

MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE.

UNIVERSITÉ MOULOUD MAMMERI, TIZI- OUZOU



FACULTÉ DES SCIENCES  
DÉPARTEMENT DE MATHÉMATIQUES

## **Mémoire de MASTER II**

SPÉCIALITÉ : MATHÉMATIQUES

OPTION : MODÉLISATION MATHÉMATIQUE

Présenté par :

**Mr Hamza SADAoui**

sujet :

---

**Réduction d'une classe de systèmes non linéaires à temps discret**

---

Devant le jury d'examen composé de :

Mr. Mohamed Morsli ;	Professeur ;	U.M.M.T.O ;	Président
Mme. Fazia Bedouhene ;	Professeur ;	U.M.M.T.O ;	Rapporteur
Mlle Smaali Mannal ;	MCA ;	U.M.M.T.O ;	Examineur
Mme Achemine Farida ;	MCB ;	U.M.M.T.O ;	Examineur

soutenu : le 30/09/2015

## Remerciements

*Au premier lieu, mes vifs remerciements vont tout d'abord à Mme. **Fazia Bedouhene** professeur de l'**UMMTO** pour avoir accepté de diriger ce modeste travail, et pour son aide, sa disponibilité, ses orientations, ses conseils et sa volonté qu'elle a manifesté tout au long de l'élaboration du présent mémoire.*

*Mes sincères remerciements et ma profonde gratitude s'adressent également aux membres de jury pour leur disponibilité et qui me feront l'honneur de juger mon travail*

*Je tient à remercier tous les enseignants du département Mathématiques qui ont contribué à ma formation.*

*Enfin, je ne peux conclure sans adresser mes plus chaleureux remerciements à tous les membres de ma famille qui m'ont toujours soutenue. Il me faut également remercier de tout mon coeur mes chers amis qui n'ont pas cessé de m'encourager dans les moments difficiles.*

*Merci à tous*

## *Dédicace*

*Je dédie ce travail.*

*A toute ma famille.*

*A mes amis.*

*A tous ceux que j'aime.*

# Table des matières

Notations et acronymes	3
Introduction générale	4
<b>1 Résultats préliminaires</b>	<b>7</b>
1.1 Généralités sur les systèmes dynamiques . . . . .	7
1.1.1 Équations différentielles ordinaires . . . . .	7
1.1.2 Stabilité des équilibres . . . . .	8
1.1.3 Analyse de la stabilité au sens de Lyapunov (cas discret) . . . . .	9
1.1.4 Discrétisation d'une équation différentielle . . . . .	11
1.2 L'approche des inégalités matricielles linéaires . . . . .	12
1.2.1 quelques rappels sur les LMIs . . . . .	12
1.2.2 Lemmes utiles pour les LMIs . . . . .	13
1.3 La réduction des systèmes dynamiques . . . . .	15
1.3.1 Problématique générale . . . . .	15
1.3.2 Méthodes de projection sur des sous-espaces de Krylov . . . . .	17
1.3.3 Méthode d'Arnoldi . . . . .	18
1.3.4 Construction de la base d'Arnoldi . . . . .	18
<b>2 La réduction des systèmes non-linéaire à temps discret</b>	<b>21</b>
2.1 Conditions suffisantes pour l'approximation d'un système non-linéaire par un système linéaire d'ordre réduit . . . . .	21
2.1.1 Formulation du problème d'optimisation . . . . .	22
2.1.2 Linéarisation du problème . . . . .	25
2.2 Approximation par un système non linéaire d'ordre réduit . . . . .	28
2.3 Exemple d'application . . . . .	32
2.3.1 Système dynamique SMD (Spring, Mass, Dimping) . . . . .	32
<b>Conclusion</b>	<b>47</b>
<b>Bibliographie</b>	<b>48</b>

# Notations et acronymes

## 1. *Acronymes* :

- LTI : Linéaire Temps Invariant (Linear Time Invariant).
- LPTV : Linéaire à Paramètres Temps Variants (Linear Parameter Time Varying).
- LMI : Inégalités Matricielles Linéaires (Linear Matrix Inequality).
- BMI : Inégalités matricielles bilinéaires (Bilinear matrix inequality).
- GAS : Globalement Asymptotiquement Stable.
- SDP : Semi Définie Positive.
- SVD : Décomposition aux valeurs singulières (Singular Value Décomposition).
- GSM : Gram-Schmidt Modifié (Gram-Schmidt Modified).

## 2. *Notations* :

- $\mathbb{R}$  est l'ensemble des réels ;
- $\mathbb{R}^n$  est l'espace vectoriel de dimension  $n$  construit sur le corps des réels ;
- $(\star)$  est utilisée pour les blocs induits par symétrie dans une matrice ou dans une inégalité matricielle ;
- $A'$  représente la matrice transposée de  $A$  ;
- $\mathbb{R}^{n \times m}$  est l'ensemble de toutes les matrices réelles de  $n$  lignes et  $m$  colonnes ;
- $I$  est une matrice d'identité de dimension appropriée et  $I_r$  représente une matrice d'identité de dimension  $r$  ;
- La valeur  $(0)$  représente une matrice nulle de dimension appropriée, et  $(0)_{n,m}$  désigne matrice nulle à  $n$  lignes et  $m$  colonnes ;
- pour une matrice carrée  $S$ ,  $S > 0$  ( $S < 0$ ) signifie que cette matrice est définie positive (définie négative) ;
- $\|\cdot\|$  est la norme Euclidienne habituelle ;
- $\text{diag}(A_1, \dots, A_i)$  est la matrice diagonale par blocs ayant  $A_1, \dots, A_i$  sur sa diagonale principale.
- $e_s(i) = \underbrace{(0, \dots, \overset{\text{ith}}{1}, \dots, 0)}_{s \text{ composantes}} \in \mathbb{R}^s, s \geq 1$  est un vecteur de la base canonique de  $\mathbb{R}^s$ .

# Introduction générale

## Motivation

L'amélioration des techniques de modélisation et l'utilisation de logiciels de conception rendent les modèles dynamiques de plus en plus complexes. Si d'un point de vue théorique cela peut sembler intéressant, en pratique en revanche, les modèles de grande dimension rendent la simulation, l'analyse de performances et la synthèse de contrôleurs problématiques. C'est notamment le cas en aéronautique où les modèles fournis par les partenaires industriels peuvent difficilement être utilisés en l'état. Il est nécessaire d'en réduire la taille afin de pouvoir appliquer des techniques d'analyse et de synthèse classiques basées sur l'optimisation.

## Position du problème de réduction

En fonction de la classe de systèmes considérée, de nombreuses techniques de réduction existent. Pour les systèmes linéaires invariant dans le temps et stables, le problème est le suivant :

**Problème 1.** *Étant donné le système dynamique ( $\Sigma$ ) d'ordre  $n$  ayant  $n_u$  entrées et  $n_y$  sorties, représenté dans le domaine temporel par*

$$\Sigma := \begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) \end{cases} \quad (0.0.1)$$

*ou dans le domaine fréquentiel par sa fonction de transfert*

$$H(s) := C(sI_n - A)^{-1}B \in \mathbb{C}^{n_y \times n_u}. \quad (0.0.2)$$

*Le problème de réduction consiste à trouver les matrices*

$$\hat{A} \in \mathbb{R}^{r \times r}, \hat{B} \in \mathbb{R}^{r \times n_u}, \hat{C} \in \mathbb{R}^{n_y \times r} \quad (r \ll n)$$

*telles que le système réduit  $\hat{\Sigma}$  :*

$$\hat{\Sigma} := \begin{cases} \dot{\hat{x}}(t) = \hat{A}\hat{x}(t) + \hat{B}u(t) \\ \hat{y}(t) = \hat{C}\hat{x}(t) \end{cases} \quad (0.0.3)$$

*dont la fonction de transfert est :*

$$\hat{H}(s) := \hat{C}(sI_r - \hat{A})^{-1}\hat{B} \in \mathbb{C}^{n_y \times n_u} \quad (0.0.4)$$

*ait un comportement entrées/sorties proche de celui du système original  $\Sigma$ .*

Les méthodes de réduction des modèles de grande dimension se divisent principalement en deux familles, les méthodes qui n'utilisent pas la projection et celles qui l'utilisent. Seule la seconde famille de méthodes est considérée ici car elle répond mieux aux exigences numériques liées aux systèmes de grande dimension. Le problème de réduction par projection s'exprime comme suit.

**Problème 2.** *Étant donné le système  $\Sigma$  à réduire, il s'agit de trouver deux matrices de projection  $V, W \in \mathbb{R}^{n \times r}$  ( $r \ll n$ ) bi-orthogonales ( $W'V = I_r$ ) telles que les matrices du modèle réduit  $\hat{\Sigma}$  s'écrivent*

$$\hat{A} = W'AV, \hat{B} = W'B, \hat{C} = CV. \quad (0.0.5)$$

Pour juger de la qualité de l'approximant dans le Problème 1, le critère  $\mathcal{H}_\infty$  est considéré :

$$\|H(s) - \hat{H}(s)\|_{\mathcal{H}_\infty}^2$$

où, la norme  $\mathcal{H}_\infty$  n'est autre que la norme infinie  $\|\cdot\|_\infty$  de la matrice de transfert  $H$ . Si on connaît l'expression de la valeur singulière de la matrice de transfert, on a

$$\|H\|_\infty = \sup_w \|H(iw)\|_2 = \sup_w \bar{\sigma}(H(iw)), \quad (0.0.6)$$

Le problème suivant apparaît alors naturellement.

**Problème 3.** *Le problème de minimisation de la norme  $H_\infty$  de l'erreur consiste à trouver le modèle réduit  $\hat{\Sigma}$  de fonction de transfert  $\hat{H}(s)$  tel que*

$$\hat{H}(s) = \arg \left( \min_{\hat{H}(s) \text{ stable}} \|H(s) - \hat{H}(s)\|_{H_\infty}^2 \right)$$

Ce problème n'étant pas convexe, le minimum global ne peut pas être déterminé facilement. A l'échelle temporelle, ce problème peut être reformulé comme suit (voir par exemple [20]) :

Minimiser le paramètre  $\gamma > 0$  sous la contrainte

$$\sum_{k=0}^{N-1} \|\hat{y}_k - y_k\|^2 < \gamma^2 \sum_{k=0}^{N-1} \|u_k\|^2. \quad (0.0.7)$$

$\forall u_k \in \mathbb{R}^n, \forall k \in [0, N-1]$ , et pour toute condition initiale  $x_0 = 0$  et  $\hat{x}_0 = 0$

Il convient de préciser que cette reformulation nécessite des arguments d'analyse fonctionnelle (Analyse de Fourier) très laborieux que ne nous présentons pas dans le cadre de ce mémoire. Ce mémoire a pour objectif d'étudier et développer l'approche de S. Ibrir et M. Bettayeb pour la réduction de modèle pour une classe de systèmes non linéaires discrets Lipschitziens. Cette approche, consignée dans la référence [1], est basée sur la méthode de projection de Krylov et quelques techniques d'optimisation convexe. Des conditions suffisantes garantissant l'existence d'un modèle linéaire et non linéaire d'ordre réduit sont données. Ces dernières sont exprimées sous forme d'un problème d'optimisation convexe sous contraintes inégalités linéaires matricielles (LMIs).

Le mémoire que nous présentons est rédigé comme suit :

Le premier chapitre est consacré à la présentation de quelques notions mathématiques utiles pour la compréhension de ce mémoire. Un bref rappel sur la stabilité des systèmes dynamiques est donné. Les théorèmes issus des techniques de Lyapunov conduisent en général à des inégalités

linéaires matricielle (LMI), nous consacrons une section pour une représentation du problème des LMIs. Nous achevons ce chapitre par la présentation du problème de réduction des systèmes dynamiques linéaires.

Le deuxième chapitre est l'essentiel de notre travail. Nous y avons développé la méthode de réduction proposée par Ibrir et Bettayeb pour la classe suivante de systèmes non linéaires :

$$\begin{cases} x_{k+1} = Ax_k + f(x_k) + Bu_k, \\ y_k = Cx_k, \end{cases} \quad (0.0.8)$$

où  $A, B$  et  $C$  sont des matrices connues de dimensions appropriées et telles que  $(A, B)$  et  $(A, C)$  sont respectivement stabilisable et détectable. La non-linéarité  $f$  est supposée être globalement lipschitzienne. Ce chapitre est composé de trois parties, la première est dédiée l'approximation du système (0.0.8) par un système linéaire d'ordre réduit de la forme

$$\begin{cases} \zeta_{k+1} = G_2^{-1}X\zeta_k + G_2^{-1}Yu_k, \\ z_k = C_r\zeta_k, \end{cases} \quad (0.0.9)$$

Grâce au théorème de la valeur moyenne, la théorie de Lyapunov et quelques techniques de linéarisation des inégalités matricielles, des conditions suffisantes sont données afin de résoudre le problème 3. ci-dessus. Le calcul des paramètres du système réduit (0.0.9) repose uniquement sur les techniques de linéarisation des systèmes non-linéaires (et n'évoque donc pas les projections de Krylov).

On présentera ensuite dans la deuxième partie la méthode basée sur les projections de Krylov et l'optimisation convexe pour l'approximation du système (0.0.8) par un système non-linéaire d'ordre réduit de la forme

$$\begin{cases} \zeta_{k+1} = A_r\zeta_k + B_f f(W_r\zeta_k) + B_r u_k, \\ z_k = C_r\zeta_k, \end{cases} \quad (0.0.10)$$

Enfin dans la troisième partie, nous présenterons les résultats de simulation que nous avons obtenus lors de l'application des deux approches précédentes sur l'exemple du système dynamique masse-ressort-amortisseur (SMD, Spring, Mass, Dimping), soumis à un frottement non linéaire. Dans chaque cas, nous y avons calculé les paramètres du systèmes réduit ainsi que la marge d'erreur tolérée. Comme langage de programmation, nous avons opté pour Matlab. Pour la programmation des conditions LMI données, nous avons utilisé le package Yalmip.

Ce mémoire s'achève par une conclusion générale synthétisant l'ensemble du travail effectué et dégageant des perspectives de recherche.

# Chapitre 1

## Résultats préliminaires

### 1.1 Généralités sur les systèmes dynamiques

Les systèmes dynamiques sont les notions mathématiques qui permettent de modéliser des phénomènes évoluant dans le temps, ces phénomènes pouvant provenir de la physique, la mécanique, l'économie, la biologie, l'écologie, la chimie... Un système dynamique est constitué d'un espace de phases, l'espace des états possibles du phénomène convenablement paramétré, muni d'une loi d'évolution qui décrit la variation temporelle de l'état du système. Dans le cadre choisi ici, celui de lois déterministes en temps continu, cette loi d'évolution prend la forme d'une équation différentielle.

#### 1.1.1 Équations différentielles ordinaires

Soit

$$f : \Omega \rightarrow \mathbb{R}^n, x \rightarrow f(x)$$

une application de classe  $C^k$ ,  $k \geq 1$ , d'un ouvert  $\Omega \subseteq \mathbb{R}^n$  dans  $\mathbb{R}^n$ . Nous écrivons,

$$x = (x_1, x_2, \dots, x_n), f = (f_1, f_2, \dots, f_n).$$

On appelle **équation différentielle vectorielle autonome** du premier ordre une équation du type

$$\dot{x} = f(x), \tag{1.1.1}$$

Le qualificatif autonome se rapporte au fait que le second membre dans l'équation (1.1.1) ne dépend pas explicitement du temps.

On appelle solution de l'équation (1.1.1) toute application dérivable  $x : I \rightarrow \mathbb{R}^n$ , définie sur un intervalle non vide  $I \subseteq \mathbb{R}$  et telle que, pour tout  $t \in I$ ,  $x(t) \in \Omega$  et  $\dot{x}(t) = f(x(t))$ .

**Théorème 1.** *Supposons que  $f$  soit de classe  $C^1$  sur  $\Omega$ . Alors, pour tout nombre réel  $t_0$  et tout vecteur  $x_0 \in \Omega$ , il existe un intervalle ouvert  $I$  contenant  $t_0$ , sur lequel il existe une solution de l'équation (1.1.1) qui satisfait à la condition initiale  $x(t_0) = x_0$  et cette solution est unique.*

Le problème de trouver une solution de l'équation (1.1.1) satisfaisant à la condition initiale  $x(t_0) = x_0$ , est appelé problème de Cauchy.

## Problème de Cauchy

Soit  $I$  un intervalle de  $\mathbb{R}$  et  $U$  un ouvert de  $\mathbb{R}^n$ . Considérons le problème de Cauchy

$$\begin{cases} \dot{x}(t) = f(t, x(t)) \\ x(t_0) = x_0 \end{cases} \quad (1.1.2)$$

où  $f$  est une application de  $I \times U$  dans  $\mathbb{R}^n$ , et  $x_0 \in U$ . Le théorème de Cauchy- Lipschitz usuel affirme l'existence et l'unicité d'une solution maximale pourvu que  $f$  soit continue, et localement lipschitzienne par rapport à sa deuxième composante.

**Définition 1.** L'application  $f$  de l'ouvert  $U$  de  $\mathbb{R} \times \mathbb{R}^n$  dans  $\mathbb{R}^n$  est Lipschitzienne par rapport à  $x$  sur  $U$  si :

$$\exists \gamma \in \mathbb{R}^+, \quad \forall (t, x_1) \in U, (t, x_2) \in U, \quad \text{alors} \quad \| f(t, x_2) - f(t, x_1) \| \leq \gamma \| x_2 - x_1 \| .$$

$\gamma$  est appelé la constante de Lipschitz. Noter qu'elle ne dépend ni de  $x_1$ , ni de  $x_2$ .

**Définition 2.** L'application  $f$  de l'ouvert  $U$  de  $\mathbb{R} \times \mathbb{R}^n$  dans  $\mathbb{R}^n$  est localement Lipschitzienne par rapport à  $x$  sur  $U$ , si  $\forall (t, x) \in U$ , on peut trouver un voisinage ouvert de  $(t, x)$  dans  $U$ , dans lequel  $f$  est Lipschitzienne.

Un résultat très important est que si  $f$  est de classe  $C^1$  sur  $U$ , alors elle est localement lipschitzienne.

En particulier en dimension 1, la constante  $\gamma$  est égale à

$$\sup_{(t,x) \in U} \left| \frac{\partial f}{\partial x}(t, x) \right|.$$

### 1.1.2 Stabilité des équilibres

La notion de stabilité d'un système dynamique caractérise le comportement de ses trajectoires autour des points d'équilibre. L'analyse de la stabilité d'un système dynamique permet donc d'étudier l'évolution de sa trajectoire lorsque l'état initial est proche d'un point d'équilibre.

Considérons l'équation différentielle autonome

$$\dot{x}(t) = f(x(t)), \quad (1.1.3)$$

où le champs de vecteurs  $f : \Omega \subset \mathbb{R}^n \rightarrow \mathbb{R}^n$  est supposé de classe  $C^1$ . Les points d'équilibre de (1.1.3) sont les solutions de l'équation  $f(x) = 0$ .

**Définition 3.** On dit qu'un point  $x^* \in \Omega$  est un équilibre de (1.1.3) si la fonction constante  $x(\cdot) \equiv x^*$  est solution de (1.1.3) ou, de façon équivalente, si  $f(x^*) = 0$ .

Quand l'équation (1.1.3) modélise l'évolution d'un phénomène physique (mécanique, biologique, . . . ), un équilibre correspond bien à la notion habituelle « d'état d'équilibre » : si le système est dans l'état  $x^*$ , alors il y reste. En pratique on sait cependant que seuls les états d'équilibre ayant certaines propriétés de stabilité sont significatifs.

**Définition 4.** Un point d'équilibre  $x^*$  du système (1.1.3) est dit

– **stable** si  $\forall \epsilon > 0 \exists \delta > 0$  tel que

$$\|x(0) - x^*\| < \delta \Rightarrow \|x(t) - x^*\| < \epsilon, \forall t \geq 0,$$

– **asymptotiquement stable** si  $x^*$  est stable et si il existe  $\delta > 0$  tel que

$$\|x(0) - x^*\| < \delta \Rightarrow \lim_{t \rightarrow \infty} x(t) = x^*,$$

– **globalement asymptotiquement stable** si  $x^*$  est stable et  $\forall x(0) \in \mathbb{R}^n$

$$\lim_{t \rightarrow \infty} x(t) = x^*.$$

– **localement exponentiellement stable** s'il existe trois nombres réels positifs  $c$ ,  $K$  et  $\lambda$  tel que

$$\forall \|x(0) - x^*\| < c \Rightarrow \|x(t) - x^*\| \leq K \|x(0) - x^*\| e^{-\lambda t},$$

– **instable** s'il n'est pas stable.

### 1.1.3 Analyse de la stabilité au sens de Lyapunov (cas discret)

La définition de la stabilité présente certains désavantages importants :

- Il est nécessaire de pouvoir calculer de manière explicite chaque solution correspondant à chacune des conditions initiales.
- Le maniement de la définition est fastidieux.

Par conséquent, des résultats permettant de déterminer la stabilité sans devoir intégrer les équations différentielles seraient les bienvenues.

Le comportement stable ou instable d'un système est relié à la fois à la caractéristique et à l'évolution de sa fonction d'énergie, la présence d'un maximum ou minimum d'énergie possède une influence critique. De plus la présence des perturbation est responsable de la croissance d'énergie complète, et influence donc la stabilité. Le comportement est stable lorsque :

1. Énergie  $E$  diminue et  $E$  est minimum au point d'équilibre.
2. Énergie  $E$  est conservée et  $E$  est minimum à l'équilibre.

Par contre, le comportement est instable lorsque :

1. L'énergie augmente.
2. L'énergie  $E$  est conservé, mais elle ne correspond pas à un minimum à l'équilibre.

La théorie de Lyapunov est fondée sur l'extension de ses concepts, la procédure de base est de générer une fonction scalaire "de type énergie" pour le système dynamique.

**Définition 5.** Une fonction de Lyapunov est une fonction continue  $V : \mathbb{R}^n \rightarrow \mathbb{R}$  telle que :

$$V(x) > 0 \quad \forall x \neq 0 \quad \text{et} \quad V(x) = 0 \quad \text{si} \quad x = 0,$$

Si l'inégalité précédente est vérifiée au sens large, alors la fonction est dite semi-définie positive. Notons qu'une classe de fonctions souvent utilisées pour l'analyse des systèmes est celle des fonctions quadratiques  $V(x) = x'Px$  où  $P$  est une matrice symétrique réelle. Une telle fonction  $V(x) = x'Px$  est dite définie positive (respectivement semi-définie positive) si toutes les valeurs propres de la matrice  $P$  sont strictement positives (respectivement positives).

Dans notre cas, nous nous focaliserons sur la stabilité au sens de Lyapunov pour les systèmes à temps discret.

Considérons maintenant un système non linéaire à temps discret, s'écrivant

$$x(k+1) = f(x(k)), \quad (1.1.4)$$

et notons  $x^* = 0$  son point d'équilibre.

Nous disposons du résultat suivant qui donne des conditions suffisantes pour la stabilité du système (1.1.4) via la théorie de Lyapunov

**Théorème 2.** *L'équilibre  $x^* = 0$  est*

- **localement stable** s'il existe une fonction  $V : \mathbb{R}^n \rightarrow \mathbb{R}$  et un voisinage  $\Omega \in \mathbb{R}^n$  de l'origine tels que :

$$V(x) > 0, \forall x \in \Omega \text{ et } V(0) = 0,$$

$$\Delta V(x(k)) \leq 0, \forall x \in \Omega, x \neq 0,$$

avec  $\Delta V(x(k)) = V(x(k+1)) - V(x(k))$ .

- **localement asymptotiquement stable** s'il existe une fonction  $V : \mathbb{R}^n \rightarrow \mathbb{R}$  et un voisinage  $\Omega \in \mathbb{R}^n$  de l'origine tels que :

$$V(x) > 0, \forall x \in \Omega \text{ et } V(0) = 0,$$

$$\Delta V(x(k)) < 0, \forall x \in \Omega, x \neq 0,$$

- **globalement asymptotiquement stable** s'il existe une fonction  $V : \mathbb{R}^n \rightarrow \mathbb{R}$  telle que :

$$V(x) > 0, \forall x \in \mathbb{R}^n \text{ et } V(0) = 0,$$

$$\Delta V(x(k)) < 0, \forall x \in \mathbb{R}^n, x \neq 0,$$

$$V(x) \rightarrow \infty \text{ lorsque } \|x\| \rightarrow \infty,$$

La stabilité asymptotique des systèmes LTI à temps discret en terme de fonction de Lyapunov se caractérise comme suit :

**Théorème 3.** *Le point  $x^* = 0$  est asymptotiquement stable si et seulement si, pour tout  $Q = Q' > 0$ , il existe une matrice  $P = P' > 0$  vérifiant l'équation de Lyapunov*

$$A'PA - P + Q = 0.$$

Il convient de rappeler le critère de stabilité asymptotique du système linéaire (1.1.4).

**Théorème 4.** *Un système discret est asymptotiquement stable si les valeurs propres de sa matrice  $A$  ont leur module strictement inférieur à 1.*

### 1.1.4 Discrétisation d'une équation différentielle

Le principe de la discrétisation (Méthodes de type Euler) est de remplacer un phénomène continu par un autre, discret. On présente une des méthodes les plus anciennes de résolution d'une EDO du premier ordre. Il existe deux types de méthode d'Euler : explicite et implicite et dans les deux cas on cherche à résoudre des équations du type :

$$\frac{\partial x}{\partial t} = f(x, t) \quad (1.1.5)$$

où  $f$  est une fonction connue de  $x$  et  $t$ .

#### Méthode d'Euler explicite

Dans la méthode d'Euler explicite, (1.1.5) est discrétisée comme suit :

$$\begin{cases} \frac{\partial x}{\partial t} \approx \frac{x_{k+1} - x_k}{\Delta t} \\ f(x, t) \approx f(x_k, t_k) \end{cases}, \quad (1.1.6)$$

où  $x_k$  est la valeur de  $x$  à la date  $t_k$ ,  $x_{k+1}$  est la valeur  $x$  à la date  $t_{k+1}$ ,  $\Delta t = t_{k+1} - t_k$  est le pas de temps.

En remplaçant (1.1.6) dans (1.1.5), il vient :

$$\frac{x_{k+1} - x_k}{\Delta t} = f(x_k, t_k). \quad (1.1.7)$$

Cette équation s'écrit :

$$x_{k+1} = x_k + f(x_k, t_k)\Delta t \quad (1.1.8)$$

Puisque  $x_k$  est connue et que la fonction  $f$  l'est aussi,  $f(x_k, t_k)$  l'est également.  $x_{k+1}$  peut être déterminée directement d'après  $x_k$ . La méthode numérique est dite explicite car la valeur de  $x$  à la date  $k + 1$  peut être déterminée explicitement à partir de la valeur de  $x$  à la date  $k$ .

#### Méthode d'Euler implicite

Dans la méthode d'Euler explicite, (1.1.5) est discrétisée comme suit :

$$\begin{cases} \frac{\partial x}{\partial t} \approx \frac{x_{k+1} - x_k}{\Delta t} \\ f(x, t) \approx f(x_{k+1}, t_{k+1}) \end{cases} \quad (1.1.9)$$

A la différence de la méthode explicite,  $f$  est maintenant calculée à partir de la valeur  $x$  à la date  $t_{k+1}$ . En remplaçant (1.1.9) dans (1.1.5), il vient :

$$\frac{x_{k+1} - x_k}{\Delta t} = f(x_{k+1}, t_{k+1}). \quad (1.1.10)$$

Cette équation s'écrit :

$$x_{k+1} = x_k + f(x_{k+1}, t_{k+1}) \quad (1.1.11)$$

Contrairement au cas explicite, cette formulation lie la valeur de  $x_{k+1}$  à une fonction de cette valeur même. En général la résolution de (1.1.11) impose d'avoir recours à des méthodes itératives (de type Newton).

## 1.2 L'approche des inégalités matricielles linéaires

Une grande quantité de problèmes d'automatique concernant les performances et la robustesse peuvent se traduire sous la forme d'une optimisation convexe avec des contraintes inégalités. Auparavant, on faisait appel à la résolution d'équations de Riccati basées sur des contraintes égalités. De nombreux travaux ont montré qu'un grand nombre de problèmes qui apparaissaient difficiles à résoudre de manière analytique, pouvaient se formuler et se résoudre par l'approche LMI. [2, 19, 29, 31]

### 1.2.1 quelques rappels sur les LMIs

**Définition 6** (Linear Matrix Inequality). *Une contrainte LMI est une contrainte sur un vecteur réel  $x \in \mathbb{R}^m$  de la forme*

$$F(x) = F_0 + \sum_{i=1}^m x_i F_i \geq 0, \quad (1.2.1)$$

où les matrices symétriques  $F_i = F_i^T \in \mathbb{R}^{n \times n}$  sont données, et le symbole d'inégalité au sens large (rep. strict) signifie que  $F$  est semi-définie (resp. définie) positive, c'est à dire que  $u^T F u \geq 0$  pour tout  $u \in \mathbb{R}^n$ .

**Définition 7** (Bilinear Matrix Inequality). *une contrainte BMI est une contrainte sur  $x \in \mathbb{R}^m$  et  $y \in \mathbb{R}^r$  qui est de la forme*

$$F(x, y) := F_{0,0} + \sum_{i=1}^m \sum_{j=1}^r x_i y_j F_{i,j} \geq 0, \quad (1.2.2)$$

avec  $F_{0,0} \in \mathbb{R}^{n \times n}$  et  $F_{i,j} \in \mathbb{R}^{n \times n}$ .

### Problèmes LMI de programmation semi-définie

Un problème de Programmation Semi-Définie (SDP) est un problème d'optimisation qui s'écrit

$$\begin{cases} \text{minimiser} & c^T x, \\ \text{sous la contrainte} & F(x) \succ 0 \end{cases} \quad (1.2.3)$$

où  $c \in \mathbb{R}^m$  est un vecteur définissant l'objectif du problème. Si une formulation SDP est obtenue pour un problème donné, alors nous pouvons considérer ce problème comme "résolu" si on démontre qu'il est faisable.

**Exemple 1.** [30] SDP pour les LMIs

Soit le système (1.3.1), sa fonction de transfert est  $H(s) = C(sI - A)^{-1}B$ . Minimiser  $\gamma$ , avec,  $\|H(s)\|_\infty < \gamma$  est un problème SDP. En effet, cela revient à écrire

$$\left\{ \begin{array}{l} \text{minimiser} \\ \text{sous la contrainte} \end{array} \right. \begin{array}{l} \gamma, \\ \left[ \begin{array}{cc} -(AX' + XA + CC') & XB \\ \star & -\gamma^2 I \end{array} \right] > 0, \\ X > 0. \end{array} \quad (1.2.4)$$

Il s'agit de la généralisation de la notion de LMI. Une grande variété de problèmes d'automatique peuvent se formuler comme des BMI. Cependant, les BMI ne sont pas convexes, ce qui génère des difficultés dans leur résolution.

L'intérêt des LMIs est résumé dans les quatre points décrits ci-dessous.

1. *Convexité* : La LMI (1.2.6) définit une contrainte convexe en  $x$ . En effet,  $\forall \{x, y\} \in \mathbb{E}$ ,  $\mathbb{E} := \{x | F(x) > 0\}$ , alors

$$F(\alpha x + (1 - \alpha)y) \leq \alpha F(x) + (1 - \alpha)F(y) \quad \forall \alpha \in ]0, 1[, \quad (1.2.5)$$

où  $F(x)$  est donnée par (1.2.1).

2. *Concaténation* : Des LMIs multiples peuvent se ramener en une seule. En effet, résoudre les deux LMIs  $F_1(x) > 0$  et  $F_2(x) > 0$  est équivalent à résoudre  $\bar{F}(x) > 0$  avec  $\bar{F} = \text{diag}(F_1, F_2)$ .
3. *Algorithmes* : Les algorithmes utilisés pour résoudre les contraintes LMI sont efficaces : une bonne initialisation garantit la convergence de l'algorithme, cette convergence étant à temps polynômial. Actuellement, les principales classes d'algorithmes sont basées sur les méthodes des points intérieurs ([30]).
4. *Applications* : De nombreuses conditions classiques en automatique peuvent se formuler sous la forme de problème LMI. De plus, il est possible de convertir certaines inégalités non linéaires (notamment de Riccati) en LMI par l'utilisation de lemme de Schur.

## 1.2.2 Lemmes utiles pour les LMIs

Nous collectons quelques inégalités mathématiques et lemmes utiles qui ont été largement utilisés dans le manuscrit. Rappelons tout d'abord le "complément de Schur" qui permet de transformer certaines inégalités matricielles non linéaires en LMI.

**Lemme 1** (Lemme de Schur ou complément de Schur). Soit  $M$ ,  $N$  et  $Q$  trois matrices de dimensions appropriées telles que  $Q = Q'$  et  $M = M'$ . Alors,

$$\left[ \begin{array}{cc} Q & N \\ N' & M \end{array} \right] < 0 \quad (1.2.6)$$

si et seulement si

$$M < 0, \text{ et } Q - NM^{-1}N' < 0$$

ou de manière équivalente

$$Q < 0, \text{ et } M - N'Q^{-1}N < 0.$$

*Démonstration.* Posons  $G = \begin{pmatrix} M & N \\ N^T & Q \end{pmatrix}$ . En utilisant le principe de congruence, on déduit que

$$G < 0 \rightarrow S^T G S < 0, \forall S \text{ inversible.}$$

En choisissant  $S = \begin{pmatrix} 0 & I \\ I & 0 \end{pmatrix}$ , on obtient

$$\begin{pmatrix} 0 & I \\ I & 0 \end{pmatrix} \begin{pmatrix} M & N \\ N^T & Q \end{pmatrix} \begin{pmatrix} 0 & I \\ I & 0 \end{pmatrix} = \begin{pmatrix} Q & N^T \\ N & M \end{pmatrix}.$$

D'où l'équivalence

$$\begin{pmatrix} Q & N^T \\ N & M \end{pmatrix} < 0 \Leftrightarrow \begin{pmatrix} M & N \\ N^T & Q \end{pmatrix} < 0.$$

Pour démontrer (1.2.6), il suffit de choisir  $S = \begin{pmatrix} I & NQ^{-1} \\ 0 & I \end{pmatrix}$  et de remarquer que la matrice  $S$  est inversible, de plus on a  $\begin{pmatrix} I & NQ^{-1} \\ 0 & I \end{pmatrix}^{-1} = \begin{pmatrix} I & -NQ^{-1} \\ 0 & I \end{pmatrix}$ . En effet, il suffit d'utiliser la factorisation suivante

$$G = \begin{pmatrix} M & N \\ N^T & Q \end{pmatrix} = \begin{pmatrix} I & NQ^{-1} \\ 0 & I \end{pmatrix} \begin{pmatrix} M - NQ^{-1}N^T & 0 \\ 0 & Q \end{pmatrix} \begin{pmatrix} I & NQ^{-1} \\ 0 & I \end{pmatrix}^T$$

pour avoir les inégalités désirées. La preuve est ainsi achevée. □

Une autre transformation est dicrit par le lemme suivant :

**Lemme 2.** [23] *Les deux conditions suivantes sont équivalents :*

1. *Il existe une matrice symétrique  $P > 0$  telle que*

$$A^T P A - P < 0.$$

2. *Il existe une matrice symétrique  $P$  et une matrice  $G$  telles que*

$$\begin{bmatrix} P & A^T G^T \\ G A & G + G^T - P \end{bmatrix} > 0.$$

*Le lemme suivant jeu un rôle centrale pour les résultats de ce travail.*

**Lemme 3.** [18] *Soit  $M \in \mathbb{R}^{n \times n}$ ,  $M = M' > 0$ ,  $\gamma$  un scalaire positif, et une fonction  $\omega : [0, \gamma] \rightarrow \mathbb{R}^n$ , alors intégration suivante est bien définie, on a*

$$\gamma \int_0^\gamma \omega'(\beta) M \omega(\beta) d\beta \geq \left( \int_0^\gamma \omega(\beta) d\beta \right)' M \left( \int_0^\gamma \omega(\beta) d\beta \right). \quad (1.2.7)$$

*Démonstration.* C'est claire en utilisant le complément de Schur que :

$$\begin{pmatrix} \omega'(\beta)M\omega(\beta) & \omega'(\beta) \\ \omega(\beta) & M^{-1} \end{pmatrix} \geq 0$$

Pour tout  $0 \leq \beta \leq \gamma$  l'intégrale de l'inégalité précédente reste toujours vraie donc,

$$\begin{pmatrix} \int_0^\gamma \omega'(\beta)M\omega(\beta)d\beta & \int_0^\gamma \omega'(\beta)d\beta \\ \int_0^\gamma \omega(\beta)d\beta & \gamma M^{-1} \end{pmatrix} \geq 0$$

En appliquant le lemme de Schur une autre fois on aura l'inégalité (1.2.5)

□

**Lemme 4.** [20] Les deux conditions suivantes sont équivalents :

- (1)  $A$  est stable et  $\|C(zI - A)^{-1}B + D\|_\infty < \gamma$ .
- (2)  $A'PA - P + C'C + (A'PB + C'D)R^{-1}(B'PA + D'C) < 0$  où  $R = \gamma^2I - B'PB - D'D$ .

## 1.3 La réduction des systèmes dynamiques

### 1.3.1 Problématique générale

La problématique générale associée à la réduction de modèle est décrite de manière schématique à la figure 1. Partant d'un système physique quelconque, et de données évaluées expérimentalement ou numériquement, la phase de modélisation consiste à déterminer un ensemble d'équations différentielles ordinaires (**EDOs**) ou d'équations aux dérivées partielles (**EDPs**) représentatif du système physique. Dans le cas où un système d'EDPs est obtenu dans la phase de modélisation, celui-ci est généralement discrétisé en espace afin d'obtenir un système d'EDOs que l'on nomme par la suite  $S$ . La phase de réduction de modèles consiste à déterminer un système dynamique  $\hat{S}$  en réduisant de manière appropriée le nombre d'EDOs nécessaire à la description du système. Finalement, le modèle réduit  $\hat{S}$  est utilisé pour simuler ou contrôler le système  $S$ . Éventuellement, le système d'EDOs est également discrétisé en temps, conduisant alors à un système dynamique discret.

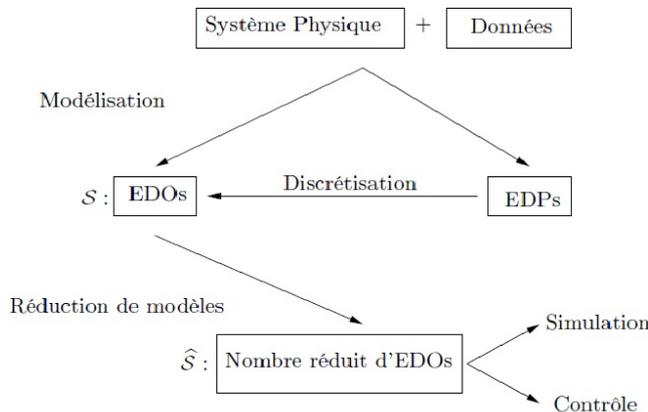


Figure 1.1 - Description schématique de la réduction de modèles.

Soit  $\mathbb{T}$  l'ensemble des valeurs prises par le temps (selon le cas, on pourra considérer  $\mathbb{T} = \mathbb{R}^+$ ,  $\mathbb{R}^-$  ou  $\mathbb{R}$  pour un système continu en temps, ou  $\mathbb{T} = \mathbb{Z}^+$ ,  $\mathbb{Z}^-$  ou  $\mathbb{Z}$  pour un système discret), on considère que le système  $\hat{\mathbf{S}}$  peut s'écrire de manière générique comme un système d'EDOs du premier ordre, soit :

$$\mathbf{S} : \begin{cases} \dot{x}(t) = f(t, x(t), u(t)), & x(t_0) = x_0 \\ y(t) = g(t, x(t), u(t)) \end{cases} \quad (1.3.1)$$

où

- $x \in \mathbb{X} = \{x : \mathbb{T} \rightarrow \mathbb{R}^n\}$ , est la variable d'état,
  - $u \in \mathbb{U} = \{u : \mathbb{T} \rightarrow \mathbb{R}^m\}$ , la variable d'entrée ou de commande,
  - $y \in \mathbb{Y} = \{y : \mathbb{T} \rightarrow \mathbb{R}^p\}$ , la variable de sortie ou l'observable,
- et où  $f$  et  $g$  sont des fonctions vectorielles correctement dimensionnées. Le système  $\mathbf{S}$  peut donc être représenté par le schéma de la figure 2.

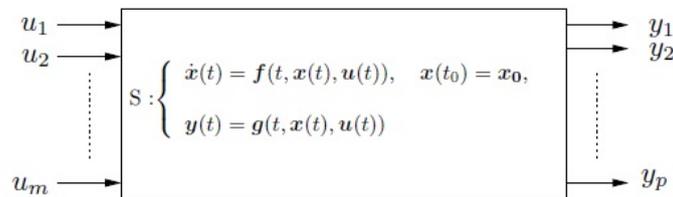


Figure 2.1 - Représentation schématisée du système dynamique  $\mathbf{S}$ .

L'objectif de la réduction de modèle est donc de remplacer le système (1.3.1) par :

$$\hat{\mathbf{S}} : \begin{cases} \hat{x}(t) = \hat{f}(t, \hat{x}(t), u(t)), \\ \hat{y}(t) = \hat{g}(t, \hat{x}(t), u(t)), \end{cases} \quad (1.3.2)$$

avec

- $\hat{x} \in \hat{\mathbb{X}} = \{\hat{x} : \mathbb{T} \rightarrow \mathbb{R}^r\}$ , avec  $r \ll n$ ,
- $\hat{y} \in \hat{\mathbb{Y}} = \{y : \mathbb{T} \rightarrow \mathbb{R}^p\}$ . Par ailleurs, la construction du modèle réduit  $\hat{\mathbf{S}}$  doit être réalisée en

s'assurant de vérifier si possible l'ensemble des conditions suivantes :

1. erreur d'approximation faible i.e.

$$\|y - \hat{y}\| < \epsilon \cdot \|u\|$$

quelle que soit l'entrée  $u$  où  $\epsilon$  est ici une tolérance donnée ;

2. stabilité du système  $S$  préservées ;
3. procédure de réduction stable numériquement et efficace.

Souvent en pratique le système  $\mathbf{S}$  non linéaire est linéarisé autour d'une solution d'équilibre. On

détermine ainsi un système linéaire dont les paramètres sont variables en temps que l'on peut encore noter  $\mathbf{S}_{LPTV}$  où *LPTV* signifie *Linear, Parameter, Time-Varying*. Ce système est donné par :

$$\mathbf{S} : \begin{cases} x(t) = A(t)x(t) + B(t)u(t), \\ y(t) = C(t)x(t) + D(t)u(t) \end{cases} \quad (1.3.3)$$

où  $A \in \mathbb{R}^{n \times n}$ ,  $B \in \mathbb{R}^{n \times m}$ ,  $C \in \mathbb{R}^{p \times n}$ ,  $D \in \mathbb{R}^{p \times m}$ .

Finalement, si les paramètres du système ne dépendent pas du temps, on obtient le système  $\mathbf{S}_{LTI}$  où *LTI* signifie ici *Linear Time-Invariant*. Ce système est alors représenté par :

$$\mathbf{S} : \begin{cases} x = Ax(t) + Bu(t), \\ y(t) = Cx(t) + Du(t) \end{cases} \quad (1.3.4)$$

Les systèmes  $\mathbf{S}_{LPTV}$  et  $\mathbf{S}_{LTI}$  ont été abondamment analysés par les automaticiens ou les mathématiciens appliqués (Zhou et al., 1996, par exemple). Dans le cas des systèmes linéaires, de nombreux résultats existent donc sur la réduction de modèles de systèmes dynamiques de grande dimension et la recherche est toujours active dans ce domaine comme le prouvent de très récentes publications sur le sujet (Antoulas, 2005; Benner et al., 2005). Selon Antoulas (2005), trois grandes classes de méthodes d'approximation existent :

1. celles basées sur une méthode de Krylov (Lanczos, Arnoldi, Interpolation),
2. celles basées sur la Décomposition aux Valeurs Singulières (SVD),
3. enfin, des méthodes itératives combinant certains aspects provenant d'une méthode de Décomposition en Valeurs Singulières et d'autres provenant d'une méthode de Krylov.

Ces méthodes possédant toutes des forces et des faiblesses différentes [21],[24],[25], il serait certainement intéressant de les tester pour des problèmes de mécanique des fluides. Cependant, les équations de Navier-Stokes étant non linéaires, peu de méthodes sont directement utilisables, à moins de linéariser le système d'état autour d'une solution d'équilibre. Citons par exemple les travaux d'Allan (2000), dans lesquels une méthode basée sur les sous-espaces de Krylov a été mise en œuvre avec succès afin de contrôler en boucle fermée l'écoulement dans une cavité entraînée. Dans la suite de ces notes, on s'intéressera exclusivement aux méthodes de Krylov et on présentera plus particulièrement la réduction de modèles par la méthode de projection (Arnoldi).

### 1.3.2 Méthodes de projection sur des sous-espaces de Krylov

La modélisation des problèmes que l'on rencontre en ingénierie et dans les domaines appliqués, conduit éventuellement après une étape de discrétisation, à la résolution de systèmes linéaires d'équations en dimension finie. Le traitement numérique de ces systèmes complexes nécessite des méthodes rapides et stables, d'où le recours à des techniques itératives de projections orthogonales qui permettent de réduire la taille du système à étudier. Ces méthodes de projection consistent en général à construire une matrice  $V$  dite projetée, de taille plus petite que celle de la matrice de départ. Parmi les approches de projection qui ont été largement étudiées et exploitées avec succès, il y a les méthodes de type Krylov qui sont basées sur des techniques de projection orthogonale sur un sous espace  $K_m$ , appelé << sous-espace de Krylov >> de taille  $m$ , où  $m$  croît avec les itérations. L'approche par des méthodes de type Krylov est assurée pour  $m = N$  (en arithmétique exacte),

bien qu'on espère qu'elle se produise pour  $m \ll N$  ( $N$  étant la taille de la matrice du problème discret).

## Les sous-espaces de Krylov

**Définition 8.** *Le sous-espace de krylov d'ordre  $m$  est défini par :*

$$K_m(A, v) = \text{vect} (v, Av, A^2v, \dots, A^{m-1}v)$$

où  $A \in \mathbb{R}^{n \times n}$  et  $v \in \mathbb{R}^n$ , est le sous-espace engendré par les puissances successives de  $A$  appliquées au vecteur  $v$ .

### 1.3.3 Méthode d'Arnoldi

la méthode d'Arnoldi est une méthode de projection orthogonale sur un sous-espace de Krylov, généralement appliquée aux matrices non symétriques. Cette méthode a été introduite par Arnoldi en 1950 [7], dans le but de factoriser une matrice sous forme Hessenberg supérieure [4, 24]. Cette stratégie est utile pour l'approximation de valeurs propres de matrices de grande taille. Si l'algorithme arrive à la même itération,  $m \ll n$ , on obtient une matrice  $\mathbf{H}_m$  de forme Hessenberg supérieure de taille  $m \times m$ , et une matrice orthonormale  $V_m$  de taille  $n \times m$  dont les colonnes sont définies par des vecteurs  $v_1, \dots, v_m$ . Ces vecteurs forment une base orthonormale du sous-espace de Krylov  $K_m(A, v_1) = [v_1, Av_1, \dots, A^{m-1}v_1]$ .

**Définition 9. (matrice d'Hessenberg).**  $\mathbf{H} = (h_{ij})_{1 \leq i, j \leq n}$  est une matrice sous forme Hessenberg supérieure (resp. inférieure) si la condition (a) (resp. (b)) est vérifiée.

$$\forall i \succ j + 1, h_{ij} = 0, \dots (a)$$

$$\forall j \succ i + 1, h_{ij} = 0, \dots (b)$$

### 1.3.4 Construction de la base d'Arnoldi

La construction des vecteurs  $v_j$  orthonormés et de la matrice  $H_m$  est implémentée par **l'Algorithme 1** Cet algorithme a comme paramètres :

1.  $v_1$ , un vecteur initial de norme 1;
2.  $A$ , la matrice du système aux valeurs propres considéré;
3.  $m$ , la taille de l'espace de Krylov.

Pour un  $m$  fixé, il est possible de calculer une base orthonormale de  $K_m(A, v)$  en utilisant l'algorithme d'Arnoldi.

En posant  $v_1 = v / \|v\|_2$ , cette méthode génère une base orthonormale  $v_i$  de  $K_m(A, v_1)$  en utilisant le procédé de GSM (Gram-Schmidt modifiée) [4].

Pour  $k = 1, \dots, m$ , l'algorithme d'Arnoldi consiste à calculer :

$$h_{ik} = v_i' Av_k, i = 1, 2, \dots, k,$$

$$w_k = Av_k - \sum_{i=1}^k h_{ik} v_i, h_{k+1,k} = \|w_k\|_2.$$

Si  $w_k = 0$ , le processus s'interrompt (on parle de *breakdown*); autrement, on pose  $v_{k+1} = w_k / \|w_k\|_2$  et on reprend l'algorithme en augmentant  $k$  de 1. On peut montrer que si la méthode s'achève à l'étape  $m$ , alors les vecteurs  $v_1, \dots, v_m$  forment une base de  $K_m(A, v)$ . Dans ce cas, en notant  $V_m \in \mathbb{R}^{n \times m}$  la matrice rectangulaire à  $n$  lignes et  $m$  colonnes dont les colonnes sont les  $m$  premiers vecteurs de la base d'Arnoldi,

$$V_m = [v_1 | \dots | v_m]_{(n \times m)},$$

alors l'orthonormalisation des vecteurs nous donne

$$V_m' V_m = I_m,$$

et l'orthogonalisation nous donne

$$V_m' w_m = 0,$$

on aura

$$V_{m+1} \tilde{H}_m = AV_m = V_m H_m + w_m e_m^T \implies V_m' AV_m = H_m.$$

La matrice  $\tilde{H}_m = (h_{ij})_{(m+1, m)}$ , tel que  $h_{ij}$  est le coefficient d'orthogonalisation de  $Av_j$  par rapport à  $v_i$ , prend la forme :

$$H_{m+1, m} = \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1j} & \cdots & h_{1m} \\ h_{21} & h_{22} & \cdots & h_{2j} & \cdots & h_{2m} \\ 0 & h_{32} & \cdots & h_{3j} & \cdots & h_{3m} \\ \vdots & \ddots & \ddots & \vdots & & \vdots \\ & & & h_{j+1, j} & & h_{j+1, m} \\ \vdots & & & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & 0 & & h_{m+1, m} \end{bmatrix}$$

et la matrice de Hessenberg  $H_m$  obtenue à partir de  $\tilde{H}_m$  en supprimant sa dernière ligne. avec

$$H_m = \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1m} \\ h_{21} & h_{22} & \cdots & h_{2m} \\ 0 & h_{32} & \ddots & h_{3m} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & h_{m-1, m} & h_{m, m} \end{bmatrix}$$

```

Données :  $A, v, m$ 
Résultats :  $V_m, H_m$ 
Initialisation :  $v_1 = \frac{v}{\|v\|}$ ;
  Pour  $j = 1$  à  $m$  faire
     $w_j = Av_j$ ;
    Pour  $i = 1$  à  $j - 1$  faire
       $h_{ij} = v_i' w_j$ ;
       $w_j = w_j - h_{ij} v_i$ ;
    Fin pour
     $h_{j+1,j} = \|w_j\|$ ;
    Si ( $h_{j+1,j} = 0$ ) Alors
      Stop;
    Sinon
       $v_{j+1} = \frac{w_j}{h_{j+1,j}}$ ;
    FinSi
  Fin pour

```

*Algorithme 1 : Procédure d'Arnoldi*

Construction d'une base orthonormale pour l'espace de  $K_m(A, v_1)$  et de la matrice d'Hessenberg associée  $\mathbf{H}_m$  [4, 24].

**Remarque 1.** Dans des situations pratiques, le calcul des valeurs propres d'une matrice carrée  $A$  est une tâche très complexe, c'est pour cela, il est commode de calculer les valeurs propres de **Ritz** ( $\mathbf{H}_m$ ) qui peuvent être considérés comme des bonnes approximations à une certaine précision. Cette approche est bien justifiée théoriquement et numériquement.

**Remarque 2.** Les valeurs propres de **Ritz** sont des approximations des valeurs propres de  $A$  qui dépendent (explicitement) du choix de vecteur initial  $v$  dans la procédure d'Arnoldi.

# Chapitre 2

## La réduction des systèmes non-linéaire à temps discret

Ce chapitre a essentiellement pour objectif de présenter et développer les récentes techniques proposées par S. Ibrir et M. Bettayeb [1] pour la réduction d'une classe de systèmes non linéaires discrets et lipschitziens. Ces techniques font intervenir le théorème de la moyenne, les méthodes de projection de Krylov et quelques résultats de linéarisation liés à l'optimisation convexe.

Nous fournissons, dans la première partie de ce chapitre, un résultat qui donne des conditions suffisantes, exprimées sous forme d'un problème d'optimisation convexe, permettant d'approcher au sens de la norme  $H_\infty$  un système non linéaire par un système linéaire d'ordre réduit (LTI). Notons que les premières conditions garantissant le critère de performance  $H_\infty$  via la stabilité de Lyapunov sont exprimés par la résolution d'un problème d'optimisation sous contraintes bilinéaires. Ces dernières sont linéarisées grâce au Lemme de M.C. de Oliveira et al.

Dans la deuxième partie nous avons développé la méthode d'approximation au sens de la norme  $H_\infty$  d'un système non linéaire par un système non linéaire d'ordre réduit. Celle-ci combine la projection sur les sous espaces de Krylov et les outils de l'optimisation convexe.

Afin de valider les performances des deux méthodes étudiées, nous présentons une application sur la réduction du système dynamique masse-ressort-amortisseur (SMD) soumis à un frottement non linéaire.

### 2.1 Conditions suffisantes pour l'approximation d'un système non-linéaire par un système linéaire d'ordre réduit

Considérons le système non-linéaire à temps discret :

$$\begin{cases} x_{k+1} = Ax_k + f(x_k) + Bu_k, \\ y_k = Cx_k, \end{cases} \quad (2.1.1)$$

où  $x_k \in \mathbb{R}^n$  représente le vecteur d'état,  $y_k \in \mathbb{R}^p$  est la sortie mesurée,  $u_k \in \mathbb{R}^m$  est la commande.  $A, B, C$  sont des matrices réelles de dimensions appropriées.

Les paramètres du système (2.1.1) sont soumis aux hypothèses suivantes :

**Hypothèses 1.** –

- $f$  est différentiable et globalement Lipschitzienne avec  $f(0) = 0$  ;
- Les sous systèmes suivants :

$$\begin{cases} x_{k+1} = (A + F_i)x_k + Bu_k, & 1 \leq i \leq \mu \\ y_k = Cx_k. \end{cases} \quad (2.1.2)$$

sont exponentiellement stable pour  $u_k = 0$  où  $F_1 \dots F_i$  sont les sommets de l'enveloppe convexe de la Jacobienne  $\frac{\partial f}{\partial x_k}(x_k)$ .

Notons bien qu'en vertu du Lemme de reformulation de la condition de Lipschitz sur  $f$  établi par Zemouche et Boutayed [32], cette deuxième hypothèse est une condition nécessaire pour la stabilité du système (2.1.1).

### 2.1.1 Formulation du problème d'optimisation

Notre objectif dans cette sous section consiste à approcher (au sens de la norme  $H_\infty$ ) le système (2.1.1) par un système d'ordre réduit linéaire à temps discret et stable de la forme :

$$\begin{cases} \zeta_{k+1} = A_r \zeta_k + B_r u_k, \\ z_k = C_r \zeta_k \end{cases} \quad (2.1.3)$$

où  $A_r \in \mathbb{R}^{r \times r}$ ,  $B_r \in \mathbb{R}^{r \times m}$  et  $C_r \in \mathbb{R}^{p \times r}$  sont des variables à déterminer qui vérifient l'inégalité suivante (critère de performance  $H_\infty$ ) :

$$\sum_{k=0}^{N-1} \|z_k - y_k\|^2 \leq \gamma^2 \sum_{k=0}^{N-1} \|u_k\|^2. \quad (2.1.4)$$

$\forall u_k \in \mathbb{R}^m, \forall k \in [0, N - 1]$ , où le paramètre  $\gamma$  est un réel positif à minimiser.

Les premières conditions suffisantes pour l'approximation du système non-linéaire (2.1.1) par un système linéaire d'ordre réduit sont résumées dans le théorème suivant :

**Théorème 5** ([1]). *Considérons les systèmes (2.1.1) et (2.1.3), et les hypothèses 1. Le système d'ordre réduit est exponentiellement stable pour  $u_k = 0$  et vérifié la condition d'optimalité (2.1.4) pour tout  $u_k \neq 0$  si pour un  $r$ , ( $0 < r < n$ ) donné, il existe trois matrices  $A_r \in \mathbb{R}^{r \times r}$ ,  $B_r \in \mathbb{R}^{r \times m}$ ,  $C_r \in \mathbb{R}^{p \times r}$ , un réel positif  $\gamma$ , et une matrice définie positive  $P \in \mathbb{R}^{(n+r) \times (n+r)}$  vérifiant l'inégalité matricielle suivante :*

$$\begin{bmatrix} -P + \mathcal{A}'(i)P\mathcal{A}(i) + C'_\Delta C_\Delta & \mathcal{A}'(i)PB_\Delta \\ (*) & -\gamma^2 I + B'_\Delta PB_\Delta \end{bmatrix} < 0, \quad (2.1.5)$$

pour tout  $1 \leq i \leq \mu$ , où

$$\mathcal{A}(i) = \begin{bmatrix} A + F_i & (\mathbf{0}) \\ (\mathbf{0}) & A_r \end{bmatrix}, B_\Delta = \begin{bmatrix} B \\ B_r \end{bmatrix}, C_\Delta = [ C \quad -C_r ].$$

*Démonstration.* Considérons le vecteur augmenté suivant

$$\xi_k = \begin{bmatrix} x_k \\ \zeta_k \end{bmatrix} \in \mathbb{R}^{(n+r)}.$$

Comme  $f(0) = 0$ , alors par le théorème de la valeur moyenne [1, 21] on a,

$$f(x_k) = \left( \int_0^1 \frac{\partial f(\lambda x_k)}{\partial x_k} d\lambda \right) x_k. \quad (2.1.6)$$

Par conséquent, la dynamique du système augmenté est donnée par

$$\begin{cases} \xi_{k+1} = A_\Delta(x_k)\xi_k + B_\Delta u_k \\ y_\Delta = C_\Delta \xi_k, \end{cases} \quad (2.1.7)$$

où

$$A_\Delta(x_k) = \begin{bmatrix} A + \int_0^1 \frac{\partial f(\lambda x_k)}{\partial x_k} d\lambda & (\mathbf{0}) \\ (\mathbf{0}) & A_r \end{bmatrix} \quad (2.1.8)$$

Une condition suffisante garantissant l'inégalité (2.1.4) consiste à trouver une fonction candidate de Lyapunov  $V_k = \xi_k' P \xi_k$ ,  $P = P' > 0$ , telle que

$$\|y_\Delta\|^2 - \gamma^2 \|u_k\|^2 + V_{k+1} - V_k < 0, \forall k \in \mathbb{Z}_{\geq 0}. \quad (2.1.9)$$

En effet, comme les conditions initiales sont nulles, on a d'une part  $V_0 = 0$  et d'autre part, si l'inégalité précédente est vérifiée on aura

$$\sum_{k=0}^{N-1} (\|y_\Delta\|^2 - \gamma^2 \|u_k\|^2) + V_N - V_0 = \sum_{k=0}^{N-1} (\|y_\Delta\|^2 - \gamma^2 \|u_k\|^2 + V_{k+1} - V_k) < 0, \quad (2.1.10)$$

est vérifié le long de la trajectoire du système (2.1.7). L'inégalité (2.1.9) est équivalente (par définition de la norme) à

$$\begin{aligned} & -\gamma^2 u_k' u_k + \xi_k' (A_\Delta'(x_k) P A_\Delta(x_k) - P) \xi_k + u_k' B_\Delta' P A_\Delta(x_k) \xi_k \\ & + \xi_k' A_\Delta'(x_k) P B_\Delta u_k + u_k' B_\Delta' P B_\Delta u_k + \xi_k' C_\Delta' C_\Delta \xi_k < 0. \end{aligned} \quad (2.1.11)$$

Soit

$$\mathcal{F}(\lambda, x_k) = \begin{bmatrix} A + \frac{\partial f}{\partial x_k}(\lambda x_k) & (\mathbf{0}) \\ (\mathbf{0}) & A_r \end{bmatrix}. \quad (2.1.12)$$

Alors  $A_\Delta(x_k)$  peut s'écrire sous la forme :

$$A_\Delta(x_k) = \int_0^1 \mathcal{F}(\lambda, x_k) d\lambda \quad (2.1.13)$$

En utilisant le **Lemme 3**, on obtient

$$\xi_k' A_\Delta'(x_k) P A_\Delta(x_k) \xi_k = \int_0^1 (\xi_k' \mathcal{F}'(\lambda, x_k)) d\lambda P \int_0^1 (\xi_k \mathcal{F}(\lambda, x_k)) d\lambda \leq \int_0^1 (\xi_k' \mathcal{F}'(\lambda, x_k) P \xi_k \mathcal{F}(\lambda, x_k)) d\lambda. \quad (2.1.14)$$

En prenant en considération (2.1.13), (2.1.14) dans l'expression de (2.1.11), une condition suffisante vérifiant (2.1.11) est donnée par

$$\begin{aligned} & -\gamma^2 \int_0^1 u'_k u_k d\lambda + \int_0^1 \xi'_k (\mathcal{F}'(\lambda, x_k) P \mathcal{F}(\lambda, x_k) - P) \xi_k d\lambda + \int_0^1 u'_k B'_\Delta P \mathcal{F}(\lambda, x_k) \xi_k d\lambda \\ & + \int_0^1 \xi' \mathcal{F}'(\lambda, x_k) P B_\Delta u_k d\lambda + \int_0^1 u'_k B'_\Delta P B_\Delta u_k d\lambda + \int_0^1 \xi'_k C'_\Delta C_\Delta \xi_k d\lambda < 0, \forall x_k. \end{aligned} \quad (2.1.15)$$

Soit  $\eta_k = \begin{bmatrix} \xi'_k & u'_k \end{bmatrix}'$ , alors (2.1.11) s'écrira comme suit

$$\int_0^1 \eta'_k \begin{bmatrix} \mathcal{F}'(\lambda, x_k) P \mathcal{F}(\lambda, x_k) - P + C'_\Delta C_\Delta & \mathcal{F}'(\lambda, x_k) P B_\Delta \\ (\star) & -\gamma^2 I + B'_\Delta P B_\Delta \end{bmatrix} \eta_k d\lambda < 0, \forall x_k. \quad (2.1.16)$$

L'inégalité (2.1.16) est équivalente grâce au complément de Schur à :

$$\int_0^1 \begin{bmatrix} -P + C'_\Delta C_\Delta & \mathcal{F}'(\lambda, x_k) P B_\Delta & \mathcal{F}'(\lambda, x_k) P \\ (\star) & -\gamma^2 I + B'_\Delta P B_\Delta & (\mathbf{0}) \\ (\star) & (\star) & -P \end{bmatrix} d\lambda < 0, \forall x_k. \quad (2.1.17)$$

D'après les **hypothèses 1**, par le principe de convexité, il existe un ensemble de scalaires positifs  $(\alpha_i(x_k))_{1 \leq i \leq \mu}$  telle que

$$\frac{\partial f}{\partial x_k}(x_k) = \sum_{i=1}^{\mu} \alpha_i(x_k) F_i, \quad \sum_{i=1}^{\mu} \alpha_i(x_k) = 1, \forall x_k, \quad (2.1.18)$$

ce qui implique que

$$\int_0^1 \mathcal{F}(\lambda, x_k) d\lambda = \int_0^1 \sum_{i=1}^{\mu} \alpha_i(\lambda x_k) \mathcal{A}(i) d\lambda, \quad \int_0^1 \sum_{i=1}^{\mu} \alpha_i(\lambda x_k) d\lambda = 1. \quad (2.1.19)$$

donc l'inégalité (2.1.16) est équivalente à

$$\int_0^1 \sum_{i=1}^{\mu} \alpha_i(\lambda x_k) \begin{bmatrix} -P + C'_\Delta C_\Delta & \mathcal{A}'(i) P B_\Delta & \mathcal{A}'(i) P \\ (\star) & -\gamma^2 I + B'_\Delta P B_\Delta & (\mathbf{0}) \\ (\star) & (\star) & -P \end{bmatrix} d\lambda < 0, \forall x_k. \quad (2.1.20)$$

Par le principe de convexité on aura

$$\begin{bmatrix} -P + C'_\Delta C_\Delta & \mathcal{A}'(i) P B_\Delta & \mathcal{A}'(i) P \\ (\star) & -\gamma^2 I + B'_\Delta P B_\Delta & (\mathbf{0}) \\ (\star) & (\star) & -P \end{bmatrix} < 0, \quad (2.1.21)$$

**Remarque 3.** Les conditions (2.1.5) entraînent grâce au lemme de Schur, que

$$-P + \mathcal{A}'(i) P \mathcal{A}(i) + C'_\Delta C_\Delta < 0, \forall i.$$

Comme  $C'_\Delta C_\Delta \geq 0$ , on a nécessairement

$$-P + \mathcal{A}'(i) P \mathcal{A}(i) < 0, \forall i.$$

Cela signifie que les matrices  $\mathcal{A}(i)$  sont Schur Stable (ou de façon équivalente les valeurs propres de  $\mathcal{A}(i)$  sont à l'intérieur du disque unité). Comme  $\mathcal{A}(i)$  sont diagonales par bloc, on déduit que les valeurs propres de  $A_r$  sont de module strictement inférieur à 1. d'où la stabilité asymptotique (exponentielle) du système réduit.

□

**Remarque 4.** Le résultat de **théorème 5** est un problème d'optimisation non convexe, ce résultat sera le point de départ pour le développement de nouvelles conditions suffisantes donnant lieu à un problème d'optimisation convexe.

## 2.1.2 Linéarisation du problème

Notre objectif dans cette sous section est de transformer le problème d'optimisation non convexe données dans **théorème 5** en un problème d'optimisation convexe.

**Théorème 6** ([1]). *Considérons le système (2.1.1) soumis aux **hypothèses 1**. Soit  $r$ , ( $r < n$ ) l'ordre du système réduit. Soit aussi  $E \in \mathbb{R}^{n \times r}$  une matrice arbitraire à plein rang ( $\text{rang}(E) = r$ ). S'il existe une matrice  $P \in \mathbb{R}^{(n+r) \times (n+r)}$ , un ensemble de matrices  $X \in \mathbb{R}^{r \times r}$ ,  $Y \in \mathbb{R}^{r \times m}$ ,  $C_r \in \mathbb{R}^{p \times r}$ ,  $G_1(i) \in \mathbb{R}^{n \times n}$ ,  $G_2 \in \mathbb{R}^{r \times r}$ ,  $G_3(i) \in \mathbb{R}^{r \times n}$ ,  $1 \leq i \leq \mu$  tel que le problème d'optimisation suivant admet une solution,*

$$\begin{cases} \min_{P>0, X, Y, C_r, (G_i)_{1 \leq i \leq \mu}} & \gamma, \\ \text{sous les contraintes} & P = P' > 0, \mathcal{K} = \mathcal{K}' < 0 \end{cases},$$

(2.1.22) où

$$\mathcal{K} = \begin{bmatrix} -P & (\mathbf{0})_{n+r, m} & \begin{bmatrix} (A + F_i)' G_1'(i) & (A + F_i)' G_3'(i) \\ X' E' & X' \end{bmatrix} & \begin{bmatrix} C & -C_r \end{bmatrix}' \\ (\star) & -\gamma^2 I_m & \begin{bmatrix} B' G_1'(i) + Y' E' & B' G_3'(i) + Y' \end{bmatrix} & (\mathbf{0})_{m, p} \\ (\star) & (\star) & G_i + G_i' + P & (\mathbf{0})_{n+r, p} \\ (\star) & (\star) & (\star) & -I_p \end{bmatrix}, \forall 1 \leq i \leq \mu. \quad (2.1.23)$$

et

$$G_i = \begin{bmatrix} G_1(i) & E G_2 \\ G_3(i) & G_2 \end{bmatrix}. \quad (2.1.24)$$

alors, le système linéaire d'ordre réduit

$$\begin{cases} \zeta_{k+1} = G_2^{-1} X \zeta_k + G_2^{-1} Y u_k, \\ z_k = C_r \zeta_k \end{cases} \quad (2.1.25)$$

est asymptotiquement stable et approxime le système non linéaire (2.1.1) au sens de la norme  $\mathcal{H}_\infty$ .

*Démonstration.* L'idée de la démonstration consiste à linéariser (2.1.5) en utilisant le lemme de M.C de Oliveira. L'inégalité (2.1.5) se factorise comme suit :

$$\begin{bmatrix} I & (\mathbf{0}) & -\mathcal{A}'(i) \\ (\mathbf{0}) & I & -B'_\Delta \end{bmatrix} \times \begin{bmatrix} -P + C'_\Delta C_\Delta & (\mathbf{0}) & \mathcal{A}'(i) \\ (\star) & -\gamma^2 I & B'_\Delta \\ (\star) & (\star) & P \end{bmatrix} \times \begin{bmatrix} I & (\mathbf{0}) \\ (\mathbf{0}) & I \\ -\mathcal{A}(i) & B_\Delta \end{bmatrix} < 0. \quad (2.1.26)$$

D'où l'on déduit que (2.1.5) est vérifié si l'inégalité suivante satisfaite

$$\begin{bmatrix} -P + C'_\Delta C_\Delta & (\mathbf{0}) & \mathcal{A}'(i) \\ (\star) & -\gamma^2 I & B'_\Delta \\ (\star) & (\star) & P \end{bmatrix} < 0. \quad (2.1.27)$$

Soit  $(G_i)_{1 \leq i \leq \mu}$  un ensemble de matrices réelles de dimension  $(n+r) \times (n+r)$  partitionnées comme suit

$$G_i = \begin{bmatrix} G_1(i) & G_2(i) \\ G_3(i) & G_4(i) \end{bmatrix}, \quad (2.1.28)$$

En appliquant le **lemme 2** (lemme de M.C de Oliveira) pour l'inégalité (2.1.27) on aura

$$\begin{bmatrix} -P + C'_\Delta C_\Delta & (\mathbf{0}) & \mathcal{A}'(i)G'_i \\ (\star) & -\gamma^2 I & B'_\Delta G'_i \\ (\star) & (\star) & G_i + G'_i + P \end{bmatrix} < 0. \quad (2.1.29)$$

L'ensemble des inégalités matricielles (2.1.29) sont équivalentes par le complément de Schur à

$$\begin{bmatrix} -P & (\mathbf{0}) & \mathcal{A}'(i)G'_i & C'_\Delta \\ (\star) & -\gamma^2 I & B'_\Delta G'_i & (\mathbf{0}) \\ (\star) & (\star) & G_i + G'_i + P & (\mathbf{0}) \\ (\star) & (\star) & (\star) & -I \end{bmatrix} < 0, \forall 1 \leq i \leq \mu. \quad (2.1.30)$$

On injecte par l'expression de  $\mathcal{A}(i)$  et (2.1.28) dans l'inégalité (2.1.30) on obtient

$$\begin{bmatrix} -P & (\mathbf{0}) & \begin{bmatrix} A'_1 G'_1(i) + F'_1 G'_1(i) & A'_1 G'_3(i) + F'_1 G'_3(i) \\ \underbrace{A'_r G'_2(i)} & \underbrace{A'_r G'_4(i)} \end{bmatrix} & [C - C_r]' \\ (\star) & -\gamma^2 I & \begin{bmatrix} B'_1 G'_1(i) + \underbrace{B'_r G'_2(i)} & B'_1 G'_3(i) + \underbrace{B'_r G'_4(i)} \end{bmatrix} & (\mathbf{0}) \\ (\star) & (\star) & G_i + G'_i + P & (\mathbf{0}) \\ (\star) & (\star) & (\star) & -I \end{bmatrix} < 0, \forall 1 \leq i \leq \mu. \quad (2.1.31)$$

On remarque que il y a des termes bilinéaires ( $A'_r G'_2(i)$ ,  $A'_r G'_4(i)$ ,  $B'_r G'_2(i)$ ,  $B'_r G'_4(i)$ ) au niveau de l'inégalité (2.1.31), de plus  $A'_r$  et  $B'_r$  sont attachées à deux variables différentes. Un changement de variable permettra de transformer ces termes bilinéaires en de termes linéaires. En effet, il suffit d'imposer le choix particulier  $G_2(i) = G_4(i)$ , mais comme les matrices  $A'_r$  et  $B'_r$  sont fixées alors on fixe aussi la matrice  $G_2(i)$  en la considérant indépendante de  $i$ , c'est à dire  $G_2(i) = G_2$ .

Dans ce cas, on aura aussi un problème de dimension au niveau de bloc  $G_{2,2}$  de la matrice  $G_i$ , pour y remédier on multiplie la matrice  $G_2$  par une autre matrice  $E$  pour compléter la dimension de la forme

$$E = \begin{bmatrix} I_r \\ \mathbf{0}_{n-r,r} \end{bmatrix}.$$

On aura ainsi une matrice de la forme

$$G_i = \begin{bmatrix} G_1(i) & EG_2 \\ G_3(i) & G_2 \end{bmatrix}. \quad (2.1.32)$$

Utilisant maintenant la partition de la matrice  $G_i$  et posant  $X = G_2 A_r$  et  $Y = G_2 B_r$  alors,

$$\begin{aligned} \mathcal{A}'(i)G'_i &= \begin{bmatrix} (A + F_i)' G'_1(i) & (A + F_i)' G'_3(i) \\ X' E' & X' \end{bmatrix} \\ B'_\Delta G'_i &= [ B'_1 G'_1(i) + Y' E' \quad B'_1 G'_3(i) + Y' ]. \end{aligned} \quad (2.1.33)$$

En injectant (2.1.33) dans l'inégalité matricielle (2.1.31) on aura

$$\left[ \begin{array}{cc} -P & (\mathbf{0})_{n+r,m} \\ (\star) & -\gamma^2 I_m \\ (\star) & (\star) \\ (\star) & (\star) \end{array} \left[ \begin{array}{cc} (A + F_i)' G_1'(i) & (A + F_i)' G_3'(i) \\ X' E' & X' \\ B' G_1'(i) + Y' E' & B' G_3'(i) + Y' \\ G_i + G_i' + P & \\ (\star) & \end{array} \right] \left[ \begin{array}{c} [ C \quad -C_r ]' \\ (\mathbf{0})_{m,p} \\ (\mathbf{0})_{n+r,p} \\ -I_p \end{array} \right] \right] < 0, \forall 1 \leq i \leq \mu. \quad (2.1.34)$$

□

**Remarque 5.** Par lemme de Schur appliqué à (2.1.34) alors le bloc

$$\left[ \begin{array}{cc} G_i + G_i' + P & (\mathbf{0})_{n+r,p} \\ (\star) & -I_p \end{array} \right] < 0, \quad (2.1.35)$$

ce qui implique que  $G_i + G_i' < 0$  i.e, pour tout vecteur  $\eta \neq 0$  alors,

$$\eta' (G_i + G_i') \eta = 2\eta' G_i \eta < 0.$$

Ce qui implique que  $G_i < 0$ , et donc  $G_2 < 0$  (et donc inversible) par le lemme de Schur.

## 2.2 Approximation par un système non linéaire d'ordre réduit

Dans cette section, nous aborderons le problème d'approximation du système non-linéaire (2.1.1) par un autre système non-linéaire d'ordre  $r$  ( $r < n$ ). Il est alors question d'étudier simultanément les trois problèmes cités dans l'introduction. Pour un choix approprié  $r$ , ( $r < n$ ), nous nous intéresserons au modèle non linéaire d'ordre réduit suivant :

$$\begin{aligned}\zeta_{k+1} &= A_r \zeta_k + B_f f(W_r \zeta_k) + B_r u_k, \\ z_k &= C_r \zeta_k,\end{aligned}\tag{2.2.1}$$

où  $\zeta \in \mathbb{R}^r$  représente le nouveau vecteur d'état,  $z \in \mathbb{R}^p$  est la sortie mesurée,  $u \in \mathbb{R}^m$  est la commande.  $A_r \in \mathbb{R}^{r \times r}$ ,  $B_f \in \mathbb{R}^{r \times n}$ ,  $W_r \in \mathbb{R}^{n \times r}$ ,  $B_r \in \mathbb{R}^{r \times m}$  et  $C_r \in \mathbb{R}^{p \times r}$  sont des matrices réelles à déterminer telles que la condition d'optimalité suivante  $\|y_k - z_k\|_2 \leq \gamma \|u_k\|_2$  est satisfaite, avec  $\gamma$  est le plus petit réel positif vérifiant l'inégalité précédente.

Le problème du choix optimal  $r$  de l'ordre du système ne sera pas abordé dans le cadre de ce mémoire. Il est possible de le faire en procédant comme suit : pour  $r$  fixé, on résout le problème d'optimisation associé ainsi que  $\gamma_r$  associé, le choix optimal de  $r$  est celui qui minimise  $\gamma_r$ .

Soit  $V \in \mathbb{R}^{r \times m}$  une matrice à plein rang. Soit  $V\zeta_k$  le vecteur d'état qui approxime  $x_k$ . Le système (2.1.1) est équivalent à :

$$\begin{aligned}x_{k+1} &= V\zeta_{k+1} = AV\zeta_k + f(V\zeta_k) + Bu_k, \\ z_k &= CV\zeta_k.\end{aligned}\tag{2.2.2}$$

La méthode de projection de Krylov consiste à trouver une matrice  $U \in \mathbb{R}^{n \times r}$  telle que  $U'V$  est inversible.

En multipliant le système (2.2.2) par  $(U'V)^{-1}U'$ , alors le système d'ordre réduit prend la forme :

$$\begin{aligned}\zeta_{k+1} &= (U'V)^{-1}U'AV\zeta_k + (U'V)^{-1}U'f(V\zeta_k) + (U'V)^{-1}U'Bu_k, \\ z_k &= CV\zeta_k.\end{aligned}\tag{2.2.3}$$

Des conditions suffisantes pour la réduction du système (2.1.1) par un système non linéaire d'ordre réduit sont données par le théorème suivant :

**Théorème 7.** *Considérons le système (2.1.1) soumis aux **Hypothèses 1**. Si pour un  $r$  ( $r < n$ ) donné et une matrice à plein rang  $V \in \mathbb{R}^{n \times r}$ , il existe une matrice symétrique définie positive  $P_1 \in \mathbb{R}^{n \times n}$  tel que le problème d'optimisation convexe a une solution pour tout  $i, j; 1 \leq i, j \leq \mu$*

$\min_{P_1 > 0} \gamma,$   
t.q.  $\Omega = \Omega' < 0$ , où

$$\Omega = \begin{bmatrix} \begin{bmatrix} -P_1 & (\mathbf{0})_{n,r} \\ (\star) & -V'P_1V \end{bmatrix} & (\mathbf{0})_{n+r,m} & \begin{bmatrix} (A+F_i)'P_1 & (\mathbf{0})_{n,r} \\ (\star) & V'(A+F_j)P_1V \end{bmatrix} & \begin{bmatrix} C \\ -CV \end{bmatrix}' \\ (\star) & -\gamma^2 I_m & \begin{bmatrix} B'P_1 & B'P_1V \end{bmatrix} & (\mathbf{0})_{m,p} \\ (\star) & (\star) & \begin{bmatrix} -P_1 & (\mathbf{0})_{n,r} \\ (\star) & -V'P_1V \end{bmatrix} & (\mathbf{0})_{n+r,p} \\ (\star) & (\star) & (\star) & -I_p \end{bmatrix}.\tag{2.2.4}$$

Alors, le système non linéaire d'ordre réduit (2.2.1) est asymptotiquement stable pour  $u_k = 0$  de

plus, la condition d'optimalité  $\|y_k - z_k\|_2 \leq \gamma \|u_k\|_2$  est vérifiée  $\forall k > 0$ .

*Démonstration.* La démonstration de ce théorème se fera en deux étapes. Premièrement, nous dérivons les contraintes LMIs du problème d'optimisation. Dans une seconde étape, on vérifiera la stabilité asymptotique du système réduit lorsque  $u_k = 0$ . Prenons le système (2.1.1) et le système (2.2.1), alors on peut les écrire sous la forme du système augmenté (2.1.7) où  $A_\Delta(\xi_k)$  est maintenant égale à

$$A_\Delta(\xi_k) = \begin{bmatrix} A + \int_0^1 \frac{\partial f}{\partial x_k}(\lambda x_k) d\lambda & (\mathbf{0}) \\ (\mathbf{0}) & A_r + B_f \left( \int_0^1 \frac{\partial f}{\partial s}(s) \Big|_{s=\lambda W_r \zeta_k} d\lambda \right) W_r \end{bmatrix}. \quad (2.2.5)$$

On a clairement (par le principe de convexité)  $A_\Delta(\xi_k) \in \mathbf{Co} \{ \mathbf{A}(i, j), 1 \leq i, j \leq \mu \}$  où

$$\mathbf{A}(i, j) = \begin{bmatrix} A + F_i & (\mathbf{0}) \\ (\mathbf{0}) & A_r + B_f F_j W_r \end{bmatrix}. \quad (2.2.6)$$

Maintenant on remplace  $\mathcal{A}(i)$  par  $\mathbf{A}(i, j)$  dans (2.1.5) on aura,

$$\begin{bmatrix} -P + \mathbf{A}'(i, j) P \mathbf{A}(i, j) + C'_\Delta C_\Delta & \mathbf{A}'(i, j) P B_\Delta \\ (\star) & -\gamma^2 I + B'_\Delta P B_\Delta \end{bmatrix} < 0, \forall 1 \leq i \leq \mu, \quad (2.2.7)$$

qui est équivalent par décomposition en somme de deux matrices à,

$$\begin{bmatrix} -P + C'_\Delta C_\Delta & (\mathbf{0}) \\ (\star) & -\gamma^2 I \end{bmatrix} + \begin{bmatrix} \mathbf{A}'(i, j) P \\ B'_\Delta P \end{bmatrix} P^{-1} \begin{bmatrix} P \mathbf{A}(i, j) & P B_\Delta \end{bmatrix} < 0, \quad (2.2.8)$$

en appliquant le lemme de Schur, alors l'inégalité précédente est équivalente à

$$\begin{bmatrix} -P + C'_\Delta C_\Delta & (\mathbf{0}) & \mathbf{A}'(i, j) P \\ (\star) & -\gamma^2 I & B'_\Delta P \\ (\star) & (\star) & -P \end{bmatrix} < 0; \forall i. \quad (2.2.9)$$

En appliquant une autre fois le complément de Schur et en posant

$A_r = (V' P_1 V)^{-1} V' P_1 A V$ ,  $P = \text{diag}(P_1, V' P_1 V)$ ,  $B_r = (V' P_1 V)^{-1} V' P_1 B$ ,  $B_f = (V' P_1 V)^{-1} V' P_1$  et  $C_r = C V$ , on aura la LMI suivante

$$\begin{bmatrix} -P & (\mathbf{0}) & \mathbf{A}'(i, j) \text{diag}(P_1, V' P_1 V) & \begin{bmatrix} C' \\ -V' C' \end{bmatrix} \\ (\star) & -\gamma^2 I & \begin{bmatrix} B \\ (V' P_1 V)^{-1} V' P_1 B \end{bmatrix} \text{diag}(P_1, V' P_1 V) & (\mathbf{0}) \\ (\star) & (\star) & -\text{diag}(P_1, V' P_1 V) & (\mathbf{0}) \\ (\star) & (\star) & (\star) & -I \end{bmatrix} < 0; \forall i. \quad (2.2.10)$$

Ce qui achève la démonstration de la première partie du théorème. Il reste à vérifier la stabilité asymptotique du système réduit lorsque  $u_k = 0$ .

Soit  $S_k = \zeta' V' P_1 V \zeta > 0$  la fonction candidate de Lyapunov associée au système (2.2.1), alors pour  $u_k = 0$ , on aura,

$$\begin{aligned}
S_{k+1} - S_k &= \zeta'_{k+1} V' P_1 V \zeta_{k+1} - \zeta'_k V' P_1 V \zeta_k \\
&= (A_r \zeta_k + B_f f(W_r \zeta_k))' V' P_1 V (A_r \zeta_k + B_f f(W_r \zeta_k)) - \zeta'_k V' P_1 V \zeta_k \\
&= [(V' P_1 V)^{-1} V' P_1 A V \zeta_k + (V' P_1 V)^{-1} V' P_1 f(W_r \zeta_k)]' V' P_1 V [(V' P_1 V)^{-1} V' P_1 A V \zeta_k \\
&\quad + (V' P_1 V)^{-1} V' P_1 f(W_r \zeta_k)] - \zeta'_k V' P_1 V \zeta_k \\
&= [\zeta'_k V' A' P_1 V (V P_1 V')^{-1} + f'(W_r \zeta_k) P_1 V (V P_1 V')^{-1}] [V' P_1 A V \zeta_k \\
&\quad + V' P_1 f(W_r \zeta_k)] - \zeta'_k V' P_1 V \zeta_k \\
&= [f'(W_r \zeta_k) + \zeta'_k V' A'] P_1 V P_2^{-1} V' P_1 [f(W_r \zeta_k) + A V \zeta_k] - \zeta'_k V' P_1 V \zeta_k
\end{aligned} \tag{2.2.11}$$

où  $P_2 = V' P_1 V$ , et comme  $P_1 > 0$  (définition d'une équation quadratique) alors par le principe de congruence

$$\begin{bmatrix} -P_1 & P_1 V \\ (\star) & -V' P_1 V \end{bmatrix} = \begin{bmatrix} I & (\mathbf{0}) \\ (\mathbf{0}) & V' \end{bmatrix} \begin{bmatrix} -P_1 & P_1 \\ P_1 & -P_1 \end{bmatrix} \begin{bmatrix} I & (\mathbf{0}) \\ (\mathbf{0}) & V \end{bmatrix} \leq 0. \tag{2.2.12}$$

Alors, par le lemme de Schur on aura,

$$P_1 V P_2^{-1} V' P_1 - P_1 \leq 0 \Leftrightarrow P_1 V P_2^{-1} V' P_1 \leq P_1.$$

On remplace dans (2.2.11) alors,

$$S_{k+1} - S_k \leq [f'(W_r \zeta_k) + \zeta'_k V' A'] P_1 [f(W_r \zeta_k) + A V \zeta_k] - \zeta'_k V' P_1 V \zeta_k. \tag{2.2.13}$$

D'autre part on a  $f(0) = 0$  et la jacobienne du système non linéaire (2.2.2) peut s'écrire comme  $\frac{\partial f}{\partial s}(s) = \sum_{i=1}^{\mu} \alpha_i(s) F_i$ ; où  $\alpha_i > 0$  et  $\sum_{i=1}^{\mu} \alpha_i = 1$ ,  $\forall s \in \mathbb{R}^n$ , alors en notant  $s_\lambda = \lambda V \zeta_k$ , on aura

$$f(V \zeta_k) = \int_0^1 \frac{\partial f(s)}{\partial s} \Big|_{s=s_\lambda} V \zeta_k d\lambda = \int_0^1 \sum_{i=1}^{\mu} \alpha_i(s_\lambda) F_i V \zeta_k d\lambda, \int_0^1 \sum_{i=1}^{\mu} \alpha_i(s_\lambda) d\lambda = 1. \tag{2.2.14}$$

Remplaçons (2.2.14) dans (2.2.13) alors,

$$S_{k+1} - S_k \leq \left[ \left( \int_0^1 \sum_{i=1}^{\mu} \alpha_i(s) F_i V \zeta_k d\lambda \right)' + \zeta'_k V' A' \right] P_1 \left[ \left( \int_0^1 \sum_{i=1}^{\mu} \alpha_i(s) F_i V \zeta_k d\lambda \right) + A V \zeta_k \right] - \zeta'_k V' P_1 V \zeta_k, \tag{2.2.15}$$

ce qui est équivalent aussi à

$$S_{k+1} - S_k \leq \zeta'_k V' \left[ \int_0^1 \left( \sum_{i=1}^{\mu} \alpha_i(s) F_i + A \right) d\lambda \right]' P_1 \left[ \int_0^1 \left( \sum_{i=1}^{\mu} \alpha_i(s) F_i + A \right) d\lambda \right] V \zeta_k - \zeta'_k V' P_1 V \zeta_k. \tag{2.2.16}$$

Appliquons maintenant le résultat du **lemme 3** sur l'inégalité précédente, et posons  $\rho_k = V\zeta_k$ , on aura,

$$S_{k+1} - S_k \leq \rho_k' \left( \int_0^1 \left( \left[ \sum_{i=1}^{\mu} \alpha_i(s) F_i + A \right]' P_1 \left[ \sum_{i=1}^{\mu} \alpha_i(s) F_i + A \right] - P_1 \right) d\lambda \right) \rho_k. \quad (2.2.17)$$

La condition suffisante pour que soit  $S_{k+1} - S_k < 0$  est donné par

$$\left[ \sum_{i=1}^{\mu} \alpha_i(s) F_i + A \right]' P_1 \left[ \sum_{i=1}^{\mu} \alpha_i(s) F_i + A \right] - P_1 < 0, \quad (2.2.18)$$

ce qui est équivalent par le lemme de Schur à

$$\sum_{i=1}^{\mu} \alpha_i(s) \begin{bmatrix} -P_1 & (A + F_i)' P_1 \\ (\star) & -P_1 \end{bmatrix} < 0, \forall i. \quad (2.2.19)$$

D'autre part on a de l'inégalité (2.2.10) que

$$(A + F_i)' P_1 (A + F_i) - P_1 < 0; 1 \leq i \leq \mu,$$

ce qui implique par le lemme de Schur que l'inégalité (2.2.18) est vérifiée, par conséquent le système non linéaire d'ordre réduit (2.2.1) est asymptotiquement stable pour  $u_k = 0$ .

**Remarque 6.** *L'inconvénient de cette méthode est que il n'y a pas une méthode directe pour calculer la matrice  $V$  qui satisfait l'ensemble des LMIs du **théorème 7** et qui rend  $\gamma$  plus petit. Mais dans le cas où  $V$  est donné alors, on vient de voir qu'il est possible de faire un choix optimal de  $U$ .*

### Le choix de la matrice $V$

- Dans le cas des systèmes linéaire, on peut choisir la matrice  $V$  comme les bases orthonormales du sous espace de Krylov  $V = [B \ AB \ A^2B \ \dots \ A^r B]$  où  $r$  est l'ordre de système réduit.
- Dans le cas des systèmes non linéaire, on résout le problème d'optimisation

$$\min_V = \sum_{k=1}^N (y_k - CV\zeta_k)' (y_k - CV\zeta_k), t.q. V'V = I,$$

donc  $V$  est une matrice à plein rang, et la minimisation du problème précédent revient à minimiser

$$\min_V = \sum_{k=1}^N (y_k - CVV'x_k)' (y_k - CVV'x_k), t.q. VV' = I$$

## 2.3 Exemple d'application

Dans ce paragraphe on se propose d'appliquer la méthode de réduction présentée dans ce chapitre sur un modèle non linéaire d'un système dynamique SMD (Spring, Mass, Damping) d'ordre 2 composé par des ressorts, amortisseurs et masses et soumis à un frottement non linéaire.

### 2.3.1 Système dynamique SMD (Spring, Mass, Damping)

Les vibrations sont des oscillations mécaniques autour d'une position d'équilibre. Dans certains cas les vibrations sont désirables, comme certaines machines industrielles ou les lignes de productions (filtrations). Mais dans la plus part de temps, les vibrations sont indésirables (la perte d'énergie, limiter la capacité, les dangers...), par exemple lorsque les ailes d'un avion vibrent, les voyages deviennent inconfortable quand la fréquence des vibrations est plus grande que celle du corps de l'être humain (4-8 Hz).

#### Modélisation et discrétisation

Considérons le système mécanique représenté dans la **Figure 2.1**,

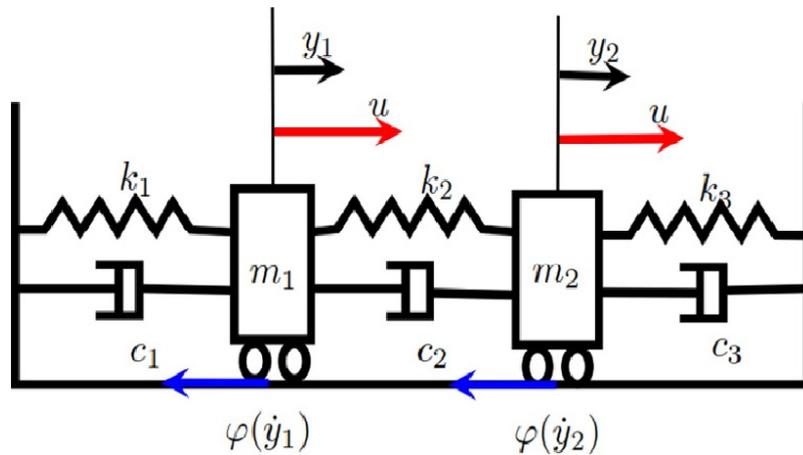


Figure 2.1 - Schéma d'un système mécanique MRA (Masse, Ressort, Amortisseur).

En appliquant la deuxième loi de Newton  $\sum F_i = M\ddot{q}$  (où  $F_i$  sont les forces externes exercés sur le système mécanique précédent et  $\ddot{q}$  est l'accélération) alors, les lois de la physique nous donnent les équations du mouvement

$$\begin{aligned} m_1\ddot{q}_1 + (c_1 + c_2)\dot{q}_1 - c_2\dot{q}_2 + (k_1 + k_2)q_1 - k_2q_2 &= u - \varphi_1 \\ m_2\ddot{q}_2 - c_2\dot{q}_1 + (c_2 + c_3)\dot{q}_2 - k_2q_1 + (k_2 + k_3)q_2 &= u - \varphi_2. \end{aligned} \quad (2.3.1)$$

Ce qui est équivalent aussi a la forme matricielle suivante

$$\begin{bmatrix} m_1 & 0 \\ 0 & m_2 \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \begin{bmatrix} c_1 + c_2 & -c_2 \\ -c_2 & c_2 + c_3 \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} + \begin{bmatrix} k_1 + k_2 & -k_2 \\ -k_2 & k_2 + k_3 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} u - \varphi_1 \\ u - \varphi_2 \end{bmatrix} \quad (2.3.2)$$

On fait le changement de variable suivant

$$\begin{cases} x_1 = q_1 \\ x_2 = \dot{q}_1 \\ x_3 = q_2 \\ x_4 = \dot{q}_2 \end{cases},$$

alors le système précédent (2.3.2) est équivalent à

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k_1+k_2}{m_1} & -\frac{c_1+c_2}{m_1} & \frac{k_2}{m_1} & \frac{c_2}{m_1} \\ 0 & 0 & 0 & 1 \\ \frac{k_2}{m_2} & \frac{c_2}{m_2} & -\frac{k_2+k_3}{m_2} & -\frac{c_2+c_3}{m_2} \end{bmatrix} x + \begin{bmatrix} 0 \\ -\frac{\varphi(x_2)}{m_1} \\ 0 \\ -\frac{\varphi(x_4)}{m_2} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m_1} \\ 0 \\ \frac{1}{m_2} \end{bmatrix} u. \quad (2.3.3)$$

Pour simplifier les calculs, nous supposons que

$$m_1 = m_2 = k_1 = k_2 = k_3 = 1, c_1 = c_2 = 14.5, c_3 = 10 \text{ et}$$

$$\varphi(s) = 2 \frac{1 - e^{-s}}{1 + e^{-s}},$$

$u \in \mathbb{R}$  est la variable d'entrée appliquée à la masse  $m_1$  et  $m_2$ , avec

$$u(t) = 13t \cos(2t)e^{-t} + \cos(2t).$$

$\varphi(x_2), \varphi(x_4)$  sont les forces de frottements non linéaires qui s'opposent au déplacement des masses  $m_1$  et  $m_2$  respectivement.

La modélisation de ce problème mécanique nous donne une classe de systèmes non linéaires à temps continu

$$\dot{x} = A_c x + f_c(x) + B_c u,$$

$$y = Cx,$$

$$\text{où, } A_c = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k_1+k_2}{m_1} & -\frac{c_1+c_2}{m_1} & \frac{k_2}{m_1} & \frac{c_2}{m_1} \\ 0 & 0 & 0 & 1 \\ \frac{k_2}{m_2} & \frac{c_2}{m_2} & -\frac{k_2+k_3}{m_2} & -\frac{c_2+c_3}{m_2} \end{bmatrix}, B_c = \begin{bmatrix} 0 \\ \frac{1}{m_1} \\ 0 \\ \frac{1}{m_2} \end{bmatrix}, C = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, f_c = \begin{bmatrix} 0 \\ -\frac{\varphi(x_2)}{m_1} \\ 0 \\ -\frac{\varphi(x_4)}{m_2} \end{bmatrix}. \quad (2.3.4)$$

La discrétisation par la méthode d'Euler pour  $T = 10^{-2}(\text{sec})$  nous donne une classe des systèmes non linéaires à temps discret

$$\begin{cases} \dot{x}_k = Ax_k + f(x_k) + Bu_k, \\ y_k = Cx_k, \end{cases} \quad (2.3.5)$$

avec,  $A = I + TA_c$ ,  $B = TB_c$ , et  $f = Tf_c$

où,

$$A = \begin{bmatrix} 1 & 0.01 & 0 & 0 \\ -0.02 & 0.71 & 0.01 & 0.145 \\ 0 & 0 & 1 & 0.01 \\ 0.01 & 0.145 & -0.02 & 0.755 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0.01 \\ 0 \\ 0.01 \end{bmatrix}, C = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, f = 10^{-2} \begin{bmatrix} 0 \\ -\varphi(x_2) \\ 0 \\ -\varphi(x_4) \end{bmatrix}.$$

La jacobienne de  $f(x_k) = T f_c(x_k)$ , noté par  $\frac{\partial f}{\partial x}(x_k)$ , est donnée par

$$\frac{\partial f}{\partial x}(x_k) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \frac{-T}{m_1} \varphi'(x_2(k)) & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{-T}{m_2} \varphi'(x_4(k)) \end{bmatrix},$$

où  $\varphi'$  ici désigne la dérivée de  $\varphi$ .

Cette dernière expression peut se décomposer comme suit :

$$\frac{\partial f}{\partial x_k}(x_k) = \text{diag}(0, \theta_1(x_2(k)), 0, 0) + \text{diag}(0, 0, 0, \theta_2(x_4(k)))$$

$$\text{où } \theta_1(x_2(k)) = \frac{-T}{m_1} \varphi'(x_2(k)) \text{ et } \theta_2(x_4(k)) = \frac{-T}{m_2} \varphi'(x_4(k))$$

En utilisant la bornitude de  $\varphi'$  ( $0 \leq \varphi' \leq 2$ ), on déduit que la matrice jacobienne  $\frac{\partial f}{\partial x_k}(x_k)$  appartient à l'ensemble convexe dont les sommets sont :

$$F_1 = \text{diag}(0, -2T, 0, -2T), F_2 = \text{diag}(0, -2T, 0, 0), F_3 = \text{diag}(0, 0, 0, -2T), F_4 = \text{diag}(0, 0, 0, 0)$$

## Simulation et comparaison

Pour tester la faisabilité des différentes méthodes de la réduction des systèmes non linéaire, on propose des programmes sur MATLAB pour simuler les LMI données et calculer les matrices du système réduit.

### Approximation par un modèle linéaire (théorème 6)

Pour la réduction de système (2.3.5) à l'ordre  $r = 2$  on pose  $E = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$

```
%%%%%%%%%% ----- %%%%%%%%%%
%%%%%%%%%% La réduction par un système linéaire %%%%%%%%%%
%%%%%%%%%% d'ordre réduit (théorème 6) sous MATLAB %%%%%%%%%%
%%%%%%%%%% ----- %%%%%%%%%%
```

```
clear all;
close all;
```

```
%%%%%%%%% Déclaration des constantes %%%%%%%%%%
```

```
T=0.01; %% pas de discrétisation par la méthode d'Euler %%%%%%%%%%
```

```

c1=14.5;
c2=c1;
c3=10;
k1=1;k2=1;k3=1;m1=1;m2=1;
Ac=[0 1 0 0;
    -(k1+k2)/m1 -(c1+c2)/m1 k2/m1 c2/m1;
    0 0 0 1;
    k2/m2 c2/m2 -(k1+k2)/m2 -(c2+c3)/m2];
Bc=[0;1/m1;0;1/m2];
C=[0 1 0 0];

F_1=[0 0 0 0;
     0 -0.02 0 0;
     0 0 0 0;
     0 0 0 -0.02];

F_2=[0 0 0 0;
     0 -0.02 0 0;
     0 0 0 0;
     0 0 0 0];

F_3=[0 0 0 0;
     0 0 0 0;
     0 0 0 0;
     0 0 0 -0.02];

F_4=[0 0 0 0;
     0 0 0 0;
     0 0 0 0;
     0 0 0 0];

F=horzcat(F_1, F_2, F_3, F_4);

r=2;
n=size(Ac,1);
m=size(Bc,2);
p=size(C,1);
A=eye(n)+T*Ac
B=T*Bc

%%%%%%%% Déclaration de la matrice E %%%%%%%%%%
E=[eye(r); zeros(n-r)];

%%%%%%%%%Déclaration des variables%%%%%%%%%
P=sdpvar(n+r); %% déclaration de la matrice P réalisant le minimum %%%
X=sdpvar(r);
Y=sdpvar(r,m);
Cr=sdpvar(p,r);
G11=sdpvar(n);
G12=sdpvar(n);

```

```

G13=sdpvar(n);
G14=sdpvar(n);
G2=sdpvar(r);
G31=sdpvar(r,n);
G32=sdpvar(r,n);
G33=sdpvar(r,n);
G34=sdpvar(r,n);
G1i =[G11 G12 G13 G14];
G3i =[G31 G32 G33 G34];
delta=sdpvar(1);%% déclaration de la valeur delta à minimiser %%

```

```

%%%%%%%%%%%% Déclaration des contraintes %%%%%%%%%%%%%
fct=set(P>0);
opt=delta; %% comande pour minimiser gamma^2 %%

```

```

%%%%%%%%%%%% la construction de LMI %%%%%%%%%%%%%

```

```

for i=1:4
f=F(:,(i-1)*n+1:i*n)
G=[G1i(:,(i-1)*n+1:i*n) E*G2;
   G3i(:,(i-1)*n+1:i*n) G2 ];
G1=G1i(:,(i-1)*n+1:i*n);
G3=G3i(:,(i-1)*n+1:i*n);
fct=fct+set(zeros<delta);
k11=-P;
k12=zeros(n+r,m);
k13=[(A+f)'*G1' (A+f)'*G3';
     X'* E' X'];
k14=[C -Cr]';
k22=(-delta)*eye(m);
k23=[B'*G1'+(Y)'*E' B'*G3'+(Y)'];
k24=zeros(m,p);
k33=G'+G+P;
k34=zeros(n+r,p);
k44=-eye(p);

```

```

LMI=[k11 k12 k13 k14;
     k12' k22 k23 k24;
     k13' k23' k33 k34;
     k14' k24' k34' k44];

```

```

fct=fct+set(LMI<0);
end

```

```

%%%%%%%%%%%% Résolution du problème d'optimisation sous la contrainte LMI %%%%%%%%%%%%%

```

```

solvesdp(fct,opt);

```

```

%%%%%%%%%%%% récupération des variables %%%%%%%%%%%%%

```

```

X=double(X)
Y=double(Y)

```

```

Cr=double(Cr)
G2=double(G2)
P=double(P)
LMI=double(LMI);
delta=double(delta);
G1i=double(G1i);
G3i=double(G3i);
%%%%%%%%%%%% récupération des matrices G1(i) et G3(i) %%%%%%%%%%
for i=1:4
G1_i=G1i(:,(i-1)*n+1:i*n);
G3_i=G3i(:,(i-1)*n+1:i*n);
end

%%%%%%%%%%%% Test de faisabilité de la LMI %%%%%%%%%%%%%%
vp_LMI=eig(LMI);
gamma=sqrt(delta) %% récupération de la valeur gamma %%%%%%%%%%

%%%%%%%%%%%% Constriction des matrices de système réduit %%%
Ar=inv(G2)*X;
Br=inv(G2)*Y;
Cr=Cr;

%%%%%%%%%%%% Tracé des graphes %%%%%%%%%%%%%%
N=size(Ar,1);
M=size(Br,2);
Phat=size(Cr,1);

Horiz=100;
x=zeros(n,Horiz+1);
y1=zeros(p,Horiz);

xhat=zeros(N,Horiz+1);
y2=zeros(Phat,Horiz);

%%%%%%%%%%%% Initialisation des parametres du système origine %%%
u=zeros(m,Horiz);
phi=zeros(m,Horiz);
x(:,1)=[0; 0; 0; 0];
phi(:,1)=0;

%%%%%%%%%%%% Initialisation des parametres du système réduit %
uhat=zeros(M,Horiz);
phihat=zeros(M,Horiz);
xhat(:,1)=[0; 0];
phihat(:,1)=0;

for t=1:Horiz
    u(:,t)=cos(2*(x(1,t)))*(13*t*exp(-(x(1,t)))+1);
    phi(:,t)=-2*(1-exp(-(x(1,t))))/(1+exp(-(x(1,t))));
    x(:,t+1)=A*x(:,t)+B*(u(:,t)+ phi(:,t));
    y1(:,t)=C*x(:,t);

```

```

    uhat(:,t)=cos(2*(x(1,t)))*(13*t*exp(-(x(1,t)))+1);
    phihat(:,t)=-2*(1-exp(-(x(1,t))))/(1+ exp(-(x(1,t))));
    xhat(:,t+1)=Ar*xhat(:,t)+Br*(uhat(:,t));
    y2(:,t)=Cr*xhat(:,t);
end

%%%%%%%%%%%% Affichages des figures des deux système %%%%%%%%%%
figure(1)

plot(y1(1:Horiz),'--b');
hold on;
plot(y2(1:Horiz),'r');
legend('origine','réduit');
xlabel('temps');
title('réduction pour r = 2');
grid;
%%%%%%%%%%%% fin du programme %%%%%%%%%%

```

En utilisant MATLAB, on a trouvé la matrice P réalisant le minimum et approxime le système non linéaire d'ordre 4 par un système linéaire d'ordre 2, et les matrices de nouveau système.

```

Result: feasible solution of required accuracy
        best objective value: 3.041003e-005
        guaranteed absolute accuracy: 2.89e-008
        f-radius saturation: 0.000% of R = 1.00e+009

```

X =

```

-7.4063  -1.0321
-1.0321  -6.6351

```

Y =

```

0.0107
0.0741

```

Cr =

```

0.0005  -1.0000

```

G2 =

```

-7.4171  -1.1059
-1.1059  -7.6461

```

P =

14.1750	0.8055	-6.0686	0.2097	7.4030	1.0079
0.8055	6.7426	-0.1729	-1.7263	0.6224	4.7996
-6.0686	-0.1729	5.4768	0.1399	0.0013	0.0099
0.2097	-1.7263	0.1399	3.7158	0.3450	2.3872
7.4030	0.6224	0.0013	0.3450	7.4026	1.0057
1.0079	4.7996	0.0099	2.3872	1.0057	7.4548

gamma = 0.0055

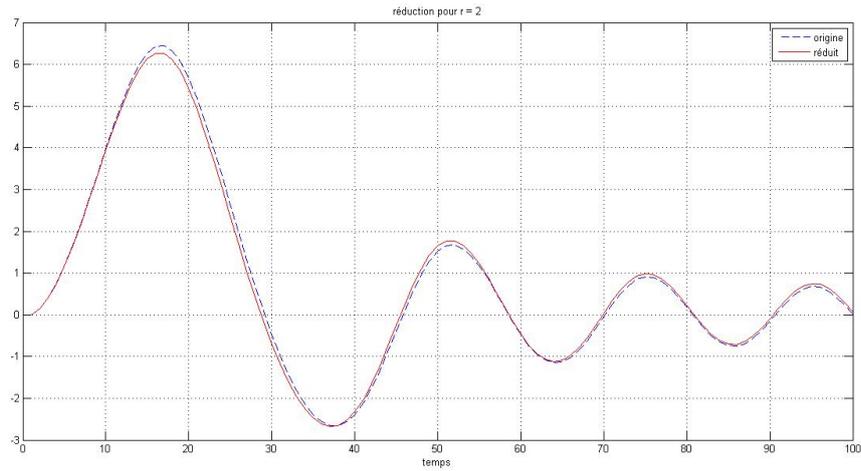


Figure 2.2 - Approximation par un système linéaire  $r = 2$ .

### Approximation par un système non linéaire d'ordre réduit (théorème 7)

Pour la réduction de système (2.3.5) à l'ordre  $r = 3$  on pose  $V = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$ , et pour  $r = 2$  on prend

$V = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 1 & 1 \end{bmatrix}$ , dans le choix de S. Ibrir et M. Bettayab [1].

Dans notre choix, comme les sous systèmes (2.1.2) sont supposés linéaires, alors on peut prendre la matrice  $V$  comme les bases orthonormales des sous espace de Krylov, i.e.  $V = [B \ AB \ \dots \ A^r B]$ , où  $r$  est l'ordre de système réduit.

```
%%%%%%%%%% ----- %%%%%%%%%%
%%%%%%%%%% La réduction par un système non linéaire %%%%%%%%%%
%%%%%%%%%% d'ordre réduit (théorème 7) sous MATLAB %%%%%%%%%%
%%%%%%%%%% ----- %%%%%%%%%%
```

```
clear all;
close all;
```

```
%%%%%%%%% Déclaration des constantes %%%%%%%%%%
```

```
T=0.01; %% pas de discrétisation par la méthode d'Euler%%%%%%%%%
```

```

c1=14.5;
c2=c1;c3=10;
k1=1;k2=1;k3=1;m1=1;m2=1;

Ac=[0 1 0 0;
    -(k1+k2)/m1 -(c1+c2)/m1 k2/m1 c2/m1;
    0 0 0 1;
    k2/m2 c2/m2 -(k1+k2)/m2 -(c2+c3)/m2];

Bc=[0;1/m1;0;1/m2];

C=[0 1 0 0];

F_1=[0 0 0 0;
     0 -0.02 0 0;
     0 0 0 0;
     0 0 0 -0.02];

F_2=[0 0 0 0;
     0 -0.02 0 0;
     0 0 0 0;
     0 0 0 0];

F_3=[0 0 0 0;
     0 0 0 0;
     0 0 0 0;
     0 0 0 -0.02];

F_4=[0 0 0 0;
     0 0 0 0;
     0 0 0 0;
     0 0 0 0];

F=horzcat(F_1, F_2, F_3, F_4);

n=size(Ac,1);
m=size(Bc,2);
p=size(C,1);
A=eye(n)+T*Ac;
B=T*Bc;

%%%%%%%% Déclaration de la matrice E %%%%%%%%%
V=[1 0 1;
   0 1 0;
   0 0 1;
   1 1 0];
r=size(V,2);

%%%%%%%% Déclaration des variables %%%%%%%%%

P=sdpvar(n);% déclaration de la matrice P réalisant le minimum %%%

```

```

delta=sdpvar(1);% déclaration de la valeur delta à minimiser %

%%%%%%%%%%%% Déclaration des contraintes %%%%%%%%%%%%%
fct=set(P>0);
opt=delta; %% comande pour minimiser gamma^2 %%

%%%%%%%%%%%% la construction de LMI %%%%%%%%%%%%%

for i=1:4
    for j=1:4
f_i=F(:,(i-1)*n+1:i*n);
f_j=F(:,(j-1)*n+1:j*n);
fct=fct+set(zeros<delta);

k11=[-P zeros(n,r);
      (zeros(n,r))' -V'*P*V];
k12=zeros(n+r,m);
k13=[(A+f_i)'*P zeros(n,r);
      (zeros(n,r))' V'*(A+f_j)'*P*V];
k14=[C -C*V]';
k22=(-delta)*eye(m);
k23=[B'*P B'*P*V];
k24=zeros(m,p);
k33=011;
k34=zeros(n+r,p);
k44=-eye(p);

LMI=[k11 k12 k13 k14;
      k12' k22 k23 k24;
      k13' k23' k33 k34;
      k14' k24' k34' k44];
fct=fct+set(LMI<0);
    end
end

%%%%%%%%%%%% Résolution de la LMI %%%%%%%%%%%%%

solvesdp(fct,opt);

%%%%%%%%%%%% récupération des variables %%%%%%%%%%%%%

P=double(P)
delta=double(delta);
LMI=double(LMI);
gamma=sqrt(delta) %% récupération de la valeur gamma %%%%%%%%%

%%%%%%%%%%%% Test de faisabilité de la LMI %%%%%%%%%%%%%
vp_LMI=eig(LMI);

%%%%%%%%%%%% Constriction des matrices de système réduit %%%%%%%%%
Ar=inv(V'*P*V)*V'*P*A*V

```

```

Br=inv(V'*P*V)*V'*P*B
Bf=inv(V'*P*V)*V'*P
Cr=C*V

%%%%%%%%%% Traser les graphes %%%%%%%%%%%
N=size(Ar,1);
M=size(Br,2);
Phat=size(Cr,1);

Horiz=100;
x=zeros(n,Horiz+1);
y1=zeros(p,Horiz);

xhat=zeros(N,Horiz+1);
y2=zeros(Phat,Horiz);

%%%%%%%%%% Initialisation des parametres du système origine %%%
u=zeros(m,Horiz);
phi=zeros(m,Horiz);
x(:,1)=[0; 0; 0; 0];
phi(:,1)=0;

%%%%%%%%%% Initialisation des parametres du système réduit %
uhat=zeros(M,Horiz);
phihat=zeros(M,Horiz);
xhat(:,1)=[0; 0; 0];
phihat(:,1)=0;

for t=1:Horiz
    u(:,t)=cos(2*(x(1,t)))*(13*t*exp(-(x(1,t)))+1);
    phi(:,t)=-2*(1-exp(-(x(1,t))))/(1+exp(-(x(1,t))));
    x(:,t+1)=A*x(:,t)+B*(u(:,t)+ phi(:,t));
    y1(:,t)=C*x(:,t);

    uhat(:,t)=cos(2*(x(1,t)))*(13*t*exp(-(x(1,t)))+1);
    phihat(:,t)=-2*(1-exp(-(x(1,t))))/(1+exp(-(x(1,t))));
    xhat(:,t+1)=Ar*xhat(:,t)+Br*(uhat(:,t)+ phihat(:,t));
    y2(:,t)=Cr*xhat(:,t);
end

%%%%%%%%%% Affichages des figures des deux système %%%%%%%%%%%
figure(1)
plot(y1(1:Horiz),'b');
hold on;
plot(y2(1:Horiz),'--r');
legend('origine','réduit');
xlabel('temps');
title('réduction pour r = 2');
grid;
%%%%%%%%%%_____ fin du programme_____ %%%%%%%%%%%

```

En utilisant MATLAB pour résoudre le problème de LMI précédent, on a trouvé la matrice P réalisant

le minimum et approxime le système non linéaire d'ordre 4 par un système non linéaire d'ordre inférieur.

### Approximation par système non linéaire d'ordre réduit [1]

Pour  $r = 3$ ,

Result: feasible solution of required accuracy  
best objective value: 6.972665e-003  
guaranteed absolute accuracy: 3.35e-006  
f-radius saturation: 0.000% of  $R = 1.00e+009$

P =

135.2565	2.0179	-114.7414	-1.8670
2.0179	32.6129	-1.7524	-24.5521
-114.7414	-1.7524	103.8367	1.6866
-1.8670	-24.5521	1.6866	24.8155

gamma = 0.0835

Ar =

0.9180	0.0093	-0.0000
0.1159	0.8562	-0.0100
0.1691	-0.0105	1.0000

Br =

0.0000
0.0100
-0.0000

Bf =

0.7942	-0.2058	-0.7942	0.2058
-0.0260	0.9740	0.0260	0.0260
0.4545	0.4545	0.5455	-0.4545

Cr =

0	1	0
---	---	---

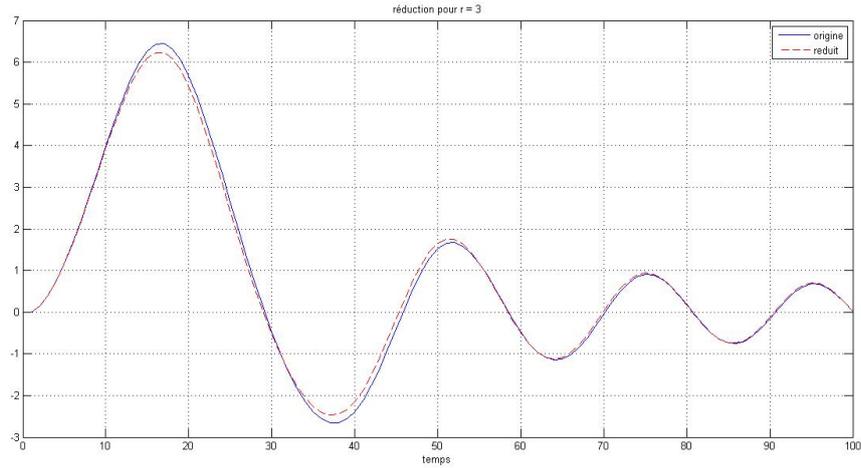


Figure 2.3 - Approximation par un système non linéaire d'ordre 3.

Pour  $r = 2$ ,

Result: feasible solution of required accuracy  
 best objective value: 6.971717e-003  
 guaranteed absolute accuracy: 6.62e-006  
 f-radius saturation: 0.000% of  $R = 1.00e+009$

P =

161.8471	1.9503	-138.5137	-1.7353
1.9503	33.4338	-1.6765	-25.2788
-138.5137	-1.6765	125.0790	1.5510
-1.7353	-25.2788	1.5510	25.4539

gamma = 0.0835

Ar =

0.9461	0.0071
0.1198	0.8557

Br =

0
0.0100

Bf =

0.8710	-0.1290	-0.7451	0.1290
-0.0150	0.9850	0.0198	0.0150

Cr =

0 1

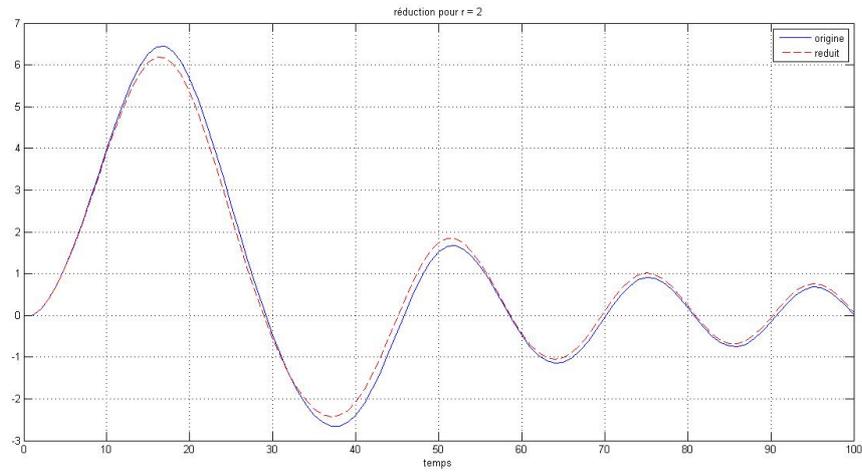


Figure 2.4 - Approximation par un système non linéaire d'ordre 2.

**Approximation par un système non linéaire d'ordre réduit (base de Krylov)**

Pour  $r = 3$ ,

Result: feasible solution

best objective value: 6.980418e-003

guaranteed absolute accuracy: 8.15e-005

f-radius saturation: 0.000% of  $R = 1.00e+009$

P =

142.2553	3.7103	-121.2280	-3.6098
3.7103	36.0576	-3.4288	-27.6662
-121.2280	-3.4288	109.8549	3.4126
-3.6098	-27.6662	3.4126	27.6394

gamma = 0.0835

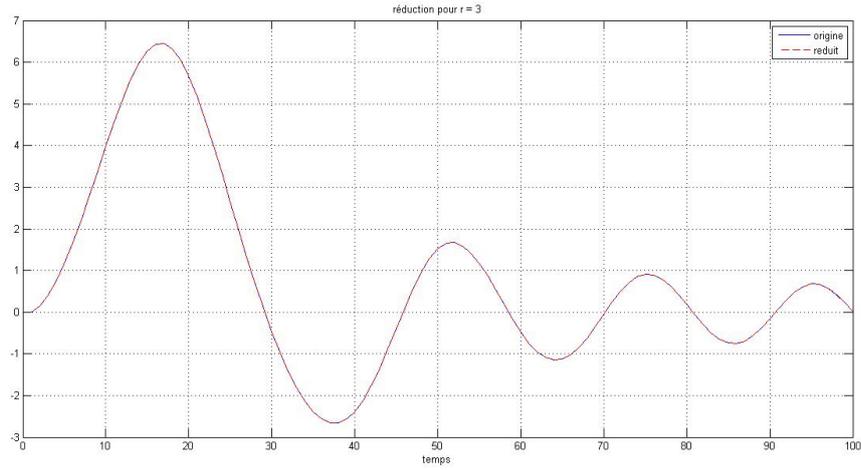


Figure 2.5 - Approximation par un système non linéaire d'ordre 3.

Pour  $r = 2$ ,

P =

145.5117	2.8778	-123.9503	-2.7208
2.8778	34.7009	-2.5922	-26.4298
-123.9503	-2.5922	112.1101	2.5224
-2.7208	-26.4298	2.5224	26.5099

gamma = 0.0835

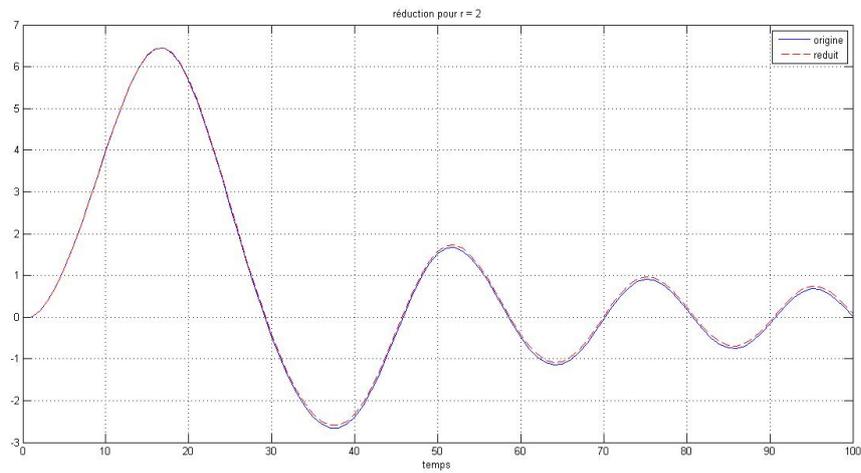


Figure 2.6 - Approximation par un système non linéaire d'ordre 2.

# Conclusion

Dans ce mémoire nous nous sommes intéressés à l'étude d'une approche pour la réduction des systèmes dynamiques, et plus précisément la réduction d'une classe de systèmes non linéaires lipschitzien à temps discret, proposée par S. Ibrir et M. Bettayeb dans [1].

Nous avons vu que le système considéré peut être approché au sens de la norme  $\mathcal{H}_\infty$  par un système linéaire d'ordre réduit en se basant sur le théorème de la valeur moyenne et par un système non linéaire d'ordre réduit en se basant sur les projections de Krylov. Dans les deux cas, nous avons donné des conditions suffisantes permettant de résoudre le problème d'optimisation associé à la réduction des systèmes dynamiques.

De nombreuses questions restent encore posées dans le domaine de réduction des systèmes dynamiques. Nous citons à titre d'exemple,

1. Étude d'autres classes de systèmes non linéaires,
2. Réduire le conservatisme lié à la linéarisation des contraintes BMIs obtenues,
3. Remédier au problème du choix de l'ordre du système,
4. Choix des matrices de Krylov.

Nous souhaitons enfin que notre travail suscite un intérêt et donne lieu à d'autres études dans ce domaine.

# Bibliographie

- [1] S. Ibrir, M. Bettayeb, Model reduction of a class of discrete-time nonlinear systems, *Applied Mathematics and Computation* 250 (2015) 78 - 93.
- [2] S.L. Niculescu, L.E. Ghaoui, Advance in Linear Matrix Inequality Methods in Control, *Studies in Applied Mathematics, SIAM, Philadelphia*, 2000.
- [3] R.D. Robinett III, D.G. Wilson, G.Richard Eisler, J.E. Hurtado, Applied Dynamic Programming for Optimization of Dynamical Systems, *Studies in Applied Mathematics, SIAM, Philadelphia*, 2005.
- [4] Henk Van Der Vorst, Iterative Krylove Methods for Large Linear Systems, *Cambridge University Press*, 2003.
- [5] V. Ionescu, C. Oara, M.D. Weiss, Generalized Riccati Theory and Robust Control : A Popov Function Approach, 1999.
- [6] D. Kavranoglu, M. Bettayeb, Characterization of the solution to the optimal H1 model reduction problem, *Syst. Control Lett*, 20 (2) (1993) 99 - 107.
- [7] M.Sahadet Hossain, P. Benner, Projection-Based Model Reduction for Time-Varying Descriptor Systems : New Results, *Max Planck Institute Magdeburg Preprints*, 2012.
- [8] B.C. Moore, Principal component analysis in linear systems : controllability, observability and model reduction, *IEEE Trans. Autom. Control*, 26 (1)(1981) 17 - 32.
- [9] U.M. Al-Saggaf, G.F. Franklin, An error bound for a discrete reduced order model of a linear multivariable system, *IEEE Trans. Autom. Control*, 32 (9)(1987) 815 - 819.
- [10] P. Benner, M. Hinze, E.J.W. ter Maten (Eds.), Model reduction for circuit simulation, *Lecture Notes in Electrical Engineering, 74, Springer*, 2011.
- [11] S.X.-D. Tan, L. He, Advanced Model Order Reduction Techniques in VLSI Design, *Cambridge University Press*, 2007.
- [12] A.C. Antoulas, Advances in Design and Control : Approximation of Largescale Dynamical Systems, vol. 6, *Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA*, 2005.
- [13] K. Fujimoto, D. Tsubakino, Computation of nonlinear balanced realization and model reduction based on Taylor series expansion, *Syst. Control Lett*, 57 (4) (2008) 283 - 289.
- [14] S. Lall, J.E. Marsden, S. Glavaski, Empirical model reduction of controlled nonlinear systems, in : *IFAC World Congress. International Federation of Automatic Control, New York*, 1999, pp. 473 - 478.
- [15] J. Scherpen, Balancing for nonlinear systems, *Syst. Control Lett*, 21 (2) (1993) 143 - 153.
- [16] J. Scherpen, H1 balancing for nonlinear systems, *Int. J. Robust Nonlinear Control*, 6 (1996) 645 - 668.

- [17] C. Sun, J. Hahn, Reduction of stable differential algebraic equation systems via projections and system identification, *J. Process Control* 15 (6) (2005) 639-650.
- [18] K. Gu, An integral inequality in the stability problem of time-delay systems, *in : Proceedings of the 39th IEEE Conference in Decision and Control*, 2000, pp. 2805 - 2810 (Sydney).
- [19] S. Boyd, L.E. Ghaoui, E. Feron, V. Balakrishnan, Linear Matrix Inequality in Systems and Control Theory, *Studies in Applied Mathematics, SIAM, Philadelphia*, 1994.
- [20] C.E. de Souza, L. Xie, On the discrete-time bounded real lemma with application in the characterization of static state feedback  $H_\infty$  controllers, *Control Lett*, 18 (1) (1992) 61 - 71.
- [21] J.M. Ortega, W.C. Rheinboldt, Iterative solution of nonlinear equations in several variables, *Philadelphia*, 2000.
- [22] J.P. Aubin, A. Cellina, Differential inclusions : set-valued maps and viability theory, *Grundlehren Der Mathematischen Wissenschaften*, vol. 264, Springer London, Limited, 2012.
- [23] M.C. de Oliveira, J. Bernussou, J.C. Geromel, A new discrete-time robust stability condition, *Syst. Control Lett*, 37 (4) (1999) 261-265.
- [24] S. Gugercin, An iterative SVD-Krylov based method for model reduction of large-scale dynamical systems, *in : Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference, Seville, Spain*, 2005, pp. 5905 - 5910.
- [25] A. Varga, Balancing free square-root algorithm for computing singular perturbation approximations, *in : Proceedings of the 30th Conference on Decision and Control, Brighton, England*, 1991.
- [26] R. Jean-Pierre, Mathématiques pour les systèmes dynamiques, *Paris, Hermès Science Publications*, 2002.
- [27] A. Quarteroni, R. Sacco, F. Saleri, Méthodes Numériques : Algorithmes, analyse et applications, *Springer - Verlag Italia, Milano*, 2004.
- [28] A. Bouzitouna, Sur Quelques Méthodes Itératives Appliquées à Certains Problèmes Inverses, *Université Badji-Mokhtar- Annaba*, 2014.
- [29] A. Touhami, Utilisation Des Filtres De Tchebychev et Construction De Pré-conditionneurs Spectraux Pour L'accélération Des Méthodes De Krylov, *L'institut National Polytechnique De Toulouse*, 2005.
- [30] Eric Magarotto, Approche LMI pour la Synthèse de Correcteurs et d'Observateurs : Application aux Systèmes Spatiaux et Bilinéaires, *Université de Caen*, 2004.
- [31] H. Kheloufi, Contribution à la commande basée sur un observateur pour les systèmes dynamiques incertains, *thèse de doctorat, UMMTO* 2015.
- [32] A. Zemouche, M. Boutayeb, On LMI conditions to design observers for Lipschitz nonlinear systems *Automatica*, 49(2) :585 - 591, 2013.