

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ENGENHARIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

MICAEL FISCHMANN

SINCRONIZAÇÃO DE CIRCUITO DE CHUA ATRAVÉS DE TÉCNICAS
DE CONTROLE BASEADAS EM LMIS

Porto Alegre

2015

MICAEL FISCHMANN

SINCRONIZAÇÃO DE CIRCUITO DE CHUA ATRAVÉS DE TÉCNICAS
DE CONTROLE BASEADAS EM LMIS

Projeto de diplomação apresentado ao
Departamento de Engenharia Elétrica da
Universidade Federal do Rio Grande do Sul
como requisito para obter a graduação em
Engenharia Elétrica.

ORIENTADOR: Prof. Dr. João Manoel Gomes da Silva Junior

Porto Alegre

2015

MICAEL FISCHMANN

SINCRONIZAÇÃO DE CIRCUITO DE CHUA ATRAVÉS DE TÉCNICAS
DE CONTROLE BASEADAS EM LMIS

Este projeto foi julgado adequado para fazer
jus aos créditos da disciplina “Projeto de
Diplomação” do Departamento de
Engenharia Elétrica e aprovado em sua
forma final pelo Orientador e pela Banca
Examinadora.

ORIENTADOR: Prof. Dr. João Manoel Gomes da Silva Junior

Aprovado em: ___/___/_____

BANCA EXAMINADORA

Prof. Dr. João Manoel Gomes da Silva Junior
Université Toulouse III Paul Sabatier

Prof. Dr. Jeferson Vieira Flores
Universidade Federal do Rio Grande do Sul

Prof. Dr. Diego Eckhard
Universidade Federal do Rio Grande do Sul

Dedico este trabalho aos meus pais e à minha
namorada, os quais sempre me apoiarem.

AGRADECIMENTOS

Aos meus pais, que sempre me apoiaram em todos os momentos, sejam eles fáceis ou difíceis, sempre me encorajando.

À minha namorada, Michele, por estar sempre ao meu lado.

Aos meus colegas e amigos do curso, pelo companheirismo, pelas madrugadas e noites em claro utilizadas para os estudos, por sempre estenderem a mão quando necessitei e me ajudarem a compreender as aulas.

A todos os professores que ajudaram a formar o profissional que sou hoje ao me incentivar a pesquisar e a ir além da matéria.

Ao meu orientador, professor João Manoel, pela discussões e debates sobre o projeto e pelas bibliografias fornecidas.

"Chaos was the law of nature;
Order was the dream of man."
Henry Adams

Resumo

Neste trabalho é apresentado um estudo sobre a sincronização de circuitos caóticos, focando no circuito de Chua. A metodologia utilizada para o projeto do controlador foi baseada em resolução de LMIs, sendo consideradas restrições práticas no equacionamento do problema. O resultado final simulado demonstra que os dois circuitos entram em sincronia perfeita.

Palavras-chave: caos, circuito de Chua, sincronização, LMI

Abstract

This work presents a study about synchronization of chaotic circuits, focusing on Chua's circuit. The controller was projected with the resolution of LMIs, being considered practical restrictions in the problem's equations. The final result simulated shows that the two circuits enter perfect synchrony.

Key-words: chaos, Chua's circuit, synchronization, LMI.

Lista de Ilustrações

Figura 1 - Esquemático de circuito de Chua com diodo de Chua	12
Figura 2 - Diodo de Chua.....	13
Figura 3 – Configuração resistor negativo.....	13
Figura 4 – Comportamento não-linear do diodo de Chua.....	15
Figura 5 – Esquemático do circuito de Chua feito no Microcap	16
Figura 6 - Simulação do circuito de Chua no Microcap (CI: $V_{C1} = V_{C2} = 0$; $I_L = 0$).....	16
Figura 7 – Representação do sistema no Simulink	19
Figura 8 – Simulação do sistema no Simulink ($m = n = o = 10^{-3}$).....	19
Figura 9 – Simulação no plano m-n do sistema no Simulink com novos parâmetros ($m = 0.7$, $n = o = 0$)	21
Figura 10 - Simulação em m-n-o do sistema no Simulink com novos parâmetros ($m = 0.7$, $n = o = 0$).....	22
Figura 11 – Esboço do sistema de controle	25
Figura 12 – Sistema em Malha Fechada.....	39
Figura 13 - Comportamento no plano x_1 - x_2 dos sistemas mestre e escravo com atraso.....	40
Figura 14 - Comportamento temporal dos sinais de erro com atraso.	41
Figura 15 – Comportamento temporal dos estados internos com atraso.....	42
Figura 16 – Comportamento no plano x_1 - x_2 dos sistemas mestre e escravo sem atraso, utilizando (17).....	43
Figura 17 - Comportamento temporal dos sinais de erro sem atraso, utilizando (17).	44
Figura 18 - Comportamento temporal dos estados internos sem atraso ao utilizar (17).	45
Figura 19 – Comportamento no plano x_1 - x_2 dos sistemas mestre e escravo sem atraso, utilizando (23).....	46
Figura 20 - Comportamento temporal dos sinais de erro sem atraso, utilizando (23).	47
Figura 21 - Comportamento temporal dos estados internos sem atraso ao utilizar (23).	48
Figura 22 - Comportamento no plano x_1 - x_2 dos sistemas mestre e escravo com saturação.	50
Figura 23 - Comportamento temporal dos sinais de erro com saturação.....	51
Figura 24 - Comportamento temporal dos estados internos com saturação.....	52
Figura 25 - Saídas dos atuadores	53
Figura 26 – 3 exemplos de sistemas utilizando o critério de Liapunov. Em (a) o estado de equilíbrio é estável, em (b) o estado de equilíbrio é assintoticamente estável e em (c) o estado de equilíbrio é instável.....	59

Lista de Tabelas

Tabela 1 – Parâmetros utilizados na segunda simulação.....	20
Tabela 2 – Estados iniciais para as simulações	39

Lista de Abreviações

LMI - Linear Matrix Inequality

Sumário

1.Introdução	10
2.Circuito Chua	12
3.Projeto do Controlador	25
3.1.Controlador com atraso	27
3.2.Controlador sem atraso	31
3.3 Controlador com Saturação	33
4. Resultados Simulados	39
4.1 Resultados Simulados do Sistema com Atraso.....	40
4.2 Resultados Simulados para o Sistema sem Atraso	42
4.3 Resultados Simulados do Sistema com Saturação.....	49
5.Conclusões Finais	54
Referências Bibliográficas	56
Apêndice A – Estabilidade de Liapunov	57
Apêndice B – LMI.....	63
B.1 – Definição	63
B.2 – Complemento de Schur.....	64
B.3 – S procedure.....	65
Apêndice C – Códigos do MATLAB	66

1.Introdução

Teoria do caos é o campo de estudo de matemática que lida com sistemas dinâmicos não lineares que são altamente sensíveis às condições iniciais. Esse efeito é comumente conhecido como “Efeito Borboleta”, pois pequenas variações nas condições iniciais gerando respostas bastante divergentes, incrementando a diferença com o tempo (Kellert, 1993). Apesar desta característica, sistemas caóticos são determinísticos, não apresentando nenhuma característica estocástica, mas não são preditivos.

Sistemas caóticos possuem alta gama de aplicações práticas, entre elas biologia, criptografia, meteorologia e sistemas de comunicações. Assim fica claro que é importante ter conhecimento de como gerar e realizar o controle de tais sistemas. Este controle é desafiador devido à sensibilidade e à não-linearidade destes. O mais simples sistema caótico que pode ser construído com componentes elétricos é o circuito de Chua.

O circuito de Chua, nome dado em homenagem ao seu criador, Leon O. Chua, é um circuito autônomo renomado pela sua simplicidade. Uma das suas maiores aplicações consiste na modulação de sistemas de comunicação, permitindo, entre outras coisas, uma maior segurança da informação, pois um sinal caótico não pode ser predito facilmente ao ser interceptado. Entretanto, para dispor de tal tecnologia é necessário que o receptor possa demodular a mensagem e, portanto, deve ser capaz de reproduzir o comportamento caótico do sistema.

Para realizar a demodulação, pode-se utilizar um sistema mestre-escravo entre o transmissor (mestre) e os receptores (escravos). Assim pode ser aplicado um controlador robusto para garantir que o circuito de Chua escravo siga à referência do mestre sem erro em regime permanente. Dessa forma, apesar de haver um atraso entre eles, mesmo com o comportamento caótico os sistemas estarão em sincronia, ou seja, possuem saídas e desempenho idênticos.

A sincronização se dará através de um controlador, cujas entradas são os parâmetros do circuito mestre e as saídas serão as entradas do circuito escravo. Para considerar todos os parâmetros o sistema será tratado da forma de espaço de estados. O controlador será projetado com técnicas de LMI, que

também serão utilizadas para garantir os cálculos da estabilidade do sistema de tipo Lure.

No capítulo 2 o circuito de Chua será analisado em sua totalidade, de porquê ele gera sinais caóticos até às equações que o descrevem.

No capítulo 3 será realizado o estudo do controlador, seu projeto e as equações que o regem. O sistema LMI será formado para dois casos, um com e um sem atraso.

No capítulo 4 será feita uma análise do sistema em malha fechada resultante, analisando o regime transitório e o permanente, novamente para esses dois casos.

No capítulo 5 será escrito as conclusões do projeto, bem como propostas para estudos futuros.

O apêndice A contém um resumo sobre estabilidade de Liapunov, teoria que serve como base deste trabalho, garantindo que a saída siga a referência.

No apêndice B está escrito um resumo sobre LMI e os métodos relacionados.

No apêndice C estarão os códigos do MATLAB utilizados.

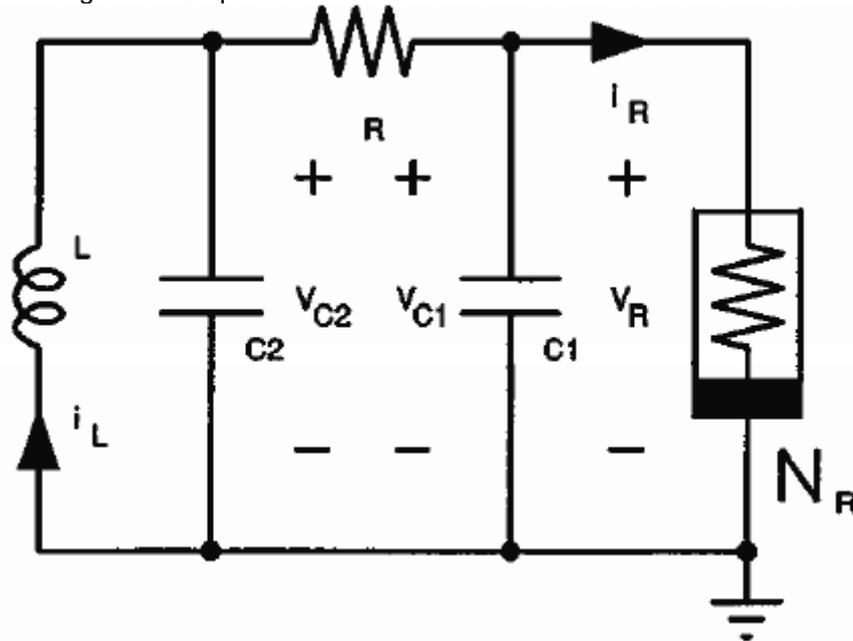
2.Circuito Chua

Para um circuito constituído por resistores, capacitores e indutores possuir comportamento caótico, de acordo com Elgar (1993), precisa cumprir as seguintes condições:

1. possuir pelo menos um componente não linear
2. possuir pelo menos um resistor ativo localmente
3. possuir pelo menos três elementos armazenadores de energia
sendo que um resistor ativo localmente é um dispositivo ativo com resistência equivalente negativa

O circuito de Chua é o circuito mais simples que cumpre as três condições acima. Um esquemático do circuito de Chua está representado na Figura 1. A condição 3 se concretiza por meio de dois capacitores ($C1$ e $C2$) e pelo indutor (L), enquanto que as outras duas condições se concretizam por meio de um dispositivo conhecido como diodo de Chua.

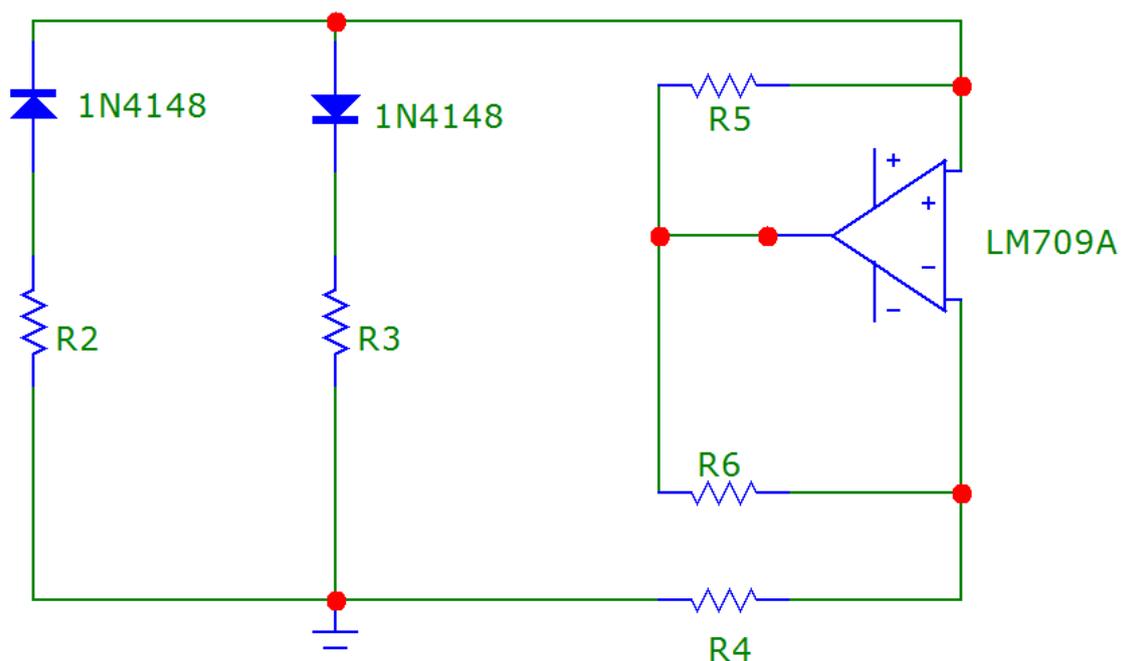
Figura 1 - Esquemático de circuito de Chua com diodo de Chua



Fonte: Elgar, Steve

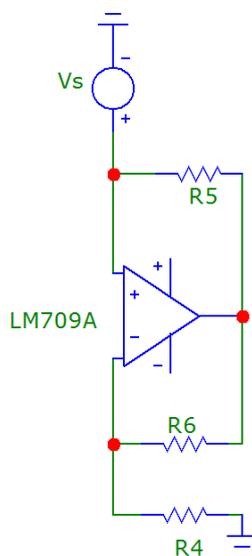
Existem diversas maneiras de se implementar o diodo de Chua, mas a maneira utilizada neste trabalho consiste em duas partes, visualizadas na Figura 2.

Figura 2 - Diodo de Chua



Na primeira são combinadas duas configurações de diodo em série com uma resistência, mas em sentidos opostos, formando assim a não linearidade e cumprindo a condição 1. A segunda parte consiste em um amplificador operacional configurado de tal forma a ter comportamento de uma resistência equivalente de valor negativo. Tal configuração de amplificador está representada na Figura 3.

Figura 3 – Configuração resistor negativo



Pode-se comprovar através de cálculos que o sistema acima comporta-se como um resistor negativo. Como um amplificador não inversor, a sua tensão de saída é descrita por (1):

$$V_{AmpOp} = V_S \left(1 + \frac{R_6}{R_4}\right) \quad (1)$$

Como a corrente que passa por R_3 é igual a $-I_S$, então:

$$I_S = -\frac{V_{AmpOp} - V_S}{R_5} = -\frac{V_S R_6}{R_4 R_5} \quad (2)$$

Combinando (1) com (2):

$$R_{eq} = \frac{V_S}{I_S} = -\frac{R_4 R_5}{R_6} \quad (3)$$

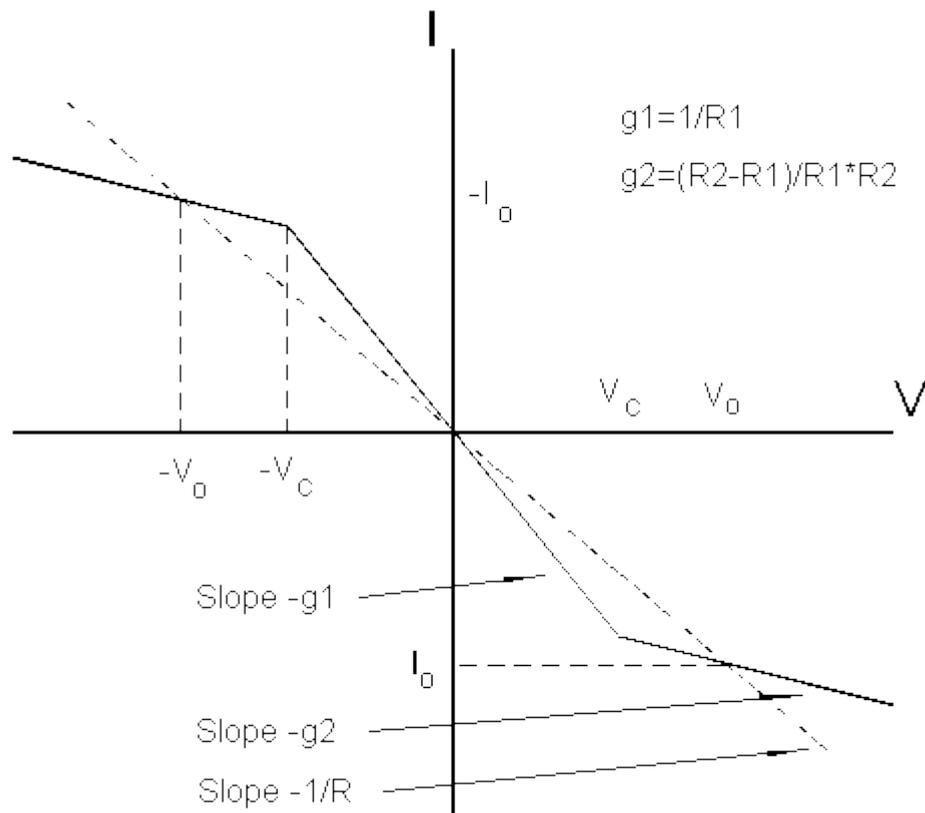
Portanto o comportamento realmente é o de um resistor de valor negativo. Como há um amplificador operacional no circuito, então este componente cumpre os requisitos da condição 2.

O efeito da não-linearidade ocorre quando a tensão nos diodos ultrapassa a tensão limiar de ativação (V_C), colocando o resistor em paralelo com o resistor equivalente negativo, ou seja, a resistência equivalente do diodo de Chua será:

$$\begin{cases} R_{dc} = -R_{eq}, & \text{se } |V| < |V_C| \\ R_{dc} = \frac{R_{eq} R}{(R_{eq} - R_2)}, & \text{se } |V| > |V_C| \end{cases}$$

Este comportamento está ilustrado na Figura 4.

Figura 4 – Comportamento não-linear do diodo de Chua



Fonte: Cross, 1996

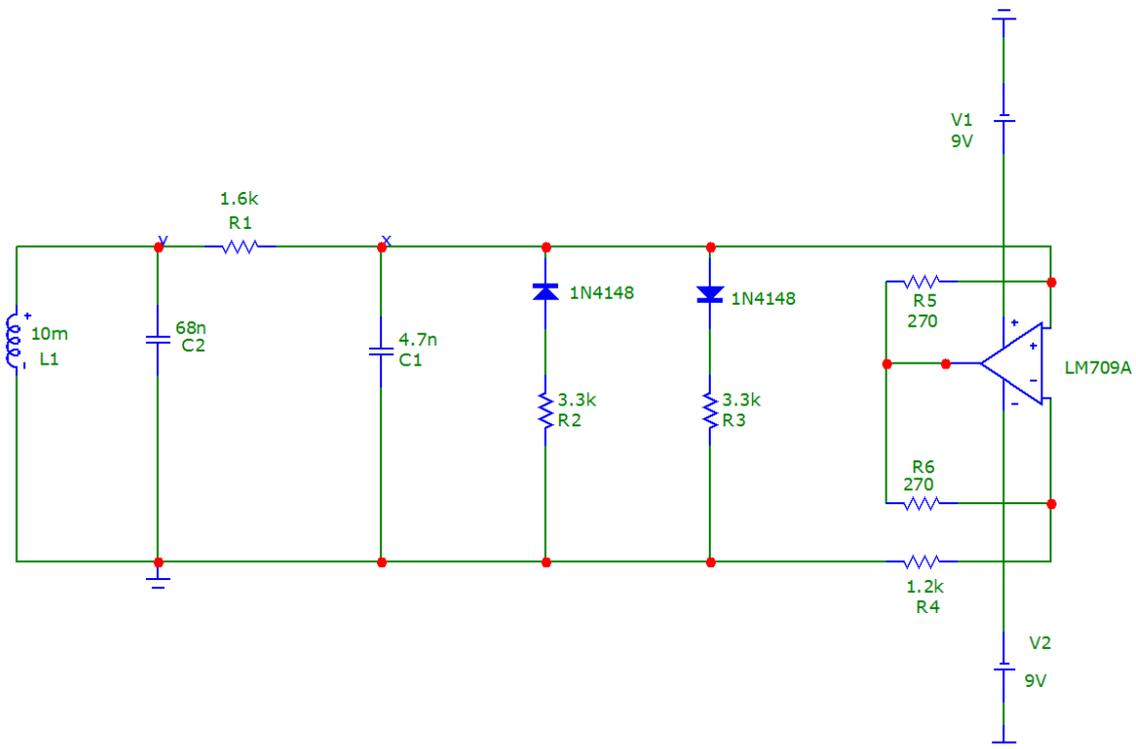
A função não linear que descreve a corrente I do diodo de Chua em função da tensão está representada em (4) e, em forma alternativa, em (5):

$$i_{dc}(V) = \begin{cases} \frac{(R_{eq} - R_2)}{R_{eq}R_2}V + \left(\frac{(R_{eq} - R_2)}{R_{eq}R_2} + \frac{1}{R_{eq}}\right)V_C, & \text{se } V < -V_C \\ -\frac{1}{R_{eq}}V, & \text{se } -V_C \leq V \leq V_C \\ \frac{(R_{eq} - R_2)}{R_{eq}R_2}V - \left(\frac{(R_{eq} - R_2)}{R_{eq}R_2} + \frac{1}{R_{eq}}\right)V_C, & \text{se } V > V_C \end{cases} \quad (4)$$

$$i_{dc}(V) = g_1V + 0.5(g_2 - g_1)(|V + V_C| - |V - V_C|) \quad (5)$$

O circuito final ficará com formato final como demonstra a Figura 5.

Figura 5 – Esquemático do circuito de Chua feito no Microcap



Três simulações do circuito da Figura 5 podem ser vistas nas Figuras 6, 7 e 8, com a tensão no capacitor C_1 no eixo x e a tensão no capacitor C_2 no eixo y, com a tensão no capacitor C_1 no eixo x e a corrente do indutor no eixo y e com a tensão no capacitor C_2 no eixo x e a corrente do indutor no eixo y, respectivamente.

Figura 6 - Simulação do circuito de Chua no Microcap, tensão em C_1 contra tensão em C_2 (CI: $V_{C1} = V_{C2} = 0$; $I_L = 0$)

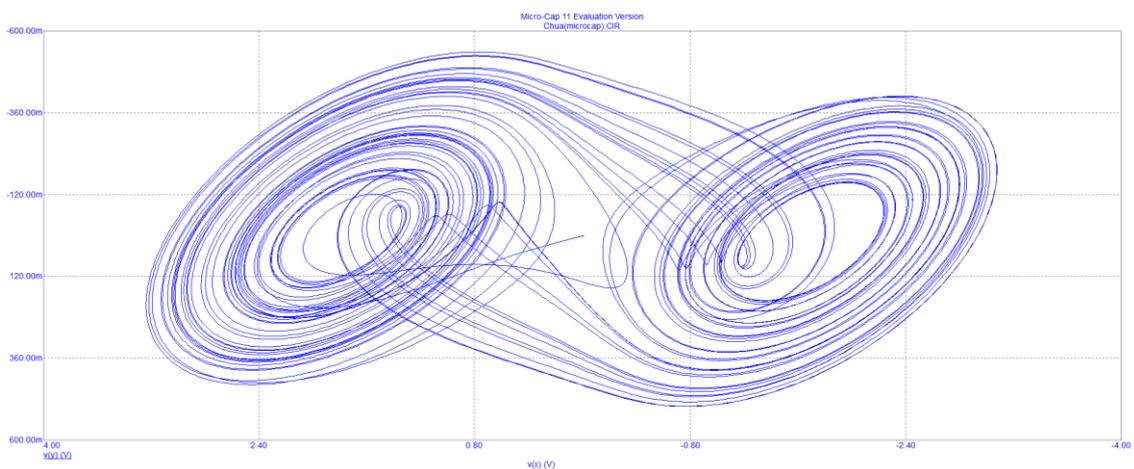


Figura 7 - Simulação do circuito de Chua no Microcap, tensão em C_1 contra corrente no indutor (CI: $VC1 = VC2 = 0; IL = 0$)

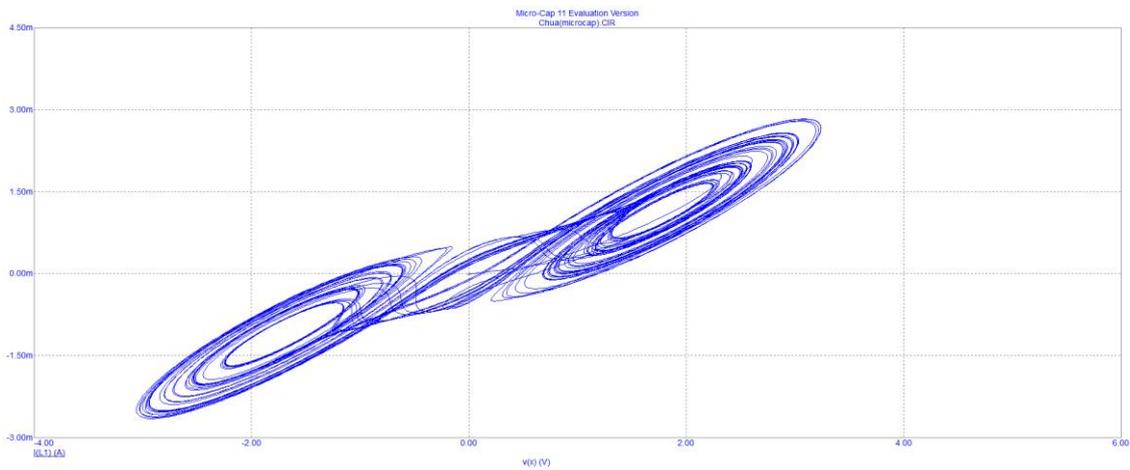
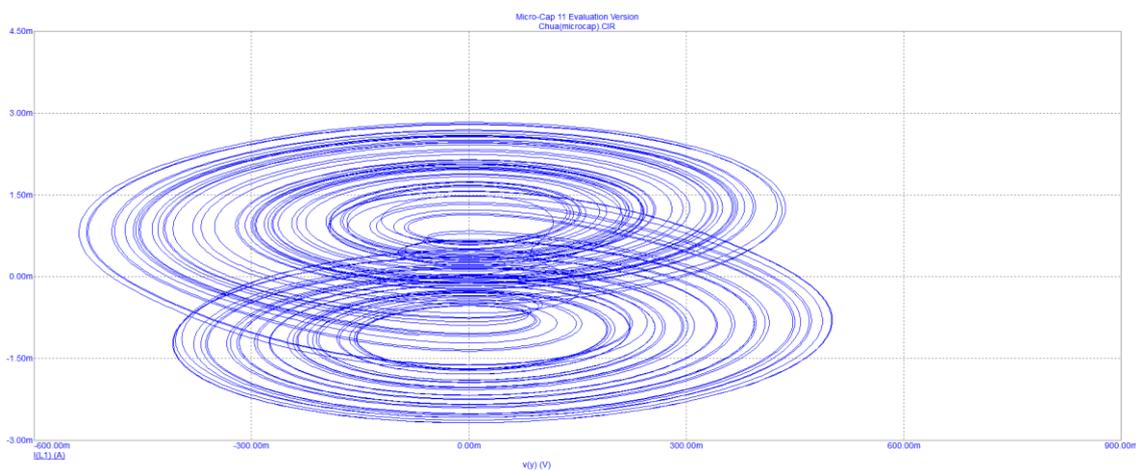


Figura 8 - Simulação do circuito de Chua no Microcap, tensão em C_2 contra corrente no indutor (CI: $VC1 = VC2 = 0; IL = 0$)



Com o circuito montado, é necessário obter uma representação de estados para o mesmo. Para o comportamento caótico do circuito, é relevante obter três informações, as tensões nos capacitores e a corrente no indutor. Utilizando a Lei de Kirchoff das correntes e da tensão, obtém-se as seguintes 3 equações, formando um sistema de equações diferenciais ordinárias não lineares:

$$\begin{cases} C_1 \frac{dV_{C_1}}{dt} = \frac{V_{C_2} - V_{C_1}}{R_1} - i_{dc}(V_{C_1}) \\ C_2 \frac{dV_{C_2}}{dt} = -\frac{V_{C_2} - V_{C_1}}{R_1} + I_L \\ L \frac{dI_L}{dt} = -V_{C_2} \end{cases}$$

É desejável encontrar uma forma sem dimensões para o sistema de equações do circuito. Portanto, ao utilizar as seguintes notações:

$$m = V_{C_1}/V_C$$

$$n = V_{C_2}/V_C$$

$$o = I_L R_1 / V_C$$

$$\tau = \frac{t}{R_1 C_2}$$

$$\alpha = C_2 / C_1$$

$$\beta = \frac{C_2 R_1^2}{L}$$

E isolando as equações diferenciais, agora em relação a τ :

$$\begin{cases} \dot{m} = \alpha(n - m - \widehat{i_{dc}(m)}) \\ \dot{n} = m - n + o \\ \dot{o} = -\beta n \end{cases} \quad (6)$$

Com a função não linear previamente representada na Figura 4 sendo:

$$\widehat{i_{dc}(m)} = R_1 i_{dc}(m) = \hat{g}_1 m + 0.5(\hat{g}_2 - \hat{g}_1)(|m + 1| - |m - 1|)$$

Após obter (6), pode-se simular o sistema encontrado para confirmar que o mesmo gera um sistema caótico. Para isto, será utilizado o recurso Simulink do software MATLAB. A representação do sistema pode ser vista na Figura 9.

O resultado na Figura 10 é de sistema instável ao invés do atrator duplo visto na Figura 6, fato que pode ser resultado de não haver as não idealidades dos diodos e do amplificador operacional, fatores que influenciam no sistema. As capacitâncias parasitas destes elementos contribuem para o valor da capacitância de C_1 , portanto para a simulação no microcap é como se o valor de C_1 fosse incrementado. Estes efeitos são visualizados ao trocar estes componentes no circuito simulado no microcap para um outro modelo do mesmo componente, ou seja, trocar um modelo de amplificador operacional por outro, que para algumas configurações possuem formato similar ao da Figura 10. Outra influência é que o efeito do amplificador operacional de mesmo com tensão de entrada zero no diodo de Chua ter energia, o que por (4) e (5), não aconteceria, pois V e i_{dc} teriam valor nulo, em V e A , respectivamente.

Ao simular com os parâmetros da Tabela 1, diferentes dos utilizados no circuito, obtém-se um resultado mais visivelmente similar ao da Figura 6, comprovando o comportamento caótico do sistema. Esse gráfico pode ser visto em 2D na Figura 11 e em 3D na Figura 12.

Tabela 1 – Parâmetros utilizados na segunda simulação

Parâmetro	Valor
α	15.6
β	28
\widehat{g}_1	-0.714
\widehat{g}_2	-1.143

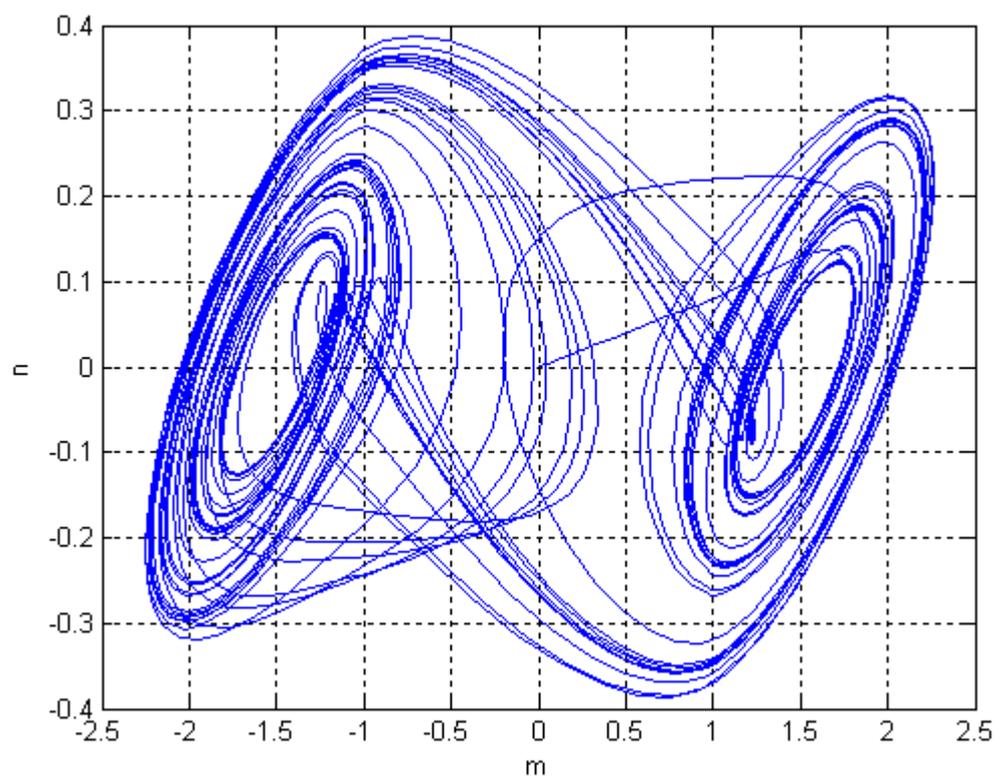
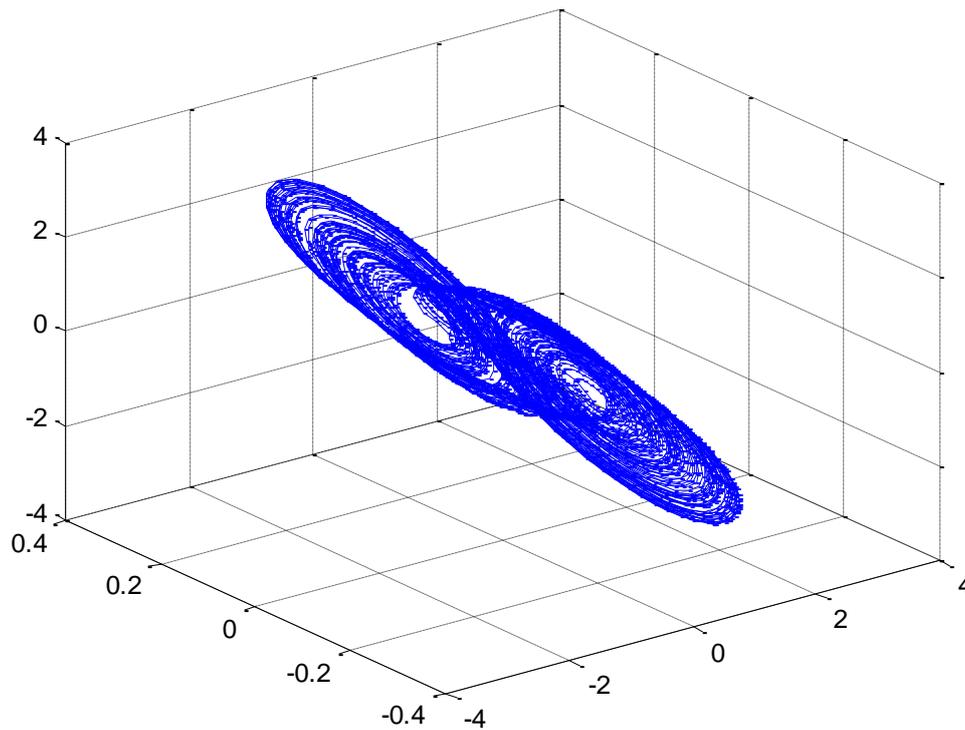
Figura 11 – Simulação no plano m-n do sistema no Simulink com novos parâmetros ($m = 0.7$, $n = 0$)

Figura 12 - Simulação em m-n-o do sistema no Simulink com novos parâmetros ($m = 0.7$, $n = 0$, $o = 0$)



Agora, escrevendo a representação do sistema no espaço de estados, obtém-se:

$$\begin{cases} \dot{x}(t) = Ax(t) + B\widehat{v}_{dc}(C^T x(t)) \\ y(t) = Hx(t) \end{cases} \quad (7)$$

Com

$$x = \begin{bmatrix} m \\ n \\ o \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

$$A = \begin{bmatrix} -\alpha & \alpha & 0 \\ 1 & -1 & 1 \\ 0 & -\beta & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} \alpha \\ 0 \\ 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

E com H sendo uma matriz arbitrária que relaciona a saída do sistema com seus estados internos. Como para o sistema é interessante os estados em si, a matriz H geralmente terá três colunas e número de linhas referente ao número de estados para serem representados. Para este trabalho, a matriz H será da forma:

$$H = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

De tal modo que $y(t)$ seja da seguinte forma:

$$y(t) = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

E lembrando que $\widehat{v}_{ac}(m) = \hat{g}_1 m + 0.5(\hat{g}_2 - \hat{g}_1)(|m + 1| - |m - 1|)$. Ao escolher os valores de forma que \hat{g}_2 tenha módulo maior que \hat{g}_1 , a função \widehat{v}_{ac} pertencerá ao setor $[-|\hat{g}_2|, 0]$, ou seja, seus valores estarão localizados sempre entre duas retas, uma que passa na origem com inclinação 0 e uma que passa na origem com inclinação de valor \hat{g}_2 , já que este parâmetro é negativo, respectivamente. Assim, a função possui a seguinte relação:

$$\widehat{v}_{ac}(m)(\widehat{v}_{ac}(m) + |\hat{g}_2|m) \leq 0$$

Que pode ser reescrita como:

$$\widehat{v}_{ac}(m)^2 + |\hat{g}_2|m\widehat{v}_{ac}(m) \leq 0$$

Como $\widehat{v}_{ac}(m)$ é uma função escalar, pode ser escrito da seguinte forma:

$$\widehat{v}_{ac}(m)^T \widehat{v}_{ac}(m) + |\hat{g}_2|m\widehat{v}_{ac}(m) \leq 0 \quad (8)$$

Esta equação será utilizada futuramente para gerar o S procedure.

3.Projeto do Controlador

Como mencionado anteriormente, aplicações de circuitos caóticos dependem da capacidade de sincronizar dois ou mais circuitos, com um mestre e um ou mais escravos. O desafio é de tornar o comportamento destes iguais ao do mestre mesmo que tenham condições iniciais distintas. Para este propósito o conjunto mestre-escravo pode ser considerado como um sistema de controle.

Um sistema de controle possui uma referência a ser seguida, uma planta a ser controlada, uma realimentação para calcular o erro entre a referência e saída da planta e um controlador para agir sobre este erro. Como deseja-se modificar o comportamento dos circuitos escravos, estes serão a planta do sistema, enquanto que a saída do circuito mestre dita o comportamento desejado e, portanto, será a referência. Portanto, é necessário haver uma realimentação da saída do escravo para o controlador, onde será feito cálculo do erro. O sistema que deve ser utilizado para a sincronização deve ser similar ao proposto na Figura 13.



É ainda importante levar em consideração, efeitos que podem prejudicar ou até causar a completa dessincronização do circuito. Um destes efeitos é a presença de um atraso de propagação entre o circuito mestre e o escravo. Além disso, tal atraso não apenas pode variar em cada transmissão, mas também pode não ser o mesmo para os diferentes estados do sistema. Isso significa que, para o caso estudado, é possível que o atraso seja diferente para cada um dos elementos dos vetores de saída.

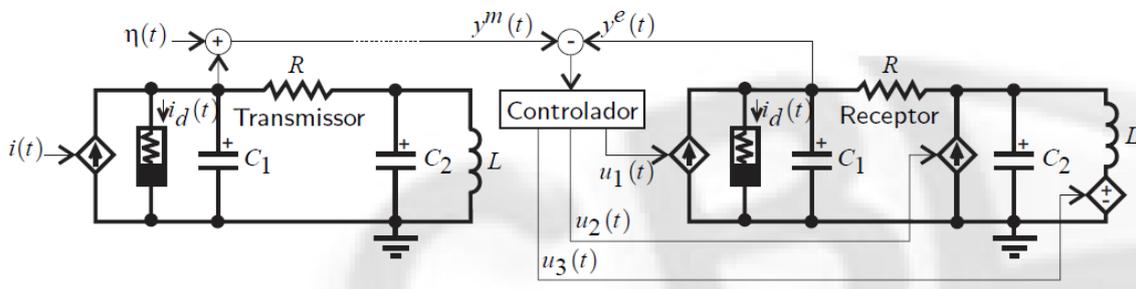
Considerando a representação em espaço de estado (7), pode-se considerar o conjunto de sistemas a serem considerados seja da seguinte forma:

$$\begin{aligned}
\mathfrak{M}: & \begin{cases} \dot{x}(t) = Ax(t) + B\widehat{v}_{dc}(C^T x(t)) \\ y(t) = Hx(t) \end{cases} \\
\mathfrak{E}: & \begin{cases} \dot{\hat{x}}(t) = A\hat{x}(t) + B\widehat{v}_{dc}(C^T \hat{x}(t)) + u(t) \\ \hat{y}(t) = H\hat{x}(t) \end{cases} \\
\mathfrak{C}: & \{u(t) = K \begin{bmatrix} y_1(t - \tau_1(t)) - \hat{y}_1(t - \tau_1(t)) \\ y_2(t - \tau_2(t)) - \hat{y}_2(t - \tau_2(t)) \end{bmatrix}\}
\end{aligned} \quad (9)$$

sendo \mathfrak{M} o sistema mestre, \mathfrak{E} o sistema escravo, \mathfrak{C} o controlador, onde τ_1 e τ_2 são os atrasos variantes no tempo e K é o ganho do controlador, com $K \in \mathbb{R}^{3 \times 2}$, para que $u(t)$ seja um vetor 3×1 .

O sinal resultante do controlador, $u(t)$, será aplicado ao circuito escravo como fontes de tensão e de corrente controláveis, utilizando-se fontes controláveis de corrente para os capacitores e uma fonte controlável de tensão para o indutor. Um esquemático de uma configuração possível de circuito transmissor-receptor se encontra na Figura 14 (Campos, 2007). Nesse circuito, a informação a ser transmitida corresponde a uma corrente gerada por uma fonte controlável denominada $i(t)$ em paralelo com um dos capacitores do circuito e utiliza-se um estado para realizar a sincronização. Quando a sincronização for concluída, a corrente que regula o capacitor do circuito receptor cuja posição é análoga ao capacitor responsável pela informação, neste caso $u_1(t)$, e a corrente $i(t)$ serão idênticos.

Figura 14 - Circuito exemplo do sistema mestre escravo.



Fonte: Campos, 2007

Definindo o erro de sincronização como sendo $e(t) = x(t) - \hat{x}(t)$, segue que o sistema que rege o erro dinâmico deve ser:

$$\begin{cases} \dot{e}(t) = \dot{x}(t) - \hat{\dot{x}}(t) \\ u(t) = KH \begin{bmatrix} e_1(t - \tau_1(t)) \\ e_2(t - \tau_2(t)) \end{bmatrix} \end{cases} \quad (10)$$

$$\begin{cases} \dot{e}(t) = Ae(t) + B\rho(C^T e(t), \hat{x}(t)) - u(t) \\ u(t) = KE_1 He(t - \tau_1(t)) + KE_2 He(t - \tau_2(t)) \end{cases}$$

com $\rho(C^T e(t), \hat{x}(t)) = \widehat{v}_{dc}(C^T e(t) + C^T \hat{x}(t)) - \widehat{v}_{dc}(C^T \hat{x}(t))$, $E_1 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$ e $E_2 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$. O sinal do controlador tem como entrada apenas dois dos sinais do erro, não levando em conta o erro gerado por x_3 . Isso se deve pelo fato de que no circuito real x_3 corresponde à corrente no indutor, que é relativamente mais difícil de se obter uma medida do que os outros dois sinais, que são as tensões dos capacitores. Não foi escolhida apenas uma saída para haver mais graus de liberdade para a solução da futura LMI. Entretanto, como $e(t)$ irá para zero, garante-se que os três estados estarão em sincronia.

O conjunto estará em sincronia quando $e(t)=0$, ou seja, busca-se $\lim_{t \rightarrow \infty} e(t) \rightarrow 0$. Para isso, primeiro é necessário garantir a estabilidade do sistema. Como é um sistema não linear, será utilizado o método de Lyapunov.

Para o procedimento utilizado para o primeiro caso, controlador com atraso, será utilizado o método descrito por (Wen, 2005), utilizando um funcional de Lyapunov-Krasovskii (Papachristodoulou, 2009) para satisfazer as condições de estabilidade.

. Após esse procedimento, será realizado uma versão sem as componentes de atraso e, por último, uma versão baseada no método de (Wen, 2005), mas adicionando componentes responsáveis pela saturação dos atuadores.

3.1. Controlador com atraso

Primeiramente será considera o sistema com atraso de transporte. Devido ao atraso será utilizado o funcional descrito por (Wen, 2005):

$$V(t) = e^T(t)Pe(t) + 2\lambda \int_0^{C^T e(t)} \widehat{\iota}_{dc}(s)ds + \sum_{j=1}^2 \int_{-\tau_j}^0 \int_{t+\theta}^t e^T(s)Z_j \dot{e}(s)dsd\theta \quad (11)$$

onde as matrizes P , Z_j e o escalar λ devem ser determinados, respeitando as restrições $P = P^T > 0$, ou seja, é uma matriz positiva definida, $Z_j = Z_j^T > 0$ e $\lambda \geq 0$.

Como ρ é baseada em $\widehat{\iota}_{dc}$, ela segue as mesmas condições de setor representada em (8). Utilizando uma aplicação do S procedure (Boyd, 1994) na equação quadrática (8) e na sua semelhante de ρ :

$$0 \geq \tilde{S} = 2(-e^T|\hat{g}_2|CS_1\widehat{\iota}_{dc} - \widehat{\iota}_{dc}^T S_1\widehat{\iota}_{dc} - e^T|\hat{g}_2|CS_2\rho - \rho^T S_2\rho) \quad (12)$$

onde $S_1 > 0$, $S_2 > 0$ e o fator 2 existe para a criação de uma matriz simétrica no futuro. A desigualdade (12) origina-se na restrição de setor descrita e representada por (8) que também é válida para ρ . Como \tilde{S} é estritamente negativa, ao fazer com que $\dot{V}(t) - \tilde{S} < 0$, isso significará que a derivada de $V(t)$ será estritamente negativa, satisfazendo o critério de estabilidade de Lyapunov.

Utilizando a fórmula de Leibniz-Newton em conjunto com (10), para um dado escalar δ e para as matrizes de dimensões apropriadas R , N e W_j , com $j = 1, \dots, 6$ seguem as seguintes relações:

$$\Delta_1 = 2[e^T R + \delta \dot{e}^T R] \cdot [\dot{e} - Ae - B\rho + KE_1 He(t - \tau_1(t)) + KE_2 He(t - \tau_2(t))] = 0 \quad (13)$$

$$\Delta_2 = 2\xi^T(t)N \cdot [e(t) - e(t - \tau_1(t)) - \int_{t-\tau_1(t)}^t \dot{e}(s)ds] = 0 \quad (14)$$

$$\Delta_3 = 2\xi^T(t)W \cdot [e(t) - e(t - \tau_2(t)) - \int_{t-\tau_2(t)}^t \dot{e}(s)ds] = 0 \quad (15)$$

onde $\xi^T = [e^T, \dot{e}^T, \widehat{\iota}_{dc}^T, \rho^T, e^T(t - \tau_1(t)), e^T(t - \tau_2(t))]$, $N = [N_1^T, \dots, N_6^T]^T$ e $W = [W_1^T, \dots, W_6^T]^T$ são matrizes livres. Nota-se que enquanto N_1 , N_2 , N_5 e N_6 são matrizes 3×3 , os termos N_3 e N_4 são vetores 1×3 . O mesmo é válido para os termos de W com o mesmo índice. Definindo $K_0 = RK$ para que o sistema seja linear novamente, derivando (11) em função do tempo e somando os termos de (12)-(15), obtém-se:

$$\begin{aligned} \dot{V}(t) \leq & \tilde{S} + \Delta_1 + \Delta_2 + \Delta_3 + 2e^T(t)P\dot{e}(t) + 2\widehat{\tau_{dc}}^T \lambda C^T \dot{e} \\ & + \sum_{j=1}^2 (\tau_j \dot{e}^T Z_j \dot{e} - \int_{t-\tau_j}^t \dot{e}^T(s) Z_j \dot{e}(s) ds) \end{aligned}$$

Como deseja-se obter uma LMI, é conveniente criar uma matriz simétrica. Assim, os argumentos da equação acima serão adaptados para uma matriz simétrica Φ . Para dar condições necessárias e suficientes para garantir a estabilidade, adiciona-se e subtrai-se o termo $\xi^T(\tau_1 N Z_1^{-1} N^T + \tau_2 W Z_2^{-1} W^T)\xi$ na equação anterior, com a parcela negativa do termo se encontrando dentro da integral.

$$\begin{aligned} \dot{V}(t) \leq & \xi^T (\Phi + \tau_1 N Z_1^{-1} N^T + \tau_2 W Z_2^{-1} W^T) \xi \\ & - \int_{t-\tau_1(t)}^t [\xi^T N + \dot{e}^T(s) Z_1] Z_1^{-1} [N^T \xi + Z_1 \dot{e}(s)] ds \\ & - \int_{t-\tau_2(t)}^t [\xi^T W + \dot{e}^T(s) Z_2] Z_2^{-1} [W^T \xi + Z_2 \dot{e}(s)] ds \end{aligned} \quad (16)$$

onde $\Phi = (\phi_{i,j}), i, j = 1, \dots, 6$ é uma matriz simétrica com:

$$\Phi = \begin{bmatrix} \phi_{11} & \phi_{12} & \phi_{13} & \phi_{14} & \phi_{15} & \phi_{16} \\ \phi_{12}^T & \phi_{22} & \phi_{23} & \phi_{24} & \phi_{25} & \phi_{26} \\ \phi_{13}^T & \phi_{23}^T & \phi_{33} & \phi_{34} & \phi_{35} & \phi_{36} \\ \phi_{14}^T & \phi_{24}^T & \phi_{34}^T & \phi_{44} & \phi_{45} & \phi_{46} \\ \phi_{15}^T & \phi_{25}^T & \phi_{35}^T & \phi_{45}^T & \phi_{55} & \phi_{56} \\ \phi_{16}^T & \phi_{26}^T & \phi_{36}^T & \phi_{46}^T & \phi_{56}^T & \phi_{66} \end{bmatrix}$$

$$\phi_{11} = -RA + N_1 + W_1 + (-RA + N_1 + W_1)^T$$

$$\phi_{12} = P + R - \delta A^T R^T + N_2^T + W_2^T$$

$$\phi_{13} = -\hat{g}_2 C S_1 + N_3^T + W_3^T$$

$$\phi_{14} = -\hat{g}_2 C S_2 - RB + N_4^T + W_4^T$$

$$\phi_{15} = K_0 E_1 H - N_1 + N_5^T + W_5^T$$

$$\phi_{16} = K_0 E_2 H - W_1 + N_6^T + W_6^T$$

$$\phi_{22} = \delta R + \tau_1 Z_1 + \tau_2 Z_2 + (\delta R + \tau_1 Z_1 + \tau_2 Z_2)^T$$

$$\begin{aligned}
\phi_{23} &= C\lambda \\
\phi_{24} &= -\delta RB \\
\phi_{25} &= \delta K_0 E_1 H - N_2 \\
\phi_{26} &= \delta K_0 E_2 H - W_2 \\
\phi_{jj} &= -2S_{j-2}, \quad j = 3, 4 \\
\phi_{34} &= 0 \\
\phi_{j5} &= -N_j, \quad j = 3, 4 \\
\phi_{j6} &= -W_j, \quad j = 3, 4 \\
\phi_{55} &= -N_5 - N_5^T \\
\phi_{56} &= -W_5 - N_6^T \\
\phi_{66} &= -W_6 - W_6^T
\end{aligned}$$

Utilizando o complemento de Schur pode ser demonstrado que $(\Phi + \tau_1 N^T Z_1^{-1} N + \tau_2 W^T Z_2^{-1} W) < 0$ é equivalente a:

$$\tilde{\Phi} = \begin{bmatrix} \Phi & \tau_1 N^T & \tau_2 W^T \\ \tau_1 N & -\tau_1 Z_1 & 0 \\ \tau_2 W & 0 & -\tau_2 Z_2 \end{bmatrix} < 0 \quad (17)$$

Ao resolver a LMI formada por (17) para um dado valor de δ , τ_1 e τ_2 obtêm-se valores de parâmetros para os quais se garante a estabilidade da origem do sistema (10). Entretanto, nem sempre haverá solução para (17). Caso o atraso seja muito grande é possível que não exista solução factível, portanto existe uma região de valores de atraso no qual a LMI admite solução.

Com ambos K_0 e R , pode-se obter a matriz do controlador K ao utilizar a seguinte equação:

$$K = R^{-1} * K_0 \quad (18)$$

Pode-se inverter R , pois se (17) é factível, isso significa que R não pode ser uma matriz singular e, portanto, ela é inversível e é possível obter K .

3.2. Controlador sem atraso

É interessante estudar o caso particular em que o sistema não possui atraso. Em muitos aspectos o procedimento é similar ao com atraso, mas simplificado. Primeiramente, o funcional que havia sido utilizado é modificado, passando a ser uma função e não necessitando o último termo de (11). Assim, a função escolhida:

$$V(t) = e^T(t)Pe(t) + 2\lambda \int_0^{C^T e(t)} \widehat{\tau}_{dc}(s) ds \quad (19)$$

Apesar das matrizes Z não estarem presentes em (12), elas ainda serão utilizadas para gerar a condição de negatividade de $\dot{V}(t)$ através de uma LMI. As próximas equações a serem modificadas são (13), (14) e (15).

$$\Delta_1 = 2[e^T R + \delta \dot{e}^T R] \cdot [\dot{e} - Ae - B\rho + KE_1 He(t) + KE_2 He(t)] = 0 \quad (20)$$

com $(E_1 + E_2) = I$.

$$\Delta_2 = 2\xi^T(t)N \cdot [e(t) - \int_0^t \dot{e}(s) ds] = 0 \quad (21)$$

$$\Delta_3 = 2\xi^T(t)W \cdot [e(t) - \int_0^t \dot{e}(s) ds] = 0 \quad (22)$$

Haverá também uma pequena mudança nos vetores auxiliares ξ^T , N e W , pois não existe mais os sinais de erro com atraso, portanto eles possuem dimensão 4 em vez de 6. Ao realizar a derivada de (18):

$$\dot{V}(t) \leq \tilde{S} + \Delta_1 + \Delta_2 + \Delta_3 + 2e^T(t)P\dot{e}(t) + 2\widehat{\tau}_{dc}^T \lambda C^T \dot{e}$$

Com procedimento análogo ao usado anteriormente, adiciona-se e subtrai-se $\xi^T(\tau_1 NZ_1^{-1} N^T + \tau_2 WZ_2^{-1} W^T)\xi$:

$$\begin{aligned} \dot{V}(t) \leq & \xi^T (\Phi + NZ_1^{-1}N^T + WZ_2^{-1}W^T)\xi - \int_{t-\tau_1(t)}^t \xi^T NZ_1^{-1}[N^T\xi + Z_1\dot{e}(s)]ds \\ & - \int_{t-\tau_2(t)}^t \xi^T WZ_2^{-1}[W^T\xi + Z_2\dot{e}(s)]ds \end{aligned}$$

Assim, se garantimos que se $(\Phi + N^T Z_1^{-1} N + W^T Z_2^{-1} W) < 0$, isto é, é uma matriz negativa definida, asseguramos que $\dot{V}(t) \leq 0$ e, logo, a estabilidade assintótica do sistema é garantida globalmente, isto é, $\forall x(0)$, tem-se que $\lim_{t \rightarrow \infty} (e(t) = 0)$ e a sincronização é atingida, com as mesmas condições de desigualdade de variáveis do sistema com atraso.

Com $\Phi = (\phi_{i,j})$, $i, j = 1, \dots, 4$ sendo uma matriz simétrica com:

$$\Phi = \begin{bmatrix} \phi_{11} & \phi_{12} & \phi_{13} & \phi_{14} \\ \phi_{12}^T & \phi_{22} & \phi_{23} & \phi_{24} \\ \phi_{13}^T & \phi_{23}^T & \phi_{33} & \phi_{34} \\ \phi_{14}^T & \phi_{24}^T & \phi_{34}^T & \phi_{44} \end{bmatrix}$$

$$\phi_{11} = -RA + N_1 + W_1 + (-RA + N_1 + W_1)^T + K_0 H$$

$$\phi_{12} = P + R - \delta A^T R^T + N_2^T + W_2^T + \delta K_0 H$$

$$\phi_{13} = -\hat{g}_2 C S_1 + N_3^T + W_3^T$$

$$\phi_{14} = -\hat{g}_2 C S_2 - RB + N_4^T + W_4^T$$

$$\phi_{22} = \delta R + \delta R^T$$

$$\phi_{23} = C\lambda$$

$$\phi_{24} = -\delta RB$$

$$\phi_{jj} = -2S_{j-2}, \quad j = 3, 4$$

$$\phi_{34} = 0$$

Por fim, a LMI resultante fica da forma:

$$\tilde{\Phi} = \begin{bmatrix} \Phi & N^T & W^T \\ N & -Z_1 & 0 \\ W & 0 & -Z_2 \end{bmatrix} < 0 \quad (23)$$

Novamente, a equação utilizada para se obter K é a mesma utilizada anteriormente no sistema com atraso, uma vez que para que (23) seja factível, R ainda precisa ser não singular.

Como pode-se notar, há duas principais diferenças entre a LMI com atraso e a sem atraso. A primeira é que, no sistema, sem atraso não há os termos τ , mas não é equivalente a zerar os termos em (17). A segunda diferença se encontra na matriz Φ . Com a remoção de τ_1 e τ_2 , os termos com referência ao atraso, isto é, os termos com índice 5 e 6 se mesclam com os de índice 1, pois agora são iguais. Portanto os termos $K_0 E_1 H$ e similares migram para a primeira coluna, saindo da posição ϕ_{15} , que não existe mais, para a posição ϕ_{11} . Isso faz com que (23) seja uma LMI menor do que (17) e, portanto, deve necessitar de menos tempo para ser resolvida.

3.3 Controlador com Saturação

Uma situação possível de acontecer ao implementar o controlador fisicamente é das limitações físicas dos componentes. No caso a ser estudado, esta limitação se dá pela saída máxima que os atuadores conseguem produzir, ou seja, pela saturação dos mesmos. Quando o atuador não consegue produzir o sinal no valor desejado, isto, no mínimo, prejudica o regime transitório do sistema, mas em casos extremos é possível que o sistema não seja capaz de seguir a referência devido à saturação.

A característica principal de um sistema com saturação é a não linearidade causada pela mesma. Suponha-se que a saturação ocorre de maneira simétrica, ou seja, o valor mínimo que o atuador consegue na sua saída é o negativo do seu valor máximo e que é a mesma para todos os atuadores. Esta não linearidade será definida como sendo:

$$sat(u(t)) = \begin{cases} u_{m\acute{a}x}, & se\ u > u_{m\acute{a}x} \\ u, & se\ -u_{m\acute{a}x} < u < u_{m\acute{a}x} \\ -u_{m\acute{a}x}, & se\ u < -u_{m\acute{a}x} \end{cases} \quad (24)$$

com $u_{m\acute{a}x}$ sendo um escalar e $sat(u(t))$ um vetor 3×1 .

Como não há garantias de que o sistema linear resultante será assintoticamente estável, deve-se realizar a seguinte mudança:

$$\psi(u(t)) = \text{sat}(u(t)) - u(t)$$

Ou seja, $\psi(u(t))$ é uma zona morta centralizada.

Para esta saturação é conveniente utilizar uma condição de setor generalizada (Tarbouriech, 2011), pois esta deve gerar condições menos conservativas do que comparado a condições de setor utilizadas previamente. Primeiramente define-se um conjunto matemático:

$$S(u - \omega, -u_{\text{máx}}, u_{\text{máx}}) = \{u \in \mathcal{R}^3; \omega \in \mathcal{R}^3; -u_{\text{máx}} \leq u - \omega \leq u_{\text{máx}}\}$$

sendo ω uma constante positiva.

Assim, a não linearidade $\psi(u)$ deve seguir a seguinte forma:

$$\psi^T(u)S_3(\psi(u) + \omega) \leq 0$$

sendo S_3 uma matriz diagonal positiva definida.

Que pode ser reescrita como:

$$\psi(u)^T S_3 \psi(u) + \psi(u)^T S_3 G H e \leq 0 \quad (25)$$

com G uma matriz de mesmas dimensões de K .

Assim como nos casos anteriores será aplicado o S procedure em (25), cujo resultado será somado à (12), substituindo u :

$$0 \geq \tilde{S} = 2(-e^T |\hat{g}_2| C S_1 \hat{\tau}_{dc} - \hat{\tau}_{dc}^T S_1 \hat{\tau}_{dc} - e^T |\hat{g}_2| C S_2 \rho - \rho^T S_2 \rho - \psi^T S_3 \psi - \psi(u)^T S_3 G H e) \quad (26)$$

Como ambas G e S_3 seriam incógnitas, para a LMI ser linear novamente, deveria haver a seguinte troca de variável $G_0 = S_3 G$. Entretanto, por causa da saturação, uma relação de inclusão que depende de G deve ser considerada,

portanto não sendo possível realizar a troca de variável e resolver a relação de inclusão simultaneamente. Portanto, S_3 será determinado, de forma que não seja mais uma variável, como sendo:

$$S_3 = \varepsilon I$$

Como houve mudanças em $\text{sat}(u)$, a derivada do erro também se modifica, tornando-se:

$$\dot{e}(t) = Ae(t) + B\rho(C^T e(t), \hat{x}(t)) - \psi(t) - u(t)$$

O formato da função de Lyapunov não se altera em comparação ao caso sem atraso, portanto ainda é utilizada (19). Entretanto as equações auxiliares (20), (21) e (22) sofrem pequenas modificações.

$$\Delta_1 = 2[e^T R + \delta \dot{e}^T R] \cdot [\dot{e} - Ae - B\rho + \psi + KHe] = 0 \quad (27)$$

Novamente, haveria a multiplicação das variáveis R e K, tornando o sistema não linear. Nos casos anteriores foi possível criar uma variável para linearizar o sistema, mas, assim como no caso de G, K deverá aparecer na relação de inclusão, então não será possível resolver a LMI e a relação de inclusão simultaneamente. Portanto escolheu-se utilizar um K previamente conhecido e será verificado se esse controlador é adequado.

As modificações em (21) e (22) se devem por uma mudança necessária em ξ , que em comparação com a situação anterior possui um quinto termo $\psi(t)$. Analogamente N e W possuem um quinto termo de dimensão 3x3.

Assim, ao derivar a função de Liapunov, obtém-se:

$$\dot{V}(t) \leq \tilde{S} + \Delta_1 + \Delta_2 + \Delta_3 + 2e^T(t)P\dot{e}(t) + 2\widehat{\tau}_{dc}^T \lambda C^T \dot{e}$$

Adicionando e subtraindo $\xi^T(NZ_1^{-1}N^T + WZ_2^{-1}W^T)\xi$:

$$\begin{aligned} \dot{V}(t) \leq & \xi^T (\Phi + NZ_1^{-1}N^T + WZ_2^{-1}W^T) \xi - \int_{t-\tau_1(t)}^t \xi^T NZ_1^{-1} [N^T \xi + Z_1 \dot{e}(s)] ds \\ & - \int_{t-\tau_2(t)}^t \xi^T WZ_2^{-1} [W^T \xi + Z_2 \dot{e}(s)] ds \end{aligned}$$

Novamente, se garantimos que $(\Phi + NZ_1^{-1}N^T + WZ_2^{-1}W^T) < 0$, é assegurado que $\dot{V}(t) \leq 0$ e, logo, a estabilidade assintótica do sistema é garantida localmente, isto é, a sincronização é atingida dependendo dos estados iniciais, como será visto na sequência.

Desta vez $\Phi = (\phi_{ij})$, $i, j = 1, \dots, 5$ é uma matriz simétrica com:

$$\Phi = \begin{bmatrix} \phi_{11} & \phi_{12} & \phi_{13} & \phi_{14} & \phi_{15} \\ \phi_{12}^T & \phi_{22} & \phi_{23} & \phi_{24} & \phi_{25} \\ \phi_{13}^T & \phi_{23}^T & \phi_{33} & \phi_{34} & \phi_{35} \\ \phi_{14}^T & \phi_{24}^T & \phi_{34}^T & \phi_{44} & \phi_{45} \\ \phi_{15}^T & \phi_{25}^T & \phi_{35}^T & \phi_{45}^T & \phi_{55} \end{bmatrix}$$

$$\phi_{11} = -RA + N_1 + W_1 + RKH(-RA + N_1 + W_1 + RKH)^T$$

$$\phi_{12} = P + R + \delta H^T K^T R^T - \delta A^T R^T + N_2^T + W_2^T$$

$$\phi_{13} = -\hat{g}_2 CS_1 + N_3^T + W_3^T$$

$$\phi_{14} = -\hat{g}_2 CS_2 - RB + N_4^T + W_4^T$$

$$\phi_{15} = R - S_3 GH + N_5^T + W_5^T$$

$$\phi_{22} = \delta R + \delta R^T$$

$$\phi_{23} = C\lambda$$

$$\phi_{24} = -\delta RB$$

$$\phi_{25} = \delta R$$

$$\phi_{jj} = -2S_{j-2}, \quad j = 3, 4, 5$$

$$\phi_{34} = \phi_{35} = \phi_{45} = 0$$

Finalmente, a LMI fica na seguinte forma:

$$\tilde{\Phi} = \begin{bmatrix} \Phi & N & W \\ N^T & -Z_1 & 0 \\ W^T & 0 & -Z_2 \end{bmatrix} < 0 \quad (28)$$

Uma característica única deste caso é que dependendo das condições iniciais, pode ser que o erro não convirja a zero, ou seja, apenas dentro de uma região em que a sincronização ocorre. Para determinar a região, determina-se as seguintes LMIs:

$$\begin{bmatrix} P & (K_{(i)} - G_{(i)})^T \\ K_{(i)} - G_{(i)} & u_{m\acute{a}x(i)}^2 \end{bmatrix} > 0, i = 1, 2, 3$$

com $u_{m\acute{a}x(i)}$ o valor da saturação do atuador i e $K_{(i)}$ e $G_{(i)}$ a linha i de suas respectivas matrizes.

Ao aplicar o complemento de Schur:

$$\begin{aligned} e^T \left(P - (K_{(i)} - G_{(i)})^T u_{m\acute{a}x(i)}^2{}^{-1} (K_{(i)} - G_{(i)}) \right) e &> 0 \\ e^T (K_{(i)} - G_{(i)})^T u_{m\acute{a}x(i)}^2{}^{-1} (K_{(i)} - G_{(i)}) e &< e^T P e \end{aligned}$$

Se, no instante inicial, aonde o erro é a diferença das condições iniciais

$$e_0^T P e_0 \leq 1 \quad (30)$$

Então:

$$e^T (K_{(i)} - G_{(i)})^T (K_{(i)} - G_{(i)}) e = |u_i|^2 \leq \overline{u_{m\acute{a}x}^2}$$

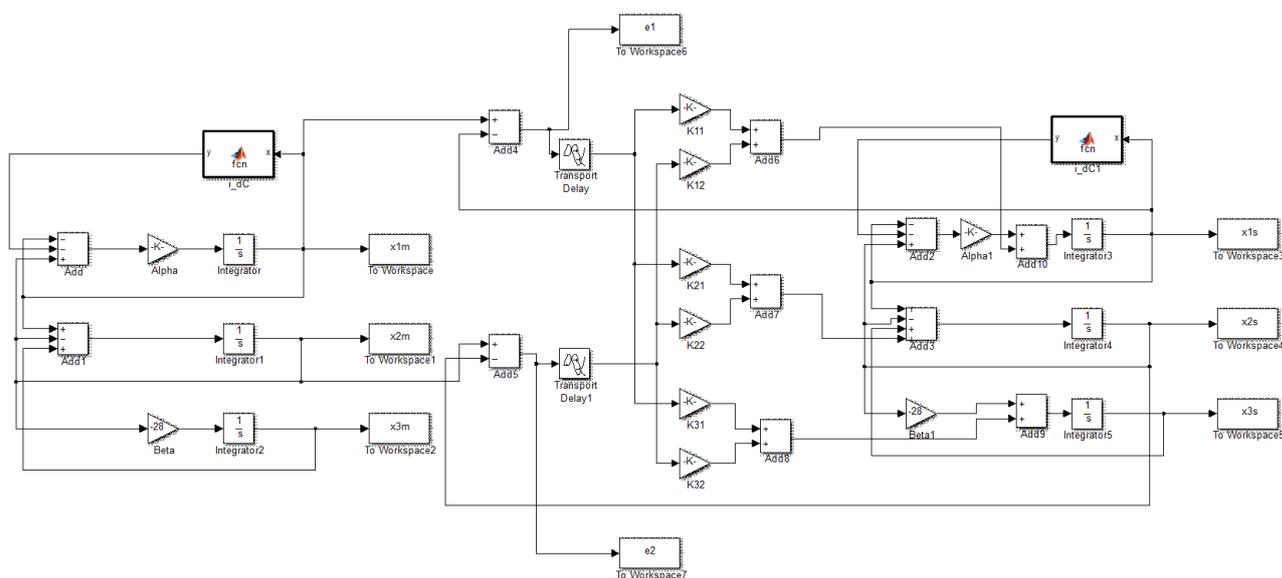
E, portanto, no instante inicial a saturação não irá instabilizar sistema e, como não há atraso, o erro nos instantes seguintes deverá ser inferior ao erro inicial, ou seja, o erro inicial é o seu valor máximo. Assim, pode-se garantir que para os valores iniciais nos quais a desigualdade (30) é verdadeira a saturação não tornará o sistema instável e haverá sincronização. Caso haja atraso, de acordo com (Kellert, 1993), deve ser utilizado o expoente de Liapunov λ próprio para o circuito de Chua para calcular o valor previsto do erro em um tempo t de acordo com:

$$e_{previsto}(t) = e_{previsto}(0)e^{\lambda t}$$

4. Resultados Simulados

Com as equações LMI, agora pode-se obter o controlador. Inicialmente adaptou-se o sistema da Figura 9 para o sistema em malha fechada descrito na Figura 13, obtendo o sistema da Figura 15. Os ganhos K descritos correspondem ao valor correspondente na matriz K do controlador.

Figura 15 – Sistema em Malha Fechada



Os parâmetros de circuito Chua utilizados foram os encontrados na Tabela 1, com o setor considerado $[-1.2, 0]$, $\delta = 0.5$, $\tau_1 = 1$, $\tau_2 = 0.1$. Os estados iniciais se encontram na Tabela 2.

Tabela 2 – Estados iniciais para as simulações

Estado	Mestre	Escravo
x_1	-0.2	0.5
x_2	-0.33	-0.1
x_3	0.2	0.66

Em todas as simulações foram analisados um período de tempo de 100 segundos, de forma a garantir a estabilidade.

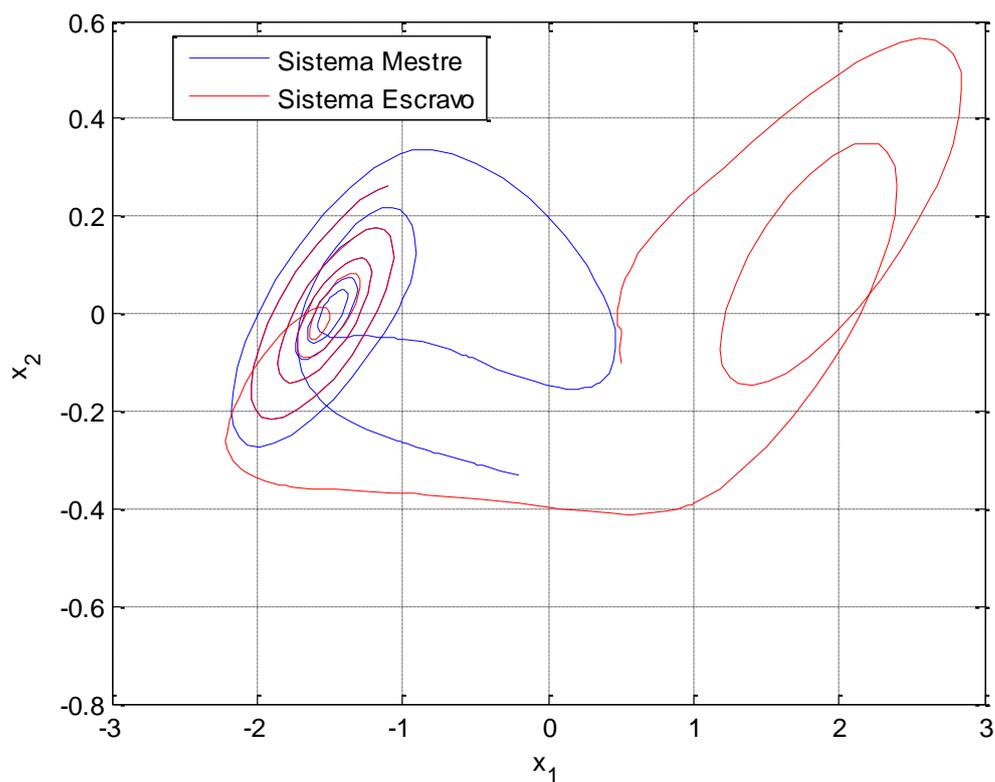
4.1 Resultados Simulados do Sistema com Atraso

Para a simulação do atraso, foi considerado o caso nos quais o atraso foi o máximo possível, para assim assegurar que para o valor projetado a sincronização é atingida. Ao resolver (17) com o comando `feaslp`, que será utilizado para todas as simulações, e (18), obtém-se os parâmetros do controlador:

$$K = \begin{bmatrix} 0.0007 & 3.7258 \\ 0.0002 & 5.0348 \\ -0.0012 & -20.2720 \end{bmatrix}$$

Substituindo K no sistema descrito na Figura 12, pode-se observar o comportamento no tempo de cada uma das variáveis. Primeiramente será conferido se o sistema escravo segue o mestre. O comportamento resultante está localizado no gráfico da Figura 16.

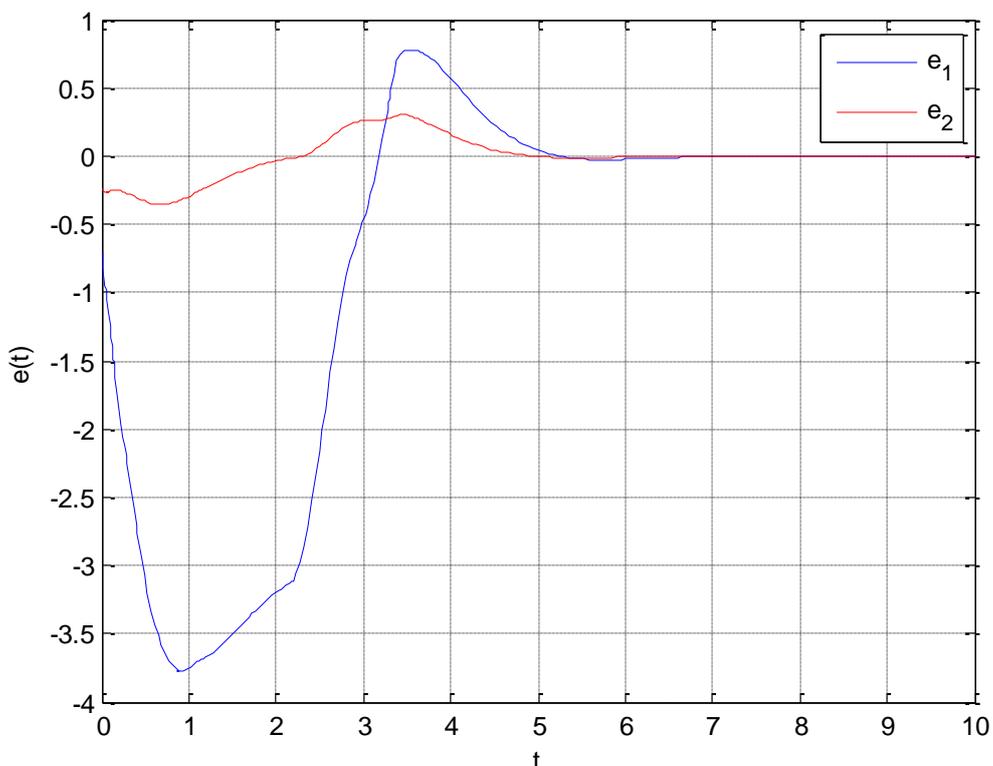
Figura 16 - Comportamento no plano x_1 - x_2 dos sistemas mestre e escravo com atraso.



Percebe-se que após um período inicial em que o erro aumenta, o sistema escravo se aproxima do mestre e chegam num ponto em que as trajetórias dos dois sistemas se sobrepõem, atingindo a sincronização.

O próximo passo será analisar os sinais de erro do sistema, e_1 e e_2 no tempo. O gráfico gerado se encontra na Figura 17.

Figura 17 - Comportamento temporal dos sinais de erro com atraso.



Observa-se que enquanto o sinal e_2 se mantém relativamente constante no início, o sinal e_1 inicialmente aumenta rapidamente em módulo. Este efeito ocorre por causa do atraso. No primeiro segundo, devido ao atraso, para e_1 é como se o sistema não tivesse controlador e, portanto, o efeito borboleta prevalece, fazendo com que os dois sistemas diverjam. Este efeito do atraso não é perceptível para e_2 porque o atraso τ_2 é pequeno. Caso este fosse da mesma ordem de grandeza de τ_1 , também seria perceptível.

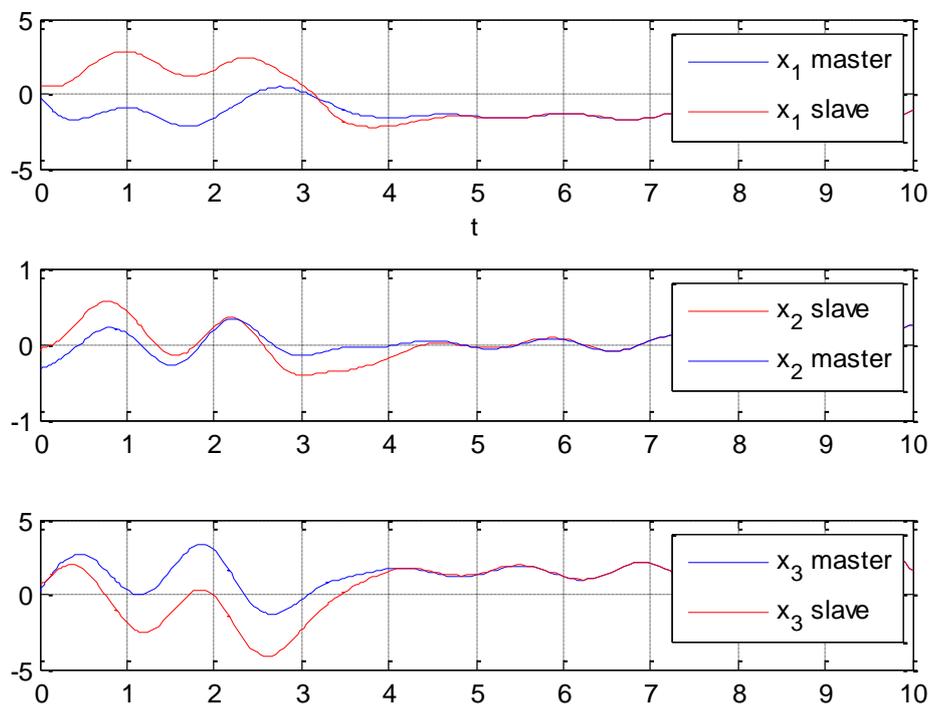
Quando o controlador começa a fazer efeito próximo a 1 s, os sistemas começam a entrar em sintonia e o erro diminui, chegando a ser nulo em regime

permanente. Entretanto, há um sobrepasso de sinal, provavelmente também ligado ao atraso.

Por causa deste efeito que para atrasos de elevado valor os sistemas não entrarão em sincronia para nenhum K . Pois ao controlador começar a funcionar com os valores atrasados, os dois sistemas estarão separados de tal maneira que os valores iniciais de erro não farão com que o erro atual diminua em módulo, portanto o erro sempre continuará a aumentar.

Por último, serão analisados os comportamentos de cada estado dos dois sistemas, para assegurar que os do sistema escravo seguem os seus respectivos estados do sistema mestre, como encontrado na Figura 18.

Figura 18 – Comportamento temporal dos estados internos com atraso.



4.2 Resultados Simulados para o Sistema sem Atraso

Para o caso particular sem atraso, foram realizadas duas simulações. Uma utilizando (17), com $\tau_1 = \tau_2 = 0$ e uma utilizando (23) diretamente. Em

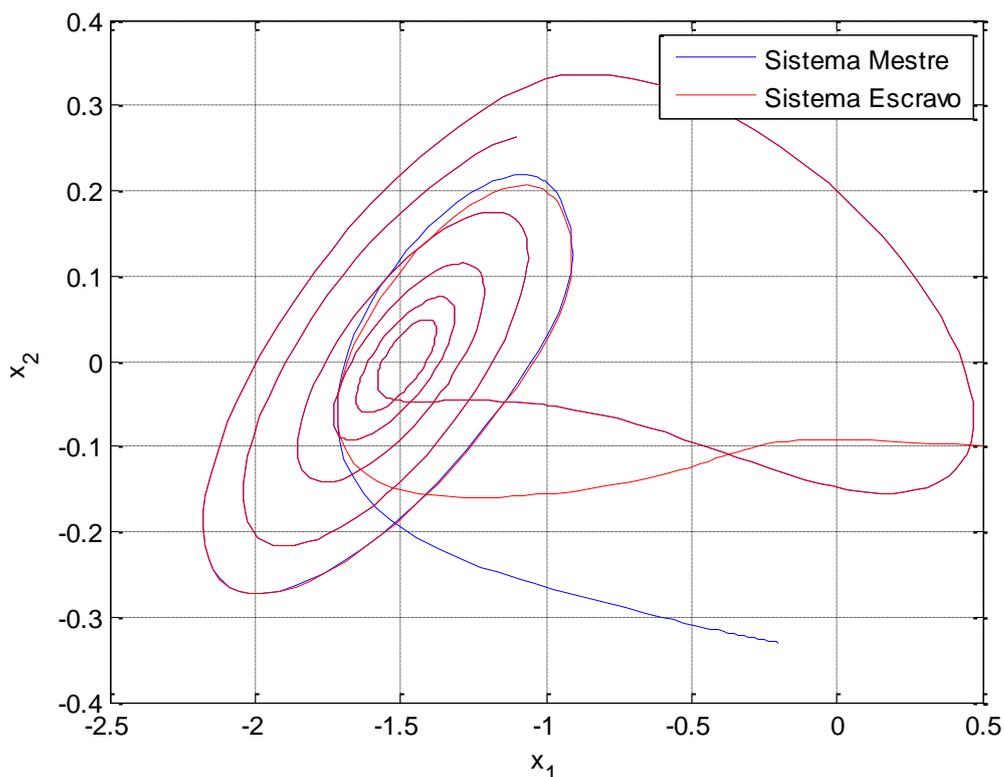
ambos os casos os parâmetros que serão utilizados são os mesmos do sistemas com atraso.

Ao utilizar (17), obtém-se o K:

$$K = \begin{bmatrix} 146.4927 & 1.2836 \\ -3.8705 & 7.0990 \\ -8.0941 & -7.6586 \end{bmatrix}$$

Novamente será analisado o comportamento caótico de ambos os sistemas. Na Figura 19 se encontram ambos o sistema mestre quanto o escravo.

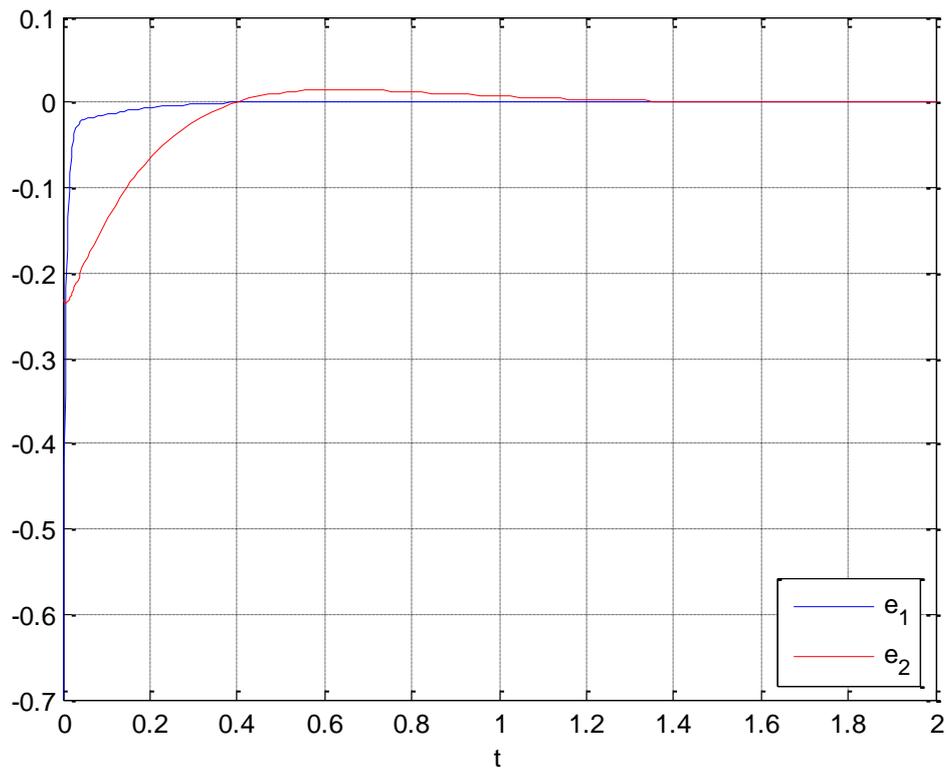
Figura 19 – Comportamento no plano x_1 - x_2 dos sistemas mestre e escravo sem atraso, utilizando (17).



Sem o atraso o sistema escravo rapidamente entra em sincronia com o sistema mestre. Ao comparar ao caso com atraso já pode ser notado que o efeito borboleta não demonstra nenhuma influência, a trajetória do sistema escravo diretamente indo ao encontro da trajetória do sistema mestre.

Na Figura 20 são mostrados os sinais de erro do sistema sem atraso, mas ao utilizar (17).

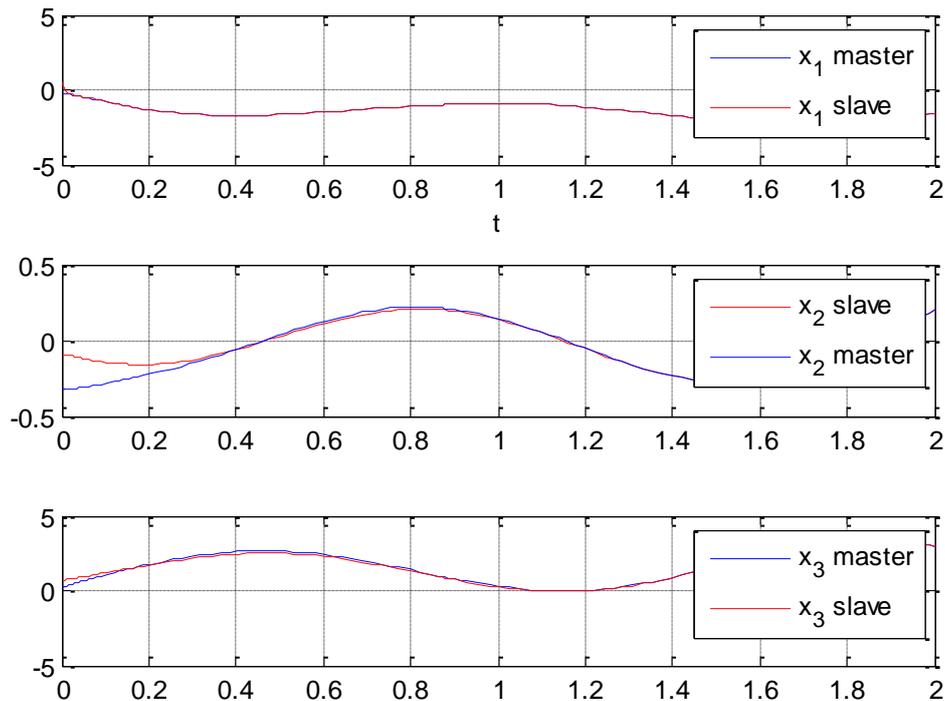
Figura 20 - Comportamento temporal dos sinais de erro sem atraso, utilizando (17).



Ambos os sinais de erro tendem a zero em regime permanente, mas em comparação ao sistema com atraso, ambos os erros rapidamente diminuem. Este sistema possui um comportamento transitório muito mais rápido e sem haver o mesmo sobrepasso encontrado anteriormente, mesmo com ganhos em K comparativamente maiores do que para o sistema com atraso.

Na Figura 21, se encontram os estados internos de cada sistema pareados.

Figura 21 - Comportamento temporal dos estados internos sem atraso ao utilizar (17).



Novamente pode-se notar o quão rápido os estados dos dois sistemas entram em sincronia rapidamente, mas não apresenta nenhum sobrepasso. Mais uma vez os três estados entram em sincronia, mesmo utilizando-se apenas dois para os sinais de erro.

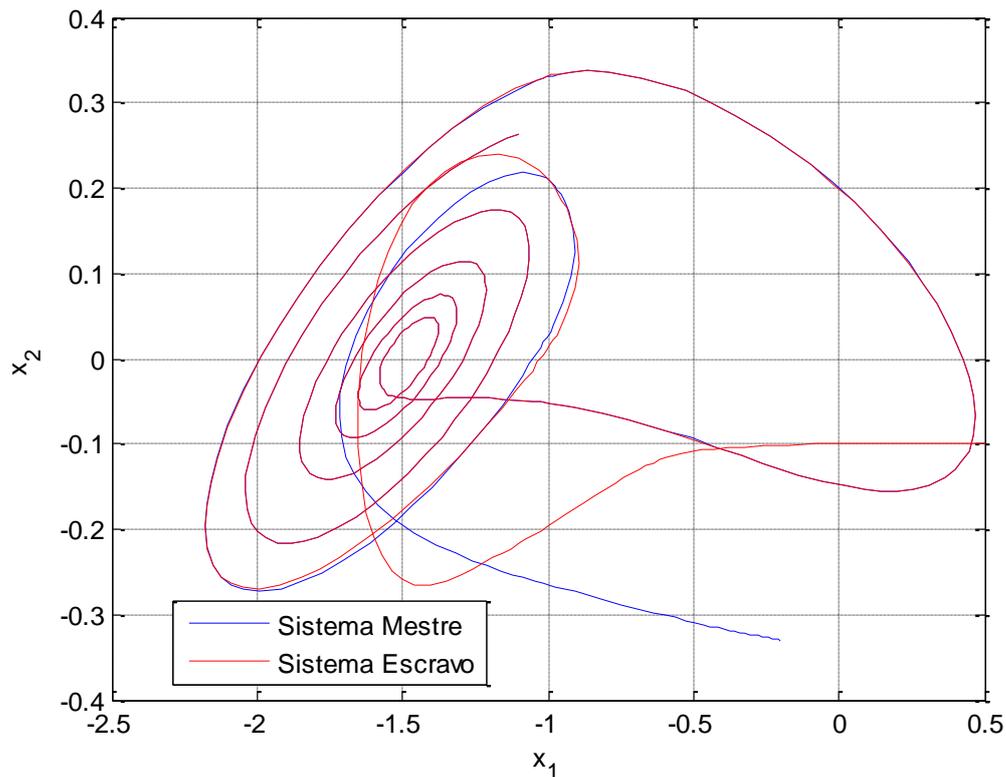
Ao utilizar-se (23) para determinar o K , encontra-se:

$$K = \begin{bmatrix} 114.6646 & 130.9447 \\ 0.3255 & 3.7626 \\ -7.5567 & 48.5296 \end{bmatrix}$$

Uma diferença que se pode perceber é no parâmetro K_{12} , o qual aumentou consideravelmente, estando agora na mesma ordem de grandeza de K_{11} .

Analisando na Figura 22 o comportamento 2D dos sistemas.

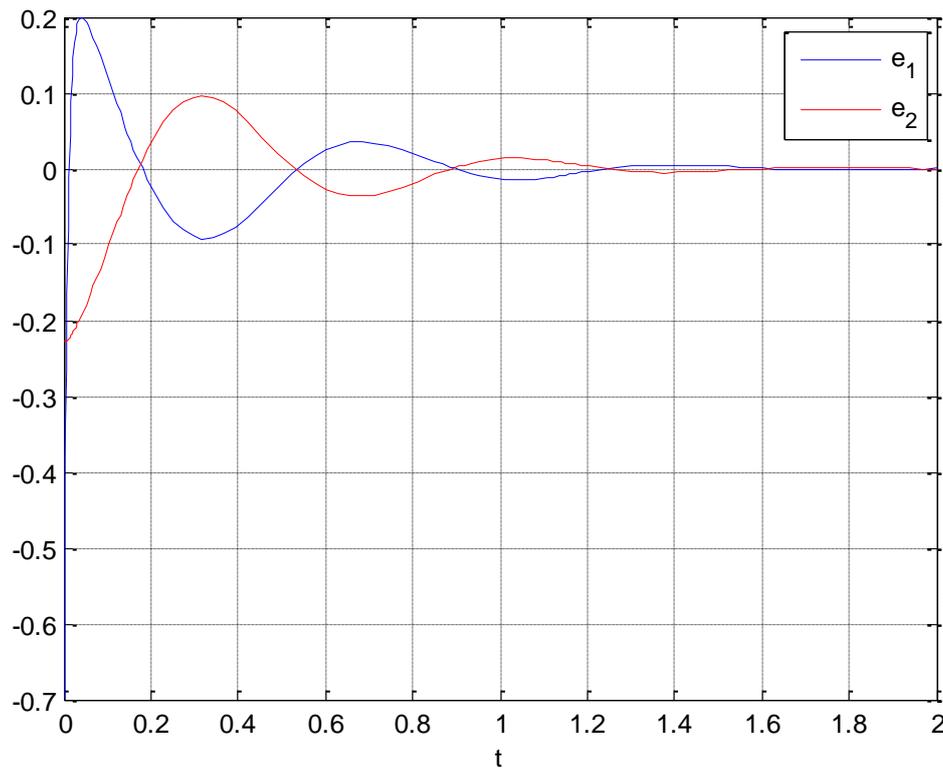
Figura 22 – Comportamento no plano x_1 - x_2 dos sistemas mestre e escravo sem atraso, utilizando (23).



Mais uma vez os dois sistemas rapidamente entram em sincronia, sem ser possível perceber muitas diferenças entre o comportamento da Figura 22 com o da Figura 19.

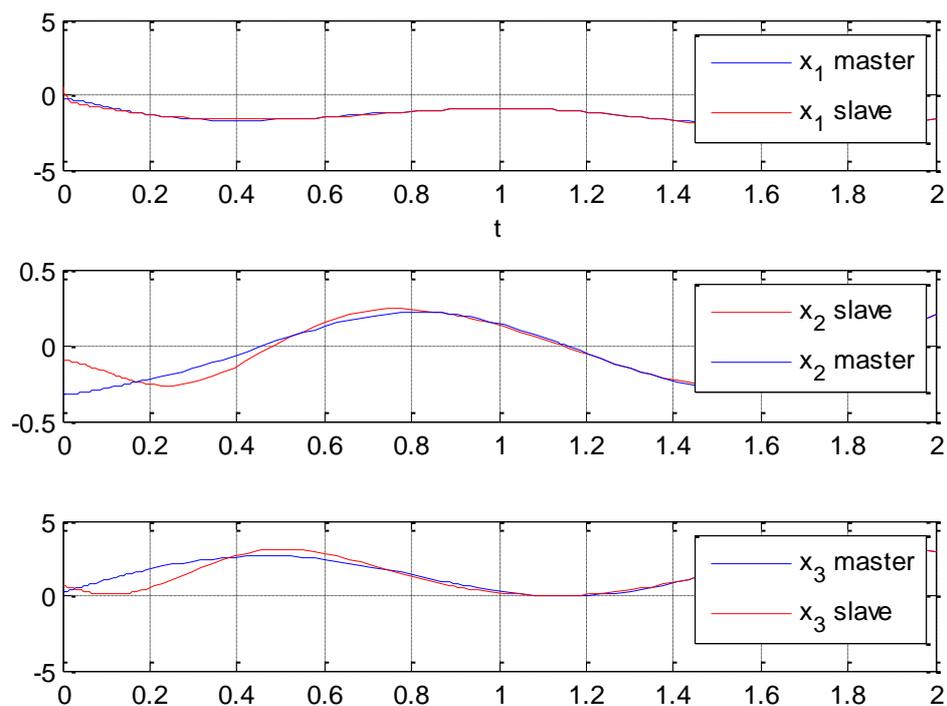
Ao analisar os sinais de erro descritos na Figura 23, nota-se que existe um comportamento oscilatório mais pronunciado do que quando foi projetado utilizando-se (17). Este efeito pode ter sido causado pelos maiores ganhos encontrados no controlador realizado por (23) em comparação ao obtido através de (17). Além disso, o erro entra em regime permanente muito mais lentamente e possui sobrepasso.

Figura 23 - Comportamento temporal dos sinais de erro sem atraso, utilizando (23).



Por último, será analisado o comportamento dos estados pareados, representados na Figura 24. Novamente percebe-se que o sistema está mais lento do que quando (17) foi utilizada para obter o controlador, mas ainda assim entra em regime permanente relativamente rápido, precisando de menos de 2 segundos. Nota-se os parâmetros x_3 , nos quais o sobrepasso é consideravelmente maior do que nos outros dois parâmetros.

Figura 24 - Comportamento temporal dos estados internos sem atraso ao utilizar (23).



Ao analisar as três situações estudadas, percebe-se que as LMIs obtidas ao serem resolvidas com sucesso conseguem garantir que o sistema escravo siga o mestre fielmente. Entretanto, para o caso de haver atraso, deve-se confirmar se para os atrasos máximos esperados há uma solução para o sistema. Além disso, quando maior o atraso mais tempo irá levar para eles entrarem em sincronia, o que, dependendo da aplicação, pode não ser desejado, ou até, um empecilho.

Há, também, o problema de que com o comando de MATLAB utilizado, o `feaslp`, não podem ser feitas restrições ou considerações quanto ao regime transitório. Por causa disso nas duas simulações sem atraso, apesar de serem similares em contexto, apresentam regimes transitórios muito variados. Portanto, caso seja de interesse controlar o sobrepasso ou fazer com que o sistema chegue em regime permanente antes de um determinado período, este comando não pode ser utilizado. Entretanto os outros são relativamente mais complexos de serem utilizados e consomem mais processamento. Apesar das limitações acima, os estados iniciais não devem, em teoria, limitar o sistema, apesar de poderem influenciar no regime permanente.

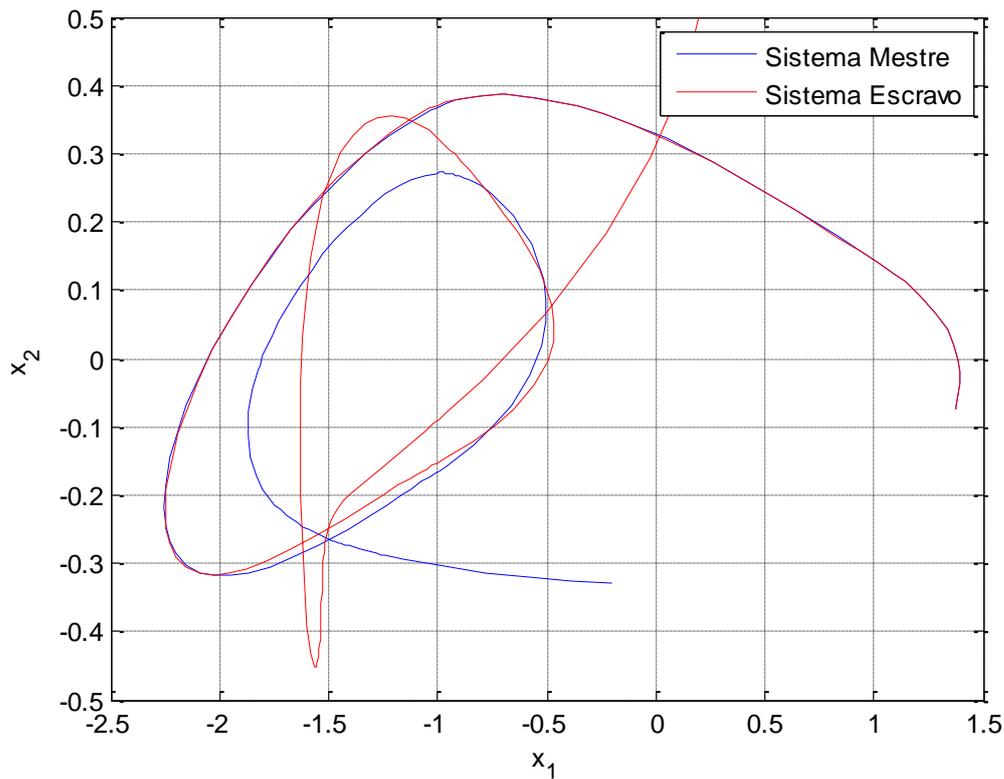
Ainda deve-se levar em conta as incertezas dos parâmetros dos sistemas, principalmente próximo da região de atraso na qual o sistema não possui resposta. Um cuidado especial sobre as incertezas recai na condição de setor. Caso os valores reais do circuito façam com que o comportamento da equação não-linear do diodo de Chua se encontre fora do setor previsto, resolver as LMIs não garante que os sistemas entrem em sincronia, enquanto que, caso ele esteja diferente do esperado mas ainda assim dentro do setor, há a garantia de sincronização. Portanto, é aconselhável considerar um setor maior do que o esperado.

4.3 Resultados Simulados do Sistema com Saturação

Para o caso de sistemas com saturação foi realizada uma nova simulação, utilizando (28), (29) e (30). Como agora será feita uma análise de K , o ganho deve ser previamente definido. Escolheu-se o último ganho calculado por conveniência. S_3 será arbitrado. Como não necessariamente para qualquer valor de S_3 o sistema será realizável, o seu valor será variável, passando por múltiplas iterações. Por último, blocos de saturação serão conectados após cada ganho, com valor de saturação $u_{m\acute{a}x} = 10$. Foram utilizados os mesmos valores iniciais encontrados na Tabela 2.

Obtendo a confirmação que a LMI é factível, pode-se iniciar as análises. Inicialmente é analisado o resultado no eixo $x_1 - x_2$, representado na Figura 25, onde pode ser visto o comportamento caótico do sistema.

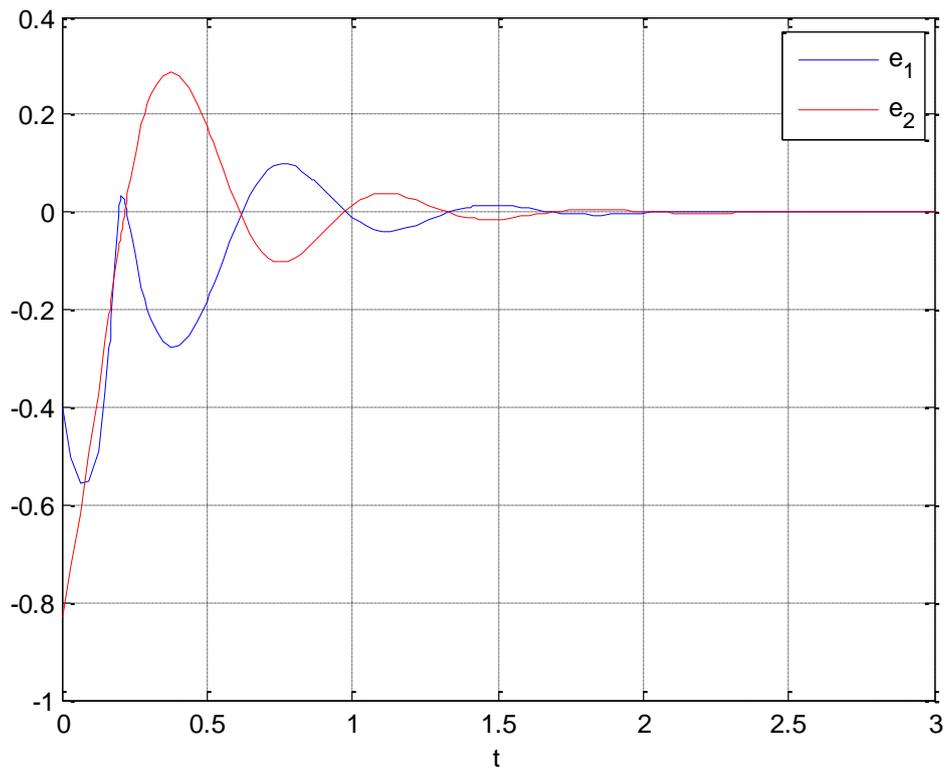
Figura 25 - Comportamento no plano x_1 - x_2 dos sistemas mestre e escravo com saturação.



Percebe-se que após um período de tempo os dois sistemas se superpõem, entrando em sincronia. Nesta imagem não se pode supor se demorou mais tempo do que originalmente no caso sem saturação.

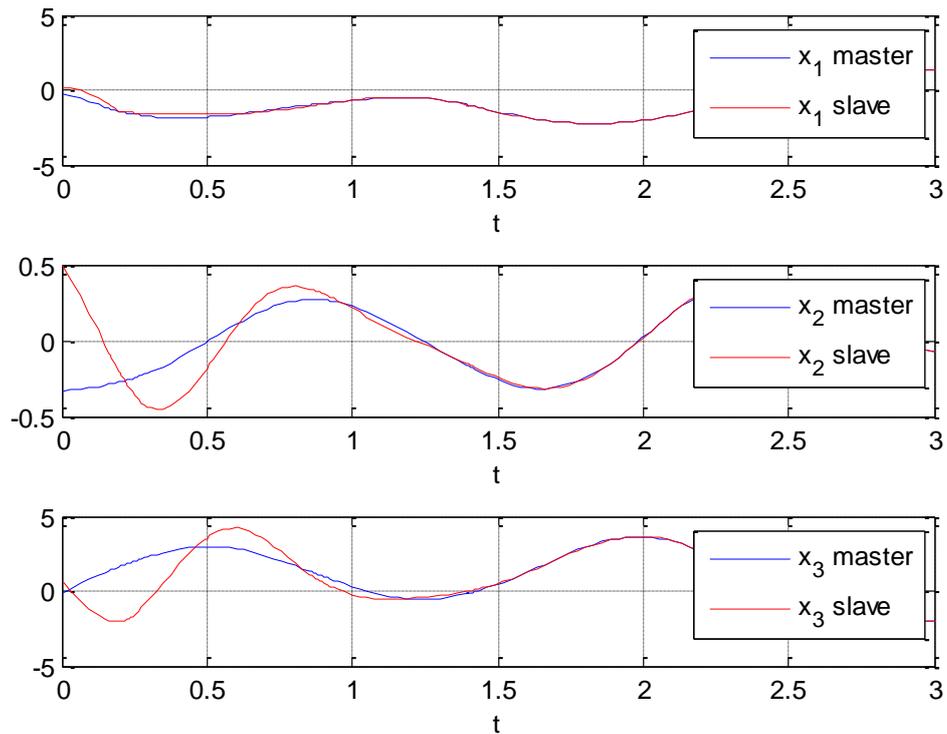
Em seguida analisa-se os dois sinais de erro monitorados, encontrados na Figura 26. Percebe-se novamente um comportamento oscilatório nos sinais de erro. Aparentemente o sistema se tornou ligeiramente mais lento, fato condizente caso algum dos atuadores tenha saturado em algum momento. Neste caso o período transitório aumentou um valor proporcionalmente pequeno, mas dependendo da saturação tal valor poderia ser maior.

Figura 26 - Comportamento temporal dos sinais de erro com saturação.



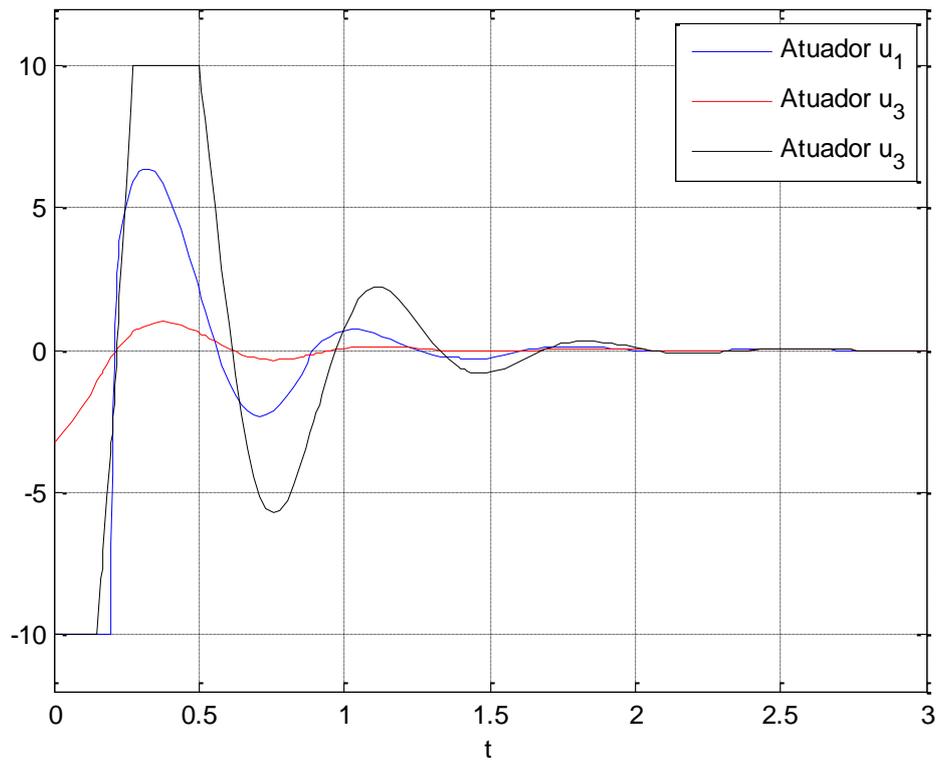
Na Figura 27 os estados internos do sistema são pareados dois a dois. Percebe-se agora que para o estado x_1 o erro é proporcionalmente insignificante, não podendo ser diferenciado o estado do sistema mestre do estado do sistema escravo. Entretanto, para os outros estados o erro é mais significativo.

Figura 27 - Comportamento temporal dos estados internos com saturação.



Na Figura 28 estão representadas as saídas dos atuadores, de forma a verificar se houve ou não a saturação. Ao analisar o resultado verifica-se que houve saturação nos atuadores responsáveis pelos estados x_1 e x_2 , fato condizente com os ganhos destes estados serem muito maiores do que os outros. Tal saturação pode ser a causa deste sistema ser ligeiramente mais lento do que no caso sem saturação.

Figura 28 - Saídas dos atuadores



5. Conclusões Finais

Este trabalho apresentou a sincronização simulada de circuitos caóticos, sendo o objeto de estudo o circuito de Chua. Para gerar o comportamento caótico o circuito possui componentes não lineares, fato que dificulta o seu estudo e implementação. Entretanto o comportamento caótico é de especial interesse de estudo e pode ser aplicado na modulação de comunicações seguras, o que leva à necessidade de sincronizar dois circuitos.

No capítulo 2 foi realizado o estudo do circuito em si, analisando as equações que o regem e seu comportamento. Foram realizadas simulações do circuito e das equações que o regem, notando discrepâncias entre elas. Tais discrepâncias são atribuídas a não idealidades que os componentes do circuito possuem, o que não é levado em conta nas equações utilizadas, mas que interagem significativamente com o comportamento do circuito.

No capítulo 3 foi feito o equacionamento para o projeto do controlador, utilizando principalmente matrizes e montando as LMIs relevantes. Foram considerados diversos casos, incluindo situações em que há atraso no sistema ou saturação nos atuadores. Analisou-se as não linearidades intrínsecas do sistema e a causada pela saturação, bem como o efeito gerado pelos atrasos desiguais entre os estados internos. No caso com saturação o sistema obtido foi limitado a análise de factibilidade de um ganho conhecido.

No capítulo 4 foi aplicado o controlador calculado através das LMIs obtidas no capítulo 3. Verificou-se que a diferença entre os dois sistemas chegou a zero e que a sincronia foi obtida e, com exceção do caso com saturação, a sincronia foi obtida rapidamente.

As conclusões obtidas por este trabalho abrem caminho para quatro principais áreas de estudo futuro. Uma destas possibilidades é realizar um estudo e subsequente equacionamento das não idealidades dos componentes físicos do sistema para considerar o efeito destes na simulação e na obtenção das LMIs para o cálculo do controlador, levando a uma maior fidelidade ao aplicar o controlador calculado em um sistema físico. Uma outra possibilidade seria de se utilizar realimentação dinâmica para o caso considerando a saturação, de forma a poder ser realizado o projeto do controlador, não apenas sua análise. A

terceira opção seria a de considerar critérios de desempenho transitório ao realizar as soluções, como por exemplo utilizar decaimento exponencial. A última área seria, tendo em mãos o resultado encontrado neste trabalho, aplicar o controlador e os circuitos de Chua em sistemas de comunicação segura, o que é um uso comum de sincronização de sistemas caóticos, e verificar o quão seguro os dados enviados estão, o que pode ser feito analogicamente ou digitalmente.

Referências Bibliográficas

- BOYD, STEPHEN et al. **Linear Matrix Inequalities in System and Control Theory**. SIAM Studies in Applied Mathematics, Vol. 15. 1994
- CAMPOS, CLÁUDIO D. et al. **Uma abordagem para sincronização de sistemas caóticos baseada em técnicas de controle robusto H_∞ a tempo discreto**. In: XVI congresso brasileiro de Automática – CBA'2006, 2006, Salvador.
- CROSS, MICHAEL. **Introduction to Chaos**. Cal Tech, 2000 Disponível em: <<http://crossgroup.caltech.edu/>>. Acesso em: 07/06/2015
- ELGAR, STEVE; KENNEDY, MICHAEL PETER. **Bispectral Analysis of Chua's Circuit**. Journal of Circuits, Systems and Computers, Vol. 3, No. 1. 1993
- KELLERT, STEPHEN H. **In the Wake of Chaos: Unpredictable Order in Dynamical Systems**. Chicago: University of Chicago Press, 1993
- OGATA, KATSUHIKO. **Modern Control Engineering**. 3.ed. New Jersey: Prentice Hall, 1997
- SILVA JUNIOR, J.M. Gomes et al. **Dynamic output feedback stabilization for systems with sector-bounded nonlinearities and saturating actuators**. Journal of the Franklin Institute, Vol. 350, No. 3. 2013
- TARBOURIECH, Sophie ; GARCIA, G. ; GOMES DA SILVA JR, João Manoel ; QUEINNEC, Isabelle . **Stability and Stabilization of Linear Systems with Saturating Actuators**. 1. ed. London: Springer-Verlag, 2011. v. 1.
- WEN, GUILIN et al. **Synthesis for robust synchronization of chaotic systems under output feedback control with multiple random delays**. Chaos, Solitons & Fractals, Vol. 29, No. 5. 2006

Apêndice A – Estabilidade de Liapunov

Existem diversos métodos de análise de estabilidade para sistemas de controle lineares invariantes no tempo, como o critério de Nyquist ou de Routh-Hurwitz. Entretanto determinar se um sistema não linear é estável pode ser extremamente difícil ou até mesmo impossível e os métodos anteriores não são aplicáveis a estes sistemas. Um critério para análise de sistemas não lineares é a análise de estabilidade de Liapunov.

Liapunov propôs dois modos de determinar a estabilidade de sistemas descritos por equações diferenciais, conhecidos como primeiro e segundo método de Liapunov, respectivamente. No primeiro método a solução explícita da equação diferencial é utilizada, mas no segundo isto não é necessário. Este fato é vantajoso quando as equações são não lineares, pois a resolução destas muitas vezes é trabalhosa.

Definindo o sistema como sendo (31)

$$\dot{\mathbf{x}} = f(\mathbf{x}, t) \quad (31)$$

onde \mathbf{x} é um vetor de estados n -dimensional e $f(\mathbf{x}, t)$ é um vetor n cujos elementos são funções de $x_1, x_2, x_3, \dots, x_n$ e t . A solução de (31) será indicada como sendo $\varphi(t; \mathbf{x}_0, t_0)$, onde \mathbf{x}_0 são as condições iniciais em $t = t_0$ e t é o instante estudado, portanto:

$$\varphi(t_0; \mathbf{x}_0, t_0) = \mathbf{x}_0$$

Para o sistema (31), um estado \mathbf{x}_e tal que

$$f(\mathbf{x}_e, t) = 0 \text{ para qualquer } t \quad (32)$$

é chamado de estado de equilíbrio do sistema. Caso o sistema (31) seja linear e invariante no tempo haverá apenas uma solução para o estado de equilíbrio, mas para sistemas não lineares pode haver um ou mais estados de equilíbrio. Nota-

se que para determinar o estado de equilíbrio não é necessário resolver as equações diferenciais em (31), apenas a equação em (32).

Para determinar a estabilidade como definida por Liapunov cria-se uma região esférica de raio k em volta de um estado de equilíbrio \mathbf{x}_e .

$$\|\mathbf{x} - \mathbf{x}_e\| \leq k$$

onde $\|\mathbf{x} - \mathbf{x}_e\|$ é a norma Euclidiana definida como

$$\|\mathbf{x} - \mathbf{x}_e\| = \sqrt{[(x_1 - x_{1e})^2 + (x_2 - x_{2e})^2 + \dots + (x_n - x_{ne})^2]}$$

Seja a região $S(\delta)$ consistida de todos os pontos que satisfaçam

$$\|\mathbf{x}_0 - \mathbf{x}_e\| \leq \delta$$

e seja $S(\varepsilon)$ definida como todos os pontos que satisfaçam

$$\|\varphi(t; \mathbf{x}_0, t_0) - \mathbf{x}_e\| \leq \varepsilon \text{ para todo } t \geq t_0$$

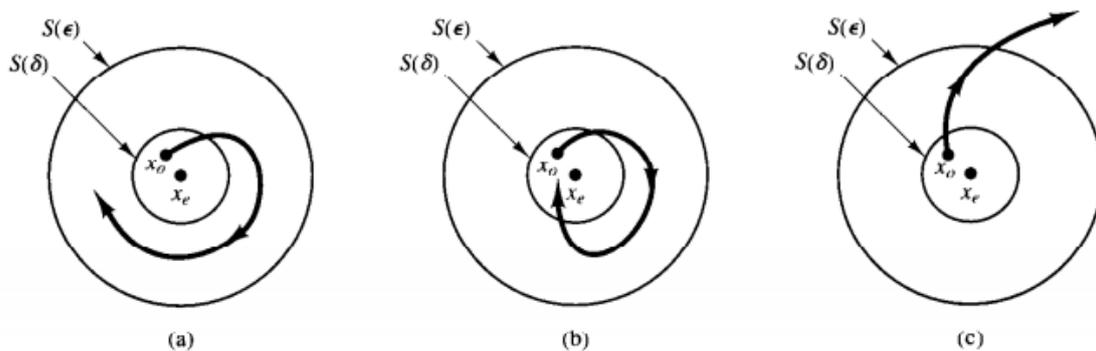
Um estado de equilíbrio \mathbf{x}_e é considerado estável no sentido de Liapunov se para cada $S(\varepsilon)$ existe um $S(\delta)$ correspondente cuja trajetória se inicia em $S(\delta)$ e não sai dos limites de $S(\varepsilon)$ quando t é incrementado. δ depende de ε e, geralmente, de t_0 . Caso δ não dependa de t_0 o estado de equilíbrio será uniformemente estável.

Quanto à classificação, o estado de equilíbrio pode ser classificado ainda como assintoticamente estável local ou no sentido global ou instável. A denominação assintoticamente estável está atribuída para o caso em que, sem violar a condição de estabilidade, a trajetória da solução converge para \mathbf{x}_e . Caso a solução seja assintoticamente estável para qualquer valor inicial ela também se denominará no sentido global. Uma característica de sistemas assintoticamente estáveis no sentido global é que deve existir apenas um estado de equilíbrio. O estado de equilíbrio será instável se para algum valor de estado inicial em que a trajetória da solução saia da região $S(\varepsilon)$.

Na prática, determinar se o sistema é estável assintoticamente é mais importante do que se ele é apenas estável. Além disso, como a estabilidade assintótica é um conceito local, saber que para uma determinada região o sistema tem este comportamento pode não ser suficiente para garantir o funcionamento adequado do sistema. Para tanto, é importante encontrar a região no qual o estado de equilíbrio é assintoticamente estável, conhecida como domínio de atração. Assim, sempre que o ponto de partida da trajetória se encontrar no domínio de atração ele irá convergir para \mathbf{x}_e .

Para problemas de controle é desejável que o sistema seja estável assintoticamente no sentido global. Caso não seja este o caso, então deve-se garantir que o domínio de atração seja grande o suficiente para que mesmo com qualquer perturbação a trajetória ainda se manterá nesta região. Um exemplo bidimensional deste critério está representado na Figura 29.

Figura 29 – 3 exemplos de sistemas utilizando o critério de Liapunov. Em (a) o estado de equilíbrio é estável, em (b) o estado de equilíbrio é assintoticamente estável e em (c) o estado de equilíbrio é instável.



Fonte: Ogata, 1997

Deve-se considerar, entretanto, que a definição de estabilidade pode variar dependendo do conceito utilizado. Para definições clássicas de controle apenas estados de equilíbrio assintoticamente estáveis são considerados estáveis, com estados de equilíbrio meramente estáveis no sentido de Liapunov mas não assintoticamente sendo considerados instáveis. Portanto é importante ser claro quanto à definição de estabilidade utilizada.

Antes de se aprofundar no segundo método de Liapunov, algumas definições devem ser feitas quanto a funções escalares. Uma função escalar $V(\mathbf{x})$ é definida como sendo estritamente positiva se seu valor é maior que zero

para qualquer valor não nulo do estado \mathbf{x} e se para $V(\mathbf{0}) = 0$. A função escalar será semi-positiva caso haja valores de \mathbf{x} que façam com que a função tenha um valor nulo, mas nunca negativo. A função é denominada como sendo estritamente negativa caso a função oposta a sua, ou seja, $-V(\mathbf{x})$ seja positiva. O mesmo vale para funções semi-negativas, se a função oposta é semi-positiva, a função será semi-negativa.

Sabe-se que um sistema é estável se a energia do mesmo (uma função estritamente positiva) decresce continuamente com o tempo até um ponto de equilíbrio, ou seja, a derivada da energia em função do tempo deve ser uma função negativa. O segundo método de Liapunov baseia-se na generalização desse comportamento. Se o sistema é assintoticamente estável, então a energia interna do sistema decresce com o incremento do tempo até atingir o seu valor mínimo no estado de equilíbrio. Enquanto esse efeito é intuitivo para sistemas baseados na realidade, para sistemas puramente matemáticos pode ser difícil encontrar uma função adequada para expressar energia.

Para contornar este problema, Liapunov definiu a função de Liapunov, uma função fictícia da energia do sistema. Esta função, entretanto, é mais geral do que a de energia e, portanto, mais facilmente aplicada em vários casos. Qualquer função que cumpra (31) e (32) pode ser considerada uma função de Liapunov. Ainda assim pode ser difícil de se encontrar uma função de Liapunov para sistemas complexos.

As funções de Liapunov dependem dos estados x_1, x_2, \dots, x_n , ou \mathbf{x} do sistema e do tempo, sendo definida como $V(\mathbf{x}, t)$ ou $V(\mathbf{x})$ caso a função não dependa explicitamente do tempo. A propriedade mais interessante do segundo método de Liapunov é que pode-se obter a informação da estabilidade do sistema através do comportamento do sinal, ou seja, se ela é positiva ou negativa, das funções $V(\mathbf{x})$ e de sua derivada no tempo $\dot{V}(\mathbf{x})$ sem ser necessário solucionar as equações em si.

Ou seja, considerando uma função escalar positiva $V(\mathbf{x})$, onde \mathbf{x} é um vetor, então os estados \mathbf{x} que satisfazem

$$V(\mathbf{x}) = C$$

onde C é uma constante positiva, se encontram em uma hipersuperfície fechada no espaço de estados n dimensional. Se $V(\mathbf{x}) \rightarrow \infty$ quando $\|\mathbf{x}\| \rightarrow \infty$ então a superfície fechada se estende por todo o espaço de estados. Além disso a hipersuperfície $V(\mathbf{x}) = C_1$ se encontra inteiramente dentro da superfície $V(\mathbf{x}) = C_2$ se $C_1 < C_2$.

Para um determinado sistema, se uma função escalar estritamente positiva $V(\mathbf{x})$ pode ser encontrada tal que a sua derivada temporal é estritamente negativa, então com o aumento do tempo $V(\mathbf{x})$ terá valores cada vez menores de C . Com o aumento de t , $V(\mathbf{x})$ eventualmente terá valor nulo e, portanto, \mathbf{x} também será nulo. Isto implica que a origem do sistema do espaço de estados é assintoticamente estável. O teorema principal da estabilidade de Liapunov, que é baseado neste comportamento, proporciona uma condição suficiente para determinar a estabilidade assintótica. O teorema pode ser descrito como segue:

Teorema: considerando um sistema descrito por

$$\dot{\mathbf{x}} = f(\mathbf{x})$$

onde

$$f(\mathbf{0}) = \mathbf{0} \text{ para qualquer } t$$

Se existir uma função escalar $V(\mathbf{x})$, com primeira derivada parcial contínua e que satisfaz as seguintes condições:

1. $V(\mathbf{x})$ é estritamente positiva
2. $\dot{V}(\mathbf{x})$ é estritamente negativa

Então o estado de equilíbrio do sistema na origem é uniformemente assintoticamente estável. Se $V(\mathbf{x}) \rightarrow \infty$ quando $\|\mathbf{x}\| \rightarrow \infty$ então o estado de equilíbrio é assintoticamente estável no sentido global.

Então o estado de equilíbrio na origem do sistema será uniformemente assintoticamente estável no sentido global.

Em um sistema linear invariante no tempo, se o equilíbrio é assintoticamente estável, então, como há apenas um ponto de equilíbrio, ele também será, necessariamente, estável no sentido global. Entretanto, para sistemas não lineares é possível que o ponto de equilíbrio seja localmente estável assintoticamente mas que não seja estável no sentido global. Portanto as implicações de estabilidade assintótica do estado de equilíbrio de sistemas lineares invariantes no tempo e de sistemas não lineares é muito diferente.

O teorema recentemente mencionado para estabilidade assintótica fornece condições suficientes para sistemas não lineares e condições necessárias e suficientes para sistemas lineares. Portanto o estado de equilíbrio de um sistema não linear pode ser estável mesmo que as condições especificadas pelo teorema não sejam cumpridas, apenas não haveria a conclusão de que é estável assintoticamente.

Apêndice B – LMI

B.1 – Definição

Desigualdade linear matricial, mais conhecida como LMI, ou linear matrix inequality, é muito utilizada para realizar uma restrição convexa em torno de uma determinada variável. Diversos problemas podem ser escritos como restrição por LMI, incluindo escalonamento de matrizes, interpolação, construção de funções quadráticas de Lyapunov e de funções de Lyapunov do tipo Lur'e para sistemas não lineares. Com o desenvolvimento do método de barreira, também conhecido como método de ponto interior, técnicas de LMI se tornaram altamente eficientes para a resolução destes problemas.

Algumas das aplicações de LMI focadas em problemas de engenharia são para a otimização de processamento de sinais, identificação de sistemas e teoria de controle. A utilização de técnicas de LMI é especialmente interessante para problemas de controle, pois pode-se expressar as condições de estabilidade como uma restrição. Além disso, restrições ao desempenho podem ser consideradas, fazendo com que, por exemplo, possa-se calcular um controlador a fim de melhorar o desempenho transitório do sistema. Portanto, além de informar sobre a estabilidade do sistema a resolução por LMI pode fornecer dados da performance do mesmo.

Definição: Define-se LMI como sendo:

$$F(\mathbf{x}) \triangleq F_0 + \sum_{i=1}^m x_i F_i > 0 \quad (33)$$

onde $\mathbf{x} \in \mathbf{R}^m$ é a variável e as matrizes F_i são simétricas e conhecidas, ou seja, $F_i = F_i^T \in \mathbf{R}^{n \times n}$, $i = 1, 2, \dots, m$. A desigualdade implica que $F(\mathbf{x})$ é estritamente positiva, isto é, para qualquer vetor não nulo $\mathbf{u} \in \mathbf{R}^n$, a relação $\mathbf{u}^T F(\mathbf{x}) \mathbf{u}$ também será estritamente positiva, e que a função é convexa. No caso particular em que as matrizes de ponderação F_i são diagonais a LMI resultante será um conjunto linear de desigualdades.

B.2 – Complemento de Schur

Nem sempre é possível utilizar técnicas de LMI diretamente, mas pode-se determinar se a função não linear é positiva definida indiretamente. Determina-se uma LMI na seguinte forma a partir de uma desigualdade não linear:

$$M(\mathbf{x}) = \begin{bmatrix} A(\mathbf{x}) & B(\mathbf{x}) \\ C(\mathbf{x}) & D(\mathbf{x}) \end{bmatrix} > 0 \quad (34)$$

onde A e D são matrizes simétricas e quadradas e A(**x**) e B(**x**) são um ou ambos não lineares.

A matriz de (34) possui o seguinte complemento de Schur S:

$$S(\mathbf{x}) = A(\mathbf{x}) - B(\mathbf{x})D(\mathbf{x})^{-1}C(\mathbf{x}) > 0$$

Para que M seja estritamente positiva é necessário que ambas S e D sejam estritamente positivas. Para tornar S menos complexa de se determinar pode-se fazer com que $C(\mathbf{x}) = B^T(\mathbf{x})$ e que o complemento fique da forma:

$$S(\mathbf{x}) = A(\mathbf{x}) - B(\mathbf{x})D(\mathbf{x})^{-1}B^T(\mathbf{x}) > 0 \quad (35)$$

Se (34) for resolvida e for revelado que M(**x**) é estritamente definida, sabe-se, portanto, que ambos S(**x**) e D(**x**) devem ser estritamente positivas. Além disso, como D(**x**) é estritamente definida, sabe-se que $B(\mathbf{x})D(\mathbf{x})^{-1}B^T(\mathbf{x})$ também é estritamente positivo e, portanto, $-B(\mathbf{x})D(\mathbf{x})^{-1}B^T(\mathbf{x})$ será estritamente negativo. Assim, para que S(**x**) seja estritamente positivo como foi determinado, não apenas é necessário que A(**x**) seja estritamente positivo, mas também que tenha valor superior a $B(\mathbf{x})D(\mathbf{x})^{-1}B^T(\mathbf{x})$ para qualquer **x**. Portanto é possível descobrir se A(**x**) e D(**x**) são estritamente positivas apenas resolvendo (34). Esse método de utilizar complemento de Schur pode ser utilizado em várias situações para obter uma LMI no mesmo formato de (34).

B.3 – S procedure

Frequentemente haverá casos em que uma determinada função quadrática sempre possua um valor inferior à outra função quadrática para cada ponto. Essa limitação pode ser expressa em alguns casos como uma LMI. Considere a seguinte função:

$$F_i(\boldsymbol{\mu}) = \boldsymbol{\mu}^T T_i \boldsymbol{\mu} + 2\mathbf{u}_i^T \boldsymbol{\mu} + v_i, \quad i = 0, \dots, p$$

onde T é uma matriz simétrica.

Deseja-se saber se $F_0(\boldsymbol{\mu}) > 0, \forall \boldsymbol{\mu}$. Satisfazendo $F_i(\boldsymbol{\mu}) > 0, \forall \boldsymbol{\mu}, i = 1, \dots, p$. Neste caso, F_0 será estritamente positiva, se existirem escalares $S_i, i=1, \dots, p$ positivos, tais que para qualquer $\boldsymbol{\mu}$:

$$F_0(\boldsymbol{\mu}) - \sum_{i=1}^p S_i F_i(\boldsymbol{\mu}) > 0 \quad (36)$$

Como $S_i F_i(\boldsymbol{\mu})$ serão todos positivos, então o somatório resultará num número positivo e para (35) for válido, $F_0(\boldsymbol{\mu})$ deve ser estritamente positivo. A equação (36) pode ser reescrita na forma:

$$\begin{bmatrix} T_0 & \mathbf{u}_0 \\ \mathbf{u}_0^T & v_0 \end{bmatrix} - \sum_{i=1}^p S_i \begin{bmatrix} T_i & \mathbf{u}_i \\ \mathbf{u}_i^T & v_i \end{bmatrix} > 0 \quad (36)$$

Apêndice C – Códigos do MATLAB

Encontra-se em anexo os códigos utilizados no MATLAB. Primeiramente o código para o sistema com atraso, depois para o sistema sem atraso e, por último, para o sistema com saturação.

LMI_Time_Delay.m

```
clear all
A = [-15.6 15.6 0; 1 -1 1; 0 -28 0];
B = [15.6; 0; 0];
C = [1; 0; 0];
g = 1.2;
delta = 0.5;
TAU1 = 1;
TAU2 = 0.1;
H = [1 0 0; 0 1 0];
E1 = [ 1 0; 0 0];
E2 = [0 0; 0 1];

setlmis([]);
P = lmivar(1, [3 1]); % P = 1
Z1 = lmivar(1, [3 1]); % Z1 = 2
Z2 = lmivar(1, [3 1]); % Z2 = 3
lambda = lmivar(1, [1 0]); % lambda = 4
S1 = lmivar(1, [1 0]); % S1 = 5
S2 = lmivar(1, [1 0]); % S2 = 6
R = lmivar(1, [3 1]); % R = 7
K0 = lmivar(2, [3 2]); % K0 = 8
W1 = lmivar(1, [3 1]); % W1 = 9
W2 = lmivar(1, [3 1]); % W2 = 10
W3 = lmivar(2, [1 3]); % W3 = 11
W4 = lmivar(2, [1 3]); % W4 = 12
W5 = lmivar(1, [3 1]); % W5 = 13
W6 = lmivar(1, [3 1]); % W6 = 14
N1 = lmivar(1, [3 1]); % N1 = 15
N2 = lmivar(1, [3 1]); % N2 = 16
N3 = lmivar(2, [1 3]); % N3 = 17
N4 = lmivar(2, [1 3]); % N4 = 18
N5 = lmivar(1, [3 1]); % N5 = 19
N6 = lmivar(1, [3 1]); % N6 = 20

lmiterm([1 1 1 R],1,-A,'s'); % LMI #1: -R*A-A'*R'
lmiterm([1 1 1 N1],1,1,'s'); % LMI #1: N1+N1'
lmiterm([1 1 1 W1],1,1,'s'); % LMI #1: W1+W1'
lmiterm([1 2 1 -P],1,1); % LMI #1: P'
lmiterm([1 2 1 -R],1,1); % LMI #1: R'
lmiterm([1 2 1 R],delta,-A); % LMI #1: -delta*R*A
lmiterm([1 2 1 N2],1,1); % LMI #1: N2
lmiterm([1 2 1 W2],1,1); % LMI #1: W2
lmiterm([1 2 2 R],.5*delta,1,'s'); % LMI #1: delta*R
lmiterm([1 2 2 Z1],.5*TAU1,1,'s'); % LMI #1: TAU1*Z1
lmiterm([1 2 2 Z2],.5*TAU2,1,'s'); % LMI #1: TAU2*Z2
lmiterm([1 2 2 -R],.5*delta,1,'s'); % LMI #1: delta*R'
lmiterm([1 2 2 -Z1],.5*TAU1,1,'s'); % LMI #1: TAU1*Z1'
```

```

lmiterm([1 2 2 -Z2],.5*TAU2,1,'s');           % LMI #1: TAU2*Z2'
lmiterm([1 3 1 S1],g*C',-1);                 % LMI #1: -g*C'*S1
lmiterm([1 3 1 N3],1,1);                     % LMI #1: N3
lmiterm([1 3 1 W3],1,1);                     % LMI #1: W3
lmiterm([1 3 2 -lambda],1,C');               % LMI #1: lambda*C'
lmiterm([1 3 3 S1],.5*2,-1,'s');            % LMI #1: -2*S1
lmiterm([1 4 1 S2],g*C',-1);                 % LMI #1: -g*C'*S2
lmiterm([1 4 1 -R],B',-1);                   % LMI #1: -B'*R'
lmiterm([1 4 1 N4],1,1);                     % LMI #1: N4
lmiterm([1 4 1 W4],1,1);                     % LMI #1: W4
lmiterm([1 4 2 -R],delta*B',-1);            % LMI #1: -delta*B'*R'
lmiterm([1 4 4 S2],.5*2,-1,'s');            % LMI #1: -2*S2
lmiterm([1 5 1 -K0],H'*E1',1);              % LMI #1: H'*E1'*K0'
lmiterm([1 5 1 -N1],1,-1);                  % LMI #1: -N1'
lmiterm([1 5 1 N5],1,1);                     % LMI #1: N5
lmiterm([1 5 1 W5],1,1);                     % LMI #1: W5
lmiterm([1 5 2 -K0],delta*H'*E1',1);        % LMI #1: delta*H'*E1'*K0'
lmiterm([1 5 2 -N2],1,-1);                  % LMI #1: -N2'
lmiterm([1 5 3 -N3],1,-1);                  % LMI #1: -N3'
lmiterm([1 5 4 -N4],1,-1);                  % LMI #1: -N4'
lmiterm([1 5 5 N5],1,-1,'s');               % LMI #1: -N5-N5'
lmiterm([1 6 1 -K0],H'*E2',1);              % LMI #1: H'*E2'*K0'
lmiterm([1 6 1 -W1],1,-1);                  % LMI #1: -W1'
lmiterm([1 6 1 N6],1,1);                     % LMI #1: N6
lmiterm([1 6 1 W6],1,1);                     % LMI #1: W6
lmiterm([1 6 2 -K0],delta*H'*E2',1);        % LMI #1: delta*H'*E2'*K0'
lmiterm([1 6 2 -W2],1,-1);                  % LMI #1: -W2'
lmiterm([1 6 3 -W3],1,-1);                  % LMI #1: -W3'
lmiterm([1 6 4 -W4],1,-1);                  % LMI #1: -W4'
lmiterm([1 6 5 -W5],1,-1);                  % LMI #1: -W5'
lmiterm([1 6 5 N6],1,-1);                   % LMI #1: -N6
lmiterm([1 6 6 W6],1,-1,'s');               % LMI #1: -W6-W6'
lmiterm([1 7 1 -N1],TAU1,1);                % LMI #1: TAU1*N1'
lmiterm([1 7 2 -N2],TAU1,1);                % LMI #1: TAU1*N2'
lmiterm([1 7 3 -N3],TAU1,1);                % LMI #1: TAU1*N3'
lmiterm([1 7 4 -N4],TAU1,1);                % LMI #1: TAU1*N4'
lmiterm([1 7 5 -N5],TAU1,1);                % LMI #1: TAU1*N5'
lmiterm([1 7 6 -N6],TAU1,1);                % LMI #1: TAU1*N6'
lmiterm([1 7 7 Z1],.5*TAU1,-1,'s');         % LMI #1: -TAU1*Z1
lmiterm([1 8 1 -W1],TAU2,1);                % LMI #1: TAU2*W1'
lmiterm([1 8 2 -W2],TAU2,1);                % LMI #1: TAU2*W2'
lmiterm([1 8 3 -W3],TAU2,1);                % LMI #1: TAU2*W3'
lmiterm([1 8 4 -W4],TAU2,1);                % LMI #1: TAU2*W4'
lmiterm([1 8 5 -W5],TAU2,1);                % LMI #1: TAU2*W5'
lmiterm([1 8 6 -W6],TAU2,1);                % LMI #1: TAU2*W6'
lmiterm([1 8 8 Z2],.5*TAU2,-1,'s');         % LMI #1: -TAU2*Z2

lmiterm([-2 1 1 P],1,1);                    % LMI #2: P

lmiterm([-3 1 1 Z1],1,1);                   % LMI #3: Z1

lmiterm([-4 1 1 Z2],1,1);                   % LMI #4: Z2

lmiterm([-5 1 1 lambda],1,1);               % LMI #5: lambda

lmiterm([-6 1 1 S1],1,1);                   % LMI #6: S1

lmiterm([-7 1 1 S2],1,1);                   % LMI #7: S2

TimeDelay=getlimis;

```

```
[tmin,xfegas] = feasp(TimeDelay);
K0o= dec2mat(TimeDelay, xfegas, K0);
Ro= dec2mat(TimeDelay, xfegas, R);
K=Ro^-1*K0o
```

LMI_No_Time_Delay.m

```
clear all
A = [-15.6 15.6 0; 1 -1 1; 0 -28 0];
B = [15.6; 0; 0];
C = [1; 0; 0];
g = 1.2;
delta = 0.5;

H = [1 0 0; 0 1 0];
E1 = [ 1 0; 0 0];
E2 = [0 0; 0 1];

setlmis([]);
P = lmivar(1, [3 1]); % P = 1
Z1 = lmivar(1, [3 1]); % Z1 = 2
Z2 = lmivar(1, [3 1]); % Z2 = 3
lambda = lmivar(1, [1 0]); % lambda = 4
S1 = lmivar(1, [1 0]); % S1 = 5
S2 = lmivar(1, [1 0]); % S2 = 6
R = lmivar(1, [3 1]); % R = 7
K0 = lmivar(2, [3 2]); % K0 = 8
W1 = lmivar(1, [3 1]); % W1 = 9
W2 = lmivar(1, [3 1]); % W2 = 10
W3 = lmivar(2, [1 3]); % W3 = 11
W4 = lmivar(2, [1 3]); % W4 = 12
N1 = lmivar(1, [3 1]); % N1 = 13
N2 = lmivar(1, [3 1]); % N2 = 14
N3 = lmivar(2, [1 3]); % N3 = 15
N4 = lmivar(2, [1 3]); % N4 = 16

lmiterm([1 1 1 R],1,-A,'s'); % LMI #1: -R*A-A'*R'
lmiterm([1 1 1 N1],1,1,'s'); % LMI #1: N1+N1'
lmiterm([1 1 1 W1],1,1,'s'); % LMI #1: W1+W1'
lmiterm([1 1 1 K0],.5*1,E1*H,'s'); % LMI #1: K0*E1*H
lmiterm([1 1 1 K0],.5*1,E2*H,'s'); % LMI #1: K0*E2*H
lmiterm([1 2 1 -P],1,1); % LMI #1: P'
lmiterm([1 2 1 -R],1,1); % LMI #1: R'
lmiterm([1 2 1 R],delta,-A); % LMI #1: -delta*R*A
lmiterm([1 2 1 N2],1,1); % LMI #1: N2
lmiterm([1 2 1 W2],1,1); % LMI #1: W2
lmiterm([1 2 1 -K0],delta*H'*E1',1); % LMI #1:
delta*H'*E1'*K0'
lmiterm([1 2 1 -K0],delta*H'*E2',1); % LMI #1:
delta*H'*E2'*K0'
lmiterm([1 2 2 R],.5*delta,1,'s'); % LMI #1: delta*R
lmiterm([1 2 2 -R],.5*delta,1,'s'); % LMI #1: delta*R'
lmiterm([1 3 1 S1],g*C',-1); % LMI #1: -g*C'*S1
lmiterm([1 3 1 N3],1,1); % LMI #1: N3
lmiterm([1 3 1 W3],1,1); % LMI #1: W3
lmiterm([1 3 2 -lambda],1,C'); % LMI #1: lambda'*C'
```

```

lmiterm([1 3 3 S1],.5*2,-1,'s');
lmiterm([1 4 1 S2],g*C',-1);
lmiterm([1 4 1 -R],B',-1);
lmiterm([1 4 1 N4],1,1);
lmiterm([1 4 1 W4],1,1);
lmiterm([1 4 2 -R],delta*B',-1);
lmiterm([1 4 4 S2],.5*2,-1,'s');
lmiterm([1 5 1 -N1],1,1);
lmiterm([1 5 2 -N2],1,1);
lmiterm([1 5 3 -N3],1,1);
lmiterm([1 5 4 -N4],1,1);
lmiterm([1 5 5 Z1],1,-1);
lmiterm([1 6 1 -W1],1,1);
lmiterm([1 6 2 -W2],1,1);
lmiterm([1 6 3 -W3],1,1);
lmiterm([1 6 4 -W4],1,1);
lmiterm([1 6 6 Z2],1,-1);

lmiterm([-2 1 1 P],1,1);

lmiterm([-3 1 1 Z1],1,1);

lmiterm([-4 1 1 Z2],1,1);

lmiterm([-5 1 1 lambda],1,1);

lmiterm([-6 1 1 S1],1,1);

lmiterm([-7 1 1 S2],1,1);

No_TimeDelay=getlmis;

[tmin,xffeas] = feasp(No_TimeDelay);
K0o= dec2mat(No_TimeDelay, xfeas, K0);
Ro= dec2mat(No_TimeDelay, xfeas, R);
K=Ro^-1*K0o
% LMI #1: -2*S1
% LMI #1: -g*C'*S2
% LMI #1: -B'*R'
% LMI #1: N4
% LMI #1: W4
% LMI #1: -delta*B'*R'
% LMI #1: -2*S2
% LMI #1: N1'
% LMI #1: N2'
% LMI #1: N3'
% LMI #1: N4'
% LMI #1: -Z1
% LMI #1: W1'
% LMI #1: W2'
% LMI #1: W3'
% LMI #1: W4'
% LMI #1: -Z2

% LMI #2: P

% LMI #3: Z1

% LMI #4: Z2

% LMI #5: lambda

% LMI #6: S1

% LMI #7: S2

No_TimeDelay=getlmis;

[tmin,xffeas] = feasp(No_TimeDelay);
K0o= dec2mat(No_TimeDelay, xfeas, K0);
Ro= dec2mat(No_TimeDelay, xfeas, R);
K=Ro^-1*K0o

```

LMI_Saturation.m

```

clear all
A = [-15.6 15.6 0; 1 -1 1; 0 -28 0];
B = [15.6; 0; 0];
C = [1; 0; 0];
g = 1.2;
delta = 0.5;
umax = 10;
xm0= [-0.2;-0.33;0.2]
xs0= [0.5;-0.1;0.66]
e0=xm0-xs0;
H=[1 0 0; 0 1 0];
H1=[1;0;0];
H2=[0;1;0];
H3=[0;0;1];

```

```

K=[114.6646 130.9447; 0.3255 3.7626; -7.5567 48.5296];
copt_best=10;
epsilon_best = 0;
for epsilon=1:40

S3=[epsilon 0 0; 0 epsilon 0; 0 0 epsilon];

setlmis([]);
P = lmivar(1, [3 1]); % P = 1
Z1 = lmivar(1, [3 1]); % Z1 = 2
Z2 = lmivar(1, [3 1]); % Z2 = 3
lambda = lmivar(1, [1 0]); % lambda = 4
S1 = lmivar(1, [1 0]); % S1 = 5
S2 = lmivar(1, [1 0]); % S2 = 6
R = lmivar(1, [3 1]); % R = 7
G = lmivar(2, [3 2]); % G = 8
W1 = lmivar(1, [3 1]); % W1 = 9
W2 = lmivar(1, [3 1]); % W2 = 10
W3 = lmivar(2, [1 3]); % W3 = 11
W4 = lmivar(2, [1 3]); % W4 = 12
W5 = lmivar(1, [3 1]); % W5 = 13
N1 = lmivar(1, [3 1]); % N1 = 14
N2 = lmivar(1, [3 1]); % N2 = 15
N3 = lmivar(2, [1 3]); % N3 = 16
N4 = lmivar(2, [1 3]); % N4 = 17
N5 = lmivar(1, [3 1]); % N5 = 18

lmiterm([1 1 1 R],1,-A,'s'); % LMI #1: -R*A-A'*R'
lmiterm([1 1 1 R],.5*1,K*H,'s'); % LMI #1: R*K*H
lmiterm([1 1 1 N1],1,1,'s'); % LMI #1: N1+N1'
lmiterm([1 1 1 W1],1,1,'s'); % LMI #1: W1+W1'
lmiterm([1 1 1 -R],.5*H'*K',1,'s'); % LMI #1: H'*K'*R'
lmiterm([1 2 1 -P],1,1); % LMI #1: P'
lmiterm([1 2 1 R],1,1); % LMI #1: R
lmiterm([1 2 1 R],delta,K*H); % LMI #1: delta*R*K*H
lmiterm([1 2 1 R],delta,-A); % LMI #1: -delta*R*A
lmiterm([1 2 1 N2],1,1); % LMI #1: N2
lmiterm([1 2 1 W2],1,1); % LMI #1: W2
lmiterm([1 2 2 R],.5*delta,1,'s'); % LMI #1: delta*R
lmiterm([1 2 2 -R],.5*delta,1,'s'); % LMI #1: delta*R'
lmiterm([1 3 1 S1],g*C',-1); % LMI #1: -g*C'*S1
lmiterm([1 3 1 N3],1,1); % LMI #1: N3
lmiterm([1 3 1 W3],1,1); % LMI #1: W3
lmiterm([1 3 2 -lambda],1,C'); % LMI #1: lambda'*C'
lmiterm([1 3 3 S1],.5*2,-1,'s'); % LMI #1: -2*S1
lmiterm([1 4 1 S2],g*C',-1); % LMI #1: -g*C'*S2
lmiterm([1 4 1 -R],B',-1); % LMI #1: -B'*R'
lmiterm([1 4 1 N4],1,1); % LMI #1: N4
lmiterm([1 4 1 W4],1,1); % LMI #1: W4
lmiterm([1 4 2 -R],delta*B',-1); % LMI #1: -delta*B'*R'
lmiterm([1 4 4 S2],.5*2,-1,'s'); % LMI #1: -2*S2
lmiterm([1 5 1 -R],1,1); % LMI #1: R'
lmiterm([1 5 1 G],S3,-H); % LMI #1: -S3*G*H
lmiterm([1 5 1 N5],1,1); % LMI #1: N5
lmiterm([1 5 1 W5],1,1); % LMI #1: W5
lmiterm([1 5 2 -R],delta,1); % LMI #1: delta*R'
lmiterm([1 5 5 0],-2*S3); % LMI #1: -2*S3
lmiterm([1 6 1 -N1],1,1); % LMI #1: N1'

```

```

lmiterm([1 6 2 -N2],1,1); % LMI #1: N2'
lmiterm([1 6 3 -N3],1,1); % LMI #1: N3'
lmiterm([1 6 4 -N4],1,1); % LMI #1: N4'
lmiterm([1 6 5 -N5],1,1); % LMI #1: N5'
lmiterm([1 6 6 Z1],1,-1); % LMI #1: -Z1
lmiterm([1 7 1 -W1],1,1); % LMI #1: W1'
lmiterm([1 7 2 -W2],1,1); % LMI #1: W2'
lmiterm([1 7 3 -W3],1,1); % LMI #1: W3'
lmiterm([1 7 4 -W4],1,1); % LMI #1: W4'
lmiterm([1 7 5 -W5],1,1); % LMI #1: W5'
lmiterm([1 7 7 Z2],1,-1); % LMI #1: -Z2

lmiterm([-2 1 1 P],1,1); % LMI #2: P
lmiterm([-2 2 1 G],H1',-H); % LMI #2: -H1'*G*H
lmiterm([-2 2 1 0],H1'*K*H); % LMI #2: H1'*K*H
lmiterm([-2 2 2 0],umax^2); % LMI #2: umax^2

lmiterm([-3 1 1 P],1,1); % LMI #3: P
lmiterm([-3 2 1 G],H2',-H); % LMI #3: -H2'*G*H
lmiterm([-3 2 1 0],H2'*K*H); % LMI #3: H2'*K*H
lmiterm([-3 2 2 0],umax^2); % LMI #3: umax^2

lmiterm([-4 1 1 P],1,1); % LMI #4: P
lmiterm([-4 2 1 G],H3',-H); % LMI #4: -H3'*G*H
lmiterm([-4 2 1 0],H3'*K*H); % LMI #4: H3'*K*H
lmiterm([-4 2 2 0],umax^2); % LMI #4: umax^2

lmiterm([5 1 1 P],e0',e0); % LMI #5: e0'*P*e0
lmiterm([-5 1 1 0],1); % LMI #5: 1

lmiterm([-6 1 1 P],1,1); % LMI #6: P

lmiterm([-7 1 1 Z1],1,1); % LMI #7: Z1

lmiterm([-8 1 1 Z2],1,1); % LMI #8: Z2

lmiterm([-9 1 1 lambda],1,1); % LMI #9: lambda

lmiterm([-10 1 1 S1],1,1); % LMI #10: S1

lmiterm([-11 1 1 S2],1,1); % LMI #11: S2

Saturation=getlmis;

nvd=decnbr(Saturation);
c=zeros(nvd, 1);
for j=1:nvd
    [Pj]=defcx(Saturation, j, P);
    c(j)=sum(diag(Pj));
end

options=[1e-5, 0, 0, 0, 0];

[copt, xopt]=mincx(Saturation, c, options);

if copt < copt_best

    copt_best = copt;

```

```
epsilon_best = epsilon;  
end  
end
```