Aula 10

21 Maio 2019

Resumo da aula passada

- Formulações de controle preditivo com garantia de factibilidade recursiva e convergência do estado para a origem
- Uso de horizonte de predição infinito e horizonte de controle finito
- Uso de restrição terminal pontual

Tópicos da aula de hoje

- Caracterização do domínio de atração da origem ao se empregar a lei de controle preditivo com restrição terminal pontual
- Alternativa para ampliação do domínio de atração: Uso de conjunto terminal

Domínio de atração

Domínio de atração

Considerando que haja garantia de factibilidade recursiva e convergência do estado para a origem, o (maior) domínio de atração pode ser definido como o conjunto das condições iniciais $x \in \mathbb{R}^n$ para as quais o problema de otimização é factível.

Pergunta: Como caracterizar tal conjunto ?

Caracterização do domínio de atração

O problema pode ser formulado da seguinte forma: Determinar o conjunto de pontos $x \in \mathcal{X} \subset \mathbb{R}^n$ para os quais existe $\hat{\mathbf{u}} \in \mathbb{R}^{pN}$ que satisfaz as seguintes restrições:

$$\begin{aligned} A_{qp}\hat{\mathbf{u}} &\leq b_{qp} \\ A_{eq}\hat{\mathbf{u}} &= b_{eq} \end{aligned}$$

com

$$b_{qp} = \begin{bmatrix} [u_{max}]_N \\ -[u_{min}]_N \\ [x_{max}]_N - \Phi_u x \\ \Phi_u x - [x_{min}]_N \end{bmatrix}$$

$$b_{eq} = -A^N x$$

e A_{qp}, A_{eq} definidas como anteriormente.

Considerando $\mathcal{X} = \{x \in \mathbb{R}^n : x_{min} \leq x \leq x_{max}\}$, a condição inicial x deve satisfazer

$$\begin{bmatrix} I_n \\ -I_n \end{bmatrix} x \leq \begin{bmatrix} x_{max} \\ -x_{min} \end{bmatrix}$$

Essa restrição, combinada com

$$\begin{bmatrix} I_{pN} \\ -I_{pN} \\ H \\ -H \end{bmatrix} \hat{\mathbf{u}} \leq \begin{bmatrix} [u_{max}]_N \\ -[u_{min}]_N \\ [x_{max}]_N - \Phi_u x \\ \Phi_u x - [x_{min}]_N \end{bmatrix}$$

pode ser reescrita como

$$\begin{bmatrix} 0_{n \times pN} & I_n \\ 0_{n \times pN} & -I_n \\ I_{pN} & 0_{pN \times n} \\ -I_{pN} & 0_{pN \times n} \\ H & \Phi_u \\ -H & -\Phi_u \end{bmatrix} \begin{bmatrix} \hat{\mathbf{u}} \\ x \end{bmatrix} \leq \begin{bmatrix} x_{max} \\ -x_{min} \\ [u_{max}]_N \\ -[u_{min}]_N \\ [x_{max}]_N \\ -[x_{min}]_N \end{bmatrix}$$

ou ainda

$$S_z z \leq b_z$$

em que

$$z = \begin{bmatrix} \hat{\mathbf{u}} \\ x \end{bmatrix}, \; S_z = \begin{bmatrix} 0_{n \times pN} & I_n \\ 0_{n \times pN} & -I_n \\ I_{pN} & 0_{pN \times n} \\ -I_{pN} & 0_{pN \times n} \\ H & \Phi_u \\ -H & -\Phi_u \end{bmatrix}, \; b_z = \begin{bmatrix} x_{max} \\ -x_{min} \\ [u_{max}]_N \\ -[u_{min}]_N \\ [x_{max}]_N \\ -[x_{min}]_N \end{bmatrix}$$

A restrição de igualdade

$$A_{eq}\hat{\mathbf{u}} = -A^N x$$

pode ser reescrita como

$$\begin{bmatrix} A_{eq} & A^N \end{bmatrix} \begin{bmatrix} \hat{\mathbf{u}} \\ x \end{bmatrix} = 0_n$$

ou ainda

$$S_{z,eq} z = 0_n$$

em que

$$S_{z,eq} = \begin{bmatrix} A_{eq} & A^N \end{bmatrix}$$

A restrição $S_z z \leq b_z$ define uma região $\mathcal{P}_z$ no espaço $\mathbb{R}^{(pN+n)}$ correspondente às variáveis $\hat{\mathbf{u}}$, x.

Considera-se aqui que tal região seja limitada em decorrência de restrições sobre a excursão dos controles e dos estados. Diz-se então que $\mathcal{P}_z$ é um **politopo**.

Politopo (Maciejowski, 2002)

Um politopo no $\mathbb{R}^n$ é uma região finita delimitada por um número finito de hiperplanos.

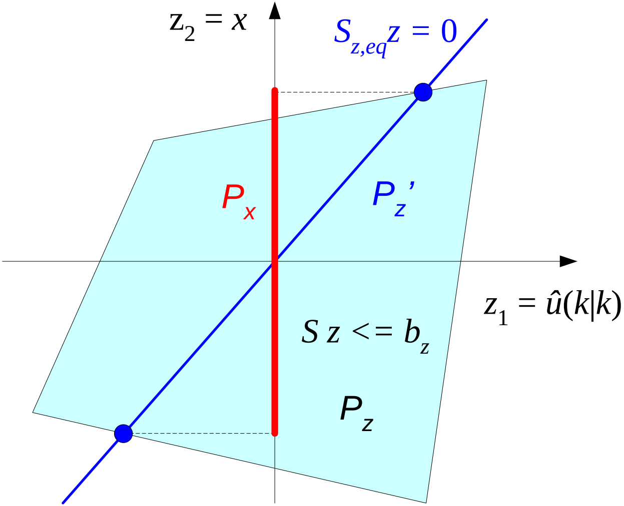
Um politopo pode ser descrito por um conjunto de desigualdades da forma $Ax \leq b$ ou por um conjunto de vértices.

A restrição $S_{z,eq}z = 0_n$ (n equações) define um **corte** no politopo inicial, resultando em um novo politopo $\mathcal{P}'_z$.

Deve-se então determinar quais os pontos $x \in \mathbb{R}^n$ para os quais existe $\hat{\mathbf{u}} \in \mathbb{R}^{pN}$ tal que $z = \begin{bmatrix} \hat{\mathbf{u}} \\ x \end{bmatrix} \in \mathcal{P}'_z$.

Para isso, pode-se projetar o politopo $\mathcal{P}'_z$ sobre as dimensões definidas por suas últimas n componentes (correspondentes a x).

Ilustração das operações envolvidas para $n = 1$, $N = 1$



Esse conjunto de operações pode ser realizado empregando ferramentas de geometria computacional.

Para ilustração, serão aqui empregadas rotinas do Multiparametric Toolbox (MPT) para Matlab (http://control.ee.ethz.ch/∼mpt/)

```
Pzlinha = Polyhedron('H',[Sz bz],'He',[Szeq zeros(n,1)]);
```

```
Px = projection(Pzlinha,p*N+1:p*N+n);
```

Rotina em Matlab para determinação do domínio de atração

- `DominioAtracao.m`

Alternativa para alargar o domínio de atração: Uso de conjunto terminal

Uso de conjunto terminal

Ideia: Ao invés de impor que o estado atinja a **origem** ao final do horizonte de predição:

$$\hat{x}(k+N|k) = 0$$

pode-se impor que o estado atinja um **conjunto terminal**:

$$\hat{x}(k+N|k) \in \mathcal{X}_f$$

sendo $\mathcal{X}_f$ um subconjunto **invariante** do conjunto de estados admissíveis $\mathcal{X}$, com $0 \in \mathcal{X}_f$.

Conjunto invariante

Considere um sistema com dinâmica descrita por

$$x(k+1) = Ax(k)$$

sendo $x(k) \in \mathbb{R}^n$ e $A \in \mathbb{R}^{n \times n}$.

Para esse sistema, um conjunto $\mathcal{X}_f$ é dito ser **positivamente invariante** (ou simplesmente "invariante") se

$$x(0) \in \mathcal{X}_f \; \Rightarrow \; x(k) \in \mathcal{X}_f, \; \forall k \geq 0$$

Vale notar que a origem é um conjunto invariante trivial.

Determinação do maior conjunto invariante contido em $\mathcal{X}$

Suponha que o conjunto $\mathcal{X}$ de estados admissíveis seja descrito na forma de desigualdades lineares:

$$\mathcal{X} = \{x \in \mathbb{R}^n \,:\, S_x x \leq b_x\}$$

Considere ainda que $\mathcal{X}$ seja limitado e contenha a origem em seu **interior**, ou seja:

$$0 < b_x$$

$$\mathcal{X} = \{x \in \mathbb{R}^n \,:\, S_x x \leq b_x\} \tag{1}$$

Seja $\mathcal{X}_1$ o conjunto de condições iniciais $x \in \mathcal{X}$ a partir das quais o estado permanece em $\mathcal{X}$ após um passo, ou seja:

$$\mathcal{X}_1 = \{x \in \mathcal{X} \,:\, Ax \in \mathcal{X}\} \tag{2}$$

De (1) e (2), tem-se

$$\mathcal{X}_1 = \{x \in \mathbb{R}^n \,:\, S_x x \leq b_x \ \text{ e } \ S_x A x \leq b_x\}$$

ou ainda:

$$\mathcal{X}_1 = \left\{ x \in \mathbb{R}^n \,:\, \begin{bmatrix} S_x \\ S_x A \end{bmatrix} x \leq \begin{bmatrix} b_x \\ b_x \end{bmatrix} \right\}$$

Seguindo desenvolvimento similar, chega-se à seguinte caracterização para o conjunto $\mathcal{X}_i$ de condições iniciais $x \in \mathcal{X}$ a partir das quais o estado permanece em $\mathcal{X}$ ao longo de i passos:

$$\begin{bmatrix} S_x \\ S_x A \\ \vdots \\ S_x A^i \end{bmatrix} x \leq \begin{bmatrix} b_x \\ b_x \\ \vdots \\ b_x \end{bmatrix}$$

Proposição 1: Se $\mathcal{X}_i = \mathcal{X}_{i+1}$ para algum i, então $\mathcal{X}_i$ é um conjunto invariante.

Prova: Inicialmente, vale notar que $\mathcal{X}_i \neq \varnothing$, pois $0 \in \mathcal{X}_i$.

Seja $x(0) \in \mathcal{X}_i = \mathcal{X}_{i+1}$. Então:

$$x(1) \in \mathcal{X},\ x(2) \in \mathcal{X}, \ldots, \ x(i+1) \in \mathcal{X}$$

Portanto, tem-se que $x(1) \in \mathcal{X}_i$. Como $\mathcal{X}_i = \mathcal{X}_{i+1}$, conclui-se ainda que $x(1) \in \mathcal{X}_{i+1}$. Logo:

$$x(2) \in \mathcal{X},\ x(3) \in \mathcal{X}, \ldots, \ x(i+2) \in \mathcal{X}$$

e, portanto, $x(2) \in \mathcal{X}_i$. Empregando esse desenvolvimento repetidas vezes, conclui-se que $x(k) \in \mathcal{X}_i$, $\forall k \geq 0$.

Proposição 1: Se $\mathcal{X}_i = \mathcal{X}_{i+1}$ para algum i, então $\mathcal{X}_i$ é um conjunto invariante.

Observação 1: Pode-se ainda mostrar que $\mathcal{X}_i$ contém todos os subconjuntos invariantes de $\mathcal{X}$.

Com efeito, seja um conjunto invariante $\mathcal{X}_{\text{inv}} \subset \mathcal{X}$. Se $x(0) \in \mathcal{X}_{\text{inv}}$, tem-se que $x(k) \in \mathcal{X}_{\text{inv}}$, $\forall k \geq 0$ e, portanto, $x(k) \in \mathcal{X}$, $\forall k \geq 0$. Trivialmente, tem-se que $x(k) \in \mathcal{X}$ para $k = 1, 2, \ldots, i$. Logo $x(0) \in \mathcal{X}_i$.

Diz-se então que $\mathcal{X}_i$ é o **maior subconjunto invariante** de $\mathcal{X}$.

Observação 2 (Determinação finita de $\mathcal{X}_i$): Consideremos novamente a caracterização de $\mathcal{X}_i$ na forma de desigualdades lineares:

$$\begin{bmatrix} S_x \\ S_x A \\ \vdots \\ S_x A^i \end{bmatrix} x \leq \begin{bmatrix} b_x \\ b_x \\ \vdots \\ b_x \end{bmatrix}$$

Por hipótese, $S_x x \leq b_x$ define um conjunto $\mathcal{X}$ limitado, contendo a origem em seu interior, ou seja $0 < b_x$.

Se o sistema for assintoticamente estável (ou seja, se todos os autovalores de A tiverem módulo menor do que um), haverá um inteiro $i+1$ para o qual a desigualdade $S_x A^{i+1} x \leq b_x$ será satisfeita para todo $x \in \mathcal{X}$.

Portanto, a restrição $S_x A^{i+1} x \leq b_x$ será redundante e, consequentemente, $\mathcal{X}_{i+1} = \mathcal{X}_i$.

Como verificar a redundância de restrições ?

Considere as seguintes restrições:

$$Sx \leq b \tag{3}$$

$$c^T x \leq d \tag{4}$$

com $x \in \mathbb{R}^n$, $S \in \mathbb{R}^{r \times n}$, $b \in \mathbb{R}^r$, $c \in \mathbb{R}^n$, $d \in \mathbb{R}$.

Como testar se a restrição (4) é redundante com respeito a (3) ?

Ideia: Resolver o seguinte problema de programação linear (PPL):

$$\max_{x \in \mathbb{R}^n} c^T x \quad \text{s.a.} \quad Sx \leq b$$

A restrição (4) será redundante se e somente se $(c^T x^* - d) \leq 0$.

Mini-tutorial: Programação linear no Matlab

Função LINPROG (Matlab Optimization Toolbox):

> `x = linprog(f,A,b)`
> minimiza `f'*x`
> sujeito a `A*x <= b`

Em nosso caso:

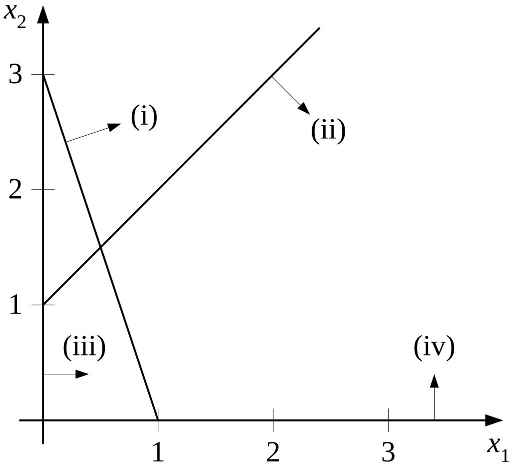
$f_{lp} = -c$
$A_{lp} = S$
$b_{lp} = b$

Uso da função LINPROG: Exemplo

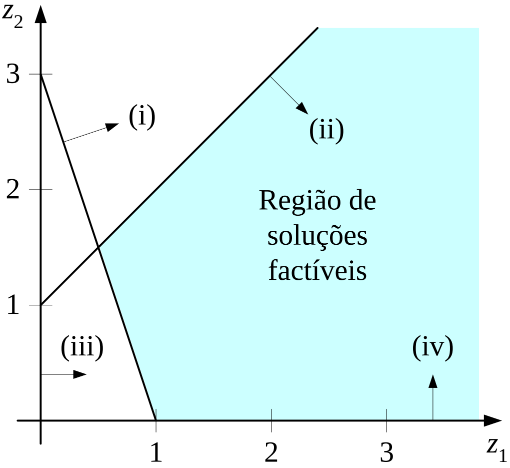
$$\min_{x \in \mathbb{R}^2} J(x) = x_1 + 2x_2$$

s.a.

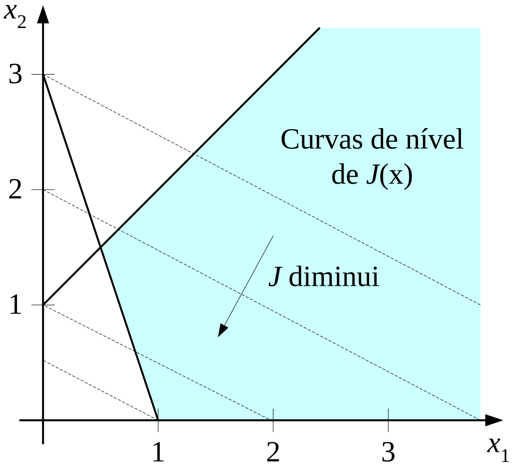
$$\begin{aligned} -3x_1 - x_2 &\leq -3 && \text{(i)} \\ -x_1 + x_2 &\leq 1 && \text{(ii)} \\ x_1 &\geq 0 && \text{(iii)} \\ x_2 &\geq 0 && \text{(iv)} \end{aligned}$$



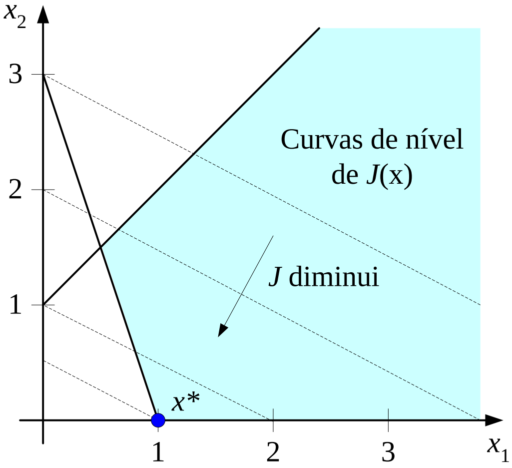
$$
\begin{aligned}
-3x_1 - x_2 &\leq -3 \quad \text{(i)} \\
-x_1 + x_2 &\leq 1 \quad \text{(ii)} \\
x_1 &\geq 0 \quad \text{(iii)} \\
x_2 &\geq 0 \quad \text{(iv)}
\end{aligned}
$$



$$
\begin{aligned}
-3x_1 - x_2 &\leq -3 && \text{(i)} \\
-x_1 + x_2 &\leq 1 && \text{(ii)} \\
x_1 &\geq 0 && \text{(iii)} \\
x_2 &\geq 0 && \text{(iv)}
\end{aligned}
$$



$$J(x) = x_1 + 2x_2$$



$$J(x) = x_1 + 2x_2$$

$$\min_{x\in\mathbb{R}^2} J(x) = x_1 + 2x_2$$

s.a.

$$\begin{aligned} -3x_1 - x_2 &\leq -3 && \text{(i)} \\ -x_1 + x_2 &\leq 1 && \text{(ii)} \\ x_1 &\geq 0 && \text{(iii)} \\ x_2 &\geq 0 && \text{(iv)} \end{aligned}$$

Usando o LINPROG:

$$f_{lp} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \quad A_{lp} = \begin{bmatrix} -3 & -1 \\ -1 & 1 \\ -1 & 0 \\ 0 & -1 \end{bmatrix}, \quad b_{lp} = \begin{bmatrix} -3 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

Implementação em Matlab

Funções a serem empregadas:

`teste_redundancia.m`

`conjunto_inv.m`

O conjunto invariante $\mathcal{X}_f$ é obtido na forma de desigualdades lineares:

$$\mathcal{X}_f = \{x \in \mathbb{R}^n \, : \, S_f x \leq b_f\}$$

Exemplo

$$A = \begin{bmatrix} 0 & 0.5 \\ -1 & 0.5 \end{bmatrix}, \quad x_{min} = \begin{bmatrix} -1 \\ -1 \end{bmatrix}, \quad x_{max} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$S_x = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 0 \\ 0 & -1 \end{bmatrix}, \quad b_x = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

Exemplo

```
>> A = [0 0.5; -1 0.5];
>> eig(A)
>> Sx = [eye(2);-eye(2)];
>> bx = ones(4,1);
>> max_iter = 10;
>> [Sf,bf] = conjunto_inv(A,Sx,bx,max_iter)
```

Visualizando o conjunto invariante com a função `plot` to MPT Toolbox:

```
>> Xf = Polyhedron('H',[Sf bf]);
>> plot(Xf)
```

Visualizando algumas trajetórias:

```
>> hold on
>> x(:,1) = [-0.5;1];
>> for k = 1:10, x(:,k+1) = A*x(:,k); end
>> plot(x(1,:),x(2,:),'-o')

>> close, plot(Xf), hold on
>> x(:,1) = [-1;0];
>> for k = 1:10, x(:,k+1) = A*x(:,k); end
>> plot(x(1,:),x(2,:),'-o')

>> close, plot(Xf), hold on
>> x(:,1) = [-0.8;0.6];
>> for k = 1:10, x(:,k+1) = A*x(:,k); end
>> plot(x(1,:),x(2,:),'-o')
```

Dual Mode Predictive Control

O que fazer se a matriz A não tiver todos os autovalores no interior do círculo unitário ?

- Custo a ser minimizado no instante k:

$$J(k) = \sum_{i=0}^{\infty} \left[||\hat{x}(k+i|k)||_Q^2 + ||\hat{u}(k+i|k)||_R^2 \right]$$

- Equações de predição:

$$\begin{aligned}
\hat{x}(k|k) &= x(k) \\
\hat{x}(k+i+1|k) &= A\hat{x}(k+i|k) + B\hat{u}(k+i|k)\,,\ i \geq 0 \\
\hat{u}(k+i|k) &= -K\hat{x}(k+i|k)\,,\ i \geq N
\end{aligned}$$

sendo K uma matriz de ganho tal que $(A - BK)$ tenha todos os autovalores no interior do círculo unitário.

- Restrições sobre os controles e estados:

$$\begin{aligned} \hat{u}(k+i|k) &\in \mathcal{U}\,, \quad i \geq 0 \\ \hat{x}(k+i|k) &\in \mathcal{X}\,, \quad i \geq 0 \end{aligned}$$

sendo $\mathcal{U} \subset \mathbb{R}^p$ e $\mathcal{X} \subset \mathbb{R}^n$ conjuntos contendo a origem em seu interior.

- Lei de controle com horizonte retrocedente:

$$u(k) = \hat{u}^*(k|k)$$

Reformulação do custo: Parcela de custo terminal

Tendo em vista que

$$\hat{u}(k+i|k) = -K\hat{x}(k+i|k)\,,\ i \geq N$$

o custo

$$J(k) = \sum_{i=0}^{\infty} \left[||\hat{x}(k+i|k)||_Q^2 + ||\hat{u}(k+i|k)||_R^2 \right]$$

pode ser reescrito como

$$J(k) = \sum_{i=0}^{N-1} \left[||\hat{x}(k+i|k)||_Q^2 + ||\hat{u}(k+i|k)||_R^2 \right] + \sum_{i=N}^{\infty} \left[||\hat{x}(k+i|k)||_Q^2 + ||\hat{u}(k+i|k)||_R^2 \right]$$

$$= \sum_{i=0}^{N-1} \left[||\hat{x}(k+i|k)||_Q^2 + ||\hat{u}(k+i|k)||_R^2 \right] + \sum_{i=N}^{\infty} \left[||\hat{x}(k+i|k)||_Q^2 + ||K\hat{x}(k+i|k)||_R^2 \right]$$

$$= \sum_{i=0}^{N-1} \left[||\hat{x}(k+i|k)||_Q^2 + ||\hat{u}(k+i|k)||_R^2 \right] + \sum_{i=N}^{\infty} ||\hat{x}(k+i|k)||_{(Q+K^TRK)}^2$$

As equações de predição

$$\begin{aligned}\hat{x}(k+i+1|k) &= A\hat{x}(k+i|k) + B\hat{u}(k+i|k)\,,\ i \geq 0 \\ \hat{u}(k+i|k) &= -K\hat{x}(k+i|k)\,,\ i \geq N\end{aligned}$$

podem ser reescritas como

$$\begin{aligned}\hat{x}(k+i+1|k) &= A\hat{x}(k+i|k) + B\hat{u}(k+i|k)\,,\ 0 \leq i \leq N-1 \\ \hat{x}(k+i+1|k) &= A_f\hat{x}(k+i|k)\,,\ i \geq N\end{aligned}$$

sendo $A_f = A - BK$.

$$
\begin{aligned}
\hat{x}(k+i+1|k) &= A\hat{x}(k+i|k) + B\hat{u}(k+i|k)\,,\; 0 \le i \le N-1 \\
\hat{x}(k+i+1|k) &= A_f\hat{x}(k+i|k)\,,\; i \ge N
\end{aligned}
$$

Portanto, o custo

$$J(k) = \sum_{i=0}^{N-1} \left[||\hat{x}(k+i|k)||_Q^2 + ||\hat{u}(k+i|k)||_R^2 \right] + \sum_{i=N}^{\infty} ||\hat{x}(k+i|k)||_{(Q+K^TRK)}^2$$

pode ser reescrito como

$$J(k) = \sum_{i=0}^{N-1} \left[||\hat{x}(k+i|k)||_Q^2 + ||\hat{u}(k+i|k)||_R^2 \right] + ||\hat{x}(k+N|k)||_{P_f}^2$$

sendo $P_f = P_f^T > 0$ obtida como solução desta equação de Lyapunov:

$$A_f^T P_f A_f - P_f + Q + K^T R K = 0$$

Reformulação do custo: Uso de conjunto terminal

Equações de predição:

$$\begin{aligned}
\hat{x}(k+i+1|k) &= A\hat{x}(k+i|k) + B\hat{u}(k+i|k)\,,\ 0 \le i \le N-1 \\
\hat{x}(k+i+1|k) &= A_f\hat{x}(k+i|k)\,,\ i \ge N \\
\hat{u}(k+i|k) &= -K\hat{x}(k+i|k)\,,\ i \ge N
\end{aligned}$$

Restrições sobre os controles e estados:

$$\begin{aligned}
\hat{u}(k+i|k) &\in \mathcal{U}\,,\quad i \ge 0 \\
\hat{x}(k+i|k) &\in \mathcal{X}\,,\quad i \ge 0
\end{aligned}$$

Considere o conjunto $\mathcal{X}_{xu}$ definido como

$$\mathcal{X}_{xu} = \{x \in \mathcal{X} \,:\, -Kx \in \mathcal{U}\}$$

Seja ainda $\mathcal{X}_f$ o maior subconjunto invariante de $\mathcal{X}_{xu}$ considerando a dinâmica terminal $x(k+1) = A_f x(k)$.

As restrições sobre os controles e estados

$$\begin{aligned} \hat{u}(k+i|k) &\in \mathcal{U}\,, \quad i \geq 0 \\ \hat{x}(k+i|k) &\in \mathcal{X}\,, \quad i \geq 0 \end{aligned}$$

podem ser reformuladas como

$$\begin{aligned} \hat{u}(k+i|k) &\in \mathcal{U}\,, \;\; 0 \leq i \leq N-1 \\ \hat{x}(k+i|k) &\in \mathcal{X}\,, \;\; 0 \leq i \leq N-1 \\ \hat{x}(k+N|k) &\in \mathcal{X}_f \end{aligned}$$

Observação: Se os conjuntos $\mathcal{X}$ e $\mathcal{U}$ estiverem descritos na forma

$$\mathcal{X} = \{x \in \mathbb{R}^n \, : \, S_x x \leq b_x\}$$

$$\mathcal{U} = \{u \in \mathbb{R}^p \, : \, S_u u \leq b_u\}$$

então

$$\mathcal{X}_{xu} = \{x \in \mathbb{R}^n \, : \, S_{xu} x \leq b_{xu}\}$$

com

$$S_{xu} = \begin{bmatrix} S_x \\ -S_u K \end{bmatrix}, \quad b_{xu} = \begin{bmatrix} b_x \\ b_u \end{bmatrix}$$

Factibilidade recursiva

Seja

$$\hat{u}^*(k|k), \hat{u}^*(k+1|k), \ldots, \hat{u}^*(k+N-1|k)$$

a sequência ótima de controle obtida como solução do problema de otimização no instante k. Seja ainda

$$\hat{x}^*(k|k), \hat{x}^*(k+1|k), \ldots$$

a respectiva sequência de estados preditos, lembrando que

$$\hat{x}^*(k|k) = x(k)$$

Aplicando-se à planta o controle $u(k) = \hat{u}^*(k|k)$, segue que

$$x(k+1) = Ax(k) + Bu(k) = A\hat{x}^*(k|k) + B\hat{u}^*(k|k) = \hat{x}^*(k+1|k)$$

No instante $(k+1)$, deve-se encontrar uma sequência de controle

$$\hat{u}(k+1|k+1), \hat{u}(k+2|k+1), \ldots, \hat{u}(k+N|k+1)$$

satisfazendo a restrição

$$\hat{u}(k+i+1|k+1) \in \mathcal{U}\,, \quad 0 \leq i \leq N-1$$

e tal que a sequência de estados preditos por meio de

$$\begin{aligned}
\hat{x}(k+1|k+1) &= x(k+1) \\
\hat{x}(k+i+2|k+1) &= A\hat{x}(k+i+1|k+1) + B\hat{u}(k+i+1|k+1)\,,\ 0 \leq i \leq N-1 \\
\hat{x}(k+i+2|k+1) &= A_f\hat{x}(k+i+1|k+1)\,,\ i \geq N
\end{aligned}$$

satisfaça as restrições

$$\hat{x}(k+i+1|k+1) \in \mathcal{X}\,, \quad 0 \leq i \leq N-1$$

$$\hat{x}(k+N+1|k+1) \in \mathcal{X}_f$$

Vamos mostrar que essas restrições são satisfeitas empregando-se a seguinte **solução candidata**:

$$
\begin{aligned}
\hat{u}^c(k+1|k+1) &= \hat{u}^*(k+1|k) \\
\hat{u}^c(k+2|k+1) &= \hat{u}^*(k+2|k) \\
&\vdots \\
\hat{u}^c(k+N-1|k+1) &= \hat{u}^*(k+N-1|k) \\
\hat{u}^c(k+N|k+1) &= -K\hat{x}^*(k+N|k)
\end{aligned}
$$

$$
\begin{aligned}
\hat{u}^c(k+1|k+1) &= \hat{u}^*(k+1|k) \\
\hat{u}^c(k+2|k+1) &= \hat{u}^*(k+2|k) \\
&\vdots \\
\hat{u}^c(k+N-1|k+1) &= \hat{u}^*(k+N-1|k) \\
\hat{u}^c(k+N|k+1) &= -K\hat{x}^*(k+N|k)
\end{aligned}
$$

Essa solução trivialmente satisfaz as restrições de controle, uma vez que

$$
\begin{aligned}
\hat{u}^*(k+1|k) &\in \mathcal{U} \\
\hat{u}^*(k+2|k) &\in \mathcal{U} \\
&\vdots \\
\hat{u}^*(k+N-1|k) &\in \mathcal{U}
\end{aligned}
$$

e $\hat{x}^*(k+N|k) \in \mathcal{X}_f \subset \mathcal{X}_{xu}$.

A sequência de estados preditos associada à solução candidata é dada por

$$\begin{aligned}
\hat{x}^c(k+1|k+1) &= x(k+1) \\
\hat{x}^c(k+2|k+1) &= A\hat{x}^c(k+1|k+1) + B\hat{u}^c(k+1|k+1) \\
&\vdots \\
\hat{x}^c(k+N|k+1) &= A\hat{x}^c(k+N-1|k+1) + B\hat{u}^c(k+N-1|k+1) \\
\hat{x}^c(k+N+1|k+1) &= A\hat{x}^c(k+N|k+1) + B\hat{u}^c(k+N|k+1)
\end{aligned}$$

Lembrando ainda que $x(k+1) = \hat{x}^*(k+1|k)$, tem-se

$$
\begin{aligned}
\hat{x}^c(k+1|k+1) &= \hat{x}^*(k+1|k) \\
\hat{x}^c(k+2|k+1) &= A\hat{x}^c(k+1|k+1) + B\hat{u}^*(k+1|k) \\
&\vdots \\
\hat{x}^c(k+N|k+1) &= A\hat{x}^c(k+N-1|k+1) + B\hat{u}^*(k+N-1|k) \\
\hat{x}^c(k+N+1|k+1) &= A\hat{x}^c(k+N|k+1) - BK\hat{x}^*(k+N|k)
\end{aligned}
$$

chegando-se à sequência

$$
\begin{aligned}
\hat{x}^c(k+1|k+1) &= \hat{x}^*(k+1|k) \in \mathcal{X} \\
\hat{x}^c(k+2|k+1) &= \hat{x}^*(k+2|k) \in \mathcal{X} \\
&\vdots \\
\hat{x}^c(k+N|k+1) &= \hat{x}^*(k+N|k) \in \mathcal{X} \\
\hat{x}^c(k+N+1|k+1) &= A\hat{x}^*(k+N|k) - BK\hat{x}^*(k+N|k) \\
&= A_f\hat{x}^*(k+N|k)
\end{aligned}
$$

Como $\hat{x}^*(k+N|k) \in \mathcal{X}_f$ e $\mathcal{X}_f$ é invariante sob a dinâmica terminal, conclui-se que $\hat{x}^c(k+N+1|k+1) \in \mathcal{X}_f \subset \mathcal{X}$.

Convergência do estado para a origem

Seja

$$J^*(k) = \sum_{i=0}^{N-1} \left[||\hat{x}^*(k+i|k)||_Q^2 + ||\hat{u}^*(k+i|k)||_R^2 \right] + ||\hat{x}^*(k+N|k)||_{P_f}^2$$

o valor mínimo do custo resultante da solução do problema de otimização no instante k. Seja ainda

$$J^*(k+1) = \sum_{i=0}^{N-1} \left[||\hat{x}^*(k+i+1|k+1)||_Q^2 + ||\hat{u}^*(k+i+1|k+1)||_R^2 \right]$$
$$+||\hat{x}^*(k+N+1|k+1)||_{P_f}^2$$

Como visto anteriormente, as sequências de estado e controle dadas por

$$
\begin{aligned}
\hat{u}^c(k+1|k+1) &= \hat{u}^*(k+1|k) \\
\hat{u}^c(k+2|k+1) &= \hat{u}^*(k+2|k) \\
&\vdots \\
\hat{u}^c(k+N-1|k+1) &= \hat{u}^*(k+N-1|k) \\
\hat{u}^c(k+N|k+1) &= -K\hat{x}^*(k+N|k)
\end{aligned}
\qquad\Bigg|\qquad
\begin{aligned}
\hat{x}^c(k+1|k+1) &= \hat{x}^*(k+1|k) \\
\hat{x}^c(k+2|k+1) &= \hat{x}^*(k+2|k) \\
&\vdots \\
\hat{x}^c(k+N|k+1) &= \hat{x}^*(k+N|k) \\
\hat{x}^c(k+N+1|k+1) &= A_f\hat{x}^*(k+N|k)
\end{aligned}
$$

formam uma solução factível (não necessariamente ótima) para o problema de otimização no instante $k+1$.

Logo:

$$J^*(k+1) \leq \sum_{i=0}^{N-1} \left[||\hat{x}^c(k+i+1|k+1)||_Q^2 + ||\hat{u}^c(k+i+1|k+1)||_R^2 \right] + ||\hat{x}^c(k+N+1|k+1)||_{P_f}^2$$

$$= \sum_{i=0}^{N-2} \left[||\hat{x}^*(k+i+1|k)||_Q^2 + ||\hat{u}^*(k+i+1|k)||_R^2 \right] + ||\hat{x}^*(k+N|k)||_Q^2 + ||-K\hat{x}^*(k+N|k)||_R^2 + ||A_f\hat{x}^*(k+N|k)||_{P_f}^2$$

$$= J^*(k) - ||\hat{x}^*(k|k)||_Q^2 - ||\hat{u}^*(k|k)||_R^2 - ||\hat{x}^*(k+N|k)||_{P_f}^2 + ||\hat{x}^*(k+N|k)||_Q^2 + ||-K\hat{x}^*(k+N|k)||_R^2 + ||A_f\hat{x}^*(k+N|k)||_{P_f}^2$$

Lembrando que $\hat{x}^*(k|k) = x(k)$ e $\hat{u}^*(k|k) = u(k)$, tem-se

$$J^*(k+1) \leq J^*(k) - ||x(k)||_Q^2 - ||u(k)||_R^2 + ||\hat{x}^*(k+N|k)||^2_{(-P_f+Q+K^TRK+A_f^TP_fA_f)}$$

$$J^*(k+1) \leq J^*(k) - ||x(k)||_Q^2 - ||u(k)||_R^2 + ||\hat{x}^*(k+N|k)||^2_{(-P_f + Q + K^T RK + A_f^T P_f A_f)}$$

Lembrando que a matriz P_f satisfaz

$$A_f^T P_f A_f - P_f + Q + K^T RK = 0$$

tem-se que

$$J^*(k+1) \leq J^*(k) - ||x(k)||_Q^2 - ||u(k)||_R^2$$

Portanto, seguindo desenvolvimento similar ao utilizado na aula passada, conclui-se que

$$u(k) \stackrel{k\to\infty}{\longrightarrow} 0$$

$$x(k) \stackrel{k\to\infty}{\longrightarrow} 0$$

Escolha do ganho K para a lei de controle terminal

Uma escolha natural para o ganho K é a solução do problema LQR de horizonte infinito, sem restrições, ou seja:

$$K = (R + B^T PB)^{-1} B^T PA$$

sendo $P = P^T > 0$ obtida como solução da seguinte Equação Algébrica de Riccati:

$$P = A^T PA - A^T PB(R + B^T PB)^{-1} B^T PA + Q$$

Pode-se mostrar que P corresponde à matriz P_f a ser usada no custo terminal.

$$K = (R + B^T PB)^{-1} B^T PA$$

$$P = A^T PA - A^T PB(R + B^T PB)^{-1} B^T PA + Q$$

Pode-se mostrar que P corresponde à matriz P_f a ser usada no custo terminal. Com efeito:

$$A_f^T PA_f - P + Q + K^T RK =$$

$$(A - BK)^T P(A - BK) - P + Q + K^T RK =$$

$$A^T PA - K^T B^T PA - A^T PBK + K^T B^T PBK - P + Q + K^T RK =$$

$$A^T PA - A^T PB(R + B^T PB)^{-1} B^T PA - A^T PB(R + B^T PB)^{-1} B^T PA$$
$$+ A^T PB(R + B^T PB)^{-1}(R + B^T PB)(R + B^T PB)^{-1} B^T PA - P + Q =$$

$$A^T PA - A^T PB(R + B^T PB)^{-1} B^T PA - P + Q = 0$$

Expressando a restrição terminal em termos de $\hat{\mathbf{u}}$

A restrição terminal é da forma

$$S_f \hat{x}(k+N|k) \leq b_f$$

com

$$\hat{x}(k+N|k) = \underbrace{\begin{bmatrix} A^{N-1}B & A^{N-2}B & \cdots & B \end{bmatrix}}_{H_N} \hat{\mathbf{u}} + A^N x(k)$$

Portanto, pode-se expressar a restrição terminal em termos de $\hat{\mathbf{u}}$ como

$$S_f \left[H_N \hat{\mathbf{u}} + A^N x(k) \right] \leq b_f$$

ou seja:

$$S_f H_N \hat{\mathbf{u}} \leq b_f - S_f A^N x(k)$$

Resumo

Informação requerida sobre a planta:

- Matrizes $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times p}$ do modelo no espaço de estados
- Limitantes sobre a excursão dos estados: $x_{min}, x_{max} \in \mathbb{R}^n$
- Limitantes sobre a excursão dos controles: $u_{min}, u_{max} \in \mathbb{R}^p$

Parâmetros de projeto:

- Matrizes de peso $Q \in \mathbb{R}^{n \times n}$ e $R \in \mathbb{R}^{p \times p}$
- Horizonte de predição N

Inicialização:

- Obter a matriz de ganho K como solução do problema DLQR de horizonte infinito sem restrições:

$$P = A^T PA - A^T PB(R + B^T PB)^{-1} B^T PA + Q$$

$$K = (R + B^T PB)^{-1} B^T PA$$

- Fazer $P_f = P$ e $A_f = A - BK$.
- Fazer

$$S_{xu} = \begin{bmatrix} I_n \\ -I_n \\ -K \\ K \end{bmatrix}, \quad b_{xu} = \begin{bmatrix} x_{max} \\ -x_{min} \\ u_{max} \\ -u_{min} \end{bmatrix}$$

- Caracterizar o conjunto terminal invariante $\mathcal{X}_f$ na forma $S_f x \leq b_f$.

- Fazer

$$H = \begin{bmatrix} B & 0 & \cdots & 0 \\ AB & B & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N-1}B & A^{N-2}B & \cdots & B \end{bmatrix}, \quad \Phi_u = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^N \end{bmatrix}$$

$$H_N = \begin{bmatrix} A^{N-1}B & A^{N-2}B & \cdots & B \end{bmatrix}$$

$$\mathbf{Q} = \begin{bmatrix} Q & 0 & \cdots & 0 \\ 0 & Q & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & P_f \end{bmatrix}_{qN \times qN}, \quad \mathbf{R} = \begin{bmatrix} R & 0 & \cdots & 0 \\ 0 & R & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & R \end{bmatrix}_{pN \times pN}$$

$$H_n = \mathbf{Q}H \,, \quad H_{qp} = 2(H^T \mathbf{Q} H + \mathbf{R})$$

- Fazer $A_{qp} = \begin{bmatrix} I_{pN} \\ -I_{pN} \\ H \\ -H \\ S_f H_N \end{bmatrix}, \quad S_{fA} = S_f A^N$

- Fazer $k = 0$

Rotina principal:

1. Ler $x(k)$ (estado da planta)
2. Calcular $\mathbf{f_u} = \Phi_u x(k)$ e $f_{qp} = 2H_n^T \mathbf{f_u}$
3. Fazer

$$b_{qp} = \begin{bmatrix} [u_{max}]_N \\ -[u_{min}]_N \\ [x_{max}]_N - \mathbf{f_u} \\ \mathbf{f_u} - [x_{min}]_N \\ b_f - S_{fA}x(k) \end{bmatrix}$$

4. Resolver o problema de otimização

$$\hat{\mathbf{u}}^* = \arg \min_{\hat{\mathbf{u}} \in \mathbb{R}^{pM}} \frac{1}{2} \hat{\mathbf{u}}^T H_{qp} \hat{\mathbf{u}} + f_{qp}^T \hat{\mathbf{u}}$$

s.a.

$$A_{qp} \hat{\mathbf{u}} \leq b_{qp}$$

5. Atualizar o controle aplicado à planta: $u(k) = \hat{u}^*(k|k)$
6. Fazer $k = k + 1$
7. Aguardar o próximo instante de amostragem e retornar ao passo 1.

Caracterização do domínio de atração

O domínio de atração corresponde ao conjunto de pontos $x \in \mathcal{X} \subset \mathbb{R}^n$ para os quais existe $\hat{\mathbf{u}} \in \mathbb{R}^{pN}$ que satisfaz as seguintes restrições:

$$A_{qp}\hat{\mathbf{u}} \quad \leq \quad b_{qp}$$

com

$$A_{qp} = \begin{bmatrix} I_{pN} \\ -I_{pN} \\ H \\ -H \\ S_f H_N \end{bmatrix} , \quad b_{qp} = \begin{bmatrix} [u_{max}]_N \\ -[u_{min}]_N \\ [x_{max}]_N - \Phi_u x \\ \Phi_u x - [x_{min}]_N \\ b_f - S_{fA} x \end{bmatrix}$$

As restrições

$$\begin{bmatrix} I_n \\ -I_n \end{bmatrix} x \leq \begin{bmatrix} x_{max} \\ -x_{min} \end{bmatrix}, \quad \begin{bmatrix} I_{pN} \\ -I_{pN} \\ H \\ -H \\ S_f H_N \end{bmatrix} \hat{\mathbf{u}} \leq \begin{bmatrix} [u_{max}]_N \\ -[u_{min}]_N \\ [x_{max}]_N - \Phi_u x \\ \Phi_u x - [x_{min}]_N \\ b_f - S_{fA} x \end{bmatrix}$$

podem ser reescritas como

$$\begin{bmatrix} 0_{n\times pN} & I_n \\ 0_{n\times pN} & -I_n \\ I_{pN} & 0_{pN\times n} \\ -I_{pN} & 0_{pN\times n} \\ H & \Phi_u \\ -H & -\Phi_u \\ S_f H_N & S_{fA} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{u}} \\ x \end{bmatrix} \leq \begin{bmatrix} x_{max} \\ -x_{min} \\ [u_{max}]_N \\ -[u_{min}]_N \\ [x_{max}]_N \\ -[x_{min}]_N \\ b_f \end{bmatrix}$$

ou ainda

$$S_z z \leq b_z$$

em que

$$z = \begin{bmatrix} \hat{\mathbf{u}} \\ x \end{bmatrix}, \; S_z = \begin{bmatrix} 0_{n \times pN} & I_n \\ 0_{n \times pN} & -I_n \\ I_{pN} & 0_{pN \times n} \\ -I_{pN} & 0_{pN \times n} \\ H & \Phi_u \\ -H & -\Phi_u \\ S_f H_N & S_{fA} \end{bmatrix}, \; b_z = \begin{bmatrix} x_{max} \\ -x_{min} \\ [u_{max}]_N \\ -[u_{min}]_N \\ [x_{max}]_N \\ -[x_{min}]_N \\ b_f \end{bmatrix}$$

A restrição $S_z z \leq b_z$ define uma região $\mathcal{P}_z$ no espaço $\mathbb{R}^{(pN+n)}$ correspondente às variáveis $\hat{\mathbf{u}}$, x.

Considera-se que tal região seja limitada em decorrência de restrições sobre a excursão dos controles e dos estados. Como resultado, $\mathcal{P}_z$ é um **politopo**.

O domínio de atração é obtido projetando-se $\mathcal{P}_z$ sobre as dimensões definidas por suas últimas n componentes (correspondentes a x).

Rotina em Matlab para determinação do domínio de atração

- `DominioAtracaoConjTerm.m`

Resumo da aula de hoje

- Caracterização do domínio de atração da origem ao se empregar a lei de controle preditivo com restrição terminal pontual.
- Alternativa para ampliação do domínio de atração: Uso de conjunto terminal.

Tópicos da próxima aula

- Gerenciamento de problemas de factibilidade