

Aula 7

30 Abril 2019

Resumo da aula passada

- Estimação de perturbações empregando observador de estados
- Determinação de valores de equilíbrio para o estado e o controle

Tópicos da aula de hoje

- Tratamento de restrições - caso SISO

Restrições a serem consideradas

Três tipos básicos de restrições serão considerados:

- Incrementos no controle Δu
- Excursão do controle u
- Excursão da saída y

Como se verá, todas essas restrições podem ser expressas em termos de restrições sobre $\Delta \hat{u}$.

Restrições sobre os incrementos no controle Δu

$$\Delta u_{min} \leq \Delta \hat{u}(k+i-1|k) \leq \Delta u_{max}, \quad i = 1, 2, \dots, M$$

Vale ressaltar que Δu_{min} e Δu_{max} são limitantes para o incremento no controle **por período de amostragem**.

Ex: Suponha que o controle u corresponda a uma deflexão medida em graus e que a taxa de variação de u esteja limitada a $\pm 10^\circ / \text{s}$.

Se o período de amostragem for $T = 10 \text{ ms}$, quais serão os valores de Δu_{min} e Δu_{max} ?

Resp: $\Delta u_{min} = -0,1^\circ$ e $\Delta u_{max} = +0,1^\circ$.

Notação alternativa:

$$\overbrace{\begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}}^{1_M} \Delta u_{min} \leq \overbrace{\begin{bmatrix} \Delta \hat{u}(k|k) \\ \Delta \hat{u}(k+1|k) \\ \vdots \\ \Delta \hat{u}(k+M-1|k) \end{bmatrix}}^{\Delta \hat{u}} \leq \overbrace{\begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}}^{1_M} \Delta u_{max}$$

$$1_M \Delta u_{min} \leq \Delta \hat{u} \leq 1_M \Delta u_{max}$$

Vale ressaltar que o símbolo \leq é aqui empregado para denotar uma desigualdade **elemento a elemento**.

$$1_M \Delta u_{min} \leq \Delta \hat{\mathbf{u}} \leq 1_M \Delta u_{max}$$

$$\begin{cases} \Delta \hat{\mathbf{u}} & \leq & 1_M \Delta u_{max} \\ -\Delta \hat{\mathbf{u}} & \leq & -1_M \Delta u_{min} \end{cases}$$

$$\overbrace{\begin{bmatrix} I_M \\ -I_M \end{bmatrix}}^{2M \times M} \Delta \hat{\mathbf{u}} \leq \overbrace{\begin{bmatrix} 1_M \Delta u_{max} \\ -1_M \Delta u_{min} \end{bmatrix}}^{2M \times 1}$$

Restrições sobre a excursão do controle u

$$u_{min} \leq \hat{u}(k+i-1|k) \leq u_{max}, \quad i = 1, 2, \dots, M$$

Obs: Se o modelo tiver sido linearizado em torno de um valor de equilíbrio \bar{u}_p para o controle (vide Aula 3), os limitantes u_{max} e u_{min} corresponderão a diferenças com respeito a \bar{u}_p .

Ex: Suponha que o controle seja gerado por uma fonte de tensão que satura em $u_{p,min} = -5 \text{ V}$ e $u_{p,max} = +5 \text{ V}$.

Se a operação se der em torno de um valor de equilíbrio $\bar{u}_p = +3 \text{ V}$, quais serão os valores de u_{min} e u_{max} a serem empregados nas restrições ?

Resp: $u_{min} = u_{p,min} - \bar{u}_p = -8 \text{ V}$ e $u_{max} = u_{p,max} - \bar{u}_p = +2 \text{ V}$.

- Notação alternativa:

$$1_M u_{min} \leq \hat{\mathbf{u}} \leq 1_M u_{max}$$

- Para expressar estas restrições em termos de $\Delta\hat{\mathbf{u}}$, deve-se obter uma relação entre $\hat{\mathbf{u}}$ e $\Delta\hat{\mathbf{u}}$.

$$\hat{u}(k|k) = u(k-1) + \Delta \hat{u}(k|k)$$

$$\begin{aligned}\hat{u}(k+1|k) &= \hat{u}(k|k) + \Delta \hat{u}(k+1|k) \\ &= u(k-1) + \Delta \hat{u}(k|k) + \Delta \hat{u}(k+1|k)\end{aligned}$$

$$\vdots$$

$$\hat{u}(k+M-1|k) = u(k-1) + \Delta \hat{u}(k|k) + \cdots + \Delta \hat{u}(k+M-1|k)$$

$$\begin{bmatrix} \hat{u}(k|k) \\ \hat{u}(k+1|k) \\ \vdots \\ \hat{u}(k+M-1|k) \end{bmatrix} = \begin{bmatrix} u(k-1) \\ u(k-1) \\ \vdots \\ u(k-1) \end{bmatrix} + \\
 + \begin{bmatrix} \Delta \hat{u}(k|k) \\ \Delta \hat{u}(k|k) + \Delta \hat{u}(k+1|k) \\ \vdots \\ \Delta \hat{u}(k|k) + \Delta \hat{u}(k+1|k) + \cdots + \Delta \hat{u}(k+M-1|k) \end{bmatrix}$$

$$\begin{aligned}
 & \overbrace{\begin{bmatrix} \hat{u}(k|k) \\ \hat{u}(k+1|k) \\ \vdots \\ \hat{u}(k+M-1|k) \end{bmatrix}}^{\hat{\mathbf{u}}(M \times 1)} = \overbrace{\begin{bmatrix} u(k-1) \\ u(k-1) \\ \vdots \\ u(k-1) \end{bmatrix}}^{1_M u(k-1)} \\
 & + \underbrace{\begin{bmatrix} 1 & 0 & \cdots & 0 \\ 1 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1 \end{bmatrix}}_{T_M(M \times M)} \underbrace{\begin{bmatrix} \Delta \hat{u}(k|k) \\ \Delta \hat{u}(k+1|k) \\ \vdots \\ \Delta \hat{u}(k+M-1|k) \end{bmatrix}}_{\Delta \hat{\mathbf{u}}(M \times 1)}
 \end{aligned}$$

$$\hat{\mathbf{u}} = 1_M u(k-1) + T_M \Delta \hat{\mathbf{u}}$$

$$1_M u_{min} \leq \hat{\mathbf{u}} \leq 1_M u_{max}$$

$$1_M u_{min} \leq 1_M u(k-1) + T_M \Delta \hat{\mathbf{u}} \leq 1_M u_{max}$$

$$1_M [u_{min} - u(k-1)] \leq T_M \Delta \hat{\mathbf{u}} \leq 1_M [u_{max} - u(k-1)]$$

$$\begin{cases} T_M \Delta \hat{\mathbf{u}} & \leq 1_M [u_{max} - u(k-1)] \\ -T_M \Delta \hat{\mathbf{u}} & \leq 1_M [u(k-1) - u_{min}] \end{cases}$$

$$\overbrace{\begin{bmatrix} T_M \\ -T_M \end{bmatrix}}^{2M \times M} \Delta \hat{\mathbf{u}} \leq \overbrace{\begin{bmatrix} 1_M [u_{max} - u(k-1)] \\ 1_M [u(k-1) - u_{min}] \end{bmatrix}}^{2M \times 1}$$

Equívoco a ser evitado

Uma restrição da forma

$$T_M \Delta \hat{\mathbf{u}} \leq \mathbf{1}_M [u_{\max} - u(k-1)]$$

não é equivalente a

$$\Delta \hat{\mathbf{u}} \leq T_M^{-1} \mathbf{1}_M [u_{\max} - u(k-1)]$$

Exemplo: $M = 2$, $u(k-1) = 0$, $u_{max} = 1$

$$T_M = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}, \quad T_M^{-1} = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix}$$

Forma **correta** de se escrever a restrição:

$$T_M \Delta \hat{\mathbf{u}} \leq \mathbf{1}_M [u_{max} - u(k-1)]$$

$$\begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} \Delta \hat{u}(k|k) \\ \Delta \hat{u}(k+1|k) \end{bmatrix} \leq \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$\begin{cases} \Delta \hat{u}(k|k) \leq 1 \\ \Delta \hat{u}(k|k) + \Delta \hat{u}(k+1|k) \leq 1 \end{cases}$$

Exemplo: $M = 2$, $u(k-1) = 0$, $u_{max} = 1$

Forma **correta** de se escrever a restrição:

$$\begin{cases} \Delta \hat{u}(k|k) \leq 1 \\ \Delta \hat{u}(k|k) + \Delta \hat{u}(k+1|k) \leq 1 \end{cases}$$

Forma **incorreta** de se escrever a restrição:

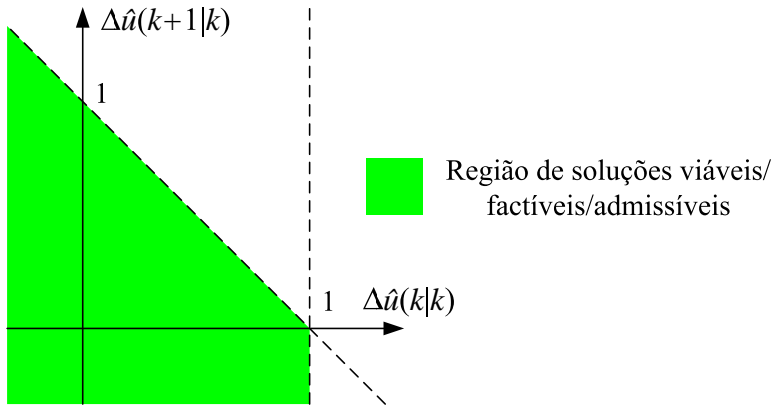
$$\Delta \hat{\mathbf{u}} \leq T_M^{-1} \mathbf{1}_M [u_{max} - u(k-1)]$$

$$\begin{bmatrix} \Delta \hat{u}(k|k) \\ \Delta \hat{u}(k+1|k) \end{bmatrix} \leq \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$\begin{cases} \Delta \hat{u}(k|k) \leq 1 \\ \Delta \hat{u}(k+1|k) \leq 0 \end{cases}$$

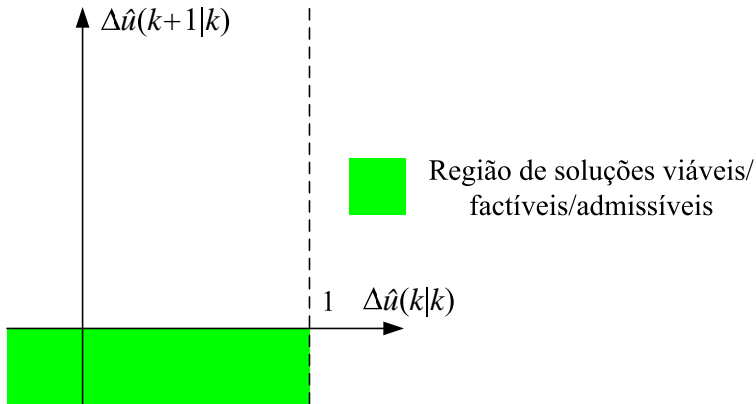
Forma **correta** de se escrever a restrição:

$$\begin{cases} \Delta \hat{u}(k|k) \leq 1 \\ \Delta \hat{u}(k|k) + \Delta \hat{u}(k+1|k) \leq 1 \end{cases}$$



Forma **incorreta** de se escrever a restrição:

$$\begin{cases} \Delta \hat{u}(k|k) \leq 1 \\ \Delta \hat{u}(k+1|k) \leq 0 \end{cases}$$



Restrições sobre a excursão da saída y

$$y_{min} \leq \hat{y}(k+i|k) \leq y_{max}, \quad i = 1, 2, \dots, N$$

Obs: Se o modelo tiver sido linearizado em torno de um valor de equilíbrio \bar{y}_p para a saída (vide Aula 3), os limitantes y_{max} e y_{min} corresponderão a diferenças com respeito a \bar{y}_p , isto é

$$y_{min} = y_{p,min} - \bar{y}_p$$

$$y_{max} = y_{p,max} - \bar{y}_p$$

Notação alternativa:



$$1_{Ny_{min}} \leq \hat{\mathbf{y}} \leq 1_{Ny_{max}}$$

Lembrando que $\hat{\mathbf{y}} = G\Delta\hat{\mathbf{u}} + \mathbf{f}$, a restrição pode ser reescrita como

$$1_{Ny_{min}} \leq G\Delta\hat{\mathbf{u}} + \mathbf{f} \leq 1_{Ny_{max}}$$

$$1_{Ny_{min}} - \mathbf{f} \leq G\Delta\hat{\mathbf{u}} \leq 1_{Ny_{max}} - \mathbf{f}$$

$$\begin{cases} G\Delta\hat{\mathbf{u}} & \leq 1_{Ny_{max}} - \mathbf{f} \\ -G\Delta\hat{\mathbf{u}} & \leq \mathbf{f} - 1_{Ny_{min}} \end{cases}$$

$$\overbrace{\begin{bmatrix} G \\ -G \end{bmatrix}}^{2N \times M} \Delta\hat{\mathbf{u}} \leq \overbrace{\begin{bmatrix} 1_{Ny_{max}} - \mathbf{f} \\ \mathbf{f} - 1_{Ny_{min}} \end{bmatrix}}^{2N \times 1}$$

Resumo das restrições

$$\begin{array}{ccc}
 \overbrace{S[(4M+2N) \times M]} & & \overbrace{b[(4M+2N) \times 1]} \\
 \left[\begin{array}{c} I_M \\ -I_M \\ T_M \\ -T_M \\ G \\ -G \end{array} \right] & \Delta \hat{\mathbf{u}} \leq & \left[\begin{array}{c} 1_M \Delta u_{max} \\ -1_M \Delta u_{min} \\ 1_M [u_{max} - u(k-1)] \\ 1_M [u(k-1) - u_{min}] \\ 1_N y_{max} - \mathbf{f} \\ \mathbf{f} - 1_N y_{min} \end{array} \right] \\
 & & \boxed{S \Delta \hat{\mathbf{u}} \leq b}
 \end{array}$$

Formulação do problema de otimização com restrições

O problema de otimização na presença das restrições consideradas pode ser formulado como

$$\min_{\Delta \hat{\mathbf{u}} \in \mathbb{R}^M} J(\Delta \hat{\mathbf{u}}) = \frac{1}{2} \Delta \hat{\mathbf{u}}^T \mathcal{H} \Delta \hat{\mathbf{u}} + \mathbf{c}^T \Delta \hat{\mathbf{u}} + \text{cte}$$

sujeito a

$$\mathbf{S} \Delta \hat{\mathbf{u}} \leq \mathbf{b}$$

- Função de custo quadrática com $\mathcal{H} > 0$
- Restrições lineares

Problema de Programação Quadrática

Pode-se mostrar que este é um problema de otimização **convexo**.

Convexidade

Para mais detalhes, vide Capítulos 2, 3 e 4 da seguinte referência:

Boyd, S.; Vandenberghe, L. Convex Optimization. Cambridge University Press, 2004.

Conjuntos convexos

Um conjunto \mathcal{C} é dito ser **convexo** se, para quaisquer $x, y \in \mathcal{C}$ e qualquer escalar λ tal que $0 \leq \lambda \leq 1$, tem-se

$$\lambda x + (1 - \lambda)y \in \mathcal{C}$$

Conjunto definido por intersecção de semiespaços

Pode-se mostrar que um conjunto \mathcal{C} não vazio definido como

$$\mathcal{C} = \left\{ x \in \mathbb{R}^M \mid Sx \leq b \right\}$$

é convexo.

Com efeito, dados $x, y \in \mathcal{C}$ e $\lambda \in [0, 1]$, tem-se

$$S[\lambda x + (1 - \lambda)y] = \lambda Sx + (1 - \lambda)Sy \leq \lambda b + (1 - \lambda)b = b$$

Portanto $\lambda x + (1 - \lambda)y \in \mathcal{C}$.

Funções convexas

Uma função $f : \mathcal{C} \rightarrow \mathbb{R}$ é dita ser **convexa** se \mathcal{C} for um conjunto convexo e

$$f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y)$$

para quaisquer $x, y \in \mathcal{C}$ e $\lambda \in [0, 1]$.

A função f é dita ser **estritamente convexa** se

$$f(\lambda x + (1 - \lambda)y) < \lambda f(x) + (1 - \lambda)f(y)$$

sempre que $x \neq y$ e $0 < \lambda < 1$.

Função quadrática

Pode-se mostrar que uma função $f : \mathcal{C} \rightarrow \mathbb{R}$ da forma

$$f(x) = x^T Q x + c^T x + d, \quad Q = Q^T > 0$$

é estritamente convexa. Com efeito, sejam $x, y \in \mathcal{C}$ e $\lambda \in \mathbb{R}$, com $x \neq y$ e $0 < \lambda < 1$. Tem-se, então:

$$\begin{aligned} f(\lambda x + (1 - \lambda)y) &= \\ (\lambda x + (1 - \lambda)y)^T Q (\lambda x + (1 - \lambda)y) &+ c^T (\lambda x + (1 - \lambda)y) + d \\ &= \lambda^2 x^T Q x + \lambda(1 - \lambda)x^T Q y + \lambda(1 - \lambda)y^T Q x + (1 - \lambda)^2 y^T Q y \\ &\quad + \lambda c^T x + (1 - \lambda)c^T y + d \end{aligned}$$

Por outro lado:

$$\lambda f(x) + (1 - \lambda)f(y) =$$

$$\lambda(x^T Qx + c^T x + d) + (1 - \lambda)(y^T Qy + c^T y + d) =$$

$$\lambda x^T Qx + (1 - \lambda)y^T Qy + \lambda c^T x + (1 - \lambda)c^T y + d$$

Logo:

$$f(\lambda x + (1 - \lambda)y) - [\lambda f(x) + (1 - \lambda)f(y)] =$$

$$\lambda^2 x^T Q x + \lambda(1 - \lambda)x^T Q y + \lambda(1 - \lambda)y^T Q x + (1 - \lambda)^2 y^T Q y$$

$$+ \cancel{\lambda c^T x + (1 - \lambda)c^T y + d}$$

$$- \lambda x^T Q x - (1 - \lambda)y^T Q y - \cancel{\lambda c^T x - (1 - \lambda)c^T y - d}$$

$$= \lambda(\lambda - 1)x^T Q x - \lambda(\lambda - 1)x^T Q y - \lambda(\lambda - 1)y^T Q x + (1 - \lambda)(-\lambda)y^T Q y$$

$$= \lambda(\lambda - 1) \left[(x - y)^T Q (x - y) \right]^* \leq 0$$

$$*Q > 0, x \neq y, 0 < \lambda < 1$$

Mínimo local de uma função convexa

Seja \mathcal{C} um conjunto convexo e $f : \mathcal{C} \rightarrow \mathbb{R}$ uma função convexa. Suponha que $x' \in \mathcal{C}$ seja um ponto de mínimo local de f , isto é:

$$f(x') = \min_{x \in \mathcal{C}, \|x - x'\| \leq r} f(x)$$

para algum $r > 0$. Então, pode-se mostrar que x' também é um ponto de mínimo global.

$$f(x') = \min_{x \in \mathcal{C}, \|x - x'\| \leq r} f(x) \quad (1)$$

Por absurdo, suponha que exista $x'' \in \mathcal{C}$ tal que

$$f(x'') < f(x')$$

Nesse caso, em vista de (1), tem-se

$$\|x'' - x'\| > r > 0 \quad (2)$$

Seja agora $y = (1 - \lambda)x' + \lambda x''$ com

$$\lambda = \frac{r}{2\|x'' - x'\|} \quad (3)$$

De (2) e (3), segue que $0 < \lambda < 1$.

$$y = (1 - \lambda)x' + \lambda x''$$

$$\lambda = \frac{r}{2\|x'' - x'\|}$$

Como \mathcal{C} é convexo e $0 < \lambda < 1$, tem-se que $y \in \mathcal{C}$. Adicionalmente:

$$\|y - x'\| = \|- \lambda x' + \lambda x''\| = \lambda \|x'' - x'\| = \frac{r}{2} < r$$

$$f(x') = \min_{x \in \mathcal{C}, \|x - x'\| \leq r} f(x) \quad (1)$$

$$\|y - x'\| < r \quad (4)$$

Como $y = (1 - \lambda)x' + \lambda x''$, com $0 < \lambda < 1$, e a função f é convexa, tem-se que

$$f(y) \leq (1 - \lambda)f(x') + \lambda \underbrace{f(x'')}_{< f(x')} < f(x')$$

o que contradiz (1) e (4).

Mínimo local de uma função estritamente convexa

Seja \mathcal{C} um conjunto convexo e $f : \mathcal{C} \rightarrow \mathbb{R}$ uma função estritamente convexa. Suponha que $x' \in \mathcal{C}$ seja um ponto de mínimo local de f , isto é:

$$f(x') = \min_{x \in \mathcal{C}, \|x - x'\| \leq r} f(x)$$

para algum $r > 0$. Então, se $x \in \mathcal{C}$ e $f(x) = f(x')$, pode-se mostrar que $x = x'$.

Se $x \in \mathcal{C}$ e $f(x) = f(x')$, pode-se mostrar que $x = x'$.

Por absurdo, suponha que $x \neq x'$. Seja então

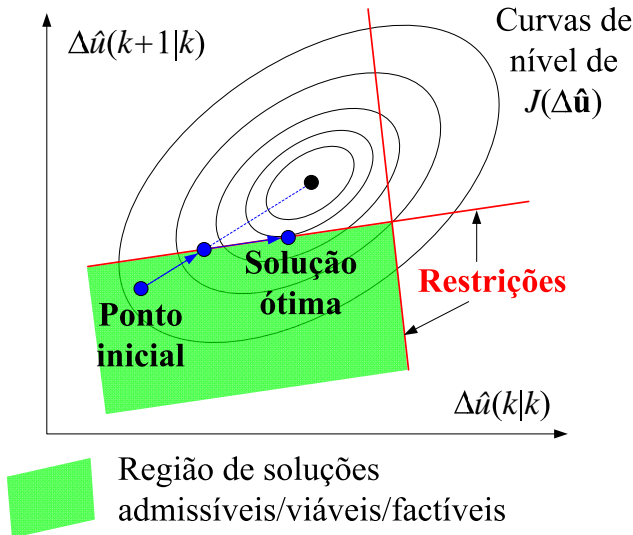
$$y = \lambda x + (1 - \lambda)x'$$

com $0 < \lambda < 1$. Da convexidade de \mathcal{C} , tem-se que $y \in \mathcal{C}$. Sabendo que f é estritamente convexa, conclui-se que

$$f(y) < \underbrace{\lambda f(x) + (1 - \lambda)f(x')}_{f(x')} = f(x')$$

o que contradiz a hipótese de que x' é mínimo local (e, portanto, global) da função f .

Programação Quadrática (Ex: $M = 2$)



Programação Quadrática no Matlab

Função QUADPROG (Optimization Toolbox):

```
x = quadprog(H,f,A,b)
minimiza 0.5*x'*H*x + f'*x
sujeito a A*x <= b
```

Em nosso caso:

$$\begin{aligned}x_{qp} &= \Delta \hat{\mathbf{u}} \\ H_{qp} &= \mathcal{H} = 2(G^T G + \rho I_M) \\ f_{qp} &= \mathbf{c} = 2G^T(\mathbf{f} - \mathbf{r}) \\ A_{qp} &= S \\ b_{qp} &= b\end{aligned}$$

podendo ser omitido o fator 2 em \mathcal{H} e \mathbf{f} .

Argumentos adicionais da função QUADPROG

$X = \text{QUADPROG}(H, f, A, b, Aeq, beq, LB, UB, X0, options)$

- $X = \text{QUADPROG}(H, f, A, b, Aeq, beq)$ solves the problem above while additionally satisfying the equality constraints $Aeq \cdot x = beq$.
- $X = \text{QUADPROG}(H, f, A, b, Aeq, beq, LB, UB)$ defines a set of lower and upper bounds on the design variables, X , so that the solution is in the range $LB \leq X \leq UB$. Use empty matrices for LB and UB if no bounds exist. Set $LB(i) = -\text{Inf}$ if $X(i)$ is unbounded below; set $UB(i) = \text{Inf}$ if $X(i)$ is unbounded above.
- $X0$: starting point.
- $options$: opções de otimização, incluindo algoritmo a ser usado e critérios de parada.

Opções de algoritmos:

- trust-region-reflective (formerly LargeScale = 'on'): default
- active-set (formerly LargeScale = 'off')
- interior-point-convex

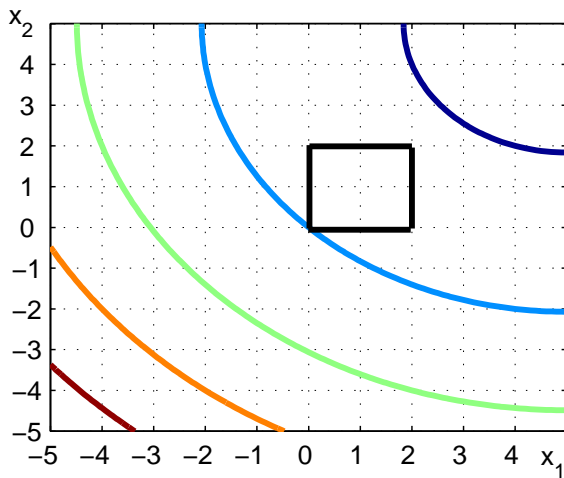
Programação Quadrática no Matlab: Exemplo

$$\min_{x_1, x_2} J = \frac{1}{2} \left[(x_1 - 5)^2 + (x_2 - 5)^2 \right]$$

s.a.

$$0 \leq x_1 \leq 2$$

$$0 \leq x_2 \leq 2$$



Custo:

$$\begin{aligned} J &= \frac{1}{2} \left[(x_1 - 5)^2 + (x_2 - 5)^2 \right] \\ &= \frac{1}{2} (x_1^2 - 10x_1 + 25 + x_2^2 - 10x_2 + 25) \\ &= \frac{1}{2} (x_1^2 + x_2^2) - 5x_1 - 5x_2 + 25 \\ &= \frac{1}{2} \begin{bmatrix} x_1 & x_2 \end{bmatrix} \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}}_{H_{qp}} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \underbrace{\begin{bmatrix} -5 & -5 \end{bmatrix}}_{f_{qp}^T} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + 25 \end{aligned}$$

Restrições:

$$0 \leq x_1 \leq 2$$

$$0 \leq x_2 \leq 2$$

$$x_1 \leq 2$$

$$x_2 \leq 2$$

$$-x_1 \leq 0$$

$$-x_2 \leq 0$$

$$\underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 0 \\ 0 & -1 \end{bmatrix}}_{A_{qp}} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \leq \underbrace{\begin{bmatrix} 2 \\ 2 \\ 0 \\ 0 \end{bmatrix}}_{b_{qp}}$$

```
>> H = eye(2);  
>> f = [-5;-5];  
>> b = [2*ones(2,1);0*ones(2,1)];  
>> A = [eye(2);-eye(2)];  
>> x = quadprog(H,f,A,b);
```

Resultado: $x = [2;2]$.

Para especificar o algoritmo a ser empregado:

```
>> options = optimset('Algorithm','active-set');  
>> quadprog(H,f,A,b,[],[],[],[],[],options);
```

Programação Quadrática: Alguns pacotes computacionais

- CVX (Matlab software for disciplined convex programming):
cvxr.com/cvx
- CVXGEN (Geração de código em C): cvxgen.com
- IPOPT (Código em C++ empregando método de pontos interiores):
www.coin-or.org/Ipopt
- QL (Código original em Fortran, traduzido posteriormente para C):
www.ai7.uni-bayreuth.de/software.htm
- CPLEX (IBM):
www-01.ibm.com/software/integration/optimization/cplex-optimizer/
- GUROBI: www.gurobi.com

MPC no espaço de estados com restrições

Informação requerida sobre a planta:

- Matrizes A, B, C do modelo no espaço de estados
- Limitantes sobre os incrementos no controle: $\Delta u_{min}, \Delta u_{max}$
- Limitantes sobre a excursão do controle: u_{min}, u_{max}
- Limitantes sobre a excursão da saída: y_{min}, y_{max}

Parâmetros de projeto:

- Peso do controle ρ
- Horizonte de predição N
- Horizonte de controle M

Inicialização:

- Fazer

$$\tilde{A} = \begin{bmatrix} A & B \\ 0_n^T & 1 \end{bmatrix}, \quad \tilde{B} = \begin{bmatrix} B \\ 1 \end{bmatrix}, \quad \tilde{C} = [C \quad 0]$$

$$G = \begin{bmatrix} \tilde{C}\tilde{B} & 0 & \dots & 0 \\ \tilde{C}\tilde{A}\tilde{B} & \tilde{C}\tilde{B} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{C}\tilde{A}^{N-1}\tilde{B} & \tilde{C}\tilde{A}^{N-2}\tilde{B} & \dots & \tilde{C}\tilde{A}^{N-M}\tilde{B} \end{bmatrix}, \quad \Phi = \begin{bmatrix} \tilde{C}\tilde{A} \\ \tilde{C}\tilde{A}^2 \\ \vdots \\ \tilde{C}\tilde{A}^N \end{bmatrix}$$

- Fazer $H_{qp} = 2(G^T G + \rho I)$, $A_{qp} = \begin{bmatrix} I_M \\ -I_M \\ T_M \\ -T_M \\ G \\ -G \end{bmatrix}$

- Fazer $k = 0$, $u(-1) = 0$

Rotina principal:

- 1 Ler $x(k)$ (estado da planta) e y_{ref} (valor de referência para a saída)
- 2 Fazer $\mathbf{r} = [y_{ref}]_N$
- 3 Fazer

$$\xi(k) = \begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix}$$

- 4 Calcular $\mathbf{f} = \Phi \xi(k)$ e $f_{qp} = 2G^T(\mathbf{f} - \mathbf{r})$

- 5 Fazer $b_{qp} = \begin{bmatrix} 1_M \Delta u_{max} \\ -1_M \Delta u_{min} \\ 1_M [u_{max} - u(k-1)] \\ 1_M [u(k-1) - u_{min}] \\ 1_N y_{max} - \mathbf{f} \\ \mathbf{f} - 1_N y_{min} \end{bmatrix}$

- 6 Resolver o problema de otimização

$$\Delta \hat{\mathbf{u}}^* = \arg \min_{\Delta \hat{\mathbf{u}} \in \mathbb{R}^M} \frac{1}{2} \Delta \hat{\mathbf{u}}^T H_{qp} \Delta \hat{\mathbf{u}} + f_{qp}^T \Delta \hat{\mathbf{u}} \quad \text{s.a.} \quad A_{qp} \Delta \hat{\mathbf{u}} \leq b_{qp}$$

- 7 Obter o incremento no controle:

$$\Delta u(k) = \Delta \hat{u}^*(k|k) = [1 \ 0 \ \cdots \ 0] \Delta \hat{\mathbf{u}}^*$$

- 8 Atualizar o controle aplicado à planta: $u(k) = u(k-1) + \Delta u(k)$
- 9 Fazer $k = k + 1$
- 10 Aguardar o próximo instante de amostragem e retornar ao passo 1.

Implementação em Matlab

- `matrizes_ss_du_restricoes.m`: Monta as matrizes Φ, G, H_{qp}, A_{qp} .
- `mpc_ss_du_restricoes.m`: S-function que implementa o controlador

Exemplo: Sistema de levitação magnética

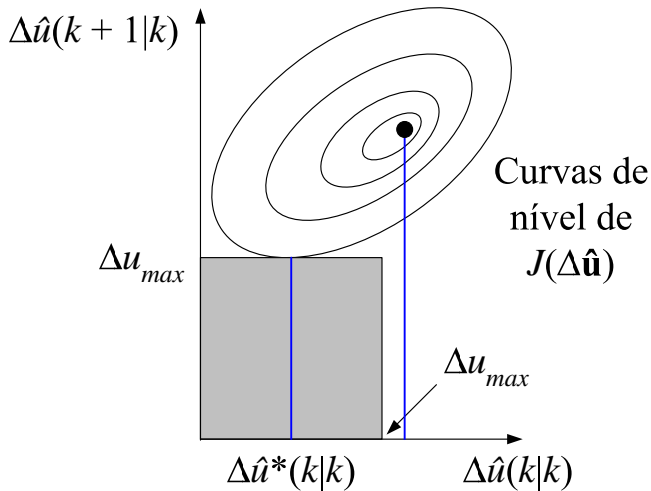
Arquivos Matlab:

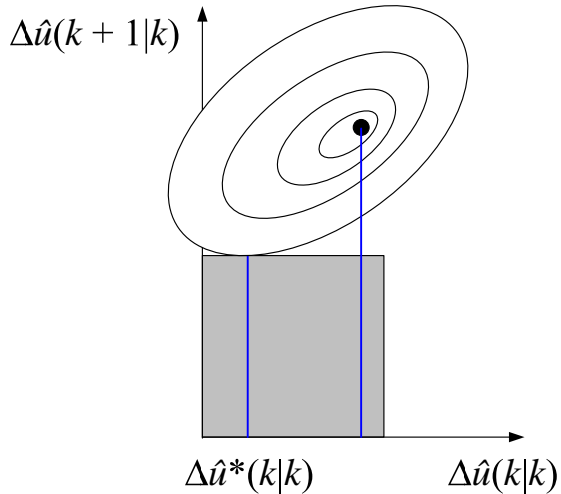
- `parametros_maglev_ss.m`: Define os parâmetros do levitador magnético
- `levitador_mpc_ss_du_restricoes.mdl`: Diagrama de simulação
- `levitador_mpc_ss_du_restricoes_pert.mdl`: Diagrama de simulação com estimação de perturbações

Importância do tratamento de restrições no controle

A importância de tratar restrições na saída y está usualmente relacionada a imposições de segurança ou requisitos de qualidade para o produto final.

- Qual a necessidade de tratar restrições sobre a entrada (em termos de excursão e/ou incrementos) ?
- Por que não calcular a sequência de controle (ou incrementos de controle) ótima e saturá-la nos limitantes inferior e superior, se necessário ?





Restrições no controle: Acomodação de falhas

- Em particular, a possibilidade de tratar restrições no controle pode ser útil para acomodação de falhas de atuadores.
- Nesse contexto, um sistema de monitoramento de falhas informaria ao controlador preditivo os novos limitantes para a excursão e/ou incrementos do sinal de controle.

Exemplo: `levitador_mpc_ss_du_restricoes_falha.mdl`

Resumo da aula de hoje

Tratamento de restrições - Caso SISO:

- Incrementos no controle Δu
- Excursão do controle u
- Excursão da saída y

Programação quadrática

Convexidade

Implementação em Matlab

Tópico da próxima aula

- Extensão ao caso MIMO