# Mise en pratique de LSPI pour la commande linéaire quadratique adaptative d'une surface de manipulation à coussin d'air actif

Guillaume Laurent

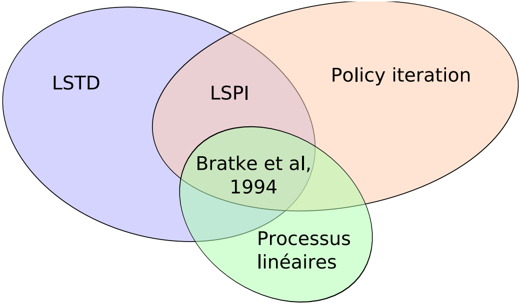
Institut FEMTO-ST, UMR CNRS 6174 - UFC/ENSMM/UTBM
Département Automatique et Systèmes Micro-Mécatroniques
24, rue Alain Savary, 25000 Besançon, France
guillaume.laurent@ens2m.fr

JFPDA 2010
Besançon, 1er et 2 juin

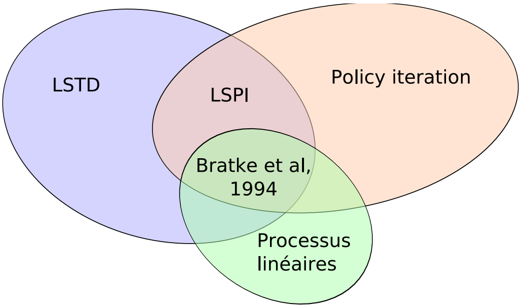
# Introduction

- Steven J. Bradtke and B. Erik Ydstie and Andrew G. Barto, Adaptive linear quadratic control using policy iteration, In *Proc. of the American Control Conference*, 1994.

- Steven J. Bradtke and B. Erik Ydstie and Andrew G. Barto, Adaptive linear quadratic control using policy iteration, In *Proc. of the American Control Conference*, 1994.



- Steven J. Bradtke and B. Erik Ydstie and Andrew G. Barto, Adaptive linear quadratic control using policy iteration, In *Proc. of the American Control Conference*, 1994.




Mise en pratique sur un système réel dont la dynamique est variable

# Plan de la présentation

# Plan de la présentation

# Représentation d'état en temps discret d'un processus linéaire

# Représentation d'état en temps discret d'un processus linéaire


$$\begin{cases} X_{k+1} & = AX_k + BU_k \\ Y_k & = CX_k + DU_k \end{cases}$$

# Représentation d'état en temps discret d'un processus linéaire


$$\begin{cases} X_{k+1} & = AX_k + BU_k \\ Y_k & = CX_k + DU_k \end{cases}$$

- $X_k$ est le vecteur d'état de dimension $n$

# Représentation d'état en temps discret d'un processus linéaire


$$\begin{cases} X_{k+1} & = AX_k + BU_k \\ Y_k & = CX_k + DU_k \end{cases}$$

- $X_k$ est le vecteur d'état de dimension $n$
- $U_k$ est le vecteur de commande de dimension $m$

# Représentation d'état en temps discret d'un processus linéaire


$$\begin{cases} X_{k+1} & = AX_k + BU_k \\ Y_k & = CX_k + DU_k \end{cases}$$

- $X_k$ est le vecteur d'état de dimension $n$
- $U_k$ est le vecteur de commande de dimension $m$
- $Y_k$ est le vecteur de sortie de dimension $p$

# Représentation d'état en temps discret d'un processus linéaire


$$\begin{cases} X_{k+1} & = AX_k + BU_k \\ Y_k & = CX_k + DU_k \end{cases}$$

- $X_k$ est le vecteur d'état de dimension $n$
- $U_k$ est le vecteur de commande de dimension $m$
- $Y_k$ est le vecteur de sortie de dimension $p$
- $A$ est la matrice de dynamique de dimension $n \times n$

# Représentation d'état en temps discret d'un processus linéaire


$$\begin{cases} X_{k+1} & = AX_k + BU_k \\ Y_k & = CX_k + DU_k \end{cases}$$

- $X_k$ est le vecteur d'état de dimension $n$
- $U_k$ est le vecteur de commande de dimension $m$
- $Y_k$ est le vecteur de sortie de dimension $p$
- $A$ est la matrice de dynamique de dimension $n \times n$
- $B$ est la matrice de commande de dimension $n \times m$

# Représentation d'état en temps discret d'un processus linéaire


$$\begin{cases} X_{k+1} & = AX_k + BU_k \\ Y_k & = CX_k + DU_k \end{cases}$$

- $X_k$ est le vecteur d'état de dimension $n$
- $U_k$ est le vecteur de commande de dimension $m$
- $Y_k$ est le vecteur de sortie de dimension $p$
- $A$ est la matrice de dynamique de dimension $n \times n$
- $B$ est la matrice de commande de dimension $n \times m$
- $C$ est la matrice d'observation de dimension $p \times n$

# Représentation d'état en temps discret d'un processus linéaire


$$\begin{cases} X_{k+1} & = AX_k + BU_k \\ Y_k & = CX_k + DU_k \end{cases}$$

- $X_k$ est le vecteur d'état de dimension $n$
- $U_k$ est le vecteur de commande de dimension $m$
- $Y_k$ est le vecteur de sortie de dimension $p$
- $A$ est la matrice de dynamique de dimension $n \times n$
- $B$ est la matrice de commande de dimension $n \times m$
- $C$ est la matrice d'observation de dimension $p \times n$
- $D$ est la matrice d'action directe de dimension $p \times m$

# Représentation d'état en temps discret d'un processus linéaire


$$\begin{cases} X_{k+1} & = AX_k + BU_k \\ Y_k & = CX_k + DU_k \end{cases}$$

- $X_k$ est le vecteur d'état de dimension $n$
- $U_k$ est le vecteur de commande de dimension $m$
- $Y_k$ est le vecteur de sortie de dimension $p$
- $A$ est la matrice de dynamique de dimension $n \times n$
- $B$ est la matrice de commande de dimension $n \times m$
- $C$ est la matrice d'observation de dimension $p \times n$
- $D$ est la matrice d'action directe de dimension $p \times m$

# Commande linéaire quadratique

Processus linéaire :

$$\begin{cases} X_{k+1} & = AX_k + BU_k \\ Y_k & = CX_k + DU_k \end{cases}$$

# Commande linéaire quadratique

Processus linéaire :

$$\begin{cases} X_{k+1} & = AX_k + BU_k \\ Y_k & = CX_k + DU_k \end{cases}$$

Loi de commande / politique :

$$U_k = \pi(X_k) = K.X_k$$

$K$ est une matrice dimension $m \times n$.

# Commande linéaire quadratique

Processus linéaire :

$$\begin{cases} X_{k+1} & = AX_k + BU_k \\ Y_k & = CX_k + DU_k \end{cases}$$

Loi de commande / politique :

$$U_k = \pi(X_k) = K.X_k$$

$K$ est une matrice dimension $m \times n$.

Fonction de coût quadratique :

$$V^K(X) = \sum_{k=0}^{\infty} X_k' E X_k + U_k' F U_k, \quad X_0 = X$$

$E$ et $F$ sont des matrices symétriques définies positives.

# Commande linéaire quadratique

Processus linéaire :

$$\begin{cases} X_{k+1} & = AX_k + BU_k \\ Y_k & = CX_k + DU_k \end{cases}$$

Objectif : déterminer $K$ minimisant $V^K(X)$.

Loi de commande / politique :

$$U_k = \pi(X_k) = K.X_k$$

$K$ est une matrice dimension $m \times n$.

Fonction de coût quadratique :

$$V^K(X) = \sum_{k=0}^{\infty} X_k' E X_k + U_k' F U_k, \quad X_0 = X$$

$E$ et $F$ sont des matrices symétriques définies positives.

# Commande linéaire quadratique

Processus linéaire :

$$\begin{cases} X_{k+1} & = AX_k + BU_k \\ Y_k & = CX_k + DU_k \end{cases}$$

Loi de commande / politique :

$$U_k = \pi(X_k) = K.X_k$$

$K$ est une matrice dimension $m \times n$.

Fonction de coût quadratique :

$$V^K(X) = \sum_{k=0}^{\infty} X_k' E X_k + U_k' F U_k, \quad X_0 = X$$

$E$ et $F$ sont des matrices symétriques définies positives.

**Objectif : déterminer $K$ minimisant $V^K(X)$.**

Si $(A, B)$ est commandable, on a :

$$K^* = -(B'PB + F)^{-1} B'PA$$

# Commande linéaire quadratique

Processus linéaire :

$$\begin{cases} X_{k+1} &= AX_k + BU_k \\ Y_k &= CX_k + DU_k \end{cases}$$

Loi de commande / politique :

$$U_k = \pi(X_k) = K.X_k$$

$K$ est une matrice dimension $m \times n$.

Fonction de coût quadratique :

$$V^K(X) = \sum_{k=0}^{\infty} X_k' E X_k + U_k' F U_k, \quad X_0 = X$$

$E$ et $F$ sont des matrices symétriques définies positives.

Objectif : déterminer $K$ minimisant $V^K(X)$.

Si $(A, B)$ est commandable, on a :

$$K^* = -(B'PB + F)^{-1} B'PA$$

$P$ est la matrice de coût solution de l'équation :

$$E + A'PA - A'PB(B'PB + F)^{-1} B'PA = P$$

# Commande linéaire quadratique

Processus linéaire :

$$\begin{cases} X_{k+1} & = AX_k + BU_k \\ Y_k & = CX_k + DU_k \end{cases}$$

Loi de commande / politique :

$$U_k = \pi(X_k) = K.X_k$$

$K$ est une matrice dimension $m \times n$.

Fonction de coût quadratique :

$$V^K(X) = \sum_{k=0}^{\infty} X_k' E X_k + U_k' F U_k, \quad X_0 = X$$

$E$ et $F$ sont des matrices symétriques définies positives.

**Objectif : déterminer $K$ minimisant $V^K(X)$.**

Si $(A, B)$ est commandable, on a :

$$K^* = -(B'PB + F)^{-1} B'PA$$

$P$ est la matrice de coût solution de l'équation :

$$E + A'PA - A'PB(B'PB + F)^{-1} B'PA = P$$

Fonction de valeur :

$$V^*(X) = X'PX$$

# Plan de la présentation

# Fonction de valeur d'action

Processus linéaire :

$$\begin{cases} X_{k+1} & = AX_k + BU_k \\ Y_k & = CX_k + DU_k \end{cases}$$

# Fonction de valeur d'action

Processus linéaire :

$$\begin{cases} X_{k+1} & = AX_k + BU_k \\ Y_k & = CX_k + DU_k \end{cases}$$

Fonction de valeur :

$$V(X) = X'PX$$

# Fonction de valeur d'action

Processus linéaire :

$$\begin{cases} X_{k+1} & = AX_k + BU_k \\ Y_k & = CX_k + DU_k \end{cases}$$

Fonction de valeur :

$$V(X) = X'PX$$

Fonction de valeur d'action pour le critère $\gamma$-pondéré (Bradtke, 1994) :

$$\begin{aligned} Q(X,U) &= \begin{bmatrix} X & U \end{bmatrix} \begin{bmatrix} E + \gamma A'PA & \gamma A'PB \\ \gamma B'PA & F + \gamma B'PB \end{bmatrix} \begin{bmatrix} X \\ U \end{bmatrix} \\ &= \begin{bmatrix} X & U \end{bmatrix} \begin{bmatrix} H_{11} & H_{12} \\ H'_{12} & H_{22} \end{bmatrix} \begin{bmatrix} X \\ U \end{bmatrix} \\ &= \begin{bmatrix} X & U \end{bmatrix} H \begin{bmatrix} X \\ U \end{bmatrix} \\ &= \phi'(X,U)\theta_t \end{aligned}$$

# LSPI pour processus linéaires avec coût quadratique (Bradtke, 1994)

1. Recueil des échantillons :

$$X_0 \xrightarrow{U_0} X_1 \xrightarrow{U_1} X_2 \xrightarrow{U_2} \dots \xrightarrow{U_{N-1}} X_N$$

$$U_k = \pi_t(X_k) + e_k = K_t X_k + e_k$$

où $e_k$ est un signal d'exploration.

# LSPI pour processus linéaires avec coût quadratique (Bradtke, 1994)

1. Recueil des échantillons :

$$X_0 \xrightarrow{U_0} X_1 \xrightarrow{U_1} X_2 \xrightarrow{U_2} \dots \xrightarrow{U_{N-1}} X_N$$

$$U_k = \pi_t(X_k) + e_k = K_t X_k + e_k$$

où $e_k$ est un signal d'exploration.

2. Calcul de la fonction de valeur d'action $Q^{\pi_t}$ (LSTDQ) :

$$\theta_{t+1} = \left[ \sum_{k=0}^{M-1} \phi(X_k, U_k)[\phi(X_k, U_k) - \gamma\phi(X_{k+1}, \pi_t(X_{k+1}))]' \right]^{-1} \left[ \sum_{k=0}^{M-1} \phi(X_k, U_k) r(X_k, U_k) \right]$$

# LSPI pour processus linéaires avec coût quadratique (Bradtke, 1994)

1. Recueil des échantillons :

$$X_0 \xrightarrow{U_0} X_1 \xrightarrow{U_1} X_2 \xrightarrow{U_2} \dots \xrightarrow{U_{N-1}} X_N$$

$$U_k = \pi_t(X_k) + e_k = K_t X_k + e_k$$

où $e_k$ est un signal d'exploration.

2. Calcul de la fonction de valeur d'action $Q^{\pi_t}$ (LSTDQ) :

$$\theta_{t+1} = \left[\sum_{k=0}^{M-1} \phi(X_k, U_k)[\phi(X_k, U_k) - \gamma\phi(X_{k+1}, \pi_t(X_{k+1}))]'\right]^{-1} \left[\sum_{k=0}^{M-1} \phi(X_k, U_k) r(X_k, U_k)\right]$$

3. Amélioration de la politique :

$$\forall X, \quad \pi_{t+1}(X) = \arg\min_V Q_{t+1}(X, V) = \arg\min_V \phi'(X, V)\theta_{t+1} = -H_{22}^{-1} H_{21} X$$

donc le nouveau correcteur est :

$$K_{t+1} = -H_{22}^{-1} H_{21}$$

# LSPI pour processus linéaires avec coût quadratique (Bradtke, 1994)

**début**
 Initialiser le correcteur initial $K_0$ (doit être stabilisant)
 $t \leftarrow 0$
 **répéter**
  $\Phi \leftarrow 0$
  $\rho \leftarrow 0$
  Obtenir l'état initial $X_0$ du processus
  **pour** $k$ *de* $0$ *à* $M-1$ **faire**
   $U_k \leftarrow K_t X_k + e_k$ (où $e_k$ est un signal d'exploration); `(partie en ligne)`
   Appliquer $U_k$ au processus et attendre $X_{k+1}$
   $\Phi \leftarrow \Phi + \phi(X_k, U_k)[\phi(X_k, U_k) - \gamma\phi(X_{k+1}, K_t X_{k+1})]$
   $\rho \leftarrow \rho + \phi(X_k, U_k) r(X_k, U_k)$
  **fin**
  $\theta_{t+1} \leftarrow \Phi^{-1}\rho$; `(partie hors ligne)`
  Trouver les coefficients de $H$ correspondants aux paramètres $\theta_{t+1}$
  $K_{t+1} \leftarrow -H_{22}^{-1} H_{21}$
  $t \leftarrow t+1$
 **jusqu'à** $\theta_t \approx \theta_{t-1}$
**fin**

# LSPI pour processus linéaires avec coût quadratique (Bradtke, 1994)

Avantages de la méthode :

# LSPI pour processus linéaires avec coût quadratique (Bradtke, 1994)

Avantages de la méthode :

- Ne nécessite pas de modèle

# LSPI pour processus linéaires avec coût quadratique (Bradtke, 1994)

Avantages de la méthode :

- Ne nécessite pas de modèle
- Le nombre de paramètres de la fonction de valeur est faible $(n+m)(n+m+1)/2$.

# LSPI pour processus linéaires avec coût quadratique (Bradtke, 1994)

Avantages de la méthode :

- Ne nécessite pas de modèle
- Le nombre de paramètres de la fonction de valeur est faible $(n+m)(n+m+1)/2$.
- Le calcul de la politique gloutonne est analytique et ne nécessite donc aucune méthode d'optimisation spécifique

# LSPI pour processus linéaires avec coût quadratique (Bradtke, 1994)

Avantages de la méthode :

- Ne nécessite pas de modèle
- Le nombre de paramètres de la fonction de valeur est faible $(n+m)(n+m+1)/2$.
- Le calcul de la politique gloutonne est analytique et ne nécessite donc aucune méthode d'optimisation spécifique
- Si le couple $(A, B)$ est commandable, si $K_0$ est stabilisant et si la commande appliquée au processus comporte une part d'exploration suffisante, il existe un nombre fini d'échantillons $M$ tel que la politique $K_t$ converge vers la politique optimale (Bradtke, 1994)

# LSPI pour processus linéaires avec coût quadratique (Bradtke, 1994)

Avantages de la méthode :

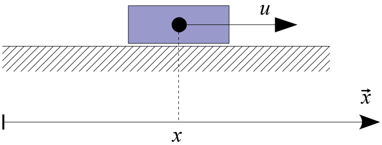
- Ne nécessite pas de modèle
- Le nombre de paramètres de la fonction de valeur est faible $(n+m)(n+m+1)/2$.
- Le calcul de la politique gloutonne est analytique et ne nécessite donc aucune méthode d'optimisation spécifique
- Si le couple $(A, B)$ est commandable, si $K_0$ est stabilisant et si la commande appliquée au processus comporte une part d'exploration suffisante, il existe un nombre fini d'échantillons $M$ tel que la politique $K_t$ converge vers la politique optimale (Bradtke, 1994)

Algorithme très adapté à la commande adaptative de systèmes linéaires stables

# Plan de la présentation

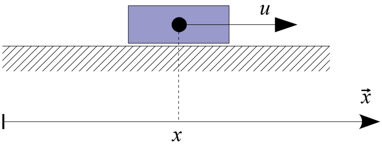
# Processus simulé

Processus simulé :



# Processus simulé

Processus simulé :

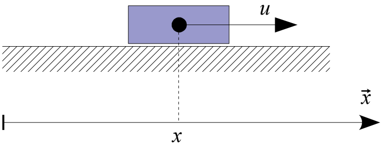



Modèle :

$$\begin{cases} X_{k+1} = \begin{bmatrix} 1.0000 & 0 & 0 \\ 0 & 0.9578 & 0 \\ 0 & 0 & 0.6756 \end{bmatrix} X_k + \begin{bmatrix} -0.0622 \\ 0.0621 \\ 0.0598 \end{bmatrix} u_k \\ x_k = \begin{bmatrix} -22.3632 & -24.6381 & 2.3784 \end{bmatrix} X_k \end{cases}$$

# Processus simulé

Processus simulé :




Modèle :

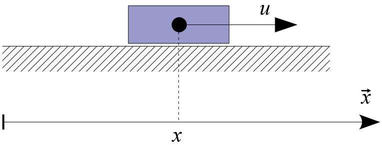
$$\begin{cases} X_{k+1} = \begin{bmatrix} 1.0000 & 0 & 0 \\ 0 & 0.9578 & 0 \\ 0 & 0 & 0.6756 \end{bmatrix} X_k + \begin{bmatrix} -0.0622 \\ 0.0621 \\ 0.0598 \end{bmatrix} u_k \\ x_k = \begin{bmatrix} -22.3632 & -24.6381 & 2.3784 \end{bmatrix} X_k \end{cases}$$

Fonction de coût immédiat :

$$r(x,u) = x^2 + 10u^2$$

# Processus simulé

Processus simulé :




Modèle :

$$\begin{cases} X_{k+1} = \begin{bmatrix} 1.0000 & 0 & 0 \\ 0 & 0.9578 & 0 \\ 0 & 0 & 0.6756 \end{bmatrix} X_k + \begin{bmatrix} -0.0622 \\ 0.0621 \\ 0.0598 \end{bmatrix} u_k \\ x_k = \begin{bmatrix} -22.3632 & -24.6381 & 2.3784 \end{bmatrix} X_k \end{cases}$$

Fonction de coût immédiat :

$$r(x,u) = x^2 + 10u^2$$

A l'instant $k=0$, l'objet est placé à 20 mm de l'origine sans vitesse initiale.
Un essai dure 300 pas de calculs (30 pas par seconde).

# Commande linéaire quadratique

Correcteur optimal :

$$K^* = \begin{bmatrix} -6.5520 & -4.1332 & 0.0218 \end{bmatrix}$$

# Commande linéaire quadratique

Correcteur optimal :

$$K^* = \begin{bmatrix} -6.5520 & -4.1332 & 0.0218 \end{bmatrix}$$

Fonction de valeur :

$$V^*(X_0) = 5.3135\text{e}3, \qquad X_0 = \begin{bmatrix} -0.8943 & 0.0000 & 0.0000 \end{bmatrix}'$$

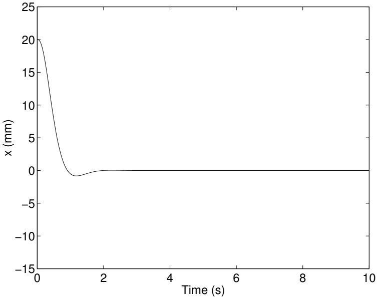
# Commande linéaire quadratique

Correcteur optimal :

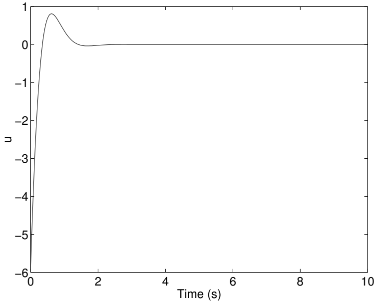
$$K^* = \begin{bmatrix} -6.5520 & -4.1332 & 0.0218 \end{bmatrix}$$

Fonction de valeur :

$$V^*(X_0) = 5.3135\text{e}3, \qquad X_0 = \begin{bmatrix} -0.8943 & 0.0000 & 0.0000 \end{bmatrix}'$$




(a) Position de l'objet en fonction du temps




(b) Commande en fonction du temps

FIGURE: Résultats de simulation avec le correcteur LQ.

# Apprentissage *ex nihilo*

Le système étant simplement stable, nous pouvons choisir :

$$K_0 = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}'$$

# Apprentissage *ex nihilo*

Le système étant simplement stable, nous pouvons choisir :

$$K_0 = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}'$$

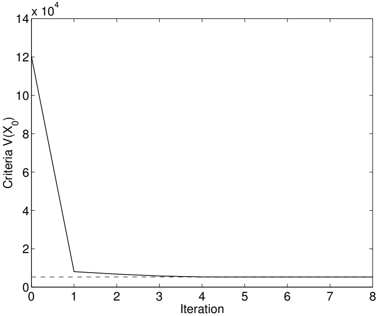
- Signal d'exploration $e_k$ = bruit gaussien $\mathcal{N}(0, 5)$
- $\gamma$ est fixé à 1 (l'horizon temporel est fini)
- Pour évaluer le critère $V(X_0)$, on utilise $e_k = 0$

# Apprentissage *ex nihilo*

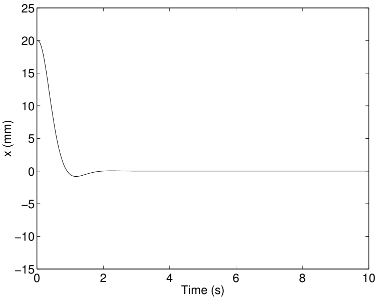
Le système étant simplement stable, nous pouvons choisir :

$$K_0 = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}'$$

- Signal d'exploration $e_k$ = bruit gaussien $\mathcal{N}(0,5)$
- $\gamma$ est fixé à 1 (l'horizon temporel est fini)
- Pour évaluer le critère $V(X_0)$, on utilise $e_k = 0$




(a) Critère $V(X_0)$ en fonction des itérations (en pointillés la valeur optimale du critère)

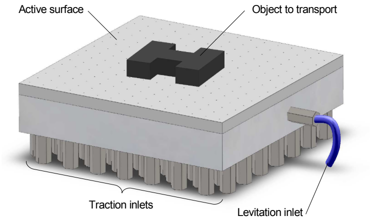


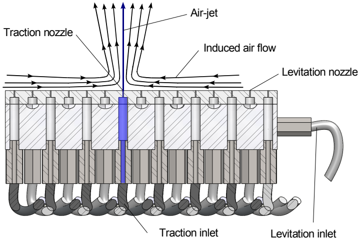

(b) Position de l'objet en fonction du temps

FIGURE: Résultats de simulation de LSPI initialisé avec une politique nulle.

# Plan de la présentation

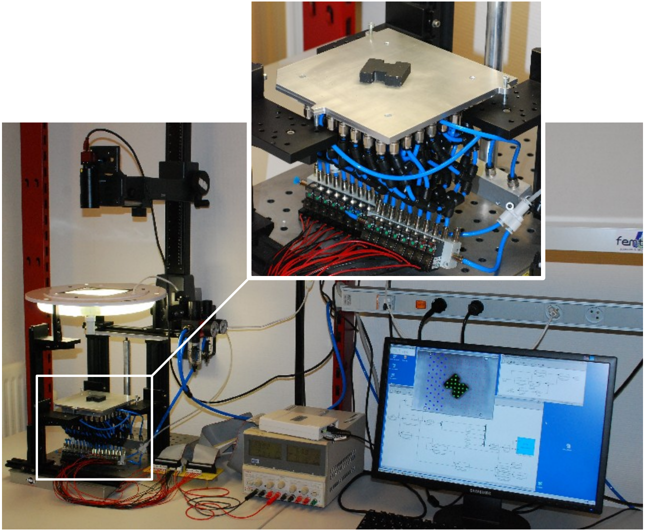
# Surface de manipulation à coussin d'air actif



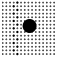
# Surface de manipulation à coussin d'air actif

Dispositif expérimental :

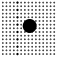


Définition des commandes :

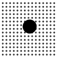
$u = -2$



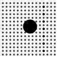
$u = -1$



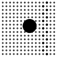
$u = 0$



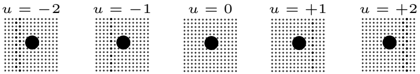
$u = +1$



$u = +2$



Définition des commandes :



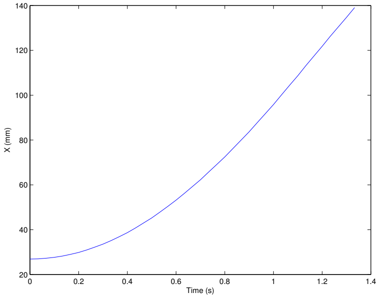
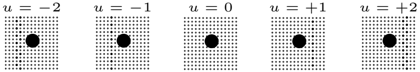




FIGURE: Position de l'objet en fonction du temps pour une commande constante $u = 2$.

Définition des commandes :



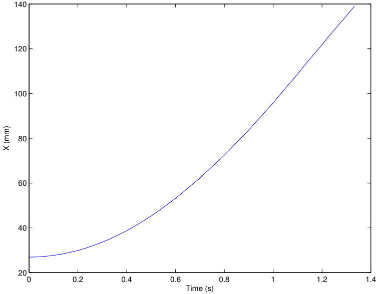
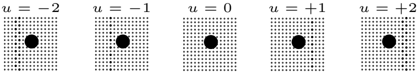




FIGURE: Position de l'objet en fonction du temps pour une commande constante $u = 2$.

Identification paramétrique :

- Système d'ordre 2 + intégrateur
- Entrée $U$ = commande $u$ (ci-contre)
- Sortie $Y$ = abscisse $x$ de l'objet
- Echantillonnage = 30 Hz

Définition des commandes :



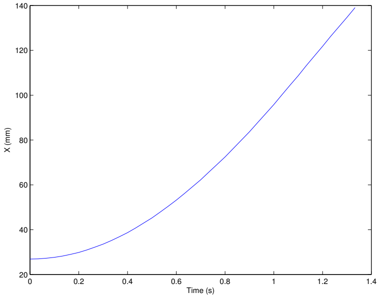




FIGURE: Position de l'objet en fonction du temps pour une commande constante $u = 2$.

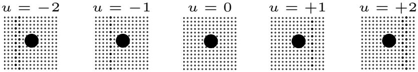
Identification paramétrique :

- Système d'ordre 2 + intégrateur
- Entrée $U$ = commande $u$ (ci-contre)
- Sortie $Y$ = abscisse $x$ de l'objet
- Echantillonnage = 30 Hz

Modèle :

$$\begin{cases} X_{k+1} = \begin{bmatrix} 1.0000 & 0 & 0 \\ 0 & 0.9578 & 0 \\ 0 & 0 & 0.6756 \end{bmatrix} X_k + \begin{bmatrix} -0.0622 \\ 0.0621 \\ 0.0598 \end{bmatrix} u_k \\ x_k = \begin{bmatrix} -22.3632 & -24.6381 & 2.3784 \end{bmatrix} X_k \end{cases}$$

Définition des commandes :



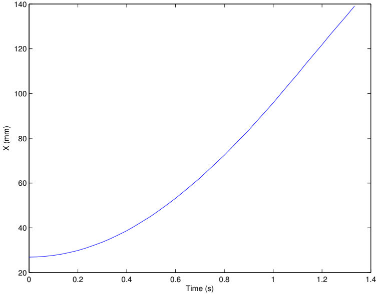




FIGURE: Position de l'objet en fonction du temps pour une commande constante $u = 2$.

Identification paramétrique :

- Système d'ordre 2 + intégrateur
- Entrée $U$ = commande $u$ (ci-contre)
- Sortie $Y$ = abscisse $x$ de l'objet
- Echantillonnage = 30 Hz

Modèle :

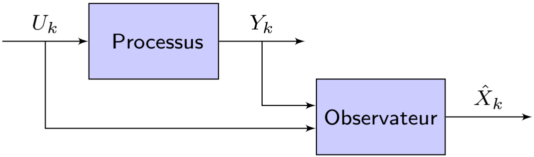
$$\begin{cases} X_{k+1} = \begin{bmatrix} 1.0000 & 0 & 0 \\ 0 & 0.9578 & 0 \\ 0 & 0 & 0.6756 \end{bmatrix} X_k + \begin{bmatrix} -0.0622 \\ 0.0621 \\ 0.0598 \end{bmatrix} u_k \\ x_k = \begin{bmatrix} -22.3632 & -24.6381 & 2.3784 \end{bmatrix} X_k \end{cases}$$

Fonction de coût immédiat

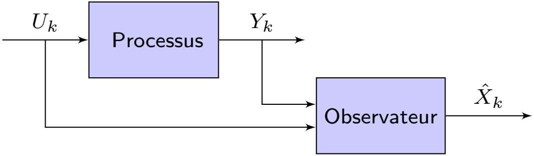
$$r(x, u) = x^2 + 10u^2$$

# Reconstruction de l'état

# Reconstruction de l'état



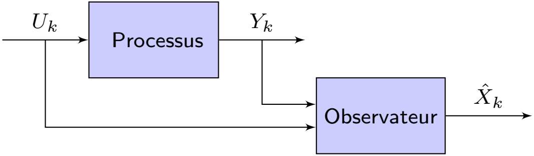
# Reconstruction de l'état




Observateur de Luenberger :

$$\begin{cases} \hat{X}_{k+1} & = A\hat{X}_k + BU_k + L(Y_k - \hat{Y}_k) \\ \hat{Y}_k & = C\hat{X}_k \end{cases}$$

# Reconstruction de l'état




Observateur de Luenberger :

$$\begin{cases} \hat{X}_{k+1} & = A\hat{X}_k + BU_k + L(Y_k - \hat{Y}_k) \\ \hat{Y}_k & = C\hat{X}_k \end{cases}$$

Coefficients de l'observateur choisis en simulation de façon à minimiser l'impact de l'observateur sur le critère $V(X_0)$.

# Plan de la présentation

# Commande linaire quadratique

Correcteur optimal (calculé à l'aide du modèle) :

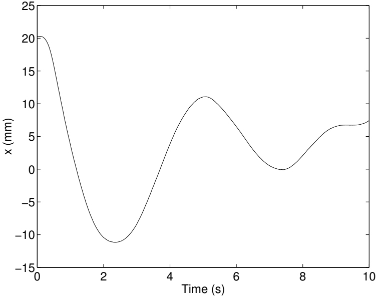
$$K^* = \begin{bmatrix} -6.5520 & -4.1332 & 0.0218 \end{bmatrix}$$

# Commande linaire quadratique

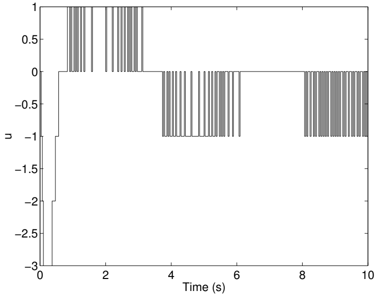
Correcteur optimal (calculé à l'aide du modèle) :

$$K^* = \begin{bmatrix} -6.5520 & -4.1332 & 0.0218 \end{bmatrix}$$

Fonction de valeur théorique : $V^*(X_0) = 5.3135e3$
Fonction de valeur mesurée : $V(X_0) = 2.3917e4$




(a) Position de l'objet en fonction du temps




(b) Commande en fonction du temps

FIGURE: Résultats expérimentaux avec le correcteur LQ.

# Apprentissage *ex nihilo*

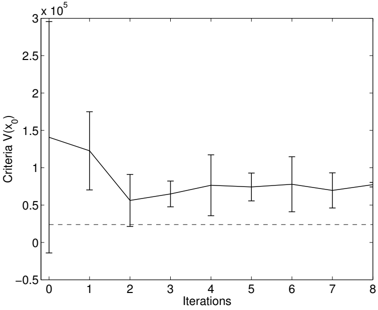
Le système étant simplement stable, nous pouvons choisir :

$$K_0 = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}'$$

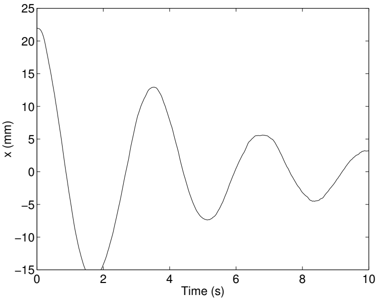
# Apprentissage *ex nihilo*

Le système étant simplement stable, nous pouvons choisir :

$$K_0 = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}'$$




(a) Critère $V(X_0)$ en fonction des itérations (moyenne sur 6 essais, en pointillés la valeur du critère pour la commande LQ)




(b) Position de l'objet en fonction du temps

FIGURE: Résultats expérimentaux de LSPI initialisé avec une politique nulle.

# Fonction de formatage

L'algorithme peut générer des commandes irréalisables :

$$U_k \leftarrow K_t X_k + e_k$$

# Fonction de formatage

L'algorithme peut générer des commandes irréalisables :

$$U_k \leftarrow K_t X_k + e_k$$

Nous avons donc remplacé cette ligne par :

$$U_k \leftarrow f(K_t X_k + e_k)$$

$f$ étant une fonction de formatage de la commande. $f$ est spécifique au système.

# Fonction de formatage

L'algorithme peut générer des commandes irréalisables :

$$U_k \leftarrow K_t X_k + e_k$$

Nous avons donc remplacé cette ligne par :

$$U_k \leftarrow f(K_t X_k + e_k)$$

$f$ étant une fonction de formatage de la commande. $f$ est spécifique au système.

Pour le système étudié ici, la fonction de formatage est définie par :

$$f(u) = \begin{cases} -5 & \text{si } u < -5 \\ 5 & \text{si } u > 5 \\ \text{Arrondi}(u) & \text{si } -5 \leq u \leq 5 \end{cases}$$

# Apprentissage *ex nihilo* avec fonction de formatage

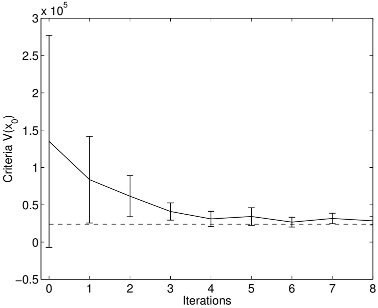
Le système étant simplement stable, nous pouvons choisir :

$$K_0 = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}'$$

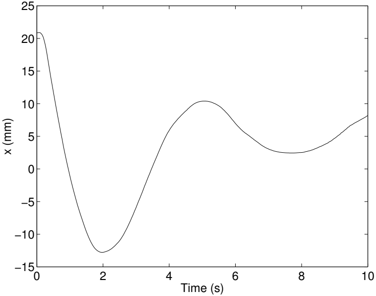
# Apprentissage *ex nihilo* avec fonction de formatage

Le système étant simplement stable, nous pouvons choisir :

$$K_0 = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}'$$




(a) Critère $V(X_0)$ en fonction des itérations (moyenne sur 7 essais, en pointillés la valeur du critère pour la commande LQ)




(b) Position de l'objet en fonction du temps

FIGURE: Résultats expérimentaux de LSPI initialisé avec une politique nulle.

# Apprentissage *ex nihilo* avec fonction de formatage

Itération 0 avec exploration

Itération 8 sans exploration

# Plan de la présentation

# Conclusions et perspectives

Conclusions :

# Conclusions et perspectives

Conclusions :

- Méthode d'apprentissage de commande optimale sans modèle

# Conclusions et perspectives

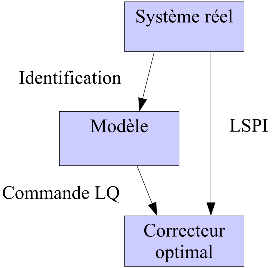
Conclusions :

- Méthode d'apprentissage de commande optimale sans modèle
- Convergence garantie et rapide $\Rightarrow$ commande adaptative

# Conclusions et perspectives

Conclusions :

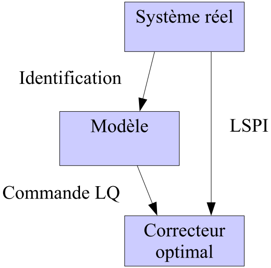
- Méthode d'apprentissage de commande optimale sans modèle
- Convergence garantie et rapide ⇒ commande adaptative
- Robuste aux non-linéarités du système réel et aux perturbations extérieures



# Conclusions et perspectives

Conclusions :

- Méthode d'apprentissage de commande optimale sans modèle
- Convergence garantie et rapide ⇒ commande adaptative
- Robuste aux non-linéarités du système réel et aux perturbations extérieures

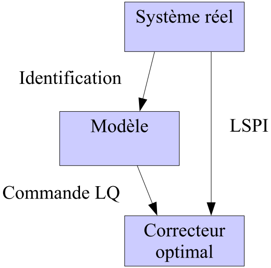



Problèmes et perspectives associées :

# Conclusions et perspectives

Conclusions :

- Méthode d'apprentissage de commande optimale sans modèle
- Convergence garantie et rapide ⇒ commande adaptative
- Robuste aux non-linéarités du système réel et aux perturbations extérieures



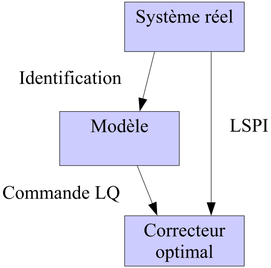

Problèmes et perspectives associées :

- Nécessite un observateur si l'état n'est pas directement observé

# Conclusions et perspectives

Conclusions :

- Méthode d'apprentissage de commande optimale sans modèle
- Convergence garantie et rapide ⇒ commande adaptative
- Robuste aux non-linéarités du système réel et aux perturbations extérieures



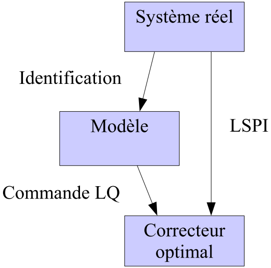

Problèmes et perspectives associées :

- Nécessite un observateur si l'état n'est pas directement observé
- Exploration plus intelligente ⇒ apprentissage actif

# Conclusions et perspectives

Conclusions :

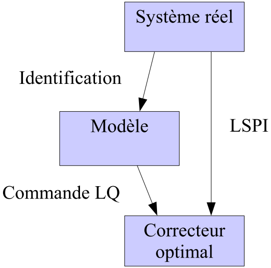
- Méthode d'apprentissage de commande optimale sans modèle
- Convergence garantie et rapide ⇒ commande adaptative
- Robuste aux non-linéarités du système réel et aux perturbations extérieures




Problèmes et perspectives associées :

- Nécessite un observateur si l'état n'est pas directement observé
- Exploration plus intelligente ⇒ apprentissage actif
- Améliorations techniques de la surface

# Conclusions et perspectives




Conclusions :

- Méthode d'apprentissage de commande optimale sans modèle
- Convergence garantie et rapide ⇒ commande adaptative
- Robuste aux non-linéarités du système réel et aux perturbations extérieures

Problèmes et perspectives associées :

- Nécessite un observateur si l'état n'est pas directement observé
- Exploration plus intelligente ⇒ apprentissage actif
- Améliorations techniques de la surface
- Commande individuelle des jets d'air : *système à 115 entrées et 3 sorties* $(x, y, \theta)$ ! ! !

Merci !

Des questions ?