vô cùng cho các hệ điều khiển tuyến tính hệ số hằng với giả thiết là hệ không điều khiển là ổn định.

Đối với cả hai cách tiếp cận (phương pháp phân đôi, phương pháp hai bước) việc tính toán các giá trị riêng và giá trị kì dị đều được thực hiện bằng các hàm sẵn có trong MATLAB. Cần chú ý thêm rằng việc tính toán chính xác các giá trị riêng và giá trị kì dị có vai trò đặc biệt quan trọng trong việc chuyển từ cách tiếp cận miền tần số sang cách tiếp cận không gian trạng thái, xem [16].

Chỉ những tài liệu được trích dẫn ở trên mới đưa vào đây thôi, còn thì bỏ đi.

Tài liệu

Sau khi bỏ thì phải cập nhật lại chính xác các trích dẫn ở trên. Nói chung chỉ nên có tối đa 7 TLTK

- [1] T. Başar and P. Bernhard. *H-infinity optimal control and related minimax design problems: a dynamic game approach*. Birkhäuser Boston, 2008.
- [2] P. Benner, V. Mehrmann. and H. Xu. A numerically stable, structure preserving method for computing the eigenvalues of real hamiltonian or symplectic pencils. *Numerische Mathematik*, 78(3):329–358, 1998.
- [3] S. Boyd, V. Balakrishnan, and P. Kabamba. A bisection method for computing the H_∞ norm of a transfer matrix and related problems. *Mathematics of Control, Signals, and Systems (MCSS)*, 2(3):207–219, 1989.
- [4] NA Bruinsma and M. Steinbuch. A fast algorithm to compute the H_∞ norm of a transfer function matrix. *Systems & Control Letters*, 14(4):287–293, 1990.
- [5] B. Datta. *Numerical methods for linear control systems*. Academic Press, 2003.
- [6] J.C. Doyle et al, Advances in multivariable control. In *Lecture Notes at ONR/Honeywell Workshop. Minneapolis, MN*, 1984.
- [7] J.C. Doyle, K. Glover, P.P. Khargonekar, and B.A. Francis. State-space solutions to standard H_2 and H_∞ control problems. *Automatic Control, IEEE Transactions on* 34(8):831–847, 1989.
- [8] B.A. Francis and J.C. Doyle. Linear control theory with an H_∞ optimality criterion. *SIAM Journal on Control and Optimization*, 25(4):815–844, 1987.
- [9] P. Gahinet. On the game riccati equations arising in H_1 control problem. *SIAM Journal on Control and Optimization*, 32(3):635–647, 1994.
- [10] P.M. Gahinet and P. Pandey. Fast and numerically robust algorithm for computing the H_∞ optimum. In *Decision and Control, 1991., Proceedings of the 30th IEEE Conference on* pages 200–205. IEEE, 1991.
- [11] K. Glover. All optimal Hankel-norm approximations of linear multivariable systems and their L_∞ error bounds. *International Journal of Control*, 39(6):1115–1193, 1984.
- [12] G. Hewer. Existence theorems for positive semidefinite and sign indefinite stabilizing solutions of H_∞ riccati equations. *SIAM journal on control and optimization*, 31(1):16–29, 1993.
- [13] P.P. Khargonekar, I.R. Petersen, and M.A. Rotea. H_∞ -optimal control with statefeedback. *Automatic Control, IEEE Transactions on*, 33(8):786–788, 1988.
- [14] P.P. Khargonekar, I.R. Petersen, and K. Zhou. Robust stabilization of uncertain linear systems: quadratic stabilizability and H_∞ control theory. *Automatic Control, IEEE Transactions on* 35(3):356–361, 1990.
- [15] D.J.N. Limebeer and M. Green. *Linear robust control..* Prentice-Hall, Englewood Clis, NJ, 1995.
- [16] W.W. Lin, C.S. Wang, and Q.F. Xu. On the computation of the optimal H_∞ norms for two feedback control problems. *Linear algebra and its applications*, 287(1):223–255, 1999.
- [17] K.A. Morris. *Introduction to feedback control*. Academic Press, Inc., 2000.

- [18] A. Packard, M.K.H. Fan, and J. Doyle. A power method for the structured singular value. In *Decision and Control, 1988., Proceedings of the 27th IEEE Conference on* pages 2132–2137. IEEE, 1988.
- [19] P. Pandey, C. Kenney, A. Packard, and AJ Laub. A gradient method for computing the optimal H_∞ -norm. *Automatic Control, IEEE Transactions on*, 36(7):887–890, 1991.
- [20] I. Petersen. Disturbance attenuation and H_∞ optimization: a design method based on the algebraic riccati equation. *Automatic Control, IEEE Transactions on* 32(5):427–429, 1987.
- [21] C. Scherer. H_∞ -control by state-feedback and fast algorithms for the computation of optimal H_∞ -norms. *Automatic Control, IEEE Transactions on* 35(10):1090–1099, 1990.
- [22] D. Tall and S. Vinner. Concept image and concept definition in mathematics with particular reference to limits and continuity. *Educational studies in mathematics*, 12(2):151–169, 1981.
- [23] G.Zames. Feedback and optimal sensitivity: Model reference transformations, multiplicative seminorms, and approximate inverses. *Automatic Control, IEEE Transactions on*, 26(2):301–320, 1981.
- [24] M. Zandieh. A theoretical framework for analyzing student understanding of the concept of derivative. *Research in Collegiate Mathematics Education. IV. CBMS Issues in Mathematics Education*, pages 103–127, 2000.
- [25] K. Zhou, J.C. Doyle, K. Glover, et al. *Robust and optimal control*, volume 40. Prentice Hall Upper Saddle River, NJ, 1996.