Automatique Avancée – Commande non-linéaire

P2: Techniques de linéarisation exacte

Pascal Morin

pascal.morin@sorbonne-universite.fr

Sorbonne Université

Plan

Plan

Introduction

Linéarisation exacte par retour de sortie

Linéarisation exacte par retour de sortie

Limites de la technique

- Introduction
- Linéarisation exacte par retour de sortie
- Linéarisation exacte par retour d'état
- Limites de la technique

Introduction

Principe des techniques de linéarisation exacte:

- Transformer un système non-linéaire en un système linéaire, via un changement de variables
- Le changement de variable peut porter sur l'état et/ou la commande
- Avantages:
  - Dans les nouvelles variables, on peut utiliser les techniques linéaires classiques
  - On agrandit le domaine de stabilité (par rapport à une technique basée sur le linéarisé tangent)
  - On sait généralement spécifier le domaine de convergence
- Inconvénients (plus de détails plus tard):
  - Il faut avoir une bonne modélisation du système
  - La technique peut conduire à compenser des termes naturellement stables (inefficace et dangereux)

Exemple 1:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \sin(x_2) + x_1^2 + u \end{cases}$$

On définit une nouvelle "variable de commande"

$$v = \sin(x_2) + x_1^2 + u$$

Le système devient alors linéaire:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = v \end{cases}$$

Une commande stabilisante est définie par $v = -k_1 x_1 - k_2 x_2$, $k_1, k_2 > 0$. On en déduit l'expression finale de la commande:

$$u = -k_1 x_1 - k_2 x_2 - (\sin(x_2) + x_1^2)$$

Exemple 2:

$$\left\{ \begin{array}{lcl} \dot{x}_1 & = & x_1^2 + x_2 \\ \dot{x}_2 & = & u \end{array} \right.$$

On commence par définir un changement de variable d'état:

$$x \longmapsto z = \begin{pmatrix} x_1 \\ x_1^2 + x_2 \end{pmatrix}$$

Le système devient alors:

$$\left\{ \begin{array}{lcl} \dot{z}_1 & = & z_2 \\ \dot{z}_2 & = & u + 2z_1 z_2 \end{array} \right.$$

On définit une nouvelle variable de commande

$$v = u + 2z_1 z_2$$

Automatique Avancée – Commande non-linéaire

P2: Techniques de linéarisation exacte

Plan
Introduction
Linéarisation exacte par retour de sortie
Linéarisation exacte par retour de sortie
Limites de la technique

Exemple 2: (suite) Le système devient alors linéaire:

$$\begin{cases} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= v \end{cases}$$

Une commande stabilisante est définie par

$$v = -k_1 z_1 - k_2 z_2 \, , \; k_1, k_2 > 0$$

On en déduit l'expression finale de la commande:

$$u = -\left(k_1 x_1 + k_2(x_2 + x_1^2) + 2x_1(x_2 + x_1^2)\right)$$

Automatique Avancée – Commande non-linéaire

P2: Techniques de linéarisation exacte

Plan
Introduction
Linéarisation exacte par retour de sortie
Linéarisation exacte par retour de sortie
Limites de la technique

Remarques:

- Dans les exemples précédents, les changements de variables
$$\begin{pmatrix} x \\ u \end{pmatrix} \longmapsto \begin{pmatrix} z \\ v \end{pmatrix}$$
sont bien définis partout et définissent des différéomorphismes globaux.
- Dans de nombreux cas, le difféomorphisme n'est que local. Le domaine de définition du difféomorphisme définit alors le domaine dans lequel la linéarisation est possible.

Linéarisation exacte par retour de sortie

Objectif: On considère un système mono-entrée, affine en la commande, avec une sortie y:

$$\begin{cases} \dot{x} & = & X_0(x) + uX_1(x) \\ y & = & h(x) \end{cases} \tag{1}$$

On souhaite "linéariser" la dynamique de y, i.e., transfomer cette dynamique dans la forme

$$y^{(K)} = v$$

où K est un entier à définir, et v une nouvelle variable de commande.

Dans la suite, les champs de vecteurs X_0, X_1 (applications définies d'un ensemble ouvert de $\mathbb{R}^n$ dans $\mathbb{R}^n$) et la fonction h (définie d'un ensemble ouvert de $\mathbb{R}^n$ dans $\mathbb{R}$) sont supposées suffisamment réguliers pour que les différentiations à venir soient possibles.

Automatique Avancée – Commande non-linéaire

P2: Techniques de linéarisation exacte

Plan

Introduction

Linéarisation exacte par retour de sortie

Linéarisation exacte par retour de sortie

Limites de la technique

Théorie:

Definition

Etant donné une fonction régulière $f : \mathbb{R}^n \longrightarrow \mathbb{R}$ et un champ de vecteur régulier $X : \mathbb{R}^n \longrightarrow \mathbb{R}^n$, on appelle dérivée de Lie de f le long de X, la fonction $L_X f : x \longmapsto L_X f(x) := \frac{\partial f}{\partial x}(x).X(x)$. Par extension, on définit la notation suivante:

$$L_X^k f := L_X(L_X^{k-1} f), \quad k = 0, 1, \cdots, \text{ avec } L_X^0 f = f$$

Definition

On dit que le système (1) est de degré relatif $d(x_0)$ en un point x_0 s'il existe un voisinage $\mathcal{U}(x_0)$ tel que:

1. $L_{X_1} L_{X_0}^k h(x) = 0, \forall 0 \leq k < d(x_0) - 1, \forall x \in \mathcal{U}(x_0)$,
2. $L_{X_1} L_{X_0}^{d(x_0)-1} h(x_0) \neq 0$.

Automatique Avancée – Commande non-linéaire

P2: Techniques de linéarisation exacte

Plan

Introduction

Linéarisation exacte par retour de sortie

Linéarisation exacte par retour de sortie

Limites de la technique

Théorie:

Remarques:

- Le degré relatif n'est pas nécessairement défini partout,
- mais s'il est défini en un point x_0, alors il est défini dans un voisinage de ce point.
- Le degré relatif correspond au nombre de fois qu'il faut dériver la sortie y (par rapport au temps) pour que cette dérivée fasse explicitement apparaître l'entrée u.

Exemple:

$$\begin{cases} \dot{x} = u \\ y = x^3 \end{cases}$$

Linéarisation exacte par retour de sortie

Théorie:

Theorem

On suppose que le degré relatif $d(x_0)$ en x_0 est bien défini. Alors,

1. $d(x_0) \leq n$;
2. *Il existe $n - d(x_0)$ fonctions différentiables $\varphi_1, \cdots, \varphi_{n-d(x_0)} : \mathbb{R}^n \to \mathbb{R}$ telles que*

$$\Psi : x \longmapsto \Psi(x) := \begin{pmatrix} \varphi_1(x) \\ \vdots \\ \varphi_{n-d(x_0)}(x) \\ h(x) \\ L_{X_0} h(x) \\ \vdots \\ L_{X_0}^{d(x_0)-1} h(x) \end{pmatrix}$$

définisse un changement de coordonnées autour de x_0.

Linéarisation exacte par retour de sortie

Théorie:

Theorem (Suite)

De plus, on peut choisir $\varphi_1, \cdots, \varphi_{n-d(x_0)}$ telles que $L_g\varphi_k(x) = 0, \forall k = 1, \cdots, n - d(x_0), \forall x \in \mathcal{U}(x_0)$.

Supposons que $d(x_0)$ soit bien défini en x_0. Avec les notations ci-dessus, posons:

$$\xi = \begin{pmatrix} \varphi_1(x) \\ \vdots \\ \varphi_{n-d(x_0)}(x) \end{pmatrix}, \qquad z = \begin{pmatrix} h(x) \\ L_{X_0}h(x) \\ \vdots \\ L_{X_0}^{d(x_0)-1}h(x) \end{pmatrix} \qquad (2)$$

Le changement de coordonnées du Théorème s'écrit alors :

$$\Psi : x \longmapsto \Psi(x) := \begin{pmatrix} \xi \\ z \end{pmatrix}$$

Linéarisation exacte par retour de sortie

Théorie:

En utilisant la définition du degré relatif, on vérifie que les composantes de z satisfont la dynamique suivante:

$$\begin{cases} \dot{z}_1 & = & z_2 \\ \dot{z}_2 & = & z_3 \\ & \vdots & \\ \dot{z}_{d(x_0)-1} & = & z_{d(x_0)} \\ \dot{z}_{d(x_0)} & = & L_{X_0}^{d(x_0)} h(x) + u L_{X_1}(L_{X_0}^{d(x_0)-1} h)(x) \end{cases}$$

Par définition du degré relatif,

$$L_{X_1}(L_{X_0}^{d(x_0)-1} h)(x) \neq 0\,,\ \forall x \in \mathcal{U}(x_0)$$

Automatique Avancée – Commande non-linéaire

P2: Techniques de linéarisation exacte

Plan

Introduction

Linéarisation exacte par retour de sortie

Linéarisation exacte par retour de sortie

Limites de la technique

Théorie:

Par conséquent, si l'on pose

$$v := L_{X_0}^{d(x_0)} h(x) + u L_{X_1} (L_{X_0}^{d(x_0)-1} h)(x) \tag{3}$$

on a une relation bijective entre u et v pour tout $x \in \mathcal{U}(x_0)$.
Ceci définit un changement de variable de commande et donne:

$$z_1^{(d(x_0))} = z_d = y^{(d(x_0))} = v \tag{4}$$

La dynamique de y a été linéarisée!

Linéarisation exacte par retour de sortie

Théorie:

Par ailleurs, la dynamique de ξ est indépendante de u si l'on a choisi $\varphi_1, \cdots, \varphi_{n-d(x_0)}$ telles que (voir Théorème):

$$L_{X_1}\varphi_k(x) = 0, \forall k = 1, \cdots, n-d(x_0), \forall x \in \mathcal{U}(x_0)$$

C'est à dire que:

$$\dot{\xi}_k = L_{X_0}\varphi_k(x), \qquad \forall k = 1, \cdots, n-d(x_0) \tag{5}$$

Finalement, puisque Ψ est un changement de coordonnées,

$$\left\{ \begin{array}{lcl} \dot{\xi} & = & \gamma(\xi, z) \\ \dot{z}_k & = & z_{k+1} \\ \dot{z}_{(d(x_0))} & = & v \end{array} \right. \quad (k = 1, \cdots, d(x_0) - 1) \tag{6}$$

avec, d'après (5),

$$\gamma_k(\xi, z) = L_{X_0}\varphi_k(\Psi^{-1}(\xi, z))$$

Théorie:

Remarques:

- Puisque la dynamique de y a été linéarisée, i.e. $y^{(d(x_0))} = v$, on peut utiliser les outils d'automatique linéaire pour contrôler y. On peut alors résoudre le problème de stabilisation d'une trajectoire de référence y_r, ainsi que le problème de planification d'une trajectoire de référence (voir plus loin).
- Il convient de toujours faire attention au domaine de définition de la linéarisation (qui peut notamment amener des limitations sur y).
- Lorsque $d(x_0) = n$, on voit que la dynamique complète de l'état se trouve linéarisée.

Linéarisation exacte par retour de sortie

Théorie:

Remarques: (Suite)

- ξ n'est pas directement affectée par la commande (v ou u). En particulier, si l'objectif est de faire converger y vers zéro, ceci va impliquer la convergence de tout z vers zéro. A la limite, la dynamique de ξ devient donc

$$\dot{\xi} = \gamma(\xi, 0)$$

On appelle ceci la zéro dynamique (dynamique lorsque y est identiquement égale à zéro). En pratique, il est souvent souhaitable que cette dynamique soit asymptotiquement stable, ou au moins marginalement stable, afin d'éviter que cette partie de l'état non directement contrôlée ne diverge.

Application à la stabilisation de trajectoires:

On se limite au cas mono-entrée/mono-sortie mais l'extension au cas général ne pose pas de problème.

- Définissons $\tilde{y} = y - y_r$ avec y_r la trajectoire de référence
- Puisque $y^{(d(x_0))} = v$, on a $\tilde{y}^{(d(x_0))} = v - y_r^{(d(x_0))}$
- On pose alors:

$$v = y_r^{(d(x_0))} - k_0\tilde{y} - k_1\tilde{y}^{(1)} - \cdots - k_{d(x_0)-1}\tilde{y}^{(d(x_0)-1)}$$

La dynamique de $\tilde{y}$ est alors celle d'un système linéaire asymptotiquement stable dès lors que les gains k_i sont choisis tels que la matrice associée soit Hurwitz-Stable, i.e., tels que les racines de

$$P(\lambda) = \lambda^{d(x_0)} + k_{d(x_0)-1}\lambda^{d(x_0)-1} + \cdots + k_0$$

soient à partie réelle < 0.

Application à la génération de trajectoires:

La linéarisation de la dynamique de y va aussi permettre de générer des trajectoires de référence y_r. On considère le problème suivant:

Etant donné une condition initiale $y_0 = h(x_0)$ et une condition finale y_f pour la sortie y, déterminer une commande, définie sur un intervalle de temps $[0, T]$, telle que la sortie y_r obtenue après application de cette commande à partir de la condition initiale $y_r(0) = y_0$ satisfasse $y_r(T) = y_f$.

Ce problème, généralement difficile à résoudre dans les coordonnées de départ, va devenir assez simple après linéarisation.

Linéarisation exacte par retour de sortie

Application à la génération de trajectoires:

- On repart de la dynamique linéarisée $y^{(d(x_0))} = v$.
- On oublie dans un premier temps la dynamique de ξ (partie de l'état dont la dynamique n'a pas été linéarisée).
- Le problème consiste à trouver une fonction $y_r : [0, T] \longrightarrow \mathbb{R}$, de classe $\mathcal{C}^{d(x_0)}$ (i.e., différentiable et de dérivée continue jusqu'à l'ordre $d(x_0)$), telle que $y_r(0) = y_0$ et $y_r(T) = y_f$.
- On considère une famille "assez large" de fonctions paramétrées par un vecteur de coefficients constants $a_0, a_1, \cdots$. A titre d'exemple, on peut prendre la famille des fonctions polynomiales en temps:

$$g_a(t) = a_0 + a_1 t + a_2 \frac{t^2}{2} + \cdots + a_N \frac{t^N}{N!}$$

Linéarisation exacte par retour de sortie

Application à la génération de trajectoires:

- On pose $y_r(t) = g_a(t)$. Le problème se ramène donc à choisir $a_0, a_1, \cdots$ tels que $g_a(0) = y_0$ et $g_a(T) = y_f$.
- Mais il y a d'autres conditions sur a:
  1. Les dérivées en $t = 0$ de g_a jusqu'à l'ordre $d(x_0)$ doivent correspondre à celle de $h(x)$
  2. Il faut spécifier les dérivées en $t = T$ de g_a
  3. Il faut veiller à ce que la trajectoire générée reste dans le domaine de définition de la linéarisation

On considère dans la suite ces différents problèmes.

Linéarisation exacte par retour de sortie

Application à la génération de trajectoires:

1. Les dérivées en $t = 0$ de g_a jusqu'à l'ordre $d(x_0)$ doivent correspondre à celle de $h(x)$

D'après la définition du degré relatif, et puisque $h(x(t)) = z_1(t)$, on a (voir Slides 12 et 13):

$$\begin{cases} h(x(0)) & = & h(x_0) \\ \frac{d}{dt}_{|t=0} h(x(t)) & = & L_{X_0} h(x_0) \\ & \vdots & \\ \frac{d^{d(x_0)-1}}{dt^{d(x_0)-1}}_{|t=0} h(x(t)) & = & L_{X_0}^{d(x_0)-1} h(x_0) \\ \frac{d^{d(x_0)}}{dt^{d(x_0)}}_{|t=0} h(x(t)) & = & L_{X_0}^{d(x_0)} h(x_0) + u(0) L_{X_1}(L_{X_0}^{d(x_0)-1} h)(x_0) \end{cases}$$

Automatique Avancée – Commande non-linéaire

P2: Techniques de linéarisation exacte

Plan

Introduction

Linéarisation exacte par retour de sortie

Linéarisation exacte par retour de sortie

Limites de la technique

Application à la génération de trajectoires:

1. Les dérivées en $t = 0$ de g_a jusqu'à l'ordre $d(x_0)$ doivent correspondre à celle de $h(x)$

On peut, par exemple, imposer la condition $u(0) = 0$, ce qui donne les conditions suivantes sur g_a:

$$\left\{ \begin{array}{lclcl} g_a(0) & = & h(x(0)) & = & h(x_0) = y_0 \\ g_a^{(1)}(0) & = & \frac{d}{dt}_{|t=0} h(x(t)) & = & L_{X_0} h(x_0) \\ & \vdots & & \vdots & \\ g_a^{(d(x_0)-1)}(0) & = & \frac{d^{d(x_0)-1}}{dt^{d(x_0)-1}}_{|t=0} h(x(t)) & = & L_{X_0}^{d(x_0)-1} h(x_0) \\ g_a^{(d(x_0))}(0) & = & \frac{d^{d(x_0)}}{dt^{d(x_0)}}_{|t=0} h(x(t)) & = & L_{X_0}^{d(x_0)} h(x_0) \end{array} \right.$$

Linéarisation exacte par retour de sortie

Application à la génération de trajectoires:

1. Les dérivées en $t = 0$ de g_a jusqu'à l'ordre $d(x_0)$ doivent correspondre à celle de $h(x)$
2. Il faut spécifier les dérivées en $t = T$ de g_a:

- On a déjà $g_a(T) = y_f$
- Il n'y a pas de règle générale sur le choix des dérivées. Si l'objectif est de stabiliser y à la valeur y_f, on pourra poser

$$\forall k = 1, \cdots, d(x_0), \quad g_a^{(k)}(T) = 0$$

Linéarisation exacte par retour de sortie

Application à la génération de trajectoires:

1. Les dérivées en $t = 0$ de g_a jusqu'à l'ordre $d(x_0)$ doivent correspondre à celle de $h(x)$
2. Il faut spécifier les dérivées en $t = T$ de g_a
3. Il faut veiller à ce que la trajectoire générée reste dans le domaine de définition de la linéarisation

- Pas de stratégie générale pour aborder cette question, qui devra être traitée au cas par cas
- Si les restrictions sur le domaine de définition portent sur $h(x), L_{X_0}h(x), \cdots$, elles peuvent être prises en compte sous forme de contraintes sur la fonction g_a
- Sinon (contraintes sur ξ), il faudra étudier plus précisément la dynamique de ξ pour évaluer/résoudre le problème potentiel

Automatique Avancée – Commande non-linéaire

P2: Techniques de linéarisation exacte

Plan
Introduction
Linéarisation exacte par retour de sortie
Linéarisation exacte par retour de sortie
Limites de la technique

Application à la génération de trajectoires:

En résumé:

- On s'est ramené à la résolution d'un système d'équations linéaires en les paramètres $a_0, a_1, \cdots$
- Ce système contient $2(d(x_0)+1)$ contraintes
- Il faut donc au minimum $2(d(x_0)+1)$ paramètres pour qu'il exsite une solution
- On peut montrer (TD) que ce système admet alors bien une solution

Linéarisation exacte par retour de sortie

Extension au cas multi-entrées/multi-sorties:

Cette approche se généralise au cas des systèmes avec autant de sorties que d'entrées:

$$\begin{cases} \dot{x} & = & X_0(x) + \sum_{k=1}^{m} u_k X_k(x) \\ y & = & h(x) \end{cases} \tag{7}$$

avec $h : \mathbb{R}^n \to \mathbb{R}^m$.

Principe: l'idée consiste à définir une matrice de découplage qui va généraliser le terme $L_{X_1} L_{X_0}^{d(x_0)-1} h(x_0)$ qui intervient dans la définition du degré relatif.

Extension au cas multi-entrées/multi-sorties:

Definition

On dit que le système (7) est de degré relatif (vectoriel) $\{d_1(x_0), \cdots, d_m(x_0)\}$ en un point x_0 s'il existe un voisinage $\mathcal{U}(x_0)$ tel que:

1. $L_{X_j} L_{X_0}^k h_i(x) = 0, \forall j = 1, \cdots, m, \ \forall 0 \leq k < d_i(x_0) - 1, \forall x \in \mathcal{U}(x_0)$,
2. La matrice

$$A(x) = \begin{pmatrix} L_{X_1} L_{X_0}^{d_1(x_0)-1} h_1(x) & \cdots & L_{X_p} L_{X_0}^{d_1(x_0)-1} h_1(x) \\ \vdots & \vdots & \vdots \\ L_{X_1} L_{X_0}^{d_m(x_0)-1} h_m(x) & \cdots & L_{X_p} L_{X_0}^{d_m(x_0)-1} h_m(x) \end{pmatrix}$$

est inversible en x_0.

Linéarisation exacte par retour de sortie

Extension au cas multi-entrées/multi-sorties:

Remarques:

- Comme dans le cas mono-entrée, $d_i(x_0)$ $(i = 1, \cdots, m)$ correspond au nombre de fois qu'il faut dériver y_i par rapport au temps pour faire apparaître une entrée de commande.
- Par continuité, le fait que $A(x)$ soit inversible en x_0 implique qu'elle le sera dans un voisinage de x_0.

Linéarisation exacte par retour de sortie

Extension au cas multi-entrées/multi-sorties:

La matrice de découplage va permettre de généraliser les calculs des slides 13 et 14. En effet, on vérifie facilement que:

$$\begin{pmatrix} h_1^{(d_1(x_0))}(x(t)) \\ \vdots \\ h_m^{(d_m(x_0))}(x(t)) \end{pmatrix} = \begin{pmatrix} L_{X_0}^{(d_1(x_0))} h_1(x(t)) \\ \vdots \\ L_{X_0}^{(d_m(x_0))} h_m(x(t)) \end{pmatrix} + A(x(t))u(t)$$

On peut donc poser (comparer avec (3)):

$$v := \begin{pmatrix} L_{X_0}^{(d_1(x_0))} h_1(x) \\ \vdots \\ L_{X_0}^{(d_m(x_0))} h_m(x) \end{pmatrix} + A(x)u$$

qui définit un changement de variable de commande entre u et v.

Linéarisation exacte par retour de sortie

Extension au cas multi-entrées/multi-sorties:

Ce changement de variable de commande donne donc (comparer avec (4)):

$$\forall k = 1, \cdots, m\,, \quad y_k^{(d_k(x_0))} = v_k$$

- On a ainsi m systèmes linéaires indépendants, mono-entrée
- On peut aussi définir de nouvelles variables d'état (cf. (2)):

$$\xi = \begin{pmatrix} \varphi_1(x) \\ \vdots \\ \varphi_{\bar{n}}(x) \end{pmatrix}, \quad z^k = \begin{pmatrix} h_k(x) \\ L_{X_0} h_k(x) \\ \vdots \\ L_{X_0}^{d_k(x_0)-1} h_k(x) \end{pmatrix} \quad (k = 1, \cdots, m)$$

avec $\bar{n} = n - \sum_{k=1}^{m} d_k(x_0)$.

Linéarisation exacte par retour de sortie

Extension au cas multi-entrées/multi-sorties:

A partir de là,

Tout le reste fonctionne comme dans le cas mono-entrée/mono-sortie

Linéarisation exacte par retour d'état

Objectif: Transformer, autour d'un point d'équilibre, la dynamique d'un système non-linéaire

$$\dot{x} = X_0(x) + \sum_{k=1}^{m} u_k X_k(x) \tag{8}$$

en une dynamique de système linéaire, via:

- Un changement de variable d'état
- Et un changement de variable de commande
- A priori définis localement seulement

Linéarisation exacte par retour d'état

Definition

On considère le système (8) et l'on suppose que $X_0(x_0) = 0$ de sorte que $(x_0, u_0) = (x_0, 0)$ est un point d'équilibre du système. On dit que (8) est linéarisable par retour d'état (statique) au voisinage de x_0 s'il existe:

- Un voisinage $\mathcal{U}$ de x_0
- Un changement de variable d'état $x \longmapsto z = \varphi(x)$ avec $\varphi : \mathcal{U} \longrightarrow \mathbb{R}^n$ telle que $\varphi(x_0) = 0$,
- Un changement de variable de commande $u \longmapsto v = \Psi(x)u + \beta(x)$ avec $\beta : \mathcal{U} \longrightarrow \mathbb{R}^m$ et $\Psi : \mathbb{R}^n \longrightarrow \mathbb{R}^{m \times m}$ telle que $\Psi(x)$ soit inversible $\forall x \in \mathcal{U}$

tels que dans les coordonnées z, la dynamique du système soit linéaire:

$$\dot{z} = Az + Bv$$

Linéarisation exacte par retour d'état

Comparaison avec la linéarisation par retour de sortie:

- Propriété plus exigeante puisque toute la dynamique est linéarisée
- Linéarisable par retour de sortie et $\sum_{k=1}^{m} d_k(x) = n \implies$ Linéarisable par retour d'état
- Linéarisable par retour d'état et $m = p = 1 \implies \exists y$ telle que la dynamique de y soit linéarisable
- Dans le cas général, Linéarisable par retour d'état $\not\Longrightarrow$ $\exists y_1, \cdots, y_m$ tels que la dynamique de chaque y_k soit linéarisable et $\sum_{k=1}^{m} d_k(x) = n$

Linéarisation exacte par retour d'état

Propriétés:

- Il existe des CNS (Conditions Nécessaires et Suffisantes) portant sur les champs de vecteurs $X_0, \cdots, X_m$ pour décider de la possibilité de linéariser un système
- Ces conditions sont cependant complexes à vérifier dès que les dimensions (n ou m) augmentent
- Nous n'allons pas les introduire (on les trouve dans les ouvrages du domaine), mais traiter q.q. exemples

Exemple 1: Dynamique des robots manipulateurs

- Système mécanique composé de n corps rigides $C_1, \ldots, C_n$, avec chaque corps lié au précédent par une liaison à un degré de liberté
- On suppose que le corps C_1 est lié à un support fixe
- Chacune des n liaisons est actionnée par le biais d'un moteur délivrant une force ou un couple assimilable à une variable de commande.

Equations de Lagrange:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = u \tag{9}$$

avec...

Linéarisation exacte par retour d'état

Exemple 1: Dynamique des robots manipulateurs

- q_i la variable articulaire associée à la liaison i
- $M(q)$ une matrice définie positive (pour tout q) liée à l'énergie cinétique T du système par la relation $2T = \dot{q}^T M(q)\dot{q}$
- $C(q, \dot{q})$ une matrice associée aux forces de Coriolis et centrifuge
- $G(q)$ un vecteur lié aux forces de gravité
- $u \in \mathbb{R}^n$ le vecteur de commande

On pose

$$u = C(q, \dot{q})\dot{q} + G(q) + M(q)v$$

On obtient $\ddot{q} = v$, i.e., n double-intégrateurs indépendants.

Linéarisation exacte par retour d'état

Exemple 2: Dynamique angulaire d'un corps rigide

- Corps rigide en rotation dans l'espace
- Muni de trois actionneurs générant chacun un couple dans une direction fixe en repère corps

Equations d'Euler:

$$\mathbb{J}\dot{\omega} = -\omega \times \mathbb{J}\omega + \sum_{k=1}^{3} u_k b_k \tag{10}$$

avec

- $\mathbb{J}$: la matrice d'inertie
- ω: les composantes du vecteur vitesse angulaire exprimé en repère corps
- u_k: les intensités des couples de commande
- b_k: les directions des couples de commande exprimées en repère corps

Automatique Avancée – Commande non-linéaire

P2: Techniques de linéarisation exacte

Plan
Introduction
Linéarisation exacte par retour de sortie
Linéarisation exacte par retour de sortie
Limites de la technique

Exemple 2: Dynamique angulaire d'un corps rigide

- On peut ré-écrire cette équation comme suit:

$$\mathbb{J}\dot{\omega} = -\omega \times \mathbb{J}\omega + Bu$$

avec $B = (b_1 \ b_2 \ b_3)$

- Si les trois directions b_1, b_2, b_3 sont indépendantes, B est inversible. On peut alors poser:

$$-\omega \times \mathbb{J}\omega + Bu = v$$

- On obtient $\mathbb{J}\dot{\omega} = v$.

Limites de la technique

- Les techniques de linéarisation sont puissantes pour prendre en compte des linéarités
- Elles permettent d'obtenir des domaines de convergence plus grands/linéarisés tangents
- Mais elles doivent être appliquées avec précaution
- Exemple...

$$\dot{x} = -ax^3 + u\,, \quad (a > 0)$$

On souhaite stabiliser x à zéro. On considère la méthode du linéarisé tangent et la méthode par linéarisation exacte.

Limites de la technique

Stabilisation par le linéarisé tangent:

- Le linéarisé tangent est donné par $\dot{x} = u$
- On pose $u = -kx$ $(k > 0)$
- On obtient en boucle fermée:

$$\dot{x} = -kx - ax^3 = -(k + ax^2)x$$

- L'équilibre $x = 0$ est globalement asymptotiquement stable, ceci $\forall a$.
- En effet, $\dot{x} > 0$ si $x < 0$ et $\dot{x} < 0$ si $x > 0$, et donc $|x(t)|$ ne fait que décroitre et ne peut que converger vers 0.

Limites de la technique

Stabilisation par linéarisation exacte:

- On pose $v = -ax^3 + u \Longrightarrow \dot{x} = v$
- On pose $v = -kx$ $(k > 0)$
- Ce feedback, appliqué au système de départ rend $x = 0$ globalement asymptotiquement stable
- Supposons maintenant que a est mal connu. On ne connait qu'une approximation $\hat{a}$
- On pose donc $v = -\hat{a}x^3 + u \Longrightarrow \dot{x} = (\hat{a} - a)x^3 + v$
- Avec $v = -kx$ on obtient

$$\dot{x} = -kx + (\hat{a} - a)x^3$$

Si $\hat{a} > a$, $x = 0$ n'est pas globalement asymptotiquement stable. En effet, on remarque que le système admet alors trois points d'équilibre!

Limites de la technique

Morale:

- Il n'est pas toujours nécessaire, ni opportun, de linéariser complètement le système
- En particulier, il peut être préférable de ne pas compenser des termes qui contribuent à la stabilité:
  - D'une part parce que les compenser peut nuire à la stabilité (voir exemple)
  - D'autre part parce que cela induira un coût énergétique importante (grandes valeurs de u)