

# Commande des Systèmes par Platitude

F. Rotella

Ecole Nationale d'Ingénieurs de Tarbes

rotella@enit.fr

26 mars 2004

# 1 Introduction

1. Introduction
2. Définitions et conséquences
3. Exemples pratiques
4. Un peu de formalisme
5. Critères de platitude
6. Cas des systèmes linéaires
7. SPD
8. Conclusion

## Introduction

- La notion de platitude a été introduite en 1992 par M. Fliess, J. Lévine, P. Martin et P. Rouchon;
- nouveau point de vue en Automatique qui remet au centre de la conception d'une commande :
  - la tâche que doit effectuer un processus sous la forme d'une **trajectoire** exécutable;
  - la fonction essentielle du **bouclage** qui est de corriger les erreurs et rejeter les perturbations;
- développée initialement dans le cadre des systèmes non linéaires de dimension finie, elle peut être appliquée dans pratiquement toutes les autres situations : systèmes à retards, dimension infinie, *systèmes linéaires*, ...

## Exemple de sensibilisation

Considérons un robot manipulateur décrit par le modèle dynamique en les variables articulaires  $q$  :

$$H(q(t))\ddot{q}(t) + \text{NL}(q(t), \dot{q}(t)) = \Gamma(t),$$

où :

- $H(q)$  est la matrice d'inertie (toujours définie positive);
- $\text{NL}(q(t), \dot{q}(t))$  est le vecteur des non linéarités (Coriolis, gravité, entraînement, ... );
- $\Gamma(t)$  est le vecteur des couples articulaires.

Dans un premier temps, on considère que les commandes sont les couples articulaires.

## Couple calculé

La méthode intuitive du **couple calculé** donne le bouclage statique :

$$\Gamma(t) = H(q(t))v(t) + \text{NL}(q(t), \dot{q}(t)), \quad (1)$$

qui conduit au système linéaire découplé :

$$\ddot{q}(t) = v(t).$$

Ainsi l'on voit que si l'on désire faire exécuter une trajectoire au robot sous la forme d'un déplacement articulaire  $q_d(t)$  sur un intervalle de temps  $t \in \tau = [t_I, t_F]$  on peut calculer les commandes à imposer sous la forme :

$$\Gamma_d(t) = H(q_d(t))\ddot{q}_d(t) + \text{NL}(q_d(t), \dot{q}_d(t)).$$

Cela constituera le premier point important de la platitude.

## Suivi de trajectoire

Deuxième point important : si l'on impose  $\Gamma_d(t)$  au niveau des commandes d'axes du robot, l'imprécision des valeurs des paramètres du modèles, les perturbations et les conditions initiales mal connues font que la trajectoire désirée  $q_d(t)$  ne va pas être exactement exécutée.

Si dans (1) on fait :

$$v(t) = \ddot{q}_d(t) + k_1(\dot{q}_d(t) - \dot{q}(t)) + k_0(q_d(t) - q(t)),$$

où  $k_1$  et  $k_0$  sont deux matrices diagonales positives alors la commande :

$$\Gamma(t) = H(q(t))(\ddot{q}_d(t) + k_1(\dot{q}_d(t) - \dot{q}(t)) + k_0(q_d(t) - q(t))) + \text{NL}(q(t), \dot{q}(t)),$$

conduit à une erreur  $q_d(t) - q(t)$  qui tendra asymptotiquement vers 0.

## Prise en compte des actionneurs

En robotique, les actionneurs sont de la forme :

$$\Gamma_i(t) = \frac{k}{p(p^2 + a_1p + a_0)} U_i(t),$$

$U_i(t)$  : commandes envoyées aux cartes d'axes. Comme :

$$U_i(t) = \frac{1}{k} (\Gamma_i^{(3)}(t) + a_1 \Gamma_i^{(2)}(t) + a_0 \Gamma_i^{(1)}(t)),$$

on arrive finalement à :

$$U(t) = \Phi(q^{(5)}(t), \dots, \dot{q}(t), q(t)).$$

On peut appliquer les principes de commande précédents.

## Trajectoire dans l'espace opérationnel

Soit  $X(t)$  les coordonnées opérationnelles du robot : ensemble des variables indépendantes qui définissent la position et l'orientation de l'organe terminal.

En robotique, on cherche à se déplacer dans l'environnement du robot : les trajectoires sont définies dans l'espace opérationnel sous la forme  $X_d(t)$ .

Modèle géométrique :

$$g(X) = f(q),$$

- modèle géométrique direct (local) :  $X = F(q)$ ;
- modèle géométrique inverse (local) :  $q = F^{-1}(X) = G(X)$ .

## Commande opérationnelle

Comme :  $\dot{q} = G_X \dot{X}, \dots, q^{(k)} = \phi(X, \dots, X^{(k)}), \dots$

$$U(t) = \Psi(X^{(5)}(t), \dots, \dot{X}(t), X(t)).$$

Ce qui permet de faire la commande directement dans l'espace opérationnel :

- par génération de trajectoire (**boucle ouverte**) :

$$U_d(t) = \Psi(X_d^{(5)}(t), \dots, \dot{X}_d(t), X_d(t));$$

- par poursuite de trajectoire (**boucle fermée**) :

$$U(t) = \Psi(X_d^{(5)}(t) + \sum_{i=1}^4 k_i (X_d^{(i)}(t) - X^{(i)}(t)), \dots, \dot{X}(t), X(t)).$$

## Points mis en évidence

- on a paramétré les solutions sans intégrer de système différentiel;
- ce paramétrage n'a pas été remis en cause par des mises en série (actionneurs, fonctions de sorties);
- il permet de construire :
  - une commande boucle ouverte  $\implies$  *génération de trajectoire*;
  - une commande boucle fermée  $\implies$  *poursuite de trajectoire*.

L'ensemble des variables permettant de paramétrer les autres constitue :

*une sortie plate du système*

Constatation : la sortie plate n'est pas unique.

## Quelques liens sur Internet

Documents ou simulations téléchargeables :

- <http://cas.ensmp.fr/~petit/ensta/>  
P. MARTIN, N. PETIT : Commande des systèmes non linéaires, le point de vue des systèmes plats;
- <http://cas.ensmp.fr/~levine/Enseignement/index.html>  
J. LÉVINE : Introduction à la commande non linéaire;
- <http://cas.ensmp.fr/~rouchon/>  
P. ROUCHON : exemples, cours et transparents
- <http://math.polytechnique.fr/xups/vol99.html>  
P. MARTIN, P. ROUCHON : Systèmes plats, planification et suivi de trajectoires.  
J.M. CORON : Quelques résultats sur la commandabilité ...

## 2 Définitions et conséquences

1. Introduction
2. Définitions et conséquences
3. Exemples pratiques
4. Un peu de formalisme
5. Critères de platitude
6. Cas des systèmes linéaires
7. SPD
8. Conclusion

## Définition d'un système plat

Un système défini par l'équation :

$$\Phi(\dot{x}(t), x(t), u(t)) = 0,$$

où  $u(t)$  est la commande, est **plat** s'il existe un vecteur  $z(t)$  et trois entiers  $\alpha$ ,  $\beta$ , et  $\delta$  tels que :

$$z(t) = h(x(t), u(t), u^{(1)}(t), \dots, u^{(\delta)}(t)), \quad (2)$$

dont les composantes soient différentiellement indépendantes;

et tel que :

$$x(t) = A(z(t), z^{(1)}(t), \dots, z^{(\alpha)}(t)), \quad (3)$$

$$u(t) = B(z(t), z^{(1)}(t), \dots, z^{(\beta)}(t)). \quad (4)$$

## Remarques (1)

1. Cette définition n'est pas restreinte aux modèles d'état, mais à tout modèle de la forme :

$$\Phi(x^{(n)}(t), \dots, \dot{x}(t), x(t), u^{(m)}(t), \dots, u(t)) = 0;$$

2. Pour une sortie  $y(t) = \Psi(x(t), u(t), \dots, u^{(p)}(t))$ , on obtient :

$$y = \mathbf{C}(z, \dots, z^{(\gamma)}). \quad (5)$$

3. *mutatis mutandis* on peut définir la même notion dans le cadre discret :

on remplace alors dérivation par avance.

**Sortie plate**

$$z(t) = h(x(t), u(t), u^{(1)}(t), \dots, u^{(\delta)}(t))$$

Une sortie plate est un ensemble de variables qui permet de paramétrer toutes les autres variables du système,

*ou bien*

Regroupe les variables libres du système,

C'est une variable endogène du système.

$\delta$ -platitude : lorsque  $z(t)$  dépend de  $u(t), \dots, u^{(\delta)}(t)$ .

## Remarques (2)

- la relation (2) s'appelle la relation d'endogénéité : la sortie plate n'est construite qu'à partir des variables du système;
- un bouclage qui n'utilise qu'elle sera un bouclage **endogène**;
- en dimension infinie, nous verrons que la relation d'endogénéité est moins “nette”;
- le nombre de sorties plates doit être égal au nombre de commandes : les commandes doivent être “libres”;
- une sortie plate n'est pas unique :  $\xi = \Phi(z)$ ,  $\Phi$  inversible;
- la relation (3) est nécessaire car elle permet de vérifier que  $z(t)$  est effectivement la sortie plate d'un système.

**Exemple 1**

Soit le système :

$$\dot{x}_1 = x_2, \dot{x}_2 = u,$$

si on définit les variables :

1.  $z(t) = x_2(t)$ , malgré  $u(t) = \dot{z}(t)$ , elle ne peut être considérée comme sortie plate car on a seulement :

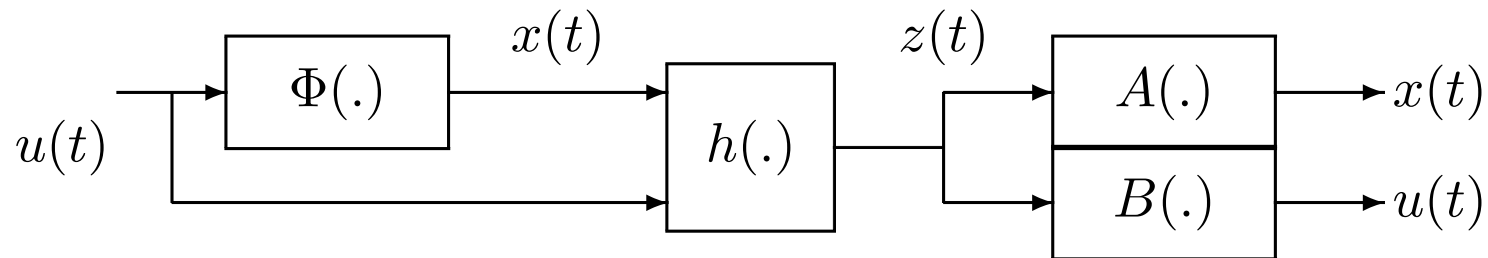
$$x_1(t) = x_1(t_0) + \int_{t_0}^t z(\tau) d\tau.$$

2.  $z(t) = x_1(t)$ , alors :

$$x_2(t) = \dot{z}(t) \text{ et } u(t) = \ddot{z}(t),$$

qui indique que  $x_1$  est une sortie plate de ce système.

## Vers une inversion des systèmes



Première conséquence : planification de trajectoire.

Si on désire obtenir la trajectoire :

$$z_d(t) \text{ pour } t \text{ de } t_0 \text{ à } t_f,$$

il suffit d'imposer la commande (boucle ouverte) :

$$u_d(t) = B(z_d(t), z_d^{(1)}(t), \dots, z_d^{(\beta)}(t)).$$

*Enseignement 1* : la trajectoire désirée doit nécessairement être au moins  $\beta$  fois dérivable.

## Les relations utiles

- la relation (4) est celle qui va permettre de construire l'algorithme de commande :

$$u(t) = B(z(t), z^{(1)}(t), \dots, v(t)),$$

où  $v(t)$  est une nouvelle commande qui conduit au système découplé :

$$z^{(\beta)}(t) = v(t);$$

- condition de découplage :

$$\frac{\partial B(\cdot)}{\partial z^{(\beta)}} \text{ localement inversible;}$$

- la relation (5) est celle qui va permettre de relier les comportements sur la sortie plate et la sortie du système.

## Remarque sur les dérivées de l'entrée

Supposons que l'on ait :

$$z = h(x(t), u(t), \dot{u}(t)).$$

Intégrateur sur la commande ( $v(t) = \dot{u}(t)$ ) et :

$$X(t) = \begin{bmatrix} x(t) \\ u(t) \end{bmatrix},$$

conduit à  $\Psi(\dot{X}(t), X(t), v(t)) = 0$  et  $z = h(X(t), v(t))$ .

Les autres relations s'écrivent :

$$X(t) = \begin{bmatrix} A(.) \\ B(.) \end{bmatrix} = \bar{A}(z(t), z^{(1)}(t), \dots, z^{(\max(\alpha, \beta))}(t)),$$

$$v(t) = \bar{B}(z(t), z^{(1)}(t), \dots, z^{(\beta+1)}(t)).$$

## Remarque (suite)

On aurait pu faire le même raisonnement en remplaçant l'intégrateur par un système du premier ordre :

$$\psi(u(t)) + \varphi(u(t))v(t) = \dot{u}(t).$$

Ainsi, modulo des filtres en entrée la sortie plate ne dépend pas vraiment des dérivées de l'entrée.

*Enseignement 2* : la mise en série est une opération interne pour l'ensemble des systèmes plats.

*Enseignement 3* : la dimension de l'état perd son importance.

**Exemple 2 : systèmes chaînés**

Modèle :

$$\begin{aligned}\dot{x}_1 &= u_1, \\ \dot{x}_2 &= u_2, \\ \dot{x}_3 &= x_2 u_1, \\ \dot{x}_4 &= x_3 u_1, \\ &\vdots \\ \dot{x}_n &= x_{n-1} u_1;\end{aligned}$$

L'élimination de  $u_1$  donne pour  $i = 3$  à  $n$ ,  $\dot{x}_i = x_{i-1} \dot{x}_1$ . La dernière :

$$\dot{x}_n = x_{n-1} \dot{x}_1,$$

indique que la connaissance de  $(x_1(t), x_n(t))$  permet d'obtenir  $x_{n-1}$ .

**Exemple 2 (suite)**

On remonte ainsi de proche en proche de  $x_{n-2}$  à  $x_2$  puis à :

$$u_1 = \dot{x}_1,$$

$$u_2 = f(\dot{x}_1, \dot{x}_n, \dots, x_1^{(n-1)}, x_n^{(n-1)}).$$

$$\implies z = \begin{bmatrix} x_1(t) & x_n(t) \end{bmatrix} \text{ est une sortie plate.}$$

*Enseignement 4* : l'élimination des commandes est une piste pour mettre en évidence des sorties plates.

### Exemple 3 : autres systèmes chaînés

Extension : Par le même raisonnement, le système plus général :

$$\dot{x}_1 = \Phi_1(u_1),$$

$$\dot{x}_2 = \Phi_2(u_1, x_2, \dots, x_n)u_2,$$

$$\dot{x}_3 = \Psi_3(u_1, x_3, \dots, x_n) + \Phi_3(u_1, x_3, \dots, x_n)x_2,$$

$$\dot{x}_4 = \Psi_3(u_1, x_4, \dots, x_n) + \Phi_3(u_1, x_4, \dots, x_n)x_3,$$

$$\vdots$$

$$\dot{x}_{n-1} = \Psi_{n-1}(u_1, x_{n-1}, x_n) + \Phi_{n-1}(u_1, x_{n-1}, x_n)x_{n-2},$$

$$\dot{x}_n = \Phi_n(u_1, x_n)x_{n-1},$$

où  $\Phi_1$  est une fonction inversible, admet également comme sortie plate  $z_1 = x_1$ ,  $z_2 = x_n$ .

**Remarque (5)**

- $\Phi_1$  est explicitement inversible, alors on peut **écrire explicitement** :

$$u_1 = \Phi_1^{-1}(\dot{x}_1),$$

et on a l'expression utilisable pour une commande par platitude;

- $\Phi_1$  n'est pas explicitement inversible (mais le théorème des fonctions implicites est vérifié), alors on peut **calculer numériquement** :

$$\text{pour chaque } \dot{x}_1, u_1 \text{ tel que } \dot{x}_1 = \Phi_1(u_1),$$

et on n'a pas d'expression utilisable pour une commande par platitude.

Comment faire dans ce cas?

## 2 formes de platitude

**Première forme** Celle où les formules de platitude, surtout  $u(t) = B(z(t), \dots, z^{(\beta)}(t))$ , sont explicites :

Le système est *explicitement* plat.

**Deuxième forme** Le système est plat mais on ne sait pas extrairement les formules donnant l'état et la commande :

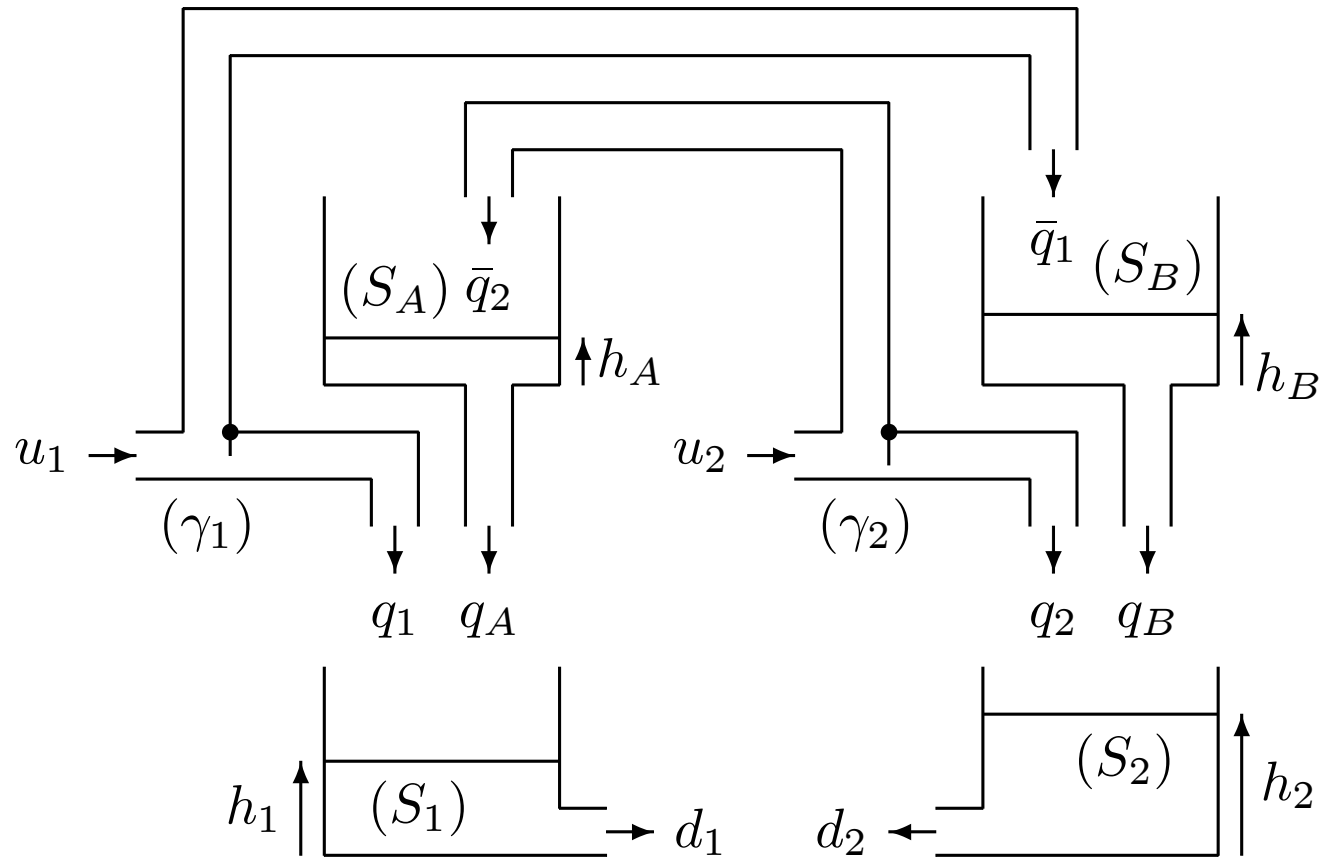
Le système est *numériquement* plat.

On ne peut calculer les commandes et les états correspondant à une trajectoire sur la sortie plate qu'à l'aide d'un algorithme numérique.

*Enseignement 5* : bien que cela ne change pas la propriété de platitude, on ne pourra pas appliquer les mêmes techniques de commande

## Exemple 4 : système numériquement plat

Soit le réseau hydrographique à quatre cuves :



## Equations des cuves

$$q_1(t) = \gamma_1 u_1(t),$$

$$\bar{q}_1(t) = (1 - \gamma_1)u_1(t),$$

$$q_2(t) = \gamma_2 u_2(t),$$

$$\bar{q}_2(t) = (1 - \gamma_2)u_2(t),$$

$$S_A \dot{h}_A(t) = \bar{q}_2(t) - q_A(t),$$

$$S_B \dot{h}_B(t) = \bar{q}_1(t) - q_B(t),$$

$$q_A(t) = k_A \sqrt{h_A(t)},$$

$$q_B(t) = k_B \sqrt{h_B(t)},$$

$$S_1 \dot{h}_1(t) = q_1(t) + q_A(t) - d_1(t), \quad S_2 \dot{h}_2(t) = q_2(t) + q_B(t) - d_2(t),$$

$$d_1(t) = k_1 \sqrt{h_1(t)},$$

$$d_2(t) = k_2 \sqrt{h_2(t)}.$$

$k_i$  : constantes relatives aux orifices d'écoulements;

commandes : taux d'ouvertures  $\gamma_1$  et  $\gamma_2$ , à valeurs dans  $[0,1]$ , des dérivations;

$u_1$  et  $u_2$  : débits (non contrôlés) alimentant le réseau hydrographique.

## Recherche de relations

L'élimination des commandes  $\gamma_1$  and  $\gamma_2$  donne :

$$S_A \dot{h}_A + S_2 \dot{h}_2 = -k_A \sqrt{h_A} - k_2 \sqrt{h_2} + k_B \sqrt{h_B} + u_2,$$

$$S_B \dot{h}_B + S_1 \dot{h}_1 = -k_B \sqrt{h_B} - k_1 \sqrt{h_1} + k_A \sqrt{h_A} + u_1.$$

Si on définit les volumes (quantités dérivées) :

$$V_{2A} = S_A h_A + S_2 h_2,$$

$$V_{1B} = S_B h_B + S_1 h_1,$$

on obtient :

$$\dot{V}_{2A} - u_2 = -k_A \sqrt{h_A} - k_2 \sqrt{h_2} + k_B \sqrt{h_B},$$

$$\dot{V}_{1B} - u_1 = -k_B \sqrt{h_B} - k_1 \sqrt{h_1} + k_A \sqrt{h_A}.$$

## Système algébrique

En groupant ces deux ensembles on obtient un système d'équations non linéaires en les inconnues :

$$H = \begin{bmatrix} h_A & h_B & h_1 & h_2 \end{bmatrix}^T.$$

Ce système est (localement) soluble si le théorème des fonctions implicites est vérifié, soit si :

$$J(H(t)) = \begin{bmatrix} S_A & 0 & 0 & S_2 \\ 0 & S_B & S_1 & 0 \\ -\frac{k_A}{2\sqrt{h_A}} & \frac{k_B}{2\sqrt{h_B}} & 0 & -\frac{k_2}{2\sqrt{h_2}} \\ \frac{k_A}{2\sqrt{h_A}} & -\frac{k_B}{2\sqrt{h_B}} & -\frac{k_1}{2\sqrt{h_1}} & 0 \end{bmatrix},$$

est inversible.

## Platitude du système des cuves

Ainsi, tant que :

$$\frac{S_1}{S_B} \frac{k_B}{k_1} \sqrt{\frac{h_1}{h_B}} + \frac{S_2}{S_A} \frac{k_A}{k_2} \sqrt{\frac{h_2}{h_A}} \neq 1,$$

$$z = \begin{bmatrix} V_{2A} \\ V_{1B} \end{bmatrix} \text{ est une sortie plate du système.}$$

On peut *numériquement* obtenir  $H = H(z, \dot{z}, u_1, u_2)$  et :

$$\gamma_1 = \frac{1}{u_1} \left( S_1 \dot{h}_1 + k_1 \sqrt{h_1} - k_A \sqrt{h_A} \right),$$

$$\gamma_2 = \frac{1}{u_2} \left( S_2 \dot{h}_2 + k_2 \sqrt{h_2} - k_B \sqrt{h_B} \right).$$

Système numériquement plat

Mais ceci n'est utile que pour la génération de trajectoire (BO).

## Exemple 5 : Dérivées des commandes

Système à deux commandes et  $a < b$  :

$$\begin{aligned}x_1^{(a)} &= u_1, \\x_2^{(b)} &= u_2, \\ \dot{x}_3 &= u_1 u_2.\end{aligned}$$

L'élimination des commandes donne l'intégrale première :

$$\dot{x}_3 = x_1^{(a)} x_2^{(b)}.$$

Une intégration par parties donne :

$$\left(x_3 - x_1^{(a)} x_2^{(b-1)}\right)^{(1)} = -x_1^{(a+1)} x_2^{(b-1)}.$$

### Exemple 5 : Recherche d'une sortie plate

Après  $b$  intégrations par parties successives :

$$\left[ \underbrace{x_3 - x_1^{(a)} x_2^{(b-1)} + x_1^{(a+1)} x_2^{(b-2)} + \dots + (-1)^b x_1^{(a+b-1)} x_2}_{\sigma} \right]^{(1)} = (-1)^b x_1^{(a+b)} x_2.$$

Conduit à choisir comme sortie plate  $(x_1, \sigma)$ .

Vérification :

$$x_2 = (-1)^b \frac{\sigma^{(1)}}{x_1^{(a+b)}}, x_3 = \sigma + x_1^{(a)} x_2^{(b-1)} - x_1^{(a+1)} x_2^{(b-2)} + \dots,$$

$$u_1 = x_1^{(a)}, u_2 = x_2^{(b)}.$$

## Remarques sur l'exemple 5

1. Comme  $x_1^{(a)} = u_1$ ,  $\sigma$  s'écrit en éliminant les dérivées de  $x_1$ , et en conservant les variables d'état  $x_3, x_2$  à  $x_2^{(b-1)}$  :

$$\sigma = x_3 - u_1 x_2^{(b-1)} + u_1^{(1)} x_2^{(b-2)} + \dots + (-1)^b u_1^{(b-1)} x_2.$$

2. *Enseignement 6* : après avoir éliminé les commandes, les variables candidates à être sorties plates sont celles qui sont dérivées dans les relations algébriques donnant les autres variables.

## Recherche des sorties plates

Pour un système :

$$\Phi(\dot{x}(t), x(t), u(t)) = 0,$$

un moyen de mettre en évidence les sorties plates consiste :

1. à construire le modèle implicite obtenu en éliminant les commandes :

$$\Xi(x^{(n)}(t), \dots, \dot{x}(t), x(t)) = 0;$$

2. à chercher à exprimer algébriquement certaines variables en fonctions des autres variables et de leur dérivées;
3. les variables dérivées sont candidates à être sorties plates.

Exemple : le convertisseur Continu-Continu

**Exemple 6 : Système implicite**

Soit le système régi par les intégrales premières :

$$\dot{x}_1 + \ddot{x}_3 \dot{x}_4 = 0, \quad \dot{x}_2 + (x_1 + \ddot{x}_3 x_4) x_4 = 0.$$

Soit  $\sigma = x_1 + \ddot{x}_3 x_4$ , alors on a :

$$\begin{aligned} \dot{\sigma} &= x_3^{(3)} x_4, \\ \dot{x}_2 + \sigma x_4 &= 0. \end{aligned}$$

*Exercice* : montrer que les couples  $(\sigma, x_3)$  ou  $(\sigma, x_4)$  ne peuvent être considérées comme sorties plates.

## Exemple 6 (suite)

Question : comment trouver une sortie plate?

Astuce : l'élimination de  $x_4$  donne :

$$\sigma \dot{\sigma} = -\dot{x}_2 x_3^{(3)} = \left( \frac{\sigma^2}{2} \right)^{(1)} .$$

Une intégration par parties (cf. exemple précédent) donne :

$$\left( \frac{\sigma^2}{2} + x_2 x_3^{(3)} \right)^{(1)} = x_2 x_3^{(4)}$$

Sorties plates candidates :  $z_1 = x_2 x_3^{(3)} + \frac{\sigma^2}{2}$  et  $z_2 = x_3$ .

**Exemple 6 (suite)**

Vérification :

$$x_2 = \frac{\dot{z}_1}{x_3^{(4)}}, \quad \sigma = \sqrt{2(x_4 - x_2 x_3^{(3)})},$$
$$x_4 = \frac{\dot{\sigma}}{x_3^{(3)}}, \quad x_1 = \sigma - x_3^{(2)} x_4.$$

Remarque : 4 variables, 2 relations

$\Rightarrow$   $\left\{ \begin{array}{l} 2 \text{ composantes pour la sortie plate;} \\ 2 \text{ variables libres;} \\ 2 \text{ commandes.} \end{array} \right.$

## De l'implicite à l'explicite

Observation : les commandes font apparaître les dérivées d'ordre le plus élevées des sorties plates.

Pour l'exemple 6, dans  $x_1$  et  $x_4$ , on trouve  $x_3^{(5)}$ , donc :

$$u_1 = x_1 \text{ et } u_2 = x_4$$

Cela ne suffit pas car  $\dot{x}_1$  et  $\dot{x}_4$  sont présentes dans les équations.

Astuce : on ajoute des intégrateurs :

$$u_1 = \dot{x}_1 \text{ et } u_2 = \dot{x}_4,$$

soit :

$$x_3^{(2)} = -\frac{u_1}{u_2} \text{ et } \dot{x}_2 = -x_1 x_4 + \frac{u_1}{u_2} x_3^2.$$

## De l'implicite à l'explicite (2)

En rajoutant l'intégrateur :

$$\dot{x}_3 = x_5,$$

$\implies$  Equation d'état :

$$\dot{x}_1 = u_1,$$

$$\dot{x}_2 = -x_1 x_4 + \frac{u_1}{u_2} x_3^2,$$

$$\dot{x}_3 = x_5,$$

$$\dot{x}_4 = u_2,$$

$$\dot{x}_5 = -\frac{u_1}{u_2}.$$

*Enseignement 7* : la mise en évidence de la platitude est un moyen de réaliser un modèle implicite.

## Conséquence de la platitude

Comme la commande en boucle ouverte n'est pas souhaitable, il y a lieu de prévoir un bouclage assurant la poursuite de trajectoire.

Deux possibilités :

- par commande non linéaire : par utilisation de la relation **explicite** de la commande (4);
- par linéarisation et commande linéaire : par linéarisation du modèle autour de la trajectoire désirée.

Remarques :

- la deuxième technique est utilisable **dans tous les cas**;
- on peut utiliser la platitude sur le modèle linéarisé.

## Commande non linéaire par platitude

1. Génération de trajectoire en boucle ouverte  $\implies z_d(t)$ .
2. Linéarisation et découplage : si à partir de (4) on fait :

$$u = B(z, \dots, z^{(\beta-1)}, v),$$

où  $v$  est la nouvelle commande, et si  $\det B_{z^{(\beta)}} \neq 0$ , on obtient le système linéaire découplé :

$$z^{(\beta)} = v.$$

Remarques :

- résultat à comparer à la linéarisation par bouclage des systèmes non linéaires;
- découplage sans dynamique des zéros;
- un bouclage supplémentaire de stabilisation est nécessaire.

## Bouclage stabilisant

La commande :

$$v = z_d^{(\beta)}(t) + \sum_{i=0}^{\beta-1} k_i (z_d^{(i)}(t) - z^{(i)}(t)),$$

où  $K(p) = p^\beta + \sum_{i=0}^{\beta-1} k_i p^i$  est un polynôme de Hurwitz, conduit à :

$$\begin{aligned} u &= \mathbf{B}(z, \dots, z^{(\beta-1)}, z_d^{(\beta)}(t) + \sum_{i=0}^{\beta-1} k_i (z_d^{(i)}(t) - z^{(i)}(t))), \\ &= \Phi(z, \dots, z^{(\beta-1)}, K(p)z_d(t)), \end{aligned}$$

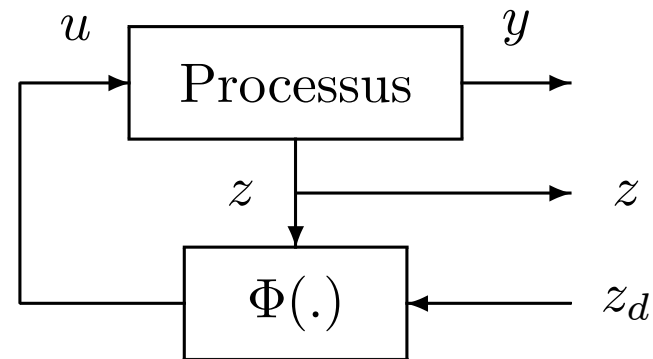
qui assure une poursuite de trajectoire asymptotique avec :

$$\lim_{t \rightarrow \infty} (z_d(t) - z(t)) = 0.$$

$u(t)$  s'appelle un bouclage ENDOGÈNE

## Structure I de commande

On met en œuvre la structure de commande :



Il est évident qu'un observateur de  $z$  est nécessaire

$\implies$  *problème délicat en non linéaire.*

## Commande linéaire par platitude

1. Génération de trajectoire en boucle ouverte :

$$z_d(t) \longmapsto (x_d(t), u_d(t), y_d(t)).$$

2. Linéarisation du modèle autour de cette trajectoire :

$$\begin{aligned}\delta\dot{x}(t) &= A(t)\delta x(t) + B(t)\delta u(t), \\ \delta y(t) &= C(t)\delta x(t) + D(t)\delta u(t),\end{aligned}$$

avec :

$$\delta x(t) = x_d(t) - x(t), \delta u(t) = u_d(t) - u(t), \delta y(t) = y_d(t) - y(t).$$

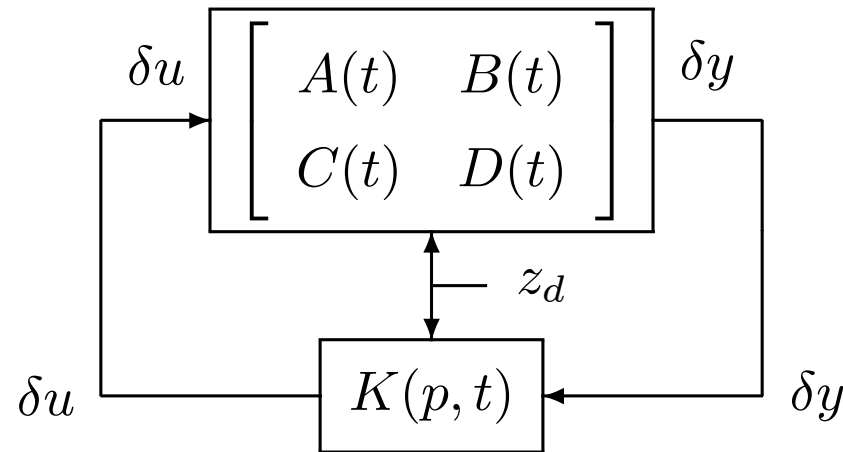
3. Elaboration de la commande :

$$u(t) = B(z_d(t), z_d^{(1)}(t), \dots, z_d^{(\beta)}(t)) - \delta u(t),$$

4. Structure bouclée :

$$\delta u(t) = K(p, t)\delta y(t),$$

tel que le système :

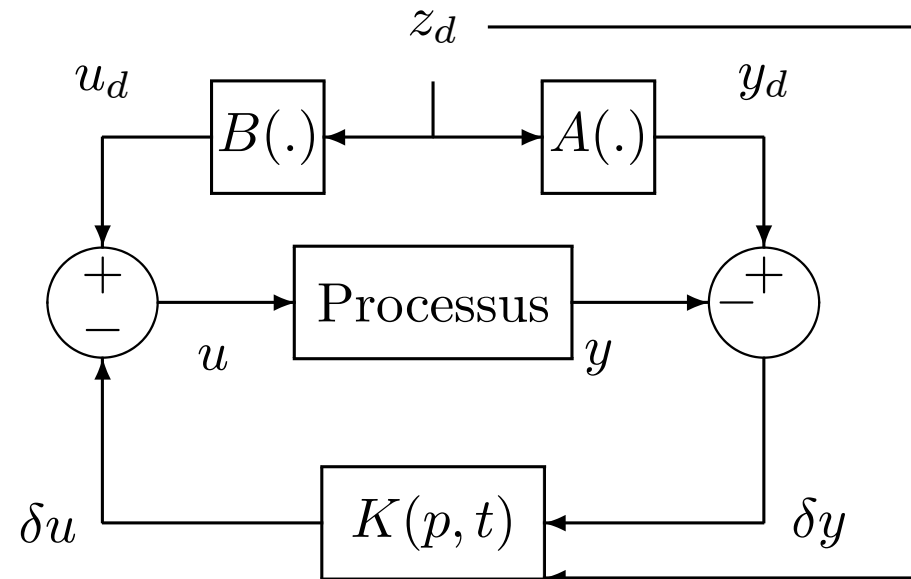


ait ses variables d'écart qui tendent vers 0.

Remarque : pour déterminer  $K(p, t)$  on peut bien sûr utiliser la platitude du système linéarisé!

## Structure II de commande

On met en œuvre la structure de commande :



Dans cette structure, le cadre linéaire simplifie la construction d'un observateur.

## Génération de trajectoire sur $z(t)$

Contrainte : elle doit être différentiable à l'ordre  $(\beta)$ .

Première solution : trajectoires polynomiales par morceaux.

- utilisation de polynômes d'interpolations;
- la plupart du temps, conditions de continuité au départ et à l'arrivée;
- on peut imposer des points de passage ou de rebroussement;
- on peut prévoir des trajectoires de contournement.

## Génération de trajectoire sur $z(t)$ (2)

Deuxième solution : optimisation en boucle ouverte.

Le critère :

$$J = \int_{t_0}^{t_1} \Psi(y, u) dt,$$

s'écrit :

$$J = \int_{t_0}^{t_1} L(z, \dots, z^{(\mu)}) dt.$$

La trajectoire optimale qui minimise ce critère est solution de l'équation d'Euler-Lagrange :

$$\frac{\partial L}{\partial z} - \frac{d}{dt} \frac{\partial L}{\partial z^{(1)}} + \frac{d^2}{dt^2} \frac{\partial L}{\partial z^{(2)}} + \dots + (-1)^\mu \frac{d^\mu}{dt^\mu} \frac{\partial L}{\partial z^{(\mu)}} = 0.$$

On peut bien sûr combiner les deux solutions (e.g. splines cubiques, courbes de Bézier, ... ).

## Génération de trajectoire sur $y(t)$

Souvent la sortie  $y(t)$  que l'on veut asservir n'est pas une sortie plate : on ne peut, sans risques, générer une trajectoire directement sur  $y(t)$ .

Comment procéder?

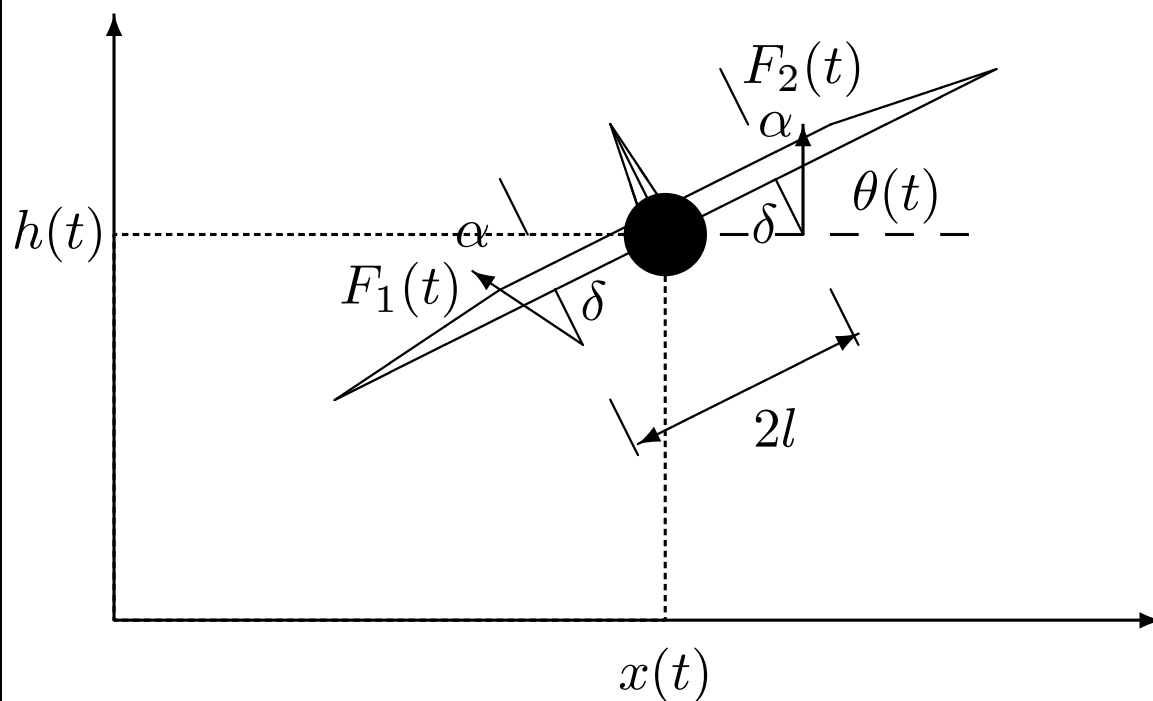
1. On impose des points de passage, pour  $i = 0$  à  $N$ ,  $y(t_i) = y_i$ .
2. On impose des contraintes aux points d'arrêts, pour  $i = 0$  et  $N$ , pour  $j = 1$  à  $M$ ,  $y^{(j)}(t_i) = y_{i,j}$ .
3. On impose des contraintes (de continuités) aux points de passage, pour  $i = 1$  à  $N - 1$ , pour  $j = 1$  à  $M$ ,  
 $y^{(j)}(t_i^-) = y^{(j)}(t_i^+)$ .
4. Compte tenu de  $y = \mathbf{C}(z, \dots, z^{(\gamma)})$ , pour  $j = 0$  à  $M$ ,  
 $y^{(j)} = \mathbf{C}_j(z, \dots, z^{(\gamma+j)})$ .

## Génération de trajectoire sur $y(t)$ (2)

1. Cela donne des contraintes, pour  $i = 0$  et  $N$ , pour  $j = 1$  à  $\gamma + M$ ,  $z^{(j)}(t_i) = z_{i,j}$ .
2. On génère la trajectoire  $z_d(t)$  de  $t_0$  à  $t_N$  qui vérifie ces contraintes.
3. Cela donne la trajectoire sur  $y(t)$ ,  $y_d(t) = \mathbf{C}(z_d(t), \dots, z_d^{(\gamma)}(t))$ .
4. Soit  $e(t) = y(t) - y_d(t)$  et  $E$  la tolérance maximale autorisée :
  - si  $\max e(t) < E$ , on exécute la commande  $u_d(t) = B(z_d(t), \dots, z_d^{(\beta)}(t))$ ;
  - si  $\max e(t) > E$ , on insère des points de passage supplémentaires et on reprend à l'étape 1.

### 3 Exemples pratiques (et instructifs!)

1. Introduction
2. Définitions et conséquences
3. Exemples pratiques
4. Un peu de formalisme
5. Critères de platitude
6. Cas des systèmes linéaires
7. SPD
8. Conclusion

**VTOL : Avion à décollage vertical**

## Modèle du VTOL

Coordonnées réduites :

$$x = \frac{x_G}{g}, \quad h = \frac{h_G}{g}, \quad \varepsilon = \frac{J \sin \alpha}{mg(l \cos \alpha + \delta \sin \alpha)},$$

$$u_1 = \frac{(F_1 + F_2) \cos \alpha}{mg}, \quad u_2 = \frac{(F_1 - F_2) \sin \alpha}{\varepsilon mg}.$$

Modèle dynamique (très) simplifié :

$$\begin{aligned} x^{(2)} &= -\sin \theta u_1 + \varepsilon \cos \theta u_2, \\ h^{(2)} &= \cos \theta u_1 + \varepsilon \sin \theta u_2 - 1, \\ \theta^{(2)} &= u_2. \end{aligned}$$

## Commande découplante du VTOL

Sortie :

$$y(t) = \begin{bmatrix} x(t) \\ h(t) \end{bmatrix}$$

1. Calcul des degrés relatifs :  $r_1 = 2$  et  $r_2 = 2$ .
2. Matrice de découplage :

$$\begin{bmatrix} x^{(2)}(t) \\ h^{(2)}(t) \end{bmatrix} = \begin{bmatrix} 0 \\ -1 \end{bmatrix} + \underbrace{\begin{bmatrix} -\sin \theta & \varepsilon \cos \theta \\ \cos \theta & \varepsilon \sin \theta \end{bmatrix}}_{B(\theta)} \begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix}.$$

$B(\theta)$  : matrice de découplage

3. Commande linéarisante découplante.

Comme  $\det B(\theta) = -\varepsilon$  et :

$$B(\theta)^{-1} = \begin{bmatrix} -\sin \theta & \cos \theta \\ \frac{\cos \theta}{\varepsilon} & \frac{\sin \theta}{\varepsilon} \end{bmatrix},$$

la commande non linéaire découplante est :

$$\begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix} = \begin{bmatrix} -\sin \theta & \cos \theta \\ \frac{\cos \theta}{\varepsilon} & \frac{\sin \theta}{\varepsilon} \end{bmatrix} \begin{bmatrix} v_1(t) \\ v_2(t) + 1 \end{bmatrix}.$$

4. Système bouclé :

$$\begin{bmatrix} x^{(2)}(t) \\ h^{(2)}(t) \end{bmatrix} = \begin{bmatrix} v_1(t) \\ v_2(t) \end{bmatrix}.$$

## Inconvénient de cette commande

Si on l'applique, l'avion se déplace bien mais part en looping.

Pourquoi?

Raison :

$$\varepsilon^{\theta^{(2)}}(t) = \cos \theta(t)v_1(t) + \sin \theta(t)(v_2(t) + 1).$$

Quand le système bouclé est stabilisé  $v_1(t)$  et  $v_2(t)$  tendent vers 0 et il reste :

$$\varepsilon^{\theta^{(2)}}(t) = \sin \theta(t),$$

qui est un système instable!

Problème des zéros instables

## Platitude du VTOL

D'après les calculs précédents :

$$\begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix} = \begin{bmatrix} -\sin \theta & \cos \theta \\ \frac{\cos \theta}{\varepsilon} & \frac{\sin \theta}{\varepsilon} \end{bmatrix} \begin{bmatrix} x^{(2)}(t) \\ h^{(2)}(t) + 1 \end{bmatrix},$$

conduit à :

$$\varepsilon \theta^{(2)}(t) = \cos \theta(t) x^{(2)}(t) + \sin \theta(t) (h^{(2)}(t) + 1).$$

Astuce : on ajoute  $\cos^2 \theta + \sin^2 \theta = 1$ , soit :

$$\cos \theta \left( x^{(2)} - \varepsilon \cos \theta \theta^{(2)} \right) = -\sin \theta (h^{(2)} + 1 - \varepsilon \sin \theta \theta^{(2)}).$$

Or :

$$\begin{aligned} x^{(2)} - \varepsilon \cos \theta \theta^{(2)} &= (x - \varepsilon \sin \theta)^{(2)} - \varepsilon \sin \theta \dot{\theta}^2, \\ h^{(2)} - \varepsilon \sin \theta \theta^{(2)} &= (h + \varepsilon \cos \theta)^{(2)} + \varepsilon \cos \theta \dot{\theta}^2. \end{aligned}$$

## Relations de platitude du VTOL

Comme :

$$\cos \theta \underbrace{(x - \varepsilon \sin \theta)}_X^{(2)} = - \sin \theta \underbrace{((h + \varepsilon \cos \theta))}_H^{(2)} + 1).$$

Sorties plates :  $X = x - \varepsilon \sin \theta$  et  $H = h + \varepsilon \cos \theta$ .

- $\alpha \cos \theta = \beta \sin \theta \implies \theta = (2k + 1)\pi/2$  si  $\beta = 0$  et  $\tan \theta = \alpha/\beta$

sinon :

$$\theta = - \arctan \frac{X^{(2)}}{H^{(2)} + 1};$$

- $\sin \theta = - \frac{X^{(2)}}{\sqrt{X^{(2)2} + (H^{(2)} + 1)^2}}$  et  $\cos \theta = \frac{H^{(2)}}{\sqrt{X^{(2)2} + (H^{(2)} + 1)^2}}$  :

$$x = x(X, X^{(2)}, H^{(2)}), h = h(H, X^{(2)}, H^{(2)});$$

## Relations de platitude du VTOL

- $u_2 = u_2(X^{(2)}, H^{(2)}, X^{(3)}, H^{(3)}, X^{(4)}, H^{(4)})$ ;
- $X^{(2)} = -\sin \theta(u_1 - \varepsilon \dot{\theta}^2)$  et  $H^{(2)} + 1 = \cos \theta(u_1 - \varepsilon \dot{\theta}^2)$ , donc :

$$u_1 = \sqrt{X^{(2)2} + (H^{(2)} + 1)^2} + \varepsilon \dot{\theta}^2 = u_1(X^{(2)}, H^{(2)}, X^{(3)}, H^{(3)}).$$

*Enseignement* : la sortie plate a souvent une interprétation physique

Ici : centre instantané de rotation

## Commande par platitude du VTOL

Les relations de platitude indiquent  $\beta = 4$ , donc il existe un bouclage linéarisant tel que :

$$X^{(4)} = v_1 \text{ et } H^{(4)} = v_2,$$

soit une dimension 8 du système en boucle fermée.

Remarque : comparer avec les dimensions :

- 6 du système;
- 4 du système bouclé par le retour non linéaire précédent.

$\implies$  bouclage dynamique de dimension 2

## Problème

Comment construire le retour d'état nécessaire?

Dans :

$$\begin{aligned}u_1 &= u_1(X^{(2)}, H^{(2)}, X^{(3)}, H^{(3)}) \\u_2 &= u_2(X^{(2)}, H^{(2)}, X^{(3)}, H^{(3)}, v_1, v_2),\end{aligned}$$

- $u_1$  ne contient pas les nouvelles commandes;
- on doit avoir  $X^{(3)}$  et  $H^{(3)}$ .

## Construction du bouclage

Comme on cherche un bouclage découplant on déroule l'algorithme de commande non linéaire mais relativement aux sorties plates.

1. Avec  $\lambda = u_1 - \varepsilon\dot{\theta}^2$ , soit  $u_1 = \lambda + \varepsilon\dot{\theta}^2$  :

$$X^{(2)} = -\sin\theta\lambda, \quad H^{(2)} + 1 = \cos\theta\lambda.$$

2. Après 2 dérivations, on doit avoir :

$$X^{(4)} = \sin\theta\dot{\theta}^2\lambda - 2\cos\theta\dot{\theta}\dot{\lambda} - \cos\theta\lambda u_2 - \sin\theta\ddot{\lambda} = v_1,$$

$$H^{(4)} = -\cos\theta\dot{\theta}^2\lambda - 2\sin\theta\dot{\theta}\dot{\lambda} - \sin\theta\lambda u_2 + \cos\theta\ddot{\lambda} = v_2.$$

soit :

$$\begin{bmatrix} -\cos\theta\lambda & -\sin\theta \\ -\sin\theta\lambda & \cos\theta \end{bmatrix} \begin{bmatrix} u_2 \\ \ddot{\lambda} \end{bmatrix} = \begin{bmatrix} v_1 - \sin\theta\dot{\theta}^2\lambda + 2\cos\theta\dot{\theta}\dot{\lambda} \\ v_2 + \cos\theta\dot{\theta}^2\lambda + 2\sin\theta\dot{\theta}\dot{\lambda} \end{bmatrix}.$$

## Bouclage

La résolution de ce système donne  $u_2$  et  $\ddot{\lambda}$  et le nombre d'états est complet.

1. Tant que  $\lambda \neq 0$  :

$$\begin{aligned}\ddot{\lambda} &= \lambda \dot{\theta}^2 - v_1 \sin \theta + v_2 \cos \theta, \\ u_1 &= \lambda + \varepsilon \dot{\theta}^2, \\ u_2 &= -\frac{1}{\lambda} (v_1 \cos \theta + v_2 \sin \theta + 2\dot{\theta}\dot{\lambda}).\end{aligned}$$

2. Bouclage stabilisant :

$$v_1 = X_d^{(4)} + \sum_{i=0}^3 k_{1,i} (X_d^{(i)} - X^{(i)}), \quad v_2 = H_d^{(4)} + \sum_{i=0}^3 k_{1,i} (H_d^{(i)} - H^{(i)}),$$

qui utilise les relations :

$$X = x - \varepsilon \sin \theta,$$

$$H = h + \varepsilon \cos \theta,$$

$$X^{(1)} = \dot{x} - \varepsilon \cos \theta \dot{\theta},$$

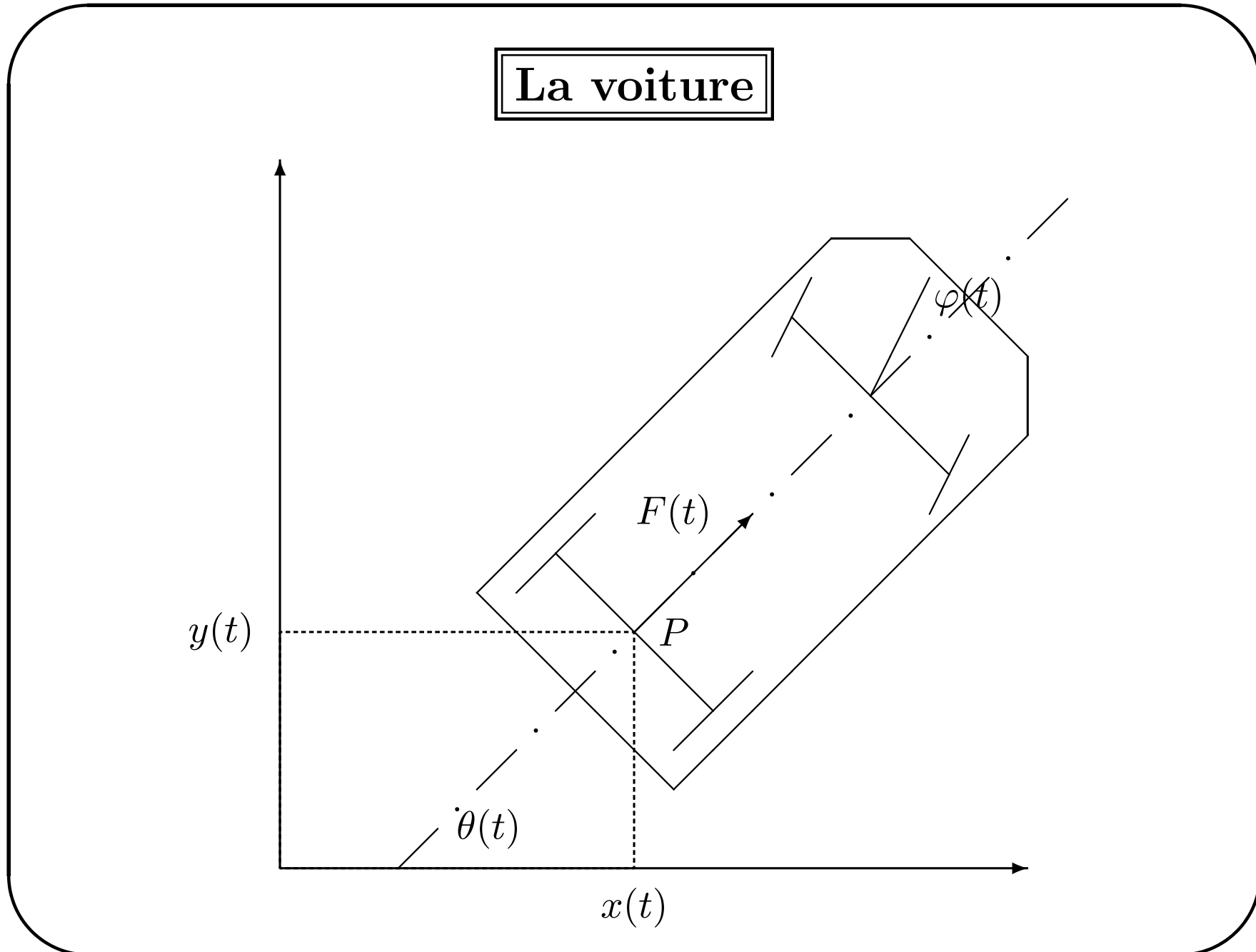
$$H^{(1)} = \dot{h} - \varepsilon \sin \theta \dot{\theta},$$

$$X^{(2)} = -\sin \theta \lambda,$$

$$H^{(2)} = \cos \theta \lambda - 1,$$

$$X^{(3)} = -\cos \theta \dot{\theta} \lambda + \sin \theta \dot{\lambda},$$

$$H^{(3)} = -\sin \theta \dot{\theta} \lambda + \cos \theta \dot{\lambda}.$$



## Modèle de la voiture

- Commandes :
  - $u$  : vitesse de P;
  - $\varphi$  : angle de braquage des roues avant;
- Modèle (roulement sans glissement) :

$$\dot{x} = u \cos \theta,$$

$$\dot{y} = u \sin \theta,$$

$$\dot{\theta} = \frac{u}{l} \tan \varphi;$$

- les coordonnées de P( $x(t), y(t)$ ) sont une sortie plate du système

## Preuve de la platitude

A partir d'une trajectoire de  $P(x(t), y(t))$  :

- la direction de la tangente à la courbe donne  $\theta$ ;
- son module donne  $u$ ;
- le cercle osculateur (celui de la rotation instantanée de  $P$ ) donne  $\varphi$

## Relations de platitude

Sortie plate  $\left[ \begin{array}{cc} x(t) & y(t) \end{array} \right] \implies$

$$\left\{ \begin{array}{l} u(t) = \sqrt{\dot{x}^2(t) + \dot{y}^2(t)}, \\ \theta(t) = \arctan\left(\frac{\dot{y}(t)}{\dot{x}(t)}\right), \\ \varphi(t) = \arctan\left(\frac{l(\dot{x}(t)\ddot{y}(t) - \ddot{x}(t)\dot{y}(t))}{\sqrt{\dot{x}^2(t) + \dot{y}^2(t)}^3}\right), \end{array} \right.$$

Interprétation de la singularité à vitesse nulle :

- on ne s'arrête ni ne démarre brutalement;
- si on tourne les roues à  $u = 0$ , il ne se passe rien.

## Traitement de la singularité

L'idée consiste à découpler :

- l'aspect géométrique de la trajectoire;
- l'évolution sur cette trajectoire.

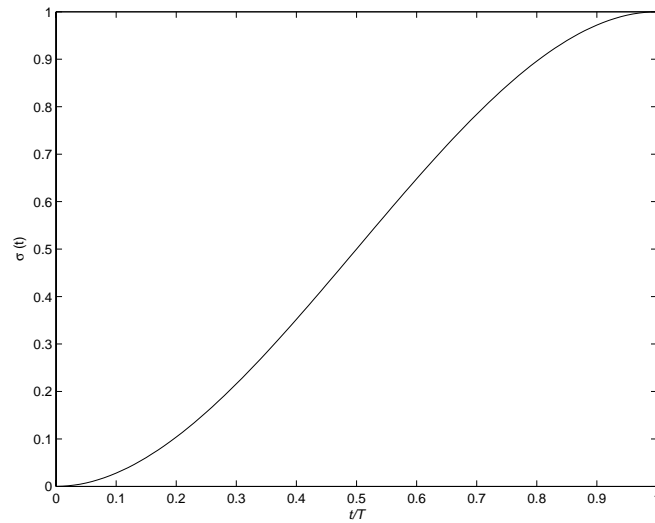
Solution :

paramétrisation de l'abscisse curviligne  $\sigma(t)$  telle que, si  $T$  est la durée du mouvement :

$$\sigma(0) = 0, \sigma(T) = 1, \dot{\sigma}(0) = \dot{\sigma}(T) = 0.$$

soit :

$$\sigma(t) = \left(\frac{t}{T}\right)^2 \left(3 - 2\frac{t}{T}\right).$$



## Traitement de la singularité

géométrie de la trajectoire :  $\begin{bmatrix} x_d(\sigma) & y_d(\sigma) \end{bmatrix}$ .

Comme  $x'^2(\sigma) + y'^2(\sigma) = 1$  :

$$\Rightarrow \begin{cases} u(t) = \dot{\sigma}(t), \\ \theta(t) = \arctan\left(\frac{y'}{x'}\right), \\ \varphi(t) = \arctan(l(x'(t)y''(t) - x''(t)y'(t))), \end{cases}$$

## Mise en œuvre de la commande par platitude

La construction suit la même démarche que pour l'avion :

- $u$  et  $\varphi$  dépendent de  $x^{(2)}$  et  $y^{(2)}$  donc il existe un bouclage tel que :

$$x^{(2)} = v_1 \text{ et } y^{(2)} = v_2;$$

- pour le construire, on dérive 2 fois les sorties plates :

$$x^{(2)} = \dot{u} \cos \theta - u \sin \theta \dot{\theta} = \dot{u} \cos \theta - \frac{u^2}{l} \sin \theta \tan \varphi = v_1,$$

$$y^{(2)} = \dot{u} \sin \theta + u \cos \theta \dot{\theta} = \dot{u} \sin \theta + \frac{u^2}{l} \cos \theta \tan \varphi = v_2,$$

soit :

$$\begin{bmatrix} \cos \theta & -\frac{u^2}{l} \sin \theta \\ \sin \theta & \frac{u^2}{l} \cos \theta \end{bmatrix} \begin{bmatrix} \dot{u} \\ \tan \varphi \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}.$$

## Bouclage dynamique

$$\dot{u} = \cos \theta v_1 + \sin \theta v_2,$$

$$\varphi = \arctan \left( \frac{l}{u^2} (\cos \theta v_2 - \sin \theta v_1) \right);$$

auquel on ajoute le retour stabilisant de poursuite de trajectoire :

$$v_1 = x_d^{(2)} + k_{1,1}(x_d^{(1)} - x^{(1)}) + k_{1,0}(x_d - x),$$

$$v_2 = y_d^{(2)} + k_{1,1}(y_d^{(1)} - y^{(1)}) + k_{1,0}(y_d - y),$$

soit :

$$v_1 = x_d^{(2)} + k_{1,1}(x_d^{(1)} - \cos \theta u) + k_{1,0}(x_d - x),$$

$$v_2 = y_d^{(2)} + k_{1,1}(y_d^{(1)} - \sin \theta u) + k_{1,0}(y_d - y);$$

Mais  $u$  n'est plus la commande.

## Traitement de la singularité

On réalise :

$$u(t) = v(t)\dot{\sigma}(t),$$

où  $v(t)$  est la nouvelle commande qui est égale à 1 sur la trajectoire désirée. Comme :

$$\frac{d}{dt} = \dot{\sigma}(t) \frac{d}{d\sigma},$$

Avec, suivant le choix de  $\sigma(t)$  :

$$\dot{\sigma}(t) = \frac{6t}{T^3}(T - t),$$

on obtient finalement comme commande par platitude qui assure la poursuite asymptotique de la trajectoire  $(x_d(t), y_d(t))$  :

## Commande avec singularité

$$u(t) = v(t)\dot{\sigma}(t),$$

$$\dot{v}(t) = \dot{\sigma}(t)(\cos \theta v_1 + \sin \theta v_2),$$

$$\varphi(t) = \arctan \left( \frac{l}{v(t)^2} (\cos \theta v_2 - \sin \theta v_1) \right),$$

$$v_1 = x_d^{(2)} + k_{1,1}(x_d^{(1)} - \cos \theta u) + k_{1,0}(x_d - x),$$

$$v_2 = y_d^{(2)} + k_{1,1}(y_d^{(1)} - \sin \theta u) + k_{1,0}(y_d - y).$$

*Enseignement* : les singularités qui apparaissent dans  $B(\cdot)$  sont résolues par un découplage :

- de l'aspect géométrique avec la génération de trajectoire  $z_d(\sigma)$ ;
- de l'évolution sur cette trajectoire par paramétrisation de l'abscisse curviligne  $\sigma(t)$ .

**Exemple simple**

Soit le système :

$$\ddot{x}(t) + u(t)x(t) = u^5(t).$$

où  $u(t)$  est la commande.

Théorème des fonctions implicites :

tant que  $5u^4(t) \neq x(t)$  :  $x(t)$  est une sortie plate.

Système numériquement plat

Remarque : la commande  $u(t)$  ne doit pas vérifier

$$15\dot{u}^2 + 5\ddot{u}u + u^3 = 0$$

## Conséquence

Commande linéaire non stationnaire autour de la trajectoire désirée (structure II de commande).

Avec  $\delta x = x_d - x$ ,  $\delta u = u_d - u$ ,  $\alpha(t) = 5u_d^4(t) - x_d(t)$  et

$$X = \begin{bmatrix} \delta x & \delta \ddot{x} \end{bmatrix} :$$

$$\dot{X} = \begin{bmatrix} 0 & 1 \\ -u_d(t) & 0 \end{bmatrix} X + \begin{bmatrix} 0 \\ \alpha(t) \end{bmatrix} \delta u.$$

Matrice de commandabilité :

$$C_{(A(t),B(t))} = \begin{bmatrix} 0 & \alpha(t) \\ \alpha(t) & -\dot{\alpha}(t) \end{bmatrix}.$$

Le système est commandable tant que  $0 < \alpha(t) < \infty$ .

## Platitude du linéarisé

En linéaire platitude et commandabilité coïncident.

Lorsque cette condition est vérifiée :

$$\delta u(t) = \frac{1}{\alpha(t)} [\delta \ddot{x}(t) + u_d(t) \delta x(t)],$$

qui montre le caractère plat de la sortie  $\delta x(t)$ .

A comparer avec la condition de platitude numérique.

Si la condition de commandabilité n'est pas vérifiée on reparamètre l'*abscisse curviligne*.

## Platitude du linéarisé

Commande par platitude du linéarisé :

$$\delta u(t) = \frac{1}{\alpha(t)} [v(t) + u_d(t)\delta x(t)],$$

avec :

$$v(t) = \delta \ddot{x}_d(t) + k_1(\delta \dot{x}_d(t) - \delta \dot{x}(t)) + k_0(\delta x_d(t) - \delta x(t)).$$

Mais, comme on veut pour la structure complète  $\delta x_d(t) \equiv 0$ , on obtient directement :

$$\delta u(t) = \frac{1}{\alpha(t)} [-k_1\delta \dot{x}(t) + (u_d(t) - k_0)\delta x(t)] = u_d(t) - u(t).$$

## Commande complète

$$\begin{aligned}
 u(t) &= u_d(t) - \frac{1}{\alpha(t)} [k_1 \delta \dot{x}(t) + (k_0 - u_d(t)) \delta x(t)], \\
 &= u_d(t) - \frac{[k_1 (\dot{x}_d(t) - \dot{x}(t)) + (k_0 - u_d(t))(x_d(t) - x(t))]}{\alpha(t)},
 \end{aligned}$$

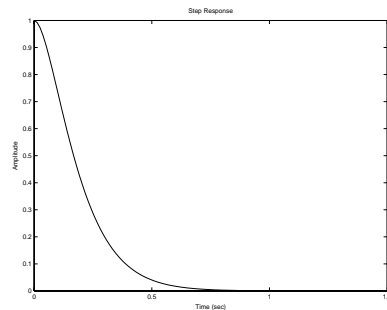
ce qui donne la commande finale :

$$\begin{aligned}
 u(t) &= \underbrace{u_d(t) - \frac{1}{\alpha(t)} [k_1 \dot{x}_d(t) + (k_0 - u_d(t)) x_d(t)]}_{\varphi_d(t)} \left. \vphantom{u_d(t)} \right\} \text{ consigne} \\
 &\quad - \underbrace{\frac{1}{\alpha(t)} [(u_d(t) - k_0)x(t) - k_1 \dot{x}(t)]}_{\text{bouclage d'état}}.
 \end{aligned}$$

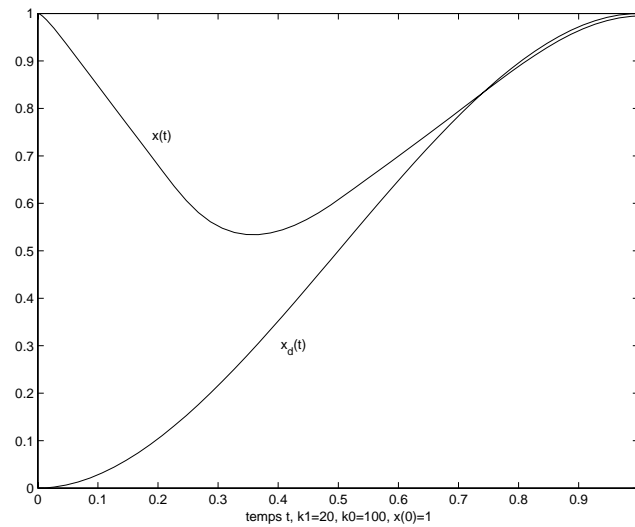
## Simulation

Trajectoire à suivre :  $x_d(t) = t^2(3 - 2t)$  pour  $t$  de 0 à 1;

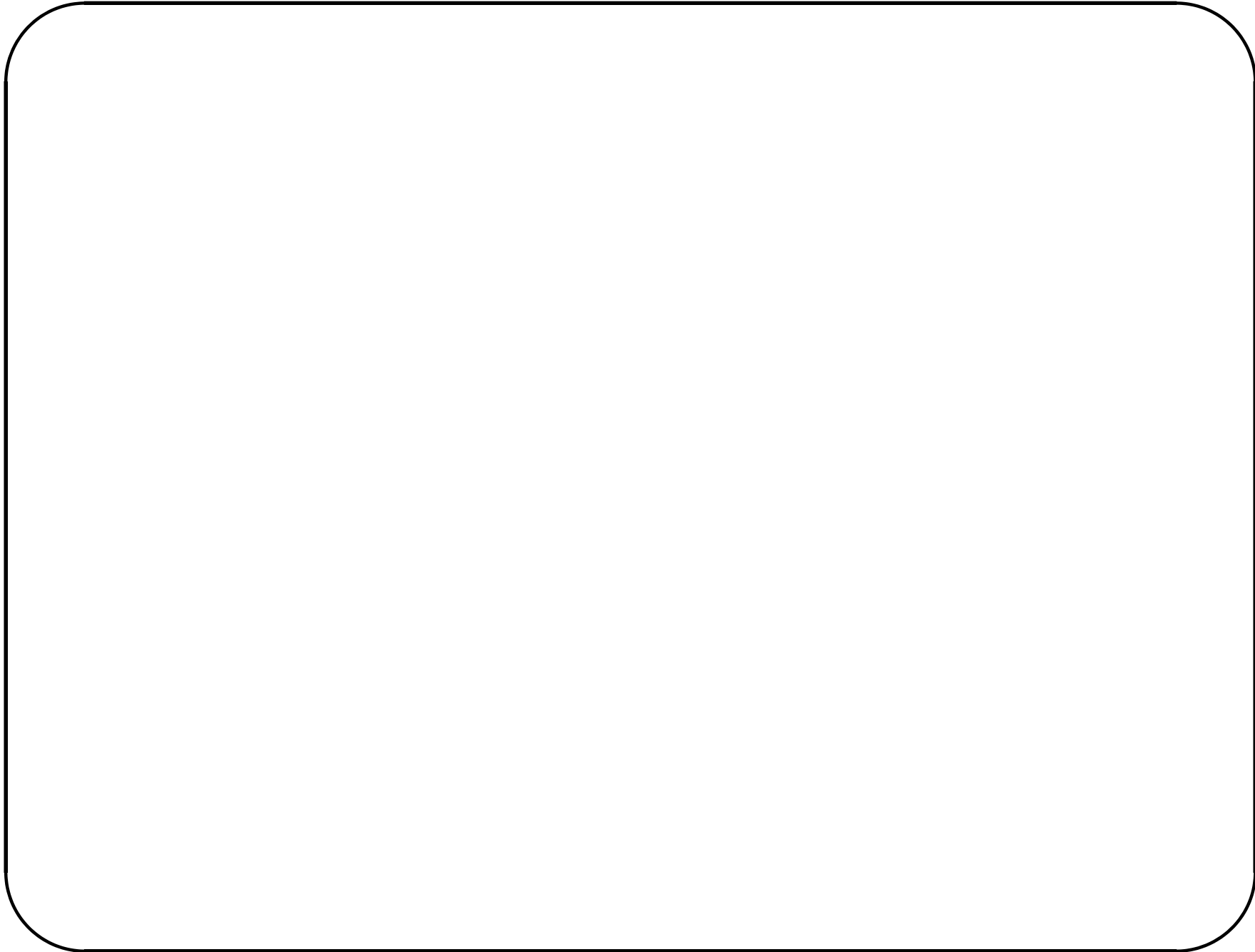
Dynamique de poursuite :  $k_1 = 20$ ,  $k_0 = 100$  :



Temps de réponse à 2% : 0.6 seconde.



Commande par platitude avec estimation grossière de l'état initial.



## 4 Formalisme

*(a little touch ... )*

1. Introduction
2. Définitions et conséquences
3. Exemples pratiques
4. Un peu de formalisme
5. Critères de platitude
6. Cas des systèmes linéaires
7. SPD
8. Conclusion

## Formalismes

Plusieurs formalismes permettent de construire la notion de platitude :

- géométrie différentielle de dimension finie et bouclage linéarisant : Charlet, Lévine, 1989; Shadwick, 1990; Sluis, 1993; Franch, 1999;
- algèbre différentielle : Jacubczyk, 1993, Fliess, Lévine, Martin, Rouchon, 1995; Aranda-Bricaire, Moog, Pomet, 1995;
- géométrie différentielle de dimension infinie des jets : Pomet, 1993; van Nieuwdadt, Rathinam, Murray, 1998; Fliess, Lévine, Martin, Rouchon, 1999; Pereira da Silva, Correa Filho, 2001,

Peut être reliée à des travaux plus anciens : Hilbert, 1901, 1912, Poincaré, 1907, Cartan, 1914.

## Systemes

1. Système différentiel (libre) :

$$\Sigma = (X, f)$$

$$\dot{x} = f(x), \forall t \in \mathbb{R}, x \in X \subseteq \mathbb{R}^n;$$

$\implies$  **une** trajectoire  $t \longmapsto x(t)$

2. Système commandé (forcé) :

$$\Sigma_u = (X, f_u)$$

$$\dot{x} = f(x, u), \forall t \in \mathbb{R}, (x, u) \in X \times U \subseteq \mathbb{R}^n \times \mathbb{R}^m;$$

$\implies$  **une infinité** de trajectoires  $t \longmapsto x_u(t)$

## Unification

Notion de trajectoire :

$$t \longmapsto \xi = (x(t)^T, u(t)^T, \dot{u}(t)^T, \dots, u^{(k)}(t)^T, \dots)^T,$$

$$\forall t \in \mathbb{R}, \xi \in \mathcal{M} \subseteq \mathbb{R}^n \times \underbrace{\mathbb{R}^m \times \dots \times \mathbb{R}^m}_{\infty\text{-té de copies de } \mathbb{R}^m} = \mathbb{R}^n \times (\mathbb{R}^m)^\infty$$

cela permet d'avoir :

$$\dot{\xi} = (f(x, u)^T, \dot{u}^T, \dots, u^{(k)T}, \dots)^T = F(\xi).$$

**Système commandé :**

$$\Sigma = (\mathcal{M}, F).$$

## Remarques

- ce formalisme est celui des jets-infinis;
- même formalisme pour les systèmes implicites  $\phi(\xi, \dot{\xi}) = 0$ ;
- on perd la notion de dimension de l'état : soit  $\Sigma^*$  défini par :

$$\begin{aligned}\dot{x} &= f(x, u), \\ \dot{u} &= v,\end{aligned}$$

que l'on peut noter  $\dot{X} = \phi(X, v)$  mais qui correspond à la représentation :

$$(\dot{X}^T, \dot{v}^T, \dots, v^{(k-1)T}, \dots) = (f(x, u)^T, \dot{u}^T, \ddot{u}^T, \dots, u^{(k)T}, \dots).$$

Les deux systèmes ont une dimension différente mais la même représentation;

- commandes : variables indépendantes (ou libres) du système.

## Système trivial

Définition (évidente) du système trivial :  $y(t) = u(t)$ .

Dans le formalisme des jets :  $T = ((\mathbb{R}^m)^\infty, F)$  où :

$$F(y_0^T, y_1^T, y_2^T, \dots) = (y_1^T, y_2^T, \dots).$$

Conséquence : *toute suite d'intégrateurs est un système trivial.*

$$y^{(n)} = u,$$

Preuve : soit  $\xi = (y^T, \dots, y^{(n-1)T}, u^T, \dot{u}^T, \dots, u^{(k)T}, \dots)^T$ , alors :

$$\dot{\xi} = (\dot{y}^T, \dots, y^{(n-1)T}, \underbrace{y^{(n)T}}_{u^T}, \dot{u}^T, \dots, u^{(k)T}, \dots)^T,$$

soit  $y_0 = y$ .

## Equivalence de systèmes

Deux systèmes  $(\mathcal{M}, F)$  et  $(\mathcal{N}, G)$  sont équivalents si il existe une transformation inversible qui échange leurs trajectoires.

Notons  $\Psi$  et  $\Phi$  une telle transformation et son inverse :

- si  $\xi$  est une trajectoire de  $F(\xi)$  alors  $\zeta = \Psi(\xi)$  est une trajectoire de  $G(\zeta) = \left[ \frac{\partial \Psi}{\partial \xi} \right]_{\xi=\Phi(\zeta)} F(\Phi(\zeta))$ ;
- si  $\zeta$  est une trajectoire de  $G(\zeta)$  alors  $\xi = \Phi(\zeta)$  est une trajectoire de  $F(\xi) = \left[ \frac{\partial \Phi}{\partial \zeta} \right]_{\zeta=\Psi(\xi)} G(\Psi(\xi))$ .

Une telle transformation s'appelle une transformation de *Lie-Backlund*.

## Implication de l'équivalence

Soit deux systèmes LB-équivalents :

$$\dot{x} = f(x, u) \text{ et } \dot{y} = g(y, v).$$

En notant  $\bar{u} = (u, \dot{u}, \ddot{u}, \dots)$  et  $\bar{v} = (v, \dot{v}, \ddot{v}, \dots)$ , on les relations :

$$\underbrace{\Psi(x, u, \dot{u}, \ddot{u}, \dots)}_{\xi} = (\underbrace{\alpha(x, \bar{u})}_y, \underbrace{\beta(x, \bar{u})}_v, \underbrace{\dot{\beta}(x, \bar{u})}_{\dot{v}}, \dots),$$

$$\underbrace{\Phi(y, v, \dot{v}, \ddot{v}, \dots)}_{\zeta} = (\underbrace{\gamma(y, \bar{v})}_x, \underbrace{\delta(y, \bar{v})}_u, \underbrace{\dot{\delta}(y, \bar{v})}_{\dot{u}}, \dots),$$

ce qui donne nécessairement les relations entre variables :

$$\begin{cases} y = \alpha(x, \bar{u}), \\ v = \beta(x, \bar{u}), \end{cases} \quad \text{et} \quad \begin{cases} x = \gamma(y, \bar{v}), \\ u = \delta(y, \bar{v}). \end{cases}$$

## Remarques

Une transformation de Lie-Backlund :

- ne conserve pas le nombre d'états;
- peut être étendue aux systèmes à temps variable et aux changements de temps;
- *peut ne pas conserver le temps* : l'équivalence est alors orbitale :

$$\begin{array}{ll}
 y = \alpha(t, x, \bar{u}) & x = \gamma(\tau, y, \bar{v}) \\
 v = \beta(t, x, \bar{u}) & \iff u = \delta(\tau, y, \bar{v}) \\
 \tau = \lambda(t, x, \bar{u}) & t = \mu(\tau, y, \bar{v})
 \end{array}$$

- comme on est en dimension finie,  $\exists k$  “assez grand” :

$$y = \alpha(t, x, \bar{u}^k), \dots, t = \mu(\tau, y, \bar{v}^k),$$

$$\text{où } \bar{u}^k = (u^T, \dot{u}^T, \dots, u^{(k)T})^T.$$

## Implication de l'équivalence

*Théorème 1 :*

Si deux systèmes sont L-B équivalents alors ils ont le même nombre de commandes.

*Théorème 2 :*

Si deux systèmes sont L-B équivalents alors il existe un bouclage dynamique endogène et un changement de coordonnées qui transforme l'un en l'autre.

## Systeme plat

Rappel : un système est plat si on a les relations :

$$z = h(x, \bar{u}), \quad x = A(\bar{z}), \quad u = B(\bar{z}),$$

Donc s'il est équivalent à un système tel que  $y$  soit dans les première composantes de  $z$ , soit le système trivial.

En résumé :

Un système  $(\mathcal{M}, F)$  est plat s'il est équivalent au système trivial.

Conséquence : le nombre de sorties plates est égal au nombre de commandes.

## Exemple de système orbitalement plat

$$\begin{aligned}\dot{x}_1 &= u, \\ \dot{x}_2 &= u^2.\end{aligned}$$

Ce système n'est pas plat, par contre si on fait la transformation :

$$\begin{aligned}\tau &= x_1, \\ y &= x_2,\end{aligned}$$

on obtient le système plat :

$$\frac{dy}{d\tau} = u(\tau).$$

## Sortie plate = sortie linéarisante

A partir de :

$$x = A(z, z^{(1)}, \dots, z^{(\alpha)}) = A(\bar{z}^\alpha).$$

Moyennant une permutation entre les variables,  $\bar{z}^\alpha$  peut être partitionné en  $[\xi, \zeta]$  où  $\xi \in \mathbb{R}^n$ , et :

$$\text{rang} (\partial A / \partial \xi) = n.$$

Par le théorème des fonctions implicites :

$$\exists F(\cdot), \quad \xi = F(x, \zeta).$$

Ainsi :

$$\begin{aligned} \dot{\zeta} &= H(\xi, \zeta, z^{(\beta+1)}) = a(x, \zeta, z^{(\beta+1)}), \\ u &= B(\xi, \zeta, z^{(\beta+1)}) = b(x, \zeta, z^{(\beta+1)}). \end{aligned}$$

## Sortie plate = sortie linéarisante (2)

Comme le bouclage par platitude consiste à poser :

$$z^{(\beta+1)} = v,$$

le bouclage :

$$\begin{aligned}\dot{\zeta} &= a(x, \zeta, v), \\ u &= b(x, \zeta, v),\end{aligned}$$

conduit nécessairement à la forme de Brunovski :

- système linéaire;
- sous forme canonique commandable;
- dont tout les coefficients significatifs sont nuls.

## 5 Critères de platitude

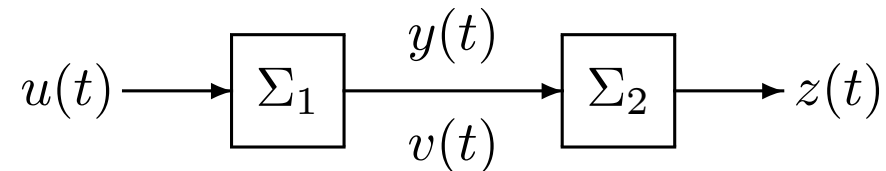
1. Introduction
2. Définitions et conséquences
3. Exemples pratiques
4. Un peu de formalisme
5. Critères de platitude
6. Cas des systèmes linéaires
7. SPD
8. Conclusion

## Caractérisation de la platitude

- il existe des conditions nécessaires et suffisantes de platitude :  
mais, caractérisation systématique (encore) difficile;
- certains résultats partiels :
  - conditions nécessaires;
  - conditions suffisantes;
  - cas particuliers,permettent de répondre (assez souvent) à cette question;
- ne pas oublier que la sortie plate a *souvent* une interprétation physique.

## Par décomposition en série

Soit le système  $\Sigma$  formé de :



- si  $\Sigma_1$ , plat de sortie plate  $y(t)$ ,  $\Sigma_2$ , plat de sortie plate  $z(t)$ , alors  $\Sigma$ , plat de sortie plate  $z(t)$ ;
- si  $\Sigma_1$ , plat de sortie plate  $y(t)$ ,  $\Sigma$ , plat de sortie plate  $z(t)$ , alors  $\Sigma_2$ , plat de sortie plate  $z(t)$ ;
- si  $\Sigma$ , plat de sortie plate  $z(t)$ , alors  $\Sigma_2$ , plat de sortie plate  $z(t)$  et  $\Sigma_1$ , plat de sortie plate  $y(t)$ .

Exemple : Prise en compte des actionneurs.

## Systèmes sans dynamique des zéros

ou systèmes linéarisables entrée-état par bouclage statique

Preuve : ils sont caractérisés par une forme normale :

$$\text{pour } i = 1 \text{ à } m, \quad y_i^{(\mu_i)} = a_i(x) + b_i(x)u(t),$$

telle que  $\sum \mu_i = n$  et :

$$\text{rang} \underbrace{\begin{bmatrix} b_1(x) \\ \vdots \\ b_m(x) \end{bmatrix}}_{B(x)} = m.$$

Le bouclage  $u = B(x)^{-1}(v - A(x))$  le transforme en :

$$\text{pour } i = 1 \text{ à } m, \quad y_i^{(\mu_i)} = v_i(t),$$

dont la sortie plate est constituée par tous les  $y_i$  appelées sorties de Brunovski du système.

Comme il existe un difféomorphisme :

$$\begin{bmatrix} y \\ \vdots \\ y^{(\mu-1)} \end{bmatrix} = \Phi(x),$$

on a  $x = \Psi(y, \dots, y^{(\mu-1)})$  donc  $u = \Xi(y, \dots, y^{(\mu)})$ , ce qui clos la démonstration.

## Cas particuliers

1. Pour les systèmes mono-entrée, platitude et linéarisation coïncident.
2. Pour les systèmes linéaires, comme la commandabilité est une CNS d'obtention de la forme de Brunovski :

platitude et commandabilité coïncident

## Systèmes affines en la commande de dimension $n$

Lorsque le nombre de commandes est  $n - 1$  :

platitude et commandabilité coïncident.

Soit le système :

$$\dot{x}(t) = f_0(x) + \sum_{i=1}^{n-1} f_i(x)u_i(t),$$

pour construire sa sortie plate :

1. on élimine les commandes  $\implies \sum_{i=1}^n a_i(x)\dot{x}_i = g(x)$ ;
2. puis :
  - si un des  $a_i(x)$  est nul (e.g.  $a_r$ ) alors une sortie plate est  $(x_1, \dots, x_{r-1}, x_{r+1}, \dots, x_n)$ ;
  - si tous les  $a_i(x)$  sont non nuls, on procède à un changement

de variable :

$$z = \varphi(x_1, x_2, \dots, x_n) \iff x_1 = \psi(z, x_2, \dots, x_n),$$

et on écrit  $g(x)$  :

$$\begin{aligned} g(x) &= a_1(x) \left[ \frac{\partial \psi}{\partial z} \dot{z} + \sum_{i=2}^n \frac{\partial \psi}{\partial x_i} \dot{x}_i \right] + \sum_{i=2}^n a_i(x) \dot{x}_i, \\ &= a_1(x) \frac{\partial \psi}{\partial z} \dot{z} + \sum_{i=2}^n \left[ a_i(x) + a_1(x) \sum_{i=2}^n \frac{\partial \psi}{\partial x_i} \right] \dot{x}_i. \end{aligned}$$

Il suffit de choisir  $\psi$  tel que :

$$\frac{\partial \psi}{\partial x_2} = -\frac{a_2}{a_1},$$

pour se ramener au cas précédent. Une sortie plate est alors donnée par  $(z_1, x_3, \dots, x_n)$ .

## Systèmes sans dérive à 2 commandes

- $n$  états,  $n - 2$  commandes : platitude = commandabilité :
  - si  $n$  est impair, il est 0-plat;
  - si  $n$  est pair, il est 1-plat;
- $n$  états, 2 commandes :

$$\dot{x}(t) = f_1(x)u_1(t) + f_2(x)u_2(t),$$

si on construit la suite de distributions :

$$\Delta_0 = \text{span}\{f_1, f_2\}, \text{ pour } k \geq 1, \Delta_{k+1} = \text{span}\{\Delta_k, [\Delta_k, \Delta_k]\},$$

alors la platitude est équivalente à :

$$\text{pour } k = 0 \text{ à } n - 2, \text{ rang } \Delta_k = k + 2.$$

Exemple : le chariot à remorque.

## Critère des variétés réglées

Critère permettant de mettre en évidence la **non platitude** d'un système.

Soit le système de dimension  $n$  à  $m$  commandes :

$$\dot{x}(t) = f(x, u)$$

dont l'élimination des commandes conduit à un système de  $n - m$  relations  $F(x, \dot{x}) = 0$ .

Critère : si le système est plat, alors :

$$\exists v \neq 0, \forall (\lambda, \mu), F(\lambda, \mu) = 0, \forall a \in \mathbb{R}, F(\lambda, \mu + av) = 0.$$

## Démonstration (esquisse)

- soit une trajectoire particulière  $\bar{x}$  vérifiant  $F(\bar{x}, \bar{x}^{(1)}) = 0$ ;
- si le système est plat alors  $x = A(z, \dots, z^{(\alpha)})$  et :

$$F(A(z, \dots, z^{(\alpha)}), \sum_{i=0}^{\alpha} \frac{\partial A}{\partial z^{(i)}} z^{(i+1)}) = 0;$$

- soit  $\bar{z}$  vérifiant  $\bar{x} = A(\bar{z}, \dots, \bar{z}^{(\alpha)})$  et  $z(t)$  telle que :

$$\text{pour } i = 0 \text{ à } \alpha, z^{(i)} = \bar{z}^{(i)}, \text{ et } z^{(\alpha+1)} = \bar{z}^{(\alpha+1)} + v;$$

- alors :

$$F(\bar{x}, \bar{x}^{(1)} + \left[ \frac{\partial A}{\partial z^{(\alpha)}} \right]_{\bar{z}} v) = 0.$$

## Exemple du pendule inversé

Soit le système constitué par un pendulé inversé sur un chariot dont la position horizontale est commandée :

$$\begin{aligned} (M + m)\ddot{x} + ml \cos \theta \ddot{\theta} - ml \sin \theta \dot{\theta}^2 &= f(t), \\ \frac{7}{3}l\ddot{\theta} + \cos \theta \ddot{x} - g \sin \theta &= 0. \end{aligned}$$

La deuxième relation fournit :

$$\begin{aligned} \frac{7}{3}l(\mu_2 + a\nu_2) + \cos \lambda(\mu_1 + a\nu_1) - g \sin \lambda &= 0, \\ \underbrace{\frac{7}{3}l\mu_2 + \cos \lambda\mu_1 - g \sin \lambda}_0 + a\left(\frac{7}{3}l\nu_2 + \cos \lambda\nu_1\right) &= 0, \end{aligned}$$

Nécessairement, on doit avoir :  $\nu_1 = 0$  donc  $\nu_2 = 0$ .

## Pendule inversé linéarisé

Linéarisation du pendulé inversé autour de  $\theta(t) = 0$  :

$$\begin{aligned}(M + m)\ddot{x} + ml\delta\ddot{\theta} &= f(t), \\ \frac{7}{3}l\delta\ddot{\theta} + \ddot{x} - g\delta\theta &= 0.\end{aligned}$$

La deuxième relation fournit :

$$\underbrace{\frac{7}{3}l\mu_2 + \mu_1 - g\lambda}_{0} + a\left(\frac{7}{3}l\nu_2 + \nu_1\right) = 0,$$

et la relation  $\nu_1 = -\frac{7}{3}l\nu_2$ , permet de montrer que le linéarisé du pendule est *potentiellement* plat (il reste à le montrer!)

## Platitude du pendule linéarisé

- évident : soit  $\rho = \frac{7}{3}l\delta\theta + x$ , alors

$$\rho(t) \implies \ddot{\rho}(t) \implies \delta\theta(t) \implies x(t) \implies f(t);$$

- équation d'état avec  $X = \begin{bmatrix} x & \dot{x} & \delta\theta & \delta\dot{\theta} \end{bmatrix}^T$  et  $k = \frac{3}{(7M+4m)l}$   
:

$$X = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -mkg l & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & k(M+m) & 0 \end{bmatrix} X + \begin{bmatrix} 0 \\ \frac{7kl}{3} \\ 0 \\ -k \end{bmatrix} f(t),$$

commandable donc plat.

## CNS de platitude

- Aranda-Bricaire, Moog, Pom et, 1995 : calcul de la base d'un module cotangent (forme infinitésimale de Brunovski) et conditions d'intégrabilités;
- Pereira da Silva, 2000 : expression des conditions sur une fonction candidate et élimination de variables;
- Chetverikov, 2001 : amélioration du premier par méthode de perturbation;
- Levine, 2004 : simplification des conditions par utilisation des jets infinis et de la forme implicite.

Mais les conditions sont relativement lourdes à mettre en œuvre  
(utilisation du calcul formel)

## Platitude et linéarisation

Il est légitime de se poser la question de la relation entre la platitude d'un modèle non linéaire et la platitude du modèle linéarisé.

Il y a lieu de distinguer :

- linéarisation autour d'un point de fonctionnement;
- linéarisation autour d'une trajectoire.

## Linéarisation autour d'un point

La platitude ne se conserve pas.

### Exemples

1. le pendule inversé sur le chariot : non linéaire non plat et linéarisé autour de 0 plat.
2. le robot unicycle :

$$\begin{aligned}\dot{x} &= \cos \theta u, \\ \dot{y} &= \sin \theta u, \\ \dot{\theta} &= v,\end{aligned}\tag{6}$$

est plat (système chaîné), mais le linéarisé autour de  $\theta = 0$ , non.

## Linéarisation autour d'une trajectoire

La situation est complètement différente et la sortie plate du linéarisé peut donner une idée de la sortie plate du système non linéaire.

Exemple : le robot unicycle.

- trajectoire :  $T_d = (x_d(t), y_d(t), \theta_d(t), u_d(t), v_d(t))$  vérifiant (6);
- linéarisé autour de  $T_d$  :

$$\delta \dot{x} = -u_d \sin \theta_d \delta \theta + \cos \theta_d \delta u,$$

$$\delta \dot{y} = u_d \cos \theta_d \delta \theta + \sin \theta_d \delta u,$$

$$\delta \dot{\theta} = \delta v.$$

- trivial :  $(\delta x, \delta y)$  est une sortie plate de ce système;
- comparer à la linéarisation de la sortie plate  $(x, y)$  du modèle non linéaire!

## Conjecture

La sortie plate du système linéarisé  
est obtenue  
par linéarisation de la sortie plate du système non linéaire.

- la preuve de ce résultat est un problème ouvert;
- permet par intégration de retrouver (*parfois*) la sortie plate du système non linéaire.

## Résultats de Lévine (2004)

- utilisation la forme implicite  $F(x, \dot{x}) = 0$ ;
- système plat si et seulement si linéarisé tangent non stationnaire est plat;
- construction effective des sorties plates par :
  - décompositions de Smith en non stationnaire;
  - intégrations de formes différentielles extérieures;
- algorithme en calcul formel.

## 6 Application aux systèmes linéaires

1. Introduction
2. Définitions et conséquences
3. Exemples pratiques
4. Un peu de formalisme
5. Critères de platitude
6. Cas des systèmes linéaires
7. SPD
8. Conclusion

## Cas des systèmes mono-entrée

Soit le système linéaire :

$$A(p)y(t) = B(p)u(t),$$

où  $A(p)$  et  $B(p)$  sont premiers entre eux :

$$A(p) = p^n + \sum_{i=0}^{n-1} a_i p^i = p^n + A^*(p),$$

$$B(p) = \sum_{i=0}^{n-1} b_i p^i.$$

Etat partiel (Kailath, 1980) :  $z(t)$  tel que :

$$A(p)z(t) = u(t),$$

$$B(p)z(t) = y(t).$$

## L'état partiel est une sortie plate

Existe-t-il  $N(p)$  et  $D(p)$  tels que :

$$z(t) = N(p)y(t) + D(p)u(t)?$$

Il faut qu'ils soient solutions de l'équation de Bezout :

$$N(p)B(p) + D(p)A(p) = 1. \quad (7)$$

Ainsi : platitude et (commandabilité-observabilité) coïncident et

$z(t)$  est une sortie plate du système.

Remarque :  $y(t)$  est une sortie plate si et seulement si  $B(p) = \text{cte}$ .

## Commande par platitude

Suivant la méthode précédente, la commande :

$$u(t) = v(t) + A^*(p)z(t),$$

où :

$$v(t) = z_d^{(n)}(t) + \sum_{i=0}^{n-1} k_i (z_d^{(i)}(t) - z^{(i)}(t)),$$

assure la poursuite asymptotique de la trajectoire  $z_d(t)$ .

Que se passe-t-il si on utilise la définition de la sortie plate?

## Commande par platitude = commande RST

Posons :  $K(p) = p^n + \sum_{i=0}^{n-1} k_i p^i = p^n + K^*(p)$ , alors :

$$u(t) = K(p)z_d(t) + [A^*(p) - K^*(p)]z(t).$$

mais  $z(t) = N(p)y(t) + D(p)u(t)$ , soit :

$$u(t) = K(p)z_d(t) + [A^*(p) - K^*(p)][N(p)y(t) + D(p)u(t)],$$

ce qui donne une commande RST :

$$R(p)u(t) = K(p)z_d(t) - S(p)y(t).$$

## Remarques

$$R(p) = 1 - [A^*(p) - K^*(p)] D(p),$$

$$S(p) = -[A^*(p) - K^*(p)] N(p).$$

1. Dynamique de régulation  $P(p) = A(p)R(p) + B(p)S(p)$  :

$$\begin{aligned} & A(p)R(p) + B(p)S(p) \\ &= A(p) [1 - [A^*(p) - K^*(p)] D(p)] - B(p) [A^*(p) - K^*(p)] N(p), \\ &= A(p) + [K^*(p) - A^*(p)] [N(p)B(p) + D(p)A(p)], \\ &= K(p). \end{aligned}$$

2. Il n'est pas conçu à partir d'un modèle de comportement  $(B_m(p)/A_m(p))$  mais à partir d'une trajectoire à suivre  $K(p)z_d(t)$ .

**Mais...**

Une commande RST ne peut être mise en œuvre que si :

$$\text{degré } S \geq \text{degré } R.$$

Or :

$$\text{deg}(1 - [A^* - K^*] D) = \text{deg}([A^* - K^*] N) - 1,$$

soit ici :

$$\text{degré } S < \text{degré } R.$$

Conclusion : cette commande n'est pas réalisable!

**Que faire?**

## Réalisation de la commande par platitude

Comme  $u(t) = A(p)z(t)$ ,  $y(t) = B(p)z(t)$  : observation de

$Z = \begin{bmatrix} z(t) & \dot{z}(t) & \dots & z^{(n-1)}(t) \end{bmatrix}^T$ , régi par :

$$\dot{Z} = AZ + bu,$$

$$y = cZ,$$

où :

$$A = \begin{bmatrix} & & & 1 \\ & & \ddots & \\ & & & 1 \\ -a_0 & -a_1 & \dots & -a_{n-1} \end{bmatrix}, b = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix},$$

$$c = \begin{bmatrix} b_0 & b_1 & \dots & b_{n-1} \end{bmatrix}.$$

## Première possibilité

Utilisation d'un observateur de Luenberger :

$$\dot{\hat{Z}} = (A - \Gamma c)\hat{Z} + bu + \Gamma y,$$

où  $F = A - \Gamma c$  est de Hurwitz.

Avec  $a = \begin{bmatrix} a_0 & a_1 & \cdots & a_{n-1} \end{bmatrix}$  et  $k = \begin{bmatrix} k_0 & k_1 & \cdots & k_{n-1} \end{bmatrix}$ ,  
la commande s'écrit :

$$u(t) = K(p)z_d(t) + (a - k)(pI - F)^{-1}(bu + \Gamma y).$$

Inconvénient : choix des pôles de l'observateur.

**Deuxième possibilité**

Par dérivations successives :

$$\begin{bmatrix} y \\ \dot{y} \\ \vdots \\ y^{(n-1)} \end{bmatrix} = O_{(A,c)} Z + M_{(A,b,c)} \begin{bmatrix} u \\ \dot{u} \\ \vdots \\ u^{(n-2)} \end{bmatrix},$$

où :

$$O_{(A,c)} = \begin{bmatrix} c \\ cA \\ \vdots \\ cA^{n-1} \end{bmatrix}, \quad M_{(A,b,c)} = \begin{bmatrix} 0 & \dots & \dots & 0 \\ cb & \ddots & & \vdots \\ cAb & cb & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ cA^{n-2}b & \dots & cAb & cb \end{bmatrix}.$$

Comme le système est observable :

$$Z = O_{(A,c)}^{-1} \left\{ \begin{bmatrix} y \\ \dot{y} \\ \vdots \\ y^{(n-1)} \end{bmatrix} - M_{(A,b,c)} \begin{bmatrix} u \\ \dot{u} \\ \vdots \\ u^{(n-2)} \end{bmatrix} \right\}. \quad (8)$$

Soit  $x(-\infty) = 0$  et  $p^{-1}$  l'opérateur d'intégration :

$$p^{-1}x(t) = \int_{-\infty}^t x(\tau)d\tau,$$

$\dot{Z} = AZ + bu$  est opérationnellement équivalent à :

$$Z = Ap^{-1}Z + bp^{-1}u,$$

et en réitérant :

$$\forall \mu \in N, Z = A^\mu p^{-\mu} Z + \sum_{i=1}^{\mu} A^{i-1} bp^{-i} u,$$

En particulier pour  $\mu = n - 1$  :

$$Z = A^{n-1} p^{-(n-1)} Z + \sum_{i=1}^{n-1} A^{i-1} bp^{-i} u.$$

## Observateur exact

Finalemment :

$$\begin{aligned}
 Z &= A^{n-1} O_{(A,c)}^{-1} p^{-(n-1)} \begin{bmatrix} y \\ \dot{y} \\ \vdots \\ y^{(n-1)} \end{bmatrix} \\
 &\quad - A^{n-1} O_{(A,c)}^{-1} M_{(A,b,c)} p^{-(n-1)} \begin{bmatrix} u \\ \dot{u} \\ \vdots \\ u^{(n-2)} \end{bmatrix} + \sum_{i=1}^{n-1} A^{i-1} b p^{-i} u.
 \end{aligned}$$

## Remarques

- (8) permet (en passant) d'obtenir la solution de l'équation de Bezout (7) :

$$N(p) = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} O_{(A,c)}^{-1} \text{diag} \{1, p, \dots, p^{n-1}\},$$

$$D(p) = - \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} O_{(A,c)}^{-1} M_{(A,b,c)} \text{diag} \{1, p, \dots, p^{n-2}\},$$

- la version discrète est l'observateur à réponse pile :  
*cf. Aström, Wittenmark, Computer controlled systems,*  
 Prentice Hall, 1990.

## Commande par platitude et observateur exact

Forme complète de la commande :

$$\begin{aligned}u(t) &= K(p)z_d(t) + [a - k] Z, \\ &= K(p)z_d(t) - S^*(p^{-1})y(t) - Q^*(p^{-1})u(t),\end{aligned}$$

$$S^*(p^{-1}) = [k - a] A^{n-1} O_{(A,c)}^{-1} \begin{bmatrix} p^{-(n-1)} \\ p^{-(n-2)} \\ \vdots \\ p^{-1} \\ 1 \end{bmatrix},$$

$$Q^*(p^{-1}) = [a - k] \times \left\{ \begin{array}{l} A^{n-1} O_{(A,c)}^{-1} M_{(A,b,c)} \\ - \left[ \begin{array}{ccc} A^{n-2}b & \dots & b \end{array} \right] \end{array} \right\} \begin{bmatrix} p^{-(n-1)} \\ p^{-(n-2)} \\ \vdots \\ p^{-1} \end{bmatrix}$$

Avec  $R^*(p^{-1}) = 1 + Q^*(p^{-1})$ , on obtient la forme RST réalisable :

$$R^*(p^{-1})u(t) = K(p)z_d(t) - S^*(p^{-1})y(t).$$

## Exemple

Soit le deuxième ordre avec  $A(p) = p^2 + a_1p + a_0$  et  $B(p) = b_1p + b_0$ , qui correspond à :

$$N(p) = -\frac{b_1}{\delta}p + \frac{b_0 - a_1b_1}{\delta}, \quad D(p) = \frac{b_1^2}{\delta},$$

avec  $\delta = b_0^2 - a_1b_0b_1 + a_0b_1^2 \neq 0$ .

- Sortie plate :  $z(t) = u(t) - py(t)$ ;
- Commande :

$$u(t) = K(p)z_d(t) + [(a_1 - k_1)p + (a_0 - k_0)]z(t),$$

où  $K(p) = p^2 + k_1p + k_0$  définit la dynamique de l'erreur de poursuite.

Réalisation de  $u(t) = [p^2 + a_1p + a_0] z(t)$  et  $y(t) = [b_1p + b_0] z(t)$  :

$$\begin{aligned}\dot{Z}(t) &= \begin{bmatrix} 0 & 1 \\ -a_0 & -a_1 \end{bmatrix} Z(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t), \\ y(t) &= \begin{bmatrix} b_0 & b_1 \end{bmatrix} Z(t),\end{aligned}$$

soit :

$$O_{(A,c)} = \begin{bmatrix} b_0 & b_1 \\ -a_0b_1 & b_0 - a_1b_1 \end{bmatrix}, M_{(A,b,c)} = \begin{bmatrix} 0 \\ b_1 \end{bmatrix}.$$

Donc :

$$Z(t) = \frac{1}{\delta} \begin{bmatrix} (b_0 - a_1b_1) y(t) - b_1\dot{y}(t) + b_1^2u(t) \\ a_0b_1y(t) + b_0\dot{y}(t) - b_0b_1u(t) \end{bmatrix},$$

D'autre part :

$$\begin{aligned}
Z(t) &= \begin{bmatrix} 0 & 1 \\ -a_0 & -a_1 \end{bmatrix} p^{-1} Z(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} p^{-1} u(t), \\
&= \frac{1}{\delta} \begin{bmatrix} a_0 b_1 p^{-1} y(t) + b_0 y(t) - b_0 b_1 p^{-1} u(t) \\ (a_0 b_1 - a_1 b_0) y(t) - a_0 b_0 p^{-1} y(t) + b_0^2 p^{-1} u(t) \end{bmatrix}.
\end{aligned}$$

Ce qui donne la forme RST réalisable avec :

$$\begin{aligned}
R^*(p^{-1}) &= 1 + \frac{1}{\delta} [(k_1 - a_1) b_0^2 - b_0 b_1 (k_0 - a_0)] p^{-1}, \\
S^*(p^{-1}) &= \frac{1}{\delta} ([ (k_1 - a_1) (a_0 b_1 - a_1 b_0) + b_0 (k_0 - a_0) ] \\
&\quad + [a_0 b_1 (k_0 - a_0) - (k_1 - a_1) b_0 a_0] p^{-1}).
\end{aligned}$$

Remarque : Avec  $A^*(p^{-1}) = p^{-2} A(p)$  et  $B^*(p^{-1}) = p^{-2} B(p)$  :

$$A^*(p^{-1}) R^*(p^{-1}) + B^*(p^{-1}) S^*(p^{-1}) = 1 + k_1 p^{-1} + k_0 p^{-2} = p^{-2} K(p).$$

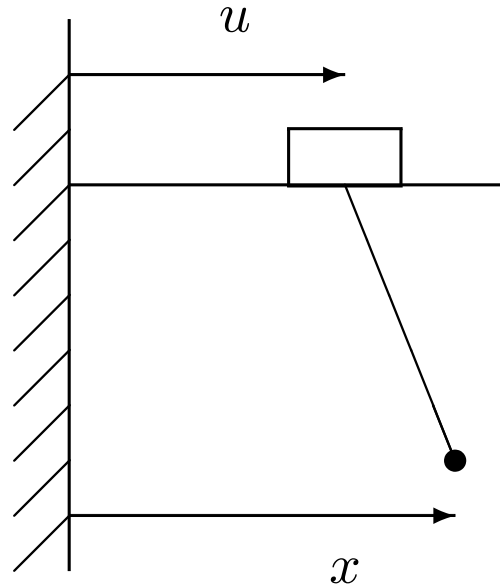
## Extension par formalisme d'état

Problème : extension aux systèmes multi-entrées ou aux systèmes non stationnaires.

Solution : Équations d'état.

- platitude  $\iff$  commandabilité;
- utilisation de la forme canonique commandable;
- détermination systématique d'une sortie plate
- extension "triviale" aux systèmes non stationnaires;
- le RST à mettre en place sera défini sans ambiguïté.

## Exemple de l'oscillateur



Equation :  $\ddot{x}(t) = \omega^2(u(t) - x(t))$ , avec ici  $\omega^2 = 1$ ;

Sortie plate :  $x(t)$

$$u(t) = x(t) + \frac{\ddot{x}(t)}{\omega^2}.$$

## Commande de l'oscillateur

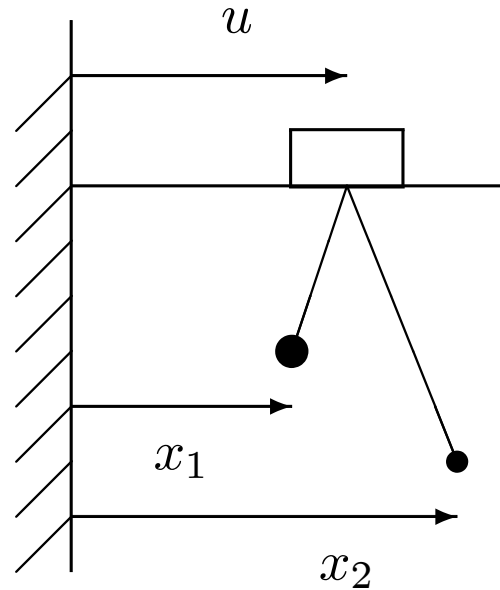
Génération de trajectoire, de  $t = 0$  à  $1$  telle que  $x_d(0) = 0$ ,  
 $x_d(1) = 1$ ,  $\dot{x}_d(0) = \dot{x}_d(1) = \ddot{x}_d(0) = \ddot{x}_d(1) = 0$  :

$$x_d(t) = 6t^5 - 15t^4 + 10t^3;$$

Commande à pôles de poursuite en  $(-10, -10)$  soit un  $t_{r,5\%} \simeq 0.5$  s. :

$$\begin{aligned} u(t) &= x(t) + \ddot{x}_d(t) + 20(\dot{x}_d(t) - \dot{x}(t)) + 100(x_d(t) - x(t)), \\ &= \underbrace{\ddot{x}_d(t) + 20\dot{x}_d(t) + 100x_d(t)}_{\varphi_d(t)=600t^5-900t^4-80t^3+420t^2+60t} - \underbrace{[20\dot{x}(t) + 99x(t)]}_{\text{bouclage d'état}}. \end{aligned}$$

## Exemple de l'oscillateur double



Equations, avec  $\omega_1^2 \neq \omega_2^2$  (sinon non commandable) :

$$\ddot{x}_1(t) = \omega_1^2(u(t) - x_1(t)),$$

$$\ddot{x}_2(t) = \omega_2^2(u(t) - x_2(t)),$$

## Sortie plate de l'oscillateur double

L'élimination de la commande :

$$x_1(t) + \frac{\ddot{x}_1(t)}{\omega_1^2} = x_2(t) + \frac{\ddot{x}_2(t)}{\omega_2^2},$$

$\Updownarrow$

$$\frac{\ddot{x}_1(t)}{\omega_1^2} - \frac{\ddot{x}_2(t)}{\omega_2^2} = x_2(t) - x_1(t).$$

Sortie plate :

$$z(t) = \frac{x_1(t)}{\omega_1^2} - \frac{x_2(t)}{\omega_2^2}.$$

## Paramétrisation de l'oscillateur double

États :

$$\begin{aligned} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} &= \frac{1}{\omega_2^2 - \omega_1^2} \begin{bmatrix} \omega_1^2 & 1 \\ \omega_2^2 & 1 \end{bmatrix} \begin{bmatrix} \ddot{z}(t) \\ \omega_1^2 \omega_2^2 z(t) \end{bmatrix}, \\ &= \frac{1}{\omega_2^2 - \omega_1^2} \begin{bmatrix} \omega_1^2 (\ddot{z}(t) + \omega_2^2 z(t)) \\ \omega_2^2 (\ddot{z}(t) + \omega_1^2 z(t)) \end{bmatrix}. \end{aligned}$$

Commande :

$$u(t) = \frac{1}{\omega_2^2 - \omega_1^2} (\omega_1^2 \omega_2^2 z(t) + (\omega_1^2 + \omega_2^2) \ddot{z}(t) + z^{(4)}(t)).$$

## Commande de l'oscillateur double

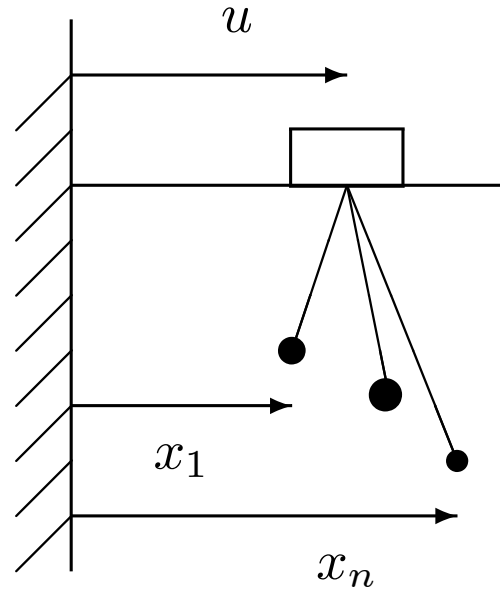
Génération de trajectoire :  $z_d(t)$

Bouclage :

$$\begin{aligned}
 u(t) &= \frac{1}{\omega_2^2 - \omega_1^2} (\omega_1^2 \omega_2^2 z + (\omega_1^2 + \omega_2^2) \ddot{z} + z_d^{(4)}) + \sum_{i=0}^3 k_i (z_d^{(i)} - z^{(i)}), \\
 &= \varphi_d(t) - \\
 &\quad \left( k_3 z^{(3)} + (k_2 - \omega_1^2 - \omega_2^2) z^{(2)} + k_1 z^{(1)} + (k_0 - \omega_1^2 \omega_2^2) z \right).
 \end{aligned}$$

Rappel : le polynôme  $p^4 + \sum_{i=0}^3 k_i p^i$  fixe les dynamiques de la poursuite de trajectoire.

## Exemple de l'oscillateur multiple



Equations, avec  $\omega_i^2 \neq \omega_j^2$  pour  $i \neq j$  (sinon non commandable) :

$$\text{pour } i = 1 \text{ à } n, \quad \ddot{x}_i(t) = \omega_i^2(u(t) - x_i(t)).$$

L'élimination des commandes conduit aux relations :

$$x_1(t) + \frac{\ddot{x}_1(t)}{\omega_1^2} = \dots = x_n(t) + \frac{\ddot{x}_n(t)}{\omega_2^2},$$

à partir desquelles il n'est pas évident de détecter une sortie plate!

## Equation d'état de l'oscillateur multiple

$$\dot{X}(t) = \begin{bmatrix} \ddots & & & \\ & -\omega_i^2 & & \\ & & \ddots & \\ & & & \ddots \end{bmatrix} X(t) + \begin{bmatrix} \vdots \\ \omega_i^2 \\ \vdots \end{bmatrix} u(t).$$

Matrice de commandabilité :

$$C = \begin{bmatrix} \vdots & \vdots & & \vdots \\ \omega_i^2 & -\omega_i^4 & \cdots & (-1)^n \omega_i^{2n} \\ \vdots & \vdots & & \vdots \end{bmatrix},$$

Transfert entrée-état :

$$(pI - A)^{-1} B = \left[ \cdots \quad \frac{\omega_i^2}{p + \omega_i^2} \quad \cdots \right]^T.$$

## Recherche d'une sortie plate

*Rappel* : la sortie plate d'un système mono-entrée correspond à un transfert entrée-sortie plate où le numérateur est une constante.

Il suffit de chercher  $C = \begin{bmatrix} c_1 & \cdots & c_n \end{bmatrix}$  tel que :

$$\sum_{i=1}^n \frac{c_i \omega_i^2}{p + \omega_i^2} = \frac{\prod_{i=1}^n c_i \omega_i^2}{\prod_{i=1}^n p + \omega_i^2},$$

alors une sortie plate sera donnée par :

$$z(t) = CX(t).$$

Cela revient à une décomposition en éléments simples :

$$\frac{1}{\prod_{i=1}^n p + \omega_i^2} = \sum_{i=1}^n \frac{r_i}{p + \omega_i^2}, \text{ avec } r_i = \frac{1}{\prod_{j=1, j \neq i}^n (\omega_j^2 - \omega_i^2)}.$$

Donc :

$$c_i = \frac{1}{\omega_i^2 \prod_{j=1, j \neq i}^n (\omega_j^2 - \omega_i^2)}$$

Sortie plate :

$$z(t) = \sum_{i=1}^n \frac{x_i(t)}{\omega_i^2 \prod_{j=1, j \neq i}^n (\omega_j^2 - \omega_i^2)}.$$

Mais : démarche difficilement généralisable au cas multi-entrées.

## Formalisme d'état

Soit le système linéaire :

$$\dot{X}(t) = AX(t) + Bu(t),$$

supposé commandable.

Il existe un changement de variables :

$$x_C(t) = T_C X(t),$$

(algorithme de Luenberger, *cf. cours de systèmes multi-entrées*) tel que :

$$\dot{x}_C = A_C x_C + B_C \underbrace{H_C u}_{u_c},$$



$$B_C = T_C B = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad H_C = \begin{bmatrix} 1 & \times & \times \\ 0 & 1 & \times \\ 0 & \dots & 1 \end{bmatrix}.$$

## Sortie plate

$x_C(t) = \left[ x_C^{1T}(t) \quad x_C^{2T}(t) \quad \dots \quad x_C^{mT}(t) \right]^T$  où chaque  $x_C^i(t)$  est un vecteur de dimension  $\mu_i$ ,  $i$ -ième indice de commandabilité.

Soit  $z_i$  la première composante de  $x_C^i(t)$  alors :

$$z(t) = \begin{bmatrix} z_1(t) \\ z_2(t) \\ \vdots \\ z_m(t) \end{bmatrix} \text{ est une sortie plate.}$$

Les  $z_2(t)$  s'appellent les sorties de Brunovski du système.

## Preuve de la platitude

D'après la forme canonique commandable :

$$u_{C,i} = z_i^{(\mu_i)}(t) + \sum_{j=1}^m \left( \sum_{k=0}^{\mu_j-1} a_{i,j,k} z_j^{(k)}(t) \right), \text{ et } u(t) = H_C^{-1} u_C(t).$$

Remarque :

$$X(t) = T_C^{-1} x_C(t) = \sum_{k=0}^{\mu-1} M_k z^{(k)}(t),$$

$$u(t) = \sum_{k=0}^{\mu} N_k z^{(k)}(t),$$

où  $\mu = \max_i \mu_i$  est l'index de commandabilité.

Remarque :  $N_\mu$  n'est égal à  $H_C^{-1}$  que si tous les  $\mu_i$  sont égaux.

## Résolution du découplage

Il suffit de poser le bouclage :

$$u_{C,i}(t) = v_i(t) - \sum_{j=1}^m A_{i,j} x_C^j(t),$$

pour obtenir le système découplé :

$$\text{pour } i = 1 \text{ à } m, \quad z_i^{(\mu_i)}(t) = v_i(t)$$

Le retour découplant s'écrit finalement :

$$u(t) = H_C^{-1} v(t) - H_C^{-1} \begin{bmatrix} A_1 & A_2 & \cdots & A_m \end{bmatrix} T_C x(t).$$

## Retour stabilisant

pour  $i = 1$  à  $m$ ,

$$\begin{aligned} v_i(t) &= z_{d,i}^{(\mu_i)}(t) + \sum_{k=0}^{\mu_j-1} \kappa_{i,k} \left( z_{d,i}^{(k)}(t) - z_j^{(k)}(t) \right), \\ &= K_i(p)z_{d,i}(t) - K_i x_C^i(t). \end{aligned}$$

Soit globalement :

$$u(t) = \varphi_d(t) - \underbrace{H_C^{-1} \left\{ \text{diag} \{K_i\} + \begin{bmatrix} A_1 & A_2 & \cdots & A_m \end{bmatrix} \right\}}_{\text{gain de retour d'état : } K} T_C x(t).$$

## Systèmes non stationnaires

Soit le système linéaire :

$$\dot{X}(t) = A(t)X(t) + B(t)u(t),$$

Critère de commandabilité (Silverman, Meadows) :

- opérateur différentiel :  $C^0(M(t)) = M(t)$ , et pour tout  $i > 1$  :

$$C^{i+1}(M(t)) = A(t)C^i(M(t)) - \dot{C}^i(M(t));$$

- test de commandabilité sur un intervalle de temps  $T = [t_1, t_2]$  :  
si il existe un entier  $\mu$  tel que la matrice :

$$C_{\{\mu\}}(t) = \begin{bmatrix} C^0(B(t)) & C^1(B(t)) & \dots & C^{\mu-1}(B(t)) \end{bmatrix},$$

soit de rang  $n$ , sur  $T$  alors le système est commandable sur  $T$ .

## Forme commandable non stationnaires

Système commandable  $\implies$  il existe un changement de variables :

$$x_C(t) = T_C(t)X(t),$$

(algorithme de Seal-Stubberud) tel que :

$$\dot{x}_C = A_C(t)x_C + B_C(t)\underbrace{H_C(t)u}_{u_c},$$

où :

$$A_C(t) = T_C(t)A(t)T_C^{-1}(t) + \dot{T}_C(t)T_C^{-1}(t)$$

et  $A_C(t)$ ,  $B_C(t)$ ,  $H_C(t)$  ont la même structure qu'en stationnaire mais où les  $\times$  dépendent de  $t$ .

## Commande par platitude en non stationnaires

- soit  $z_i$  la première composante de  $x_C^i(t)$  alors :

$$z(t) = \begin{bmatrix} z_1(t) \\ z_2(t) \\ \vdots \\ z_m(t) \end{bmatrix} \text{ est une sortie plate;}$$

- commande découplante avec poursuite asymptotique :

$$u(t) = \varphi_d(t) - \underbrace{H_C^{-1} \left\{ \text{diag} \{K_i\} + \begin{bmatrix} A_1 & \cdots & A_m \end{bmatrix} \right\}}_{\text{gain de retour d'état : } K(t)} T_C x(t).$$

## Exemple du moteur

Moteur à courant continu à excitation séparée dont le flux statorique  $\Phi(t)$  varie au cours du temps :

$$L \frac{dI(t)}{dt} + RI(t) = V(t) - K_e \Phi(t) \Omega(t),$$
$$J \frac{d\Omega(t)}{dt} + f\Omega(t) = K_m \Phi(t) I(t),$$

où :

- $\Phi(t)$  : le flux induit par le stator;
- $I(t)$  : courant rotorique;
- $V(t)$  : tension rotorique (commande);
- $\Omega(t)$  : vitesse de rotation du rotor (sortie).

## Forme canonique commandable

Matrice de commandabilité :

$$C_{\{2\}}(t) = \begin{bmatrix} \frac{1}{L} & -\frac{R}{L} \\ 0 & \frac{K_m \dot{\Phi}(t)}{JL} \end{bmatrix},$$

$\forall t$ , commandable  $\iff \Phi(t)$  est toujours non nul et borné;

Forme commandable :

$$\dot{Z}(t) = \begin{bmatrix} 0 & 1 \\ -\Psi_0(t) & -\Psi_1(t) \end{bmatrix} Z(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} V(t),$$

$$J\Omega(t) = \begin{bmatrix} K_m L \Phi(t) & 0 \end{bmatrix} Z(t),$$

$$\Psi_0(t) = \frac{\ddot{\Phi}\Phi - \dot{\Phi}^2}{\Phi^2} + \frac{K_m K_e}{JL} \Phi^2 + \frac{Rf}{JL} + \frac{R\dot{\Phi}}{J\Phi}, \quad \Psi_1(t) = \frac{\dot{\Phi}}{\Phi} + \frac{f}{J} + \frac{R}{L}.$$

## Platitude

Sortie plate :  $z(t)$ , première composante du vecteur  $Z(t)$ .

Commande par platitude : trajectoire désirée  $z^d(t)$  et :

$$V(t) = \ddot{z}^d + k_1 \left( \dot{z}^d - \dot{\hat{z}} \right) + k_0 \left( z^d - \hat{z} \right) + \Psi_1 \dot{\hat{z}} + \Psi_0 \hat{z},$$

où  $\dot{\hat{z}}(t)$  et  $\hat{z}(t)$  sont les valeurs observées de la sortie plate et sa dérivée.

Utilisation de la forme observable

## Formes différentielles entrée-sortie

Par élimination du courant :

$$\ddot{\Omega} + \left( \frac{f}{J} + \frac{R}{L} - \frac{\dot{\Phi}}{\Phi} \right) \dot{\Omega} + \left( \frac{K_m K_e \Phi^2}{JL} - \frac{f\dot{\Phi}}{J\Phi} + \frac{Rf}{LJ} \right) \Omega = \frac{K_m \Phi}{JL} V.$$

transformable en :

$$\ddot{\Omega} + \underbrace{\left[ \left( \frac{f}{J} + \frac{R}{L} - \frac{\dot{\Phi}}{\Phi} \right) \Omega \right]}_{\alpha_1(t)} \quad (1) \quad + \quad \underbrace{\left( \frac{K_m K_e}{JL} - \frac{f\dot{\Phi}}{J\Phi} + \frac{Rf}{LJ} + \frac{\ddot{\Phi}\Phi - \dot{\Phi}^2}{\Phi^2} \right)}_{\alpha_0(t)} \Omega = \frac{K_m \Phi}{JL} V.$$

**Forme canonique observable**

Pour les systèmes mono-entrée mono-sortie, réalisation de la 2<sup>ème</sup> forme différentielle entrée-sortie conduit à la forme observable :

$$\begin{aligned}\dot{X}(t) &= \begin{pmatrix} 0 & -\alpha_0(t) \\ 1 & -\alpha_1(t) \end{pmatrix} X(t) + \begin{pmatrix} \beta(t) \\ 0 \end{pmatrix} V_r(t) \\ \Omega(t) &= \begin{pmatrix} 0 & 1 \end{pmatrix} X(t)\end{aligned}$$

## Observateur réduit

On a :

$$\dot{x}_1(t) = -\alpha_0(t)\Omega(t) + \beta(t)V, \text{ et } x_1(t) = \dot{\Omega}(t) + \alpha_1(t)\Omega(t)$$

Observateur “naïf” :

$$\dot{\hat{x}}_1(t) = -\alpha_0(t)\Omega(t) + \beta(t)V + L(t) \left( \dot{\Omega}(t) + \alpha_1(t)\Omega(t) - \hat{x}_1(t) \right)$$

Pour éliminer la dérivée de la sortie, posons  $\xi(t) = \hat{x}_1(t) - L(t)y(t)$  :

$$\dot{\xi}(t) = -L(t)\xi(t) + \beta(t)V + \left( -\alpha_0(t) + \alpha_1(t)L(t) - L^2(t) - \dot{L}(t) \right) \Omega(t)$$

Remarque :  $L(t)$  peut être pris égal à une constante positive  $\lambda$ .

## Observateur réduit (2)

Avec  $L(t) = \lambda$  :

$$\dot{\xi}(t) = -\lambda\xi(t) + \beta(t)V_r(t) + (-\alpha_0(t) + \lambda\alpha_1(t) - \lambda^2)\Omega(t).$$

Estimation de  $\hat{x}_1$  :

$$\hat{x}_1(t) = \xi(t) + L(t)y(t)$$

et de  $\hat{Z}(t)$  par :

$$\hat{Z}(t) = \frac{1}{K_m\Phi(t)} \begin{bmatrix} 0 & JL \\ JL & -(fL + RJ) \end{bmatrix} \hat{X}(t).$$

## Génération de trajectoire

Trajectoire plate désirée :  $z^d(t) =$

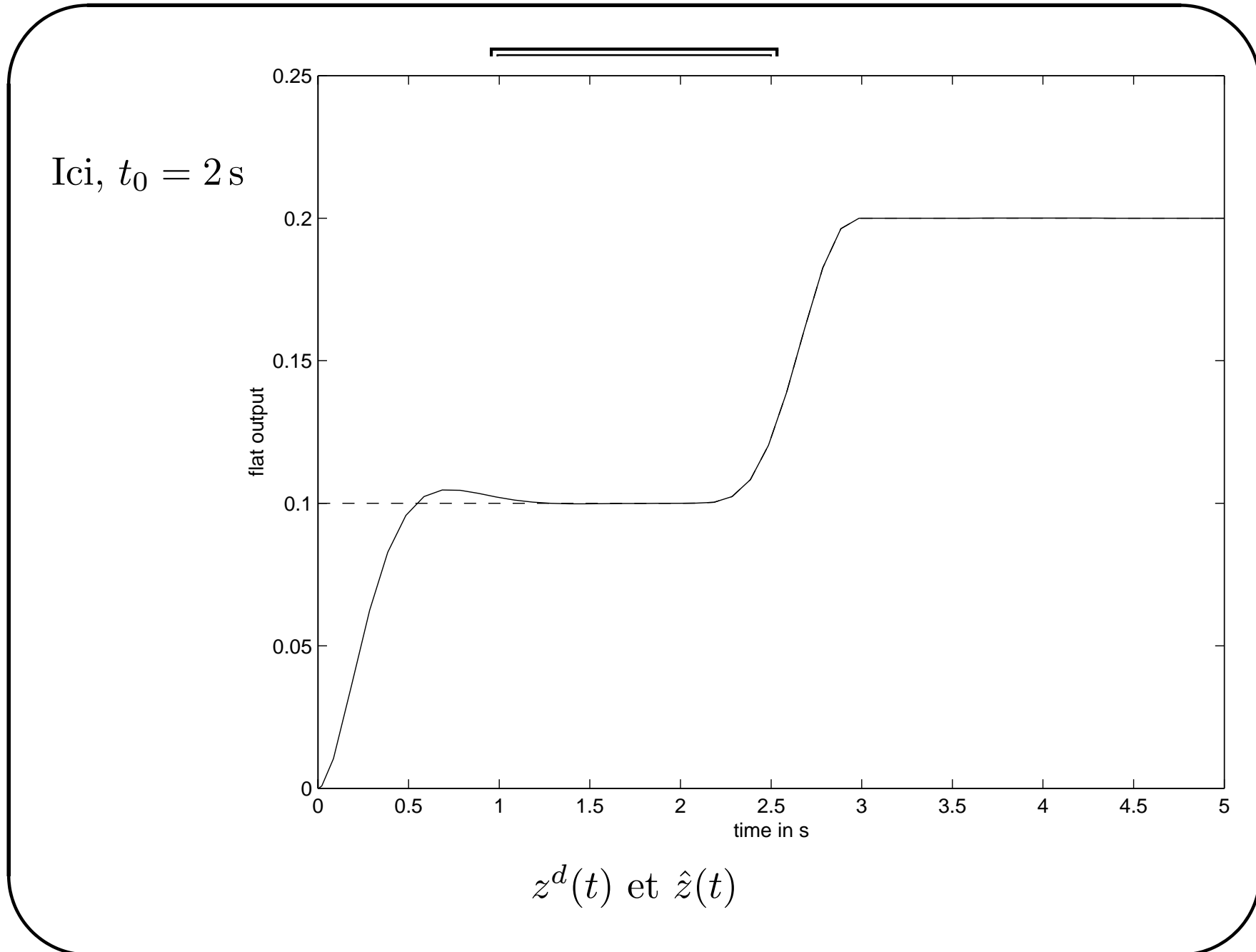
$$\begin{cases} 0.1 & \text{si } t \leq t_0 \\ z^d(t_0) + (21\tau^5 - 35\tau^6 + 15\tau^7) (z^d(t_f) - z^d(t_0)) & \text{si } t_0 \leq t \leq t_f \\ 0.2 & \text{si } t \geq t_f \end{cases}$$

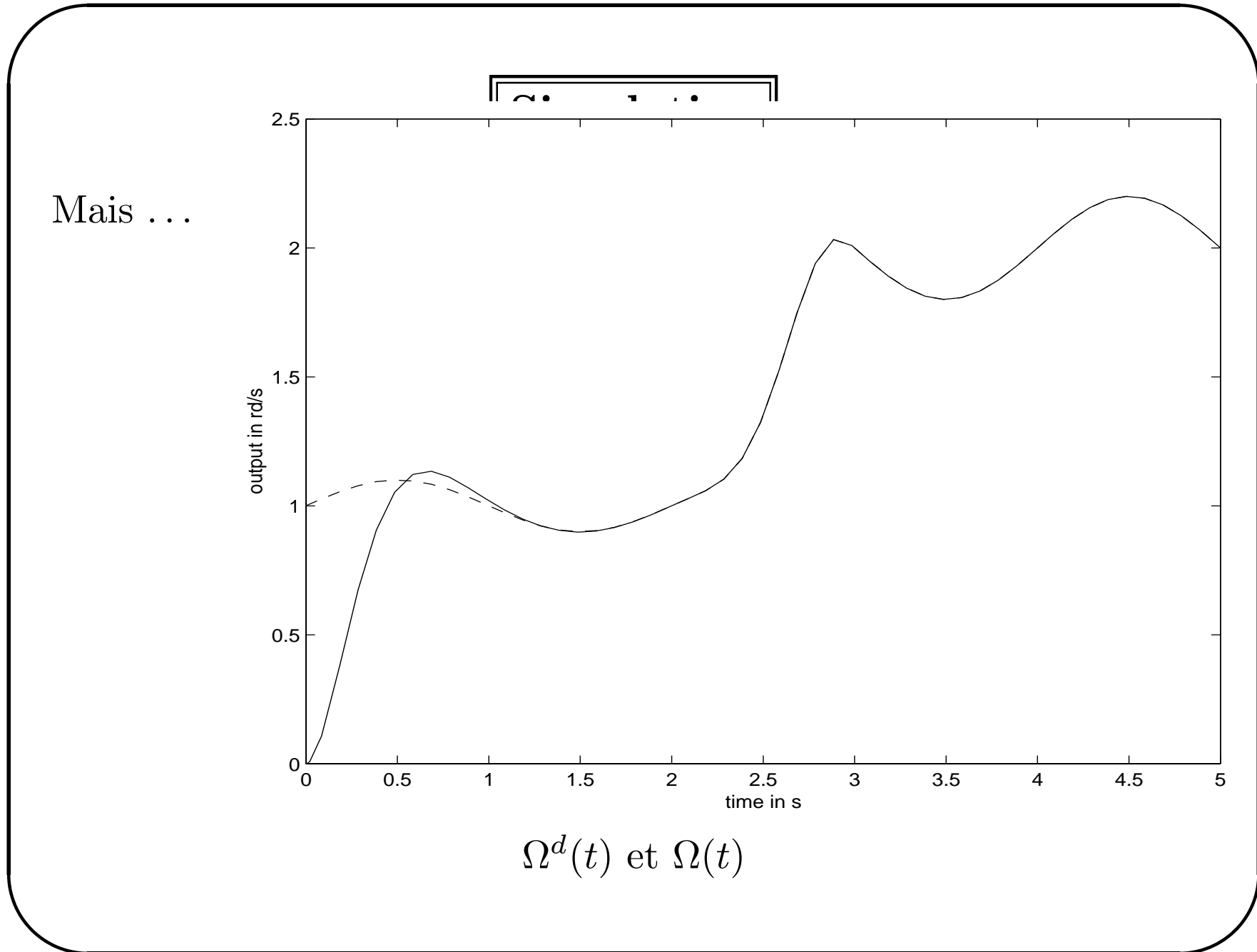
avec :

$$\tau = \frac{t - t_0}{t_f - t_0}.$$

Choix des pôles :

- poursuite :  $-4.20 + 4.28i$  et  $-4.20 - 4.28i$ ;
- observateur :  $-5$ .





## Poursuite de vitesse

Il suffit d'inverser la relation sans dynamique liant  $\Omega(t)$  et  $z(t)$  :

$$z^d(t) = \frac{J}{K_m L \Phi(t)} \Omega^d(t).$$

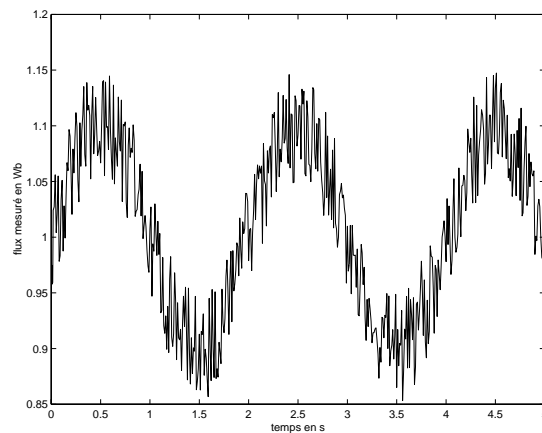
où  $\Omega^d(t)$  est le profil de vitesse désiré. Par exemple pour une vitesse angulaire constante désirée  $\Omega^d(t) = \Omega^d$  :

$$\dot{z}^d(t) = -\frac{J\Omega^d\dot{\Phi}(t)}{K_m L \Phi^2(t)}, \quad \ddot{z}^d(t) = \frac{J\Omega^d(2\dot{\Phi}^2(t) - \Phi(t)\ddot{\Phi}(t))}{K_m L \Phi^3(t)}.$$

Comme  $\hat{X}(t) = \begin{bmatrix} \xi(t) + \lambda\Omega(t) & \Omega(t) \end{bmatrix}^T$ , et  $\hat{Z}(t) = P(t)\hat{X}(t)$ , on obtient le régulateur-observateur générant la commande sous la forme :

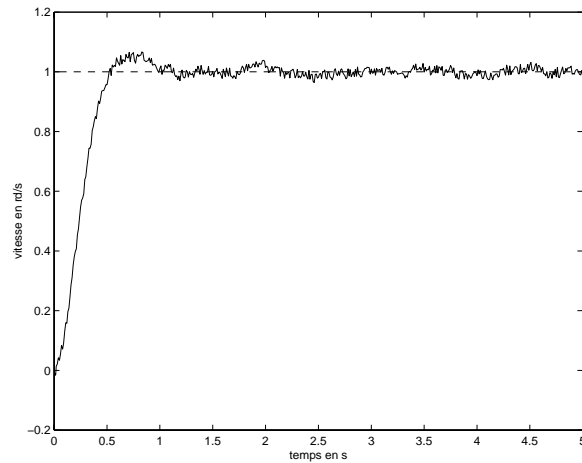
$$V(t) = (p^2 + k_1 p + k_0) \frac{J}{K_m L_r \Phi} \Omega^d - \begin{bmatrix} k_0 - \psi_0 & k_1 - \psi_1 \end{bmatrix} P \hat{X}.$$

## Résultats expérimentaux



Mesure du flux induit.

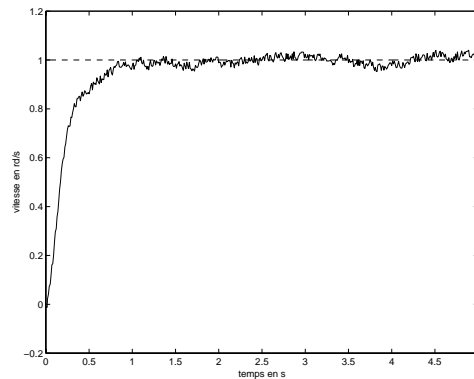
## Résultats expérimentaux



Vitesse de rotation du rotor.

## Test de robustesse

Même commande sur un moteur présentant des écarts de -20% pour le paramètre  $J$  et -50% pour le paramètre  $L_r$ .

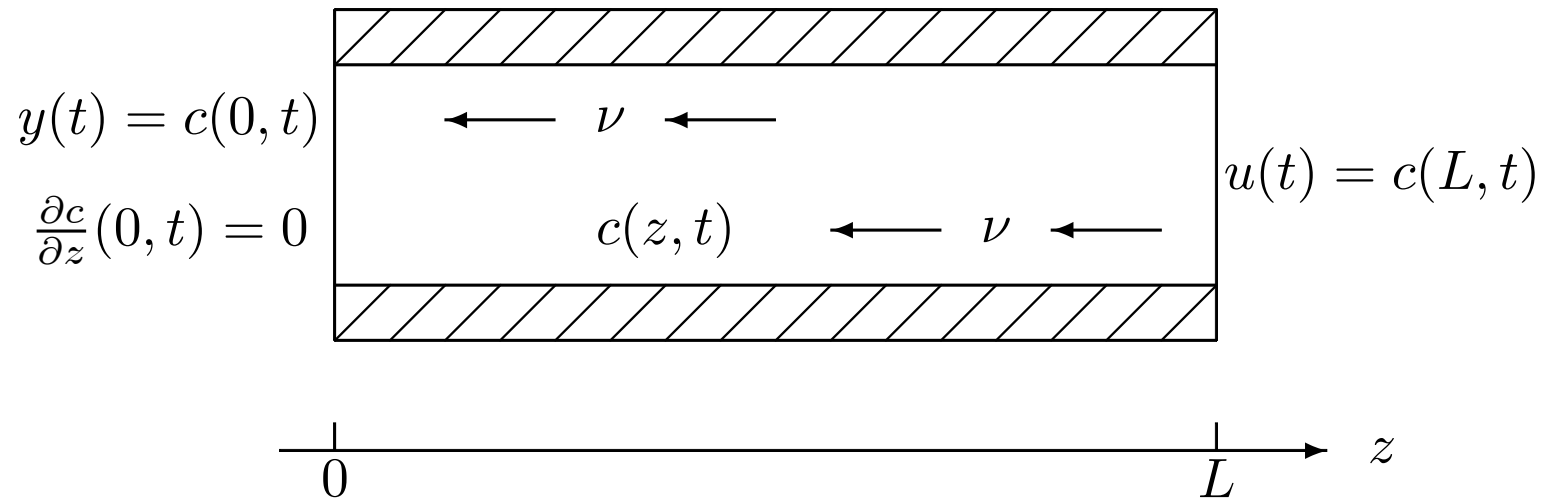


Effet d'une erreur sur les paramètres.

## 7 Systèmes à paramètres distribués

1. Introduction
2. Définitions et conséquences
3. Exemples pratiques
4. Un peu de formalisme
5. Critères de platitude
6. Cas des systèmes linéaires
7. SPD
8. Conclusion

## Réacteur tubulaire



variables :

$c(z, t)$  : concentration du réactant,

$u(t)$  : commande frontière amont,

$y(t)$  : sortie frontière aval,

## Réacteur tubulaire : modélisation

Modélisation : 
$$\frac{\partial c(z, t)}{\partial t} = \gamma \frac{\partial^2 c(z, t)}{\partial z^2} - \nu \frac{\partial c(z, t)}{\partial z} - rc(z, t),$$

Conditions limites :  $c(L, t) = u(t), \quad \frac{\partial c(0, t)}{\partial z} = 0,$

Conditions initiales :  $c(z, 0) = 0,$

Sortie :  $y(t) = c(0, t).$

coefficients significatifs :

$\gamma > 0$  : dispersion axiale,

$\nu \leq 0$  : vitesse interstitielle du réactant,

$r$  : taux de réaction (cinétique de la réaction).

## Réacteur tubulaire : simplification

$$x(z, t) = \exp \left[ \left( r + \frac{\nu^2}{4\gamma} \right) t - \frac{\nu}{2\gamma} z \right] c(z, t),$$

⇒ équation de diffusion :

$$\frac{\partial x(z, t)}{\partial t} = \gamma \frac{\partial^2 x(z, t)}{\partial z^2}$$

Conditions limites :  $x(L, t) = \exp \left[ \frac{\nu L}{2\gamma} - \left( r + \frac{\nu^2}{4\gamma} \right) t \right] u(t),$

$$\frac{\partial x(0, t)}{\partial z} = -\frac{\nu}{2\gamma} x(0, t),$$

Conditions initiales :  $x(z, 0) = 0,$

Sortie :

$$y(t) = \exp \left[ - \left( r + \frac{\nu^2}{4\gamma} \right) t \right] x(0, t).$$

## Réacteur tubulaire : platitude

Par résolution analytique :

$$x(z, t) = \sum_{i \geq 0} a_i(t) \frac{z^i}{i!}.$$

Soit  $x(0, t) = \eta(t)$ , alors, pour  $k \geq 0$  :

$$\begin{cases} a_{2k}(t) = \frac{1}{\gamma^k} \eta^{(k)}(t), \\ a_{2k+1}(t) = -\frac{\nu}{2\gamma^{k+1}} \eta^{(k)}(t), \end{cases}$$

soit :

$$x(z, t) = \sum_{k \geq 0} \left[ 1 - \frac{\nu}{2(2k+1)\gamma} z \right] \frac{z^{2k}}{(2k)! \gamma^k} \eta^{(k)}(t).$$

D'autre part :

$$\begin{aligned}
 u(t) &= \exp \left[ \left( r + \frac{\nu^2}{4\gamma} \right) t - \frac{\nu L}{2\gamma} \right] x(L, t), \\
 &= \exp \left[ \left( r + \frac{\nu^2}{4\gamma} \right) t - \frac{\nu L}{2\gamma} \right] \sum_{k \geq 0} \left[ 1 - \frac{\nu L}{2(2k+1)\gamma} \right] \frac{L^{2k}}{(2k)! \gamma^k} \eta^{(k)}(t),
 \end{aligned}$$

indique que  $\eta(t)$  permet d'exprimer toutes les variables  $x(z, t)$ ,  
 $z \in [0, L]$  et  $u(t)$ .

## Remarques

1. Si on cherche  $\eta(t)$  en fonction des autres variables :  
la relation d'endogénéité n'est pas satisfaite MAIS on considère quand même  $\eta(t)$  comme sortie plate du système.

2. Comme :

$$\eta(t) = \exp \left[ \left( r + \frac{\nu^2}{4\gamma} \right) t \right] y(t),$$

$y(t)$  est sortie plate de ce système.

3. La trajectoire désirée  $y_d(t)$  doit être infiniment dérivable :

- polynômes;
- fonctions Gevrey : fonctions dont une série des dérivées itérées converge.

4. Commande boucle ouverte :

$$y_d(t) \implies \eta_d(t) \implies x_d(L, t) \implies u_d(t).$$

## Fonctions Gevrey

Nécessaires à utiliser pour éviter les problèmes aux extrémités.

Définition :  $f(t)$  de  $[0, T]$  dans  $R$  est de Gevrey d'ordre  $\alpha$  si :

$$\exists(m, \gamma), \forall t \in [0, T], \forall k \geq 0, \quad \left| f^{(k)}(t) \right| \leq m \frac{(k!)^\alpha}{\gamma^k}.$$

Exemple : fonction de Gevrey d'ordre  $d > 1$ ,

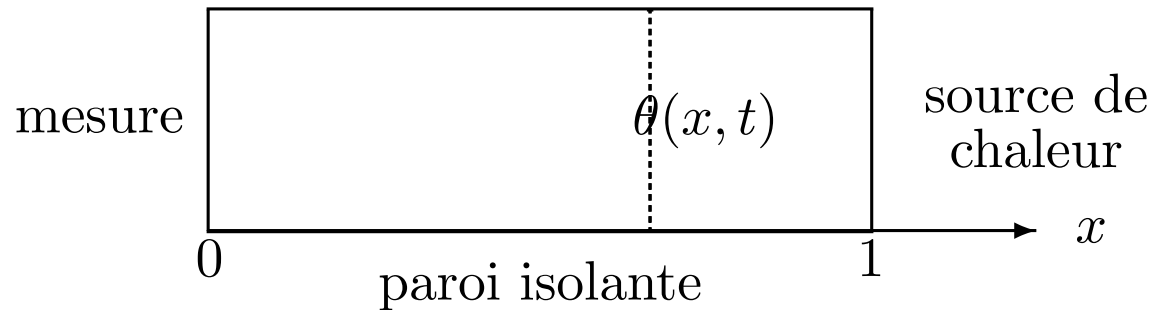
$$f(t) = \begin{cases} 0 & \text{pour } t \leq 0, \\ \frac{\int_0^{t/T} \exp\left(-\frac{1}{\tau^d(1-\tau)^d}\right) d\tau}{\int_0^1 \exp\left(-\frac{1}{\tau^d(1-\tau)^d}\right) d\tau} & \text{pour } 0 \leq t \leq T, \\ 1 & \text{pour } t \geq T. \end{cases}$$

## Commande du réacteur

- génération de trajectoire : on prend pour  $y_d(t)$  une fonction de Gevrey d'ordre  $\alpha < 2$ ;
- en pratique, on tronque la série à un ordre arbitraire  $N$ ;
- on génère une commande en boucle fermée comme en dimension finie à partir de cette troncature.

## Equation de la chaleur

Barre chauffée à une extrémité :



Equation de la chaleur : 
$$\frac{\partial \theta(x, t)}{\partial t} = \frac{\partial^2 \theta(x, t)}{\partial x^2};$$

Conditions limites : 
$$\frac{\partial \theta(0, t)}{\partial x} = 0, \theta(x, 0) = 0;$$

Commande : 
$$u(t) = \theta(1, t).$$

## Résolution opérationnelle

Avec  $p \triangleq \frac{d}{dt}$  :

$$x''(z, t) = px(z, t),$$

soit :

$$\theta(x, t) = \cosh(\sqrt{p}x)A(t) + \sinh(\sqrt{p}x)B(t).$$

Conditions limites  $\implies$

$$\theta(x, t) = \frac{\cosh(\sqrt{p}x)}{\cosh(\sqrt{p})}u(t).$$

**Platitude**

Soit la sortie  $y(t) = \theta(0, t)$ , alors :

$$\theta(x, t) = \cosh(\sqrt{p}x)y(t),$$

et :

$$u(t) = \cosh(\sqrt{p})y(t).$$

Conséquence :  $y(t)$  est une sortie plate de ce système.

## Remarques

- le caractère endogène de  $y(t)$  n'est pas établi;
- on a retrouvé la même interprétation de la sortie plate à partir du transfert en tant qu'état partiel que dans le cas des systèmes linéaires de dimension finie;

- comme :

$$\cosh(\sqrt{p}x) = \sum_{i \geq 0} \frac{x^{2i} p^i}{(2i)!},$$

on a :

$$\theta(x, t) = \sum_{i \geq 0} \frac{x^{2i} y^{(i)}(t)}{(2i)!} \quad \text{et} \quad u(t) = \sum_{i \geq 0} \frac{y^{(i)}(t)}{(2i)!},$$

et on obtient le même résultat qu'avec l'écriture en série;

- utilisation de fonctions de Gevrey.

## Commande par troncature

Soit  $y_d(t)$  sur  $[0, T]$  et la troncature :

$$U_N(t) = \sum_{i=0}^{N-1} \frac{y^{(i)}(t)}{(2i)!} + \frac{y^{(N)}(t)}{(2N)!}.$$

Commande mise en œuvre :

$$\begin{aligned} U_N(t) &= \sum_{i=0}^{N-1} \frac{y^{(i)}(t)}{(2i)!} + \frac{y_d^{(N)}(t) + \sum_{i=0}^{N-1} k_i (y_d^{(i)}(t) - y^{(i)}(t))}{(2N)!}, \\ &= \frac{y_d^{(N)}(t) + \sum_{i=0}^{N-1} k_i y_d^{(i)}(t)}{(2N)!} - \sum_{i=0}^{N-1} \left( k_i \frac{(2i)!}{(2N)!} - 1 \right) \frac{y^{(i)}(t)}{(2i)!}. \end{aligned}$$

où les  $k_i$  assurent une poursuite asymptotique de la trajectoire.

## Choix de la troncature

Pour choisir  $N$  on remarque que :

$$\frac{(y_d^{(N)}(t) - y^{(N)}(t)) + \sum_{i=0}^{N-1} k_i (y_d^{(i)}(t) - y^{(i)}(t))}{(2N)!} = \sum_{i \geq N+1} \frac{y^{(i)}(t)}{(2i)!} \approx \sum_{i \geq N+1} \frac{y_d^{(i)}(t)}{(2i)!}.$$

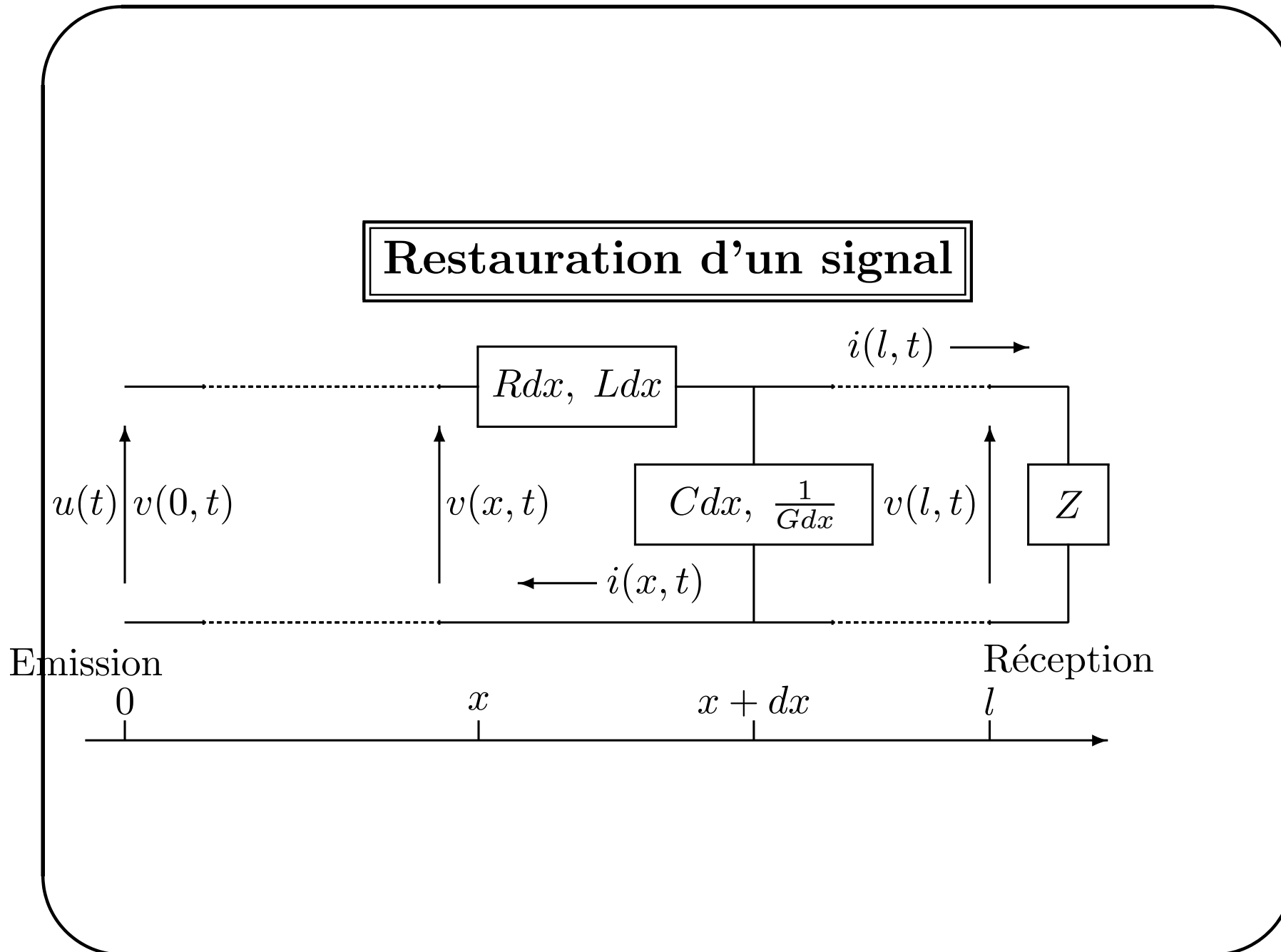
Soit  $N$  tel que, sur  $[0, T]$  :

$$\left| \sum_{i \geq N+1} \frac{y_d^{(i)}(t)}{(2i)!} \right| \leq \varepsilon \text{ fixé,}$$

alors, sur  $[0, T]$  :

$$|y_d(t) - y(t)| \leq \frac{\varepsilon(2N)!}{k_0}.$$

Le choix de la précision désirée permet de fixer  $N$ .



## 8 Conclusion

*(provisoire ... )*

1. Introduction
2. Définitions et conséquences
3. Exemples pratiques
4. Un peu de formalisme
5. Critères de platitude
6. Cas des systèmes linéaires
7. SPD
8. Conclusion

## Systèmes non plats

De nombreux exemples (bille sur rail, pendule de Kapitza, double pendule inversé, tracteur à deux remorques déportées, *etc...*) indiquent qu'il existe des systèmes non plats : une ou plusieurs des variables fait défaut.

Exemple de système non plat :

$$\dot{x}_1 = x_3^2 - x_2,$$

$$\dot{x}_2 = x_3,$$

$$\dot{x}_3 = u.$$

En effet on peut obtenir  $u$  et  $x_3$  à partir de  $x_2$ , mais on n'obtient que  $\dot{x}_1$  :  $x_1$  fait défaut!

Lorsque le système considéré n'est pas plat, on dispose de techniques pour appliquer les principes de la platitude.

## Commandes de systèmes non plats

1. Commande à deux dynamiques ou commande haute fréquence :

$$u(t) = u_0(t) + u_1(t) \sin \frac{t}{\varepsilon},$$

on est conduit à un système *moyen* qui peut être plat.

2. Approximation par un système plat en éliminant les termes qui nuisent à la platitude.
3. Immersion dans un système plat plus grand.
4. Approximation par un système bilinéaire plat.
5. Utilisation du caractère liouvillien du système (c'est à dire que les variables en défaut peuvent être définies par des intégrations à partir d'une sortie plate).

## Principes généraux

Dans les quatres derniers cas le principe de la commande par platitude consiste :

1. à générer une trajectoire désirée;
2. à linéariser le système non linéaire autour de cette trajectoire: on obtient alors un système linéaire non stationnaire;
3. s'il est commandable, il est plat;
4. construire un bouclage linéaire non stationnaire de poursuite de trajectoire.

Deuxième structure de commande!

## Que reste-t-il à faire?

- simplifier les algorithmes et diminuer la complexité;
- intégrer des contraintes de structure (robustesse ou rejet de perturbations);
- étendre aux autres systèmes : SPD non linéaires, à retards, fractionnaires;
- étudier la commande numérique;
- générer les trajectoires sous contraintes;
- appliquer le concept de platitude à d'autres problèmes de l'Automatique : identification, observation, diagnostic, ...