Commande linéaire quadratique

GEL-4250 / GEL-7015 Commande multivariable

André Desbiens, ing., Ph.D.

Département de génie électrique et de génie informatique

Hiver 2020

UNIVERSITÉ LAVAL

PARTIE I: OPTIMISATION

Rappels mathématiques

Vecteur: $x \in \Re^n$ Matrice: $M \in \Re^{n \times n}$ Fonction scalaire d'un vecteur: $J(x) : \Re^n \to \Re$

- La matrice M est définie positive si et seulement si $x^T M x > 0 \quad \forall x \neq 0$ (toutes les valeurs propres de M sont alors positives). Si $M = M_1{}^T M_1$ alors M est symétrique et définie positive ou nulle par construction.

- $$\begin{aligned} \frac{\delta M x}{\delta x} &= M^T \\ \frac{\delta u^T M u}{\delta u} &= M u + M^T u \\ &= 2 M u \quad \text{si } M \text{ est symétrique} \end{aligned}$$

- Gradient de J: $$\nabla J(x) = \frac{\delta J(x)}{\delta x} = \left[\frac{\delta J(x)}{\delta x_1} \frac{\delta J(x)}{\delta x_2} \cdots \frac{\delta J(x)}{\delta x_n} \right]$$

- Hessienne de J: $$H_e(x) = \frac{\delta^2 J(x)}{\delta x^2} = \begin{bmatrix} \frac{\delta J(x)}{\delta x_1^2} & \frac{\delta J(x)}{\delta x_1 x_2} & \cdots & \frac{\delta J(x)}{\delta x_1 x_n} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\delta J(x)}{\delta x_n x_1} & \frac{\delta J(x)}{\delta x_n x_2} & \cdots & \frac{\delta J(x)}{\delta x_n^2} \end{bmatrix}$$

Optimisation sans contraintes

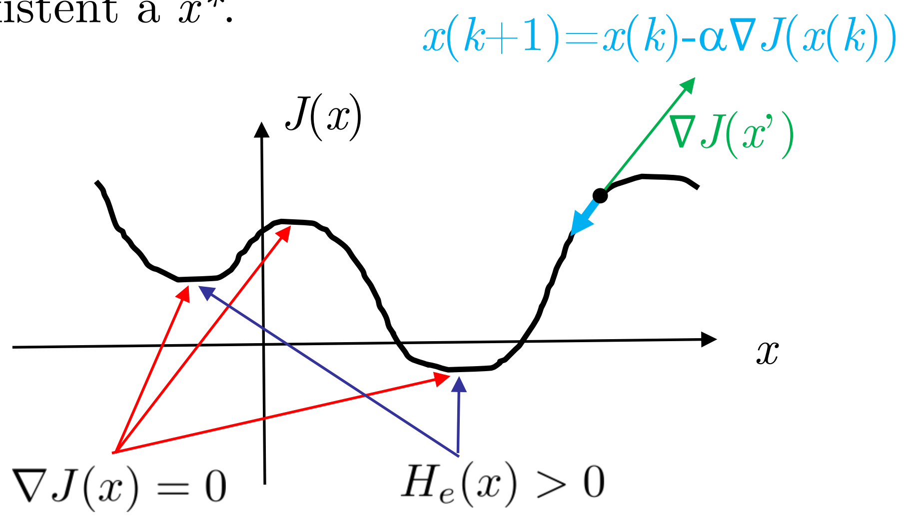
- On cherche $x^* \in \Re^n$ pour que $J(x^*) \leq J(x) \;\; \forall x \in \Re^n$:

$$x^* = \arg \min_{x \in \Re^n} J(x)$$

- J: critère, fonction objectif, fonction coût.

- Conditions nécessaires et suffisantes pour que x^* soit un minimum (pas nécessairement global):

  - Les deux première dérivées de J existent à x^*.
  - $\nabla J(x^*) = 0$
  - $H_e(x^*)$ est définie positive.



Optimisation avec contraintes d'égalité

- On cherche $x^* \in \Re^n$ pour que $J(x^*) \leq J(x) \;\; \forall x \in \Re^n$ tout en respectant les c contraintes d'égalités $G_i(x) = 0$, $i = 1$ à c, où $G_i(x) : \Re^n \rightarrow \Re$ et

$$G(x) = \begin{bmatrix} G_1(x) \\ G_2(x) \\ \ldots \\ G_c(x) \end{bmatrix}$$

- Les 2 problèmes d'optimisation suivants sont équivalents:

$$\min_{x \in \Re^n,\; G(x)=0} J(x) \qquad\qquad \min_{x \in \Re^n, \lambda \in \Re^c} \underbrace{\left(J(x) + \lambda^T G(x)\right)}$$

Lagrangien: $L(x, \lambda)$
Multiplicateur de Lagrange: λ

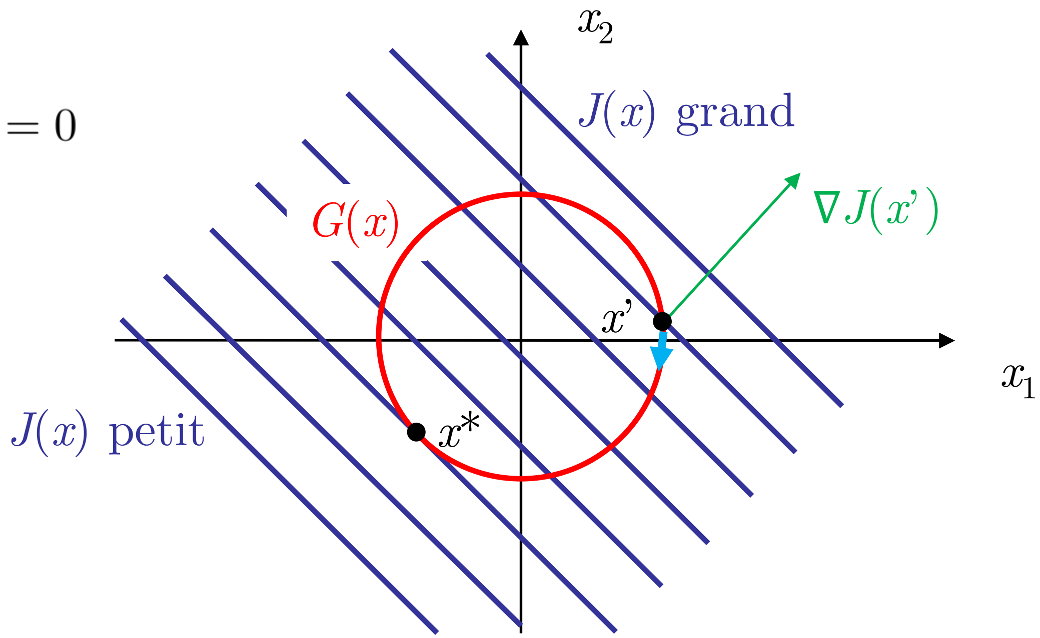
Si la contrainte est respectée alors $J(x)$ et $L(x, \lambda)$ sont égaux et ont donc le même minimum. Selon la 2e condition de la page précédente, pour trouver le minimum non contraint de $L(x, \lambda)$, il faut:

$$\frac{\delta L(x^*, \lambda)}{\delta x} = \frac{\delta L(x^*, \lambda)}{\delta \lambda} = 0$$

… Optimisation avec contraintes d'égalité

- Au minimum on a: $$\frac{\delta L(x^*, \lambda)}{\delta x} = 0$$
$$\frac{\delta J(x^*)}{\delta x} + \lambda^T \frac{\delta G(x^*)}{\delta x} = 0$$
donc
$$\frac{\delta J(x^*)}{\delta x} = -\lambda^T \frac{\delta G(x^*)}{\delta x}$$

- Exemple: $J(x) = x_1 + x_2$
$$G(x) = x_1^2 + x_2^2 - 1 = 0$$



… Optimisation avec contraintes d'égalité

$$L(x,\lambda) = x_1 + x_2 + \lambda(x_1^2 + x_2^2 - 1)$$

$$\left.\begin{aligned}
\frac{\delta L(x,\lambda)}{\delta x_1} &= 1 + 2\lambda x_1 = 0 \\
\frac{\delta L(x,\lambda)}{\delta x_2} &= 1 + 2\lambda x_2 = 0 \\
\frac{\delta L(x,\lambda)}{\delta \lambda} &= x_1^2 + x_2^2 - 1 = 0
\end{aligned}\right\}\ \text{3 équations et 3 inconnues}$$

$$\lambda = -0.707 \quad x_1 = x_2 = 0.707 : \text{ maximum}$$

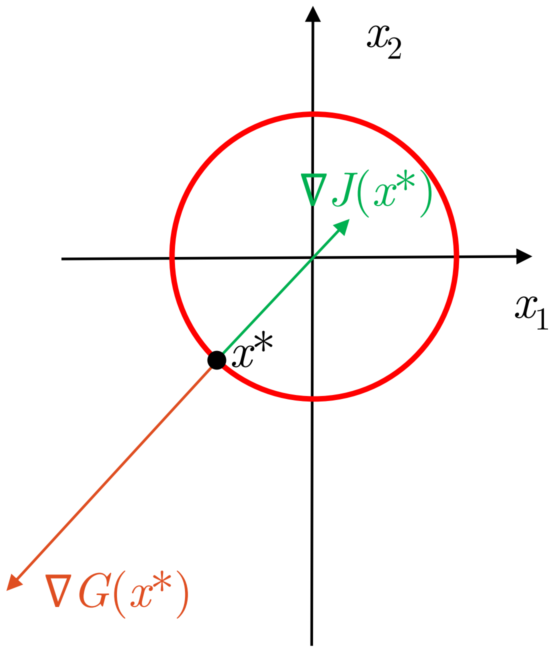
$$\lambda = 0.707 \quad x_1 = x_2 = -0.707 : \text{ minimum}$$

Au minimum:

$$\frac{\delta J(x^*)}{\delta x} = [1 \quad 1]$$

$$\frac{\delta G(x^*)}{\delta x} = [2x_1^* \quad 2x_2^*] = [-1.414 \quad -1.414]$$

$$\frac{\delta J(x^*)}{\delta x} = -\lambda \frac{\delta G(x^*)}{\delta x}$$



PARTIE II: PRINCIPE DU MAXIMUM DE PONTRYAGIN

UNIVERSITÉ LAVAL

Le problème

- Un système dynamique non linéaire $\dot{x} = f(x,u)$ et ses conditions initiales $x(0) = \boldsymbol{\theta}$:

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \qquad u = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{n_u} \end{bmatrix} \qquad f() = \begin{bmatrix} f_1() \\ f_2() \\ \vdots \\ f_n() \end{bmatrix} \qquad \dot{x}_i = f_i(x,u) \quad i = 1,2,\ldots,n$$

- Une fonction (scalaire) objectif à minimiser: $J = \underbrace{F(x)|_{t=t_1}}_{\text{Coût terminal}} + \underbrace{\int_0^{t_1} f_0(x,u)dt}_{\text{Coût intégral}}$

  tout en respectant k contraintes terminales: $g_i(x)|_{t=t_1} = 0 \quad i = 1,2,\ldots,k \ \ (k \leq n)$

  et des contraintes sur u: $u(t) \in \Omega \subset \Re^{n_u} \quad$ pour $0 \leq t \leq t_1$

- Le temps final t_1 peut être connu initialement ou non.
- Quelle est la trajectoire optimale $u^*(t)$ pour $t = 0$ à $t = t_1$?

La solution (par Lev Semyonovich Pontryagin en 1956)

- Ajout d'un état défini à partir du critère et de sa condition initiale:

$$\dot{x}_0 = f_0(x,u) + \sum_{i=1}^{n} \frac{\delta F(x)}{\delta x_i} \dot{x}_i \qquad x_0(0) = F(x)|_{t=0}$$

- Vecteur étendu d'état: $\hat{x} = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \qquad \hat{f}() = \begin{bmatrix} f_0() \\ f_1() \\ f_2() \\ \vdots \\ f_n() \end{bmatrix}$

- Étape 1 – Trouver les expressions temporelles des états adjoints: $z_i(t),\ i = 0$ à n
- Étape 2 – Trouver u qui maximise l'Hamiltonien

$$H(\hat{x}, z, u) = z^T \hat{f} = \sum_{i=0}^{n} z_i(t) f_i(x,u)$$

tout en respectant les contraintes sur u: $u(t) \in \Omega \subset \Re^{n_u}$ pour $0 \leq t \leq t_1$

donc la commande optimale est: $u^*(t) = \arg\min_{u\in\Omega} -H$

… La solution (par Lev Semyonovich Pontryagin en 1956)

- Équations d'évolution des $n+1$ états adjoints:

$$\dot{z}_0 = 0$$
$$\dot{z}_i = -\frac{\delta H}{\delta x_i} \qquad i = 1, 2, \ldots, n$$

- n conditions de transversalité (les k constantes c_j sont des scalaires à déduire):

$$z_i(t_1) = -\frac{\delta F(x)}{\delta x_i}\bigg|_{t=t_1} + \sum_{j=1}^{k} c_j \frac{\delta g_j(x)}{\delta x_i}\bigg|_{t=t_1} \qquad i = 1, 2, \ldots, n$$

Si la solution optimale est u^*, alors:

1) $z_0 =$ - 1
2) $u^* =$ commande pour laquelle H est à son supremum
3) H est constant le long de la trajectoire optimale:

$$H(\hat{x}^*, z^*, u^*) = \begin{cases} \text{constant} & \text{si } t_1 \text{ est connu initialement} \\ 0 & \text{si } t_1 \text{ est inconnu} \end{cases}$$

… La solution (par Lev Semyonovich Pontryagin en 1956)

- Il faut donc résoudre $2n+2$ équations différentielles du 1er ordre et trouver k inconnues:

$$\dot{z}_0 = 0$$

$$\dot{z}_i = -\frac{\delta H}{\delta x_i} \qquad i = 1, 2, \ldots, n$$

$$\dot{x}_0 = f_0(x,u) + \sum_{i=1}^{n} \frac{\delta F(x)}{\delta x_i}\dot{x}_i$$

$$\dot{x}_i = f_i(x,u) \qquad i = 1, 2, \ldots, n$$

$$c_i = ? \qquad i = 1, 2, \ldots, k$$

On sait déjà que z_0 = - 1

- Pour y parvenir, on a n +2 conditions initiales, k contraintes terminales et n conditions de transversalité:

$$x(0) = \theta$$

$$z_0(0) = -1$$

$$x_0(0) = F(x)|_{t=0}$$

$$g_i\big(x(t_1)\big) = 0 \qquad i = 1, 2, \ldots, k$$

$$z_i(t_1) = -\left.\frac{\delta F(x)}{\delta x_i}\right|_{t=t_1} + \sum_{j=1}^{k} c_j \left.\frac{\delta g_j(x)}{\delta x_i}\right|_{t=t_1} \qquad i = 1, 2, \ldots, n$$

… La solution (par Lev Semyonovich Pontryagin en 1956)

- Si le temps t_1 est inconnu, on ajoute l'équation suivante:

$$H(\hat{x}^*, z^*, u^*) = 0$$

Remarques sur la solution:

- Les conditions de la page précédente (et celle ci-haut dans le cas de t_1 inconnu) sont nécessaires pour obtenir une solution optimale, si elle existe.
- Il peut y avoir plus d'une solution respectant ces conditions. Il faut alors choisir celle qui donne la plus petite valeur de J.
- La solution optimale n'est pas nécessairement unique (il peut y avoir plus d'une solution donnant la même valeur de J).
- S'il est impossible de satisfaire toutes les conditions alors il n'y a pas de solution optimale.

Exemple 1 - Jardinier

Problème:

- Faire pousser ses plantes d'une hauteur initiale $x_1(0) = 0$ à $x_1(1) = 4$ en minimisant les coûts du matériel et de l'énergie requis (u: engrais, éclairage, etc.).
- Croissance des plantes: $\dot{x}_1 = 3 + u$
- Coût de u: $J = \int_0^1 0.5u^2 dt$
- Que vaut $u^*(t)$ pour $t = 0$ à 1?

On a donc:

$$
\begin{aligned}
x_1(0) &= 0 \\
t_1 &= 1 \\
f_1 &= 3 + u \\
g_1 &= x - 4 \\
F &= 0 \\
f_0 &= 0.5u^2 \\
\dot{x}_0 &= 0.5u^2 \\
x_0(0) &= 0
\end{aligned}
$$

$$H = -0.5u^2 + z_1(3 + u)$$

$$\left.\begin{aligned} \dot{z}_1 &= 0 \\ z_1(1) &= 0 + c_1 \end{aligned}\right\} \quad z_1(t) = c_1$$

$$H = -0.5u^2 + c_1(3 + u)$$

… Exemple 1 - Jardinier

$$u^*(t) = \arg\min_{u\in\Re} -H$$
$$= \arg\min_{u\in\Re} \left(0.5u^2 - 3c_1 - c_1 u\right)$$

$$\frac{\delta(-H)}{\delta u} = u - c_1 = 0$$
$$u = c_1$$
$$\frac{\delta^2(-H)}{\delta u^2} = 1 \qquad \text{(minimum)}$$

$$\left.\begin{aligned} \dot{x}_1 &= 3 + c_1 \\ x_1(t) &= (3 + c_1)t + a \end{aligned}\right\} \quad \begin{aligned} x_1(0) &= 0 = a \\ x_1(1) &= 4 = 3 + c_1 \end{aligned}$$

$$\Downarrow$$

$$c_1 = -1$$
$$u = 1$$

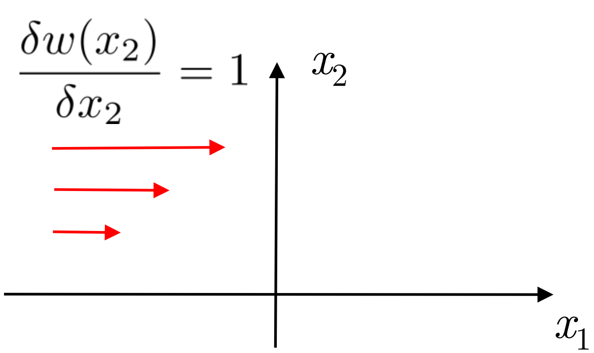
Exemple 2 - Marin

Problème:

- Calculer la trajectoire d'un bateau pour couvrir le maximum de distance vers la droite en présence du courant en un temps donné t_1.
- La vitesse du courant w augmente selon x_2: $\dfrac{\delta w(x_2)}{\delta x_2} = 1$
- Sans courant, le moteur pousse le bateau à une vitesse de 1. En manipulant le gouvernail, on modifie la direction de ce vecteur vitesse (u_1, u_2):

$$\begin{aligned} \dot{x}_1 &= w(x_2) + u_1(t) \\ \dot{x}_2 &= u_2(t) \end{aligned} \qquad u(t) \in \Omega = \left\{(u_1, u_2) : u_1^2 + u_2^2 = 1\right\}$$

- La position initiale est $x_1(0) = x_2(0) = 0$
- Que vaut $u^*(t)$ pour $t = 0$ à t_1?



… Exemple 2 - Marin

Solution: $$J = -x_1(t)\int_0^{t_1} 0dt$$

$$f_1 = w(x_2) + u_1$$
$$f_2 = u_2$$
$$x_1(0) = 0$$
$$x_2(0) = 0$$
$$t_1 = 1$$
$$F = -x_1$$
$$f_0 = 0$$
$$\dot{x}_0 = 0 - \dot{x}_1 + 0 = -w(x_2) + u_1$$
$$x_0(0) = -x_1(0) = 0$$

$$H = 0 + z_1 w(x_2) + z_1 u_1 + z_2 u_2$$

$$\left.\begin{aligned} \dot{z}_1 &= 0 \\ \dot{z}_2 &= -z_1 \frac{\delta w(x_2)}{\delta x_2} = -z_1 \\ z_1(1) &= 1 \\ z_2(1) &= 0 \end{aligned}\right\} \begin{aligned} z_1(t) &= 1 \\ z_2(t) &= -t + 1 \end{aligned}$$

$$H = w(x_2) + u_1 + (-t+1)u_2$$

… Exemple 2 - Marin

$$\begin{aligned} u^*(t) &= \arg\min_{u\in\Omega} -H \\ &= \arg\min_{u_1^2+u_2^2=1} -w(x_2) - u_1 + (t-1)u_2 \end{aligned}$$

Il s'agit d'une optimisation avec une contrainte d'égalité (page 5):

$$u^*(t) = \arg\min_{u_1\in\Re, u_2\in\Re} \underbrace{\left[-w(x_2) - u_1(t) + (t-1)u_2(t) + \lambda(u_1^2(t) + u_2^2(t) - 1)\right]}_{L}$$

$$\left.\begin{aligned} \frac{\delta L}{\delta u_1} &= 0 \\ \frac{\delta L}{\delta u_2} &= 0 \\ \frac{\delta L}{\delta \lambda} &= 0 \end{aligned}\right\} \qquad \begin{aligned} u_1 &= \frac{1}{\sqrt{1+(1-t)^2}} \\ u_2 &= \frac{1-t}{\sqrt{1+(1-t)^2}} \end{aligned}$$

Conclusion

- Méthode très puissante et élégante.
- Elle n'utilise pas les mesures: pas de rétroaction (à moins de continuellement recalculer la solution en se basant sur de nouvelles conditions initiales correspondant à l'état présent).
- Elle pourrait être utiliser pour pré-calculer la consigne à suivre par un algorithme de commande.
- Nous l'utiliserons pour définir la commande linéaire quadratique.

PARTIE III: COMMANDE LINÉAIRE QUADRATIQUE

Commande linéaire quadratique (horizon fini)

- LQR (linear-quadratic regulator).
- Problème de commande (modèle linéaire, critère quadratique):

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

(stabilisable, i.e. les états non gouvernables, si présents, sont stables)

$$x(0) = \theta$$

$$J = \frac{1}{2}x^T(t_1)P(t_1)x(t_1) + \frac{1}{2}\int_0^{t_1}(x^TQx + u^TRu)dt$$

$$Q \geq 0, R > 0, P(t_1) = P_1 \geq 0$$

- Solution par le principe du maximum de Pontryagin:

$$g_1 = 0$$

$$F = \frac{1}{2}x^TPx$$

$$f_0 = \frac{1}{2}\left[x^TQx + u^TRu\right]$$

$$H = z_o f_0 + z_1^T f$$

$$= -\frac{1}{2}\left[x^TQx + u^TRu\right] + z_1^T(Ax + Bu)$$

$$\dot{z}_1 = -\frac{\delta H}{\delta x} = Qx - A^T z_1$$

$$z_1(t_1) = -\left.\frac{\delta F}{\delta x}\right|_{t=t_1} = -P(t_1)x(t_1)$$

… Commande linéaire quadratique (horizon fini)

$$u^* = \arg\min_{u\in\Re} -H = \frac{1}{2}\left[x^TQx + u^TRu\right] - z_1^TAx - z_1^TBu$$

$$\frac{\delta - H}{\delta u} = Ru - B^Tz_1 = 0$$

$$\boxed{u(t) = R^{-1}B^Tz_1(t)}$$

$z_1(t) = ?$ On essaie avec $z_1(t) = -P(t)x(t)$:

$$\dot{z}_1 = Qx - A^Tz_1$$

$$= Qx + A^TPx = -\dot{P}x - P\dot{x} = -\dot{P}x - P(Ax - BR^{-1}B^TPx)$$

$$\boxed{-\dot{P} = PA + A^TP - PBR^{-1}B^TP + Q}$$

$$z_1(t_1) = -P(t_1)x(t_1)$$

$$\boxed{P(t_1) = P_1}$$

Équation différentielle ordinaire de Ricatti: sa solution (comme une éq. diff. ord. avec une condition initiale P_1 mais en temps inverse) donne $P(t)$ pour $t = t_1$ à 0 (calculé a priori).

$$\boxed{u(t) = -R^{-1}B^TP(t)x(t)}$$

Commande par retour d'état à gain variant dans le temps pour $t = 0$ à t_1.

… Commande linéaire quadratique (horizon fini)

$$x(k+1) = Ax(k) + Bu(k) \qquad \text{(stabilisable)}$$
$$x(0) = \theta$$

$$J = \frac{1}{2}x^T(N)P(N)x(N) + \frac{1}{2}\sum_{k=0}^{N-1}\left[x^T(k)Qx(k) + u^T(k)Ru(k)\right]$$
$$Q \geq 0, R > 0, P(N) = P_N \geq 0$$

$$\boxed{P(k) = A^T\left[P(k+1) - P(k+1)B(B^TP(k+1)B + R)^{-1}B^TP(k+1)\right]A + Q}$$

$$\boxed{P(N) = P_N}$$

En reculant dans le temps, on peut calculer $P(k)$ pour $k = N$ - 1 à 1 (calculé a priori).

$$\boxed{u(k) = -(B^TP(k+1)B + R)^{-1}B^TP(k+1)Ax(k)}$$

Commande par retour d'état à gain variant dans le temps pour $k = 0$ à N - 1.

Commande linéaire quadratique (horizon infini)

- Problème de commande:

$$
\begin{aligned}
\dot{x} &= Ax + Bu = f(x,u) \qquad \text{(stabilisable)} \\
x(0) &= \theta \\
J &= \frac{1}{2}\left[\int_0^{\infty} (x^T Q x + u^T R u) dt\right] \\
Q &\geq 0, R > 0
\end{aligned}
$$

- Solution par le principe du maximum de Pontryagin:

$$
\begin{aligned}
g_1 &= 0 \\
F &= 0 \\
f_0 &= \frac{1}{2}\left[x^T Q x + u^T R u\right] \\
H &= z_o f_0 + z_1^T f \\
&= -\frac{1}{2}\left[x^T Q x + u^T R u\right] + z_1^T (Ax + Bu)
\end{aligned}
$$

$$
\begin{aligned}
\dot{z}_1 &= -\frac{\delta H}{\delta x} = Qx - A^T z_1 \\
z_1(\infty) &= 0
\end{aligned}
$$

… Commande linéaire quadratique (horizon infini)

$$u^* = \arg\min_{u\in\Re} -H = \frac{1}{2}\left[x^T Q x + u^T R u\right] - z_1^T A x - z_1^T B u$$

$$\frac{\delta - H}{\delta u} = Ru - B^T z_1 = 0$$

$$\boxed{u(t) = R^{-1} B^T z_1(t)}$$

$z_1(t) = ?$ On essaie avec $z_1(t) = -Px(t)$ (P constant):

$$\dot{z}_1 = Qx - A^T z_1$$

$$= Qx + A^T P x = -P\dot{x} = -P(Ax - BR^{-1}B^T P x)$$

$$\boxed{0 = PA + A^T P - PBR^{-1}B^T P + Q}$$

Équation algébrique de Ricatti. Solution avec Matlab: *care*

$$\boxed{\begin{aligned} u(t) &= -R^{-1}B^T P x(t) \\ &= -Kx(t) \end{aligned}}$$

Commande par retour d'état (gain -K calculé a priori).
Calcul du gain avec Matlab: *lqr*

LQR_1x1.m

… Commande linéaire quadratique (horizon infini)

Le résultat est similaire en discret:

$$x(k+1) = Ax(k) + Bu(k) \qquad \text{(stabilisable)}$$
$$x(0) = \theta$$
$$J = \frac{1}{2}\sum_{k=0}^{\infty}\left[x^T(k)Qx(k) + u^T(k)Ru(k)\right]$$
$$Q \geq 0, R > 0$$

$$\boxed{P = A^T\left[P - PB(B^TPB+R)^{-1}B^TP\right]A + Q}$$

Solution avec
Matlab: *dcare*

$$\boxed{u(k) = -(B^TPB+R)^{-1}B^TPAx(k)}$$

Calcul du gain avec
Matlab: *dlqr*

… Commande linéaire quadratique (horizon infini)

$$\dot{x} = Ax + Bu$$
$$x(0) = x_0$$
$$J = \frac{1}{2}\left[\int_0^\infty (x^T Q x + u^T R u)dt\right]$$
$$\boxed{Q = M^T M} \geq 0, R > 0$$

Si

1) le système est stabilisable,
2) $(A,\ M)$ est observable (i.e. observabilité en supposant $C = M$) – cette condition revient à peu près à dire que tous les modes du procédé apparaissent dans le critère,

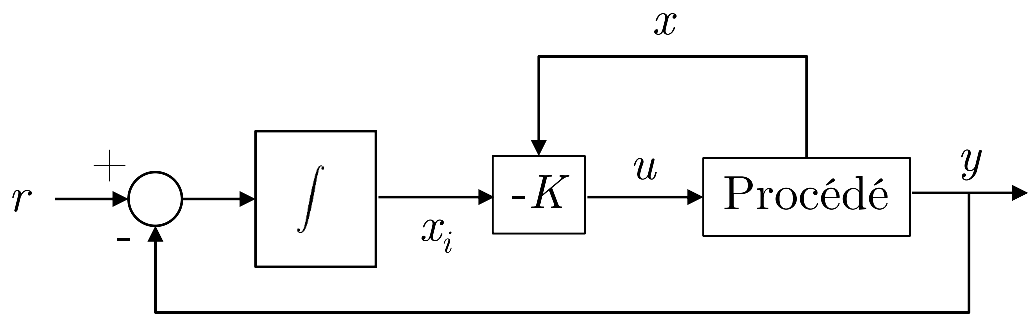
alors

1) l'équation de Ricatti converge,
2) le système en boucle fermée est stable.

Ces propriétés sont également vraies pour le cas discret.

Poursuite avec action intégrale

$$\dot{x} = Ax + Bu$$
$$y = Cx$$
$$\dot{x}_i = r - y = r - Cx$$



Modèle augmenté (on suppose $r = 0$, agira comme une perturbation):

$$\begin{bmatrix} \dot{x} \\ \dot{x}_i \end{bmatrix} = \begin{bmatrix} A & 0_{n \times n_y} \\ -C & 0_{n_y \times n_y} \end{bmatrix} \begin{bmatrix} x \\ x_i \end{bmatrix} + \begin{bmatrix} B \\ 0_{n_y \times n_u} \end{bmatrix} u$$

$$y = \begin{bmatrix} C & 0_{n_y \times n_y} \end{bmatrix} \begin{bmatrix} x \\ x_i \end{bmatrix}$$

$$\dot{z} = A_a z + B_a u$$
$$y = C_a z$$

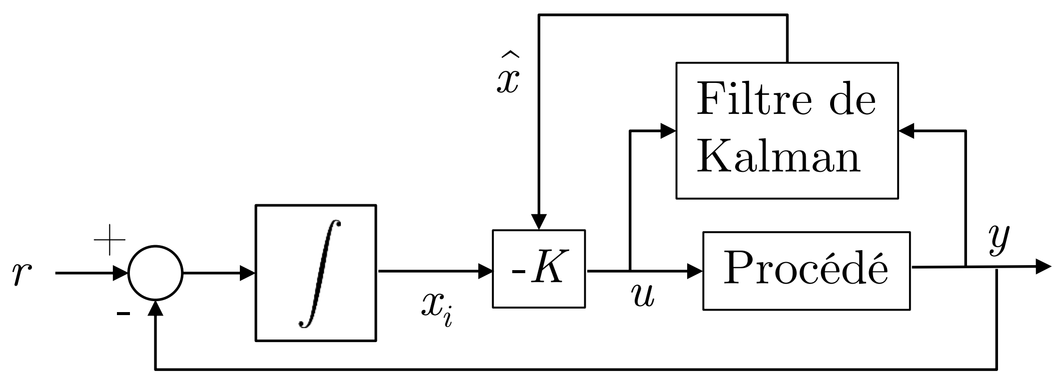
Design d'un LQR avec le modèle augmenté => action intégrale.
Solution avec Matlab (continu et discret): *lqi*

$$u = -Kz = -K_x x - K_i x_i$$

LQI_1x1.m
LQI_3x3.m
LQI_discret_3x3.m

Commande linéaire quadratique gaussienne

- LQG (linear-quadratic Gaussian regulator).
- Problème de commande: modèle linéaire stabilisable et observable, critère quadratique, bruits blancs gaussiens de procédé et de mesure, états pas tous mesurés.
- Principe de séparation: concevoir une commande optimale par rétroaction pour un système stochastique = concevoir un observateur optimal pour le système stochastique qui fournit les états à un contrôleur déterministe (conçu en négligeant les bruits) calculé en supposant les états mesurés.
- Dans notre cas: commande linéaire quadratique + filtre de Kalman = LQG.



LQI_3x3.m
LQI_discret_3x3.m

Principe de séparation

Système à commander:

$$\dot{x} = Ax + Bu$$
$$y = Cx$$

Observateur: $\dot{\hat{x}} = (A - K_{obs}C)\hat{x} + Bu + K_{obs}y$

Loi de commande: $u = -K_x\hat{x}$

Comment évoluent les états du système et l'erreur d'observation si on applique l'observateur et la loi de commande au système?

$$\left.\begin{aligned} e &= x - \hat{x} \\ \dot{e} &= (A - K_{obs}C)e \\ u &= -K_x(x - e) \end{aligned}\right\} \begin{bmatrix} \dot{x} \\ \dot{e} \end{bmatrix} = \begin{bmatrix} A - BK_x & BK_x \\ 0 & A - K_{obs}C \end{bmatrix} \begin{bmatrix} x \\ e \end{bmatrix}$$

Matrice triangulaire: ses valeurs propres sont celles de A - BK_x et celles de A - $K_{obs}C$.

Par conséquent: si le régulateur conduit à un système stable en supposant les états connus (valeurs propres de $A - BK_x$) et que l'observateur est stable sans la commande (valeurs propres de $A - K_{obs}C$), alors le système total est stable.

Choix des pondérations

$$J = \frac{1}{2}\left[\int_0^{\infty}(x^T Q x + u^T R u)dt\right]$$

$$Q = \begin{bmatrix} q_1 & & 0 \\ & \ddots & \\ 0 & & q_n \end{bmatrix} \quad R = \rho \begin{bmatrix} r_1 & & 0 \\ & \ddots & \\ 0 & & r_{n_u} \end{bmatrix}$$

- Exemple pour le choix des q_i :

  Variation acceptable pour $x_1 = 0.01$ m => $q_1 = (1/0.01)^2 = 10000$
  Variation acceptable pour $x_4 = 2$ degrés => $q_4 = (1/2)^2 = 0.25$
  $q_1 \, {x_1}^2 = 1$ quand $x_1 = 0.01$ m
  $q_4 \, {x_4}^2 = 1$ quand $x_4 = 2$ degrés

- Même approche pour le choix des r_i.

- Ajustement final de ρ par essais et erreurs.

- La performance nominale et la robustesse devraient être vérifiées en simulation (généralement, la commande LQR est assez robuste et la commande LQG est moins robuste que LQR).

Dualité commande et estimation

- Similitude remarquable entre les équations de commande LQ à horizon fini et le filtre de Kalman.

- Commande LQ à horizon fini

$$P(k) = A^T \left[P(k+1) - P(k+1)B(B^T P(k+1)B + R)^{-1} B^T P(k+1) \right] A + Q$$
$$P(N) = P_N$$
$$K(k) = (B^T P(k+1)B + R)^{-1} B^T P(k+1)A$$

- Filtre de Kalman:

$$P(k+1) = A \left[P(k) - P(k)C^T (CP(k)C^T + R)^{-1} CP(k) \right] A^T + Q$$
$$P(0) = P_0$$
$$K(k) = AP(k)C^T (CP(k)C^T + R)^{-1}$$

- Résolution de P en temps inverse ou non, B devient C^T, A devient A^T, etc.

- On obtiendrait également cette similitude dans le domaine continu.