

UNIVERSIDADE DE CAXIAS DO SUL – UCS
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA – CCET
ENGENHARIA DE CONTROLE E AUTOMAÇÃO

VINICIUS BINOTTI

**ANÁLISE DE SISTEMAS COM RESTRIÇÕES NO ATUADOR POR MEIO DA
MODELAGEM POR REGIÕES DE SATURAÇÃO**

Caxias do Sul
2015

Vinicius Binotti

**ANÁLISE DE SISTEMAS COM RESTRIÇÕES NO ATUADOR POR MEIO DA
MODELAGEM POR REGIÕES DE SATURAÇÃO**

Trabalho de Conclusão de Curso apresentado
como parte dos requisitos para a obtenção do
título de Engenheiro de Controle e Automação
da Universidade de Caxias do Sul.

Orientador:
Prof. Dr. Fernando Augusto Bender

Caxias do Sul
2015

Vinicius Binotti

**ANÁLISE DE SISTEMAS COM RESTRIÇÕES NO ATUADOR POR MEIO DA
MODELAGEM POR REGIÕES DE SATURAÇÃO**

Trabalho de Conclusão de Curso apresentado
como parte dos requisitos para a obtenção do
título de Engenheiro de Controle e Automação
da Universidade de Caxias do Sul.

Orientador:

Prof. Dr. Fernando Augusto Bender

Aprovado em ____/____/____

Banca Examinadora

Prof. Dr. Fernando Augusto Bender
Universidade de Caxias do Sul - UCS

Prof. Dr. Mauricio Zardo Oliveira
Universidade de Caxias do Sul - UCS

Prof. Dr. Julio Cesar Ceballos Aya
Universidade de Caxias do Sul - UCS

*“The known is finite, the unknown infinite;
intellectually we stand on an islet in the midst of an illimitable ocean of inexplicability.
Our business in every generation is to reclaim a little more land”.*

Thomas Henry Huxley - 1882

RESUMO

Este trabalho de conclusão de curso aborda a análise de sistemas lineares de tempo contínuo e discreto sujeitos a restrições no atuador. O desenvolvimento dessa metodologia tem por objetivo explorar a modelagem da saturação conhecida por regiões de saturação para problemas formulados com base em conjuntos elipsoidais. Inicialmente, apresenta-se uma revisão bibliográfica dos principais trabalhos, definições, teoremas e lemas necessários para o entendimento deste trabalho. Um conjunto de condições na forma de Inequações Matriciais Lineares, do inglês *Linear Matrix Inequalities* (LMIs), são apresentadas para investigar a estabilidade de sistemas saturantes, em que é proposto a utilização de um elipsóide específico para cada região de saturação. Em seguida, apresenta-se um método para a obtenção de um controlador baseado na realimentação de estados que garanta a estabilidade do sistema em malha fechada. Além disso, exemplos numéricos são expostos para comprovar a efetividade dos teoremas apresentados, sendo estes resolvidos com a utilização do *software* MATLAB e o *parser* YALMIP.

Palavras-chave: Análise. Estabilidade. Regiões de Saturação. LMI.

ABSTRACT

This bachelor thesis addresses the analysis of continuous-time and discrete-time linear systems subject to actuator restrictions. The development of this methodology is aimed to explore the saturation model known as regions of saturation for problems formulated in terms of ellipsoidal sets. Firstly, it is presented a review of the main works, definitions, theorems and lemmas necessary for the understanding of this work. A set of conditions in the form of Linear Matrix Inequalities (LMI) are presented to ensure the stability of saturated systems, in which is proposed the usage of a specific ellipsoid for each region of saturation. In the sequel, it is presented a methods to obtain a state feedback controller that ensures the closed-loop stability of the system. In addition, numerical examples are exposed to verify the effectiveness of the theorems presented, being solved with MATLAB and the parser YALMIP.

Keywords: Analysis. Stability. Regions of Saturation. LMI.

LISTA DE FIGURAS

Figura 1 – A função saturação.	16
Figura 2 – Conjunto convexo e não convexo.	16
Figura 3 – Invólucro convexo de um conjunto.	17
Figura 4 – Elipsóide.	17
Figura 5 – Condição de Setor.	18
Figura 6 – Politopo.	18
Figura 7 – Diagrama de blocos de um sistema sujeito a saturação.	26
Figura 8 – Região de linearidade R_L	27
Figura 9 – Ilustração gráfica do Lema 2.6.	29
Figura 10 – Regiões de saturação do exemplo numérico (KIM; BIEN, 1994).	36
Figura 11 – Elipsóides definidos por P_1 e P_2 , respectivamente	38
Figura 12 – Elipsóides definidos por P_3 e P_4 , respectivamente.	38
Figura 13 – Elipsóides definidos por P_5 e P_j , $j = 1, \dots, 5$	39
Figura 14 – Trajetórias do sistema para condições iniciais $x_1 = 50$, $x_2 = 50$ e $x_1 =$ -75 , $x_2 = 60$, respectivamente.	39

LISTA DE SIGLAS

LMI	<i>Linear Matrix Inequality</i>
BMI	<i>Bilinear Matrix Inequality</i>
CAN	<i>Controlled Area Network</i>

LISTA DE SÍMBOLOS

\star	bloco simétrico
\forall	para todo
\in	pertence a
\subset	subconjunto de
$:$	tal que
\mapsto	mapeia para
\rightarrow	tende para
\Rightarrow	implica que
\Leftrightarrow	equivalente a
\exists	existe
\dot{x}	derivada temporal de x
$\ x\ $	norma Euclideana do vetor x
I	matrix identidade de ordem apropriada
I_n	matrix identidade de ordem n
A^T	matrix transposta da matriz real A
$Co\{\cdot\}$	invólucro convexo
$sat\{\cdot\}$	função saturação
$rank\{\cdot\}$	posto de (\cdot)
\mathfrak{R}	conjunto dos números reais
\mathfrak{R}^n	conjunto dos vetores reais de dimensão n
$\mathfrak{R}^{n \times n}$	conjunto das matrizes reais de dimensão $n \times n$
\mathbb{Z}	conjunto dos números inteiros
■	designação para o fim de teorema, corolário e lema
□	designação para o fim de prova
▽	designação para o fim de exemplo numérico

SUMÁRIO

1 INTRODUÇÃO	10
1.1 Justificativa do Trabalho	11
1.2 Objetivos	12
1.2.1 Objetivo Geral	12
1.2.2 Objetivos Específicos	12
1.3 Limites do Trabalho	12
2 CONCEITOS PRELIMINARES	13
2.1 Estabilidade no Sentido de Lyapunov	13
2.1.1 Estabilidade por Lyapunov para Sistemas Discretos no Tempo	14
2.2 Ferramentas Matemáticas	15
2.3 Sistemas com Atraso	20
2.3.1 Estabilidade de Sistemas Discretos com Atraso	21
2.3.2 Estabilidade de Sistemas Discretos com Atraso via Finsler	24
2.3.3 Exemplos Numéricos	25
2.4 Sistemas Sujeitos a Saturação	26
2.4.1 Modelagem da Saturação via Representação Politópica	28
2.4.2 Modelagem da Saturação via Não Linearidade de Setor	29
2.4.3 Modelagem por Regiões de Saturação	31
3 ANÁLISE DE SISTEMAS SATURANTES VIA REGIÕES DE SATURAÇÃO	32
3.1 Sistemas de tempo contínuo	32
3.2 Sistemas de tempo discreto	34
3.3 Exemplo Numérico	35
4 ESTABILIZAÇÃO DE SISTEMAS SATURANTES VIA REGIÕES DE SATURAÇÃO	40
5 CONCLUSÕES E PERSPECTIVAS	42
REFERÊNCIAS	43
APÊNDICE A – SCRIPT MATLAB EXEMPLO NUMÉRICO	46

1 INTRODUÇÃO

De acordo com Tewari (2002), a palavra *controle* refere-se ao ato de produzir um resultado desejado. Similarmente, em problemas na engenharia de controle deseja-se que um determinado sistema, seja ele mecânico, elétrico ou térmico se comporte de maneira esperada e que atenda certos requisitos de desempenho, definidos *a priori*. Para tanto, a realimentação juntamente com leis de controle são aplicadas para alterar o comportamento do sistema quando operando em malha fechada. Com isso, surge a necessidade de obter um modelo matemático que descreva o sistema. Diversos modelos podem representar um mesmo sistema dinâmico em que cada modelo apresentará diferentes graus de fidelidade. Na natureza, todo sistema físico possui diversos modos e não linearidades associadas ao seu comportamento. Em muitos problemas, desconsiderar dinâmicas de ordem superiores bem como as não linearidades pode produzir resultados satisfatórios atendendo aos requisitos. Em contrapartida, em problemas em que é preciso maior precisão e acuracidade no controle, essas aproximações podem não ser suficientes e, portanto, devemos ser capazes de lidar com peculiaridades do sistema. Para Mahmoud (2010), a escolha de um modelo que descreva o sistema com maior fidelidade possível é refletido diretamente na qualidade dos resultados. Em vista disso, propõem-se o estudo da estabilidade e estabilização de sistemas sujeitos ao atraso e a saturação.

Este trabalho pretende dar continuidade ao trabalho (BINOTTI, 2015), que apresentou duas metodologias para estabilização, através da realimentação de estados, de sistemas discreto com restrições no atuador e com a presença de atraso. Essas duas abordagens diferem na maneira em que a saturação é modelada, sendo utilizado a modelagem politópica e a pela não linearidade de setor. Entretanto, em ambos procedimentos não foi possível a obtenção de condições na forma de LMIs, dificultando assim que os teoremas propostos sejam resolvidos numericamente. Além das duas modelagens da saturação utilizadas em (BINOTTI, 2015), existe ainda a modelagem conhecida por regiões de saturação. Esta por sua vez, não tem sido tão explorada para conjuntos elipsóidais quanto as outras representações disponíveis. Existem dois principais resultados para a análise de sistemas saturantes na literatura que faz uso das regiões de saturação. Esses resultados foram apresentados em (FONG; HSU, 2000) e (JOHANSSON; RANTZER, 1998), respectivamente, onde teoremas com condições na forma de BMIs são propostos. Esses mesmo procedimentos são também exposto em (TARBOURIECH et al., 2011). Assim, este trabalho explora a representação da saturação por regiões de saturação para a obtenção de condições com a finalidade de garantir a estabilidade do sistema sujeitos a saturação em malha fechada.

Com o avanço das ferramentas computacionais, tornou-se possível a resolução numérica de diversos problemas complexos que anteriormente eram impossíveis de serem resolvidos. Por consequência, na década de 90 foi verificado um aumento significativo em pesquisas para formular problemas de controle por meio de Inequações Matriciais Lineares, do inglês LMIs (*Linear Matrix Inequality*). Uma das metodologias mais difundidas no meio acadêmico para síntese e análise de sistemas dinâmicos é através da utilização da Teoria de Lyapunov. Em

geral, tenta-se formular o problema por um conjunto de LMIs, onde no caso de análise tenta-se garantir a estabilidade do sistema para um dado conjunto de valores numéricos e para a síntese além da garantia de estabilidade busca-se a obtenção das variáveis do controlador. Uma das principais vantagens dessa abordagem é que pode-se aplicar a uma ampla gama de sistemas, tais como sistemas não lineares.

Este trabalho apresenta-se na seguinte estrutura:

O Capítulo 2 aborda conceitos preliminares, tais como, a estabilidade de sistemas dinâmicos no sentido de Lyapunov, estabilidade de sistemas com atraso, sistemas saturantes e ferramentas matemáticas para o desenvolvimento deste trabalho.

O Capítulo 3 apresenta uma nova metodologia para a análise de sistemas saturantes, que baseia-se na obtenção de um elipsóide para cada região de saturação.

O Capítulo 4 apresenta uma metodologia para estabilização de sistemas saturantes, utilizando a modelagem por regiões de saturação.

1.1 Justificativa do Trabalho

Sistemas dinâmicos, em que a saturação bem como a presença de atraso no tempo podem ser observados, são amplamente encontrados em diferentes áreas do conhecimento. Com a finalidade de aplicar técnicas de controle a tais sistemas, surge a necessidade de obter-se um modelo matemático que leve em consideração particularidades do sistema de interesse. Portanto, o atraso no tempo bem como não linearidades como a saturação devem estar presentes no modelo do sistema. Considerar a saturação no modelo do sistema vai além de apenas possibilitar a obtenção de melhores resultados no que diz respeito ao desempenho e performance do sistema em malha fechada. Uma vez que um sistema submetido a restrições no atuador pode se tornar instável, mesmo esse sistema sendo estável em malha aberta (TARBOURIECH et al., 2011), em algumas aplicações surge a necessidade de levar em consideração a saturação devido a questões de segurança. Existem registros de acidentes envolvendo aeronaves, onde verificou-se que em alguns casos a saturação foi o principal fator na falha dos sistemas de controle (WHY THE GRIPPEN CRASHED, 1994) e (BOLKCOM, 2005). Além disso, um dos maiores acidentes nucleares já ocorrido, o desastre de Chernobyl, também foi influenciado pela ocorrência da saturação nos atuadores da instalação (STEIN, 2003). Com base nisso, podemos observar a importância de levar em consideração essas não linearidades presentes no sistema físico para que se possa alcançar resultados melhores no desempenho e confiabilidade.

1.2 Objetivos

1.2.1 Objetivo Geral

O objetivo geral deste trabalho é obter técnicas de análise baseados na realimentação de estados para sistemas lineares de tempo contínuo e discreto sujeitos a saturação.

1.2.2 Objetivos Específicos

- Revisão bibliográfica da literatura para análise de sistemas lineares discretos com atraso.
- Revisão bibliográfica da literatura para análise de sistemas lineares discretos sujeitos a saturação.
- Propor teorema para análise de sistemas saturantes com realimentação de estados por meio da modelagem da regiões de saturação.
- Obter solução numérica para os teoremas propostos por meio dos exemplos numéricos correntes na literatura.

1.3 Limites do Trabalho

Este trabalho considera que a planta de interesse a ser controlada é linear, invariante no tempo e discreta, não sendo abordado problemas relacionados com a linearização e discretização de sistemas com atraso. Também está fora do escopo deste trabalho levar em consideração os efeitos da quantização no sinal de amplitude, uma vez que os sistemas considerados são somente discretos no tempo. Os exemplos numéricos utilizados ao longo deste trabalho foram obtidos de publicações relacionadas a este tema e são utilizados na íntegra, não representando nenhuma aplicação específica das metodologias apresentadas a um problema prático.

2 CONCEITOS PRELIMINARES

Neste capítulo é apresentado o tipo de sistema estudado e as principais definições, lemas e teorias aplicadas ao longo deste trabalho.

2.1 Estabilidade no Sentido de Lyapunov

Em 1892 na sua tese de doutorado o matemático russo Aleksandr M. Lyapunov apresentou dois métodos para analisar a estabilidade de soluções de equações diferenciais ordinárias (LYAPUNOV, 1992). Esses métodos fornecem condições suficientes na forma de LMI para determinação da estabilidade de sistemas dinâmicos. A teoria de Lyapunov, de acordo com Boyd et al. (1994), só foi de fato utilizada em problemas práticos na década de quarenta, especificamente em problemas de controle para tratar não linearidades no atuador. Além disso, é hoje a base para uma ampla gama de trabalhos acadêmicos na área da engenharia de controle, não somente para problemas de análise, mas principalmente para síntese de leis de controle. A estabilidade no sentido de Lyapunov, também conhecida como estabilidade interna, é investigada por meio da análise do ponto de equilíbrio do sistema (KHALIL, 2002). Desta maneira, o ponto de equilíbrio é considerado estável se as trajetórias do sistema iniciadas na vizinhança desse ponto permanecerem nesta região. Se as trajetórias do sistema não somente ficarem nos arredores desse ponto, como ainda convergirem ao ponto de equilíbrio, portanto, esse ponto é assintoticamente estável. Considerando o seguinte sistema autônomo de tempo contínuo

$$\dot{x} = f(x), \quad x(t_0) = x_0 \quad (2.1)$$

onde $x \in \mathbb{R}^n$ e supondo que $\bar{x} \in \mathbb{R}^n$ é um ponto de equilíbrio de (2.1), i.e., $f(\bar{x}) = 0$. Assume-se que o ponto de equilíbrio do sistema é sempre a origem, i.e. $\bar{x} = 0$, o que segundo Khalil (2002) pode ser afirmado sem nenhuma perda de generalidade uma vez que qualquer ponto de equilíbrio pode ser deslocado para a origem via troca de variáveis. Deste modo, a seguinte definição é apresentada (KHALIL, 2002):

Definição 2.1. (*Caracterização do Ponto de Equilíbrio*) O ponto de equilíbrio $\bar{x} = 0$ de (2.1) é

- *estável se, para cada $\epsilon > 0$ existir $\delta = \delta(\epsilon) > 0$ de tal maneira que*

$$\|x(t_0)\| < \delta \Rightarrow \|x(t)\| < \epsilon, \forall t \geq 0;$$

- *assintoticamente estável, se for estável e δ possa ser escolhido de tal maneira que;*
- *instável, caso contrario.*

$$\|x(t_0)\| < \delta \Rightarrow \lim_{t \rightarrow \infty} x(t) = 0.$$

O segundo método proposto por Lyapunov, também conhecido como método direto de Lyapunov, pode ser facilmente entendido com base na noção de energia de um sistema. Pode-se argumentar que um sistema físico, contendo uma determinada quantidade de energia e que com o passar do tempo parte de sua energia é constantemente dissipada, tende ao repouso. Em outras palavras, pode-se dizer que se a variação da energia do sistema for negativa para todos os instantes seguintes ao instante inicial, esse sistema é estável. É possível de fato utilizar a função que descreve a energia do sistema para investigar sua estabilidade. No entanto, nem sempre essa é uma abordagem viável. Segundo Åström e Murray (2012), a generalização dessa técnica para um sistema dinâmico arbitrário é baseado no uso de uma função de Lyapunov ao invés da função de energia e essa função deve atender certos critérios. Formalmente, tem-se a seguinte definição:

Definição 2.2. (*Função de Lyapunov*) Uma função $V(x) : \mathbb{R}^n \mapsto \mathbb{R}$ é:

- *positiva definida* se $V(x) > 0, \forall x \neq 0$ e $V(0) = 0$;
- *negativa definida* se $V(x) < 0, \forall x \neq 0$ e $V(0) = 0$;
- *positiva semidefinida* $V(x) \geq 0, \forall x \neq 0$.

Assim o seguinte teorema pode ser enunciado vide (ÅSTRÖM; MURRAY, 2012):

Teorema 2.1. (*Estabilidade por Lyapunov*) Seja $V(x)$ uma função não negativa e $\dot{V}(x)$ a derivada temporal de $V(x)$ ao longo das trajetórias do sistema (2.1), i.e.

$$\dot{V}(x) = \frac{\partial V(x)}{\partial x} \frac{dx}{dt} = \frac{\partial V(x)}{\partial x} f(x)$$

Seja $B_r = B_r(0)$ uma esfera de raio r em torno da origem tal que

$$B_r = \{x \in \mathbb{R}^n : \|x\| \leq r\}$$

Se existir um $r > 0$ de tal maneira que $V(x)$ é positiva definida e $\dot{V}(x)$ é semidefinida negativa $\forall x \in B_r$, logo $x = 0$ é localmente estável no sentido de Lyapunov. Se $V(x)$ é positiva definida e $\dot{V}(x)$ é negativa definida em B_r , logo $x = 0$ é localmente assintoticamente estável. ■

2.1.1 Estabilidade por Lyapunov para Sistemas Discretos no Tempo

Considerando o seguinte sistema discreto no tempo

$$x[k + 1] = Ax[k], \quad k \in \mathbb{Z} \tag{2.2}$$

onde $x \in \mathbb{R}^n$ e a matriz A possui dimensões apropriadas. Supondo-se que $x = 0$ é o ponto de equilíbrio, o Teorema 2.1 pode ser aplicado para se analisar a estabilidade do sistema (2.2).

Entretanto, por se tratar de um sistema discreto no tempo, busca-se uma função $V(x)$ que seja positiva definida tal que sua variação

$$\Delta V = V(x[k+1]) - V(x[k])$$

seja negativa. Uma função comumente utilizada como função de Lyapunov é a função quadrática, por ser positiva definida. Em vista disso, considera-se a função candidata de Lyapunov a seguinte função quadrática

$$V(x) = x[k]^T P x[k] \quad (2.3)$$

onde P é uma matrix positiva definida, e o seguinte teorema pode ser enunciado vide (OGATA, 1994):

Teorema 2.2. *Se existir uma matriz simétrica positiva definida P , i.e. $P = P^T > 0$ de tal maneira que*

$$A^T P A - P < 0$$

então o ponto de equilíbrio $x = 0$ do sistema (2.2) é assintoticamente estável. ■

Conforme Ogata (1994), a LMI apresentada no Teorema 2.2 é uma condição necessária e suficiente para a estabilidade assintótica do sistema (2.2).

2.2 Ferramentas Matemáticas

Matrizes simétricas aparecem com frequência durante o desenvolvimento algébrico deste trabalho. Com a finalidade de simplificar sua representação, a seguinte notação é utilizada.

$$\text{sym}\{\cdot\} : \mathfrak{R}^{n \times n} \mapsto \mathfrak{R}^{n \times n}, \text{sym}\{A\} = A + A^T$$

A função saturação, $\text{sat}(\cdot)$, é uma das principais funções consideradas neste trabalho e será extensivamente utilizada, portanto é definida como:

$$\text{sat}(u) : \mathfrak{R}^m \mapsto \mathfrak{R}^m, \text{sat}(u) = \begin{cases} u_{max}, & \text{se } u \geq u_{max} \\ u, & \text{se } u_{min} < u < u_{max} \\ u_{min}, & \text{se } u \leq u_{min} \end{cases} \quad (2.4)$$

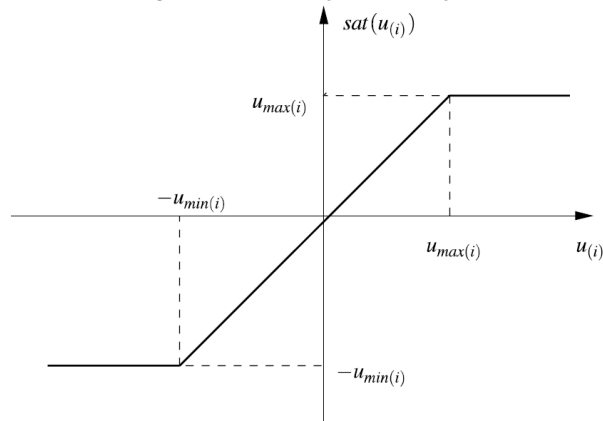
A função saturação é representada na figura 1.

Além da função saturação, a seguinte função zona morta também será necessária para representar a saturação por uma não linearidade de setor, então define-se como:

$$\psi(\cdot) : \mathfrak{R}^m \mapsto \mathfrak{R}^m, \psi(\cdot) = (\cdot) - \text{sat}(\cdot)$$

Posteriormente, modelos que representam a saturação são apresentados com a finalidade de

Figura 1 – A função saturação.



Fonte: (TARBOURIECH et al., 2011).

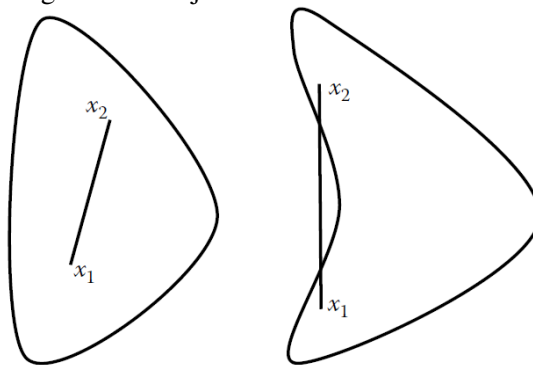
incluir-se a saturação no equacionamento do sistema. Para isso, é necessário o conhecimento de alguns conceitos matemáticos, sendo apresentado as seguintes definições de acordo com (HAI-HUA; GUANGREN, 2013) e (HINDI, 2004):

Definição 2.3. (*Conjunto Convexo*) Um conjunto \mathcal{F} é convexo se o segmento de reta entre quaisquer dois pontos em \mathcal{F} está contido em \mathcal{F} , i.e., para quaisquer $x_1, x_2 \in \mathcal{F}$ e $0 \leq \theta \leq 1$, verifica-se que

$$\theta x_1 + (1 - \theta)x_2 \in \mathcal{F}$$

Na figura 2, é possível verificar geometricamente um conjunto convexo e um não convexo.

Figura 2 – Conjunto convexo e não convexo.



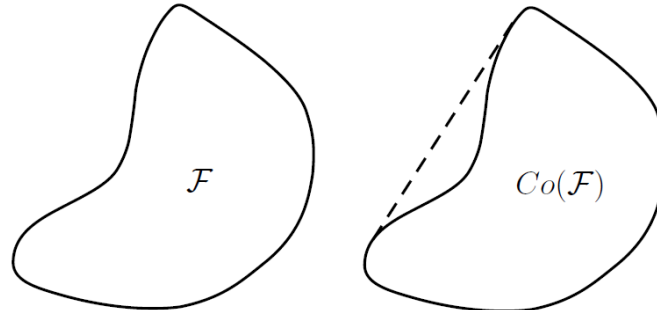
Fonte: (HAI-HUA; GUANGREN, 2013).

Definição 2.4. (*Combinação Convexa*) Sejam pontos $x_i \in \mathfrak{R}^n$ e $\theta_i \in \mathfrak{R}$, assim $x_1\theta_1 + \dots + x_k\theta_k$ é dito uma combinação convexa se $\sum_i \theta_i = 1$, $\theta_i \geq 0$. A combinação convexa pode ser considerada a média aritmética ponderada dos pontos x_i .

Definição 2.5. (*Invólucro Convexo*) Seja $\mathcal{F} \subset \mathfrak{R}^n$, a interseção de todos os conjuntos convexos nos \mathfrak{R}^n contendo \mathcal{F} é chamado de invólucro convexo e representado por $Co(\mathcal{F})$. Em outras

palavras, o invólucro convexo é o menor conjunto convexo que contém o conjunto \mathcal{F} , conforme figura 3.

Figura 3 – Invólucro convexo de um conjunto.



Fonte: Autor 2016.

Lema 2.1. Seja $u, u^1, \dots, u^{\mathcal{I}} \in \mathbb{R}^m$ e $v, v^1, \dots, v^{\mathcal{J}} \in \mathbb{R}^m$. Se $u \in Co\{u^i : i \in [1, \mathcal{I}]\}$ e $v \in Co\{v^i : i \in [1, \mathcal{J}]\}$, tem-se que

$$\begin{bmatrix} u \\ v \end{bmatrix} \in Co \left\{ \begin{bmatrix} u^i \\ v^j \end{bmatrix} : i \in [1, \mathcal{I}], j \in [1, \mathcal{J}] \right\}$$

■

A prova do Lema 2.1 pode ser encontrada em (HU; LIN, 2001) e (HU; LIN; CHEN, 2002).

De acordo com Hindi (2004), além de ser conveniente computacionalmente, um elipsóide é bastante usado para aproximar outros conjuntos convexos, desta maneira tem-se a seguinte definição:

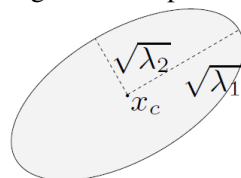
Definição 2.6. Um elipsóide pode ser definido por:

$$\mathcal{E}(A, 1) = \{x \in \mathbb{R}^n : (x - x_c)^T A^{-1} (x - x_c) \leq 1\}$$

onde $A = A^T > 0$ e o centro $x_c \in \mathbb{R}^n$.

Considerando que λ_i são os autovalores de A , um elipsóide é ilustrado na figura 4.

Figura 4 – Elipsóide.



Fonte: (HINDI, 2004).

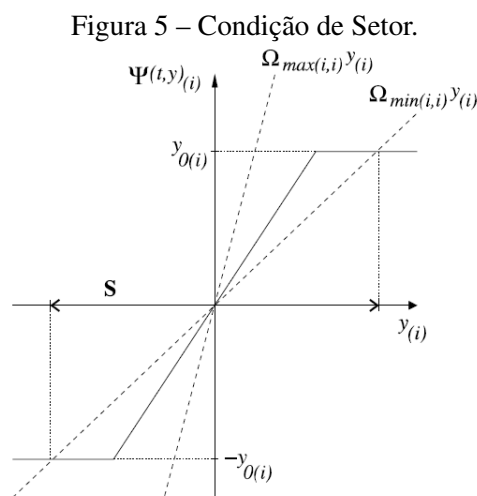
Neste trabalho é utilizado dois modelos para representar a saturação, assim a seguinte definição é enunciada (TARBOURIECH et al., 2011):

Definição 2.7. (Condição de Setor) Uma não linearidade $\psi : \mathbb{R}^m \mapsto \mathbb{R}^m$ é dito satisfazer uma condição de setor se

$$(\psi(t, y) - \Omega_{min}y)^T (\psi(t, y) - \Omega_{max}y) \leq 0, \forall t \geq 0, \forall y \in \mathcal{S} \subset \mathbb{R}^m$$

para matrizes reais Ω_{max} e Ω_{min} , onde $\Omega = \Omega_{max} - \Omega_{min}$ é simétrico positivo definido e a origem está contida em \mathcal{S} .

Quando a condição acima é satisfeita é dito que $\psi(t, y)$ pertence a condição de setor $(\Omega_{min}, \Omega_{max})$, conforme ilustrado na figura 5.

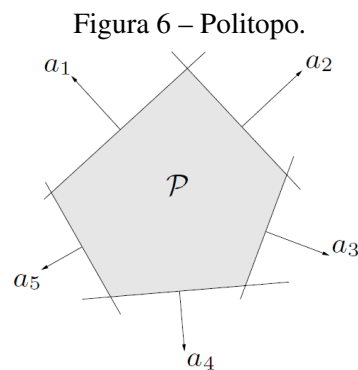


Fonte: (TARBOURIECH et al., 2011).

Definição 2.8. Um poliedro é definido pela interseção de um número finito de semi-espacos e pode ser representado matematicamente por:

$$\mathcal{P} = \{x \in \mathbb{R}^n : a_i^T x \leq b_i, i = 1, \dots, k\}$$

Um poliedro fechado é chamado de politopo. Na figura 6, é apresentado um politopo.



Fonte: (HINDI, 2004).

Lema 2.2. (Complemento de Schur) Supondo que as matrizes Q e R são simétricas, as seguintes condições são equivalentes:

$$R > 0, Q - S^T R^{-1} S > 0 \Leftrightarrow \begin{bmatrix} Q & S^T \\ S & R \end{bmatrix} > 0$$

■

A prova do Lema 2.2 pode ser encontrada em (BOYD et al., 1994) e (HAI-HUA; GUANGREN, 2013). Conforme Paim (2003), o Complemento de Schur serve para transformar algumas desigualdades não lineares em LMIs.

Lema 2.3. (Lema de Finsler) Considere-se um vetor $v \in \mathfrak{R}^n$, uma matriz simétrica $\mathcal{P} \in \mathfrak{R}^{n \times n}$, uma matriz $\mathcal{B} \in \mathfrak{R}^{m \times n}$, tal que $\text{rank}(\mathcal{B}) < n$ e \mathcal{B}_0 é uma base para o espaço nulo de \mathcal{B} . As seguintes condições são equivalentes

1. $v^T \mathcal{P} v < 0, \forall v \in \mathfrak{R}^n : \mathcal{B} v = 0, v \neq 0$;
2. $\mathcal{B}_0^T \mathcal{P} \mathcal{B}_0 < 0$;
3. $\exists \mu \in \mathfrak{R} : \mathcal{P} - \mu \mathcal{B}^T \mathcal{B} < 0$;
4. $\exists \mathbf{F} \in \mathfrak{R}^{n \times m} : \mathcal{P} + \text{sym}\{\mathbf{F} \mathcal{B}\}$.

■

A prova do Lema 2.3 pode ser encontrada em (OLIVEIRA; SKELTON, 2001). O Lema de Finsler pode ser utilizado para eliminação de variáveis segundo (BOYD et al., 1994), e recentemente tem sido bastante utilizado para reescrever condições com graus de liberdade adicionais via o uso de multiplicadores (PAIM, 2003).

Lema 2.4. (\mathcal{S} -procedure) Considere $T_0, \dots, T_p \in \mathfrak{R}^{n \times n}$ sendo matrizes simétricas e se existir escalares positivos τ_i de tal maneira que

$$T_0 - \sum_{i=1}^p \tau_i T_i > 0$$

então

$$x^T T_0 x > 0 \forall x \neq 0, \text{ de tal maneira que } x^T T_i x \geq 0, i = 1, \dots, p$$

■

Mais detalhes sobre o Lema 2.4 podem ser encontradas em (BOYD et al., 1994) e (JOHANSSON, 2003). De acordo com (BENDER, 2006), o \mathcal{S} -procedure é uma técnica de relaxação, que restringe a uma região específica do espaço de estados a validação das condições, como por exemplo, somente onde o modelo da não-linearidade é válido e não em todo espaço de estados.

Lema 2.5. (*Lema de Projeção*) Seja uma matriz simétrica $\Psi \in \mathbb{R}^{n \times n}$ e duas matrizes $P, Q \in \mathbb{R}^{m \times n}$. Assim, existe uma matriz $\Theta \in \mathbb{R}^{m \times m}$ tal maneira que

$$\Psi + P^T \Theta^T Q + Q^T \Theta P < 0$$

se e somente se

$$\mathcal{N}_P^T \Psi \mathcal{N}_Q < 0 \quad \mathcal{N}_Q^T \Psi \mathcal{N}_P < 0$$

onde $\mathcal{N}_P, \mathcal{N}_Q$ são quaisquer matrizes cujas colunas formam bases dos espaços nulos de P, Q . ■

O Lema 2.5 foi primeramente proposto por (GAHINET; APKARIAN, 1994) e sua prova também pode ser encontrada em (BENDER, 2010) e (HAI-HUA; GUANGREN, 2013). Conforme (HAI-HUA; GUANGREN, 2013), o Lema de Projeção é largamente utilizado para eliminação de variáveis em uma BMI com a finalidade de obter uma LMI.

2.3 Sistemas com Atraso

Equações diferenciais são geralmente empregadas para a modelagem de sistemas dinâmicos. Considerando a representação em espaço de estados de um sistema, (2.1), pode-se verificar que a taxa de variação dos estados depende única e exclusivamente dos estados naquele instante específico. Em alguns sistemas essa dependência não está somente relacionada ao estado atual, mas também aos estados passados. Esse efeito é conhecido na literatura como atraso e tem sido o foco de diversas pesquisas acadêmicas nos últimos anos. Conforme Mahmoud (2010), existem três principais fontes para a ocorrência do atraso:

1. Natureza do processo: muito comum em processos químicos, em que o atraso ocorre pelo próprio funcionamento interno do sistema, tais como, em reatores químicos e motores diesel;
2. Atraso de transporte: pode-se citar um sistema de aquecimento baseado na transferência de calor por convecção em que existe um determinado atraso devido ao deslocamento do ar quente;
3. Atraso de comunicação: dois tipos de atrasos são encontrados em sistemas de comunicação, um deles devido ao tempo de propagação, i.e. o tempo que o sinal leva para se propagar no meio de um ponto a outro. O outro acontece quando existem limitações no acesso ao barramento de comunicação, especialmente quando diversos dispositivos compartilham o mesmo barramento.

Com base no acima exposto, é possível identificar a existência do atraso em diversos exemplos práticos, como pode ser visto em (FRIDMAN, 2014), (GU; KHARITONOV; CHEN, 2003), onde exemplos nas áreas de biologia, química, física, mecânica e economica são expostos.

Com a finalidade de aplicar leis de controle a sistemas com atraso, duas principais abordagens têm sido utilizadas pela literatura científica. Uma delas, chamada de independente do atraso, não leva em consideração o tamanho do atraso. Logo, apesar de ser mais fácil algebricamente a obtenção de condições para análise e síntese, os resultados via estas abordagens tendem a ser mais conservativos. A razão para isso, se deve ao fato de que os resultados obtidos por uma abordagem independente do atraso tem que ser verificada para qualquer atraso e isso restringe muito as possíveis soluções. Segundo Mahmoud (2010), existem diversas razões para que os resultados sejam conservadores:

- a premissa de que a magnitude do atraso é pequena nem sempre é verdadeira e, portanto, para esses casos as condições não serão satisfeitas;
- em muitos casos, o atraso é fixo, assim a verificação de estabilidade via abordagem independente do atraso, impõe muitas condições desnecessárias para este sistema em particular;
- não se aplica a sistemas instáveis em malha aberta.

A outra abordagem largamente utilizada, conhecida como dependente do atraso, leva em consideração o tamanho do atraso e é empregada em dois tipos de problemas (MAHMOUD, 2010): para um dado sistema em que o atraso é conhecido, verificar a estabilidade do mesmo; para um dado sistema, verificar qual é o máximo atraso o qual a estabilidade pode ser garantida.

2.3.1 Estabilidade de Sistemas Discretos com Atraso

Condições suficientes de estabilidade, tanto para o caso independente quanto para o dependente ao atraso, podem ser obtidas via funções de Lyapunov-Razumikhin ou funcionais de Lyapunov-Krasovskii, podendo ser encontrado em (RICHARD, 2003), (GU; KHARITONOV; CHEN, 2003), (MICHIELS; NICULESCU, 2007).

No entanto, antes do trabalho de Hetel, Daafouz e Iung (2008) soluções para as seguintes questões não estavam claramente respondidas na literatura:

- Qual é o funcional de Lyapunov-Krasovskii que apresenta o menor conservadorismo possível?
- É possível obter condições suficientes e necessárias na forma de LMI para a existência de um funcional de Lyapunov-Krasovskii?
- É possível obter condições na forma de LMI sem limitar seus termos?

O trabalho de Hetel, Daafouz e Iung (2008) propõem uma extensão do funcional de Lyapunov-krasovskii clássico, solucionando assim as questões expostas acima. Esses resultados são apresentados na sequência e são a base para o desenvolvimento principal deste trabalho. Uma vez

que esse trabalho considera sistemas de tempo discreto, pode-se começar apresentando o seguinte sistema discreto no tempo com atraso

$$x[k+1] = Ax[k] + A_d x[k - \tau[k]], \quad k \in \mathbb{Z} \quad (2.5)$$

onde $x[k] \in \mathbb{R}^n$ é o vetor dos estados. As matrizes A , A_d são reais constantes e de dimensões apropriadas. O atraso variante no tempo $\tau[k]$ é um número inteiro e positivo dentro do seguinte intervalo $\underline{\tau} \leq \tau[k] \leq \bar{\tau}$, onde $\underline{\tau}$ e $\bar{\tau}$ são valores inteiros positivos e conhecidos. As condições iniciais deste sistema podem ser descritas como

$$x[k_o] = \phi[k_o], \quad \forall k_o \in [-\bar{\tau}, 0]$$

Significando que mesmo para o valor de atraso máximo, antes do instante inicial os estados do sistema são conhecidos e limitados em amplitude. Essa suposição é razoável especialmente porque sistemas digitais são geralmente implementados em um espaço de memória. Além disso, é aceitável supor que o atraso é um múltiplo inteiro do período de amostragem, uma vez que a informação está apenas disponível em instante específicos, e não de maneira contínua. Então, qualquer evento que acontece entre duas amostras consecutivas irá se manifestar nas amostras seguintes. Assim, para o sistema (2.5), o valor dos estados na próxima amostra (i.e. $x[k+1]$), será uma função da amostra atual (i.e. $Ax[k]$) e, como estamos levando em consideração o atraso neste sistema, de algum estado anterior (i.e. $A_d x[k - \tau[k]]$), que vai depender do valor do atraso naquele instante. Assumindo que a variação do atraso no tempo significa quantos estados anteriores podem afetar o próximo. A seguinte matriz, que representa a dinâmica do sistema de um instante $(k - \bar{\tau})$ até o presente (k) , pode ser definida (HETEL; DAAFOUZ; IUNG, 2008):

$$\Lambda_{\tau[k]} = \begin{bmatrix} A & \Sigma_1 & \Sigma_2 & \cdots & \Sigma_{\bar{\tau}} \\ I & 0 & \cdots & 0 & 0 \\ 0 & I & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & I & 0 \end{bmatrix}, \quad \Sigma_l = \begin{cases} A_d, & \tau[k] = l \\ 0, & \text{caso contrario} \end{cases}, \quad l \in 1, \dots, \bar{\tau}$$

Pode-se aumentar o vetor dos estados para que contenha não somente o estado atual (i.e. $x[k]$) ou o atrasado (i.e. $x[k - \tau[k]]$), mas todos os estados passados que afetam o próximo, uma vez que $\bar{\tau}$ é conhecido *a priori*.

$$z[k] = \begin{bmatrix} x[k]^T & x[k-1]^T & \cdots & x[k-\bar{\tau}]^T \end{bmatrix}^T$$

onde $z[k] \in \mathbb{R}^v$, $v = (\bar{\tau} + 1) \times n$.

Comentário 1. De acordo com Fridman (2014), esta abordagem sofre com a dimensionalidade

quando o atraso é consideravelmente grande, tendo em vista que a dimensão do vetor $z[k]$ é diretamente relacionada ao atraso máximo $\bar{\tau}$.

Desta maneira é possível reescrever o sistema com atraso (2.5) no seguinte sistema chaveado

$$z[k+1] = \Lambda_{\tau[k]} z[k] \quad (2.6)$$

que possui uma matriz $\Lambda_{\tau[k]}$ diferente para cada atraso $\tau[k]$. Agora, enuncia-se o seguinte teorema:

Teorema 2.3. *O sistema (2.5) é assintoticamente estável se existirem matrizes simétricas e positivas definidas $\Pi_{\tau[k]} \in \mathfrak{R}^{v \times v}$, $\tau[k] \in \{\underline{\tau}, \dots, \bar{\tau}\}$, de tal maneira que*

$$\begin{bmatrix} -\Pi_{\tau[k]} & \star \\ \Pi_{\tau[k+1]} \Lambda_{\tau[k]} & -\Pi_{\tau[k+1]} \end{bmatrix} < 0 \quad (2.7)$$

■

Prova. O seguinte funcional de Lyapunov-Krasovskii dependente do atraso é apresentado em (HETEL; DAAFOUZ; IUNG, 2008) como sendo a forma mais genérica que pode ser obtida usando a soma de todas as combinações possíveis de termos em formas quadráticas.

$$V[k] = \sum_{i=0}^{\bar{\tau}} \sum_{j=0}^{\bar{\tau}} x[k-i]^T P_{\tau[k]}^{(i,j)} x[k-j] \quad (2.8)$$

Pode-se perceber que no funcional cada termo quadrático $x[k-i]^T P_{\tau[k]}^{(i,j)} x[k-j]$ possui uma matriz de Lyapunov $P_{\tau[k]}^{(i,j)}$ diferente que varia de acordo com o atraso $\tau[k]$.

Considerando o funcional (2.8) para o sistema (2.6) chega-se a seguinte função candidata de Lyapunov

$$V[k] = z[k]^T \Pi_{\tau[k]} z[k] > 0$$

$$\Pi_{\tau[k]} = \begin{bmatrix} P_{\tau[k]}^{(0,0)} & P_{\tau[k]}^{(0,1)} & \dots & P_{\tau[k]}^{(0,\bar{\tau})} \\ P_{\tau[k]}^{(1,0)} & P_{\tau[k]}^{(1,1)} & \dots & P_{\tau[k]}^{(1,\bar{\tau})} \\ \vdots & \ddots & \ddots & \vdots \\ P_{\tau[k]}^{(\bar{\tau},0)} & P_{\tau[k]}^{(\bar{\tau},1)} & \dots & P_{\tau[k]}^{(\bar{\tau},\bar{\tau})} \end{bmatrix} > 0, \begin{cases} \forall \tau[k] \in \{1, \dots, \bar{\tau}\} \\ P_{\tau[k]}^{(i,j)} = (P_{\tau[k]}^{(j,i)})^T \end{cases}$$

É importante salientar que não é necessário que todas as matrizes $P_{\tau[k]}^{(i,j)}$ sejam positivas definidas, apenas o bloco $\Pi_{\tau[k]}$ deve ser positivo definido.

Conforme apresentado na seção (2.1.1), para garantir estabilidade assintótica de um sistema de tempo discreto, deseja-se encontrar um funcional positivo definido tal que sua variação seja

negativa, portanto tem-se

$$\begin{aligned}
\Delta V &= V[k+1] - V[k] < 0 \\
&= z[k+1]^T \Pi_{\tau[k+1]} z[k+1] - z[k]^T \Pi_{\tau[k]} z[k] < 0 \\
&= z[k]^T \Lambda_{\tau[k]}^T \Pi_{\tau[k+1]} \Lambda_{\tau[k]} z[k] - z[k]^T \Pi_{\tau[k]} z[k] < 0 \\
&= z[k]^T (\Lambda_{\tau[k]}^T \Pi_{\tau[k+1]} \Lambda_{\tau[k]} - \Pi_{\tau[k]}) z[k] < 0
\end{aligned}$$

Desta maneira, pode-se escrever a seguinte desigualdade

$$\Lambda_{\tau[k]}^T \Pi_{\tau[k+1]} \Lambda_{\tau[k]} - \Pi_{\tau[k]} < 0 \quad (2.9)$$

Por intermédio do Complemento de Schur, Lema 2.2, (2.9) é equivalente a:

$$\begin{bmatrix} -\Pi_{\tau[k]} & * \\ \Lambda_{\tau[k]} & -\Pi_{\tau[k+1]}^{-1} \end{bmatrix} < 0 \quad (2.10)$$

Inequação (2.10) não é uma LMI devido ao termo $\Pi_{\tau[k+1]}^{-1}$. De modo a contornar esta não linearidade, aplicamos uma transformação linear, pré e pós multiplicando (2.10) pelo bloco diagonal $blockdiag = \{I_v, \Pi_{\tau[k+1]}\}$ e sua transposta respectivamente. Obtendo assim a LMI (2.7) do Teorema 2.3. \square

Comentário 2. *O Teorema 2.3 possui condições necessárias e suficientes para estabilidade do sistema (2.5). Isso significa que quando não é possível encontrar uma solução para a desigualdade (2.7), não existe um funcional na literatura que encontre, uma vez que o funcional proposto, conforme Hetel, Daafouz e Jung (2008) é o mais genérico dentre todos os que envolvem o somatório de termos quadráticos.*

Comentário 3. *Conforme mencionado anteriormente, obter solução numérica para o funcional levando em consideração o maior atraso possível é uma das principais causas do conservadorismo dos outros funcionais propostos na literatura. Além disso, através do Teorema 2.3 não é necessário considerar todos os atrasos possíveis entre o mínimo e o máximo, i.e. $\underline{\tau} \leq \tau[k] \leq \bar{\tau}[k]$, pode-se por exemplo analisar a estabilidade do sistema (2.5) para $\tau[k] \in \{1, 6, 11, 14\}$, aspecto bastante útil quando se trata de sistemas de comunicação de dados.*

2.3.2 Estabilidade de Sistemas Discretos com Atraso via Finsler

Nesta seção é apresentado uma extensão do Teorema 2.3, proposto por Sun e Chen (2012), que faz uso do Lema 2.3 e é exposto em detalhes na sequência.

Corolário 2.4. *O sistemas (2.5) é assintoticamente estável se existirem matrizes simétricas e positivas definidas $\Pi_{\tau[k]} \in \mathfrak{R}^{v \times v}$ e matrizes $G_{\tau[k]} \in \mathfrak{R}^{v \times v}$, $F_{\tau[k]} \in \mathfrak{R}^{v \times v}$, $\tau[k] \in \{\underline{\tau}, \dots, \bar{\tau}\}$ de*

tal maneira que

$$\begin{bmatrix} -\Pi_{\tau[k]} + \text{sym}\{F_{\tau[k]}\Lambda_{\tau[k]}\} & \star \\ G_{\tau[k]}\Lambda_{\tau[k]} - F_{\tau[k]} & \Pi_{\tau[k+1]} - \text{sym}\{G_{\tau[k]}\} \end{bmatrix} < 0 \quad (2.11)$$

■

Prova. Partindo da prova do Teorema 2.3, tem-se

$$\begin{aligned} \Delta V &= V[k+1] - V[k] < 0 \\ &= z[k+1]^T \Pi_{\tau[k+1]} z[k+1] - z[k]^T \Pi_{\tau[k]} z[k] < 0 \end{aligned}$$

Por meio do vetor $\eta[k]$ e da matrix M

$$\eta[k] = \begin{bmatrix} z[k]^T & z[k+1]^T \end{bmatrix}^T, \quad M = \begin{bmatrix} -\Pi_{\tau[k]} & 0 \\ 0 & \Pi_{\tau[k+1]} \end{bmatrix}$$

a variação ΔV pode ser reescrita em forma matricial

$$\eta[k]^T M \eta[k] = \begin{bmatrix} z[k]^T & z[k+1]^T \end{bmatrix}^T \begin{bmatrix} -\Pi_{\tau[k]} & 0 \\ 0 & \Pi_{\tau[k+1]} \end{bmatrix} \begin{bmatrix} z[k] \\ z[k+1] \end{bmatrix}$$

Deseja-se aplicar o Lema 2.3, o que não garante $M < 0$ para qualquer $\eta[k]$, mas somente para as trajetórias do sistema em malha fechada (2.6). Condições resultantes dessa metodologia são potencialmente menos conservativas. Desta forma, definindo a matrix \mathcal{B} e aplicando o Lema 2.3

$$\mathcal{B} = \begin{bmatrix} \Lambda_{\tau[k]} & -I \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} F_{\tau[k]} \\ G_{\tau[k]} \end{bmatrix}, \quad \mathbf{F}\mathcal{B} = \begin{bmatrix} F_{\tau[k]}\Lambda_{\tau[k]} & -F_{\tau[k]} \\ G_{\tau[k]}\Lambda_{\tau[k]} & -G_{\tau[k]} \end{bmatrix}$$

obtemos a LMI do Corolário 2.4. □

2.3.3 Exemplos Numéricos

Nessa seção, por meio de um dado sistema discreto sujeito ao atraso, deseja-se verificar para qual o maior intervalo do valor do atraso o sistema apresenta estabilidade assintótica.

Exemplo 2.1. Considere o sistema (2.5) dado pelas seguintes matrizes vide (SUN; CHEN, 2012):

$$A = \begin{bmatrix} 0.4 & -0.8 \\ 0.5 & 1.0 \end{bmatrix}, \quad A_d = \begin{bmatrix} 0.1 & 0.2 \\ 0.2 & 0.4 \end{bmatrix}$$

Aplicando o Teorema 2.3, pode-se concluir que o sistema é assintoticamente estável para atrasos dentro do intervalo $1 \leq \tau[k] \leq 4$. Para o Corolário 2.4 igualmente $1 \leq \tau[k] \leq 4$. Para obtenção desses resultados utilizou-se o software MATLAB juntamente com o parser YALMIP (LÖFBERG, 2004) e o solver SDPT3 (TUTUNCU; TOH; TODD, 1999).



Exemplo 2.2. Considere o sistema (2.5) dado pelas seguintes matrizes vide (HETEL; DAAFOUZ; IUNG, 2008):

$$A = \begin{bmatrix} 0.8 & 0 \\ 0 & 0.97 \end{bmatrix}, \quad A_d = \begin{bmatrix} -0.1 & 0 \\ -0.1 & -0.1 \end{bmatrix}$$

Por intermédio do Teorema 2.3 e do Corolário 2.4, o sistema é assintoticamente estável para atrasos dentro do intervalo, $1 \leq \tau[k] \leq 14$ e $1 \leq \tau[k] \leq 10$, respectivamente. Para obtenção desses resultados assim como no exemplo anterior foi utilizado o software MATLAB juntamente com o parser YALMIP (LÖFBERG, 2004) e o solver SDPT3 (TUTUNCU; TOH; TODD, 1999).

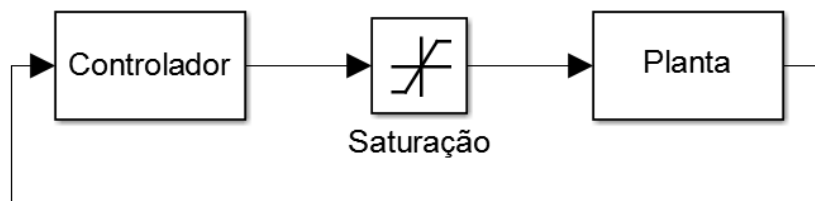


Comentário 4. O resultado obtido neste exemplo numérico difere-se do apresentado em Hetel, Daafouz e Iung (2008), em que foi obtido solução para atrasos dentro do intervalo $3 \leq \tau[k] \leq 13$. Atribui-se essa discrepância aos pacotes computacionais utilizados para resolução numérica.

2.4 Sistemas Sujeitos a Saturação

Em todo sistema físico existem limitações relacionadas com as quantidades máximas e mínimas de energia que é possível aplicar a esse sistema. Esses limites podem ser restrições físicas do atuador ou limitações impostas por questões de segurança. Esse fenômeno é conhecido na literatura como saturação e tem sido um tema de relevância acadêmicas nas últimas décadas. Conforme Tarbouriech et al. (2011) e Hu e Lin (2001), a saturação é uma das principais fontes de degradação de desempenho de um sistema em malha fechada, causando, por exemplo, um aumento no sobre-sinal bem como no tempo de acomodação do sistema. Além disso, pode até levar um sistema, que é estável em malha aberta, a instabilidade quando em malha fechada. A saturação pode ser vista no diagrama de blocos da figura 7.

Figura 7 – Diagrama de blocos de um sistema sujeito a saturação.



Fonte: Autor 2016.

Considere o seguinte sistema discreto no tempo

$$x[k + 1] = Ax[k] + A_d x[k - \tau[k]] + Bu[k], \quad k \in \mathbb{Z} \quad (2.12)$$

onde o vetor $x[k] \in \mathfrak{R}^n$ e $u[k] \in \mathfrak{R}^m$ são respectivamente o vetor de estados e entradas. As matrizes A , A_d e B são reais constantes e de dimensões apropriadas. O atraso variante no tempo $\tau[k] \in \{\underline{\tau}, \dots, \bar{\tau}\}$, onde $\underline{\tau}$ e $\bar{\tau}$ são valores inteiros positivos e conhecidos. As condições iniciais deste sistema podem ser descritas como

$$x[k] = \phi[k], \forall k \in [-\bar{\tau}, 0]$$

A entrada deste sistema é limitada tal que

$$-u_{o(i)} \leq u_{(i)} \leq u_{o(i)}, \quad u_{o(i)} > 0, \quad i = 1, \dots, m \quad (2.13)$$

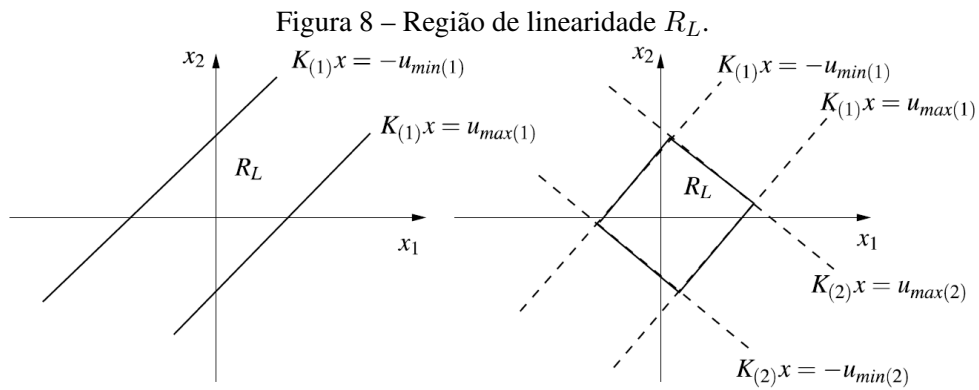
A seguinte definição é apresentada vide (TARBOURIECH et al., 2011):

Definição 2.9. (*Região de Linearidade*) A região de linearidade do sistema (2.12), denota-se R_L , é definida como o conjunto de todos os estados $x[k] \in \mathfrak{R}^n$ tal que $\text{sat}(K_{\tau[k]}x[k]) = K_{\tau[k]}x[k]$.

O seguinte conjunto poliédrico define a região de linearidade:

$$R_L = \{x[k] \in \mathfrak{R}^n : |K_{\tau[k]}x[k]| \leq u_o\}$$

esse conjunto é limitado quando $\text{rank}(K_{\tau[k]}) = n$ e aberto quando $\text{rank}(K_{\tau[k]}) < n$, conforme pode ser visto na figura 8. É possível verificar que o espaço de estados é dividido em



Fonte: (TARBOURIECH et al., 2011).

duas regiões. Uma delas é a R_L , em que a saturação pode ser desconsiderada e, portanto, o sistema se comporta de acordo com seu modelo linear. A outra é onde a saturação ocorre e segundo Tarbouriech et al. (2011) o sistema apresenta comportamento não linear. Desta maneira, a convergência das trajetórias do sistema para a origem dependem das condições iniciais. Formalmente, tem-se a seguinte definição (TARBOURIECH et al., 2011):

Definição 2.10. (*Região de Atração*) A região de atração da origem para o sistema (2.12),

denota-se R_A , é definida como o conjunto de todos os pontos $x[k] \in \mathbb{R}^n$ do espaço de estados no qual sendo condições iniciais as trajetórias convergem assintoticamente para origem.

A saturação claramente não é uma função linear, como pôde ser visto na figura 1. Portanto, esta não linearidade deve ser modelada de tal maneira que possa-se incluir na formulação do problema. Na literatura existem diferentes maneiras de modelar a saturação como pode ser visto em (TARBOURIECH et al., 2011). Neste trabalho a representação da saturação se dará pela modelagem politópica bem como pela não linearidade do tipo zona morta, que são apresentadas na sequência.

2.4.1 Modelagem da Saturação via Representação Politópica

Segundo Tarbouriech et al. (2011), a ideia básica da modelagem politópica consiste em usar um vetor como variável auxiliar para obter a saída da função saturação como uma combinação convexa entre o sinal de controle e a variável auxiliar. Considere um conjunto \mathcal{D} com matrizes diagonais nos quais os elementos são 0 ou 1. Por exemplo, para $m = 2$:

$$\mathcal{D} = \left\{ \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right\}$$

É possível observar que o conjunto \mathcal{D} possui 2^m elementos, portanto, defini-se as matrizes do conjunto \mathcal{D} como Γ_j^+ , $j \in [1, 2^m]$, logo:

$$\mathcal{D} = \{\Gamma_j^+ : j \in [1, 2^m]\}$$

Sendo que $\Gamma_j^- = I_m - \Gamma_j^+$ também é um elemento do conjunto \mathcal{D} . Assim, o seguinte Lema pode ser enunciado vide (HU; LIN, 2001) e (HU; LIN; CHEN, 2002):

Lema 2.6. *Seja dois vetores $v \in \mathbb{R}^m$ and $h \in \mathbb{R}^m$. Se $-u_{o(i)} \leq h_{(i)} \leq u_{o(i)}$, para $i = 1, \dots, m$, logo*

$$\text{sat}(v) \in \text{Co}\{\Gamma_j^+ v + \Gamma_j^- h, j = 1, \dots, 2^m\}$$

■

Prova. Uma vez que $|h_{(i)}| \leq u_{o(i)}$ tem-se que

$$\text{sat}(v_j) \in \text{Co}\{u_j, h_j\}, \forall j \in [1, m]$$

aplicando o Lema 2.1 tem-se

$$\text{sat}(v_1) \in \text{Co}\{v_1, h_1\}$$

$$\text{sat} \left(\begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \right) \in \text{Co} \left\{ \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}, \begin{bmatrix} v_1 \\ h_2 \end{bmatrix}, \begin{bmatrix} h_1 \\ v_2 \end{bmatrix}, \begin{bmatrix} h_1 \\ h_2 \end{bmatrix} \right\}$$

$$\vdots$$

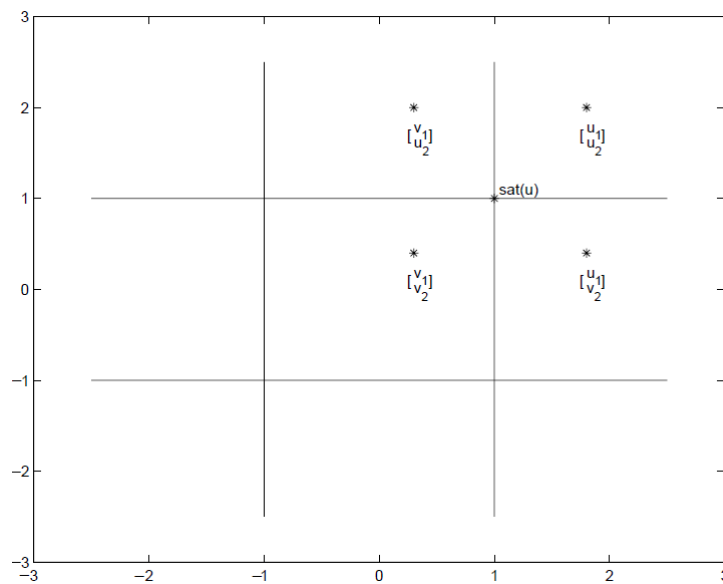
assim pode-se concluir que:

$$\text{sat}(v) \in \text{Co}\{\Gamma_j^+ v + \Gamma_j^- h, \forall j \in [1, 2^m]\}.$$

□

A ilustração gráfica desse Lema pode ser vista na figura 9. É possível observar que cada vértice do polítopo pertence a uma região diferente no espaço de estados. Neste caso, sendo dois estados temos quatro casos: nenhum estado saturado, apenas o primeiro, apenas o segundo e ambos saturados.

Figura 9 – Ilustração gráfica do Lema 2.6.



Fonte: (HU; LIN; CHEN, 2002).

2.4.2 Modelagem da Saturação via Não Linearidade de Setor

Conforme Tarbouriech et al. (2011), o sistema (2.12) pode ser tratado como um sistema linear e estável onde a realimentação possui uma não linearidade de setor. Esse caso é conhecido como problema de Lure. Por intermédio da Função Zona-morta $\psi(\cdot)$ o seguinte lema pode ser enunciado (SILVA JR.; TARBOURIECH, 2005):

Lema 2.7. (Condição de Setor Generalizada) Considerando a matriz $G \in \mathfrak{R}^{m \times n}$ e o conjunto poliedral $\mathcal{S}(u_o)$:

$$\mathcal{S}(u_o) = \{x[k] \in \mathfrak{R}^n : |(K_{\tau[k](i)} - G_{(i)})x[k]| \leq u_{o(i)}, i = 1, \dots, m\}$$

se $x[k] \in \mathcal{S}(u_o)$ segue que:

$$\psi(K_{\tau[k]}x[k])^T T (\psi(K_{\tau[k]}x[k]) - Gx[k]) \leq 0$$

é definida para qualquer matriz diagonal positiva definida $T \in \mathfrak{R}^{m \times m}$. ■

Prova. Considerando $\psi(K_{\tau[k]}x[k]) = K_{\tau[k]}x[k] - \text{sat}(K_{\tau[k]}x[k])$, têm-se os três casos abaixo:

1. $-u_o \leq K_{\tau[k]}x[k] \leq u_o$. Neste caso $\psi(K_{\tau[k]}x[k]) = 0$, então

$$\psi(K_{\tau[k]}x[k])^T T (\psi(K_{\tau[k]}x[k]) - Gx[k]) = 0$$

2. $K_{\tau[k]}x[k] > u_o$. Neste caso $\psi(K_{\tau[k]}x[k]) = K_{\tau[k]}x[k] - u_o$, se $x[k] \in \mathcal{S}(u_o)$ então:

$$K_{\tau[k]}x[k] - Gx[k] \leq u_o$$

Por isso, pode-se afirmar que

$$\psi(K_{\tau[k]}x[k]) - Gx[k] = K_{\tau[k]}x[k] - Gx[k] - u_o \leq 0$$

e como $\psi(K_{\tau[k]}x[k]) > 0$, segue que:

$$\psi(K_{\tau[k]}x[k])^T T (\psi(K_{\tau[k]}x[k]) - Gx[k]) \leq 0, \forall T > 0$$

3. $K_{\tau[k]}x[k] < -u_o$. Neste caso $\psi(K_{\tau[k]}x[k]) = K_{\tau[k]}x[k] + u_o$ se $x[k] \in \mathcal{S}(u_o)$ então:

$$K_{\tau[k]}x[k] - Gx[k] \geq -u_o$$

Por isso, pode-se afirmar que

$$\psi(K_{\tau[k]}x[k]) - Gx[k] = K_{\tau[k]}x[k] - Gx[k] + u_o \leq 0$$

e como $\psi(K_{\tau[k]}x[k]) > 0$, segue que:

$$\psi(K_{\tau[k]}x[k])^T T (\psi(K_{\tau[k]}x[k]) - Gx[k]) \leq 0, \forall T > 0$$

□

2.4.3 Modelagem por Regiões de Saturação

A saturação pode ser representada por diferentes modelos, uma abordagem apresentada por Silva Jr. e Tarbouriech (1999), divide o espaço de estados em regiões considerando todas as possíveis possibilidades. Por exemplo, em um sistema com apenas uma entrada, i.e. $m = 1$, existem 3^m regiões, a entrada é menor que o limite mínimo, i.e. $u \leq -u_o$, a entrada é maior que o limite superior, i.e. $u \geq u_o$ e a entrada não está saturada, i.e. $-u_o \leq u \leq u_o$. Segundo TARBOURIECH et al. (2011), essas regiões são chamadas de regiões de saturação e são definidas por polígonos abertos e fechados. Para apresentar essa representação, primeiramente considera-se um sistema com saturação na entrada

$$\dot{x}(t) = Ax(t) + Bsat(u(t)) \quad (2.14)$$

e defini-se um vetor

$$\xi \in \mathfrak{R}^m, \xi_{(i)} = \begin{cases} 1, & \text{se } u(t) \geq u_{o(i)} \\ 0, & \text{se } u_{o(i)} < u(t) < u_{o(i)} \\ -1, & \text{se } u(t) \leq -u_{o(i)} \end{cases}$$

e uma função

$$\delta : \mathfrak{R}^m \mapsto \mathfrak{R}^m, \delta_{(i)}(\xi_j) = \begin{cases} u_{o(i)}, & \text{se } \xi_{j(i)} = 1 \\ 0, & \text{se } \xi_{j(i)} = 0 \\ -u_{o(i)}, & \text{se } \xi_{j(i)} = -1 \end{cases}$$

O seguinte sistema pode ser modelado como um sistema afim em cada região de saturação.

$$\begin{aligned} \dot{x}(t) &= (A + Bdiag(1_m - |\xi_j|)K)x(t) + B\delta(\xi_j); \quad i = 1, \dots, m; \quad j = 1, \dots, 3^m \\ \dot{x}(t) &= (A + B\Gamma_j K)x(t) + B\delta(\xi_j), \quad \Gamma_j = diag(1_m - |\xi_j|) \end{aligned}$$

3 ANÁLISE DE SISTEMAS SATURANTES VIA REGIÕES DE SATURAÇÃO

Este capítulo apresenta condições para verificar a estabilidade de sistemas com atuadores saturantes representando a saturação pela modelagem por regiões de saturação. Inicialmente o problema é exposto, após os teoremas obtidos juntamente com exemplos numéricos da utilização dos mesmos são apresentados.

3.1 Sistemas de tempo contínuo

Conforme apresentado no capítulo anterior uma das possíveis maneiras de representar a saturação é por meio da modelagem por regiões de saturação. O resultado apresentado nesta seção consideram sistemas com saturação na entrada e com realimentação de estados.

$$\dot{x}(t) = (A + B\Gamma_j K)x(t) + B\delta(\xi_j) \quad (3.1)$$

O seguinte teorema é proposto para análise da estabilidade do sistema (3.1).

Teorema 3.1. *Se existir matrizes simétricas positivas definidas $P_j \in \mathfrak{R}^{n \times n}$, matrizes $F_{1j}, F_{2j} \in \mathfrak{R}^{n \times n}$, $F_{3j} \in \mathfrak{R}^{1 \times n}$ e escalares $\lambda_{j(l)} \in \mathfrak{R}$, onde $j = 1, \dots, 3^m$ e $\Lambda = [\lambda_{j(1)} \cdots \lambda_{j(l)}]$, de tal maneira que*

$$\left[\begin{array}{cc} -F_1 - F_1^T & \star \\ -F_2 + A^T F_1^T + K^T \Gamma_j B^T F_1^T & \left(\begin{array}{c} \text{sym}\{P_j A + P_j B \Gamma_j K\} + \\ \text{sym}\{F_2 A + F_2 B \Gamma_j K\} \end{array} \right) \\ -F_3 + \delta(\xi_j)^T B^T F_1^T & \left(\begin{array}{c} \delta(\xi_j)^T B^T P_j - 0.5 \Lambda_j R_j + \\ F_3 A + F_3 B \Gamma_j K + \delta(\xi_j)^T B^T F_2^T \end{array} \right) \end{array} \left(\begin{array}{c} \Lambda_j d_j + \\ \text{sym}\{F_3 B \delta(\xi_j)\} \end{array} \right) \right] < 0 \quad (3.2)$$

$$\left[\begin{array}{cc} 1 & \frac{R_{j(i)}^T P_j}{d_{j(i)}} \\ P_j \frac{R_{j(i)}}{d_{j(i)}} & P_j \end{array} \right] > 0, \quad i = 1, \dots, m \quad (3.3)$$

$\mathcal{E}(P_j, 1)$ estima a região de atração da origem do sistema (3.1). ■

Prova. Considerando a seguinte função quadrática $V(x) = x^T P_j x \forall j = 1, \dots, 3^m$, como a função candidata de Lyapunov, deseja-se obter condições para garantir que a sua derivada seja negativa. Pode-se observar que neste caso procura-se não apenas um único elipsóide que estime

a região de atração do sistema, mas um elipsóide para cada região de saturação. Então

$$\begin{aligned}\dot{V}(x) &= \dot{x}^T P_j x + x^T P_j \dot{x} < 0 \\ &= [(A + B\Gamma_j K)x + B\delta(\xi_j)]^T P_j x + x^T P_j [(A + B\Gamma_j K)x + B\delta(\xi_j)] < 0 \\ &= x^T (A + B\Gamma_j K)^T P_j x + \delta(\xi_j)^T B^T P_j x + x^T P_j (A + B\Gamma_j K)x + x^T P_j B\delta(\xi_j) < 0\end{aligned}$$

Pode-se reescrever a inequação acima em forma matricial por

$$\begin{bmatrix} x^T & 1 \end{bmatrix}^T \begin{bmatrix} \text{sym}\{P_j A + P_j B\Gamma_j K\} & \star \\ \delta(\xi_j)^T B^T P_j & 0 \end{bmatrix} \begin{bmatrix} x \\ 1 \end{bmatrix} < 0$$

Na desigualdade acima é possível verificar um zero na diagonal principal, o que dificulta a resolução numérica dessa condição. Para tanto, utiliza-se o \mathcal{S} -procedure para verificar a validade das condições somente dentro dos polítopos que descrevem cada região de saturação.

Comentário 5. *É importante salientar que a condição obtida até aqui, por mais que difícil de verificar numericamente, é uma condição suficiente e necessária para a estabilidade do sistema (3.1). De acordo com Johansson (2003), a utilização do \mathcal{S} -procedure para verificar uma função quadrática em políedros resulta em condições apenas suficientes e não necessárias. Por isso, neste caso a utilização do \mathcal{S} -procedure torna as condições do Teorema 3.1 apenas suficientes.*

Antes de aplicarmos o \mathcal{S} -procedure deve-se definir matematicamente as regiões de saturação, então

$$\mathcal{S}(R_j, d_j) = \{x \in \mathbb{R}^n : R_{j(i)}x \leq d_{j(i)}\}$$

onde (i) representa i -ésima linha da matriz R e i -ésimo elemento do vetor d . Assim, conforme Lema 2.4, tem-se

$$\text{sym}\{x^T (P_j A + P_j B\Gamma_j K)x\} + x^T P_j B\delta(\xi_j) + \delta(\xi_j)^T B^T P_j x - \sum_{i=1}^l \lambda_{j(i)} (R_{j(i)}x - d_{j(i)}) < 0$$

Pode-se reescrever a inequação acima em forma matricial por

$$\begin{bmatrix} x^T & 1 \end{bmatrix}^T \begin{bmatrix} \text{sym}\{P_j A + P_j B\Gamma_j K\} & \star \\ \delta(\xi_j)^T B^T P_j - 0.5 \sum_{i=1}^l \lambda_{j(i)} R_{j(i)} & \sum_{i=1}^l \lambda_{j(i)} d_{j(i)} \end{bmatrix} \begin{bmatrix} x \\ 1 \end{bmatrix} < 0 \quad (3.4)$$

Agora, com a finalidade de reescrever a condição (3.4) com graus de liberdade adicionais, pode-se aplicar o Lema de Finsler. Deste modo, de acordo com o Lema 2.3 define-se o vetor \mathcal{B} de tal maneira que $\mathcal{B}x = 0$. Assim, tem-se

$$\mathcal{B} = \begin{bmatrix} -I_n & A + B\Gamma_j K & B\delta(\xi_j) \end{bmatrix}, \mathbf{F} = \begin{bmatrix} F_{1j} \\ F_{2j} \\ F_{3j} \end{bmatrix}, \mathbf{FB} = \begin{bmatrix} -F_{1j} & F_{1j}(A + B\Gamma_j K) & F_{1j}B\delta(\xi_j) \\ -F_{2j} & F_{2j}(A + B\Gamma_j K) & F_{2j}B\delta(\xi_j) \\ -F_{3j} & F_{3j}(A + B\Gamma_j K) & F_{3j}B\delta(\xi_j) \end{bmatrix}$$

Definindo o vetor $\eta = [x^T \ 1]^T$, pode-se representar a desigualdade (3.4) por $\eta^T M \eta$. Desta maneira, segundo o Lema de Finsler, se existir uma matriz F a condição (3.4) é equivalente a $M + \text{sym}\{FB\} < 0$, que corresponde a desigualdade (3.2) do Teorema 3.1.

Além de garantir a estabilidade do sistema (3.1) para todos os pontos dentro do elipsóide, deve-se garantir que o elipsóide obtido contém o polítopo em que o modelo do sistema saturado é válido. De acordo com Boyd et al. (1994), a verificação da LMI, $v_i^T P v_i \leq 1$, é suficiente para garantir que o elipsóide definido por:

$$\mathcal{E}(P, 1) = \{x \in \mathbb{R}^n : x^T P x \leq 1\}$$

contenha um polítopo definido pelos vértices v . Assim, pode-se modificar a definição de $\mathcal{S}(R_j, d_j)$ para

$$\mathcal{S}(R_j, d_j) = \left\{ x \in \mathbb{R}^n : \left(\frac{R_{j(i)}}{d_{j(i)}} \right) x \leq 1 \right\}$$

Então, tem-se que

$$\left(\frac{R_{j(i)}}{d_{j(i)}} \right)^T P \left(\frac{R_{j(i)}}{d_{j(i)}} \right) \leq 1$$

o que pode ser reescrito como

$$1 - \left(\frac{R_{j(i)}}{d_{j(i)}} \right)^T P \left(\frac{R_{j(i)}}{d_{j(i)}} \right) \geq 0$$

Agora, por meio do Complemento de Schur, Lema 2.2, tem-se

$$\begin{bmatrix} 1 & \left(\frac{R_{j(i)}}{d_{j(i)}} \right)^T \\ \left(\frac{R_{j(i)}}{d_{j(i)}} \right) & P^{-1} \end{bmatrix} > 0 \quad (3.5)$$

Multiplicando-se a desigualdade (3.5) pela esquerda e pela direita por $\begin{bmatrix} 1 & 0 \\ 0 & P \end{bmatrix}$ chega-se a condição (3.3), concluindo assim a prova do Teorema 3.1. \square

3.2 Sistemas de tempo discreto

O resultado apresentado nesta seção considera o mesmo tipo de sistemas abordados na seção anterior, porém para o caso de tempo discreto. Considerando o seguinte sistema discreto

$$x[k+1] = (A + B\Gamma_j K)x[k] + B\delta(\xi_j) \quad (3.6)$$

o teorema a seguir é proposto para análise da estabilidade do sistema (3.6).

Teorema 3.2. *Se existir matrizes simétricas positivas definidas $P_j \in \mathbb{R}^{n \times n}$, onde $j = 1, \dots, 3^m$,*

de tal maneira que

$$\begin{bmatrix} (A + B\Gamma_j K)^T P_j (A + B\Gamma_j K) - P_j & \star \\ \delta(\xi_j) B (A + B\Gamma_j K) & \delta(\xi_j)^T B^T P_j B \delta(\xi_j) \end{bmatrix} < 0 \quad (3.7)$$

$$\begin{bmatrix} 1 & \frac{R_{j(i)}^T}{d_{j(i)}} P_j \\ P_j \frac{R_{j(i)}}{d_{j(i)}} & P_j \end{bmatrix} > 0 \quad (3.8)$$

$\mathcal{E}(P_j, 1)$ estima a região de atração da origem do sistema (3.1). ■

Prova. Considerando a seguinte função quadrática $V[x] = x^T P_j x \forall j = 1, \dots, 3^m$, como a função candidata de Lyapunov, deseja-se obter condições para garantir que a sua diferença seja negativa. Para tanto, tem-se que

$$\begin{aligned} \Delta V &= V[k+1] - V[k] \\ &= x[k+1]^T P_j x[k+1] - x[k]^T P_j x[k] \\ &= (x[k]^T (A + B\Gamma_j K)^T + \delta(\xi_j)^T B^T) P_j [(A + B\Gamma_j K)x[k] + B\delta(\xi_j)] - x[k]^T P_j x[k] \\ &= x[k]^T (A + B\Gamma_j K)^T P_j (A + B\Gamma_j K)x[k] + 2x[k]^T (A + B\Gamma_j K)^T P_j B \delta(\xi_j) \\ &\quad + \delta(\xi_j)^T B^T P_j B \delta(\xi_j) - x[k]^T P_j x[k] \end{aligned}$$

Definindo-se o vetor $\begin{bmatrix} x[k]^T & 1 \end{bmatrix}^T$, pode-se reescrever a desigualdade acima em forma matricial, obtendo assim a condição (3.7) do Teorema 3.2.

A condição (3.8) é obtida de forma análoga a condição (3.3) do Teorema 3.1. □

3.3 Exemplo Numérico

Neste seção é apresentado um exemplo numérico para comprovar a efetividade dos teoremas propostos neste capítulo.

Exemplo 3.1. Considere o sistema (3.1) dado pelas seguintes matrizes vide (KIM; BIEN, 1994):

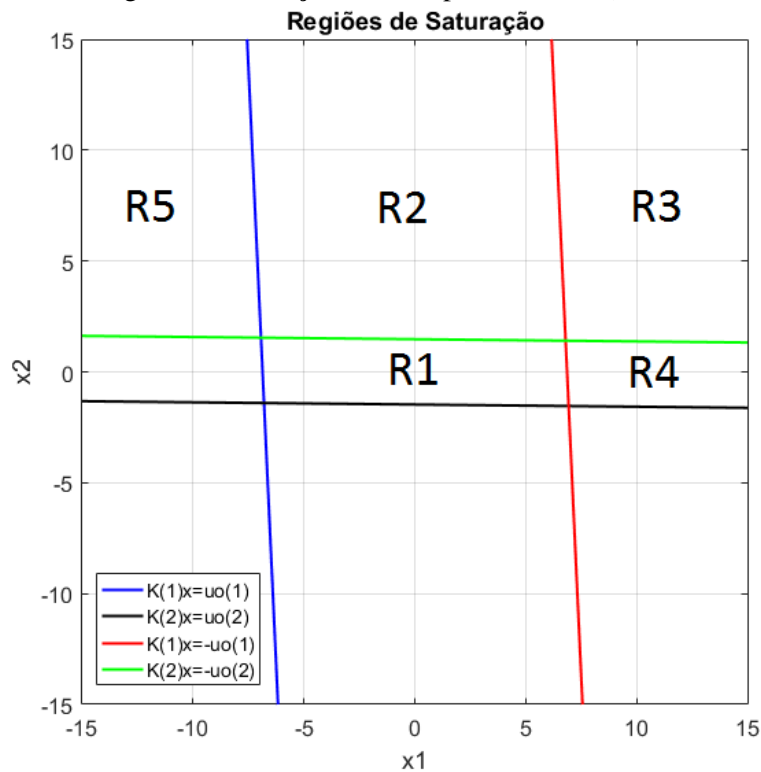
$$A = \begin{bmatrix} -0.1 & -0.1 \\ 0.1 & -3.0 \end{bmatrix}, \quad B = \begin{bmatrix} 5 & 0 \\ 0 & 1 \end{bmatrix}, \quad K = \begin{bmatrix} -0,7283 & -0,0338 \\ -0,0135 & -1,3583 \end{bmatrix}, \quad u_o = \begin{bmatrix} 5 \\ 2 \end{bmatrix}$$

As matrizes R_j e vetores d_j que caracterizam as regiões de saturação do sistema são:

$$\begin{aligned}
 R_1 &= \begin{bmatrix} K \\ K \end{bmatrix} & d_1 &= \begin{bmatrix} u_o \\ -u_o \end{bmatrix}, & R_2 &= \begin{bmatrix} K \\ K_{(1)} \end{bmatrix} & d_2 &= \begin{bmatrix} u_{o(1)} \\ -u_{o(2)} \\ -u_{o(1)} \end{bmatrix} \\
 R_3 &= \begin{bmatrix} K \end{bmatrix} & d_3 &= \begin{bmatrix} -u_o \end{bmatrix}, & R_4 &= \begin{bmatrix} K \\ K_{(2)} \end{bmatrix} & d_4 &= \begin{bmatrix} -u_o \\ u_{o(2)} \end{bmatrix} \\
 R_5 &= \begin{bmatrix} K \end{bmatrix} & d_5 &= \begin{bmatrix} u_{o(1)} \\ -u_{o(2)} \end{bmatrix}
 \end{aligned}$$

Os polítopos descritos pelas matrizes R_j e os vetores d_j podem ser vistos na figura 10.

Figura 10 – Regiões de saturação do exemplo numérico (KIM; BIEN, 1994).



Fonte: Autor 2016.

É possível observar que devido a simetria dos polítopos que descrevem as regiões de saturação não é necessário considerar todas as regiões. Neste caso como $m = 2$, $j = 1, \dots, 9$, entretanto apenas as regiões $1, \dots, 5$ serão consideradas. Além disso, verifica-se que as regiões R_2, R_3, R_4 e R_5 , não são conjuntos numéricos fechados, para tanto inclui-se a variável β , de

tal maneira que:

$$\begin{aligned}
 R_1 &= \begin{bmatrix} K \\ K \end{bmatrix} & d_1 &= \begin{bmatrix} u_o \\ -u_o \end{bmatrix}, \\
 R_2 &= \begin{bmatrix} K \\ K_{(1)} \\ K_{(2)} \end{bmatrix} & d_2 &= \begin{bmatrix} u_{o(1)} \\ -u_{o(2)} \\ -u_{o(1)} \\ -\beta u_{o(2)} \end{bmatrix} & R_3 &= \begin{bmatrix} K \\ K \end{bmatrix} & d_3 &= \begin{bmatrix} -u_o \\ -\beta u_o \end{bmatrix}, \\
 R_4 &= \begin{bmatrix} K \\ K_{(2)} \\ K_{(1)} \end{bmatrix} & d_4 &= \begin{bmatrix} -u_o \\ u_{o(2)} \\ \beta u_{o(1)} \end{bmatrix} & R_5 &= \begin{bmatrix} K \\ K \end{bmatrix} & d_5 &= \begin{bmatrix} u_{o(1)} \\ -u_{o(2)} \\ \beta u_{o(1)} \\ -\beta u_{o(2)} \end{bmatrix}
 \end{aligned}$$

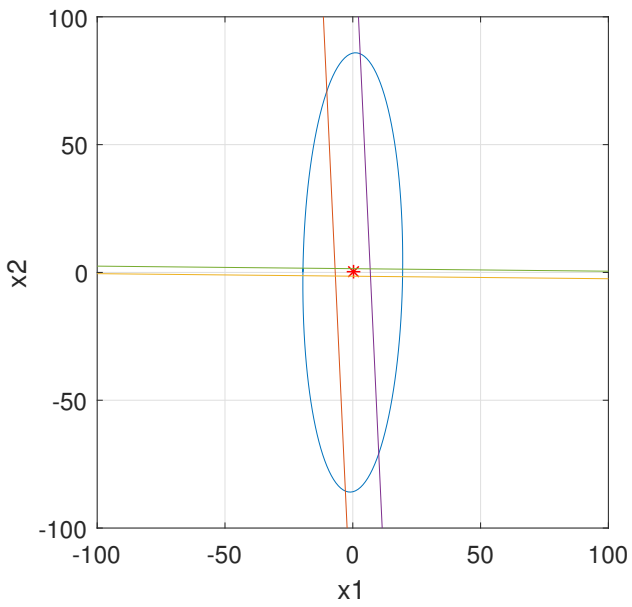
A variável β representa quantas vezes maior o elipsóide que contem o politopo que descreve a respectiva região de saturação deve ser. Portanto, a maximização de β forçará a obtenção dos maiores elipsóides. Neste exemplo numérico, não será considerado a maximização de β , mas isso poderia ser realizado por exemplo com a função `>> fminsearch` do MATLAB.

Deste maneira, aplica-se o Teorema 3.1 com a finalidade de investigar as regiões no qual o sistema é estável. Os elipsóides definidos pelas matrizes P_j podem ser vistos nas figuras 11, 12 e 13. O script criado e executado para este exemplo numérico pode ser visto no Apêndice A. Para obtenção desses resultados utilizou-se o software MATLAB juntamente com o parser YALMIP (LÖFBERG, 2004) e o solver SDPT3 (TUTUNCU; TOH; TODD, 1999).

Como forma de ilustração da simetria bem como da estabilidade do sistema dentro dos elipsóides obtidos. A figura 14, apresenta as trajetórias do sistema quando as condições iniciais pertencem a região R_3 e R_5 . Para tanto, o sistema foi inicializado em $x_1 = 50, x_2 = 50, x_1 = -50, x_2 = -50$ e $x_1 = -75, x_2 = 60, x_1 = 75, x_2 = -60$. É possível observar que quando o sistema é inicializado nas regiões opostas as regiões R_3 e R_5 , as trajetórias são idênticas quando as condições iniciais pertencem as regiões R_3 e R_5 . Esses resultados foram obtidos por meio do software SIMULINK.

▽

Figura 11 – Elipsóides definidos por P_1 e P_2 , respectivamente



Fonte: Autor 2016.

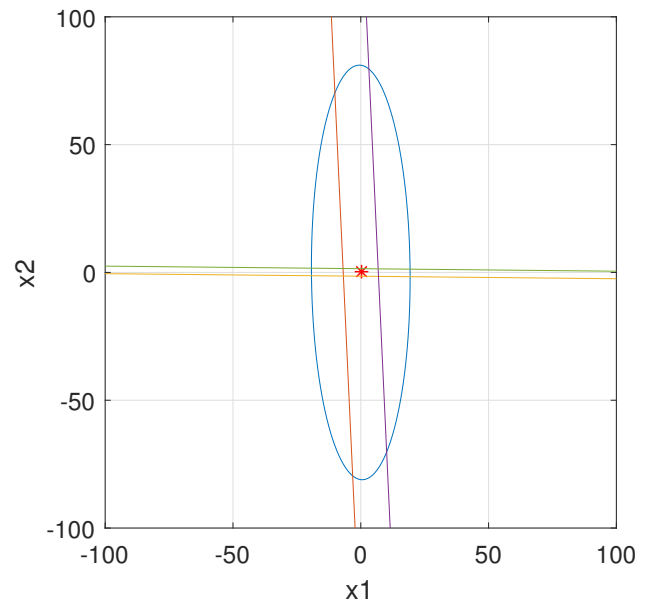


Figura 12 – Elipsóides definidos por P_3 e P_4 , respectivamente.



Fonte: Autor 2016.

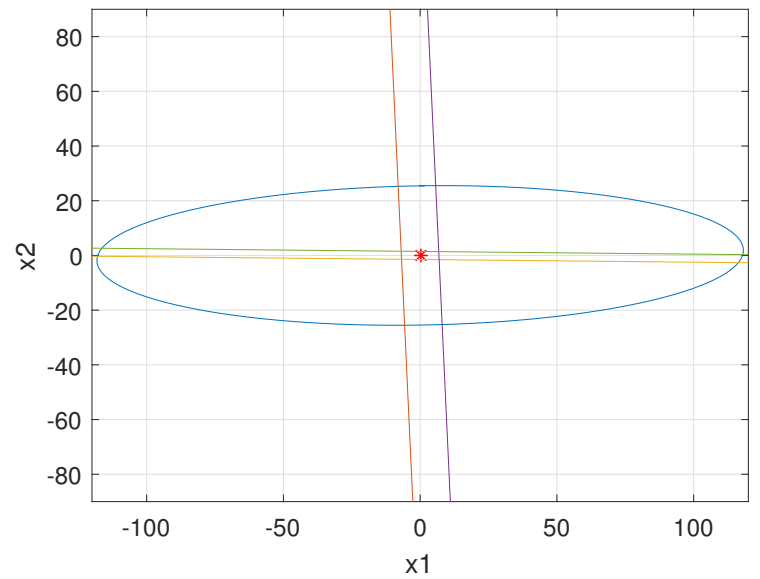
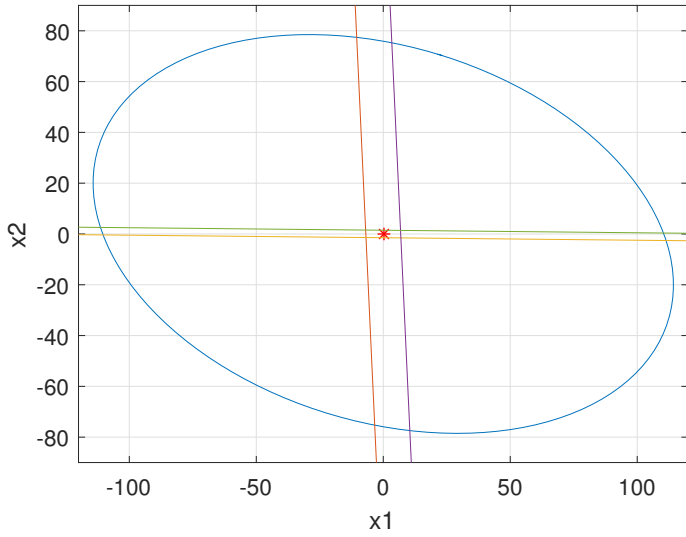


Figura 13 – Elipsóides definidos por P_5 e P_j , $j = 1, \dots, 5$.



Fonte: Autor 2016.

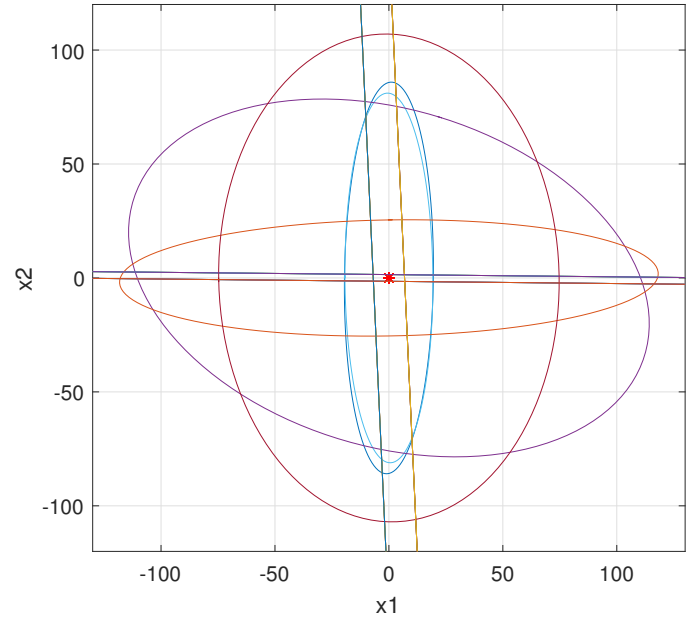
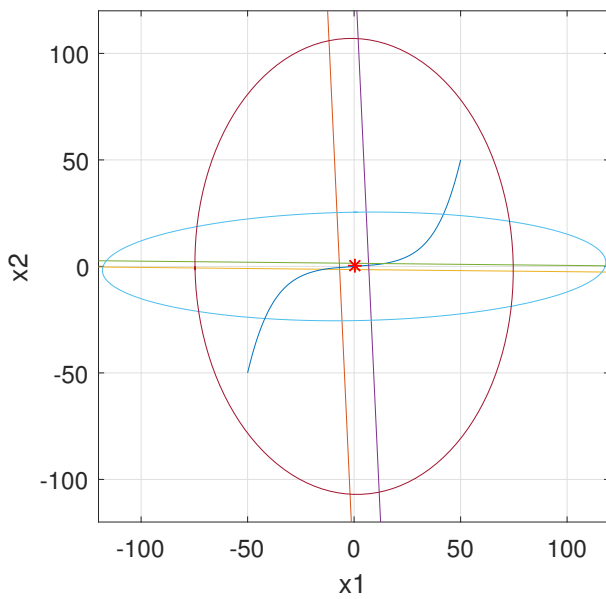
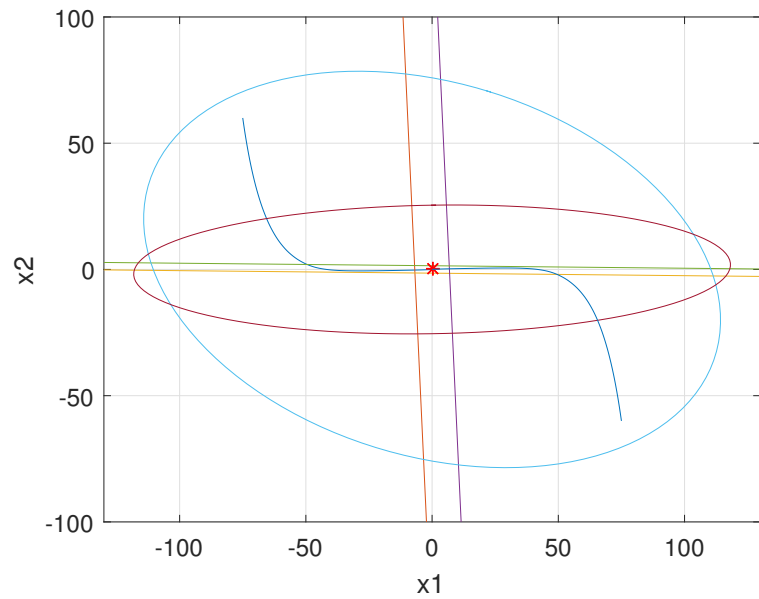


Figura 14 – Trajetórias do sistema para condições iniciais $x_1 = 50$, $x_2 = 50$ e $x_1 = -75$, $x_2 = 60$, respectivamente.



Fonte: Autor 2016.



4 ESTABILIZAÇÃO DE SISTEMAS SATURANTES VIA REGIÕES DE SATURAÇÃO

Este capítulo apresenta condições para a estabilização de sistemas com atuadores saturantes representando a saturação pela modelagem de regiões de saturação por meio da realimentação de estados. Inicialmente o problema é exposto, após o teorema obtido é apresentado.

Diferentemente do capítulo anterior, em que o objetivo era obter regiões definidas por elipsóides no qual o sistema (3.1) é estável, neste capítulo deseja-se não somente definir regiões do espaço de estados em que o sistema é estável, mas também a matriz de ganhos K . Deste modo, tornando um problema de análise em um problema de síntese. Para tanto, o seguinte teorema é proposto para síntese de uma matriz de ganhos K que estabilize o sistema (3.1) por meio da realimentação de estados.

Teorema 4.1. *Se existir matrizes simétricas positivas definidas $Q_j \in \mathfrak{R}^{n \times n}$, uma matriz $K \in \mathfrak{R}^{m \times n}$ e escalar $\alpha \in \mathfrak{R}$, onde $j = 1, \dots, 2^m$, de tal maneira que*

$$\begin{bmatrix} \text{sym}\{AQ_j + B\Gamma_j K Q_j\} + \alpha Q_j & \star \\ \delta(\xi_j)^T B^T & -\alpha \delta(\xi_j)^T \delta(\xi_j) \end{bmatrix} < 0 \quad (4.1)$$

$$\begin{bmatrix} 1 & \frac{(\Xi_j K)_{(i)}}{d_{j(i)}} \\ \frac{(\Xi_j K)_{(i)}^T}{d_{j(i)}} & Q_j \end{bmatrix} > 0, \quad i = 1, \dots, m \quad (4.2)$$

o sistema (3.1) é estabilizado pela matriz de ganho K e $\mathcal{E}(P_j, 1)$, onde $Q_j = P_j^{-1}$, estima a região de atração da origem. ■

Prova. Considerando o seguinte sistema com a saturação presente no atuador:

$$\dot{x}(t) = Ax(t) + B \text{sat}(Kx(t)) + B_\omega \omega(t) \quad (4.3)$$

onde $\omega \in \mathfrak{R}^m$ é um vetor de perturbações externas limitadas em amplitude e constante. Substituindo o termo $\text{sat}(Kx(t))$ por $\Gamma_j Kx(t)$, considerando que $B_\omega = B$ e que $\omega = \delta(\xi_j)$ o sistema (4.3) pode ser reescrito por:

$$\dot{x}(t) = Ax(t) + B\Gamma_j Kx(t) + B\delta(\xi_j) \quad (4.4)$$

É possível verificar que o sistema (4.4) é equivalente ao sistema (3.1). Desta maneira, considerando a seguinte função quadrática $V(x) = x^T P_j x \quad \forall j = 1, \dots, 2^m$, como a função candidata de Lyapunov, deseja-se obter condições para garantir que a sua derivada seja negativa. Para

tanto, defini-se a função auxiliar $\mathcal{J}(t)$, assim como proposto em (ZUO, 2004):

$$\begin{aligned}
\mathcal{J}(t) &= \dot{V}(x) + \alpha P_j - \alpha \omega^T \omega < 0 \\
&= \dot{x}^T P_j x + x^T P_j \dot{x} + \alpha P_j - \alpha \delta(\xi_j)^T \delta(\xi_j) < 0 \\
&= [(A + B\Gamma_j K)x + B\delta(\xi_j)]^T P_j x + x^T P_j [(A + B\Gamma_j K)x + B\delta(\xi_j)] + \alpha P_j \\
&\quad - \alpha \delta(\xi_j)^T \delta(\xi_j) < 0 \\
&= x^T (A + B\Gamma_j K)^T P_j x + \delta(\xi_j)^T B^T P_j x + x^T P_j (A + B\Gamma_j K)x + x^T P_j B \delta(\xi_j) \\
&\quad + \alpha P_j - \alpha \delta(\xi_j)^T \delta(\xi_j) < 0
\end{aligned}$$

Pode-se reescrever a inequação acima em forma matricial por

$$\begin{bmatrix} x^T & 1 \end{bmatrix}^T \begin{bmatrix} \text{sym}\{P_j A + P_j B\Gamma_j K\} + \alpha P_j & \star \\ \delta(\xi_j)^T B^T P_j & -\alpha \delta(\xi_j)^T \delta(\xi_j) \end{bmatrix} \begin{bmatrix} x \\ 1 \end{bmatrix} < 0$$

Multiplica-se a desigualdade pela esquerda e pela direita por $\begin{bmatrix} P_j^{-1} & 0 \\ 0 & 1 \end{bmatrix}$ e realizando a seguinte troca de variável, $Q_j = P_j^{-1}$, obtemos assim a BMI (4.1) do Teorema 4.1. Essa condição se caracteriza por uma BMI, devido a ocorrência do produto de duas variáveis nos termos: $B\Gamma_j K Q_j$ e αQ_j .

Comentário 6. *É importante observar, que a variável α neste contexto, representa a taxa de convergência do sistema (4.4), conforme (ZUO, 2004). De acordo com Hu e Lin (2001), a maximização de α resulta na obtenção do maior elipsóide definido pelas matrizes P_j . Entretanto, por se tratar de uma BMI, não se dispõe ainda de ferramentas computacionais para sua resolução numérica.*

A segunda condição do Teorema 4.1, garante que os elipsóides obtidos contêm os respectivos politopos da região de saturação. A sua prova segue de maneira análoga a da condição (3.3) do Teorema 3.1. Com a diferença de que a matriz R_j que define a região de saturação j é representada pelo produto $\Xi_j K$. Onde a matriz Ξ é utilizada para selecionar as linhas da matriz K , com o propósito de ter a matriz K explícita na condição, uma vez que deseja-se obter a matriz K . \square

5 CONCLUSÕES E PERSPECTIVAS

Este trabalho estudou a análise de sistemas lineares de tempo contínuo e discreto sujeitos a restrições no atuador, em que a saturação foi modelada pelas regiões de saturação. Uma revisão bibliográfica de trabalhos importantes na área foi apresentada, bem como definições, lemas e ferramentas matemáticas importante para o entendimento desse trabalho. Os teoremas expostos propõem a utilização de um elipsóide para cada região de saturação, no qual as condições desses teoremas são expressas na forma de Inequações Matriciais Lineares (LMI), para o problema de análise e na forma de Inequações Matriciais Bilinares (BMI), para o problema de síntese. Um exemplo numérico foi apresentado, para comprovar a efetividade do teorema de análise proposto, sendo a sua resolução realizada por meio do *software* MATLAB juntamente com o *parser* YALMIP. Este trabalho deu continuidade a (BINOTTI, 2015), entre as possibilidades de continuação destacam-se:

- Resolução numérica das BMIs do Teorema para síntese de uma realimentação de estados;
- Adicionar um compensador *antiwindup* estático aos Teoremas propostos;
- Desenvolver métodos para estabilização de sistemas discretos no tempo e quantizados sujeitos a saturação e ao atraso.

REFERÊNCIAS

- BENDER, F. A. **Técnicas de compensação dinâmica para sistemas lineares com saturação de controle**. 2006. 85 p. Dissertação (Mestrado em Engenharia Elétrica) — Programa de Pós-Graduação em Engenharia Elétrica, UFRGS, Porto Alegre, 2006.
- BENDER, F. A. **Técnicas de síntese de compensadores antiwindup para sistemas com atraso**. 2010. 96 p. Tese (Doutorado em Engenharia Elétrica) — Programa de Pós-Graduação em Engenharia Elétrica, UFRGS, Porto Alegre, 2010.
- BINOTTI, V. **Estabilidade e estabilização de sistemas de tempo discreto sujeitos a saturação e atraso**. 2015. 45 p. Trabalho de Conclusão de Curso para Engenharia de Controle e Automação — Universidade de Caxias do Sul, Caxias do Sul, 2015.
- BOLKCOM, C. V-22 osprey tilt-rotor aircraft. **CRS Report for Congress**, [S.l.], p. RL31384, 2005.
- BOYD, S. P. et al. **Linear matrix inequalities in system and control theory**. 1. ed. Philadelphia, PA: Society for Industrial and Applied Mathematic, 1994. 193 p. (Studies in Applied Mathematics 15).
- FONG, I.; HSU, C. State feedback stabilization of single input systems through actuators with saturation and dead zone characteristics. **Conference on Decision and Control**, Sydney, Australia, 2000.
- FRIDMAN, E. **Introduction to time-delay systems: analysis and control**. 1. ed. [S.l.]: Birkhäuser Basel, 2014. 362 p. (Systems & Control: Foundations & Applications).
- GAHINET, P.; APKARIAN, P. A linear matrix inequality approach to h_∞ control. **International Journal of Robust and Nonlinear Control**, [S.l.], p. 1–20, 1994.
- GU, K.; KHARITONOV, V. L.; CHEN, J. **Stability of time-delay systems**. 1. ed. [S.l.]: Birkhäuser Boston, 2003. (Control Engineering).
- HAI-HUA, Y.; GUANGREN, D. **Lmis in control systems : analysis, design and applications**. 4. ed. Upper Saddle River: CRC Press, 2013. 483 p.
- HETEL, L.; DAAFOUZ, J.; IUNG, C. Equivalence between the lyapunov-krasovskii functional approach for discrete delay systems and the stability conditions for switched systems. **Nonlinear Analysis: Hybrid Systems**, [S.l.], v. 2, n. 3, p. 697–705, 2008.
- HINDI, H. A tutorial on convex optimization. **Proceedings of the 2004 American Control Conference**, Boston, MA, p. 3252–3265, 2004.
- HU, T.; LIN, Z. **Control systems with actuator saturation: analysis and design**. 1. ed. [S.l.]: Birkhäuser Basel, 2001. 430 p. (Control Engineering).
- HU, T.; LIN, Z.; CHEN, B. M. Analysis and design for discrete-time linear systems subject to actuator saturation. **Syst. & Contr. Lett.**, [S.l.], v. 45, p. 97–112, 2002.
- JOHANSSON, M. **Piecewise linear control systems a computational approach**. [S.l.]: Springer, 2003. (Lecture Notes in Control and Information Sciences 284).

- JOHANSSON, M.; RANTZER, A. Computation of piecewise quadratic lyapunov functions for hybrid systems. , [S.l.], v. 43, p. 555–559, 1998.
- KHALIL, H. K. **Nonlinear systems**. 3. ed. Upper Saddle River, NJ: Prentice Hall, 2002. 750 p.
- KIM, J.-H.; BIEN, Z. Robust stability of uncertain linear systems with saturating actuators. **IEEE Trans. Autom. Contr.**, [S.l.], v. 39, n. 1, p. 202–207, 1994.
- LYAPUNOV, A. M. **General problem of the stability of motion**. 1. ed. [S.l.]: CRC Press, 1992.
- LÖFBERG, J. Yalmip : a toolbox for modeling and optimization in MATLAB. **Proceedings of the CACSD Conference**, Taipei, Taiwan, 2004.
- MAHMOUD, M. S. **Switched time-delay systems: stability and control**. 1. ed. New York, NY: Springer, 2010. 433 p.
- MICHIELS, W.; NICULESCU, S.-I. **Stability and stabilization of time-delay systems**. Philadelphia, PA: SIAM, 2007. 378 p. (Advances in Design & Control).
- OGATA, K. **Discrete-time control systems**. 2. ed. Upper Saddle River, NJ: Pearson Education, 1994. 745 p.
- OLIVEIRA, M. C. de; SKELTON, R. E. **Stability tests for constrained linear systems**. [S.l.]: Springer London, 2001.
- PAIM, C. **Análise e controle de sistemas lineares sujeitos a saturação**. 2003. 101 p. Tese (Doutorado em Engenharia Elétrica) — Programa de Pós-Graduação em Engenharia Elétrica, UFSC, Florianópolis, 2003.
- RICHARD, J.-P. Time-delay systems: an overview of some recent advances and open problems. **Automatica**, [S.l.], v. 39, p. 1667–1694, 2003.
- SILVA JR., J. M. G. da; TARBOURIECH, S. Contractive polyhedra for linear continuous-time systems with saturating controls. **Proceedings of the American Control Conference**, San Diego. California, 1999.
- SILVA JR., J. M. G. da; TARBOURIECH, S. Antiwindup design with guaranteed regions of stability: an lmi-based approach. **IEEE Trans. Autom. Contr.**, [S.l.], v. 50, p. 106–111, 2005.
- STEIN, G. Respect the unstable. **IEEE Control Systems Magazine**, [S.l.], p. 12–25, 2003.
- SUN, J.; CHEN, J. A note on stability and stabilization of discrete-time systems with time-varying delay. **Proceedings of the 31st Chinese Control Conference**, Hefei, China, p. 1430–1434, 2012.
- TARBOURIECH, S. et al. **Stability and stabilization of linear systems with saturating actuators**. 1. ed. [S.l.]: Springer-Verlag London, 2011. 430 p.
- TEWARI, A. **Modern control design: with matlab and simulink**. [S.l.]: Wiley, 2002.
- TUTUNCU, R. H.; TOH, K. C.; TODD, M. J. Sdpt3 — a matlab software package for semidefinite-quadratic-linear programming. **Optimization Methods and Software**, [S.l.], v. 11, p. 545–581, 1999.

Why the grippen crashed. **Aerospace America**, [S.l.], p. 11, 1994.

ZUO, Z. Persistent bounded disturbance rejection for linear systems subject to actuator saturation. **Sixteenth International Symposium on Mathematical Theory of Networks and Systems**, Belgium, 2004.

ÅSTRÖM, K. J.; MURRAY, R. M. **Feedback systems**: an introduction for scientists and engineers. 2.11b. ed. Princeton, NJ: Princeton University Press, 2012. 396 p.

APÊNDICE A – SCRIPT MATLAB EXEMPLO NUMÉRICO

```

1  %*****%
2  % This script performs stability analysis %
3  % Numerical example from TarGom11 book %
4  % 15APR2016 %
5  % Vinicius Binotti %
6  %*****%
7
8  clear all
9  close all
10 clc
11
12 % Plant
13 A = [-.1  -.1;
14      .1  -3];
15
16 B = [5  0;
17      0  1];
18
19 K = [-0.7283  -0.0338;
20      -0.0135  -1.3583];
21
22 u_o = [5;
23        2];
24
25 % Dimensions
26 n = size(A,1);
27 m = size(B,2);
28
29 % Parameters
30 beta = 10;
31
32 % Regions of Saturation
33 R1 = [K;
34       K];
35 d1 = [u_o;
36       -u_o];
37
38 R2 = [K(1,:);
39       K(1,:);
40       K(2,:);
41       K(2,:)];
42 d2 = [u_o(1);

```

```

43     -u_o(1);
44     -u_o(2);
45     -beta*u_o(2)];
46
47 R3 = [K;
48       K];
49 d3 = [-u_o;
50       -beta*u_o];
51
52 R4 = [K;
53       K(2,:);
54       K(1,:)];
55 d4 = [u_o(1);
56       -u_o(2);
57       u_o(2);
58       beta*u_o(1)];
59
60 R5 = [K;
61       K];
62 d5 = [ u_o(1);
63       -u_o(2);
64       beta*u_o(1);
65       -beta*u_o(2)];
66
67 % Variables
68 lbd21 = sdpvar(1, 1, 'full', 'real');
69 lbd22 = sdpvar(1, 1, 'full', 'real');
70 lbd23 = sdpvar(1, 1, 'full', 'real');
71 lbd24 = sdpvar(1, 1, 'full', 'real');
72
73 lbd31 = sdpvar(1, 1, 'full', 'real');
74 lbd32 = sdpvar(1, 1, 'full', 'real');
75 lbd33 = sdpvar(1, 1, 'full', 'real');
76 lbd34 = sdpvar(1, 1, 'full', 'real');
77
78 lbd41 = sdpvar(1, 1, 'full', 'real');
79 lbd42 = sdpvar(1, 1, 'full', 'real');
80 lbd43 = sdpvar(1, 1, 'full', 'real');
81 lbd44 = sdpvar(1, 1, 'full', 'real');
82
83 lbd51 = sdpvar(1, 1, 'full', 'real');
84 lbd52 = sdpvar(1, 1, 'full', 'real');
85 lbd53 = sdpvar(1, 1, 'full', 'real');
86 lbd54 = sdpvar(1, 1, 'full', 'real');
87
88 Lbd2 = [lbd21 lbd22 lbd23 lbd24];
89 Lbd3 = [lbd31 lbd32 lbd33 lbd34];

```



```

90 Lbd4 = [lbd41 lbd42 lbd43 lbd44];
91 Lbd5 = [lbd51 lbd52 lbd53 lbd54];
92
93 for k = 1:5
94     assignin('base',strcat('P',int2str(k)),sdpvar(n,n, 'symmetric', ...
95         'real'));
96
97 for k = 2:5
98     assignin('base',strcat('F1',int2str(k)),sdpvar(n,n, 'full', 'real'));
99     assignin('base',strcat('F2',int2str(k)),sdpvar(n,n, 'full', 'real'));
100    assignin('base',strcat('F3',int2str(k)),sdpvar(1,n, 'full', 'real'));
101 end
102
103 Gamma2 = [1 0;
104           0 0];
105 Gamma3 = [0 0;
106           0 0];
107 Gamma4 = [0 0;
108           0 1];
109 Gamma5 = [0 0;
110           0 0];
111
112 Δ2 = [ 0;
113       u_o(2)];
114 Δ3 = [u_o(1);
115       u_o(2)];
116 Δ4 = [-u_o(1);
117        0];
118 Δ5 = [-u_o(1);
119        u_o(2)];
120
121 %***** LMIs *****%
122 LMIs = [];
123 %***** R1 *****%
124 M01 = [P1*A+P1*B*K];
125 M01 = M01 + M01';
126
127 M11 = [1 (R1(1,:)/d1(1))*P1;
128        P1*(R1(1,:)'/d1(1)) P1];
129
130 M12 = [1 (R1(2,:)/d1(2))*P1;
131        P1*(R1(2,:)'/d1(2)) P1];
132
133 M13 = [1 (R1(3,:)/d1(3))*P1;
134        P1*(R1(3,:)'/d1(3)) P1];
135

```

```

136 M14 = [1 (R1(4,:)/d1(4))*P1;
137 P1*(R1(4,:)/d1(4)) P1];
138
139 LMIs = [LMIs, M01 < 0, M11 > 0, M12 > 0, M13 > 0, M14 > 0, P1 > 0];
140
141 %***** R2 *****%
142 M02 = [-F12 zeros(n,n) ...
143 -F22+A'*F12'+K'*Gamma2*B'*F12' ...
144 P2*A+P2*B*Gamma2*K+F22*A+F22*B*Gamma2*K ...
145 zeros(n,1);
146 -F32+Δ2'*B'*F12' Δ...
147 2'*B'*P2-.5*Lbd2*R2+F32*A+F32*B*Gamma2*K+Δ2'*B'*F22' ...
148 .5*Lbd2*d2+F32*B*Δ2];
149 M02 = M02 + M02';
150
151 M21 = [1 (R2(1,:)/d2(1))*P2;
152 P2*(R2(1,:)/d2(1)) P2];
153
154 M22 = [1 (R2(2,:)/d2(2))*P2;
155 P2*(R2(2,:)/d2(2)) P2];
156
157 M23 = [1 (R2(3,:)/d2(3))*P2;
158 P2*(R2(3,:)/d2(3)) P2];
159
160 M24 = [1 (R2(4,:)/d2(4))*P2;
161 P2*(R2(4,:)/d2(4)) P2];
162
163 LMIs = [LMIs, lbd21 > 0, lbd22 > 0, lbd23 > 0, lbd24 > 0];
164 LMIs = [LMIs, M02 < 0, M21 > 0, M22 > 0, M23 > 0, M24 > 0, P2 > 0];
165
166 %***** R3 *****%
167 M03 = [-F13 zeros(n,n) ...
168 -F23+A'*F13'+K'*Gamma3*B'*F13' ...
169 P3*A+P3*B*Gamma3*K+F23*A+F23*B*Gamma3*K ...
170 zeros(n,1);
171 -F33+Δ3'*B'*F13' Δ...
172 3'*B'*P3-.5*Lbd3*R3+F33*A+F33*B*Gamma3*K+Δ3'*B'*F23' ...
173 .5*Lbd3*d3+F33*B*Δ3];
174 M03 = M03 + M03';
175
176 M31 = [1 (R3(1,:)/d3(1))*P3;
177 P3*(R3(1,:)/d3(1)) P3];
178
179 M32 = [1 (R3(2,:)/d3(2))*P3;
180 P3*(R3(2,:)/d3(2)) P3];

```

```

173
174 M33 = [1 (R3(3, :)/d3(3))*P3;
175          P3*(R3(3, :)' /d3(3)) P3];
176
177 M34 = [1 (R3(4, :)/d3(4))*P3;
178          P3*(R3(4, :)' /d3(4)) P3];
179
180 LMIs = [LMIs, lbd31 > 0, lbd32 > 0, lbd33 > 0, lbd34 > 0];
181 LMIs = [LMIs, M03 < 0, M31 > 0, M32 > 0, M33 > 0, M34 > 0, P3 > 0];
182
183 %***** R4 *****%
184 M04 = [-F14 zeros(n,n) ...
185          zeros(n,1);
186          -F24+A'*F14'+K'*Gamma4*B'*F14' ...
187          P4*A+P4*B*Gamma4*K+F24*A+F24*B*Gamma4*K ...
188          zeros(n,1);
189          -F34+Δ4'*B'*F14' Δ...
190          4'*B'*P4-.5*Lbd4*R4+F34*A+F34*B*Gamma4*K+Δ4'*B'*F24' ...
191          .5*Lbd4*d4+F34*B*Δ4];
192 M04 = M04 + M04';
193
194
195 M41 = [1 (R4(1, :)/d4(1))*P4;
196          P4*(R4(1, :)' /d4(1)) P4];
197
198 M42 = [1 (R4(2, :)/d4(2))*P4;
199          P4*(R4(2, :)' /d4(2)) P4];
200
201 M43 = [1 (R4(3, :)/d4(3))*P4;
202          P4*(R4(3, :)' /d4(3)) P4];
203
204 M44 = [1 (R4(4, :)/d4(4))*P4;
205          P4*(R4(4, :)' /d4(4)) P4];
206
207 LMIs = [LMIs, lbd41 > 0, lbd42 > 0, lbd43 > 0, lbd44 > 0];
208 LMIs = [LMIs, M04 < 0, M41 > 0, M42 > 0, M43 > 0, M44 > 0, P4 > 0];
209
210 %***** R5 *****%
211 M05 = [-F15 zeros(n,n) ...
212          zeros(n,1);
213          -F25+A'*F15'+K'*Gamma5*B'*F15' ...
214          P5*A+P5*B*Gamma5*K+F25*A+F25*B*Gamma5*K ...
215          zeros(n,1);
216          -F35+Δ5'*B'*F15' Δ...
217          5'*B'*P5-.5*Lbd5*R5+F35*A+F35*B*Gamma5*K+Δ5'*B'*F25' ...
218          .5*Lbd5*d5+F35*B*Δ5];
219 M05 = M05 + M05';
220

```

```

210 M51 = [1 (R5(1,:)/d5(1))*P5;
211         P5*(R5(1,:)'/d5(1)) P5];
212
213 M52 = [1 (R5(2,:)/d5(2))*P5;
214         P5*(R5(2,:)'/d5(2)) P5];
215
216 M53 = [1 (R5(3,:)/d5(3))*P5;
217         P5*(R5(3,:)'/d5(3)) P5];
218
219 M54 = [1 (R5(4,:)/d5(4))*P5;
220         P5*(R5(4,:)'/d5(4)) P5];
221
222 LMIs = [LMIs, lbd51 > 0, lbd52 > 0, lbd53 > 0, lbd54 > 0];
223 LMIs = [LMIs, M05 < 0, M51 > 0, M52 > 0, M53 > 0, M54 > 0, P5 > 0];
224
225 options = sdpsettings('solver','sdpt3','verbose',1,'radius',1e2);
226 solution = solvesdp(LMIs, [], options)
227
228 %***** Check Solution *****%
229
230 mP1 = double(P1);
231 mP2 = double(P2);
232 mP3 = double(P3);
233 mP4 = double(P4);
234 mP5 = double(P5);
235
236 mlbd21 = double(lbd21);
237 mlbd22 = double(lbd22);
238 mlbd23 = double(lbd23);
239 mlbd24 = double(lbd24);
240
241 mlbd31 = double(lbd31);
242 mlbd32 = double(lbd32);
243 mlbd33 = double(lbd33);
244 mlbd34 = double(lbd34);
245
246 mlbd41 = double(lbd41);
247 mlbd42 = double(lbd42);
248 mlbd43 = double(lbd43);
249 mlbd44 = double(lbd44);
250
251 mlbd51 = double(lbd51);
252 mlbd52 = double(lbd52);
253 mlbd53 = double(lbd53);
254 mlbd54 = double(lbd54);
255
256 mLbd2 = [mlbd21 mlbd22 mlbd23 mlbd24];

```

```

257 mLbd3 = [mlbd31 mlbd32 mlbd33 mlbd34];
258 mLbd4 = [mlbd41 mlbd42 mlbd43 mlbd44];
259 mLbd5 = [mlbd51 mlbd52 mlbd53 mlbd54];
260
261 mF12 = double(F12);
262 mF22 = double(F22);
263 mF32 = double(F32);
264 mF13 = double(F13);
265 mF23 = double(F23);
266 mF33 = double(F33);
267 mF14 = double(F14);
268 mF24 = double(F24);
269 mF34 = double(F34);
270 mF15 = double(F15);
271 mF25 = double(F25);
272 mF35 = double(F35);
273
274 %***** R1 *****%
275 disp('Check conditions Region of Linearity R1')
276 mM01 = [mP1*A+mP1*B*K];
277 mM01 = mM01 + mM01';
278
279 mM11 = [1 (R1(1, :)/d1(1))*mP1;
280          mP1*(R1(1, :)' /d1(1)) mP1];
281
282 mM12 = [1 (R1(2, :)/d1(2))*mP1;
283          mP1*(R1(2, :)' /d1(2)) mP1];
284
285 mM13 = [1 (R1(3, :)/d1(3))*mP1;
286          mP1*(R1(3, :)' /d1(3)) mP1];
287
288 mM14 = [1 (R1(4, :)/d1(4))*mP1;
289          mP1*(R1(4, :)' /d1(4)) mP1];
290 eig(mP1)' > 0
291 eig(mM01)' < 0
292 eig(mM11)' > 0
293 eig(mM12)' > 0
294 eig(mM13)' > 0
295 eig(mM14)' > 0
296
297 %***** R2 *****%
298 disp('Check conditions Region of Saturation R2')
299 mM02 = [-mF12 zeros(n,n) ... zeros(n,1);
300          -mF22+A'*mF12'+K'*Gamma2*B'*mF12' ...
           mP2*A+mP2*B*Gamma2*K+mF22*A+mF22*B*Gamma2*K ...
           zeros(n,1)];

```

```

301      -mF32+Δ2'*B'*mF12'          Δ...
          2'*B'*mP2-.5*mLbd2*R2+mF32*A+mF32*B*Gamma2*K+Δ2'*B'*mF22'    ...
          .5*mLbd2*d2+mF32*B*Δ2];
302 mM02 = mM02 + mM02';
303
304 mM21 = [1                      (R2(1,:)/d2(1))*mP2;
305          mP2*(R2(1,:)'/d2(1))    mP2];
306
307 mM22 = [1                      (R2(2,:)/d2(2))*mP2;
308          mP2*(R2(2,:)'/d2(2))    mP2];
309
310 mM23 = [1                      (R2(3,:)/d2(3))*mP2;
311          mP2*(R2(3,:)'/d2(3))    mP2];
312
313 mM24 = [1                      (R2(4,:)/d2(4))*mP2;
314          mP2*(R2(4,:)'/d2(4))    mP2];
315 eig(mP2) ' > 0
316 eig(mM02) ' < 0
317 eig(mM21) ' > 0
318 eig(mM22) ' > 0
319 eig(mM23) ' > 0
320 eig(mM24) ' > 0
321 mLbd2 > 0
322
323 %***** R3 *****%
324 disp('Check conditions Region of Saturation R3')
325 mM03 = [-mF13                      zeros(n,n) ...
                                                  zeros(n,1);
326          -mF23+A'*mF13'+K'*Gamma3*B'*mF13'    ...
          mP3*A+mP3*B*Gamma3*K+mF23*A+mF23*B*Gamma3*K    ...
          zeros(n,1);
327          -mF33+Δ3'*B'*mF13'          Δ...
          3'*B'*mP3-.5*mLbd3*R3+mF33*A+mF33*B*Gamma3*K+Δ3'*B'*mF23'    ...
          .5*mLbd3*d3+mF33*B*Δ3];
328 mM03 = mM03 + mM03';
329
330 mM31 = [1                      (R3(1,:)/d3(1))*mP3;
331          mP3*(R3(1,:)'/d3(1))    mP3];
332
333 mM32 = [1                      (R3(2,:)/d3(2))*mP3;
334          mP3*(R3(2,:)'/d3(2))    mP3];
335
336 mM33 = [1                      (R3(3,:)/d3(3))*mP3;
337          mP3*(R3(3,:)'/d3(3))    mP3];
338
339 mM34 = [1                      (R3(4,:)/d3(4))*mP3;
340          mP3*(R3(4,:)'/d3(4))    mP3];

```

```

341 eig(mP3)' > 0
342 eig(mM03)' < 0
343 eig(mM31)' > 0
344 eig(mM32)' > 0
345 eig(mM33)' > 0
346 eig(mM34)' > 0
347 mLbd3 > 0
348 %***** R4 *****%
349 disp('Check conditions Region of Saturation R4')
350 mM04 = [-mF14                                zeros(n,n) ...
                                                zeros(n,1);
351         -mF24+A'*mF14'+K'*Gamma4*B'*mF14'    ...
           mP4*A+mP4*B*Gamma4*K+mF24*A+mF24*B*Gamma4*K ...
                                                zeros(n,1);
352         -mF34+Δ4'*B'*mF14'                    Δ...
           4'*B'*mP4-.5*mLbd4*R4+mF34*A+mF34*B*Gamma4*K+Δ4'*B'*mF24' ...
           .5*mLbd4*d4+mF34*B*Δ4];
353 mM04 = mM04 + mM04';
354
355 mM41 = [1                                (R4(1,:)/d4(1))*mP4;
          mP4*(R4(1,:)'/d4(1))            mP4];
356
357
358 mM42 = [1                                (R4(2,:)/d4(2))*mP4;
          mP4*(R4(2,:)'/d4(2))            mP4];
359
360
361 mM43 = [1                                (R4(3,:)/d4(3))*mP4;
          mP4*(R4(3,:)'/d4(3))            mP4];
362
363
364 mM44 = [1                                (R4(4,:)/d4(4))*mP4;
          mP4*(R4(4,:)'/d4(4))            mP4];
365
366 eig(mP4)' > 0
367 eig(mM04)' < 0
368 eig(mM41)' > 0
369 eig(mM42)' > 0
370 eig(mM43)' > 0
371 eig(mM44)' > 0
372 mLbd4 > 0
373 %***** R5 *****%
374 disp('Check conditions Region of Saturation R5')
375 mM05 = [-mF15                                zeros(n,n) ...
                                                zeros(n,1);
376         -mF25+A'*mF15'+K'*Gamma5*B'*mF15'    ...
           mP5*A+mP5*B*Gamma5*K+mF25*A+mF25*B*Gamma5*K ...
                                                zeros(n,1);
377         -mF35+Δ5'*B'*mF15'                    Δ...
           5'*B'*mP5-.5*mLbd5*R5+mF35*A+mF35*B*Gamma5*K+Δ5'*B'*mF25' ...
           .5*mLbd5*d5+mF35*B*Δ5];

```

```

378 mM05 = mM05 + mM05';
379
380 mM51 = [1 (R5(1, :)/d5(1)) *mP5;
381          mP5*(R5(1, :)' /d5(1)) mP5];
382
383 mM52 = [1 (R5(2, :)/d5(2)) *mP5;
384          mP5*(R5(2, :)' /d5(2)) mP5];
385
386 mM53 = [1 (R5(3, :)/d5(3)) *mP5;
387          mP5*(R5(3, :)' /d5(3)) mP5];
388
389 mM54 = [1 (R5(4, :)/d5(4)) *mP5;
390          mP5*(R5(4, :)' /d5(4)) mP5];
391 eig(mP5)' > 0
392 eig(mM05)' < 0
393 eig(mM51)' > 0
394 eig(mM52)' > 0
395 eig(mM53)' > 0
396 eig(mM54)' > 0
397 mLbd5 > 0
398 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
399 num. of constraints = 71
400 dim. of sdp var = 92, num. of sdp blk = 30
401 dim. of socp var = 72, num. of socp blk = 1
402 dim. of linear var = 16
403 *****
404 SDPT3: Infeasible path-following algorithms
405 *****
406 version predcorr gam expon scale_data
407 HKM 1 0.000 1 0
408 it pstep dstep pinfeas dinfeas gap prim-obj dual-obj cputime
409 -----
410 0|0.000|0.000|1.3e+03|2.2e+00|1.8e+04| 1.048528e+03 0.000000e+00| ...
411 0:0:00| chol 1 1
412 1|0.685|0.749|4.2e+02|5.6e-01|6.0e+03| 6.370000e+02 0.000000e+00| ...
413 0:0:01| chol 1 1
414 2|0.778|0.719|9.3e+01|1.6e-01|2.1e+03| 5.361953e+02 0.000000e+00| ...
415 0:0:01| chol 1 1
416 3|0.804|0.802|1.8e+01|3.1e-02|6.4e+02| 3.373769e+02 0.000000e+00| ...
417 0:0:01| chol 1 1
418 4|0.717|0.737|5.2e+00|8.2e-03|2.5e+02| 1.661278e+02 0.000000e+00| ...
419 0:0:01| chol 1 1
420 5|0.890|0.851|5.7e-01|1.2e-03|6.3e+01| 5.337652e+01 0.000000e+00| ...
421 0:0:01| chol 1 1
422 6|0.870|0.816|7.5e-02|2.3e-04|1.6e+01| 1.492385e+01 0.000000e+00| ...
423 0:0:01| chol 1 1
424 7|0.855|1.000|1.1e-02|8.7e-04|6.7e+00| 5.976278e+00 0.000000e+00| ...

```



```

0:0:01| chol 1 1
418 8|0.895|1.000|1.1e-03|5.1e-04|1.5e+00| 1.330283e+00 0.000000e+00| ...
0:0:01| chol 1 1
419 9|1.000|1.000|2.0e-09|2.3e-04|3.8e-01| 3.598549e-01 0.000000e+00| ...
0:0:01| chol 1 1
420 10|0.975|1.000|5.0e-11|4.1e-10|1.6e-02| 1.648878e-02 0.000000e+00| ...
0:0:01| chol 1 1
421 11|0.989|1.000|5.6e-13|1.1e-11|1.8e-04| 1.831129e-04 0.000000e+00| ...
0:0:01| chol 1 1
422 12|0.989|1.000|6.1e-15|1.1e-12|2.0e-06| 2.012732e-06 0.000000e+00| ...
0:0:01| chol 1 1
423 13|0.596|1.000|2.5e-15|1.0e-12|1.2e-06| 1.180613e-06 0.000000e+00| ...
0:0:01| chol 1 1
424 14|0.630|1.000|9.1e-16|1.0e-12|6.7e-07| 6.722202e-07 0.000000e+00| ...
0:0:01| chol 1 1
425 15|0.670|1.000|3.0e-16|1.0e-12|3.6e-07| 3.642982e-07 0.000000e+00| ...
0:0:01| chol 1 1
426 16|0.693|1.000|9.3e-17|1.0e-12|1.9e-07| 1.912262e-07 0.000000e+00| ...
0:0:01| chol 1 1
427 17|0.700|1.000|2.8e-17|1.0e-12|9.9e-08| 9.943302e-08 0.000000e+00| 0:0:01|
428 stop: max(relative gap, infeasibilities) < 1.00e-07
429 -----
430 number of iterations = 17
431 primal objective value = 9.94330182e-08
432 dual objective value = 0.00000000e+00
433 gap := trace(XZ) = 9.94e-08
434 relative gap = 9.94e-08
435 actual relative gap = 9.94e-08
436 rel. primal infeas = 2.78e-17
437 rel. dual infeas = 1.00e-12
438 norm(X), norm(y), norm(Z) = 1.2e-06, 9.9e+01, 1.9e+03
439 norm(A), norm(b), norm(C) = 2.0e+02, 1.0e+00, 1.0e+02
440 Total CPU time (secs) = 1.45
441 CPU time per iteration = 0.09
442 termination code = 0
443 DIMACS: 2.8e-17 0.0e+00 1.0e-12 0.0e+00 9.9e-08 9.9e-08
444 -----
445 solution =
446
447 yalmiptime: 5.2467
448 solvertime: 2.4903
449 info: 'Successfully solved (SDPT3-4)'
450 problem: 0
451
452 Check conditions Region of Linearity R1
453 ans =
454 1 1

```

```

455 ans =
456     1     1
457 ans =
458     1     1     1
459 ans =
460     1     1     1
461 ans =
462     1     1     1
463 ans =
464     1     1     1
465 Check conditions Region of Saturation R2
466 ans =
467     1     1
468 ans =
469     1     1     1     1     1
470 ans =
471     1     1     1
472 ans =
473     1     1     1
474 ans =
475     1     1     1
476 ans =
477     1     1     1
478 ans =
479     1     1     1     1
480 Check conditions Region of Saturation R3
481 ans =
482     1     1
483 ans =
484
485     1     1     1     1     1
486 ans =
487     1     1     1
488 ans =
489     1     1     1
490 ans =
491     1     1     1
492 ans =
493     1     1     1
494 ans =
495     1     1     1     1
496 Check conditions Region of Saturation R4
497 ans =
498     1     1
499 ans =
500     1     1     1     1     1
501 ans =

```

```
502      1      1      1
503 ans =
504      1      1      1
505 ans =
506      1      1      1
507 ans =
508      1      1      1
509 ans =
510      1      1      1      1
511 Check conditions Region of Saturation R5
512 ans =
513      1      1
514 ans =
515      1      1      1      1      1
516 ans =
517      1      1      1
518 ans =
519      1      1      1
520 ans =
521      1      1      1
522 ans =
523      1      1      1
524 ans =
525      1      1      1      1
```

