

BERNARD DOYON

**De l'importance des orbites périodiques :  
Détection et applications**

Thèse  
présentée  
à la Faculté des études supérieures  
de l'Université Laval  
pour l'obtention  
du grade de Philosophiae Doctor (Ph.D.)

Département de physique  
FACULTÉ DES SCIENCES ET DE GÉNIE  
UNIVERSITÉ LAVAL  
QUÉBEC

Octobre 2002

# Résumé

L'ensemble des Orbites Périodiques Instables (OPIs) d'un système chaotique est intimement relié à ses propriétés dynamiques. À partir de l'ensemble (en principe infini) d'OPIs cachées dans l'espace des phases, on peut obtenir des quantités dynamiques importantes telles les exposants de Lyapunov, la mesure invariante, l'entropie topologique et la dimension fractale. En chaos quantique (i.e. l'étude de systèmes quantiques qui ont un équivalent *chaotique* dans la limite classique), ces mêmes OPIs permettent de faire le pont entre le comportement classique et quantique de systèmes non-intégrables.

La localisation de ces cycles fondamentaux est un problème complexe. Cette thèse aborde dans un premier temps le problème de la détection des OPIs dans les systèmes chaotiques. Une étude comparative de deux algorithmes récents est présentée. Nous approfondissons ces deux méthodes afin de les utiliser sur différents systèmes dont des flots continus dissipatifs et conservatifs. Une analyse du taux de convergence des algorithmes est aussi réalisée afin de dégager les forces et les limites de ces schèmes numériques.

Les méthodes de détection que nous utilisons reposent sur une transformation particulière de la dynamique initiale. Cette astuce nous a inspiré une méthode alternative pour cibler et stabiliser une orbite périodique quelconque dans un système chaotique. Le ciblage est en général combiné aux méthodes de contrôle pour stabiliser rapidement un cycle donné. En général, il faut connaître la position et la stabilité du cycle en question. La nouvelle méthode de ciblage que nous présentons ne demande pas de connaître *a priori* la position et la stabilité des orbites périodiques. Elle pourrait être un outil complémentaire aux méthodes de ciblage et de contrôle actuelles.

# Avant-Propos

L'accomplissement de ce travail de recherche a demandé encouragements et support de plusieurs personnes de mon entourage. Je souhaite maintenant prendre quelques lignes pour leur témoigner mon appréciation.

Je tiens d'abord à remercier Louis Dubé pour ses bons conseils et la confiance qu'il m'a accordée. Sa passion pour la recherche, son souci du détail ainsi que sa grande culture scientifique ont été pour moi une source de motivation importante tout au long du projet. Bien que nous ayons eu à travailler à distance à quelques reprises, il a toujours su comment être attentif à mes questionnements et ouvert aux idées nouvelles. L'analyse et le recul qu'il prenait face à cette recherche m'ont souvent amené à y voir plus clair et à reprendre confiance en ce qui avait été accompli.

Des remerciements aussi à M. Jean-Jacques Gervais, M. Gabriel Bédard et M. Reinhold Blümel pour la lecture et la correction de ce manuscrit. Leurs nombreux conseils et commentaires ont contribué sans aucun doute à améliorer la version finale du travail.

Un merci à mes collègues de travail, Greg, Gabrielle, Philippe, Frédéric et Babak. Même si nos interactions ont été de courte durée, elles ont été, du moins pour ma part, bien appréciées...

Une mention aussi doit aller à mes parents, Henri et Céline, beaux-parents, Hugues et Pierrette et ami(e)s (pour n'en nommer que quelques uns : Michel, Philippe, Carl, Daniel, Chantal, Anik et Sébastien) qui m'ont tous offert de bonnes doses de ressources (particulièrement par les bons repas du samedi et dimanche, les randos de ski et les séances de musique).

Enfin, je ne peux passer sous silence la contribution de Johanne dans ce travail. En plus d'une lecture attentive de la thèse, elle m'apporte bien de la couleur et de bons moments.

# Table des matières

Résumé	i
Avant-Propos	ii
Table des Figures	viii
Liste des Tableaux	xii
Glossaire	xiii
Introduction	1
<b>1 Les orbites périodiques</b>	<b>5</b>
1.1 Les orbites périodiques, squelette de la dynamique . . . . .	6
1.2 Un exemple utile : l'application logistique . . . . .	9
1.2.1 OPIs et stabilité . . . . .	9
1.2.2 Utilité des OPIs : calcul du taux d'échappement . . . . .	15
1.3 Détection d'orbites périodiques : Théorie . . . . .	23
1.3.1 Formulation générale du problème . . . . .	23
1.3.2 L'algorithme de Schmelcher-Diakonos ( <i>SD</i> ) . . . . .	27

1.3.2.1	Idée générale de l'algorithme <i>SD</i> . . . . .	27
1.3.2.2	La transformation <i>SD</i> pour un système quelconque à 2 dimensions . . . . .	29
1.3.2.3	Taux de convergence et nombre d'opérations par itération	32
1.3.3	L'algorithme de Davidchack-Lai ( <i>DL</i> ) . . . . .	33
1.3.3.1	Idée générale de l'algorithme <i>DL</i> . . . . .	33
1.3.3.2	L'algorithme <i>DL</i> sur un système particulier en 1 dimension . . . . .	35
1.3.3.3	Taux de convergence et nombre d'opérations par itération	37
1.3.4	Résumé des algorithmes de <i>SD</i> et de <i>DL</i> . . . . .	38
<b>2</b>	<b>Présentation des modèles chaotiques</b>	<b>41</b>
2.1	Considérations générales . . . . .	42
2.2	Applications analytiques . . . . .	43
2.2.1	Application de Hénon . . . . .	43
2.2.2	Application de Ikéda . . . . .	45
2.2.3	Application Standard . . . . .	47
2.2.4	Application Hénon-3D . . . . .	48
2.2.5	Le Pendule Double . . . . .	51
2.3	Applications semi-analytiques : les Billards . . . . .	55
2.3.1	Billards Libres . . . . .	56
2.3.2	Billards Magnétiques . . . . .	59
2.3.3	Billards Gravitationnels . . . . .	60
2.4	Flots continus . . . . .	66

2.4.1	Considérations générales . . . . .	66
2.4.1.1	Application discrète à partir du flot . . . . .	67
2.4.1.2	Matrice jacobienne de l'application discrète . . . . .	71
2.4.1.3	Détection des points stationnaires du flot . . . . .	78
2.4.2	Flot dissipatif : Système de Lorenz . . . . .	79
2.4.3	Flot Non-Autonome : Oscillateur de Duffing . . . . .	81
2.4.4	Flot conservatif : DKP . . . . .	83
<b>3</b>	<b>Tests numériques et résultats</b>	<b>90</b>
3.1	Étapes préliminaires . . . . .	91
3.1.1	Production du code . . . . .	91
3.1.2	Exemples numériques pour les flots continus . . . . .	93
3.1.3	Premier test : Détection d'OPIs chez Hénon . . . . .	95
3.2	Dans le “coeur” des algorithmes . . . . .	95
3.2.1	Visualisation du fonctionnement des algorithmes . . . . .	96
3.2.2	Ikéda comme banc d'essais . . . . .	97
3.2.3	Optimisation de l'algorithme <i>SD</i> . . . . .	104
3.2.4	Ikéda revisité . . . . .	111
3.2.5	Quelques nuances dans l'algorithme <i>DL</i> . . . . .	113
3.3	Détection d'OPIs pour d'autres systèmes . . . . .	115
3.3.1	Systèmes conservatifs . . . . .	115
3.3.2	Systèmes de dimensions 3 et 4 . . . . .	123
3.3.3	Systèmes continus . . . . .	125

3.4 Discussion . . . . .	128
<b>4 Ciblage à l'aveuglette</b>	<b>132</b>
4.1 La méthode du ciblage à l'aveuglette . . . . .	133
4.1.1 Utilité d'une méthode de ciblage en général . . . . .	133
4.1.2 Description de l'algorithme du ciblage à l'aveuglette . . . . .	133
4.1.2.1 Ciblage à plusieurs paramètres . . . . .	136
4.1.2.2 Ciblage à un paramètre . . . . .	137
4.1.3 Résumé de l'algorithme . . . . .	139
4.2 Quelques résultats . . . . .	139
4.2.1 Ciblage à $D$ paramètres . . . . .	141
4.2.2 Ciblage à 1 paramètre . . . . .	142
4.3 Ciblage dans un contexte expérimental . . . . .	159
4.3.1 Construction d'un prédicteur . . . . .	159
4.3.1.1 Prédicteur de Lorenz . . . . .	160
4.3.1.2 Prédicteur linéaire . . . . .	160
4.3.2 Ciblage à l'aveuglette sur un système dynamique reconstruit . .	163
4.3.3 Cas particulier : ciblage sur un système conservatif reconstruit .	173
4.4 Efficacité du ciblage à l'aveuglette . . . . .	175
4.5 Discussion . . . . .	181
<b>Conclusion</b>	<b>187</b>
<b>A Démonstrations de l'algorithme de <math>SD</math> en <math>2D</math></b>	<b>191</b>

<b>B Billards : méthode de propagation</b>	<b>196</b>
B.1 Billards Libres . . . . .	196
B.2 Billards Magnétiques . . . . .	198
B.2.1 Techniques numériques . . . . .	198
B.2.2 Un mot sur l'algorithme . . . . .	201
<b>C Application discrète à partir du flot DKP</b>	<b>204</b>
<b>D Publications associées à cette thèse</b>	<b>214</b>

# Table des figures

1.1	Orbites périodiques et attracteur du système de Hénon. . . . .	7
1.2	$x_n$ en fonction du paramètre $R$ pour l'application logistique. . . . .	11
1.3	Itérations de l'application logistique autour de points fixes stable et instable. . . . .	13
1.4	Fonction de premier et deuxième retour pour l'application logistique. .	15
1.5	Fraction des “survivants” après $k$ itérations pour l'application logistique avec $R = 5$ (calculée avec $N = 10^7$ conditions initiales). . . . .	17
1.6	Points fixes de $\cos(x_n^2) + x_n$ et leurs bassins d'attraction. . . . .	37
2.1	Orbites périodiques et attracteur du système de Hénon ( $B = 0.3$ ). . . .	44
2.2	Attracteur de l'application Ikéda avec $a = 1.0$ , $b = 0.9$ , $k = 0.4$ et $\eta = 6.0$ . .	46
2.3	Application standard pour quelques valeurs de $K$ . . . . .	49
2.4	Attracteur du système de Hénon-3D. . . . .	50
2.5	Le Pendule Double . . . . .	51
2.6	Projection de l'attracteur du Pendule Double. . . . .	54
2.7	Dynamique du Billard Libre. . . . .	56
2.8	Dynamique du Billard Magnétique. . . . .	59
2.9	Dynamique du Billard Gravitationnel. . . . .	61

2.10 Diagramme des phases d'un Billard Ovale Libre. . . . .	63
2.11 Diagramme des phases d'un Billard Ovale Magnétique. . . . .	64
2.12 Diagramme des phases d'un Billard Gravitationnel. . . . .	65
2.13 Section de Poincaré et définition de $h(\tilde{\mathbf{v}}(t))$ . . . . .	69
2.14 Interprétation géométrique de l'équation 2.39. . . . .	72
2.15 Interprétation géométrique de l'équation 2.45. . . . .	76
2.16 Attracteur de Lorenz et Section de Poincaré. . . . .	80
2.17 Section de Poincaré et portions d'une trajectoire pour l'oscillateur de Duffing. . . . .	84
2.18 Trajectoires continues de l'oscillateur de Duffing. . . . .	85
2.19 Diagrammes des phases du flot DKP sur une Section de Poincaré. . . .	88
2.20 Dynamique du flot DKP et Section de Poincaré. . . . .	89
3.1 Dynamique des algorithmes de $SD$ et $DL$ sur le système d'Ikéda. . . .	98
3.2 Dynamique des algorithmes de $SD$ et $DL$ sur l'attracteur de Hénon. . .	99
3.3 Grandeur des pas $\delta\mathbf{w}_{SD}$ et $\delta\mathbf{w}_{DL}$ en fonction du voisinage $g$ (Système Ikéda). . . . .	100
3.4 Nombre d'OPIs trouvées en fonction de $\lambda$ et $\beta$ . . . . .	102
3.5 Nombre d'itérations $SD$ par rapport $DL$ en fonction de la période recherchée (Système Ikéda). . . . .	103
3.6 Nombre moyen de pas $SD$ pour atteindre un résidu $g \equiv  \mathbf{G}(\mathbf{w}) $ . . . .	106
3.7 Dynamique autour du point fixe dans le système transformé $\mathcal{W}_k$ . . . . .	109
3.8 Nombre moyen de pas $SD$ pour atteindre un résidu $g \equiv  \mathbf{G}(\mathbf{w}) $ dans Ikéda avec $C_k = C_1, C_3$ . . . . .	111
3.9 Algorithme $SD$ avec l'ajout de corrections Newton-Raphson. . . . .	114

3.10 Algorithme <i>DL</i> avec bruit sur la matrice jacobienne. . . . .	116
3.11 Rapport $N_{SD}/N_{DL}$ pour différents niveaux bruit sur la matrice jacobienne.	117
3.12 Nombre d'OPIs détectées par <i>DL</i> en fonction de $\beta$ pour des conditions initiales choisies aléatoirement sur l'attracteur. . . . .	117
3.13 Dynamique de l'algorithme <i>DL</i> sur l'Application Standard ( $C_k = C_1$ ; $\beta = 10$ ). . . . .	118
3.14 Recherche des OPIs de l'Application Standard. . . . .	120
3.15 OPIs elliptiques de période-11 dans l'Application Standard ( $K = 1.25$ ). .	121
3.16 OPIs elliptiques de période-11 dans l'Application Standard ( $K = 1.7$ ). .	122
3.17 Nombre d'OPIs détectées en fonction de $\beta$ pour le système Hénon-3D ( $A = 1.76, B = 0.1$ ). . . . .	124
4.1 Évolutions de $\mathbf{v}_n$ (libre et sous la transformation de SD) . . . . .	136
4.2 Ciblage de période 1 et 2 sur Ikéda . . . . .	145
4.3 Orbites de période 1 et 2 sur Ikéda . . . . .	146
4.4 Ciblage de période 3 et 4 sur le <i>Billard Magnétique</i> . . . . .	147
4.5 Orbites de période 3 et 4 dans un <i>Billard Magnétique</i> . . . . .	148
4.6 Ciblage de période 1 et 2 sur le système de Lorenz . . . . .	149
4.7 Orbites de période 3 et 4 de Lorenz . . . . .	150
4.8 Ciblage de période 1, 2, 4 et 7 sur Hénon . . . . .	151
4.9 Orbites de période 1, 2, 4 et 7 dans le système de Hénon . . . . .	152
4.10 Ciblage de période 3 et 4 sur le <i>Billard Ovale Libre</i> . . . . .	153
4.11 Orbites de période 3 et 4 dans un <i>Billard Ovale Libre</i> . . . . .	154
4.12 Ciblage de période 2 et 3 sur le système de DKP . . . . .	155

4.13 Orbites de période 2 et 3 dans le système DKP . . . . .	156
4.14 Ciblage de période 1 pour le Pendule Double . . . . .	157
4.15 Point fixe du Pendule Double . . . . .	158
4.16 Erreur moyenne normalisée des prédicteurs en fonction de $N_{app}$ . . . . .	164
4.17 Reconstruction de l'attracteur de Duffing . . . . .	166
4.18 L'attracteur de Duffing reconstruit sur la Section de Poincaré $\mathcal{M}(\tau_n) = 1$ , $\frac{d\mathcal{M}}{dt}\Big _{\tau_n} > 0$ . . . . .	167
4.19 Premier scénario de ciblage sur l'oscillateur de Duffing . . . . .	170
4.20 Deuxième scénario de ciblage sur l'oscillateur de Duffing . . . . .	172
4.21 Troisième scénario de ciblage sur l'oscillateur de Duffing . . . . .	174
4.22 Diagramme des phases et orbites de période-6 pour le <i>Billard Gravitationnel</i>	176
4.23 Ciblage à l'aveuglette sur un <i>Billard Gravitationnel</i> . . . . .	177
4.24 Reconstruction dynamique du <i>Billard Gravitationnel</i> . . . . .	178
4.25 Temps de transition moyen en fonction de la perturbation seuil $\delta p_{max}$ .	182
4.26 Voisinage efficace $\epsilon_{eff}$ en fonction de la perturbation seuil $\delta p_{max}$ . . . .	183
4.27 Temps de transition moyen pour un ciblage à 1 et 2 paramètres . . . .	184
<b>B.1 Définition des variables du Billard Libre.</b> . . . . .	197
<b>B.2 Billard magnétique en deux dimensions.</b> . . . . .	200
<b>B.3 Recherche de <math>\varphi_{n+1}</math> : géométrie du problème.</b> . . . . .	203

# Liste des tableaux

1.1	Positions et stabilité des premières OPIs de l'application logistique ( $R = 5$ ). . . . .	22
1.2	Premiers coefficients de $D(z)$ (équation 1.20), première racine $z^*$ et coefficient $\gamma$ correspondant (Application logistique $R = 5$ ). . . . .	22
1.3	Choix des matrices $\mathbf{C}_k$ en $2D$ en fonction du type d'OPIs. . . . .	31
1.4	Résumé des algorithmes et $SD$ et $DL$ . . . . .	40
3.1	Quelques exemples numériques pour les flots continus. . . . .	94
3.2	Les OPIs de l'Application Standard ( $K = 1.25$ ). . . . .	123
3.3	Les OPIs du système de Lorenz. . . . .	127
3.4	Les OPIs du système $DKP$ . . . . .	127
4.1	Résumé de l'algorithme du ciblage à l'aveuglette . . . . .	140
4.2	Résumé des scénarios de ciblage à $D$ paramètres . . . . .	142
4.3	Résumé des scénarios de ciblage à 1 paramètre . . . . .	144
4.4	Points fixes de l'oscillateur de Duffing dans un espace reconstruit . . . . .	168
C.1	Aide-mémoire pour les notations du chapitre 2. . . . .	211
C.1	Aide-mémoire pour les notations du chapitre 2 (suite). . . . .	212
C.1	Aide-mémoire pour les notations du chapitre 2 (suite). . . . .	213

# Glossaire

**Notations utilisées dans cette thèse :**

- Les vecteurs et matrices sont notés par une lettre en caractère gras :

- Vecteur à  $D$  dimensions :  $\mathbf{v} = \begin{pmatrix} v^1 \\ \vdots \\ v^D \end{pmatrix}$
- Matrice  $D \times D$  :  $\mathbf{J} = \begin{pmatrix} J_{11} & \cdots & J_{DD} \\ \vdots & \ddots & \vdots \\ J_{D1} & \cdots & J_{DD} \end{pmatrix}$

*Note : Les composantes des vecteurs sont notées par un chiffre placé en exposant.*

- L'état d'un vecteur  $\mathbf{v}$  à l'itération  $n$  est noté :

- $\mathbf{v}_n$

- La dérivée d'un vecteur  $\mathbf{F}$  de dimension  $D$  par rapport à un paramètre  $p$  est notée :

- $D_p \mathbf{F} = \begin{pmatrix} \frac{dF^1}{dp} \\ \vdots \\ \frac{dF^D}{dp} \end{pmatrix}$

- La dérivée d'un vecteur  $\mathbf{F}$  de dimension  $M$  par rapport à un autre vecteur  $\mathbf{v}$  de dimension  $N$  est notée :

- $D_{\mathbf{v}} \mathbf{F} = \begin{pmatrix} \frac{dF^1}{dv^1} & \cdots & \frac{dF^1}{dv^N} \\ \vdots & \ddots & \vdots \\ \frac{dF^M}{dv^1} & \cdots & \frac{dF^M}{dv^N} \end{pmatrix}$

# Introduction

Le concept de périodicité semble profondément ancré dans l'esprit de l'être humain. L'idée de revenir au point de départ après un certain temps se retrouve dans bien des domaines, tant en sciences humaines qu'en sciences naturelles. En philosophie par exemple, Camus discute de l'absurde en nous faisant redécouvrir le pauvre Sisyphe, personnage mythique, condamné par les dieux à rouler une pierre pour l'éternité. Jour après jour, il pousse cette pierre jusqu'en haut d'une montagne pour la voir inévitablement redescendre de l'autre côté quelques instants plus tard... Un parcours fermé, comme celui de Sisyphe, peut paraître complètement absurde dans certaines situations et devenir bien rassurant à d'autres occasions....

Quoi qu'il en soit, en physique, les solutions périodiques se présentent souvent comme *solution naturelle* à un problème donné. En astronomie par exemple, on reconnaît dès le 17<sup>ième</sup> siècle toute l'importance de ces solutions. Les observations de Brahe combinées à l'analyse de Kepler résument le mouvement des planètes à des trajectoires elliptiques. Le calcul différentiel et la loi de gravitation universelle, fruits du travail de Newton, apparaissent quelques années après les découvertes de Kepler, et prédisent également la trajectoire elliptique comme solution possible au mouvement des corps célestes.

Les premiers travaux sur les oscillateurs non-linéaires, plus particulièrement ceux de Lord Rayleigh, conduisent aussi à des cycles périodiques (cycles limites) comme solutions du mouvement. À cette époque, Rayleigh est particulièrement intéressé à décrire le fonctionnement des instruments de musique. Dans une des ses œuvres [78], il présente plusieurs modèles pour décrire la vibration de différents instruments. Malgré la non-linéarité des oscillateurs considérés, il trouve pour certains d'entre eux des cycles limites comme solution du mouvement. Ces cycles sont atteints par les oscillateurs peu importe la condition initiale. Cette notion de cycle limite se retrouve plus tard dans les travaux de van der Pol sur *l'oscillateur cardiaque* [22].

Il existe cependant plusieurs autres systèmes physiques qui ne peuvent se laisser décrire par de simples fonctions périodiques. Le problème classique à 3 corps (Soleil-Terre et Lune) a hanté bien des scientifiques aux 17<sup>ième</sup> et 18<sup>ième</sup> siècles, assez même

pour faire l'objet d'un concours dans le célèbre journal de mathématiques *Acta Mathematica*. On fait alors appel au génie des scientifiques de l'époque pour trouver une fonction analytique qui décrirait correctement la position de ces 3 corps en fonction du temps.

Ce problème classique a sans aucun doute contribué au développement des théories modernes du chaos et de la dynamique non-linéaire. Le mathématicien Henri Poincaré consacre une partie de sa vie à l'étude théorique du mouvement des corps célestes. Son approche est complètement originale à l'époque. On lui doit des outils fort astucieux dont la section qui porte son nom (*Section de Poincaré*). Il est le premier à reconnaître toute l'importance de l'espace des phases pour la compréhension de la dynamique. Poincaré réalise que derrière toute la complexité apparente de certains systèmes, se cache une "charpente" rigide, constituée de cycles périodiques de plus en plus longs.

Étant données des équations... et une solution particulière quelconque de ces équations, on peut toujours trouver une solution périodique (dont la période peut, il est vrai, être très longue), telle que la différence entre les deux solutions soit aussi petite que l'on veut, pendant un temps aussi long qu'on le veut. D'ailleurs, ce qui nous rend ces solutions périodiques si précieuses, c'est qu'elles sont, pour ainsi dire, la seule brèche par où nous puissions essayer de pénétrer dans une place jusqu'ici réputée inabordable.[74]Vol.1  
§36

Même pour un système complexe comme le problème à trois corps, les orbites ou les cycles périodiques sont, pour ainsi dire, la clé d'une meilleure compréhension et description de la dynamique. Tout le mouvement des systèmes chaotiques se regroupe essentiellement autour de ces cycles fondamentaux, généralement instables. Ainsi, la valeur moyenne d'un observable quelconque peut se faire par une moyenne pondérée de cet observable évalué uniquement sur les orbites périodiques (instables).

On entend ici par orbite périodique (ou cycle périodique), l'ensemble des états du système dans l'espace des phases qui sont visités à des intervalles de temps réguliers. On ajoute le qualificatif *instable* car en général, un système chaotique tend à s'éloigner de ce type d'orbite s'il n'est pas positionné *précisément* sur celle-ci. Dans le cas contraire, on parle d'une orbite périodique *stable*. Le "degré" d'instabilité/stabilité est relié au taux d'éloignement entre le système et le cycle périodique en question. La connaissance du "degré" d'instabilité des orbites périodiques dans les systèmes chaotiques permet d'évaluer les quantités dynamiques qui le caractérisent et les observables physiques qui lui sont associés.

Dans un autre ordre d'idée, la localisation précise des cycles périodiques est généralement nécessaire à l'implémentation des algorithmes de contrôle. Ceux et celles pour qui le chaos n'était qu'une discipline condamnée à une description abstraite (et

souvent colorée !) des systèmes dynamiques complexes n'ont pu rester indifférents face aux travaux de Ott, Grebogi et York publiés en '90 [67]. Ces derniers ont présenté une méthode fort astucieuse pour maintenir un système chaotique sur une de ces orbites périodiques, réalisant ainsi ce que von Neumann avait entrevu 40 ans plus tôt. Cette publication a donné naissance à une nouvelle branche d'étude en dynamique non-linéaire : le contrôle du chaos [12],[27],[28]. Toutefois, le contrôle des orbites périodiques instables passe généralement par une étape préliminaire, soit celle de leur localisation et de leur caractérisation (i.e. le calcul de stabilité).

Il s'avère donc essentiel de développer des algorithmes généraux pour la détection des cycles instables dans les systèmes chaotiques. Cette thèse se consacre principalement à l'étude de ces méthodes de détection. Elle est divisée en 4 chapitres. Au **chapitre 1**, nous reprenons avec plus de détails certains éléments de cette brève introduction. L'application logistique, qui est un système dynamique unidimensionnel, est alors utilisée pour introduire de façon plus intuitive la notion de stabilité des orbites périodiques et la problématique reliée à leur détection. Cet exemple permet également de montrer le lien étroit qui existe entre une propriété dynamique globale du système chaotique et la stabilité de ses cycles périodiques. Ce chapitre sert aussi à exposer le problème de la détection des orbites périodiques de façon générale, i.e. pour un système chaotique dans un espace des phases à  $D$  dimensions. Deux méthodes de détection récemment publiées sont expliquées en détails, soit celle proposée par Schmelcher & Diakonos [90] ainsi que celle de Davidchack & Lai [19].

Le **chapitre 2** présente l'ensemble des systèmes chaotiques utilisés dans ce travail. Nous choisissons ici une vaste gamme de systèmes dynamiques, qui s'étend des applications analytiques discrètes jusqu'aux flots continus non-autonomes. Mentionnons ici que les deux méthodes de détection citées ci-haut n'ont été validées, lors de leur publication, que sur des applications discrètes analytiques. Pourbohloul [75] a utilisé la première de ces méthodes [90] sur des flots conservatifs mais n'a pu obtenir de résultats positifs pour de tels systèmes. Cet échec a en partie amorcé notre recherche il y a 3 ans. Une attention particulière est donc portée aux systèmes continus. Ainsi, nous expliquons en détails comment obtenir numériquement et avec une bonne précision les quantités qui sont nécessaires à l'implémentation de chacune de ces méthodes de détection.

Tout récemment, un groupe de recherche a publié la suite des travaux de Schmelcher & Diakonos sur les systèmes continus [72]. Leurs résultats sont en partie en accord avec ce que nous présentons au **chapitre 3**. Ce chapitre regroupe donc l'ensemble des résultats que nous avons obtenus (i.e. nombre d'orbites détectées) en appliquant les 2 algorithmes sur différents systèmes chaotiques. Nous essayons alors de faire ressortir les différences entre les deux algorithmes de détection tout en mentionnant les forces et les faiblesses de ceux-ci.

Comme nous le verrons au chapitre 1, les méthodes de détection que nous utilisons

reposent sur une transformation particulière de la dynamique initiale. Cette astuce nous a inspiré une méthode alternative pour cibler et stabiliser une orbite périodique quelconque dans un système chaotique. Le *ciblage* est un processus qui consiste à perturber le système chaotique pour l'amener sur une des ses orbites périodiques. Cette perturbation se fait en changeant légèrement dans le temps la valeur d'un paramètre du système. Le ciblage est en général combiné aux méthodes de contrôle pour stabiliser rapidement un cycle donné. En général, il faut connaître la position et la stabilité du cycle en question.

La nouvelle méthode de ciblage que nous présentons au **chapitre 4** ne demande pas de connaître *a priori* la position et la stabilité des orbites périodiques. Elle pourrait être un outil complémentaire aux méthodes de ciblage et de contrôle actuelles. Le dernier chapitre est donc consacré à décrire cette nouvelle méthode et à la valider sur quelques systèmes dynamiques. Nous avons publié cette méthode au printemps 2002 [25].

# Chapitre 1

## Les orbites périodiques

Ce chapitre présente les éléments essentiels qui concernent les orbites périodiques et leur détection. Nous commençons d'abord par donner un bref aperçu du rôle important que ces cycles ont à jouer pour une meilleure compréhension des systèmes chaotiques. Un exemple simple de système chaotique, soit l'application logistique, est ensuite utilisé afin de présenter la problématique reliée à la détection de ces cycles particuliers. Cet exemple nous permet de définir les quantités importantes pour l'ensemble de cette thèse. De plus, nous montrons, pour cette même application unidimensionnelle, le lien étroit qui existe entre une propriété globale de sa dynamique (soit le taux d'échappement) et la stabilité de ses orbites périodiques. Enfin, nous présentons 2 méthodes de détection particulières qui seront étudiées en détail au chapitre 3.

## 1.1 Les orbites périodiques, squelette de la dynamique

... ce qui nous rend ces solutions périodiques si précieuses, c'est qu'elles sont, pour ainsi dire, la seule brèche par où nous puissions essayer de pénétrer dans une place jusqu'ici réputée inabordable.

Cette citation de Poincaré, que nous reprenons encore une fois ici, résume l'essentiel de ce que nous présentons dans cette section. Toute la description quantitative des propriétés dynamiques globales d'un système chaotique passe d'abord par la connaissance de la position et la stabilité de ses cycles périodiques. La dynamique repose en quelque sorte sur une “charpente” rigide constituée des orbites périodiques du système.

Afin de donner une première image de cette “charpente”, on considère à la figure 1.1 *l'application de Hénon*. Cette application sera reprise au chapitre suivant. Pour l'instant, mentionnons qu'il s'agit d'une application discrète  $\mathbb{R}^2 \rightarrow \mathbb{R}^2$  qui transforme un état donné par un vecteur  $\mathbf{v}_n = (x_n, y_n)$  en un nouveau vecteur  $\mathbf{v}_{n+1}$ . La figure 1.1a montre l'espace des phases de l'attracteur de Hénon pour les paramètres ( $A = 1.4, B = 0.3$ ). L'état du système évolue de façon chaotique sur cet attracteur (Nous avons généré cette figure en itérant l'application 12000 fois). Sur les figures 1.1b à 1.1d, on place uniquement les orbites périodiques instables de différentes longueurs. Par exemple sur la figure 1.1b, on retrouve uniquement la position des orbites instables de période-1 à période-6. Si le système de Hénon était placé précisément sur l'orbite de période-6, il reviendrait au même point après 6 itérations. On remarque sur la figure qu'avec seulement la position des orbites de période-1 à période-18 (environ 12000 positions), on retrouve la forme de l'attracteur de Hénon. Les premières orbites instables de Hénon forment le squelette de l'attracteur, comme le constatait Poincaré il y a de cela plus de 100 ans ! Insistons ici sur le fait que ces orbites sont *instables*. Même placé “sur” une orbite périodique au départ (i.e. à la précision de la machine), le système s'en éloigne graduellement à chaque itération.

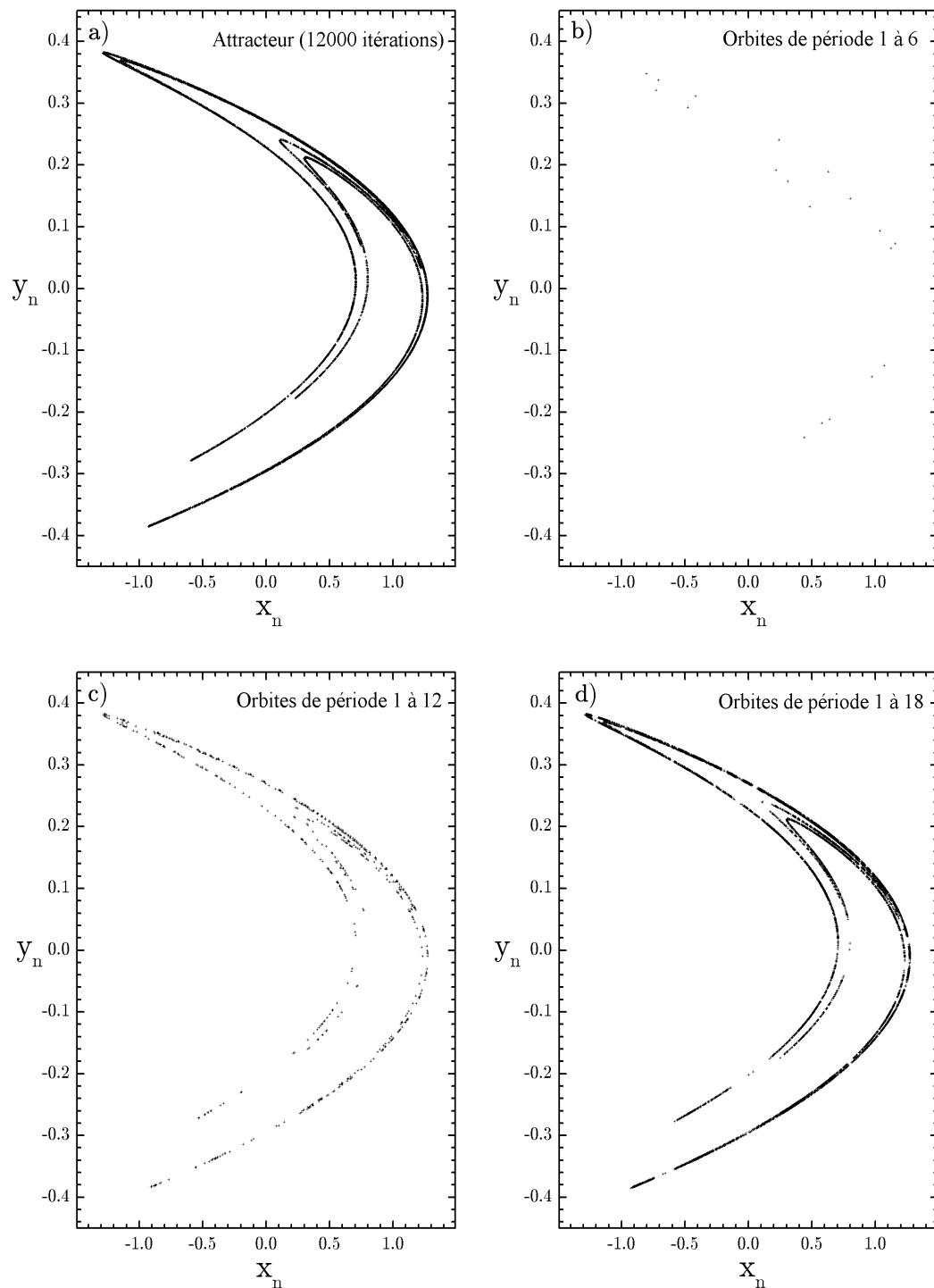


FIG. 1.1 – Orbites périodiques et attracteur du système de Hénon.

L’ensemble des Orbites Périodiques Instables (OPIs) d’un système chaotique est pour ainsi dire, intimement relié à ses propriétés dynamiques. À partir de l’ensemble (en principe infini) d’OPIs cachées dans l’espace des phases, on peut obtenir des quantités telles les exposants de Lyapunov [52], la mesure invariante [37], l’entropie topologique et la dimension fractale [3]. Dans les systèmes ouverts (i.e. les systèmes où l’état dans l’espace des phases n’est pas confiné à une région donnée), on peut estimer le taux d’échappement en connaissant la stabilité des OPIs les plus courtes [15] [17]. La connaissance de la stabilité des OPIs permet ainsi d’évaluer un grand nombre de quantités invariantes d’une dynamique chaotique [1], [2].

En chaos quantique (i.e. l’étude de systèmes quantiques qui ont un équivalent *chaotique* dans les limites classiques, voir par exemple [11]), ces mêmes OPIs permettent de faire le pont entre le comportement classique et quantique de systèmes non-intégrables [16]. Par exemple, la formule de la trace de Gutzwiller [38] relie directement les densités d’états quantiques (les niveaux d’énergie) aux orbites périodiques du Hamiltonien *classique*. C’est en quelque sorte ce que fait le modèle semi-classique de Bohr pour obtenir les niveaux d’énergie de l’atome d’hydrogène. Ce modèle ne convient cependant plus pour l’hélium. Le traitement semi-classique de cet atome passe alors par un développement sur les cycles périodiques (soit la formule de la trace de Gutzwiller). Cette démarche a été réalisée au début des années ’90 par le regretté Wintgen [107]. L’approche semi-classique est ici toujours souhaitable puisqu’elle permet d’illuminer les solutions numériques plus obscures obtenues par un traitement purement quantique.

Les niveaux d’énergie du problème anisotrope de Kepler (anisotropic Kepler problem, AKP), problème qui décrit le mouvement d’un électron dans les semi-conducteurs comme le silicium ou le germanium [38], peuvent aussi s’obtenir par une approche semi-classique [102], [103], [14]. De la même façon, on peut évaluer le spectre en énergie de l’atome d’hydrogène dans un champ magnétique [101] et sa section efficace de photo-ionization [58].

Bien que la théorie reliant les observables physiques aux orbites périodiques soit relativement bien établie, on doit inévitablement localiser les cycles de différentes longueurs pour évaluer ces observables. La détection des orbites périodiques dans l'espace des phases est cependant une tâche complexe. Comme les orbites recherchées sont instables, la trajectoire tend à s'éloigner naturellement de celles-ci. Les propriétés intrinsèques aux systèmes chaotiques ne favorisent donc pas la détection de ces cycles périodiques. De plus, le nombre d'OPIs augmente de façon exponentielle avec la période, ce qui rend difficile la détection de *l'ensemble complet* des orbites pour les périodes élevées. Ce dernier point est crucial puisque la connaissance de la *totalité* des orbites périodiques est en principe nécessaire à l'évaluation des quantités dynamiques ([37], [1], [7] et [35]).

Avant de présenter les méthodes de détection utilisées dans ce travail, nous allons d'abord exposer la problématique reliée à la détection d'OPIs d'une façon plus intuitive à partir d'un exemple en une dimension. Ce cas particulier nous permettra de préciser d'avantage le terme *orbite périodique* ainsi que la notion de *stabilité*. De plus, nous effectuerons le calcul, pour cette même application 1D, du taux d'échappement, en considérant seulement les premières OPIs de ce système. Ce dernier calcul permettra d'entrevoir toute la puissance et l'élégance de la belle théorie du développement en cycles périodiques.

## 1.2 Un exemple utile : l'application logistique

### 1.2.1 OPIs et stabilité

Malgré sa grande simplicité, *l'application logistique* [59] offre une dynamique riche et variée. Il s'agit d'un modèle simplifié en écologie pour décrire l'évolution d'une population particulière d'insectes. Cette application est donnée par la fonction itérative

suivante :

$$\begin{aligned} x_{n+1} &= F(x_n) \\ &= Rx_n(1 - x_n), \end{aligned} \tag{1.1}$$

où  $R$  est un paramètre donné. Pour  $0 < R < 3$ , les itérés convergent vers un point particulier  $x^*$ , et ce, peu importe la condition initiale choisie. Ce point  $x^*$  est en fait la solution de l'équation :

$$\begin{aligned} x^* &= F(x^*) \\ &= Rx^*(1 - x^*). \end{aligned} \tag{1.2}$$

Si on itère l'application logistique à partir du point  $x^*$ , on revient précisément au même endroit après une seule itération. Ce point  $x^*$  est appelé *point fixe*.

Pour des valeurs de  $R$  comprises entre 3 et 3.446.., les itérés ne convergent plus vers un seul point, mais oscillent plutôt entre 2 valeurs particulières. Autrement dit, après une phase transitoire, les itérés reviennent au même endroit après 2 itérations. On parle alors d'une orbite de période-2. De façon générale, une composante  $x(k)$  d'une orbite de période- $m$  est caractérisée par :

$$x(k) = F^{(m)}(x(k)), \tag{1.3}$$

où  $F^{(m)}$  représente  $m$  itérations de l'application logistique à partir du point  $x(k)$ . En étudiant attentivement le comportement de l'application logistique sur une plage de valeurs de  $R$  donnée, on retrouve le scénario classique de doublement de période de Feigenbaum ([32], [34]). La figure 1.2 illustre ce scénario.

Cette figure représente, pour une valeur de  $R$  donnée, les états possibles du système après 5000 itérations. On remarque que pour  $1 < R < 3$ , les itérés convergent vers un point fixe (période-1). Si  $3 < R < 3.446..$ , la suite converge vers une période-2. Il y a apparition d'une période-4 pour  $R > 3.446$ . La période de l'orbite tend vers l'infini lorsque  $R$  s'approche de 3.569946.. On entre ensuite dans la région chaotique.

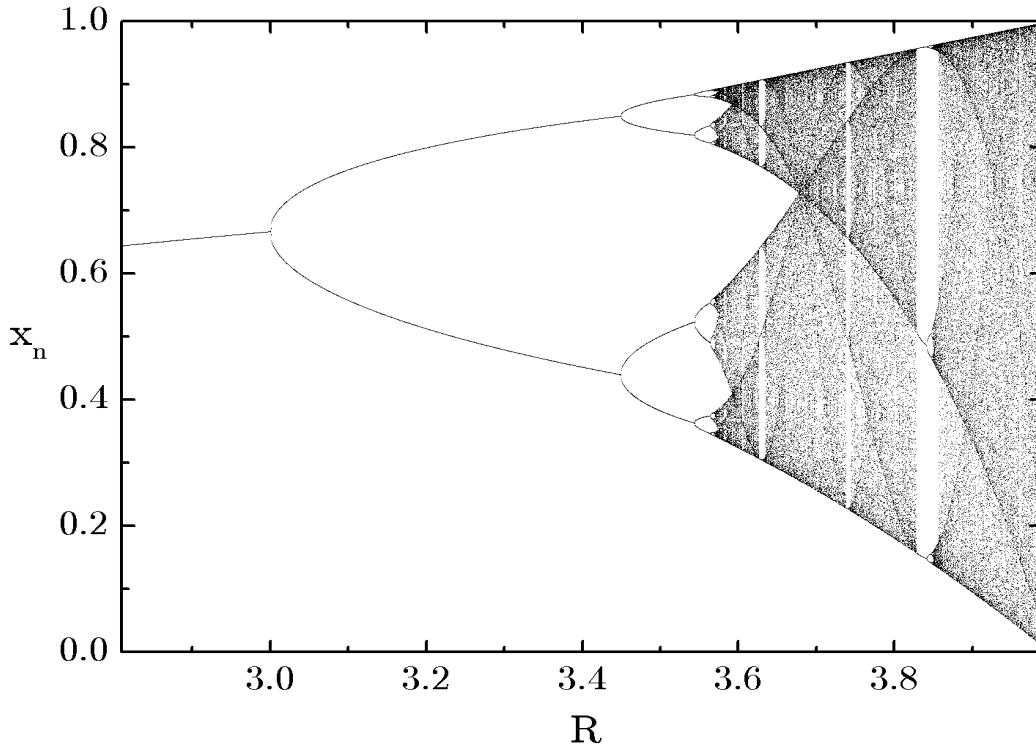


FIG. 1.2 –  $x_n$  en fonction du paramètre  $R$  pour l’application logistique.

Une description plus complète de ce scénario est faite par E. Ott [66]. Dans la région chaotique, il est important de préciser qu’il existe toujours des orbites de période- $m$  et ce, quel que soit  $m$ . En effet, on peut utiliser l’équation 1.2 pour écrire, par exemple, la position du point fixe en fonction de  $R$  :

$$x^* = \frac{R - 1}{R}. \quad (1.4)$$

Ce point fixe existe peu importe la valeur de  $R$ . Si un tel point existe pour toute valeur de  $R$ , pourquoi alors l’équation 1.1 ne converge-t-elle pas toujours vers celui-ci ? La réponse se cache dans le fait que la *stabilité* du point fixe varie en fonction de  $R$ . On peut perturber le point fixe d’une valeur  $\delta x$ . Dans certains cas, le système revient sur le point fixe après quelques itérations ; dans d’autres, il s’en éloigne rapidement. Pour distinguer quantitativement ces deux cas, on peut, en première approximation, regarder comment évolue la perturbation  $\delta x$  en développant 1.1 autour du point fixe  $x^*$  :

$$x^* + \delta x_{n+1} = F(x^*) + \frac{dF}{dx}\Big|_{x^*} \delta x_n + \dots \quad (1.5)$$

Nous utiliserons le symbole  $J$  pour représenter la dérivée  $\frac{dF}{dx}\Big|_{x^*}$  de l'application  $F$  au point fixe  $x^*$  considéré (En général,  $J$  se retrouve sous forme matricielle dans les systèmes à plusieurs dimensions, on parle alors de *la matrice jacobienne*). Cette dérivée nous renseigne sur la stabilité de l'orbite périodique.

Pour l'application logistique, le développement en série mène à :

$$\begin{aligned} \delta x_{n+1} &= J \delta x_n + \dots \\ &= (R - 2Rx^*) \delta x_n \\ &= (2 - R) \delta x_n \quad (\text{réf. équation 1.4}) . \end{aligned} \quad (1.6)$$

On remarque que si  $|2 - R| > 1$ , la valeur de  $\delta x_n$  augmente à chaque itération ; on s'éloigne ainsi du point fixe, qui est alors qualifié *d'instable*. À l'inverse, si  $|2 - R| < 1$ , le système revient au point fixe ; celui-ci est dit *stable*. Nous utilisons la quantité  $\rho$  pour désigner la valeur propre de  $J$  de l'application  $F$  au point fixe considéré. En une dimension,  $\rho$  et  $J$  sont évidemment identiques. Le critère pour distinguer un point stable d'un point instable peut ainsi s'écrire :

$$\begin{aligned} \text{Point fixe stable : } |\rho| &< 1 \\ \text{Point fixe instable : } |\rho| &> 1 . \end{aligned} \quad (1.7)$$

La figure 1.3 permet de visualiser ces critères. On représente, sur ces deux graphiques, comment évolue une perturbation autour de points fixes. Dans le premier cas (point fixe stable), on remarque que la perturbation disparaît après quelques itérations. La valeur absolue de la pente de la fonction  $F$  au point fixe  $x^*$  (soit  $|\rho|$ ) est plus petite que 1 ; la valeur de  $x_n$  se rapproche du point fixe à chaque itération. L'autre graphique représente la situation inverse, soit une perturbation qui augmente à chaque itération ; le point fixe est instable.

Comment obtient-on maintenant la stabilité des orbites de période- $m$  ? Il suffit de généraliser l'équation 1.5 et prendre  $F^{(m)}$  au lieu de  $F$ . Le développement en série

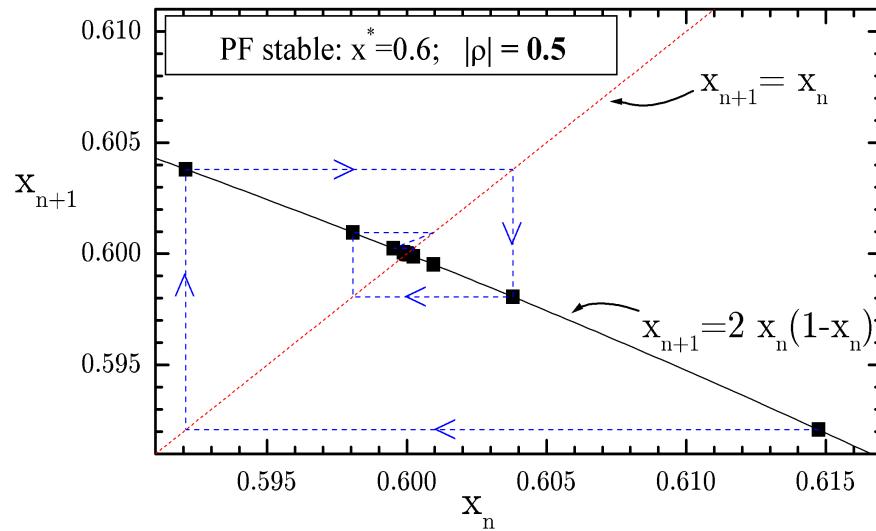
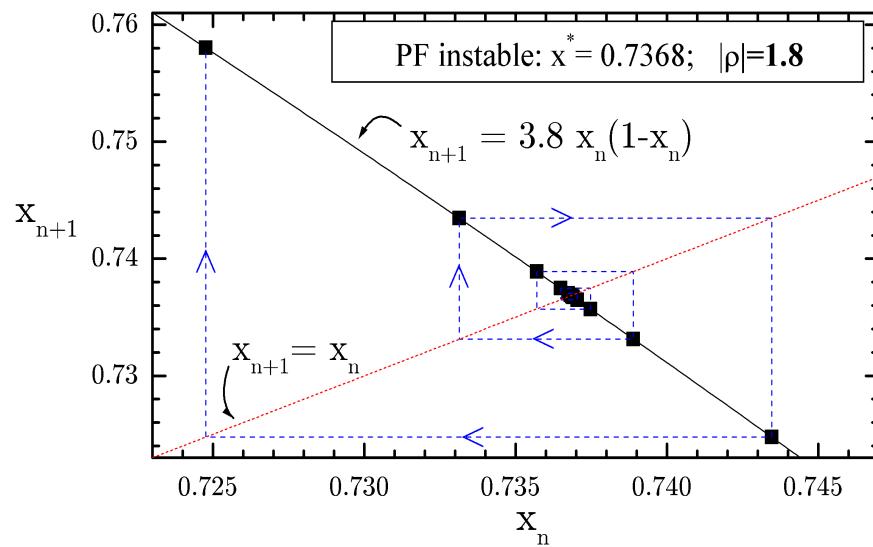
Application logistique avec R = 2.0Application logistique avec R = 3.8

FIG. 1.3 – Itérations de l’application logistique autour de points fixes stable et instable.

autour d'une composante  $x(k)$  de l'orbite périodique mène alors à :

$$\delta x_{n+m} = J^{(m)}(x(k)) \delta x_n + \dots \quad (1.8)$$

où  $J^{(m)}(x(k))$  représente la dérivée de  $F^{(m)}$  évaluée sur la  $k^{\text{ième}}$  composante de l'orbite périodique ( $OP$ ) considérée :

$$\begin{aligned} J^{(m)}(x(k)) &= \left. \frac{dF^{(m)}}{dx} \right|_{OP} \\ &= \underbrace{\left. \frac{dF}{dx} \right|_{x(k-1)} \cdots \left. \frac{dF}{dx} \right|_{x(2)} \left. \frac{dF}{dx} \right|_{x(1)} \left. \frac{dF}{dx} \right|_{x(m)} \cdots \left. \frac{dF}{dx} \right|_{x(k+1)} \left. \frac{dF}{dx} \right|_{x(k)}}_{\text{Produit de } m \text{ dérivées}}, \end{aligned} \quad (1.9)$$

où les  $x(m), x(m-1), \dots, x(1)$  sont les composantes de l'orbite de période- $m$ . Pour une orbite de période- $m$ , il y a  $m$  de ces dérivées et elles sont toutes identiques<sup>1</sup>. Le critère de stabilité est toujours donné par l'équation 1.7; il faut cependant prendre la valeur propre  $\rho$  de  $J^{(m)}$  (équation 1.9).

Cette dernière équation montre que  $\rho$  devrait en général augmenter avec le nombre d'itérations. On peut visualiser cet effet à l'aide de la figure 1.4, qui représente l'allure des fonctions  $F(x_n)$  et  $F^{(2)}(x_n)$ . Règle générale, pour un  $x_n$  donné, la pente de  $F^{(2)}(x_n)$  est plus forte que celle de  $F(x_n)$ . L'instabilité d'un point fixe devrait augmenter avec sa période. Sur ce dernier graphique, nous représentons en noir les régions où  $|\rho| < 1$ . Si l'orbite se trouve dans cette région, elle sera stable. On peut noter que les régions de stabilité diminuent avec la période. De plus, le nombre d'OPIs pour une période- $m$  donnée correspond au nombre d'intersections entre la fonction  $F^{(m)}(x_n)$  et la droite pointillée. Ce nombre d'intersections est directement relié aux oscillations de  $F^{(m)}(x_n)$ , un produit de  $m$  polynômes. Le nombre d'OPIs augmente donc de façon exponentielle avec la période.

On voit ainsi apparaître une double difficulté pour la détection de toutes les OPIs

---

<sup>1</sup>Pour les systèmes à  $D$  dimensions comme nous le verrons plus loin, les  $m$  matrices jacobniennes sont différentes car le produit matriciel n'est pas commutatif. Leur Jacobien (qui est le déterminant de la matrice) est toutefois identique.

de période- $m$  lorsque  $m$  devient grand : les OPIs deviennent en général de plus en plus instables et leur nombre augmente de façon exponentielle.

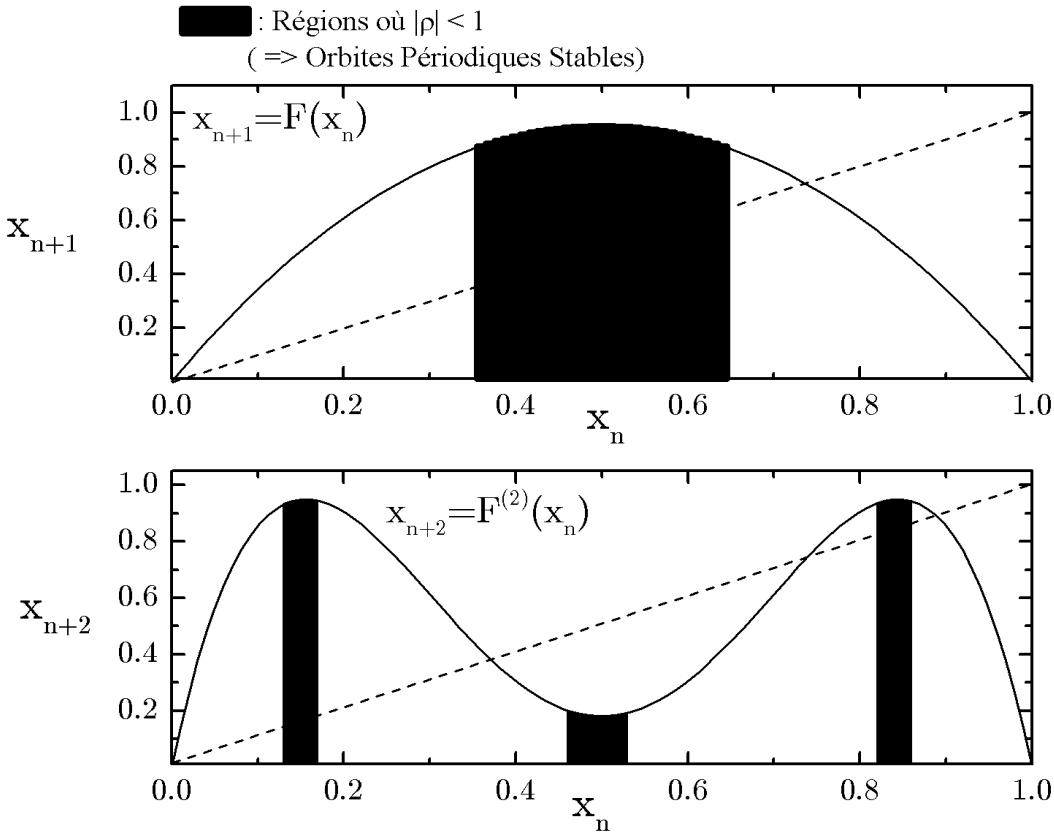


FIG. 1.4 – Fonction de premier et deuxième retour pour l’application logistique.

### 1.2.2 Utilité des OPIs : calcul du taux d’échappement

Avant d’aborder les différents algorithmes de la détection d’OPIs, nous allons ici supposer que ces cycles sont connus et voir quelles sont les informations que l’on peut en tirer. Nous poursuivons avec l’exemple de l’application logistique.

Cette application est un système fermé pour  $R \leq 4$ ; toute condition initiale choisie entre 0 et 1 et itérée  $k$  fois reste dans cet intervalle peu importe la valeur de  $k$ . Par contre, pour  $R > 4$ , l’état  $x_n$  peut sortir de l’intervalle  $[0, 1]$  après  $k$  itérations et on parle

maintenant d'un système ouvert. Un tel système pourrait par exemple correspondre à la dynamique d'un électron non-lié dans un potentiel donné.

Il peut être intéressant d'estimer le taux d'échappement moyen  $\gamma$  d'un ensemble de conditions initiales dans cette dynamique *ouverte*. On peut faire ce calcul par la "force brute" : on choisit alors plusieurs conditions initiales dans l'intervalle  $[0, 1]$  et on itère chacune de celles-ci jusqu'à ce qu'elles sortent de l'intervalle. On devrait s'attendre à voir le nombre de "survivants" diminuer de façon exponentielle avec  $k$ . Si  $\Gamma_k$  représente la fraction de survivants après  $k$  itérations, on peut ainsi écrire :

$$\Gamma_k = e^{-k\gamma_k} \quad (1.10)$$

Le coefficient  $\gamma$  est défini à la limite  $k \rightarrow \infty$  [15] :

$$\gamma \equiv \lim_{k \rightarrow \infty} \frac{\ln \Gamma_k}{k} \quad (1.11)$$

En traçant un graphique  $(\log \Gamma_k)$  versus  $k$ , on obtient une droite de pente  $\gamma$ . Le graphique 1.5 représente les résultats obtenus avec  $N = 10^7$  pour l'application logistique ( $R = 5$ ). On obtient par cette méthode  $\gamma = 0.5529 \pm 0.0004$ .

Cette première approche est très intuitive. Il s'agit d'une méthode purement statistique. Nous allons maintenant voir comment estimer cette constante  $\gamma$  au moyen des OPIs de l'application logistique. Il est possible de représenter l'expérience que nous venons de réaliser à l'aide d'un ordinateur sous la forme d'une intégrale particulière. Si  $\mathcal{M}$  représente l'intervalle  $[0, 1]$ , alors  $\Gamma_k$  peut s'écrire :

$$\Gamma_k = \frac{\int_0^1 dy \int_0^1 dx \delta(x - F^{(k)}(y))}{\int_0^1 dy} \quad (1.12)$$

La fonction  $F^{(k)}(y)$  à l'intérieur de la fonction Delta de Dirac représente  $k$  itérations de l'application logistique. La double intégrale au numérateur a la signification suivante : l'intégrale sur  $y$  choisit toutes les conditions initiales entre 0 et 1 et celle sur  $x$ , vérifie, au moyen de la fonction Delta, que cette condition est toujours à l'intérieur de l'intervalle 0 à 1 après  $k$  itérations. Le résultat au numérateur correspond donc à la longueur d'un

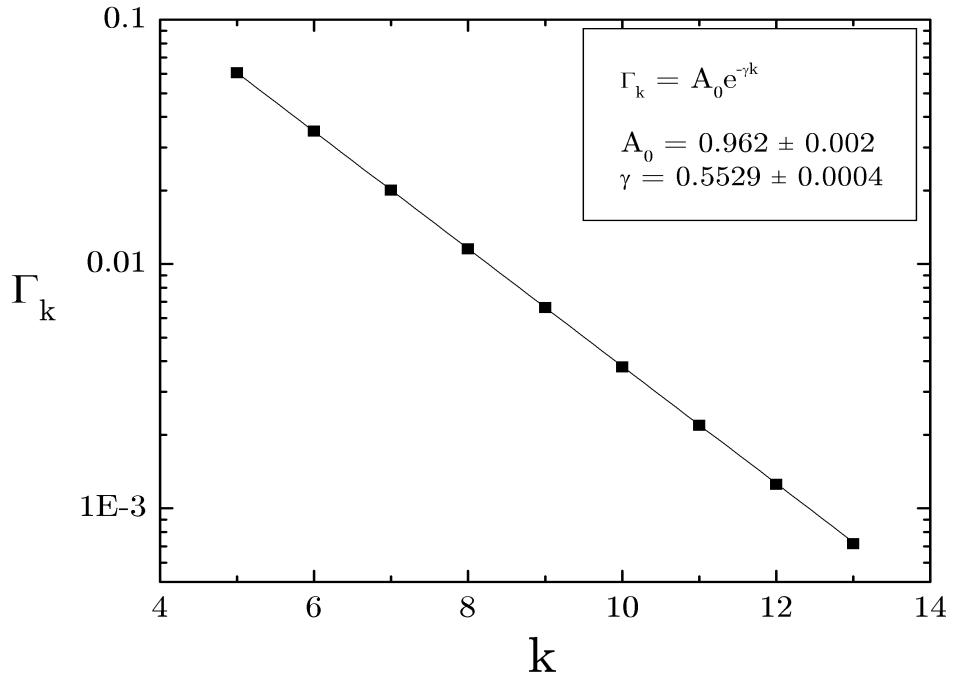


FIG. 1.5 – Fraction des “survivants” après  $k$  itérations pour l’application logistique avec  $R = 5$  (calculée avec  $N = 10^7$  conditions initiales).

intervalle de conditions initiales qui *demeurent* à l’intérieur de la région 0 à 1 pour  $k$  itérations. Au dénominateur, on retrouve simplement la longueur de l’intervalle initial (soit 1 dans le cas qui nous intéresse).

Pour  $k \rightarrow \infty$ , les seules contributions à la double intégrale sont les conditions initiales  $x$  choisies sur les OPIs du système (les autres conditions s’étant échappées) :

$$\Gamma_k = \int_0^1 dy \int_0^1 dx \delta(x - F^{(k)}(y)) \xrightarrow{k \rightarrow \infty} \int_0^1 dx \delta(x - F^{(k)}(x)) \quad (1.13)$$

La fonction de Dirac est facilement évaluable :

$$\begin{aligned} \int_0^1 d\xi \delta(\xi) &= 1 \text{ si } \xi = 0 \\ &= 0 \text{ si } \xi \neq 0 \end{aligned}$$

En posant  $\xi = x - F^{(k)}(x)$  et  $d\xi = \left|1 - \frac{dF^{(k)}(x)}{dx}\right| dx$ , on remarque que l’intégrale de l’équation 1.13 est différente de 0 partout où  $\xi = 0$  (soit sur les orbites périodiques de

l'application logistique) :

$$\begin{aligned} \int_0^1 dx \delta(x - F^{(k)}(x)) &= \int_0^1 \frac{d\xi \delta(\xi)}{\left|1 - \frac{dF^{(k)}(x)}{dx}\right|} \\ &= \lim_{k \rightarrow \infty} \sum_{\substack{OP \text{ de} \\ F^{(k)}}} \frac{1}{\left|1 - \frac{dF^{(k)}(x)}{dx}\right|_{OP}} \end{aligned} \quad (1.14)$$

La dernière somme se fait uniquement sur les orbites périodiques ( $OP$ ) de  $F^{(k)}$ . Les orbites périodiques font ici référence à tous les points  $x_i$  qui satisfont l'équation  $\xi = 0$  soit  $x_i = F^{(k)}(x_i)$ . Il s'agit donc de compter toutes les orbites de période- $m_f$ ,  $m_f$  étant un facteur de  $k$ . Par exemple pour  $k = 4$ , on retrouve dans l'ensemble des  $x_i$ , les périodes-1, les 2 composantes des périodes-2 et les 4 composantes des périodes-4.

La dérivée de  $F^{(k)}(x)$  qui apparaît au dénominateur doit être évaluée sur les différentes orbites périodiques. Nous avons déjà défini cette dérivée pour une orbite de période- $m$  : il s'agit de  $J^{(m)}$  (équation 1.9). Pour une période- $m_f$ , où  $m_f$  est facteur de  $k$  avec  $k = r \cdot m_f$  on obtient :

$$\begin{aligned} \frac{dF^{(k)}(x)}{dx} \Big|_{OP} &= \frac{dF^{(r \cdot m_f)}(x)}{dx} \Big|_{OP} \\ &= \prod_{i=1}^r \frac{dF^{(m_f)}(x)}{dx} \Big|_{OP} \\ &= (J^{(m_f)})^r \quad (\text{réf. équation 1.9}) \end{aligned}$$

De plus, chacune des composantes de la période- $m_f$  ( $m_f$  composantes) contribue à l'ajout du terme  $1/|1 - (J^{(m_f)})^r|$  dans la somme. Avec toutes ces considérations, il est possible d'écrire la somme de l'équation 1.14 comme :

$$\lim_{k \rightarrow \infty} \sum_{\substack{OP \text{ de} \\ F^{(k)}}} \frac{1}{\left|1 - \frac{dF^{(k)}}{dx}\right|_{OP}} = \lim_{k \rightarrow \infty} \sum_{\substack{OP \text{ de} \\ \text{periode-}m_f}} m_f \sum_{r=1}^{\infty} \frac{\delta_{k,r \cdot m_f}}{\left|1 - (J^{(m_f)})^r\right|} \quad (1.15)$$

La fraction des “survivants”  $\Gamma_k$  après  $k$  itérations est donc directement reliée à la stabilité des orbites périodiques. En regroupant les équations 1.13, 1.14 et 1.15, on obtient, dans la limite où  $k \rightarrow \infty$  :

$$\Gamma_{k_\infty} \equiv \lim_{k \rightarrow \infty} \Gamma_k = \lim_{k \rightarrow \infty} \sum_{\substack{OP \text{ de} \\ \text{periode-}m_f}} m_f \sum_{r=1}^{\infty} \frac{\delta_{k,r \cdot m_f}}{\left|1 - (J^{(m_f)})^r\right|}$$

Sous cette dernière forme, cette somme est plus ou moins pratique étant donné la présence du delta de Kronecker. Il est possible de le supprimer en prenant la transformée de Laplace (sous forme discrète) de la fonction  $\Gamma_k$ . On définit ainsi une nouvelle fonction  $\Gamma(z)$  dont l'utilité deviendra apparente sous peu :

$$\Gamma(z) \equiv \sum_{k=1}^{\infty} z^k \Gamma_{k\infty} \approx \sum_{\substack{OP \text{ de} \\ \text{periode-}m_f}} m_f \sum_{r=1}^{\infty} \frac{z^{m_f r}}{|1 - (J^{(m_f)})^r|} \quad (1.16)$$

Comme  $\Gamma_{k\infty} \rightarrow e^{-\gamma k}$  pour  $k \rightarrow \infty$  (ref. équation 1.11), le membre de gauche de l'équation 1.16 devient :

$$\sum_{k=1}^{\infty} z^k \Gamma_{k\infty} \approx \sum_{k=1}^{\infty} z^k e^{-\gamma k} = \frac{ze^{-\gamma}}{1 - ze^{-\gamma}}$$

Ainsi, on obtient une nouvelle fonction qui relie la stabilité des OPIs (les  $J^{(m_f)}$ ) à la quantité  $\gamma$  que l'on cherche à évaluer :

$$\frac{ze^{-\gamma}}{1 - ze^{-\gamma}} = \sum_{\substack{OP \text{ de} \\ \text{periode-}m_f}} m_f \sum_{r=1}^{\infty} \frac{z^{m_f r}}{|1 - (J^{(m_f)})^r|} \quad (1.17)$$

Cette dernière équation <sup>2</sup> justifie la définition de la fonction  $\Gamma(z)$ . En effet, en examinant la partie de gauche de l'équation 1.17, on constate qu'elle diverge pour une valeur particulière de  $z$  soit  $z^* = e^\gamma$ . C'est précisément ce qui nous intéresse ! Si on trouve le premier point  $z^*$  où le membre de droite de l'équation 1.17 diverge (membre qui fait intervenir la stabilité des OPIs que l'on suppose connue), on en déduira la valeur de  $\gamma$  soit  $\gamma = \ln z^*$ .

Avant d'aller plus loin, mentionnons que l'équation 1.17 a été obtenue pour un exemple bien précis soit du calcul d'un taux d'échappement  $\gamma$ . Nous avons choisi cet exemple car il permet d'aboutir, à partir de quelques idées sur la statistique, sur le lien fondamental qui existe entre un observable et la stabilité des OPIs dans un système chaotique. Cette dérivation n'est que notre résumé d'une théorie beaucoup plus générale. La double intégrale de l'équation 1.12 porte le nom *d'opérateur densité*. Dans un contexte plus

---

<sup>2</sup>Rigoureusement, il faudrait remplacer l'égalité de l'équation 1.17 par “approximativement égal”.

global, cet opérateur pondère l'*observable* dont on cherche la moyenne. La *trace* de cet opérateur, qui fait apparaître une double somme semblable au membre de droite de l'équation 1.17, permet d'obtenir la moyenne de l'observable en question. On réfère au livre de Cvitanovic pour plus de détails [18].

On revient maintenant à l'équation 1.17. Comme nous l'avons mentionné, il suffit de chercher le premier pôle de la fonction pour connaître le taux d'échappement  $\gamma$ . On remarque que ces pôles correspondent aux endroits où le dénominateur  $1 - ze^{-\gamma}$  devient égal à zéro. Comme il est toujours plus simple de rechercher des zéros que d'extraire des pôles, on modifie l'équation 1.17 afin de relier le dénominateur  $1 - ze^{-\gamma}$  à la stabilité des OPIs. En utilisant l'identité :

$$-z \frac{d}{dz} \ln(1 - ze^{-\gamma}) = \frac{ze^{-\gamma}}{1 - ze^{-\gamma}}$$

On écrit :

$$\begin{aligned} \ln(1 - ze^{-\gamma}) &= - \int dz \left( \frac{1}{z} \right) \frac{ze^{-\gamma}}{1 - ze^{-\gamma}} \\ &= - \int dz \sum_{\substack{OP \text{ de} \\ \text{periode-}m_f}} m_f \sum_{r=1}^{\infty} \frac{z^{m_f r - 1}}{|1 - (J^{(m_f)})^r|} \\ &= - \sum_{\substack{OP \text{ de} \\ \text{periode-}m_f}} \sum_{r=1}^{\infty} \frac{1}{r} \frac{z^{m_f r}}{|1 - (J^{(m_f)})^r|} + K \end{aligned}$$

où  $K$  est une constante d'intégration. On peut donc ramener le dénominateur (que l'on notera  $D(z)$ ) de l'équation 1.17 à :

$$D(z) \equiv 1 - ze^{-\gamma} = K' \exp \left\{ - \sum_{\substack{OP \text{ de} \\ \text{periode-}m_f}} \sum_{r=1}^{\infty} \frac{1}{r} \frac{z^{m_f r}}{|1 - (J^{m_f})^r|} \right\} \quad (1.18)$$

Le premier zéro de cette dernière équation nous donne le coefficient  $\gamma$  recherché. Mentionnons que cette dernière équation peut être réécrite de différentes façons. L'une d'elle fait apparaître une fonction caractéristique de l'observable recherché. Cette fonction porte le nom de *fonction Zeta dynamique*  $\zeta(z)$  (Dynamical zeta function), qui a été introduite par Ruelle [86]. Nous n'avons pas à aller dans ces détails puisqu'une sim-

ple série de Taylor nous permet d'évaluer le premier zéro de l'équation 1.18. On réfère cependant à l'article de Artuso & al. [1] pour plus d'informations sur la fonction  $\zeta$ .

On termine finalement le calcul de taux d'échappement  $\gamma$  en évaluant le premier zéro de l'équation 1.18. Pour ce faire, on écrit les 2 sommes dans l'exponentielle du membre de droite comme une somme de puissances en  $z^i$  et on développe ensuite en série de Taylor :

$$\begin{aligned} \exp \left\{ - \sum_{\substack{\text{OP de} \\ \text{periode-}m_f}} \sum_{r=1}^{\infty} \frac{1}{r} \frac{z^{m_f r}}{|1 - (J^{m_f})^r|} \right\} &\equiv \exp \left\{ - \sum_{i=1}^{\infty} b_i z^i \right\} \\ &= 1 - \left( \sum_{i=1}^{\infty} b_i z^i \right) + \frac{1}{2!} \left( \sum_{i=1}^{\infty} b_i z^i \right)^2 - \dots \quad (1.19) \\ &= 1 - b_1 z + \left( \frac{1}{2} b_1^2 - b_2 \right) z^2 - \left( \frac{1}{6} b_1^3 - b_1 b_2 + b_3 \right) z^3 + \dots \end{aligned}$$

avec les  $b_i$  donnés par :

$$\begin{aligned} b_1 &= \sum_{\substack{\text{OP de} \\ \text{periode-1}}} \frac{1}{|1 - (J^{(1)})^1|} \\ b_2 &= \sum_{\substack{\text{OP de} \\ \text{periode-1}}} \frac{1}{2} \frac{1}{|1 - (J^{(1)})^2|} + \sum_{\substack{\text{OP de} \\ \text{periode-2}}} \frac{1}{|1 - (J^{(2)})^1|} \\ b_3 &= \sum_{\substack{\text{OP de} \\ \text{periode-1}}} \frac{1}{3} \frac{1}{|1 - (J^{(1)})^3|} + \sum_{\substack{\text{OP de} \\ \text{periode-3}}} \frac{1}{|1 - (J^{(3)})^1|} \\ b_4 &= \sum_{\substack{\text{OP de} \\ \text{periode-1}}} \frac{1}{4} \frac{1}{|1 - (J^{(1)})^4|} + \sum_{\substack{\text{OP de} \\ \text{periode-2}}} \frac{1}{2} \frac{1}{|1 - (J^{(2)})^2|} + \sum_{\substack{\text{OP de} \\ \text{periode-4}}} \frac{1}{|1 - (J^{(4)})^1|} \\ &\vdots \end{aligned}$$

En regroupant les équations 1.18 et 1.19 on obtient :

$$D(z) \equiv 1 - ze^{-\gamma} = 1 - b_1 z + \left( \frac{1}{2} b_1^2 - b_2 \right) z^2 - \left( \frac{1}{6} b_1^3 - b_1 b_2 + b_3 \right) z^3 + \dots \quad (1.20)$$

Le premier  $z^*$  pour lequel  $D(z^*) = 0$  correspond à  $z^* = e^\gamma$ . Ainsi :

$$\gamma = \ln(z^*)$$

Le tableau 1.1 donne la position des OPIs et leur stabilité (i.e. le Jacobien  $J^{(m)}$ ) pour l'application logistique avec  $R = 5$  (Ces OPIs ont été repérées par un des algorithmes

présentés à la prochaine section). Le tableau 1.2 contient la valeur des coefficients  $b_i$  du développement 1.19. Dans ce même tableau, on donne les valeurs de  $z^*$  et  $\gamma$  obtenues à partir de la série de puissance en  $z^i$  à un ordre donné. On remarque que le coefficient  $\gamma$  converge vers 0.55276... ce qui est identique à l'estimé fait de façon statistique (réf. pente du graphique 1.5). On peut obtenir une bonne précision en allant vers des ordres supérieurs.

Ceci complète notre survol du rôle important et fondamental qu'occupe les OPIs dans le processus d'évaluation des propriétés globales d'une dynamique chaotique. Nous passons maintenant à une présentation des algorithmes de détection qui sont les objets d'étude de cette thèse.

Période	Première composante	Stabilité
1	$x(1) = 0.8$	$J^{(1)} = -3.0$
1	$x(1) = 0$	$J^{(1)} = 5$
2	$x(1) = 0.2535898384862245$	$J^{(2)} = -11.0$
3	$x(1) = 0.1805630191413292$	$J^{(3)} = 35.42640687$
3	$x(1) = 0.0575620544982973$	$J^{(3)} = -49.42640687$
4	$x(1) = 0.2063024311116012$	$J^{(4)} = -103.54620088$
4	$x(1) = 0.0366524739349423$	$J^{(4)} = 167.50759065$
4	$x(1) = 0.0118376682324109$	$J^{(4)} = -240.96138977$

TAB. 1.1 – Positions et stabilité des premières OPIs de l'application logistique ( $R = 5$ ).

Coefficients $b_i$	Polynôme $D(z)$	$z^*$ et $\gamma$
$b_1 = 0.5$	$D(z) = 1 - \frac{1}{2}z$	$z^* = 2.0 ; \gamma = 0.69314718$ (ordre 1)
$b_2 = 1/6$	$- \frac{1}{24}z^2$	$z^* = 1.74597\dots ; \gamma = 0.55730838$ (ordre 2)
$b_3 = .06347129$	$-0.00097128z^3$	$z^* = 1.73806\dots ; \gamma = 0.55277087$ (ordre 3)
$b_4 = .02739608$	$-7.1613817e-7 z^4$	$z^* = 1.73805\dots ; \gamma = 0.55276511$ (ordre 4)

TAB. 1.2 – Premiers coefficients de  $D(z)$  (équation 1.20), première racine  $z^*$  et coefficient  $\gamma$  correspondant (Application logistique  $R = 5$ ).

## 1.3 Détection d'orbites périodiques : Théorie

Dans cette section, nous formulons la problématique associée à la détection d'OPIs de façon plus générale. Nous présentons ensuite quelques méthodes connues pour effectuer cette tâche. Deux de ces méthodes sont expliquées et analysées de façon plus approfondie.

### 1.3.1 Formulation générale du problème

Un système dynamique discret peut en général se ramener à une application discrète  $\mathbf{v}_n \mapsto \mathbf{v}_{n+1}$ . Le vecteur  $\mathbf{v}_n$  fait ici référence à l'état du système à la  $n^{ième}$  itération. L'application logistique et l'application de Hénon dont nous avons fait mention dans les sections précédentes sont des exemples d'applications discrètes. Dans le cas de la logistique,  $\mathbf{v}$  n'a qu'une composante. Pour Hénon,  $\mathbf{v}$  en a deux. En général, le vecteur aura  $D$  composantes et on dit alors que l'espace des phases a  $D$  dimensions. Dans certains cas, les  $\mathbf{v}_n$  s'obtiennent en itérant une fonction connue analytiquement (comme par exemple l'équation 1.1). Dans d'autres cas, notamment pour les systèmes dynamiques obtenus par l'intégration numérique d'équations différentielles, les  $\mathbf{v}_n$  correspondent au zéro d'une fonction particulière. Nous verrons les différents cas possibles au chapitre suivant.

On considère donc pour l'instant une application dynamique à  $D$ -dimensions de la forme  $\mathbf{v}_{n+1} = \mathbf{F}(\mathbf{v}_n)$ , où  $\mathbf{F}$  est une fonction connue analytiquement ou encore obtenue numériquement. Pour cette application, les orbites de période- $m$  correspondent aux zéros de la fonction  $\mathbf{G}(\mathbf{v})$  définie par :

$$\mathbf{G}(\mathbf{v}) \equiv \mathbf{F}^{(m)}(\mathbf{v}) - \mathbf{v}, \quad (1.21)$$

où  $\mathbf{F}^{(m)}$  représente  $m$  itérations de l'application  $\mathbf{F}$ . Chaque composante d'une orbite de

période- $m$  est une solution à l'équation 1.21. Ces composantes sont notées  $\mathbf{v}(k)$  avec :

$$\mathbf{v}(k+1) = \mathbf{F}(\mathbf{v}(k)). \quad (1.22)$$

La stabilité du cycle s'obtient en étudiant l'évolution d'une perturbation  $\delta\mathbf{v}$  ajoutée à une composante  $\mathbf{v}(k)$  de l'orbite périodique. En développant l'équation 1.21 autour d'une des composantes  $\mathbf{v}(k)$ , on obtient :

$$\delta\mathbf{v}_{n+m} = \mathbf{J}^{(m)}(\mathbf{v}(k)) \delta\mathbf{v}_n, \quad (1.23)$$

où  $\mathbf{J}^{(m)}(\mathbf{v}(k))$  représente la matrice jacobienne de  $\mathbf{F}^{(m)}$  évaluée sur la composante  $\mathbf{v}(k)$  de l'orbite périodique considérée :

$$\begin{aligned} \mathbf{J}^{(m)}(\mathbf{v}(i)) &= D_{\mathbf{v}}\mathbf{F}^{(m)}(\mathbf{v}) \Big|_{OP} \\ &= \underbrace{D_{\mathbf{v}}\mathbf{F}(\mathbf{v}) \Big|_{\mathbf{v}(k-1)} \cdots \mathbf{F}(\mathbf{v}) \Big|_{\mathbf{v}(1)} D_{\mathbf{v}}\mathbf{F}(\mathbf{v}) \Big|_{\mathbf{v}(m)} \cdots D_{\mathbf{v}}\mathbf{F}(\mathbf{v}) \Big|_{\mathbf{v}(k+1)} D_{\mathbf{v}}\mathbf{F}(\mathbf{v}) \Big|_{\mathbf{v}(k)}}_{\text{Produit de } m \text{ matrices}}. \end{aligned} \quad (1.24)$$

Contrairement à l'équation 1.9, le produit de ces  $m$  matrices n'est pas commutatif, il y a donc  $m$  matrices différentes, soit une pour chaque composante de l'orbite. Par contre, ces  $m$  matrices ont toutes le même déterminant (Jacobien) et les mêmes valeurs propres  $\rho_d$ . La condition de stabilité s'écrit en fonction des ces valeurs propres :

$\text{OP stable de période-}m :  \rho_d  < 1 \quad [\forall d \in \{1, \dots, D\}]$ $\text{OP instable de période-}m :  \rho_d  > 1 \quad [\text{si } \exists d \in \{1, \dots, D\}]$	(1.25)
---	--------

La tâche de tout algorithme de détection revient, en fait, à trouver les zéros de  $\mathbf{G}$  (équation 1.21). On distingue dans la littérature deux approches bien différentes. La première utilise la *dynamique symbolique*. Il faut, dans un premier temps, construire une suite de symboles qui représente la dynamique du système pour lequel on recherche les OPIs. Par exemple, on pourrait diviser judicieusement l'espace des phases en plusieurs régions et assigner une lettre à chacune de ces régions : l'évolution du système serait alors décrit par un “alphabet” correspondant aux régions visitées. Chaque OPI se verrait ainsi assigner une séquence symbolique particulière. La recherche des OPIs se ferait

sur une base de critères reliés à l'évolution des symboles. Cette technique très astucieuse a été utilisée pour quelques systèmes chaotiques, notamment : l'application de Hénon [8], l'application standard dissipative [106] et les billards [7] [98]. Cette méthode n'est cependant pas assez générale pour être utilisée sur nombre de systèmes chaotiques connus.

Une autre approche qui peut être envisagée pour la recherche des *zéros* de la fonction  $\mathbf{G}$  consiste à choisir arbitrairement une condition initiale  $\mathbf{v}_0$  et de la modifier graduellement jusqu'à ce que la condition  $\mathbf{G}(\mathbf{v}_0) = 0$  soit remplie. Cette approche est celle que nous privilégions dans cette recherche. La procédure se résume par deux points importants :

- I – Choisir une condition initiale  $\mathbf{w}_n$
  - II – Modifier cette condition selon  $\mathbf{w}_{n+1} = \mathbf{w}_n + \delta\mathbf{w}_n$  jusqu'à ce que  $\mathbf{G}(\mathbf{w}_{n+1}) = 0$ .
- (1.26)

Chaque méthode numérique que nous allons étudier aura ses propres règles pour apporter la correction  $\delta\mathbf{w}$ . Nous utilisons le symbole  $\mathbf{w}$  pour insister sur le fait que, peu importe la méthode utilisée, l'itération de  $\mathbf{w}_{n+1} = \mathbf{w}_n + \delta\mathbf{w}_n$  est en soi un *nouveau* système dynamique qui peut avoir des comportements aussi complexes que le système initial évoluant selon  $\mathbf{F}(\mathbf{v}_n)$ . Ce nouveau système dynamique doit être choisi tel que la correction  $\delta\mathbf{w}$  s'annule lorsque  $\mathbf{w}_n$  est égal à n'importe quelle composante  $\mathbf{v}(k)$  des orbites de  $\mathbf{F}$ .

Une des méthodes numériques bien connue qui utilise le schème particulier décrit par l'équation 1.26 est l'algorithme de Newton-Raphson (*NR*) [76]. Pour cet algorithme, les corrections  $\delta\mathbf{w}_n$  apportées à l'estimé en question se font à partir de :

$$\delta\mathbf{w}_n^{NR} = -\mathbf{J}_G^{-1}(\mathbf{w}_n)\mathbf{G}(\mathbf{w}_n), \quad (1.27)$$

où  $\mathbf{J}_G$  est la matrice jacobienne de  $\mathbf{G}$ . Cette matrice est évaluée au point  $\mathbf{w}_n$ . On remarque que la correction  $\delta\mathbf{w}^{NR} = 0$  lorsque  $\mathbf{w}_n$  est égale à une composante  $\mathbf{v}(k)$  de

l'orbite périodique (par définition  $\mathbf{G}(\mathbf{v}(k)) = 0$ ). Cet algorithme converge rapidement vers la solution si l'estimé de départ est convenable.

Auerbach et al. [3] ont proposé une méthode par *récurrence* afin d'obtenir un estimé initial de la position des OPIs. On doit avoir en main une séquence de  $\{\mathbf{v}_n\}$ , générée à l'aide d'une application  $\mathbf{F}(\mathbf{v}_n)$ . Pour une orbite de période- $m$ , on mesure la distance entre  $\mathbf{v}_n$  et  $\mathbf{v}_{n+m}$ . Si cette distance est inférieure à  $\epsilon$ , on soupçonne ces deux points d'être dans le voisinage d'une orbite de période- $m$ . Si la séquence de  $\mathbf{v}_n$  est suffisamment longue, on peut se permettre de choisir  $\epsilon$  petit et ainsi augmenter la qualité de l'estimé de départ. Typiquement, pour un voisinage de volume  $V \propto \epsilon^\alpha$  autour de l'OPI, on doit attendre  $N \approx \epsilon^{-\alpha}$  itérations pour un passage dans ce voisinage. Le symbole  $\alpha$  représente ici la dimension “point” de l'OPI ([4],[30], [68]).

Cette méthode s'avère toutefois moins efficace pour les périodes élevées (car plus nombreuses et plus denses). De plus, certains systèmes dynamiques peuvent mettre un temps relativement long avant de revenir dans une région de l'espace des phases déjà visitée, ce qui demande des séquences de  $\mathbf{v}_n$  extrêmement longues. La dynamique mixte des systèmes hamiltoniens présente souvent ce genre de comportement ; la trajectoire des  $\mathbf{v}_n$  peut être piégée autour de quelques îlots de stabilité avant de visiter l'espace des phases dans son ensemble ([99],[5],[57]). La méthode par récurrence, combinée à la correction  $\delta\mathbf{w}^{NR}$  prescrite par Newton-Raphson, bien que très efficace pour les petites périodes, devient inutilisable pour une recherche systématique de *toutes* les OPIs de périodes élevées.

Vers la fin des années ‘90, un nouvel algorithme de détection d'OPIs a été publié par Schmelcher et Diakonos ([89],[90]). Leur méthode suit toujours le schème général présenté à l'équation 1.26, cependant la correction  $\delta\mathbf{w}_n$  est tout-à-fait originale ! Elle “force” le système  $\mathbf{w}_{n+1}$  à converger vers les *zéros* de  $\mathbf{G}$ . Bien qu'ayant été initialement présenté pour une application dissipative en deux dimensions, leur algorithme est général et peut être utilisé pour d'autres classes de systèmes. Quelques années plus tard,

une autre équipe de chercheurs (Davidchack et Lai [19]) a proposé une correction  $\delta\mathbf{w}_n$  qui combine à la fois celles prescrites par Newton-Raphson et Schmelcher-Diakonos.

Les prochaines sections de ce chapitre seront consacrées à présenter en détails les algorithmes de Schmelcher-Diakonos et de Davidchack-Lai. Nous nous efforcerons de faire ressortir les différences entre ces 2 méthodes.

### 1.3.2 L’algorithme de Schmelcher-Diakonos (*SD*)

#### 1.3.2.1 Idée générale de l’algorithme *SD*

On sait que les orbites périodiques recherchées sont instables dans le système chaotique décrit par  $\mathbf{F}(\mathbf{v}_n)$ . Serait-il possible de construire un nouveau système dynamique qui modifierait la stabilité des orbites sans changer leur position dans l'espace des phases ? Autrement dit, existe-t-il une correction  $\delta\mathbf{w}$  qui puisse forcer l’itération 1.26 à converger vers une composante  $\mathbf{v}(k)$  des orbites de  $\mathbf{F}$  ? Schmelcher & Diakonos ont répondu par l'affirmative à ces questions et ont été les premiers à proposer cette transformation fort astucieuse. Dans le système ainsi transformé, chaque condition initiale peut converger vers une orbite périodique. Il faut itérer un nombre suffisant de ces conditions initiales afin de détecter l'ensemble des cycles.

La transformation ne doit pas modifier la position des orbites périodiques et doit en générale être linéaire afin qu'il y ait une correspondance biunivoque entre les orbites du système initial et transformé. Si le système chaotique initial  $\mathcal{V}$  est donné par :

$$\mathcal{V} : \mathbf{v}_{n+1} = \mathbf{F}(\mathbf{v}_n) \quad (1.28)$$

alors le système transformé  $\mathcal{W}_k$  doit avoir (pour la recherche d'orbites de période- $m$ ),

la forme suivante :

$$\begin{aligned}\mathcal{W}_k : \mathbf{w}_{n+1} &= \mathbf{w}_n + \lambda \mathbf{C}_k [\mathbf{F}^{(m)}(\mathbf{w}_n) - \mathbf{w}_n] \\ &= \mathbf{w}_n + \lambda \mathbf{C}_k \mathbf{G}(\mathbf{w}_n) \\ &= \mathbf{w}_n + \delta \mathbf{w}_n^{SD},\end{aligned}\tag{1.29}$$

où  $\mathbf{C}_k$  est une matrice  $D \times D$  possédant un inverse et  $0 < \lambda \ll 1$ . On peut vérifier aisément que cette transformation ne modifie pas la position des orbites périodiques. Pour reprendre la formulation générale du schème de détection proposé à la section précédente (équation 1.26), on voit que la correction  $\delta \mathbf{w}$  selon l'algorithme *SD* s'écrit comme :

$$\delta \mathbf{w}_n^{SD} = \lambda \mathbf{C}_k \mathbf{G}(\mathbf{w}_n).\tag{1.30}$$

Il faut maintenant choisir convenablement la matrice  $\mathbf{C}_k$  afin de modifier la stabilité des cycles périodiques dans le système  $\mathcal{W}_k$ . Les propriétés de stabilité des orbites s'obtiennent à partir de leur matrice jacobienne (réf. équation 1.25). En utilisant les symboles  $\mathbf{J}^{(m)}$  et  $\mathbf{J}_{\mathcal{W}_k}$  pour représenter les matrices jacobienes de l'orbite dans les systèmes  $\mathcal{V}$  et  $\mathcal{W}_k$  respectivement, on peut écrire, à partir de l'équation 1.29 :

$$\begin{aligned}\mathbf{J}_{\mathcal{W}_k} &= \mathbf{1} + \lambda \mathbf{C}_k (\mathbf{J}^{(m)} - \mathbf{1}) \\ &= \mathbf{1} + \lambda \mathbf{A}_k,\end{aligned}\tag{1.31}$$

où nous avons posé  $\mathbf{A}_k \equiv \mathbf{C}_k (\mathbf{J}^{(m)} - \mathbf{1})$ . Si les orbites périodiques sont stables dans le système  $\mathcal{W}_k$ , cela implique que les valeurs propres de  $\mathbf{J}_{\mathcal{W}_k}$  sont *toutes* de module inférieur à 1 :

$$|\rho_d^{\mathcal{W}_k}| = |1 + \lambda \rho_d^{A_k}| < 1 \quad [\forall d \in \{1, \dots, D\}]\tag{1.32}$$

où les  $\rho_d^{A_k}$  et  $\rho_d^{\mathcal{W}_k}$  correspondent à l'ensemble des valeurs propres de  $\mathbf{A}_k$  et  $\mathbf{J}_{\mathcal{W}_k}$  respectivement. La matrice  $\mathbf{C}_k$  doit ainsi être choisie afin de rendre la partie réelle des valeurs propres de  $\mathbf{A}_k$  négatives. On peut facilement comprendre cette dernière affirmation si on écrit les valeurs propres de  $\mathbf{J}_{\mathcal{W}_k}$  en fonction de celles de  $\mathbf{A}_k$  :

$$|\rho_d^{\mathcal{W}_k}| = \sqrt{\left[1 + \lambda \cdot \operatorname{Re}\{\rho_d^{A_k}\}\right]^2 + \left[\lambda \cdot \operatorname{Im}\{\rho_d^{A_k}\}\right]^2}.$$

Sous cette dernière forme, il est clair que si les parties réelles des valeurs propres de  $\mathbf{A}_k$  sont toutes négatives, on peut toujours, en ajustant  $\lambda$ , faire en sorte que le module de celles de  $\mathbf{J}_{\mathcal{W}_k}$  soit inférieur 1. Si tel est le cas, la matrice jacobienne  $\mathbf{J}_{\mathcal{W}_k}$  représente une orbite stable.

Il est possible de choisir une classe restreinte de matrices  $\mathbf{C}_k$ , qui à elle seule, stabilise l'ensemble des orbites périodiques du système. Cette classe correspond à des transformations particulières de réflexion et de rotation dans l'espace des phases. On construit ces matrices en choisissant des éléments  $c_{ij} \in \{0, \pm 1\}$  de sorte qu'un seul élément par ligne ou par colonne soit différent de 0. Ces matrices doivent être orthogonales entre elles et posséder la propriété d'être leur inverse ( $\mathbf{C}_k * \mathbf{C}_k = \mathbf{1}$ )<sup>3</sup>. Pour un système en  $D$  dimensions, on peut construire  $2^D D!$  matrices avec la première contrainte ( $c_{ij} \in \{0, \pm 1\}$ ), mais certaines de ces matrices n'ont pas la propriété d'être leur inverse ; elles ne sont pas utiles dans la transformation. Par exemple, pour une dimension  $D = 2$ , on peut construire 8 matrices mais seulement 6 ont la propriété  $\mathbf{C}_k * \mathbf{C}_k = \mathbf{1}$ .

Nous ne ferons pas la preuve de ces dernières affirmations. Cependant, nous allons traiter un cas général en deux dimensions dans la prochaine section pour se convaincre que ce choix de matrice  $\mathbf{C}_k$  est convenable. L'idée d'un tel exercice est aussi de décortiquer la transformation dans les moindres détails afin d'en faire ressortir ses propriétés et ses particularités.

### 1.3.2.2 La transformation $SD$ pour un système quelconque à 2 dimensions

Nous allons maintenant étudier en détail comment la transformation proposée par  $SD$  se comporte pour un système quelconque à deux dimensions. En particulier, nous allons voir que la classe de matrices  $\mathbf{C}_k$  proposée dans la dernière section est suffisante

---

<sup>3</sup>Dans l'article de [90], il est démontré que le groupe de matrices  $\mathbf{C}_k$  qui a la propriété d'être leur inverse stabilise l'ensemble des OPIs du système.

pour retrouver toutes les OPIs du système. Nous montrons comment un choix approprié de matrices  $C_k$  peut retrouver une famille d'OPIs particulière et comment le paramètre  $\lambda$  peut intervenir pour “contrôler” le degré d'instabilité de l'orbite qui sera détectée.

**Choix et influence des matrices  $C_k$**  Soit le système  $\mathcal{V}$  en deux dimensions donné par :

$$\begin{aligned} v_{n+1}^1 &= f(v_n^1, v_n^2), \\ v_{n+1}^2 &= g(v_n^1, v_n^2). \end{aligned} \quad (1.33)$$

La matrice jacobienne  $\mathbf{J}(\mathbf{v})$  du système en un point  $\mathbf{v}$  donné prend alors la forme :

$$\mathbf{J}(\mathbf{v}) = D_{\mathbf{v}}\mathbf{F}(\mathbf{v}) = \begin{pmatrix} \frac{\partial f}{\partial v^1} & \frac{\partial f}{\partial v^2} \\ \frac{\partial g}{\partial v^1} & \frac{\partial g}{\partial v^2} \end{pmatrix}_{\mathbf{v}}, \quad (1.34)$$

les dérivées étant évaluées au point  $\mathbf{v}$  en question. La matrice jacobienne de la composante  $\mathbf{v}(k)$  d'une orbite de période- $m$  peut ainsi s'écrire :

$$\begin{aligned} \mathbf{J}^{(m)}(\mathbf{v}(k)) &= \mathbf{J}(\mathbf{v}(k-1)) \cdots \mathbf{J}(\mathbf{v}(1)) \mathbf{J}(\mathbf{v}(m)) \cdots \mathbf{J}(\mathbf{v}(k+1)) \mathbf{J}(\mathbf{v}(k)) \\ &\equiv \begin{pmatrix} J_{11}^{(m)} & J_{12}^{(m)} \\ J_{21}^{(m)} & J_{22}^{(m)} \end{pmatrix}. \end{aligned} \quad (1.35)$$

On suppose cette matrice réelle. Il faut maintenant choisir convenablement les matrices  $\mathbf{C}_k$  afin de stabiliser les orbites périodiques de l'application par la transformation 1.29. Schmelcher et Diakonos proposent différentes matrices  $\mathbf{C}_k$  en fonction du type d'orbites recherché. En notant  $\rho_1$  et  $\rho_2$  les valeurs propres de la matrice jacobienne du cycle périodique qui sera détecté, on retrouve les résultats présentés dans le tableau 1.3. Les affirmations (i) à (iii) de ce tableau ont été proposées dans la papier original de Schmelcher et al.([90]), sans toutefois être démontrées. Nous complétons ces 3 affirmations par une 4<sup>ième</sup> qui concerne les orbites elliptiques (affirmation (iv)), orbites courantes dans les systèmes conservatifs. Les preuves pour ces affirmations sont présentées à l'annexe A.

	Types d'OPIs	Matrices $\mathbf{C}_k$ à utiliser
(i)	Orbites hyperboliques avec inversion : $\rho_1 < -1$ et $-1 < \rho_2 < 0$	$\Rightarrow \mathbf{C}_1 = \mathbf{1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$
(ii)	Orbites hyperboliques avec réflexion : $\rho_1 < -1$ et $0 < \rho_2 < 1$	$\Rightarrow \mathbf{C}_1 = \mathbf{1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$
(iii)	Orbites hyperboliques avec réflexion : $\rho_1 > 1$ et $-1 < \rho_2 < 0$ et orbites hyperboliques sans réflexion : $\rho_1 > 1$ et $0 < \rho_2 < 1$	<ul style="list-style-type: none"> <li>• Si <math>J_{11}^{(m)} &gt; J_{22}^{(m)}</math> :  <math>\Rightarrow \mathbf{C}_2 = \begin{pmatrix} -1 &amp; 0 \\ 0 &amp; 1 \end{pmatrix}</math></li> <li>• Si <math>J_{11}^{(m)} &lt; J_{22}^{(m)}</math> :  <math>\Rightarrow \mathbf{C}_3 = \begin{pmatrix} 1 &amp; 0 \\ 0 &amp; -1 \end{pmatrix}</math></li> <li>• Si <math>J_{11}^{(m)} = J_{22}^{(m)}</math> :  <math>(J_{12}^{(m)} + J_{21}^{(m)} &gt; 0) \Rightarrow \mathbf{C}_4 = \begin{pmatrix} 0 &amp; -1 \\ -1 &amp; 0 \end{pmatrix}</math>  <math>(J_{12}^{(m)} + J_{21}^{(m)} &lt; 0) \Rightarrow \mathbf{C}_5 = \begin{pmatrix} 0 &amp; 1 \\ 1 &amp; 0 \end{pmatrix}</math></li> </ul>
(iv)	Orbites elliptiques (Systèmes conservatifs seulement) $ \rho_{1,2}  = 1$ avec $ Im\{\rho_{1,2}\}  > 0$	$\Rightarrow \mathbf{C}_1 = \mathbf{1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$

TAB. 1.3 – Choix des matrices  $\mathbf{C}_k$  en 2D en fonction du type d'OPIs.

**Influence du paramètre  $\lambda$  dans la transformation  $SD$**  Le paramètre  $\lambda$  qui intervient dans la correction  $SD$  (équation 1.30) a aussi son rôle à jouer dans le processus de détection. Ce paramètre détermine en quelque sorte quelles orbites peuvent être détectées et ce, en fonction de leur stabilité. Par exemple, nous avons vu que la matrice  $C_1$  permet, entre autre, de stabiliser les OPIs avec  $\rho_1 < -1, -1 < \rho_2 < 0$  (tableau 1.3). Pour ce cas particulier, on peut écrire directement les valeurs propres de la matrice Jacobienne  $\mathbf{J}_{\mathcal{W}_k}$  (système transformé) en fonction de celles de  $\mathbf{J}^{(m)}$  (système initial). À partir de l'équation 1.31, on obtient pour  $C_k = C_1$  :

$$\rho_{1,2}^{\mathcal{W}_k} = 1 + \lambda(\rho_{1,2} - 1). \quad (1.36)$$

La condition de stabilité pour l'orbite du système transformé conduit alors à une valeur critique pour la grandeur de  $\rho_1$ , valeur au-delà de laquelle l'orbite ne peut être détectée :

$$|\rho_{1,2}^{\mathcal{W}_k}| < 1 \Rightarrow |\rho_{1,2}| < \frac{2}{\lambda} - 1. \quad (1.37)$$

Pour une valeur de  $\lambda$  donnée, on déduit donc que seules les orbites avec  $|\rho_{1,2}| < \frac{2}{\lambda} - 1$  pourront être stabilisées par la transformation. En diminuant  $\lambda$  graduellement, des orbites de plus en plus instables seront détectées. À l'inverse, on peut imposer la détection des OPIs les “moins instables” par un choix de  $\lambda$  suffisamment grand.

Bien qu'un raisonnement similaire peut être fait pour les autres matrices  $C_k$ , nous n'irons pas plus loin dans cette analyse et référons plutôt à l'article de Diakonos & al. [24] et de Pingel & al.[73] pour plus de détails.

### 1.3.2.3 Taux de convergence et nombre d'opérations par itération

Comme nous voulons éventuellement comparer l'algorithme  $SD$  à celui de  $DL$  sous la base de certains critères, il est maintenant approprié de jeter un coup d'oeil au taux de convergence de la transformation que nous venons de présenter.

Il est clair que dans le voisinage des composantes d'une orbite périodique, le taux de convergence de  $\mathbf{w}_{n+1}$  diminue linéairement avec le nombre d'itérations. On peut s'en convaincre en développant l'équation 1.29 autour d'une composante et en regardant comment varie, à chaque itération, la correction  $\delta\mathbf{w}_n^{SD}$  :

$$\delta\mathbf{w}_{n+1}^{SD} = \lambda \mathbf{C}_k \mathbf{J}_G \delta\mathbf{w}_n^{SD}. \quad (1.38)$$

Comme nous le verrons au chapitre 3, la convergence linéaire de cette méthode s'avère probablement son principal défaut.

Cette convergence linéaire est en partie due à la grande simplicité de la correction  $\delta\mathbf{w}^{SD}$ . Très peu d'opérations mathématiques doivent être effectuées à chaque pas. On peut évaluer, de façon approximative, ce nombre d'opérations (multiplications et additions confondues). En se reportant encore une fois à l'équation 1.29, on déduit que, pour un système à  $D$  dimensions, le nombre d'opérations par itération pour l'algorithme de  $SD$  ( $\equiv N_{op}^{SD}$ ) est :

$$N_{op}^{SD} = 2D + (\text{évaluation de } \mathbf{G}). \quad (1.39)$$

### 1.3.3 L'algorithme de Davidchack-Lai (DL)

#### 1.3.3.1 Idée générale de l'algorithme *DL*

Nous avons vu à la section précédente que la transformation de  $SD$  avait des propriétés de convergence globale. Cependant, cette méthode souffre aussi d'un taux de convergence relativement lent. La méthode proposée par Davidchack et Lai ([19],[20]) combine les propriétés globales de cette transformation avec celles d'une méthode de type Newton-Raphson, qui converge de façon quadratique. Il en résulte ainsi un algorithme hybride avec des propriétés intéressantes pour la détection d'OPIs.

Le processus itératif  $SD$  (équation 1.29) est en fait la méthode d'Euler [76] appliquée

à l'équation différentielle :

$$\frac{d\mathbf{w}}{d\lambda} = \mathbf{C}_k \mathbf{G}(\mathbf{w}), \quad (1.40)$$

avec un pas de temps  $h = \lambda$ . La méthode d'Euler est un schéma numérique d'intégration d'ordre 1, d'où la convergence linéaire de l'équation 1.29. L'idée de Davidchack et Lai est d'utiliser un schème d'intégration plus précis.

Les solutions stationnaires de l'équation 1.40 sont les zéros de  $\mathbf{C}_k \mathbf{G}(\mathbf{w})$  (les mêmes que ceux de  $\mathbf{G}(\mathbf{w})$ ). On a ainsi une correspondance biunivoque entre les solutions stationnaires de l'équation 1.40 et les orbites périodiques de l'équation 1.29. Une solution stationnaire  $\mathbf{w}^*$  de 1.40 est stable si toutes les valeurs propres de la matrice jacobienne de  $\mathbf{C}_k \mathbf{G}(\mathbf{w})$  évaluée en  $\mathbf{w}^*$  (soit la matrice  $\mathbf{A}_k$  définie en 1.31) ont leur partie réelle négative. Ainsi, une solution stationnaire de 1.40 est stable si et seulement si c'est une orbite périodique stable de 1.29 (revoir paragraphe qui suit l'équation 1.32). Si  $\mathbf{w}^*$  est une solution stationnaire stable de 1.40, alors pour tout schème numérique d'intégration convergent, (si  $\mathbf{w}_0$  est assez près de  $\mathbf{w}^*$  et le pas de temps suffisamment petit) les itérés convergent vers  $\mathbf{w}^*$ . Cependant, pour les problèmes dits raides (*stiff*) en analyse numérique, les méthodes explicites, telle la méthode d'Euler, sont rarement performantes, le pas de temps requis étant beaucoup trop petit pour des besoins pratiques. Pour des problèmes raides, ce qui est vraisemblablement le cas ici, on utilise plutôt des méthodes implicites. C'est ce type de méthode qui a été proposé par Davidchack et Lai. Leur point de départ est la méthode d'Euler implicite appliquée à 1.40 :

$$\mathbf{w}_{n+1} = \mathbf{w}_n + h \mathbf{C}_k \mathbf{G}(\mathbf{w}_{n+1}). \quad (1.41)$$

On considère ensuite l'approximation

$$\mathbf{G}(\mathbf{w}_{n+1}) \approx \mathbf{G}(\mathbf{w}_n) + \mathbf{J}_G(\mathbf{w}_n)[\mathbf{w}_{n+1} - \mathbf{w}_n],$$

où  $\mathbf{J}_G$  représente la matrice Jacobienne de  $\mathbf{G}$  évaluée au point  $\mathbf{w}_n$ . On reporte cette approximation dans 1.41 pour obtenir le schéma semi-implicite :

$$\mathbf{w}_{n+1} = \mathbf{w}_n + h \mathbf{C}_k [\mathbf{G}(\mathbf{w}_n) + \mathbf{J}_G(\mathbf{w}_n)[\mathbf{w}_{n+1} - \mathbf{w}_n]],$$

ce qui donne :

$$[\mathbf{1} - h\mathbf{C}_k \mathbf{J}_G(\mathbf{w}_n)][\mathbf{w}_{n+1} - \mathbf{w}_n] = h\mathbf{C}_k \mathbf{G}(\mathbf{w}_n),$$

et finalement :

$$\begin{aligned} \delta\mathbf{w}_n \equiv \mathbf{w}_{n+1} - \mathbf{w}_n &= [\mathbf{1} - h\mathbf{C}_k \mathbf{J}_G(\mathbf{w}_n)]^{-1} h \mathbf{C}_k \mathbf{G}(\mathbf{w}_n) \\ &= \left[ \frac{1}{h} \mathbf{1} - \mathbf{C}_k \mathbf{J}_G(\mathbf{w}_n) \right]^{-1} \mathbf{C}_k \mathbf{G}(\mathbf{w}_n). \end{aligned} \quad (1.42)$$

Davidchack-Lai ont proposé un pas de temps variable donné par  $h = 1/\beta g(\mathbf{w}_n)$ . Le scalaire  $g(\mathbf{w}_n) \equiv \|\mathbf{G}(\mathbf{w}_n)\| \geq 0$  qui est introduit correspond à la norme de  $\mathbf{G}(\mathbf{w}_n)$  au point  $\mathbf{w}_n$ . Il s'agit donc de la distance euclidienne entre le point  $\mathbf{w}_n$  et son  $m^{ième}$  itéré par la fonction  $\mathbf{F}$  (ref. équation 1.21). Le paramètre  $\beta$  est maintenant un paramètre ajustable et dont la valeur dépendra de la période- $m$  recherchée. Ainsi, on obtient une correction  $\delta\mathbf{w}_n^{DL}$  déterminée à partir de :

$$\delta\mathbf{w}_n^{DL} = [\mathbf{1}\beta g(\mathbf{w}_n) - \mathbf{C}_k \mathbf{J}_G(\mathbf{w}_n)]^{-1} \mathbf{C}_k \mathbf{G}(\mathbf{w}_n). \quad (1.43)$$

La matrice jacobienne  $\mathbf{J}_G(\mathbf{w}_n)$  est évaluée à partir de celle de l'application considérée soit  $\mathbf{J}(\mathbf{w}_n) = D_{\mathbf{w}}\mathbf{F}(\mathbf{w})|_{\mathbf{w}_n}$  :

$$\begin{aligned} \mathbf{J}_G(\mathbf{w}_n) &= \mathbf{J}^{(m)}(\mathbf{w}_n) - \mathbf{1} \\ &= \mathbf{J}(\mathbf{w}_{n+m-1}) \cdots \mathbf{J}(\mathbf{w}_{n+1})\mathbf{J}(\mathbf{w}_n) - \mathbf{1} \end{aligned} \quad (1.44)$$

Le schème proposé par Davidchack-Lai combine à la fois l'efficacité de la méthode de Newton-Raphson tout en conservant le caractère globale de la transformation de  $SD$ . Si le système est loin de l'orbite périodique alors le terme en  $\mathbf{C}_k \mathbf{J}_G(\mathbf{w})$  devient négligeable devant  $\mathbf{1}\beta g(\mathbf{w})$  et on retombe sur la correction  $SD$  (équation 1.30). À l'inverse, près de l'orbite,  $g(\mathbf{w}) \rightarrow 0$  et les corrections de  $DL$  tendent vers celles prescrites par une méthode de type Newton-Raphson (équation 1.27).

### 1.3.3.2 L'algorithme $DL$ sur un système particulier en 1 dimension

Afin de faire ressortir les particularités de l'algorithme de  $DL$ , nous l'utiliserons pour repérer quelques points fixes ( $m = 1$ ) d'un système en une dimension. Nous choisissons

le système particulier suivant :

$$F(x_n) = \cos(x_n^2) + x_n. \quad (1.45)$$

Les points fixes sont ainsi donnés par :

$$x^* = \pm\sqrt{\frac{\pi}{2}}, \pm\sqrt{\frac{3\pi}{2}}, \dots$$

Le Jacobien de cette fonction  $F$  particulière s'écrit :

$$J(x) = -2x \sin(x^2) + 1,$$

et la valeur propre des points fixes de  $F$ , s'obtient en évaluant  $J(x^*)$  :

$$\rho = \mp 2\sqrt{\frac{\pi}{2}} + 1, \mp 2\sqrt{\frac{3\pi}{2}} + 1, \dots$$

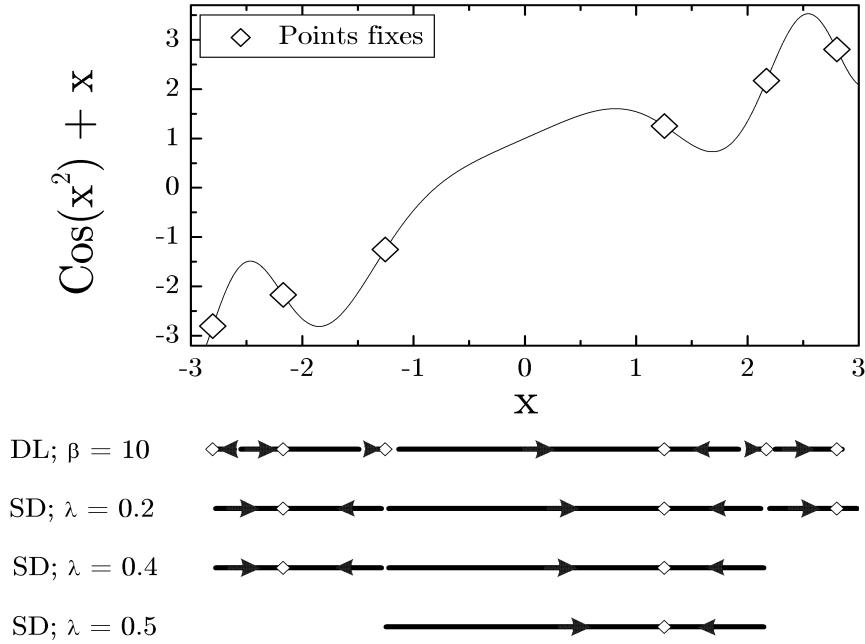
La figure 1.6 représente la fonction  $F(x)$  dans l'intervalle  $[-3, 3]$ . Nous avons placé les points fixes de  $F(x_n)$  repérés par l'algorithme *DL* sur un axe horizontal. Sur ce même axe, on retrouve également les bassins d'attraction du point fixe. À titre comparatif, nous présentons un axe horizontal similaire pour l'algorithme de *Schmelcher-Diakonos*. Les deux méthodes ont été appliquées en utilisant  $\mathbf{C}_k = \mathbf{C}_1 = 1$ .

Il est intéressant de noter que l'algorithme *SD* est incapable de repérer les points fixes à pente positive ( $\rho > 0$ ) en utilisant  $\mathbf{C}_1 = 1$ . Ces points pourraient être détectés en choisissant  $\mathbf{C}_2 = -1$  (il s'agit du cas (ii) présenté dans le tableau 1.3). Le paramètre  $\lambda$ , dans la méthode *SD*, doit être suffisamment petit afin de repérer les 3 points fixes à pente négative de l'intervalle  $[-3, 3]$ . L'équation 1.37 donne la limite de la valeur propre du point fixe qui peut être détecté en fonction de  $\lambda$  :

$$|\rho| < \frac{2}{\lambda} - 1.$$

En inversant cette relation, on obtient analytiquement les valeurs de  $\lambda$  critiques pour repérer les différents points fixes à pente négative :

$$x^* = \sqrt{\frac{\pi}{2}} \Rightarrow \lambda < 0.79788$$

FIG. 1.6 – Points fixes de  $\cos(x_n^2) + x_n$  et leurs bassins d’attraction.

$$x^* = \sqrt{\frac{3\pi}{2}} \Rightarrow \lambda < 0.46066$$

$$x^* = \sqrt{\frac{5\pi}{2}} \Rightarrow \lambda < 0.35682$$

Ce résultat est confirmé numériquement sur la figure. Pour  $\lambda = 0.5$ , le point fixe  $x^* = \sqrt{\frac{5\pi}{2}}$  ne peut être détecté par l'algorithme *SD*. En revanche, la méthode *DL* repère l'ensemble des points fixes sans avoir recours à la matrice  $\mathbf{C}_2$ . Par contre, il semble difficile d'établir un lien entre le paramètre  $\beta$  et l'instabilité des points fixes détectés.

### 1.3.3.3 Taux de convergence et nombre d'opérations par itération

La démonstration des propriétés dynamiques de l'algorithme de *DL*, en particulier sa convergence à la fois globale et quadratique ont été démontrées récemment de façon rigoureuse par Klebanoff et Bollt [49]. Nous référons donc à cet article pour les détails

de la preuve. Retenons pour l'instant que l'algorithme de *DL* converge de façon quadratique vers la solution  $\mathbf{G}(\mathbf{w}) = 0$  dû à la présence du Jacobien de  $\mathbf{G}$  dans la correction  $\delta\mathbf{w}_n^{DL}$ .

Évidemment, l'évaluation de  $\mathbf{J}_G$  doit se faire à chaque itération, ce qui rend l'algorithme de *DL* légèrement plus complexe à utiliser. Nous verrons ces différentes subtilités apparaître lors de l'étude numérique faite au prochain chapitre. Pour l'instant, on peut évaluer approximativement qu'à chaque correction  $\delta\mathbf{w}_n^{DL}$ , on doit effectuer un nombre d'opérations  $N_{op}^{DL}$  (multiplications et additions confondues) équivalent à :

$$N_{op}^{DL} \approx 4D(D + 1) + (\text{évaluation de } \mathbf{G}) + (\text{évaluation de } \mathbf{J}_G). \quad (1.46)$$

### 1.3.4 Résumé des algorithmes de *SD* et de *DL*

Nous résumons dans le tableau 1.4 les algorithmes de *SD* et de *DL* pour la recherche des orbites de période- $m$  d'un système à  $D$  dimensions.

*Remarque : Dans la version originale de leur publication, SD suggèrent :*

- De diminuer  $\lambda$  lorsque la période  $m$  de l'orbite recherchée augmente.
- De choisir la condition initiale  $\mathbf{w}_0$  aléatoirement sur l'attracteur.
- (Le nombre d'itération  $N_{max}$  pour une condition initiale et une matrice  $\mathbf{C}_k$  est au choix de l'opérateur).

*Remarque : Dans la version originale de leur publication, DL suggèrent :*

- D'augmenter  $\beta$  lorsque la période  $m$  de l'orbite recherchée augmente ( $\beta \propto e^m$ ).
- De choisir comme condition initiale  $\mathbf{w}_0$  les composantes des orbites de période  $m - 1$ .
- De choisir le nombre d'itération  $N_{max}$  de l'ordre de 4 à 6 fois  $\beta$ .

Il est à noter qu'aucun critère rigoureux ne peut être établi pour certifier que l'ensemble des OPIs de période- $m$  détectées par ces algorithmes est bien complet. L'algorithme de *SD* ne comporte pas de critère précis quant au nombre de conditions initiales à itérer. L'algorithme de *DL* utilise pour sa part les composantes des orbites de période  $m - 1$ . Le calcul s'arrête ainsi lorsque toutes ces composantes ont été itérées.

Algorithme de Schmelcher-Diakonos	Algorithme de Davidchack-Lai
<p><b>0.</b> Choisir une valeur de <math>\lambda</math> (<math>1 &lt; \lambda &lt; 0</math>)</p> <p><b>1.</b> Choisir une condition initiale <math>\mathbf{w}_0</math> (Aléatoirement sur l'attracteur)</p> <p><b>2.</b> Choisir une matrice <math>\{\mathbf{C}_k\}</math> (avec <math>k = 1, ..2^D D!</math>)</p> <p><b>3. WHILE</b></p> <p style="text-align: center;"><math> \delta\mathbf{w}^{SD}  &gt; \text{PRÉCISION}</math></p> <p style="text-align: center;">ou</p> <p style="text-align: center;"><math>N_{itération} &lt; N_{MAX}</math></p> <ul style="list-style-type: none"> <li>• Calculer <math>\delta\mathbf{w}^{SD}</math> (équation 1.30)</li> <li>• Calculer <math>\mathbf{w}_{n+1} = \mathbf{w}_n + \delta\mathbf{w}^{SD}</math></li> </ul> <p>end WHILE</p> <p><b>4. IF</b> (<math> \delta\mathbf{w}^{SD}  &lt; \text{PRÉCISION}</math>)  <math>\Rightarrow \mathbf{w}_{n+1} = \text{composante d'une OP}</math></p> <ul style="list-style-type: none"> <li>• GOTO <b>2</b> (avec <math>k = k+1</math> )</li> </ul> <p>ELSE GOTO <b>2</b> (avec <math>k = k+1</math> )</p> <p>Fin Choix de matrices <math>\mathbf{C}_k</math> GOTO <b>1</b></p>	<p><b>0.</b> Choisir une valeur de <math>\beta</math> (<math>\beta &gt; 0</math>)</p> <p><b>1.</b> Choisir une condition initiale <math>\mathbf{w}_0</math> (Composante d'une orbite de période <math>m - 1</math>)</p> <p><b>2.</b> Choisir une matrice <math>\{\mathbf{C}_k\}</math> (avec <math>k = 1, ..2^D D!</math>)</p> <p><b>3. WHILE</b></p> <p style="text-align: center;"><math> \delta\mathbf{w}^{DL}  &gt; \text{PRÉCISION}</math></p> <p style="text-align: center;">ou</p> <p style="text-align: center;"><math>N_{itération} &lt; N_{MAX}</math> (<math>\approx 4 - 6</math> fois <math>\beta</math>)</p> <ul style="list-style-type: none"> <li>• Calculer <math>\delta\mathbf{w}^{DL}</math> (équation 1.43)</li> <li>• Calculer <math>\mathbf{w}_{n+1} = \mathbf{w}_n + \delta\mathbf{w}^{DL}</math></li> </ul> <p>end WHILE</p> <p><b>4. IF</b> (<math> \delta\mathbf{w}^{DL}  &lt; \text{PRÉCISION}</math>)  <math>\Rightarrow \mathbf{w}_{n+1} = \text{composante d'une OP}</math></p> <ul style="list-style-type: none"> <li>• GOTO <b>2</b> (avec <math>k = k+1</math> )</li> </ul> <p>ELSE GOTO <b>2</b> (avec <math>k = k+1</math> )</p> <p>Fin Choix de matrices <math>\mathbf{C}_k</math> GOTO <b>1</b></p>

TAB. 1.4 – Résumé des algorithmes et *SD* et *DL*.

# Chapitre 2

## Présentation des modèles chaotiques

Ce chapitre est consacré à la présentation des modèles chaotiques utilisés dans ce travail. La plupart d'entre eux sont bien connus et documentés dans la littérature. Pour ces modèles, nous nous limitons à présenter les équations de la dynamique et citons les références appropriées. Si possible, nous donnons la forme analytique de la matrice jacobienne, matrice utile pour déterminer la stabilité des orbites périodiques et aussi nécessaire à l'implémentation de l'algorithme *DL* présenté au chapitre précédent. Une attention particulière est portée aux flots continus afin d'expliquer comment les ramener à des applications discrètes et obtenir numériquement leur matrice jacobienne.

## 2.1 Considérations générales

Nous avons vu, au chapitre précédent, deux méthodes qui permettent d'obtenir la position des orbites périodiques pour un système dynamique discret en dimension  $D$ . Soit  $\mathbf{F}$  une application différentiable de  $\Re^D$  dans  $\Re^D$  et considérons le système dynamique :

$$\mathbf{v}_{n+1} = \mathbf{F}(\mathbf{v}_n), \quad (2.1)$$

soit

$$\begin{pmatrix} v_{n+1}^1 \\ \vdots \\ v_{n+1}^D \end{pmatrix} = \begin{pmatrix} F^1(v_n^1, \dots, v_n^D) \\ \vdots \\ F^D(v_n^1, \dots, v_n^D) \end{pmatrix}. \quad (2.2)$$

Ainsi, le vecteur  $\mathbf{v}_n$  représente l'état des  $D$  variables à la  $n^{ième}$  itération. Pour les premiers systèmes que nous présentons,  $\mathbf{F}$  est connue analytiquement. Ces systèmes serviront, entre autre, à vérifier l'efficacité des deux algorithmes de détection d'OPIs, l'implémentation étant alors grandement simplifiée. Nous abordons ensuite la dynamique des *Billards*, où  $\mathbf{F}$  est construite de façon *semi-analytique* : On obtient les  $\mathbf{v}_{n+1}$  en calculant numériquement le zéro d'une équation obtenue analytiquement. Enfin, nous terminons avec les flots continus, soit les systèmes où la dynamique se détermine en intégrant numériquement les équations différentielles qui lui sont associées. L'application discrète  $\mathbf{F}$  pour ces flots est calculée à l'aide d'une *Section de Poincaré*.

La matrice jacobienne de  $\mathbf{F}$ , qui est nécessaire à l'implémentation de l'algorithme *DL* ainsi qu'au calcul de stabilité des orbites, est donnée par :

$$\mathbf{J} = D_{\mathbf{v}} \mathbf{F}(\mathbf{v}) \equiv \begin{pmatrix} \frac{dF^1}{dv^1} & \cdots & \frac{dF^1}{dv^D} \\ \vdots & \ddots & \vdots \\ \frac{dF^D}{dv^1} & \cdots & \frac{dF^D}{dv^D} \end{pmatrix} = \begin{pmatrix} \frac{dv_{n+1}^1}{dv_n^1} & \cdots & \frac{dv_{n+1}^1}{dv_n^D} \\ \vdots & \ddots & \vdots \\ \frac{dv_{n+1}^D}{dv_n^1} & \cdots & \frac{dv_{n+1}^D}{dv_n^D} \end{pmatrix}. \quad (2.3)$$

Afin d'alléger quelque peu les notations, nous n'indiquons pas la dépendance explicite de  $\mathbf{J}$  par rapport à  $\mathbf{v}$ . Il ne faut cependant pas oublier que cette matrice varie en fonction de  $\mathbf{v}$  dans l'espace des phases.

## 2.2 Applications analytiques

### 2.2.1 Application de Hénon

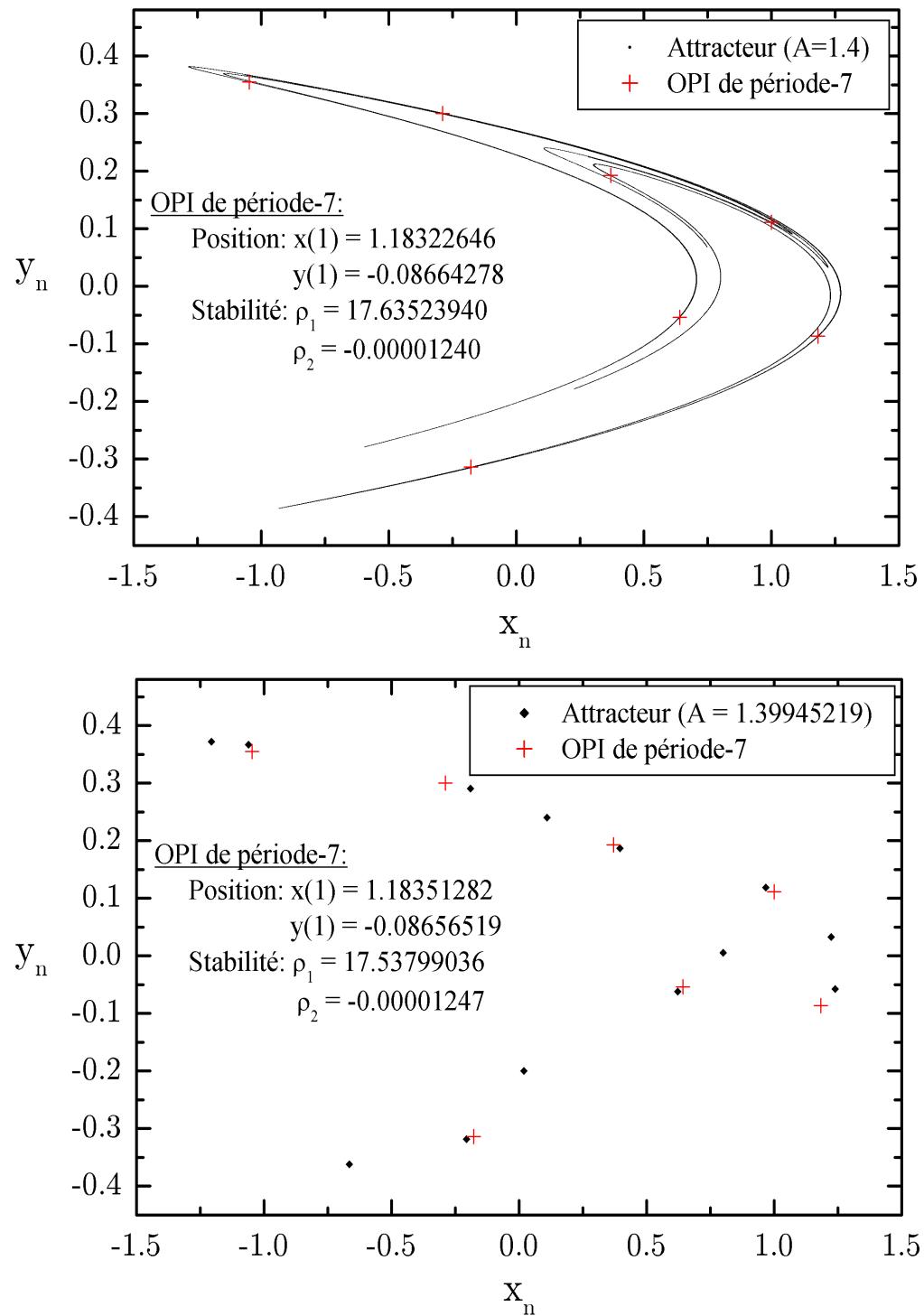
Une application classique dans la littérature de la dynamique non-linéaire est sans doute *l'Application de Hénon*. Présentée par Hénon vers la fin des années '70 [42], cette application représente un modèle simplifié des équations de Lorenz (réf. section 2.4.2) sur une Section de Poincaré. Sa forme générale est :

$$\text{Hénon : } \mathbf{v}_{n+1} = \begin{pmatrix} v_{n+1}^1 \\ v_{n+1}^2 \end{pmatrix} \equiv \begin{pmatrix} x_{n+1} \\ y_{n+1} \end{pmatrix} = \begin{pmatrix} 1 + y_n - Ax_n^2 \\ Bx_n \end{pmatrix}, \quad (2.4)$$

avec la matrice jacobienne donnée par :

$$\mathbf{J}_{\text{Hénon}} = \begin{pmatrix} -2Ax_n & 1 \\ B & 0 \end{pmatrix}. \quad (2.5)$$

Pour  $0 < |B| < 1$ , chaque itération appliquée sur un amas de conditions initiales choisies dans le plan  $XY$  “comprime” cet amas dans la direction de l’axe  $Y$ . Pour  $B = 0.3$  et  $A = 1.4$ , on obtient l’attracteur de Hénon, représenté sur la figure 2.1. La dynamique de l’application est aussi présentée pour  $A = 1.39945219$  et  $B = 0.3$ . Avec cette légère modification de  $A$ , l’attracteur est maintenant une orbite de période-13 stable. Bien qu’il y ait un changement drastique au niveau de l’attracteur, les orbites périodiques instables sont quant à elles très peu modifiées. L’encadré sur la figure indique la position de la première composante d’une OPI de période-7 et sa stabilité pour ces deux valeurs de paramètres. On remarque que la stabilité de l’OPI est très peu modifiée. Pour reprendre les mots de Cvitanović [15], “les OPIs sont *robustes* face au changement de paramètre”. C’est en partie ce qui les rend si “précieuses” pour une description globale de la dynamique. Une étude détaillée de l’application de Hénon est donnée par Mira [62]. Le système de Hénon sera utile pour vérifier le bon fonctionnement des deux algorithmes puisque les OPIs de cette application sont connues et documentées (période-11 à période-28) [8].

FIG. 2.1 – Orbites périodiques et attracteur du système de Hénon ( $B = 0.3$ ).

## 2.2.2 Application de Ikéda

L'étude du chaos dans les lasers en anneau remonte au début des années '80 [47]. Ikéda a développé à cette époque une application simple pour décrire la dynamique d'une telle cavité laser. Bien que limitée à un régime physique particulier, l'application d'Ikéda demeure aujourd'hui l'une des plus populaires et des plus citées pour l'introduction aux systèmes chaotiques en dimension 2. En général cependant, l'étude de la dynamique d'une cavité en anneau ne peut se réduire à une simple fonction de deux variables. Le modèle complet utilise plutôt une équation à délai pour le champ électrique couplée à une équation différentielle représentant l'évolution du milieu de gain. Il s'agit alors d'un système dynamique en dimension infinie à cause de la présence de l'équation à délai. Cependant, avec certaines approximations, ces équations se ramènent à l'application de Ikéda-Hammel-Jones-Moloney [47],[40], donnée explicitement par :

$$\text{Ikéda : } \mathbf{v}_{n+1} = \begin{pmatrix} v_{n+1}^1 \\ v_{n+1}^2 \end{pmatrix} \equiv \begin{pmatrix} x_{n+1} \\ y_{n+1} \end{pmatrix} = \begin{pmatrix} a + b(x_n \cos(\phi) - y_n \sin(\phi)) \\ b(x_n \sin(\phi) + y_n \cos(\phi)) \end{pmatrix}, \quad (2.6)$$

avec  $\phi = k - \eta/(1 + x_n^2 + y_n^2)$  et la matrice jacobienne a la forme suivante :

$$\mathbf{J}_{Ikeda} = b \begin{pmatrix} \frac{d\phi}{dx} \left\{ \left( \frac{dx}{d\phi} - y_n \right) \cos(\phi) - x_n \sin(\phi) \right\} & -\frac{d\phi}{dy} \left\{ \left( \frac{dy}{d\phi} + x_n \right) \sin(\phi) + y_n \cos(\phi) \right\} \\ \frac{d\phi}{dx} \left\{ \left( \frac{dx}{d\phi} - y_n \right) \sin(\phi) + x_n \cos(\phi) \right\} & \frac{d\phi}{dy} \left\{ \left( \frac{dy}{d\phi} + x_n \right) \cos(\phi) - y_n \sin(\phi) \right\} \end{pmatrix}, \quad (2.7)$$

où  $\frac{d\phi}{dz} \equiv 2\eta z_n/(1 + x_n^2 + y_n^2)^2$  avec  $z = x$  ou  $y$ . L'application est chaotique entre autre pour :  $a = 1.0$ ,  $b = 0.9$ ,  $k = 0.4$  et  $\eta = 6.0$ . La figure 2.2 montre l'attracteur pour ces valeurs des paramètres. Contrairement à Hénon, il n'existe pas de dynamique symbolique connue pour Ikéda. Cette particularité en a fait un véritable "banc d'essais" pour mesurer l'efficacité des méthodes de détection d'orbites périodiques.

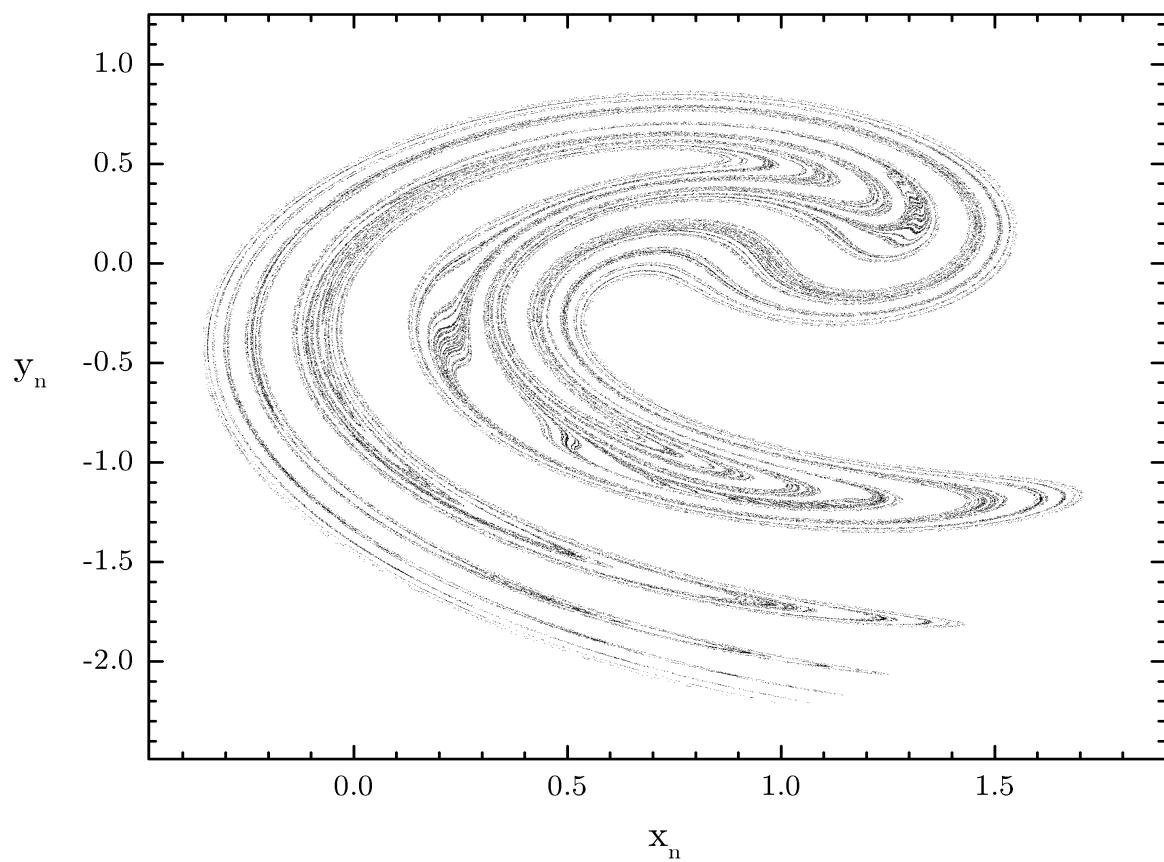


FIG. 2.2 – Attracteur de l’application Ikéda avec  $a = 1.0$ ,  $b = 0.9$ ,  $k = 0.4$  et  $\eta = 6.0$ .

### 2.2.3 Application Standard

Les systèmes conservatifs occupent une place importante parmi l'ensemble des systèmes dynamiques connus. En mécanique classique, ils se caractérisent par le fait que leur énergie ne dépend pas du temps (constante du mouvement). En général, le terme conservatif réfère cependant à la conservation d'un élément de volume dans l'espace des phases lors de l'évolution du système (“Area-preserving”). Le déterminant de la matrice jacobienne en chaque point de l'espace des phases est égal à 1.

Une application discrète fort utile pour l'étude des systèmes conservatifs est *l'Application Standard*. Elle peut servir à modéliser la dynamique d'une balle rebondissant sur une surface oscillante ou encore celle d'une chaîne d'atomes dans un réseau particulier [48]. La forme utilisée dans cette thèse est :

$$\text{App. Standard : } \mathbf{v}_{n+1} = \begin{pmatrix} v_{n+1}^1 \\ v_{n+1}^2 \end{pmatrix} \equiv \begin{pmatrix} q_{n+1} \\ p_{n+1} \end{pmatrix} = \begin{pmatrix} q_n + p_{n+1} \\ p_n - \frac{K}{2\pi} \sin(2\pi q_n) \end{pmatrix} \pmod{1}, \quad (2.8)$$

avec la matrice jacobienne donnée par :

$$\mathbf{J}_{\text{App-Standard}} = \begin{pmatrix} 1 - K \cos(2\pi q_n) & 1 \\ -K \cos(2\pi q_n) & 1 \end{pmatrix}. \quad (2.9)$$

On vérifie facilement que le déterminant de cette matrice est bien égal à 1. Pour  $K = 0$ , le système évolue sur des orbites périodiques dans l'espace des phases ou sur un ensemble de points limités à une courbe fermée. On parle ici d'un régime régulier (intégrable au sens de la mécanique classique). Le système explore une partie limitée de l'espace des phases. L'augmentation de la valeur de  $K$  entraîne la destruction de cette régularité. Du point de vue théorique, cette transition a d'abord été formulée par Kolmogorov, Arnold et Moser (KAM). Ces derniers ont réalisé que dans un régime intégrable, le système dynamique évolue dans l'espace des phases sur une surface bornée : un tore (de KAM). Dans la transition vers le chaos, ces tores se détruisent graduellement et font place à des régions irrégulières (ergodiques) dans l'espace des phases. La destruction

du dernier tore de KAM marque le début d'un régime complètement chaotique.

Les systèmes conservatifs posent un défi intéressant aux méthodes de détection d'OPIs. Nous avons vu au chapitre 1 que certaines d'entre elles reposent sur le fait que l'espace des phases est ergodique (c'est le cas des méthodes par *récurrence*). Dans le régime où les tores de KAM coexistent avec les régions chaotiques, on obtient une dynamique mixte qui globalement n'est pas ergodique, d'où l'intérêt d'inclure ces systèmes dans notre étude. Un survol de la théorie des systèmes dynamiques hamiltoniens est fait par MacKay et Meiss [56].

## 2.2.4 Application Hénon-3D

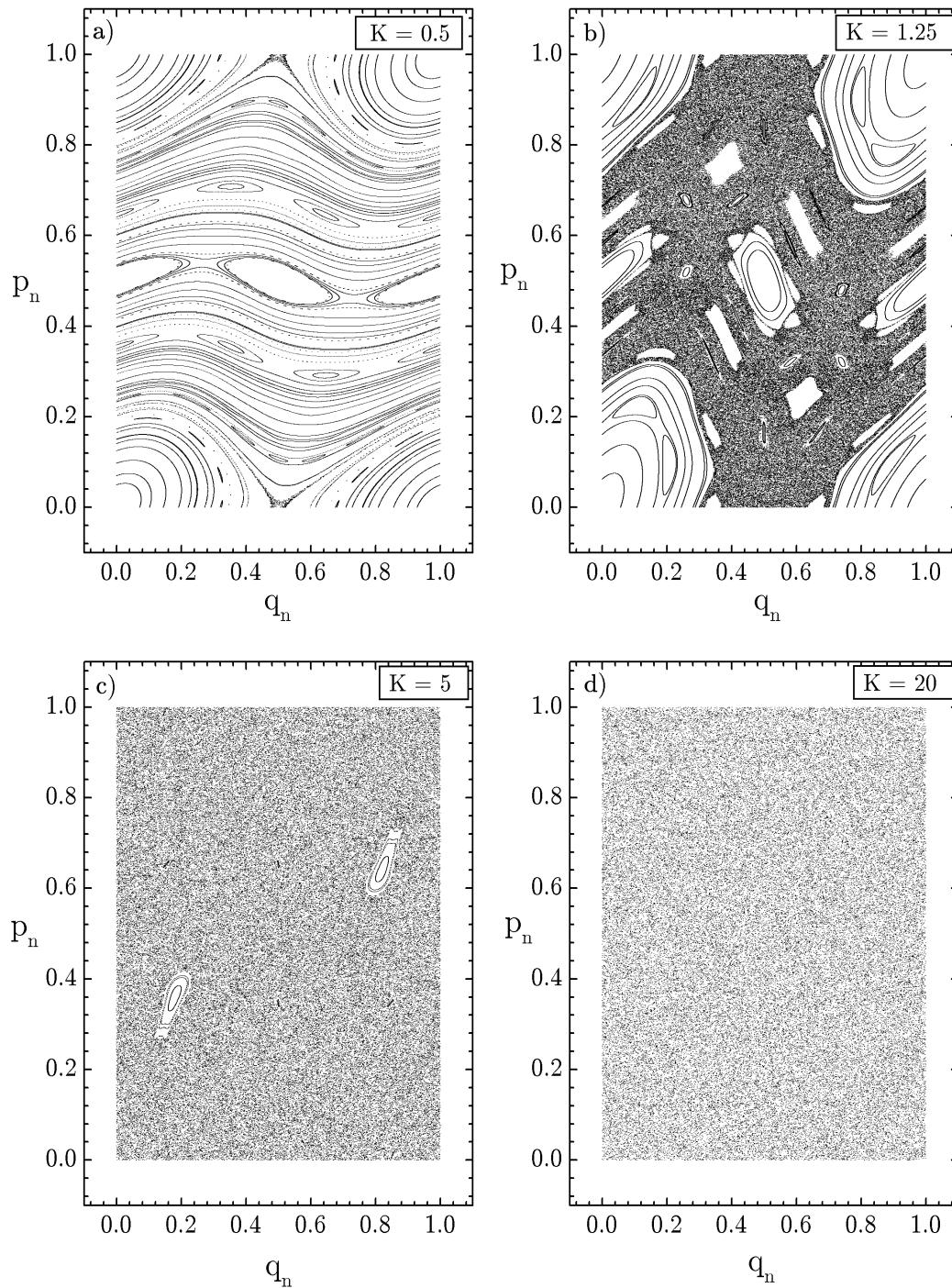
La plupart des méthodes de détection d'OPIs présentées dans la littérature se limitent à des applications en dimension 2. Nous incluons dans ce travail des applications en dimensions 3 et 4 (espace des phases de dimensions 3 et 4). L'application de Hénon-3D est donnée par :

$$\text{Hénon-3D} \quad \mathbf{v}_{n+1} = \begin{pmatrix} v_{n+1}^1 \\ v_{n+1}^2 \\ v_{n+1}^3 \end{pmatrix} \equiv \begin{pmatrix} x_{n+1} \\ y_{n+1} \\ z_{n+1} \end{pmatrix} = \begin{pmatrix} A - y_n^2 - Bz_n \\ x_n \\ y_n \end{pmatrix}, \quad (2.10)$$

et sa matrice jacobienne est :

$$\mathbf{J}_{\text{Henon-3D}} = \begin{pmatrix} 0 & -2y_n & -B \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}. \quad (2.11)$$

Les paramètres utilisés sont ici  $A = 1.76$  et  $B = 0.1$ , paramètres pour lesquels ce système est chaotique. La figure 2.4 montre l'attracteur dans un espace des phases de dimension 3. On montre aussi sur cette même figure, deux coupes de l'attracteur. Ces deux "tranches" ont été obtenues en collectant les points pour lesquels  $z = 1.8 \pm 10^{-4}$  (figure 2.4 b) et  $z = 0 \pm 10^{-4}$  (figure 2.4c).

FIG. 2.3 – Application standard pour quelques valeurs de  $K$ .

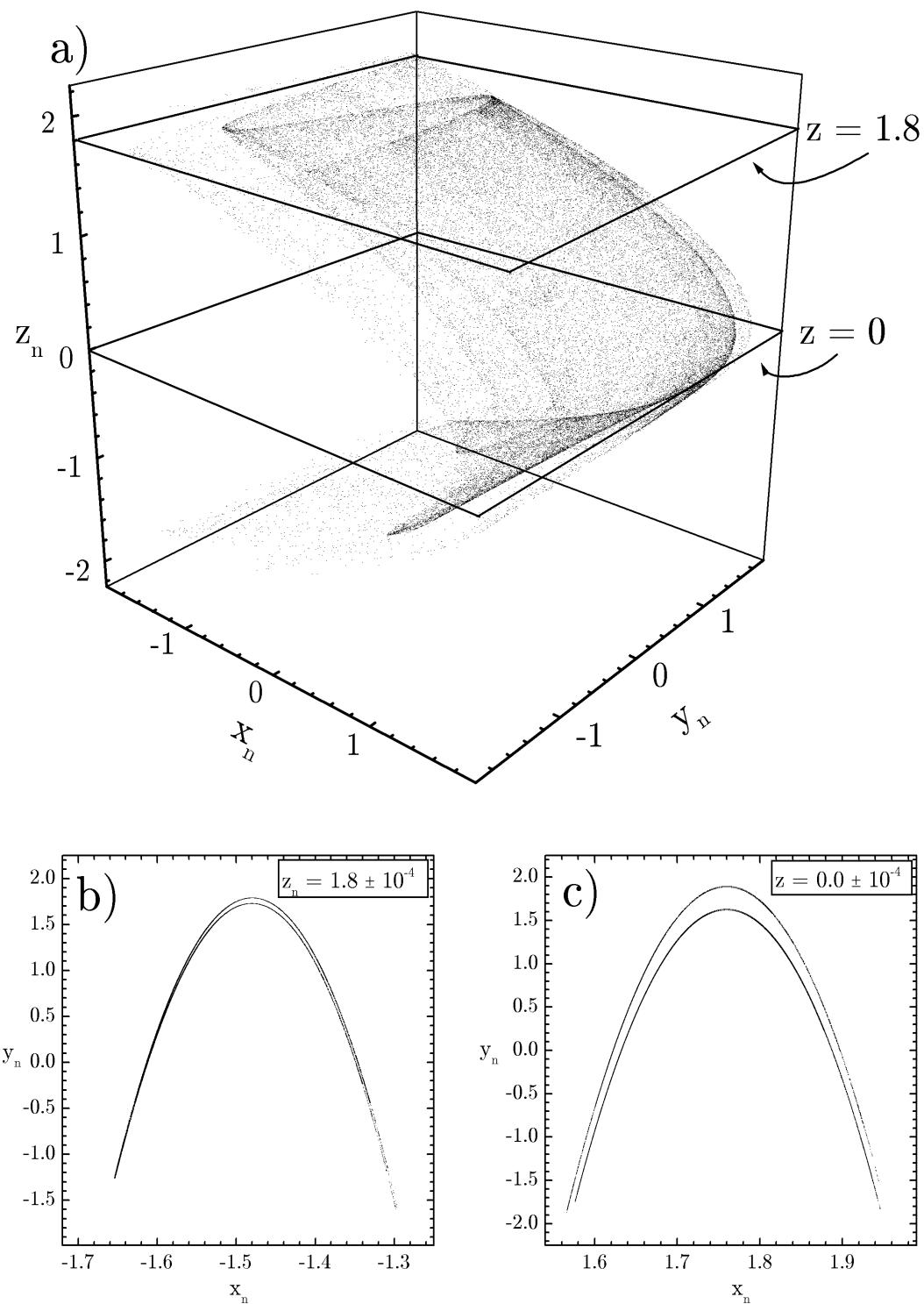


FIG. 2.4 – Attracteur du système de Hénon-3D.

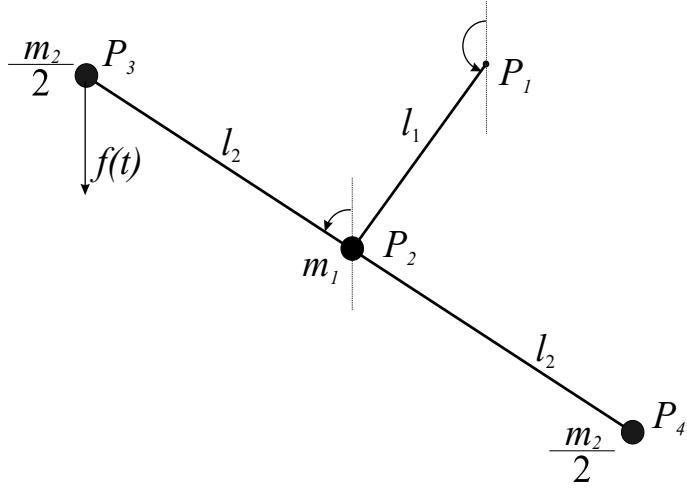


FIG. 2.5 – Le Pendule Double

### 2.2.5 Le Pendule Double

Le *Pendule Double* est une application en dimension 4 qui décrit l'évolution d'un système mécanique formé de deux pendules liés et entretenus par de courtes impulsions [36] (figure 2.5). Ce système pose un bon défi tant pour la détection des orbites périodiques (espace des phases de dimension 4) que pour son contrôle (dimension de Lyapunov = 2.8).

Ce pendule est composé de deux tiges. L'une d'elle, de longueur  $l_1$  et de masse négligeable, est suspendue par une de ses extrémités au point  $P_1$  (figure 2.5). À l'autre extrémité, on accroche une masse  $m_1$  ainsi que la deuxième tige (de longueur  $2l_2$  et de masse négligeable ; point  $P_2$ ). Sur cette deuxième tige, on fixe deux masses  $m_2/2$  (points  $P_3$  et  $P_4$ ). Les impulsions “ $f(t)$ ” de période  $T$  et de grandeur  $f_0$  sont appliquées verticalement au point  $P_3$ . Les positions angulaires des tiges 1 et 2 sont données respectivement par  $\theta_1$  et  $\theta_2$ . Tout le système est soumis à un frottement visqueux proportionnel à la vitesse angulaire  $\frac{d\theta_{1,2}}{dt}$  de chacune des 2 tiges. Le coefficient de proportionnalité entre la force de frottement et la vitesse angulaire pour la tige “ $k$ ” est  $\nu_k$ .

La dérivation de cette application particulière est présentée en détails par Romeiras

& al.[84]. En utilisant le symbole  $\theta_n^k$  pour représenter la position de la tige  $k$  au moment de la  $n^{ième}$  impulsion et la variable  $\omega_n^k$  pour décrire la vitesse angulaire de cette tige *juste après* l'impulsion, l'application prend la forme suivante :

$$\begin{array}{c} \text{Pendule} \\ \text{Double} \end{array} : \begin{pmatrix} v_{n+1}^1 \\ v_{n+1}^2 \\ v_{n+1}^3 \\ v_{n+1}^4 \end{pmatrix} \equiv \begin{pmatrix} \theta_{n+1}^1 \\ \theta_{n+1}^2 \\ \omega_{n+1}^1 \\ \omega_{n+1}^2 \end{pmatrix} = \begin{pmatrix} M_{11}\omega_n^1 + M_{12}\omega_n^2 + \theta_n^1 \\ M_{21}\omega_n^1 + M_{22}\omega_n^2 + \theta_n^2 \\ L_{11}\omega_n^1 + L_{12}\omega_n^2 + c_1 \sin(\theta_{n+1}^1) \\ L_{21}\omega_n^1 + L_{22}\omega_n^2 + c_2 \sin(\theta_{n+1}^2) \end{pmatrix}, \quad (2.12)$$

avec :

$$\begin{aligned} L_{ij} &\equiv \sum_{k=1}^2 W_{ij}^{(k)} e^{\lambda_k T} \\ M_{ij} &\equiv \sum_{k=1}^2 W_{ij}^{(k)} \left\{ \frac{e^{\lambda_k T} - 1}{\lambda_k} \right\} \\ W_{11}^{(1)} &= W_{22}^{(2)} = \frac{1}{2} \left( 1 + \frac{\nu_1}{\Delta} \right) & W_{12}^{(1)} &= -W_{12}^{(2)} = -\frac{\nu_2}{\Delta} \\ W_{21}^{(1)} &= -W_{21}^{(2)} = -\frac{\nu_2}{\Delta} & W_{22}^{(1)} &= W_{11}^{(2)} = \frac{1}{2} \left( 1 - \frac{\nu_1}{\Delta} \right) \\ \lambda_1 &= -\frac{1}{2}(\nu_1 + 2\nu_2 + \Delta) & \lambda_2 &= -\frac{1}{2}(\nu_1 + 2\nu_2 - \Delta) & \Delta &= (\nu_1^2 + \nu_2^2)^{1/2}. \end{aligned}$$

Les paramètres  $c_{1,2}$  sont définis par :

$$c_k = \frac{f_0}{I} l_k \quad \text{avec} \quad I = (m_1 + m_2)l_1^2 = m_2 l_2^2. \quad (2.13)$$

Dans cette thèse, nous étudions le cas  $\nu_1 = \nu_2 \equiv \nu$ . Les valeurs pour les différents paramètres sont :

$$\begin{aligned} \nu &= T = I = m_2 = m_2 = l_2 = 1 \\ l_1 &= 1/\sqrt{2}. \end{aligned} \quad (2.14)$$

Aussi, nous prenons pour l'ensemble de ce travail  $f_0 = 9.0$ , qui est aussi la valeur choisie par Romeiras [84] (rappelons que  $f_0$  correspond à la force des impulsions appliquées).

La matrice jacobienne est donnée par :

$$\mathbf{J}_{Pendule} = \begin{pmatrix} 1 & 0 & M_{11} & M_{12} \\ 0 & 1 & M_{21} & M_{22} \\ c_1 \cos(\theta_{n+1}^1) & 0 & L_{11} + c_1 \cos(\theta_{n+1}^1)M_{11} & L_{12} + c_1 \cos(\theta_{n+1}^1)M_{12} \\ 0 & c_2 \cos(\theta_{n+1}^2) & L_{21} + c_2 \cos(\theta_{n+1}^2)M_{21} & L_{22} + c_2 \cos(\theta_{n+1}^2)M_{22} \end{pmatrix}. \quad (2.15)$$

L'attracteur dans l'espace des phases en dimension 4 est ici difficilement représentable. On peut cependant définir une projection de cet attracteur sur une surface de dimension 2. Cette projection est obtenue en collectant les points de l'attracteur regroupés dans une bande définie par :

$$|\mathbf{K} \cdot (\mathbf{v} - \mathbf{v}_c)| < w, \quad (2.16)$$

avec  $w \sim 10^{-2}$  et  $\mathbf{K}$  un vecteur de  $\Re^4$ . Le point  $\mathbf{v}_c$  est un point autour duquel on souhaite projeter l'attracteur. Le vecteur  $\mathbf{K}$  utilisé est étroitement lié au point  $\mathbf{v}_c$ . Cette projection, qui est identique à celle proposée par Romeiras & al. [84], ne garantit pas que les points collectés sont réellement regroupés dans l'espace de dimension 4. En fait, on trouve une certaine justification à cette définition lorsqu'on aborde l'aspect du contrôle des OPIs. La figure 2.6 montre une “tranche” de l'attracteur obtenue en collectant les points qui respectent le critère 2.16 avec  $\mathbf{K} = (0, 0, 1, 1)$ ,  $w = 10^{-2}$ . Pour la figure 2.6a, on choisit  $\mathbf{v}_c = (0, 0, 0, 0)$ . Ici, les deux premières composantes de  $v_c$  n'ont pas vraiment d'importance car  $K_1 = 0$  et  $K_2 = 0$ . Les autres composantes ( $v_c^3$  et  $v_c^4$ ) correspondent aux coordonnées en  $\omega^k$  d'un des points fixes. Pour la figure 2.6b, on choisit  $\mathbf{v}_c = (0, 0, 10.374, 5.820)$ , où  $v_c^3$  et  $v_c^4$  sont les positions en  $\omega^k$  d'un autre point fixe.

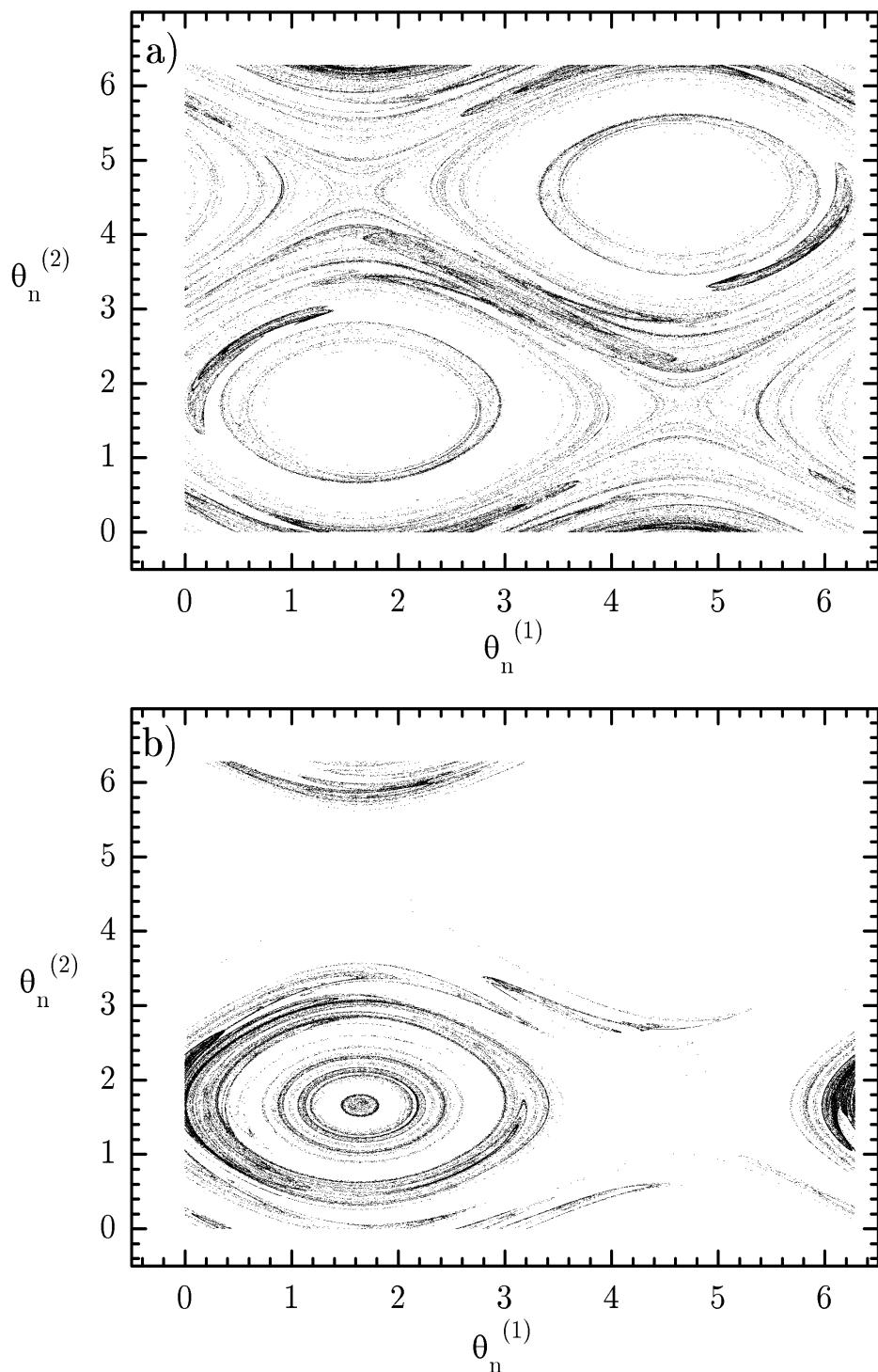


FIG. 2.6 – Projection de l'attracteur du Pendule Double.

La projection est définie par  $|\mathbf{K} \cdot (\mathbf{v} - \mathbf{v}_c)| < w$  avec  $w = 0.01$ ,  $\mathbf{K} = (0, 0, 1, 1)$ . Figure a) :  $\mathbf{v}_c = (0, 0, 0, 0)$ ; Figure b) :  $\mathbf{v}_c = (0, 0, 10.374, 5.820)$

## 2.3 Applications semi-analytiques : les Billards

L'hypothèse ergodique de Boltzman [45] a joué un rôle important dans le développement des théories en chaos. Cette hypothèse a un impact considérable en physique statistique puisqu'elle permet d'évaluer des quantités thermodynamiques fondamentales reliées à des systèmes physiques, la chaleur spécifique n'étant qu'un exemple. La validité de l'hypothèse ergodique a été démontrée seulement pour quelques systèmes particuliers.

Les théorèmes de von Neumann [105] et Birkhoff [9] sur l'hypothèse ergodique furent publiés au début des années 1900 et ont conduit à des notions essentielles en dynamique du chaos, dont l'entropie de Kolmogorov. Dans la recherche d'une preuve à l'ergodicité, on en était venu à rechercher des systèmes mathématiques simples où l'hypothèse ergodique serait vérifiable. Hadamard [39] avait considéré un système composé d'une particule rebondissant à l'intérieur d'une cavité (un jeu de billard) et ce système fût repris plus tard par Birkhoff.

En plus de sa grande simplicité et de son aspect pédagogique remarquable, la dynamique d'un billard est très riche en phénomènes non-linéaires. On peut observer, dans un tel système, un comportement régulier (intégrable) ou complètement irrégulier (ergodique) en passant par différentes combinaisons de ces deux comportements [6], [50]. Du point de vue numérique, ce système comporte un avantage majeur : aucune intégration n'est nécessaire pour obtenir l'évolution du système.

En général, la particule dans le billard évolue en ligne droite entre ses différents rebonds ou peut être sous l'influence d'un champ quelconque (gravitationnel, magnétique ou électrique). Nous présentons ici trois modèles de billards : le billard libre, magnétique et gravitationnel, modèles qui ont tous, comme nous le verrons, de grandes similitudes avec des systèmes physiques. Au chapitre 4, nous développons une nouvelle approche pour le ciblage et le contrôle d'orbites périodiques ; ces modèles de billards seront alors

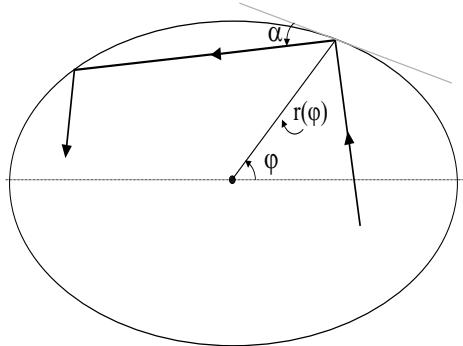


FIG. 2.7 – Dynamique du Billard Libre.

utilisés pour valider notre méthode.

### 2.3.1 Billards Libres

Le système du *Billard Libre* correspond à une particule qui glisse sans friction dans une région plane limitée par une courbe fermée quelconque. La particule rebondit indéfiniment dans la cavité ainsi formée. Aucune énergie n'est perdue lors des rebonds (collisions parfaitement élastiques), qui sont traités selon un principe d'optique géométrique : l'angle d'incidence, mesuré par rapport à la normale de la cavité au point de contact, doit être égal à l'angle réfléchi.

La dynamique du billard est décrite complètement par deux variables soit  $\varphi_n$ , la position angulaire de la particule lors du  $n^{ième}$  rebond, et  $\alpha_n$ , l'angle de réflexion mesuré par rapport à la tangente de la cavité au  $n^{ième}$  rebond. Nous utilisons, pour générer la suite des  $(\varphi_n, \alpha_n)$ , une technique numérique simple décrite dans [50] et présentée à l'annexe B :

$$\text{Billard Libre : } \mathbf{v}_{n+1} = \begin{pmatrix} v_{n+1}^1 \\ v_{n+1}^2 \end{pmatrix} \equiv \begin{pmatrix} \varphi_{n+1} \\ \alpha_{n+1} \end{pmatrix} = \mathbf{A}_{B-L}(\varphi_n, \alpha_n). \quad (2.17)$$

On représente ici l'application semi-analytique pour le billard libre par  $\mathbf{A}_{B-L}$ , application qui s'appuie sur des principes de géométrie. Pour une cavité donnée (dont la forme

est connue de façon analytique), on obtient  $\varphi_{n+1}$  en calculant le zéro d'une fonction particulière à l'aide d'un algorithme de Newton-Raphson et en respectant la contrainte : la solution doit être entre 0 et  $2\pi$  et ne peut être  $\varphi_n$ . Supposons que le contour de la cavité est décrit par une fonction  $r(\varphi)$  :

$$\text{Contour de la cavité} = r(\varphi). \quad (2.18)$$

La matrice jacobienne ne peut être dérivée directement de l'application puisque celle-ci n'est pas donnée analytiquement. En utilisant un système de coordonnées différent, Berry [6] a obtenu une expression pour cette matrice et ce, par des considérations géométriques. On définit ainsi les variables  $s$  et  $p$  par :

$$\begin{aligned} s_n &= \int_0^{\varphi_n} d\varphi \sqrt{r^2 + (\frac{dr}{d\varphi})^2} \\ p_n &= \cos \alpha_n. \end{aligned} \quad (2.19)$$

La variable  $s_n$  donne la longueur d'arc de la cavité correspondant à l'angle  $\varphi_n$ . La vitesse tangentielle de la particule est égale à  $p_n$  (la vitesse étant normalisée à 1).

Dans ce système de coordonnées, on peut montrer ([6] et [23]) que la matrice jacobienne prend la forme suivante :

$$\mathbf{J}^{s-p} = \begin{pmatrix} \left\{ -\frac{q_n}{q_{n+1}} + \frac{l_n}{q_{n+1}\rho_n} \right\} & \left\{ -\frac{l_n}{q_n q_{n+1}} \right\} \\ \left\{ -\frac{l_n}{\rho_n \rho_{n+1}} + \frac{q_{n+1}}{\rho_n} + \frac{q_n}{\rho_{n+1}} \right\} & \left\{ -\frac{q_{n+1}}{q_n} + \frac{l_n}{q_n \rho_{n+1}} \right\} \end{pmatrix}, \quad (2.20)$$

avec  $q_n = \sin \alpha_n$ . La quantité  $l_n$  est la longueur du segment entre les points d'impacts  $n$  et  $n+1$  :

$$l_n = \sqrt{r_n^2 + r_{n+1}^2 - 2r_n r_{n+1} \cos(\varphi_{n+1} - \varphi_n)}, \quad (2.21)$$

et  $\rho_n$ , le rayon de courbure à  $\varphi_n$  :

$$\rho_n = \frac{(r_n^2 + r'_n)^{3/2}}{r_n^2 + 2r'^2_n - r_n r''_n}, \quad (2.22)$$

avec  $r_n = r(\varphi_n)$ ,  $r'_n = dr/d\varphi \Big|_{\varphi_n}$  et  $r''_n = d^2r/d\varphi^2 \Big|_{\varphi_n}$ .

Dans l'espace des variables  $(s, p)$ , l'application  $A_{B-L}$  préserve les aires car  $\det[\mathbf{J}^{s-p}] = 1$ . Comme on connaît la transformation  $(s_n, p_n) \mapsto (\varphi_n, \alpha_n)$ , il est possible d'écrire

la matrice jacobienne dans l'espace  $(\varphi_n, \alpha_n)$ . En utilisant l'équation 2.19, on déduit d'abord que :

$$\begin{aligned} ds_n &= \sqrt{r_n^2 + r'^2_n} \ d\varphi_n \\ dp_n &= -\sin \alpha_n \ d\alpha_n. \end{aligned}$$

Cette dernière relation et la définition de la matrice jacobienne (équation 2.3, expression de droite) conduisent directement à la matrice jacobienne dans l'espace  $(\varphi, \alpha)$  :

$$\mathbf{J}_{\text{Billard-Libre}}^{\varphi-\alpha} = \begin{pmatrix} J_{11}^{s-p} \left\{ \frac{r_n^2 + r'^2_n}{r_{n+1}^2 + r'^2_{n+1}} \right\}^{1/2} & J_{12}^{s-p} \left\{ \frac{-\sin \alpha_n}{(r_{n+1}^2 + r'^2_{n+1})^{1/2}} \right\} \\ J_{21}^{s-p} \left\{ \frac{-(r_n^2 + r'^2_n)^{1/2}}{\sin \alpha_{n+1}} \right\} & J_{22}^{s-p} \left\{ \frac{\sin \alpha_n}{\sin \alpha_{n+1}} \right\} \end{pmatrix}. \quad (2.23)$$

La dynamique de ce système est influencée par la forme de la cavité. Pour une cavité circulaire, le comportement de la particule est régulier (intégrable au sens de la mécanique classique). La particule explore une partie limitée de l'espace des phases. La déformation de la cavité entraîne la destruction de cette régularité. Tout comme l'application standard, le système dynamique évolue dans l'espace des phases sur une surface bornée : un tore lorsque la déformation est légère. En augmentant la déformation, on obtient une transition vers le chaos et les tores se détruisent un-à-un permettant au système d'explorer l'ensemble de l'espace des phases. Cette transition est bien documentée dans la littérature [6], [81], [50], [56]. À la figure 2.10, on présente quelques diagrammes des phases pour un billard ovale sous différentes déformations. La cavité ovale est définie par :

$$r_{\text{ovale}}(\varphi) = (1 + \epsilon/2)^{-1/2} \{1 + \epsilon \cos(2\varphi)\}, \quad (2.24)$$

où le paramètre  $\epsilon$  contrôle la déformation de l'ovale.

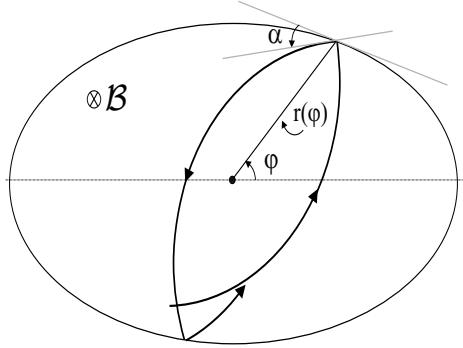


FIG. 2.8 – Dynamique du Billard Magnétique.

### 2.3.2 Billards Magnétiques

L’analyse classique du comportement des *Billards Magnétiques* a été entreprise par Robnick et Berry au milieu des années ’80 [82]. Le sujet est toujours d’actualité [44]. Ce système peut servir de modèle pour expliquer le comportement d’une particule chargée limitée à une région de l’espace en mésoscopie [41], [64], [104]. Plus spécifiquement, l’application d’un champ magnétique au billard s’avère utile pour l’analyse des états semi-classiques de systèmes liés (atomes) en présence d’un champ magnétique [85],[10],[97]. Dans ce contexte, le contrôle d’une particule chargée dans un Billard Magnétique par de faibles variations du champ nécessite une investigation. À notre connaissance, un tel contrôle n’a pas encore été démontré théoriquement. Le Billard Magnétique est utilisé comme “banc d’essais” pour la méthode de ciblage et de contrôle que nous présentons au chapitre 4.

Tout comme le Billard Libre, la dynamique du Billard Magnétique en 2D ne peut être obtenue de façon analytique en général. Une méthode numérique sert à générer la suite des rebonds  $(\varphi_n, \alpha_n)$ , ces deux variables ayant la même signification que précédemment (réf. figure 2.8). La trajectoire de la particule chargée correspond à un arc de cercle de rayon égal au rayon de Larmor ( $\equiv b$ ) :

$$\text{Rayon de Larmor} : b \equiv \frac{mv}{qB}, \quad (2.25)$$

où  $m$  est la masse de la particule,  $v$  sa vitesse,  $q$  sa charge et  $\mathcal{B}$  le champ magnétique externe. Ce champ est uniforme et perpendiculaire à la surface du billard. Par convention, le sens positif du champ magnétique “entre” dans la surface. Pour les simulations, le rayon de Larmor est normalisé par rapport au rayon de la cavité.

L’application semi-analytique pour le billard magnétique, décrite à l’annexe B, est notée par  $\mathbf{A}_{B-M}$  et on a :

$$\text{Billard Magnétique : } \mathbf{v}_{n+1} = \begin{pmatrix} v_{n+1}^1 \\ v_{n+1}^2 \end{pmatrix} \equiv \begin{pmatrix} \varphi_{n+1} \\ \alpha_{n+1} \end{pmatrix} = \mathbf{A}_{B-M}(\varphi_n, \alpha_n). \quad (2.26)$$

La figure 2.11 montre l’évolution du diagramme des phases pour quelques valeurs de  $b$  (*rayon de Larmor*) pour un billard ovale ( $\epsilon = 0.15$ ). On remarque que pour de faibles valeurs de  $b$  (ce qui correspond à un champ magnétique  $\mathcal{B}$  plus intense), certaines zones chaotiques laissent place à des régions régulières [82].

### 2.3.3 Billards Gravitationnels

Nous présentons ici une configuration de billards particulière. Il s’agit d’une particule soumise à l’influence d’un champ gravitationnel et rebondissant sur deux plans en forme de coin symétrique par rapport à ce champ (réf. figure 2.9). Ce système fut introduit par Lehtihet & Miller vers le milieu des années ’80 [54]. Tout comme les billards précédents, ce système offre une dynamique riche, qui est influencée notamment par l’angle formé par le coin. Contrairement aux deux billards en cavité, l’application peut ici s’obtenir analytiquement (Nous l’incluons toutefois dans cette section des applications semi-analytiques). On réfère à [54] ou [50] pour ce développement analytique.

On construit l’application à partir de deux variables  $(x_n, y_n)$ , qui correspondent respectivement aux vitesses radiale (selon  $\mathbf{e}_r$ ) et azimutale (selon  $\mathbf{e}_\theta$ ) de la particule *juste après* l’impact sur une des deux parois du billard (réf. figure 2.9). On distingue alors deux cas bien différents notés  $T_A$  et  $T_B$ . Le premier correspond à 2 rebonds

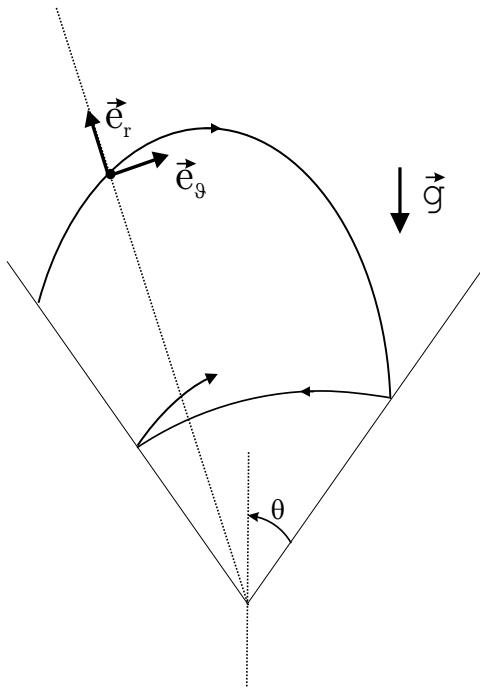


FIG. 2.9 – Dynamique du Billard Gravitationnel.

consécutifs sur la même paroi du coin “gravitationnel”. Le second décrit un rebond sur la paroi opposée. En notant  $\theta$  l’angle que fait une paroi du coin avec la verticale, on définit dans un premier temps les variables  $\alpha$  et  $\xi$  :

$$\begin{aligned}\alpha &\equiv \tan(\theta) \\ \xi &\equiv \frac{1 - \alpha^2}{(1 + \alpha^2)^2}.\end{aligned}\tag{2.27}$$

La dynamique s’obtient en calculant la suite des  $(x_n, y_n)$  par :

$$\text{Bill. gravit. : } \begin{pmatrix} v_{n+1}^1 \\ v_{n+1}^2 \end{pmatrix} \equiv \begin{pmatrix} x_{n+1} \\ y_{n+1} \end{pmatrix} = \begin{pmatrix} x_n - 2y_n \cot \theta \\ y_n \end{pmatrix} \quad (T_A) \\ \begin{pmatrix} x_n - 2y_n \cot \theta \\ y_n \end{pmatrix} = \begin{pmatrix} -x_n + (y_n - y_{n+1}) \cot \theta \\ \sqrt{2 \sin^2 \theta + 2\xi(y_n - x_n \tan \theta)^2 - y_n^2} \end{pmatrix} \quad (T_B). \tag{2.28}$$

On utilise le critère :

$$Si \quad [(x - 2y \cot \theta)^2 + y^2 \leq 1] \quad alors \quad T_A, \tag{2.29}$$

afin de déterminer les équations à utiliser. La matrice jacobienne est donnée par :

$$\begin{aligned} J_A &= \begin{pmatrix} 1 & -2 \cot \theta \\ 0 & 1 \end{pmatrix} \\ J_B &= \frac{1}{y_{n+1}} \begin{pmatrix} \{2\xi(y_n - x_n \tan \theta) - y_{n+1}\} & \{[y_{n+1} + (1 - 2\xi)y_n] \cot \theta + 2\xi x_n\} \\ \{2\xi \tan \theta(x_n \tan \theta - y_n)\} & \{(2\xi - 1)y_n - 2\xi x_n \tan \theta\} \end{pmatrix}. \end{aligned} \quad (2.30)$$

Une étude numérique détaillée du Billard Gravitationnel est présentée dans [54], [80] et [98]. On y montre entre autre que pour  $\theta < 45^0$ , l'espace des phases est mixte. La proportion des régions chaotiques fluctue avec la variation de l'angle  $\theta$  et passe par des minima pour  $\theta_n = 90^0/n$  avec  $n = 3, 4, 5, \dots$ . Ces minima ont été observés expérimentalement sur un Billard Gravitationnel constitué d'atomes de Cesium refroidis et confinés dans un plan par des faisceaux laser [61]. La figure 2.12 montre le diagramme des phases pour différentes valeurs de  $\theta$ . Les figures 2.12a et 2.12c correspondent à une situation où  $\theta = 90^0/n$  (soit  $n = 4$  et  $n = 5$  respectivement). On remarque alors une diminution des régions chaotiques.

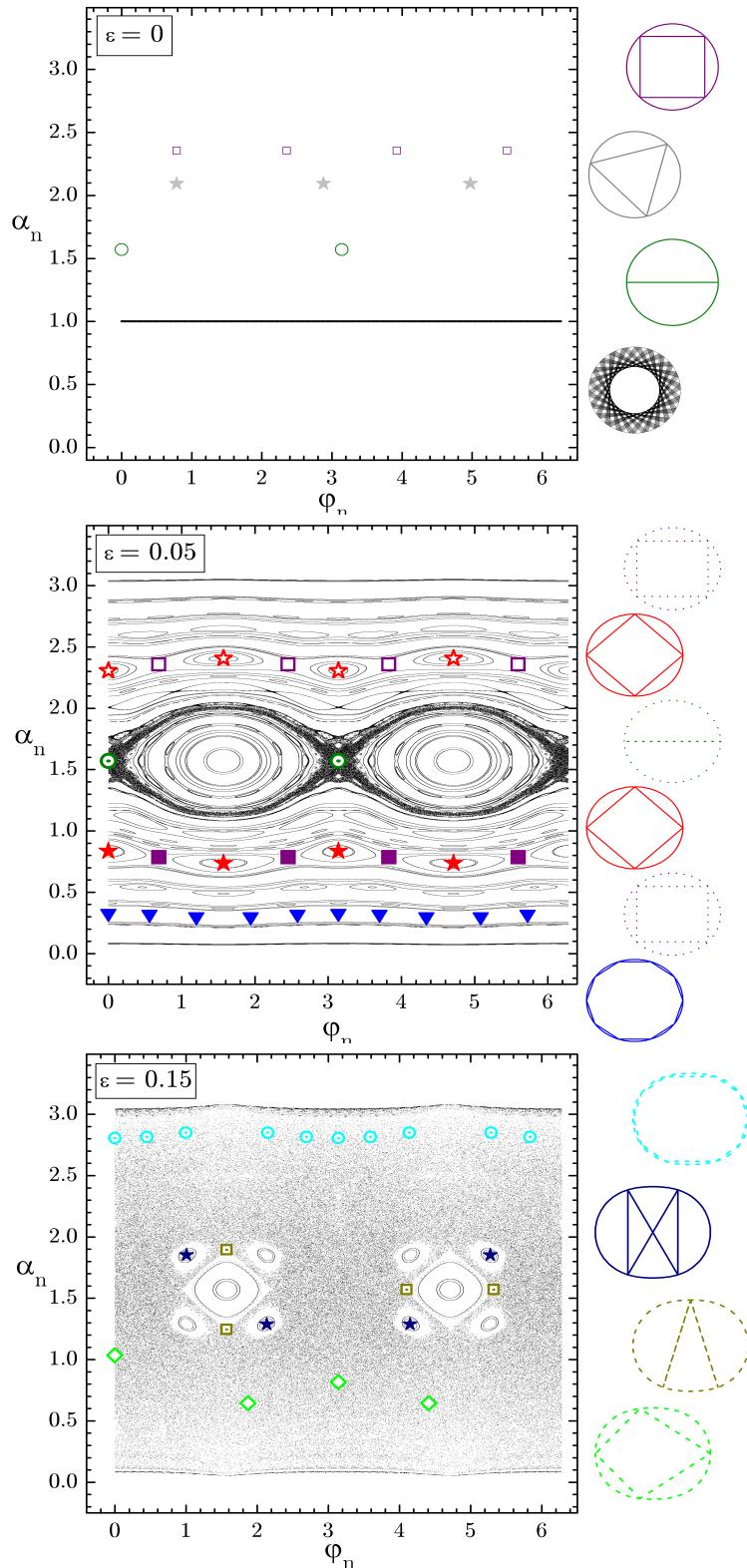


FIG. 2.10 – Diagramme des phases d'un Billard Ovale Libre.

Cette figure montre le diagramme des phases du Billard Ovale Libre pour différentes déformations  $\epsilon$ . Certaines trajectoires périodiques sont aussi illustrées. Les trajectoires pointillées sont des OPIS.

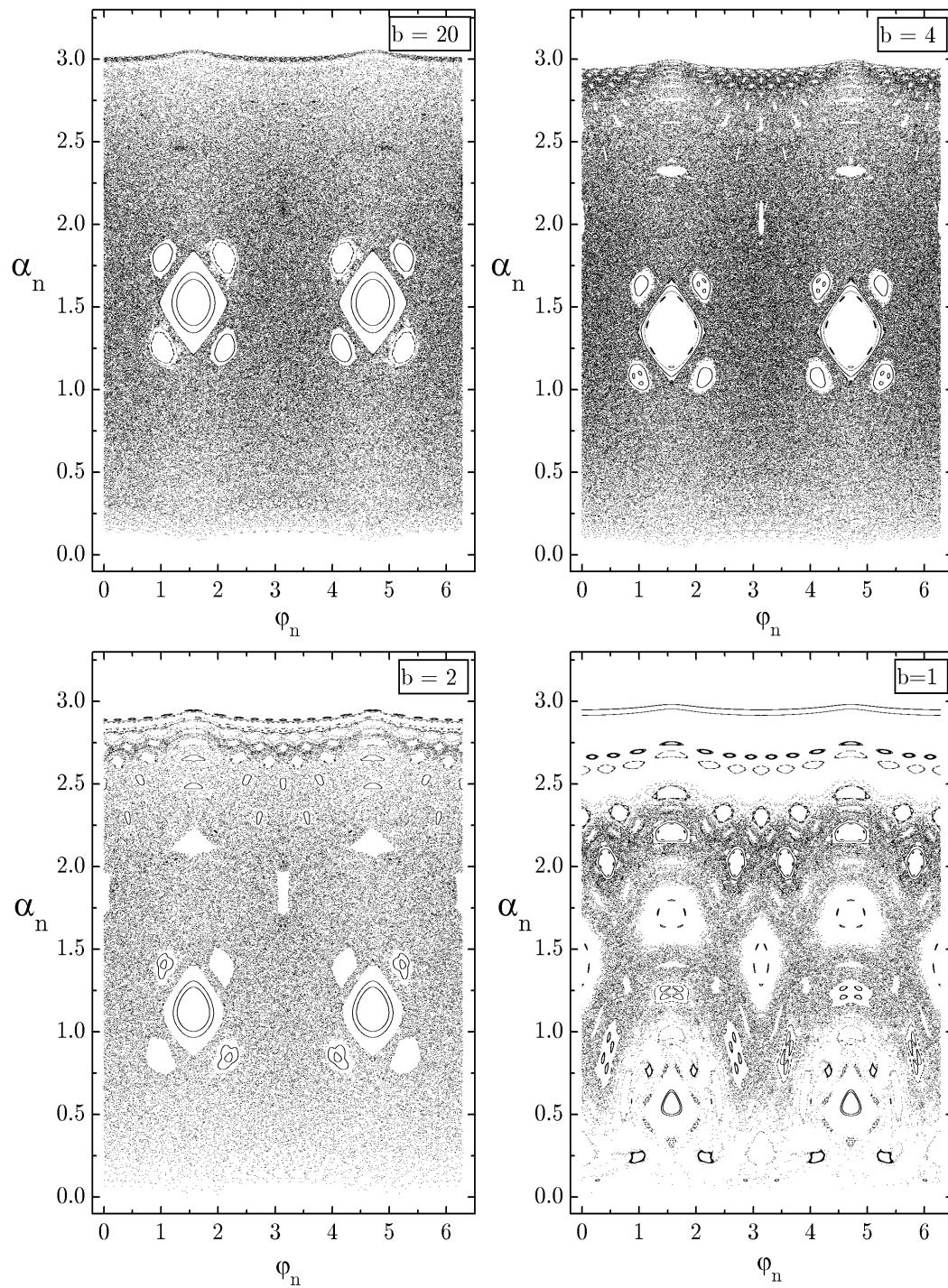


FIG. 2.11 – Diagramme des phases d'un Billard Ovale Magnétique.

Cette figure montre le diagramme des phases du Billard Ovale Magnétique ( $\epsilon = 0.15$ ) pour différentes valeurs de rayons de Larmor  $b$ . On remarque que pour un champ magnétique plus intense ( $b = 1$ ), le régime redevient plus régulier.

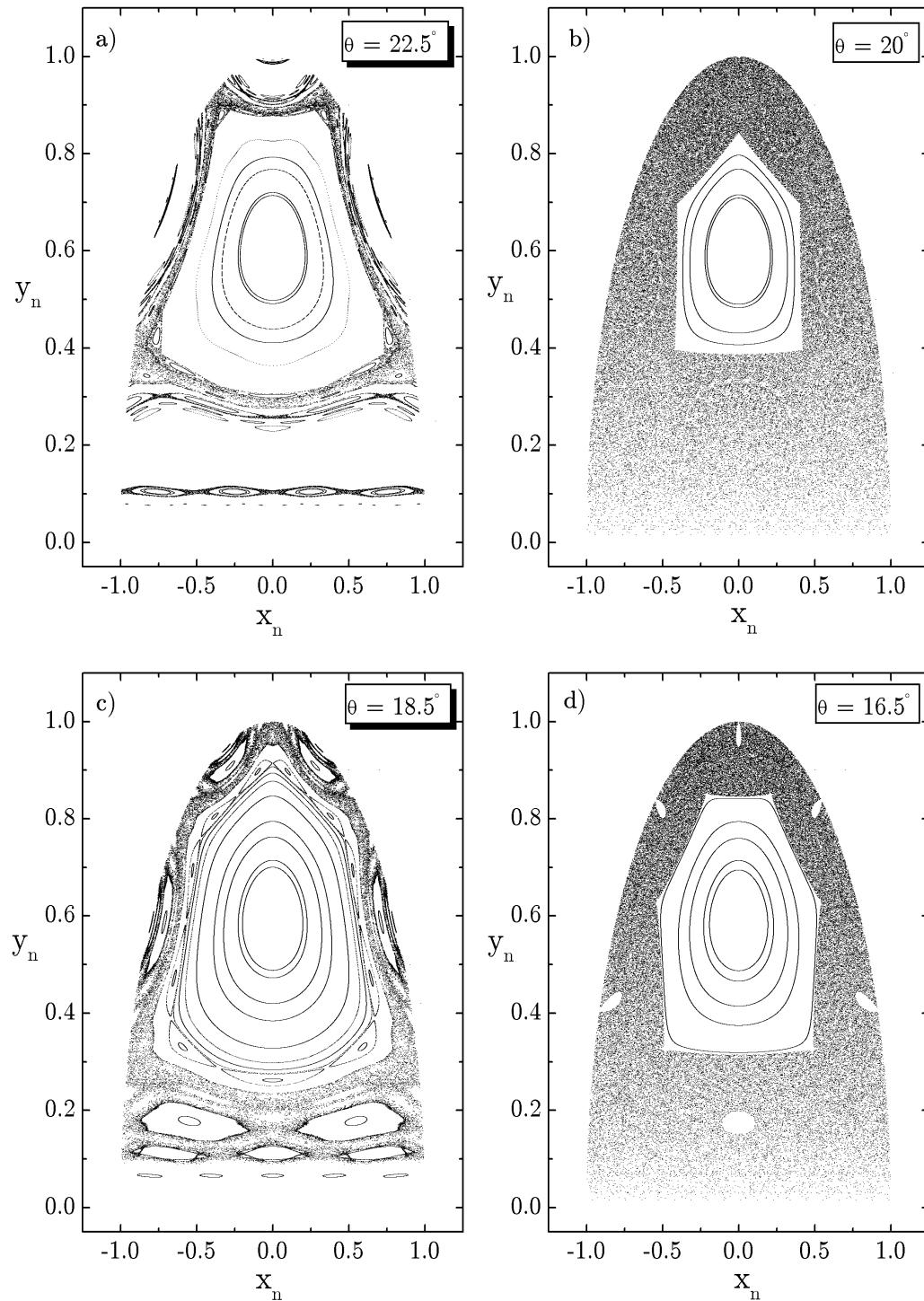


FIG. 2.12 – Diagramme des phases d'un Billard Gravitationnel.

Cette figure montre le diagramme des phases du Billard Gravitationnel pour différents angles  $\theta$ . Les figures a) et c) correspondent à une situation où  $\theta = 90^\circ/n$  : a)  $\rightarrow n = 4$ ; c)  $\rightarrow n = 5$ . Il y a pour ces cas une diminution des régions chaotiques.

## 2.4 Flots continus

### 2.4.1 Considérations générales

Les applications discrètes présentées jusqu'à maintenant ont toutes un aspect pédagogique indéniable. Certaines posent aussi de bons défis quant à la détection complète de leurs OPIs. Toutefois, notre étude ne saurait être complète sans y inclure les systèmes dynamiques *continus*. Les *flots continus* occupent une place importante dans le monde des systèmes dynamiques en général car la modélisation des phénomènes naturels se ramène souvent sous la forme d'équations différentielles. Que l'on s'intéresse à la trajectoire d'un électron sous l'influence combinée d'un potentiel coulombien et d'un champ magnétique ou encore au mouvement des "rouleaux" qui décrivent la convection dans l'atmosphère, on doit éventuellement passer par la résolution d'un système à  $N$  équations différentielles :

$$\frac{d\tilde{\mathbf{v}}}{dt} = \tilde{\mathbf{f}}(\tilde{\mathbf{v}}), \quad (2.31)$$

soit :

$$\begin{pmatrix} \frac{d\tilde{v}^1}{dt} \\ \vdots \\ \frac{d\tilde{v}^N}{dt} \end{pmatrix} = \begin{pmatrix} \tilde{f}^1(\tilde{\mathbf{v}}) \\ \vdots \\ \tilde{f}^N(\tilde{\mathbf{v}}) \end{pmatrix}.$$

La dynamique s'obtient ici en solutionnant (de façon numérique dans la plupart des cas) ce système d'équations différentielles. L'état  $\tilde{\mathbf{v}}(t)$  est obtenu en spécifiant une condition initiale  $\tilde{\mathbf{v}}$  et un temps d'intégration  $t$ . On note alors  $(\tilde{\mathbf{v}}, t) \mapsto \tilde{\phi}(t, \tilde{\mathbf{v}}) \equiv \tilde{\mathbf{v}}(t)$  où  $\tilde{\mathbf{v}}(t)$  est la solution obtenue en intégrant le système pendant un temps  $t$  à partir de la condition initiale  $\tilde{\mathbf{v}}$ .  $\tilde{\phi}$  est le flot, on note aussi  $\tilde{\phi}(t, \tilde{\mathbf{v}}) = \tilde{\phi}_t(\tilde{\mathbf{v}})$ . Aussi, nous utilisons dans ce qui suit le symbole  $\sim$  sur les vecteurs pour signaler qu'il s'agit de vecteurs colonnes à  $N$  composantes ( $N$  dimensions). Nous avons inclus un tableau "aide-mémoire" à l'annexe C qui résume les notations employées pour les flots continus.

### 2.4.1.1 Application discrète à partir du flot

Pour une dynamique chaotique confinée à une certaine région de l'espace des phases, il peut être utile de construire une application discrète à partir du flot continu. Comme nous l'avons vu précédemment, les algorithmes de *SD* et *DL* demandent de ramener ce flot à une application discrète  $\mathbf{F}(\mathbf{v}_n)$  de dimension  $D$ . Cette application s'obtient en collectant les coordonnées des intersections successives entre la trajectoire continue et une hypersurface  $\mathcal{P}$  de dimension  $N - 1$  appelée *Section de Poincaré*. Si cette surface est la seule contrainte disponible pour la construction de  $\mathbf{F}(\mathbf{v}_n)$ , alors on obtient une application en dimension  $D = N - 1$ . En général cependant, il peut y avoir au total  $N_c$  contraintes entre les variables du système (dues aux propriétés physiques d'un système donné : conservation de l'énergie, du moment cinétique,...) et on peut écrire :

$$D = N - N_c.$$

Ces  $N_c$  contraintes (incluant celle de la Section de Poincaré) sont importantes puisqu'elles permettent d'établir un sous système discret de dimension inférieure à celle du flot initial. Nous verrons un peu plus loin un exemple particulier où la conservation de l'énergie impose une autre contrainte de sorte que  $D = N - 2$  (Système DKP).

Dans ce qui suit, nous utilisons le symbole  $\tilde{\mathbf{v}}(t)(= \tilde{\phi}_t(\tilde{\mathbf{v}}))$  pour désigner l'état du flot continu au temps  $t$ , il s'agit d'un vecteur à  $N$  composantes. Le symbole  $\tilde{\mathbf{v}}_n$  est réservé pour décrire l'état *sur* la Section de Poincaré à l'itération  $n$ ; il s'agit aussi d'un vecteur à  $N$  composantes. Enfin, le vecteur  $\mathbf{v}_n$ , comme précédemment, spécifie l'évolution du système dans le sous-système discret  $\mathbf{F}(\mathbf{v}_n)$ . (vecteur à  $D$  composantes). C'est ce dernier vecteur que nous utilisons pour la détection des orbites périodiques. Le vecteur  $\mathbf{v}_n$  n'est pas un nouveau système de coordonnées mais bien un sous vecteur de  $\tilde{\mathbf{v}}$ . Les  $N_c$  contraintes permettent de passer de  $\tilde{\mathbf{v}}_n$  à  $\mathbf{v}_n$  en éliminant  $N_c$  variables. Les variables éliminées sont notées  $\check{\mathbf{v}}_n$ .

Il faut ici ajouter que les  $N_c$  contraintes sont en fait spécifiées par des équations.

Par exemple, pour un flot continu avec  $N = 3$  et une seule contrainte provenant d'une Section de Poincaré plane, les composantes de  $\tilde{\mathbf{v}}_n$  sont reliées entre elles par l'équation du plan qui définit la section. En spécifiant  $D = N - 1$  de ces composantes (soit  $\mathbf{v}_n$ ), on peut déduire celle qui manque à l'aide de l'équation du plan. En général, les composantes que l'on élimine ( $\check{\mathbf{v}}$ ) dépendent de celles que l'on garde ( $\mathbf{v}_n$ ) et il faut écrire  $\check{\mathbf{v}}_n(\mathbf{v}_n)$ . Ainsi, un état discret obtenu sur la Section de Poincaré  $\tilde{\mathbf{v}}_n$  comprend un ensemble de composantes que nous utilisons pour construire l'application discrète  $(\mathbf{F}(\mathbf{v}_n))$  et un ensemble de composantes qui dépendent de  $\mathbf{v}_n$  par différentes contraintes :

$$\tilde{\mathbf{v}}_n = (\mathbf{v}_n, \check{\mathbf{v}}_n(\mathbf{v}_n))^T . \quad (2.32)$$

Cette notation peut paraître quelque peu lourde mais elle s'avère toutefois nécessaire pour établir correctement la matrice jacobienne de l'application discrète  $\mathbf{F}(\mathbf{v}_n)$ , comme nous le verrons un peu plus loin.

Nous allons décrire comment calculer la suite des  $\tilde{\mathbf{v}}_n$  à partir du flot continu. Cette suite s'obtient numériquement en déterminant les points d'intersection de la trajectoire avec une hypersurface. Comme hypersurface, il suffit en général, comme nous le ferons ici, de choisir un hyperplan. L'équation d'un hyperplan s'obtient facilement : soit  $\tilde{\mathbf{v}}_0$  un point de l'hyperplan et  $\tilde{\boldsymbol{\xi}}$ , un vecteur orthogonal à l'hyperplan (figure 2.13), alors l'équation de l'hyperplan est  $h(\tilde{\mathbf{v}}) = 0$  où

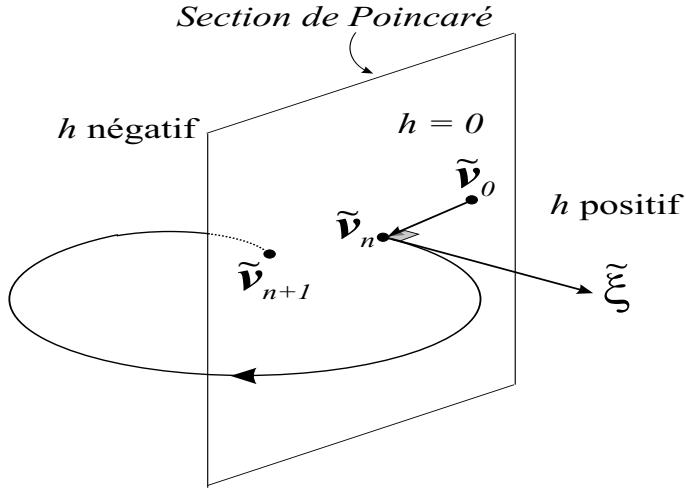
$$h(\tilde{\mathbf{v}}(t)) = \tilde{\boldsymbol{\xi}}^T (\tilde{\mathbf{v}}(t) - \tilde{\mathbf{v}}_0) , \quad (2.33)$$

soit

$$h(\tilde{\mathbf{v}}(t)) = (\tilde{\boldsymbol{\xi}}^1, \dots, \tilde{\boldsymbol{\xi}}^N) \begin{pmatrix} \tilde{v}^1(t) - \tilde{v}_0^1 \\ \vdots \\ \tilde{v}^N(t) - \tilde{v}_0^N \end{pmatrix} .$$

Soit  $\tilde{\mathbf{v}}(t)$  une solution. Alors  $h(\tilde{\mathbf{v}}(t)) = 0$  lorsque  $\tilde{\mathbf{v}}(t)$  est précisément sur l'hyperplan et  $h(\tilde{\mathbf{v}}(t))$  change de signe lorsque  $\tilde{\mathbf{v}}(t)$  passe d'un côté à l'autre de l'hyperplan.

On obtient l'application discrète  $\tilde{\mathbf{v}}_n \mapsto \tilde{\mathbf{v}}_{n+1}$  en intégrant le système 2.31 à partir

FIG. 2.13 – Section de Poincaré et définition de  $h(\tilde{\mathbf{v}}(t))$ .

de  $\tilde{\mathbf{v}}_n$  pendant un temps  $\tau(\tilde{\mathbf{v}}_n) = \tau_n$  :

$$\tilde{\mathbf{v}}_{n+1} = \tilde{\phi}(\tilde{\mathbf{v}}_n, \tau(\tilde{\mathbf{v}}_n)),$$

de telle sorte que  $\tilde{\mathbf{v}}_{n+1}$  soit sur l'hyperplan et vérifie la condition d'orientation que nous allons maintenant spécifier. Comme la trajectoire peut traverser la section dans un sens comme dans l'autre, il faut aussi spécifier une direction pour le flot. Par convention, on choisit de garder les points d'intersections pour lesquels le flot se dirige dans le sens positif, qui est défini comme le sens pour lequel  $h(\tilde{\mathbf{v}}(t))$  est positif. On désignera :

$$\tilde{\mathbf{v}}_{n+1} = \tilde{\mathbf{F}}_{SP}(\tilde{\mathbf{v}}_n) \equiv \tilde{\phi}(\tilde{\mathbf{v}}_n, \tau(\tilde{\mathbf{v}}_n)). \quad (2.34)$$

Une méthode simple pour collecter ces points est d'avancer la solution numériquement et de vérifier si  $h(\tilde{\mathbf{v}}(t))$  change de signe. Si tel est le cas, on revient en arrière et on avance avec un pas d'intégration plus petit. On reprend jusqu'à ce que le critère de précision soit atteint.

Une autre méthode connue est celle de Hénon [43], [69]. C'est la technique que nous utilisons pour tous les flots présentés dans cette thèse. Pour cette méthode particulière, on doit d'abord introduire la variable  $u \equiv h(\tilde{\mathbf{v}}(t))$  et augmenter d'une équation le

système d'équations différentielles 2.31. La  $(N + 1)^{\text{ième}}$  dérivée est alors donnée par :

$$\frac{du}{dt} = D_{\tilde{\mathbf{v}}} h(\tilde{\mathbf{v}}) \frac{d\tilde{\mathbf{v}}}{dt} = \tilde{\boldsymbol{\xi}}^T \tilde{\mathbf{f}}(\tilde{\mathbf{v}}). \quad (2.35)$$

En général, comme  $\tilde{\mathbf{f}}(\tilde{\mathbf{v}})$  est transverse à l'hyperplan, on a  $\frac{du}{dt} \neq 0$ . Ainsi, on peut inverser et considérer  $t$  comme une fonction de  $u$ . On obtient alors  $\frac{dt}{du} = 1/\frac{du}{dt}$ . En prenant  $u$  comme variable indépendante, le système 2.31 augmenté de 2.35 devient :

$$\frac{d\tilde{\mathbf{v}}^+}{du} = \frac{d\tilde{\mathbf{v}}^+}{dt} \frac{dt}{du} = \begin{pmatrix} \frac{d\tilde{v}^1}{du} \\ \vdots \\ \frac{d\tilde{v}^N}{du} \\ \frac{dt}{du} \end{pmatrix} = \begin{pmatrix} \frac{\tilde{f}^1(\tilde{\mathbf{v}}^+)}{\tilde{\boldsymbol{\xi}}^T \tilde{\mathbf{f}}(\tilde{\mathbf{v}}^+)} \\ \vdots \\ \frac{\tilde{f}^N(\tilde{\mathbf{v}}^+)}{\tilde{\boldsymbol{\xi}}^T \tilde{\mathbf{f}}(\tilde{\mathbf{v}}^+)} \\ \frac{1}{\tilde{\boldsymbol{\xi}}^T \tilde{\mathbf{f}}(\tilde{\mathbf{v}}^+)} \end{pmatrix}. \quad (2.36)$$

Ici nous avons ajouté l'exposant  $+$  après  $\tilde{\mathbf{v}}$  pour indiquer qu'il s'agit d'un système à  $N + 1$  équations. La dimension supplémentaire sert à calculer l'évolution du temps en fonction de la nouvelle variable  $u$ . Ceci est essentiel pour évaluer  $\tau(\tilde{\mathbf{v}}_n)$  précisément. Mentionnons que l'équation associée à la dimension supplémentaire est très similaire à celle ajoutée pour ramener un système non-autonome à la forme autonome.

Pour obtenir numériquement l'itéré  $\tilde{\mathbf{v}}_{n+1}$ , on procède alors comme suit :

1. On intègre 2.31 à partir de  $\tilde{\mathbf{v}}_n$  jusqu'à ce que la fonction  $h$  change de signe 2 fois (passe d'abord de positif à négatif en traversant la section dans le sens négatif et ensuite de négatif à positif lorsque le flot traverse cette section dans le sens positif). On arrête d'intégrer *juste après* avoir traversé la section dans le sens positif.
2. À ce stade, on est en  $\tilde{\mathbf{v}}(t^*)$  avec  $t^* > \tau_n$  et  $u = h(\tilde{\mathbf{v}}(t^*))$  positif. Rappelons que  $u = 0$  sur la Section de Poincaré. On change maintenant de variable d'intégration et on prend  $u$  à la place de  $t$  (équation 2.36). Il faut intégrer de  $u = h(\tilde{\mathbf{v}}(t^*))$  à  $u = 0$ .
3. On propose un pas  $du = -h(\tilde{\mathbf{v}}(t^*))$  à l'intégrateur. Si le critère de précision est respecté, alors le système sera amené d'un seul pas à  $u = 0$  et à  $\tilde{\mathbf{v}}_{n+1}$ . Le temps  $dt^*$

correspondant au pas  $du$  est donné par la dernière variable du système d'équations 2.36. Ainsi,  $\tau_n = t^* + dt^*$ . Si le critère de précision n'est pas respecté, on diminue simplement la grandeur du pas et on effectue plusieurs pas pour tomber sur  $u = 0$ .

Le vecteur dont nous avons besoin ( $\mathbf{v}_{n+1}$ ) a cependant  $D$  composantes. Ainsi, on ne garde que  $D$  composantes de  $\tilde{\mathbf{v}}_{n+1}$  pour construire l'application  $\mathbf{v}_{n+1} = \mathbf{F}(\mathbf{v}_n)$ .

#### 2.4.1.2 Matrice jacobienne de l'application discrète

Le calcul de la matrice jacobienne de l'application discrète  $\mathbf{F}(\mathbf{v}_n)$  obtenue à partir d'un flot continu peut être quelque peu subtil et il faut procéder avec "prudence". À notre connaissance, il n'existe pas de référence qui décrive la démarche à suivre pour évaluer cette matrice dans le cas où il y aurait plusieurs contraintes entre les variables. Nous allons donc présenter une dérivation de la procédure à suivre, d'abord d'une façon plus intuitive et ensuite de façon plus rigoureuse.

Pour connaître la matrice jacobienne de l'application discrète  $\mathbf{F}(\mathbf{v}_n)$ , il faut passer par l'intégration des équations variationnelles du système continu. On note  $\tilde{\mathbf{J}}_{\tilde{\phi}}(t, \tilde{\mathbf{v}})$  la matrice jacobienne du *flot*  $\tilde{\phi}_t(\tilde{\mathbf{v}})$ . Elle est solution du système d'équations suivant [69], [18] :

$$\frac{d\tilde{\mathbf{J}}_{\tilde{\phi}}(t, \tilde{\mathbf{v}})}{dt} = D_{\tilde{\mathbf{v}}}\tilde{\mathbf{f}}(\tilde{\mathbf{v}}) \tilde{\mathbf{J}}_{\tilde{\phi}}(t, \tilde{\mathbf{v}}), \quad (2.37)$$

avec la condition initiale  $\tilde{\mathbf{J}}_{\tilde{\phi}}(t = 0, \tilde{\mathbf{v}}) = \mathbf{1}$ , où  $\tilde{\mathbf{f}}(\tilde{\mathbf{v}})$  correspond aux dérivées du système dynamique par rapport au temps (équation 2.31).

Cette première matrice est celle des variations du flot par rapport à  $\tilde{\mathbf{v}}(t)$  :

$$\tilde{\mathbf{J}}_{\tilde{\phi}}(t, \tilde{\mathbf{v}}) \equiv D_{\tilde{\mathbf{v}}}\tilde{\phi}(t, \tilde{\mathbf{v}}) = \begin{pmatrix} \frac{\partial \tilde{\phi}_t^1}{\partial \tilde{v}^1} & \dots & \frac{\partial \tilde{\phi}_t^1}{\partial \tilde{v}^N} \\ \vdots & \ddots & \vdots \\ \frac{\partial \tilde{\phi}_t^N}{\partial \tilde{v}^1} & \dots & \frac{\partial \tilde{\phi}_t^N}{\partial \tilde{v}^N} \end{pmatrix}. \quad (2.38)$$

La figure 2.14 donne une interprétation géométrique de ce que fait cette dernière matrice sur une perturbation  $\delta\tilde{\mathbf{v}}_n$  ajoutée à  $\tilde{\mathbf{v}}_n$ . Le vecteur  $\delta\tilde{\mathbf{v}}(\tau_n)$  correspond à la perturbation

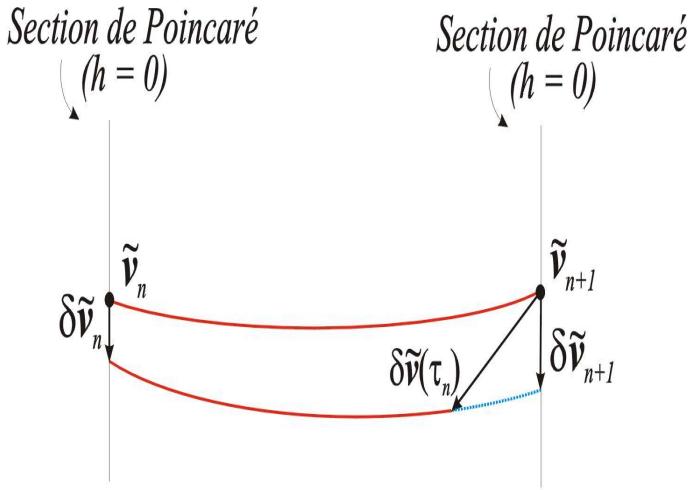


FIG. 2.14 – Interprétation géométrique de l'équation 2.39.

au temps  $\tau_n$  et on a :

$$\delta\tilde{\mathbf{v}}(\tau_n) = \tilde{\mathbf{J}}_{\tilde{\phi}}(t = \tau_n, \tilde{\mathbf{v}} = \tilde{\mathbf{v}}_n) \delta\tilde{\mathbf{v}}_n.$$

Sur cette figure, les 2 lignes pleines (—) sont des trajectoires intégrées pendant un temps  $\tau_n$ . Une de ces trajectoires retourne sur la Section de Poincaré après ce laps de temps. Comme en général  $\tau$  dépend de  $\tilde{\mathbf{v}}_n$ , l'autre trajectoire n'aboutit pas sur la section. Pendant ce temps  $\tau_n$ , la matrice donnée par l'équation 2.38 évolue jusqu'à  $\tilde{\mathbf{J}}_{\tilde{\phi}}(\tau_n)$ ; elle relie ainsi la perturbation  $\delta\tilde{\mathbf{v}}_n$  (initialement sur la Section de Poincaré) à une autre perturbation  $\delta\tilde{\mathbf{v}}(\tau_n)$ , cette dernière n'étant pas nécessairement sur la Section de Poincaré.

Il faut donc d'abord chercher à déterminer la matrice jacobienne de l'application  $\tilde{\mathbf{F}}_{SP}(\tilde{\mathbf{v}}_n)$  (équation 2.34), soit l'application discrète sur la Section de Poincaré. Cette matrice particulière relie la perturbation  $\delta\tilde{\mathbf{v}}_n$  à  $\delta\tilde{\mathbf{v}}_{n+1}$ , ces quantités étant situées *sur* la Section de Poincaré :

$$\begin{aligned}\tilde{\mathbf{J}}_{SP}(\tilde{\mathbf{v}}_n) &= D_{\tilde{\mathbf{v}}} \tilde{\mathbf{F}}_{SP}(\tilde{\mathbf{v}}) \Big|_{\tilde{\mathbf{v}}_n, \tau_n} \\ &= D_{\tilde{\mathbf{v}}} \tilde{\phi}(\tilde{\mathbf{v}}, \tau(\tilde{\mathbf{v}})) \Big|_{\tilde{\mathbf{v}}_n, \tau_n} \\ &= D_{\tilde{\mathbf{v}}} \tilde{\phi} \Big|_{\tilde{\mathbf{v}}_n, \tau_n} + \left[ \frac{\partial \tilde{\phi}}{\partial \tau} D_{\tilde{\mathbf{v}}} \tau \right]_{\tilde{\mathbf{v}}_n, \tau_n}\end{aligned}$$

$$= D_{\tilde{\mathbf{v}}}\tilde{\phi}|_{\tilde{\mathbf{v}}_n, \tau_n} + [\tilde{\mathbf{f}}(\tilde{\phi})D_{\tilde{\mathbf{v}}}\tau]_{\tilde{\mathbf{v}}_n, \tau_n}. \quad (2.39)$$

Le terme  $\tilde{\mathbf{f}}(\tilde{\phi})D_{\tilde{\mathbf{v}}}\tau|_{\tilde{\mathbf{v}}_n, \tau_n}$  nous donne en quelque sorte ce qu'il "manque" à  $D_{\tilde{\mathbf{v}}}\tilde{\phi}|_{\tilde{\mathbf{v}}_n, \tau_n}$  ( $= \tilde{\mathbf{J}}_{\tilde{\phi}}(t, \tilde{\mathbf{v}})|_{\tilde{\mathbf{v}}_n, \tau_n}$ ) pour amener la perturbation  $\delta\tilde{\mathbf{v}}(\tau_n)$  sur la Section de Poincaré (trait — — sur la figure 2.14).

Le premier terme de l'équation 2.39 ( $D_{\tilde{\mathbf{v}}}\tilde{\phi}$ ) est donné directement par la matrice des variations  $\tilde{\mathbf{J}}_{\tilde{\phi}}$  (équation 2.38) évaluée à  $(\tilde{\mathbf{v}}_n, \tau_n)$ . Il s'agit donc d'intégrer l'équation 2.37 à partir de  $\tilde{\mathbf{v}}_n$  pendant un temps  $t = \tau_n$ . Une partie du deuxième terme est aussi connue, soit le terme  $\tilde{\mathbf{f}}(\tilde{\phi})|_{\tilde{\mathbf{v}}_n, \tau_n}$  : ce sont les dérivées du système par rapport au temps (équation 2.31) évaluées à  $\tilde{\phi}(\tilde{\mathbf{v}}_n, \tau_n) = \tilde{\mathbf{v}}_{n+1}$ . Reste maintenant à calculer les dérivées  $D_{\tilde{\mathbf{v}}}\tau$ . Nous avons déjà défini la fonction  $h(\tilde{\mathbf{v}}(t)) = h(\tilde{\phi}_t(\tilde{\mathbf{v}}))$ , fonction qui est précisément égale à zéro sur la Section de Poincaré. Ainsi :

$$h(\tilde{\phi}(\tilde{\mathbf{v}}_n, \tau(\tilde{\mathbf{v}}_n))) = \tilde{\xi}^T (\tilde{\phi}(\tilde{\mathbf{v}}_n, \tau(\tilde{\mathbf{v}}_n)) - \tilde{\mathbf{v}}_0) = 0.$$

En dérivant  $h$  par rapport à  $\tilde{\mathbf{v}}$ , on déduit alors que :

$$\tilde{\xi}^T \left( D_{\tilde{\mathbf{v}}}\tilde{\phi} + \frac{\partial \tilde{\phi}}{\partial \tau} D_{\tilde{\mathbf{v}}}\tau \right)_{\tilde{\mathbf{v}}_n, \tau_n} = 0.$$

Comme  $\frac{\partial \tilde{\phi}}{\partial \tau} = \tilde{\mathbf{f}}(\tilde{\phi})$ , on tire de cette dernière relation :

$$D_{\tilde{\mathbf{v}}}\tau|_{\tilde{\mathbf{v}}_n, \tau_n} = -\frac{\tilde{\xi}^T D_{\tilde{\mathbf{v}}}\tilde{\phi}|_{\tilde{\mathbf{v}}_n, \tau_n}}{\tilde{\xi}^T \tilde{\mathbf{f}}(\tilde{\phi})_{\tilde{\mathbf{v}}_n, \tau_n}}. \quad (2.40)$$

En pratique, il faut résoudre numériquement et simultanément les systèmes 2.31 (à partir d'une condition initiale  $\tilde{\mathbf{v}}_n$  choisie sur la Section de Poincaré) et 2.37 (à partir de  $\tilde{\mathbf{J}}_{\tilde{\phi}} = \mathbf{1}$ ). On intègre pendant un temps  $\tau_n$  (ce temps peut être déterminé, par exemple, par la méthode de Hénon). Après l'intégration, on connaît  $\tilde{\mathbf{v}}_{n+1}$  et  $\tilde{\mathbf{J}}_{\tilde{\phi}}$ . On calcule finalement  $\tilde{\mathbf{J}}_{SP}(\tilde{\mathbf{v}}_n)$  à partir de 2.39 et 2.40.

Nous avons ainsi réalisé la première étape qui consiste à obtenir la matrice jacobienne

sur la Section de Poincaré<sup>1</sup>. Il s'agit d'une matrice  $N \times N$ . Nous avons cependant besoin de la matrice jacobienne de l'application  $\mathbf{F}(\mathbf{v}_n)$ , qui est une matrice  $D \times D$ . On ne peut ici simplement prendre la matrice  $\tilde{\mathbf{J}}_{SP}$  et rejeter les lignes et les colonnes des variables qui ont été éliminées lors du passage de  $\tilde{\mathbf{v}}_n$  vers  $\mathbf{v}_n$  car ces premières dépendent des  $\mathbf{v}_n$ .

Afin bien comprendre ce que nous venons d'affirmer, nous allons d'abord supposer que les  $D$  premières composantes de  $\tilde{\mathbf{v}}_n$  sont gardées pour former le vecteur  $\mathbf{v}_n$  et que les  $N_c$  autres composantes (que nous avons notées par  $(\check{v}_n)$ ) sont celles éliminées par les différentes contraintes ( $N_c = N - D$ ) :

$$\begin{aligned}\tilde{\mathbf{v}}_n &= (\mathbf{v}_n, \check{\mathbf{v}}_n(\mathbf{v}_n))^T \\ &= \begin{pmatrix} v_n^1 \\ \vdots \\ v_n^D \\ \check{v}_n^1(\mathbf{v}_n) \\ \vdots \\ \check{v}_n^{N_c}(\mathbf{v}_n) \end{pmatrix}. \end{aligned} \quad (2.41)$$

Écrivons ensuite explicitement la forme de la matrice jacobienne  $\tilde{\mathbf{J}}_{SP}$  sur la Section de Poincaré et son effet sur une perturbation  $\delta\tilde{\mathbf{v}}_n$ .

$$\delta\tilde{\mathbf{v}}_{n+1} = \tilde{\mathbf{J}}_{SP}(\tilde{\mathbf{v}}_n) \delta\tilde{\mathbf{v}}_n,$$

soit :

$$\begin{pmatrix} \delta v_{n+1}^1 \\ \vdots \\ \delta v_{n+1}^D \\ \delta \check{v}_{n+1}^1 \\ \vdots \\ \delta \check{v}_{n+1}^{N_c} \end{pmatrix} = \begin{pmatrix} \tilde{J}_{11} & \dots & \tilde{J}_{1D} & \dots & \tilde{J}_{1N} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \tilde{J}_{D1} & \dots & \tilde{J}_{DD} & \dots & \tilde{J}_{DN} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \tilde{J}_{N1} & \dots & \tilde{J}_{ND} & \dots & \tilde{J}_{NN} \end{pmatrix} \begin{pmatrix} \delta v_n^1 \\ \vdots \\ \delta v_n^D \\ \delta \check{v}_n^1 \\ \vdots \\ \delta \check{v}_n^{N_c} \end{pmatrix}. \quad (2.42)$$

---

<sup>1</sup>Le développement plus formel de cette première étape est fait dans Parker & Chua [69][Annexe D].

où les  $\tilde{J}_{ij}$  sont donnés par l'équation 2.39 soit  $\tilde{J}_{ij} = \left[ \frac{\partial \tilde{\phi}^i}{\partial \tilde{v}^j} + \tilde{f}^i(\tilde{\phi}) \frac{\partial \tau}{\partial \tilde{v}^j} \right]_{\tilde{\mathbf{v}}_n, \tau_n}$ . Les  $N_c$  équations des contraintes permettent d'écrire les  $\check{\mathbf{v}}_n$  en fonction des  $\mathbf{v}_n$ . Pour la  $k^{ième}$  composante de  $\check{\mathbf{v}}_n$  on a :

$$\delta \check{v}_n^k = \sum_{j=1}^D \frac{\partial \check{v}^k}{\partial v^j} \Big|_{\mathbf{v}_n} \delta v_n^j. \quad (2.43)$$

On cherche maintenant la matrice jacobienne  $\mathbf{J}$  de l'application  $\mathbf{F}(\mathbf{v}_n)$  soit celle qui décrit l'évolution d'une perturbation  $\delta \mathbf{v}_n$  :

$$\begin{pmatrix} \delta v_{n+1}^1 \\ \vdots \\ \delta v_{n+1}^D \end{pmatrix} = \begin{pmatrix} J_{11} & \dots & J_{1D} \\ \vdots & \ddots & \vdots \\ J_{D1} & \dots & J_{DD} \end{pmatrix} \begin{pmatrix} \delta v_n^1 \\ \vdots \\ \delta v_n^D \end{pmatrix}. \quad (2.44)$$

En regroupant les équation 2.42, 2.43 et 2.44 on arrive à une relation pour les éléments de la matrice jacobienne  $\mathbf{J}$  :

$$J_{ij} = \tilde{J}_{ij} + \sum_{k=1}^{N_c} \tilde{J}_{i(D+k)} \frac{\partial \check{v}^k}{\partial v^j} \Big|_{\mathbf{v}_n} \quad \text{pour } 1 \leq i, j \leq D.$$

Sachant qu'il y a  $D \times D$  de ces éléments, on peut ramener la matrice jacobienne  $\mathbf{J}$  à la forme suivante :

$$\mathbf{J}(\mathbf{v}_n) = \underbrace{\overbrace{D_{\mathbf{v}} \tilde{\phi} \Big|_{\tilde{\mathbf{v}}_n, \tau_n}}^{Sous matrice D \times D de \tilde{\mathbf{J}}_{\phi}} + \left[ \mathbf{f}(\tilde{\phi}) D_{\mathbf{v}} \tau \right]_{\tilde{\mathbf{v}}_n, \tau_n} + \sum_{k=1}^{N_c} \left\{ \frac{\partial \phi}{\partial \check{v}^k} + \mathbf{f}(\tilde{\phi}) \frac{\partial \tau}{\partial \check{v}^k} \right\}_{\tilde{\mathbf{v}}_n, \tau_n} D_{\mathbf{v}} \check{v}^k \Big|_{\mathbf{v}_n}}_{Sous matrice D \times D de \tilde{\mathbf{J}}_{SP}} \underbrace{\text{Projection de la sous matrice } \tilde{\mathbf{J}}_{SP} \text{ dans le système discret de coordonnées } \mathbf{v}_n}_{(2.45)}$$

Il est à noter que nous avons enlevé la plupart des  $\sim$  sur les vecteurs, ce qui indique maintenant un sous vecteur de dimension  $D$ . Par exemple  $\mathbf{f}(\tilde{\phi})$  correspond aux  $D$  composantes de  $\tilde{\mathbf{f}}$  que l'on garde pour construire l'application discrète (On laisse cependant le  $\sim$  sur  $\tilde{\phi}$  car  $\mathbf{f}$  est fonction des  $N$  composantes de  $\tilde{\phi}$ ).

La figure 2.15 permet de visualiser le rôle du dernier terme de cette équation ( $\sum_k$ ). Sur la figure, deux trajectoires sont tracées pour montrer comment évolue une perturbation dans un système à  $N = 2$  variables. On choisit arbitrairement d'éliminer  $\check{v}^2$  et

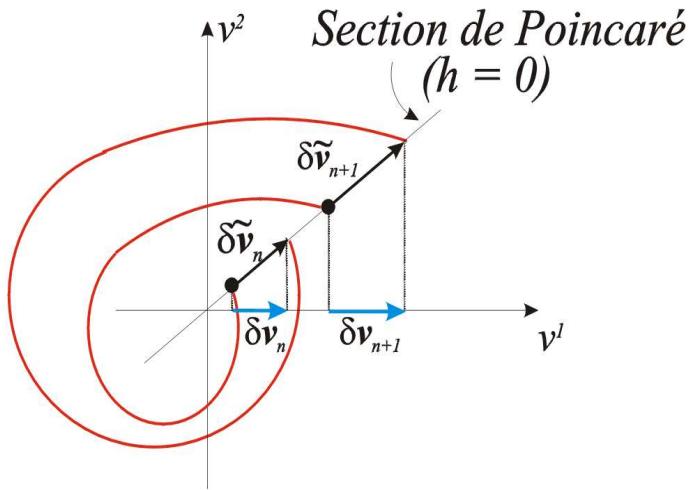


FIG. 2.15 – Interprétation géométrique de l'équation 2.45.

de construire une dynamique discrète à partir de  $\tilde{v}^1$  seulement. La matrice  $\tilde{\mathbf{J}}_{SP}$  relie les perturbations  $\tilde{\mathbf{v}}_n$  et  $\tilde{\mathbf{v}}_{n+1}$ . La matrice  $\mathbf{J}$  (équation 2.45) de l'application discrète  $v_{n+1}^1 \mapsto v_n^1$  fait le lien entre les projections de ces perturbations (vecteurs → ).

Nous avons présenté la dérivation de l'équation 2.15 en 2 étapes afin de faire ressortir la “nature” des différents éléments qui sont nécessaires au calcul de la matrice jacobienne de l'application  $\mathbf{F}(\mathbf{v}_n)$  obtenue à partir d'un flot continu. On peut évidemment arriver beaucoup plus rapidement à cette équation en partant directement de l'équation 2.34, que l'on réécrit ici :

$$\tilde{\mathbf{v}}_{n+1} = \tilde{\mathbf{F}}_{SP}(\tilde{\mathbf{v}}_n) = \tilde{\phi}(\tilde{\mathbf{v}}_n, \tau(\tilde{\mathbf{v}}_n)).$$

On sait que  $\tilde{\mathbf{v}}_n$  est composé en partie d'un vecteur contenant les états  $\mathbf{v}_n$  sur la Section de Poincaré et d'un vecteur  $\check{\mathbf{v}}_n$  contenant ceux qui sont éliminés par les contraintes. Ce dernier vecteur dépend de  $\mathbf{v}_n$  :

$$\tilde{\mathbf{v}}_n = (\mathbf{v}_n, \check{\mathbf{v}}_n(\mathbf{v}_n))^T.$$

L'application discrète  $\mathbf{F}(\mathbf{v}_n)$  de dimension  $D$  obtenue à partir du flot continu peut donc s'écrire :

$$\mathbf{v}_{n+1} = \mathbf{F}(\mathbf{v}_n) = \phi(\mathbf{v}_n, \check{\mathbf{v}}_n(\mathbf{v}_n), \tau(\mathbf{v}_n, \check{\mathbf{v}}_n(\mathbf{v}_n))).$$

Cette dernière équation indique que pour connaître le prochain itéré  $\mathbf{v}_{n+1}$  de l'application discrète de dimension  $D$ , il faut d'abord spécifier l'état  $\mathbf{v}_n$ , les  $N_c = N - D$  états  $\check{\mathbf{v}}_n$  (car pour intégrer le flot, il nous faut  $N$  conditions initiales). Ces  $\check{\mathbf{v}}_n$  s'obtiennent par les équations des contraintes et dépendent donc des  $\mathbf{v}_n$ . Finalement, il faut connaître le temps d'intégration  $\tau$  (ce temps dépend des  $N$  variables du système, soit les  $\mathbf{v}_n$  et les  $\check{\mathbf{v}}_n$ ).

La matrice jacobienne de  $\mathbf{F}$  s'obtient directement en dérivant cette dernière équation par rapport à  $\mathbf{v}$  :

$$\begin{aligned}\mathbf{J}(\mathbf{v}_n) &= D_{\mathbf{v}}\mathbf{F}(\mathbf{v}_n) \\ &= D_{\mathbf{v}}\phi(\mathbf{v}_n, \check{\mathbf{v}}_n(\mathbf{v}_n), \tau(\mathbf{v}_n, \check{\mathbf{v}}_n(\mathbf{v}_n))) \\ &= D_{\mathbf{v}}\phi\Big|_{\check{\mathbf{v}}_n, \tau_n} + [\mathbf{f}(\tilde{\phi})D_{\mathbf{v}}\tau]_{\check{\mathbf{v}}_n, \tau_n} + \sum_{k=1}^{N_c} \left\{ \frac{\partial \phi}{\partial \check{v}^k} + \mathbf{f}(\tilde{\phi}) \frac{\partial \tau}{\partial \check{v}^k} \right\}_{\check{\mathbf{v}}_n, \tau_n} D_{\mathbf{v}}\check{v}^k\Big|_{\mathbf{v}_n},\end{aligned}$$

ce qui est identique à l'équation 2.45.

À l'annexe C, nous donnons les équations spécifiques à l'implémentation d'un flot particulier, le flot DKP, qui est présenté un peu plus loin dans cette section. En plus d'une contrainte obtenue par la Section de Poincaré, il existe un lien entre les différentes variables pour ce système dû à la conservation de l'énergie. Mentionnons en terminant que si l'il n'y a qu'une contrainte provenant de la Section de Poincaré, le calcul de la matrice jacobienne peut être légèrement simplifié si on choisit comme section le plan  $\check{v} = \text{constante}$ . Autrement dit, la variable que l'on élimine de la dynamique discrète ne dépend pas des autres variables ( $D_{\mathbf{v}}\check{v} = 0$ ). On peut alors prendre directement la matrice jacobienne obtenue sur l'application de Poincaré ( $\tilde{\mathbf{J}}_{SP}$ ) et lui enlever la ligne et la colonne correspondant à la variable éliminée de la dynamique discrète.

### 2.4.1.3 Détection des points stationnaires du flot

Nous avons vu qu'il est possible d'obtenir une application discrète  $\mathbf{F}(\mathbf{v}_n)$  à partir du flot continu à l'aide d'une Section de Poincaré. Une question fort légitime qui peut survenir alors est : "Quelle section doit-on prendre ?" Cette section doit être traversée par le flot le plus fréquemment possible. Si l'hyperplan est mal positionné, on peut manquer certaines orbites périodiques.

Un bon critère est de faire en sorte que la Section de Poincaré passe par un *point stationnaire* du flot ( $\tilde{\mathbf{v}}^*$ ). On entend ici par point stationnaire le point pour lequel  $\tilde{\mathbf{f}}(\tilde{\mathbf{v}}^*) = 0$  i.e. un endroit où les dérivées du système dynamique sont égales à zéro (équation 2.31). En général, la dynamique est riche autour de ces points et il semble justifié de faire passer la section par un de ces points particuliers. Les points stationnaires peuvent généralement s'obtenir analytiquement en cherchant les zéros de  $\tilde{\mathbf{f}}$ . Si  $\tilde{\mathbf{f}}$  n'est pas continue analytiquement, on peut obtenir ces points stationnaires numériquement ... par l'algorithme de *SD* !

Cet algorithme a été inventé pour détecter, entre autre, les points fixes  $\mathbf{v}^*$  d'une application discrète  $\mathbf{F}(\mathbf{v})$  soit :

$$\mathbf{v}^* = \mathbf{F}(\mathbf{v}^*) .$$

Nous avons vu au chapitre 1.3 que ces points fixes sont en fait les zéros de la fonction  $\mathbf{G}$  (réf. équation 1.21) :

$$\mathbf{G}(\mathbf{v}) \equiv \mathbf{F}(\mathbf{v}) - \mathbf{v} .$$

Comme nous avons besoin des zéros de  $\tilde{\mathbf{f}}$ , il suffit de faire correspondre  $\mathbf{G}$  à  $\tilde{\mathbf{f}}$  et de chercher les points fixes d'une application discrète définie par :

$$\tilde{\mathbf{F}}(\tilde{\mathbf{v}}_n) \equiv \tilde{\mathbf{f}}(\tilde{\mathbf{v}}_n) + \tilde{\mathbf{v}}_n . \quad (2.46)$$

Les points fixes de cette fonction à  $N$  variables correspondent aux zéros de  $\tilde{\mathbf{f}}$  : ce sont les points stationnaires  $\tilde{\mathbf{v}}^*$  du flot. Ainsi pour la recherche des orbites périodiques des flots continus, on peut, comme étape préliminaire, rechercher les points stationnaires

du flot à l'aide de l'algorithme *SD*. On utilise alors cet algorithme en prenant comme application discrète la fonction  $\tilde{\mathbf{F}}$  de l'équation 2.46. Comme c'est une fonction à  $N$  variables, on doit prendre  $D = N$  dans l'algorithme. Lorsque les points stationnaires ont été repérés, on place la Section de Poincaré sur un de ceux-ci. Le vecteur  $\tilde{\xi}$ , qui spécifie l'orientation de l'hyperplan, doit être si possible tangent au flot. Un bon choix est de prendre  $\tilde{\xi} = \tilde{\mathbf{f}}(\tilde{\mathbf{v}}_0)$ . La recherche des orbites périodiques peut alors commencer.

### 2.4.2 Flot dissipatif : Système de Lorenz

Lorenz [55] a présenté au début des années '60, une étude théorique et numérique de la convection Rayleigh-Bénard, soit celle présente dans un fluide coincé entre deux plaques parallèles de température différente et soumis à un champ gravitationnel. La dynamique du fluide s'exprime au moyen d'équations aux dérivées partielles couplées. Lorenz a considéré une version simplifiée du système d'équations proposé un an plus tôt par Saltzman [87], version dans laquelle n'interviennent que trois fonctions du temps :

$$\text{Lorenz : } \frac{d\tilde{\mathbf{v}}}{dt} = \begin{pmatrix} \frac{d\tilde{v}^1}{dt} \\ \frac{d\tilde{v}^2}{dt} \\ \frac{d\tilde{v}^3}{dt} \end{pmatrix} \equiv \begin{pmatrix} \frac{dx}{dt} \\ \frac{dy}{dt} \\ \frac{dz}{dt} \end{pmatrix} = \begin{pmatrix} \sigma(y - x) \\ x(r - z) - y \\ xy - bz. \end{pmatrix}. \quad (2.47)$$

En prenant la divergence du flot, on remarque que le volume  $V(t)$  de l'espace des phases varie de façon exponentielle soit  $V(t) \propto \exp(-ct)$  avec  $c = \sigma + b + 1$ . Pour les valeurs standards,  $\sigma = 16$ ,  $b = 4$  et  $R = 45.92$ , il s'agit d'un flot dissipatif. L'attracteur de Lorenz en dimension 3 et sa section sur la surface définie par  $z(\tau_n) = R - 1 = 44.92$ ;  $\frac{dz}{dt}\Big|_{\tau_n} > 0$  sont illustrés à la figure 2.16. On choisit cette surface particulière car elle passe par les deux points stationnaires du flot  $(C^+, C^-)$  :  $x^* = y^* = \pm\sqrt{b(R - 1)}$ ,  $z^* = R - 1$ . Ces deux points stationnaires deviennent instables pour  $R > R_c \equiv \sigma(\sigma + b + 3)/(\sigma - b - 1)$  (Pour les valeurs choisies  $R_c = 33\frac{5}{11}$ ).

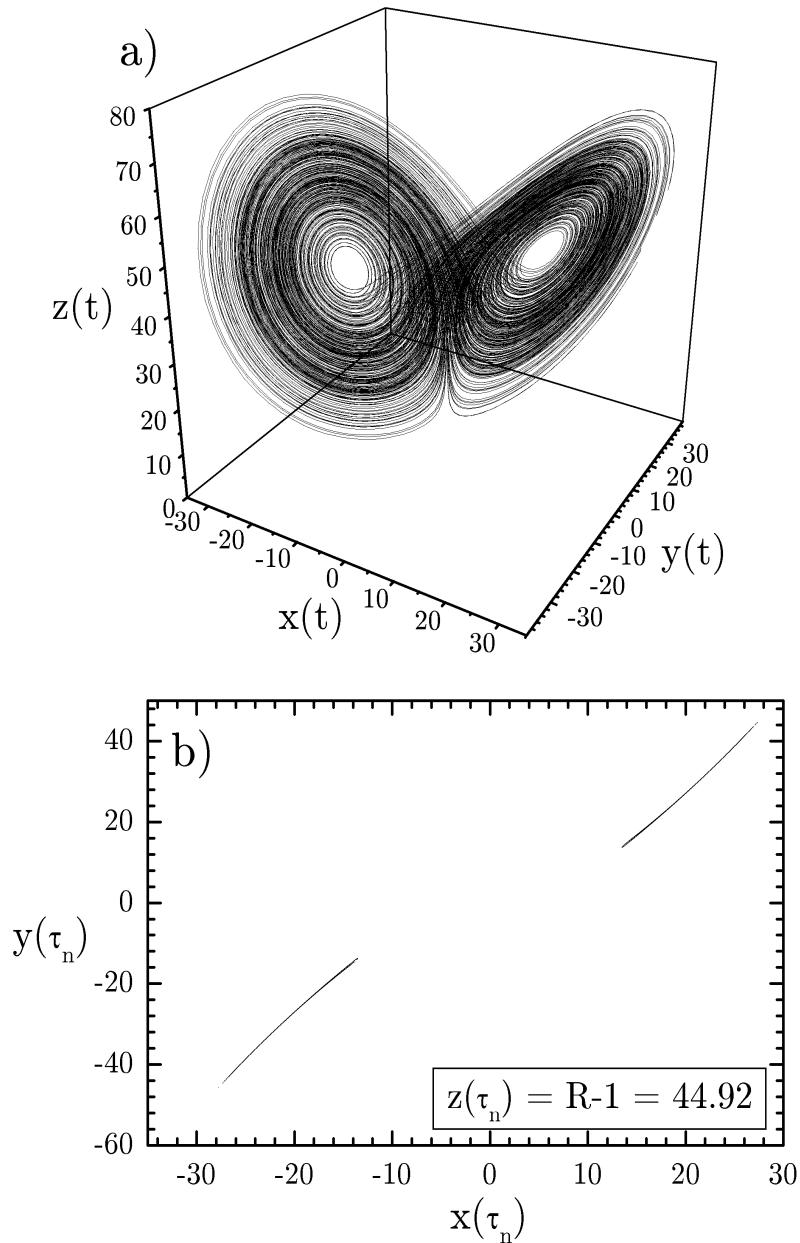


FIG. 2.16 – Attracteur de Lorenz et Section de Poincaré.

La figure 2.16a) illustre l'attracteur du système de Lorenz ( $\sigma = 16$ ,  $b = 4$  et  $R = 45.92$ ).

On montre en b) la trace de l'attracteur obtenue sur la Section de Poincaré définie par  $z(\tau_n) = R - 1 = 44.92$  et  $\frac{dz}{dt} \Big|_{\tau_n} > 0$ .

### 2.4.3 Flot Non-Autonome : Oscillateur de Duffing

L'oscillateur linéaire forcé et amorti est un sujet classique pour l'introduction aux phénomènes ondulatoires. Il est caractérisé par des oscillations périodiques régulières dont l'amplitude dépend de la fréquence de la force excitatrice. Si l'oscillateur est non-linéaire, le mouvement devient plus complexe. Son comportement à long terme dépend alors fortement des conditions initiales. Un tel oscillateur non linéaire est celui de Duffing dont l'équation du mouvement est :

$$\ddot{x} + \gamma \dot{x} + x + x^3 = f \cos(\omega t).$$

Cette équation peut se ramener à :

$$\text{Duffing : } \frac{d\tilde{\mathbf{v}}}{dt} = \begin{pmatrix} \frac{d\tilde{v}^1}{dt} \\ \frac{d\tilde{v}^2}{dt} \\ \frac{d\tilde{v}^3}{dt} \end{pmatrix} \equiv \begin{pmatrix} \frac{dx}{dt} \\ \frac{dy}{dt} \\ \frac{dz}{dt} \end{pmatrix} = \begin{pmatrix} y \\ -\gamma y - x - x^3 + f \cos(2\pi z) \\ \frac{\omega}{2\pi} \end{pmatrix}. \quad (2.48)$$

On introduit les variables  $y$  et  $z$  afin de ramener l'équation de Duffing à un système d'équations différentielles autonomes d'ordre 1 (comme l'équation 2.31). Ici,  $y$  représente une vitesse et  $z$ , un temps normalisé avec la période  $T \equiv 2\pi/\omega$  de la force excitatrice (soit le terme en cosinus de l'équation de Duffing). On note que les équations sont périodiques en  $z$  (période = 1). Ceci est vrai en général si le terme qui rend l'équation non-autonome est périodique.

Un bon choix ici pour la Section de Poincaré est de prendre  $z = \text{constante}$ . On intègre ainsi l'équation 2.48 en appliquant  $z \bmod 1$  le long de l'intégration. Autrement dit, lorsque le flot dépasse  $z = 1$ , on le ramène immédiatement à  $z = 0$ . Ce choix particulier fait en sorte que  $\tau_n$ , le temps d'intégration entre deux passages positifs à travers la Section de Poincaré, devient *indépendant* de l'endroit où l'on traverse cette section (soit  $\tilde{\mathbf{v}}_n$ ). Pour Duffing, ce choix donne  $\tau = T \equiv 2\pi/\omega$ .

On montre à la figure 2.17a, l'attracteur de Duffing sur la section ( $z(\tau_n) = 0, \frac{dz}{dt}|_{\tau_n} > 0$ ) pour les paramètres  $\gamma = 0.2$ ,  $\omega = 0.661$  et  $f = 36.0$ . Sur la figure 2.17b, on voit des

sections d'une trajectoire particulière qui tourbillonne dans le plan  $XY$  tout en montant vers  $z = 1$  (Les traits continus (—) montrent quelques portions de la trajectoire, qui explore en fait, une très grande région de l'espace des phases tout en montant). Lorsqu'elle arrive en  $z = 1$ , on la ramène instantanément en  $z = 0$  où elle reprend sa montée. La figure 2.17a est donc obtenue en collectant les coordonnées  $(x_n, y_n)$  lorsque  $z = 0$ .

En prenant la Section de Poincaré en  $z = \text{constante}$ , on obtient de plus une “balise” pour le calcul numérique de la matrice jacobienne. En effet, le flot de Duffing est dissipatif, i.e. qu'un volume  $V_0$  dans l'espace des phases se contracte selon :

$$\begin{aligned} V(t) &= V_0 e^{\nabla \cdot \mathbf{f}(\mathbf{v})t} \\ &= V_0 e^{-\gamma t} \end{aligned}$$

Entre deux passages sur la Section de Poincaré, le temps avance de  $\tau = T = 2\pi/\omega$ . Le volume doit donc changer d'un facteur :

$$\frac{V(t = \tau)}{V_0} = e^{-2\pi\gamma/\omega}$$

Ce volume change au même rythme que l'aire d'une surface sur la Section de Poincaré. À la figure 2.17b, nous avons placé un cône qui correspond à un certain volume dans l'espace des phases. La hauteur “ $h$ ” de ce cône correspond à la variation de  $z$  lors de deux passages consécutifs sur la Section de Poincaré. Cette hauteur est indépendante des trajectoires (en fait  $h = 1$ ). Entre deux passages sur la section, un volume donné change par un facteur  $e^{-2\pi\gamma/\omega}$ . Ce changement est donc uniquement dû à la variation d'aire  $dA$  sur la section de Poincaré. Ainsi, un élément d'aire  $dA_n$  change, à chaque itération sur la Section de Poincaré, dans les mêmes proportions qu'un élément de volume dans un temps  $\tau = T = \frac{2\pi}{\omega}$  :

$$\frac{dA_{n+1}}{dA_n} = \frac{V(t = \tau)}{V_0} = e^{-2\pi\gamma/\omega}$$

D'autre part, le déterminant de la matrice jacobienne au point  $\mathbf{v}_n$  sur la Section de Poincaré ( $|\mathbf{J}|$ ) est aussi égale à  $dA_{n+1}/dA_n$  et on déduit, pour l'oscillateur de Duffing,

avec cette Section de Poincaré (SP) particulière :

$$|\mathbf{J}_{Duffing}| = e^{-2\pi\gamma/\omega} = constante \quad (2.49)$$

(Avec SP donnée par  $z = constante$ )

Mentionnons aussi que l'équation 2.48 doit être intégrée "minutieusement" car les bassins d'attraction sont particulièrement enchevêtrés. Pour des valeurs identiques de  $\gamma$ ,  $\omega$  et  $f$ , différents régimes permanents peuvent être obtenus et ce, en fonction des conditions initiales. La figure 2.18 illustre une projection du diagramme des phases pour  $\gamma = 0.2$ ,  $\omega = 0.661$  et  $f = 36.0$ . La figure du haut correspond au régime chaotique (soit celui de la figure 2.17 ; c'est le régime étudié dans cette thèse). On l'obtient en intégrant l'équation 2.48 à partir de la condition initiale  $\{x(t=0) = 0.1, y(t=0) = 0.1\}$ . Pour une autre condition initiale  $\{x(t=0) = 1, y(t=0) = 0.1\}$ , on tombe sur un cycle limite (figure du bas). Une étude détaillée des diagrammes de bifurcation pour l'équation de Duffing est présentée par Parlitz & al. [70].

L'oscillateur de Duffing a été utilisé pour démontrer les limites de la méthode de contrôle *OGY* sur les systèmes obtenus par reconstruction dynamique [26]. Nous discutons de ces particularités au chapitre 4, chapitre dans lequel nous présentons une nouvelle approche de ciblage et de contrôle.

#### 2.4.4 Flot conservatif : DKP

L'étude du mouvement d'un électron sous l'influence d'un champ magnétique et d'un potentiel coulombien peut devenir extrêmement complexe. Si on place un atome d'hydrogène dans un champ magnétique, on observe le phénomène bien connu d'effet Zeeman, soit une séparation des différents niveaux d'énergie. On retrouve ce résultat théorique en ajoutant un terme de perturbation (proportionnel au champ) à l'Hamiltonien classique. Dans le cas d'un champ magnétique homogène et plus intense, cette ap-

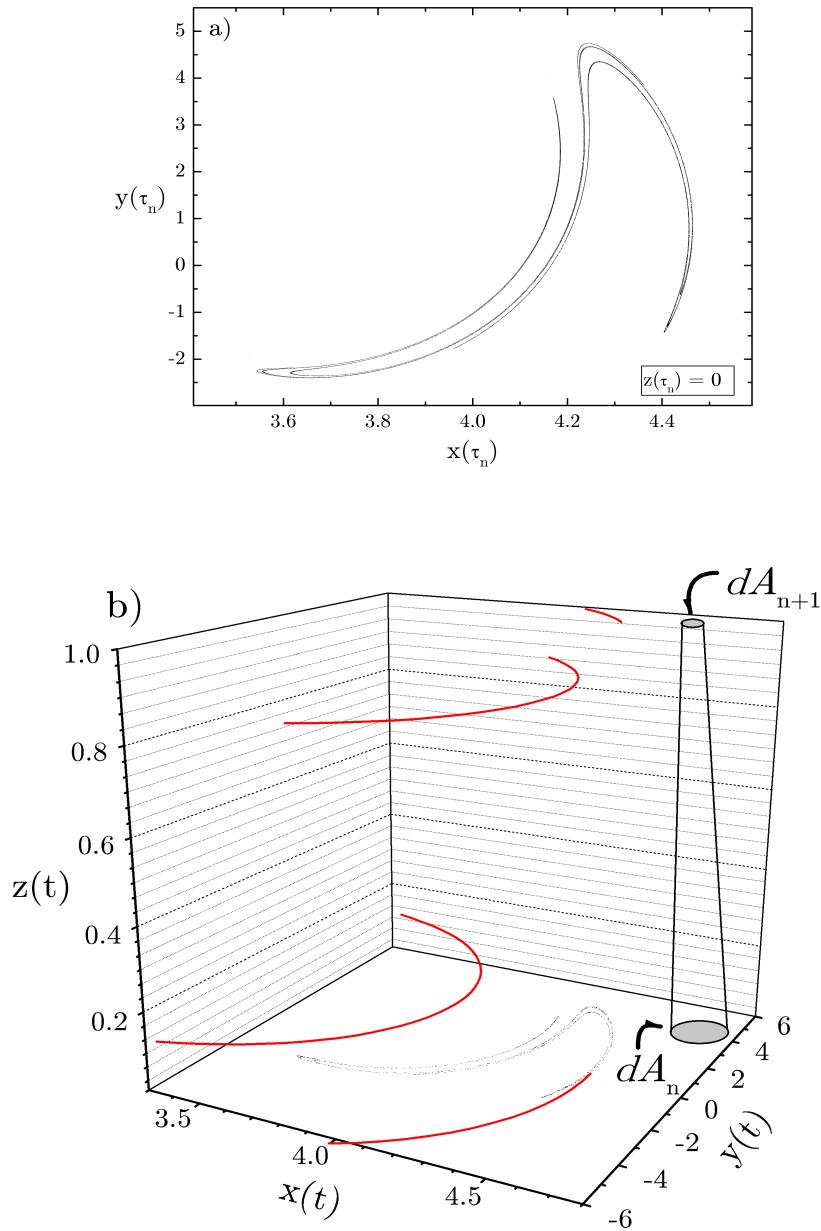


FIG. 2.17 – Section de Poincaré et portions d’une trajectoire pour l’oscillateur de Duffing.

La figure 2.17a) montre la dynamique discrète du flot de Duffing ( $\gamma = 0.2$ ,  $\omega = 0.661$  et  $f = 36.0$ ) sur une Section de Poincaré définie par  $z(\tau_n) = 0$  et  $\frac{dz}{dt} \Big|_{\tau_n} > 0$ . En b), on montre certaines portions d’une seule trajectoire (trait —) qui monte graduellement vers le haut. Lorsqu’elle arrive en  $z = 1$ , la trajectoire est ramenée instantanément à  $z = 0$  où elle reprend sa montée.

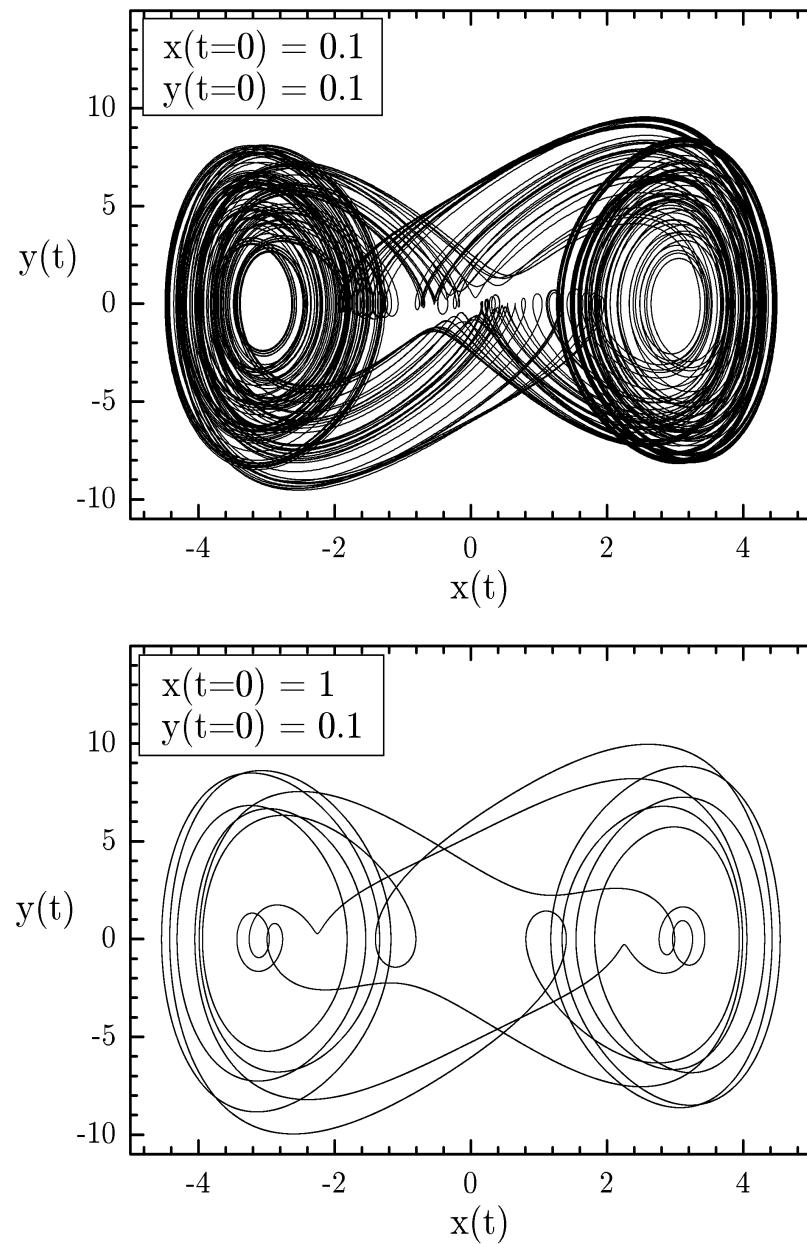


FIG. 2.18 – Trajectoires continues de l’oscillateur de Duffing.

On montre ici les projections de deux trajectoires dans le plan  $XY$  pour l’oscillateur de Duffing ( $\gamma = 0.2$ ,  $\omega = 0.661$  et  $f = 36.0$ ). La figure du haut, qui correspond au régime chaotique de la figure 2.17, est obtenue à partir de la condition initiale  $\{x(t = 0) = 0.1, y(t = 0) = 0.1\}$ . Pour une autre condition initiale  $\{x(t = 0) = 1, y(t = 0) = 0.1\}$ , on obtient un cycle limite (figure du bas)).

proche par perturbation n'est plus valide, ce régime est celui du *problème diamagnétique de Kepler* (Diamagnetic Kepler Problem, DKP) [21], [33], [38].

La dynamique de ce problème s'obtient à l'aide du (pseudo)-Hamiltonien [21], [75] :

$$\hat{h} = \frac{1}{2}(p_\mu^2 + p_\nu^2) - \epsilon(\mu^2 + \nu^2) + \frac{1}{8}\mu^2\nu^2(\mu^2 + \nu^2) \equiv 2, \quad (2.50)$$

dérivé à partir d'un système de coordonnées semi-paraboliques et pour un moment angulaire  $L = 0$ . Le paramètre  $\epsilon$  est relié à l'énergie physique ( $E$ ) du système ( $\epsilon = \gamma^{-2/3}E$ ), où  $\gamma = B/B_c$  donne l'intensité relative du champ magnétique en unités de  $B_c \simeq 2.35 \times 10^5 T$ . Le paramètre  $\epsilon$  dicte ici le comportement dynamique, qui peut passer d'un régime quasi-intégrable à une dynamique complètement chaotique sous une variation de ce paramètre [21], [33].

Ici,  $\hat{h}$  donne la (pseudo)-énergie du système, qui est une constante du mouvement :  $\hat{h} \equiv 2$ . Cette constante impose une contrainte supplémentaire pour la construction de l'application discrète. En général, pour un système hamiltonien, les équations du mouvement sont données par les coordonnées généralisées  $q_i, p_i$  :

$$\begin{aligned} \frac{dq_i}{dt} &= p_i \\ \frac{dp_i}{dt} &= -\frac{\partial V(\mathbf{q})}{\partial q_i} \end{aligned}$$

Pour DKP, on a  $q_1 \equiv \mu$ ,  $q_2 \equiv \nu$  et  $V(\mu, \nu) = \epsilon(\mu^2 + \nu^2) + \frac{1}{8}\mu^2\nu^2(\mu^2 + \nu^2)$  et les équations du mouvement sont :

$$\text{DKP} : \frac{d\tilde{\mathbf{v}}}{dt} = \begin{pmatrix} \frac{d\tilde{v}^1}{dt} \\ \frac{d\tilde{v}^2}{dt} \\ \frac{d\tilde{v}^3}{dt} \\ \frac{d\tilde{v}^4}{dt} \end{pmatrix} \equiv \begin{pmatrix} \frac{d\nu}{dt} \\ \frac{dp_\nu}{dt} \\ \frac{d\mu}{dt} \\ \frac{dp_\mu}{dt} \end{pmatrix} = \begin{pmatrix} p_\nu \\ \nu(2\epsilon - \frac{1}{4}\mu^4 - \frac{1}{2}\nu^2\mu^2) \\ p_\mu \\ \mu(2\epsilon - \frac{1}{4}\nu^4 - \frac{1}{2}\nu^2\mu^2) \end{pmatrix}. \quad (2.51)$$

Il y a  $N = 4$  variables qui décrivent la dynamique continue. La Section de Poincaré permet d'obtenir une relation entre les variables du système discret et d'éliminer ainsi l'une d'entre elles. De plus, ces quatre variables sont reliées par une contrainte sur

la (pseudo) énergie  $\hat{h}$  (équation 2.50), ce qui permet d'éliminer une autre variable. Pour DKP, on choisit ainsi de travailler avec le système discret  $(\nu_n, p_{\nu_n})$ . On présente à l'annexe C les détails pour obtenir le flot discret ainsi que la matrice jacobienne de l'application  $(\nu_n, p_{\nu_n}) \mapsto (\nu_{n+1}, p_{\nu_{n+1}})$ . Il faut noter que pour ce choix de variable et une Section de Poincaré particulière ( $\mu(\tau_n) = \text{constante}$ ), le déterminant de la matrice jacobienne de l'application discrète doit être égal à 1 [38]§7.

On montre à la figure 2.19 les diagrammes des phases de l'application discrète  $(\nu_n, p_{\nu_n}) \mapsto (\nu_{n+1}, p_{\nu_{n+1}})$  pour quelques valeurs de  $\epsilon$ . Le régime que nous utilisons au chapitre suivant ( $\epsilon = -0.1$ ) est présenté à la figure 2.20, d'abord pour la dynamique continue et aussi sur la Section de Poincaré  $\mu(\tau_n) = 0$  ;  $\frac{d\mu}{dt}\Big|_{\tau_n} > 0$ .

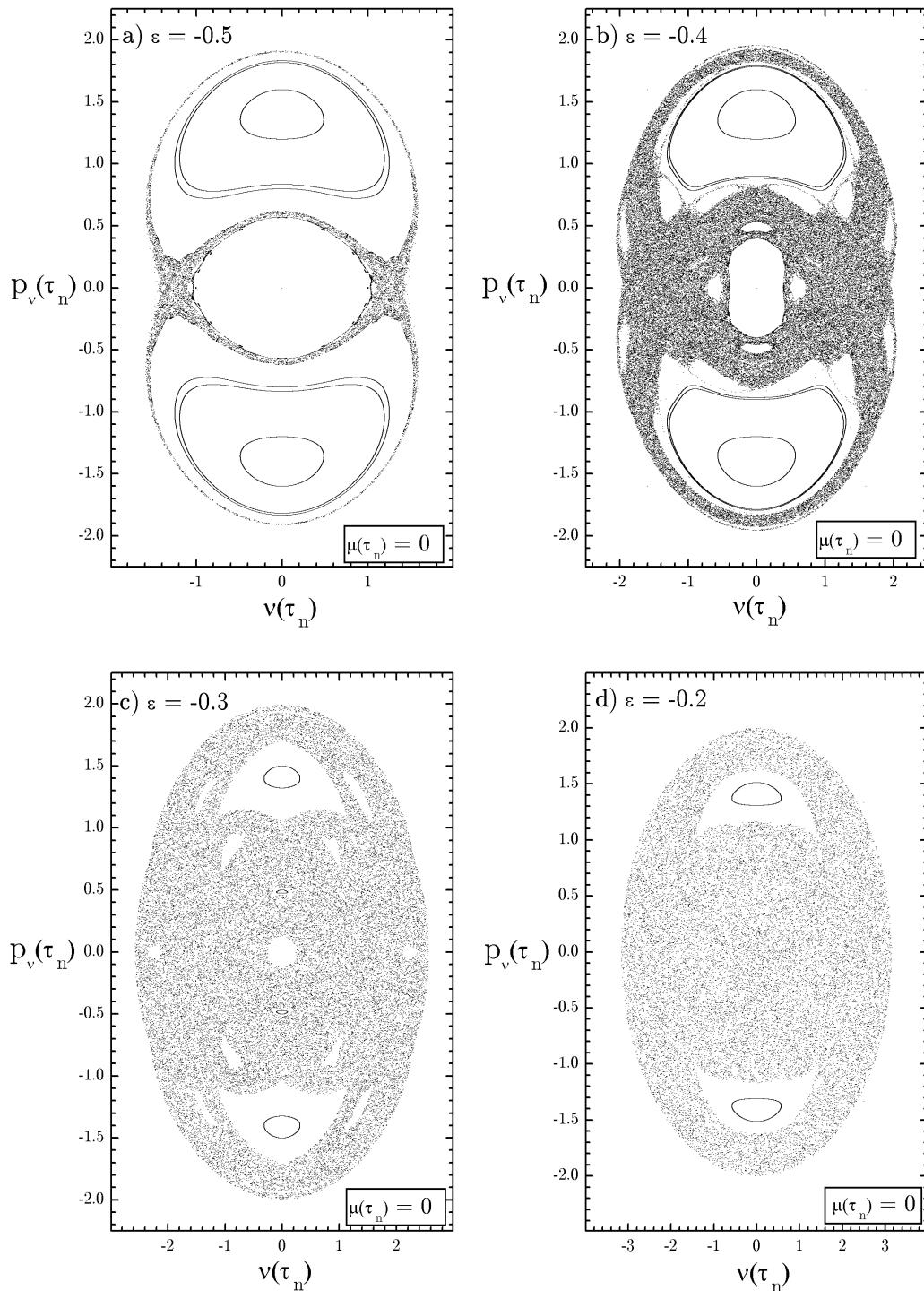


FIG. 2.19 – Diagrammes des phases du flot DKP sur une Section de Poincaré.

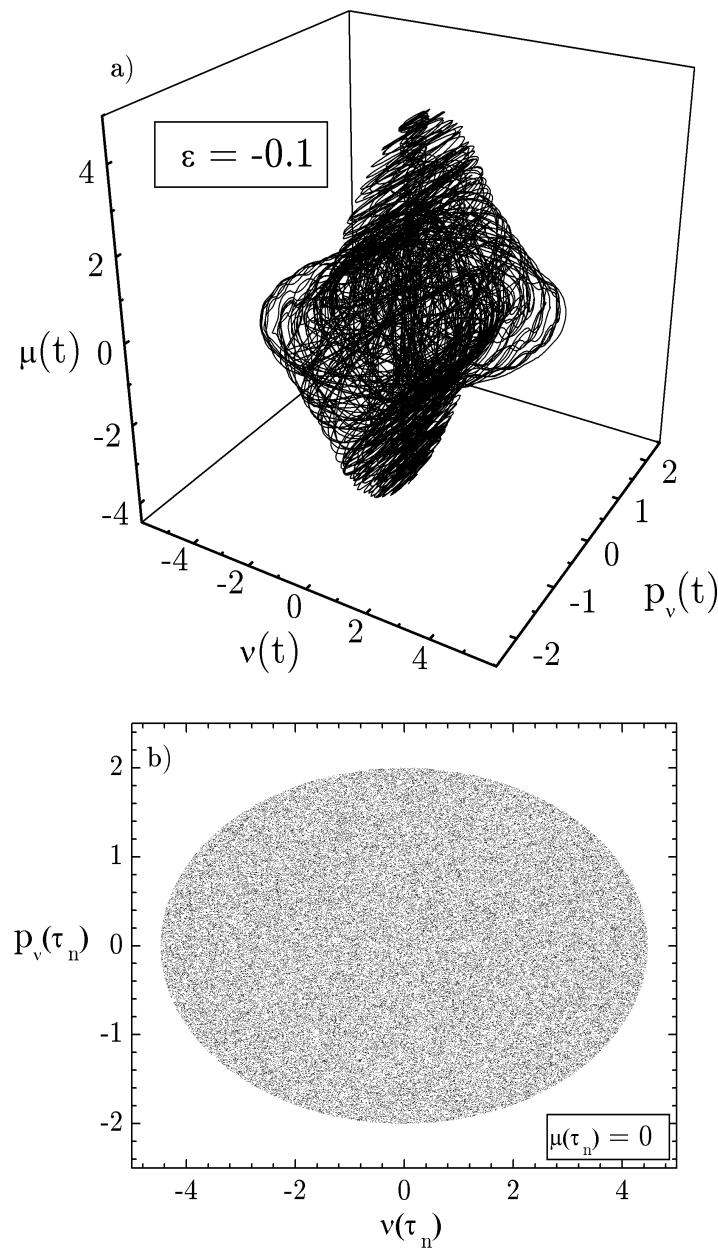


FIG. 2.20 – Dynamique du flot DKP et Section de Poincaré.

La figure 2.20a) illustre la dynamique continue du système conservatif DKP ( $\epsilon = -0.1$ ).

On montre en b) la dynamique correspondante sur la Section de Poincaré définie par  $\mu(\tau_n) = 0$  et  $\frac{d\mu}{dt}\Big|_{\tau_n} > 0$ .

# Chapitre 3

## Tests numériques et résultats

Ce chapitre regroupe l'ensemble des résultats que nous avons obtenus quant à la détection des OPIs de différents systèmes chaotiques. La première section est consacrée à décrire brièvement le code numérique qui a été réalisé tout en mentionnant les aspects plus techniques reliés à l'implémentation des algorithmes *SD* et *DL*. Nous analysons en second lieu la convergence des deux algorithmes sur un système particulier (Système Ikéda). En plus d'aider à comprendre le fonctionnement des algorithmes, cette analyse permet d'exprimer quantitativement les limites de l'algorithme *SD*. Nous montrons ensuite les résultats obtenus pour la recherche d'OPIs dans quelques systèmes chaotiques. Nous terminons par une discussion qui reprend les éléments importants de ce chapitre.

## 3.1 Étapes préliminaires

### 3.1.1 Production du code

Les algorithmes *SD* et *DL* (tableau 1.4) ont été codés en langage C. Le programme qui a été fait comporte deux modules principaux.

Le premier module comprend la liste des fonctions reliées aux modèles chaotiques présentés au chapitre 2. La tâche de ces fonctions est de retourner l'état  $\mathbf{v}_{n+1}$  à partir d'un état  $\mathbf{v}_n$ . Un "pointeur" de fonction est initialisé lors de l'exécution du programme et "pointe" vers la fonction qui décrit l'évolution du modèle chaotique choisi par l'utilisateur (On initialise également le pointeur de la matrice jacobienne si elle est disponible analytiquement). Dans le cas des systèmes continus, la fonction qui est initialisée donne l'évolution du système sur la Section de Poincaré : L'état  $\mathbf{v}_{n+1}$  calculé correspond au retour du flot continu sur la section à partir de l'état  $\mathbf{v}_n$ . Pour ces systèmes, il faut donc également initialiser un "pointeur" de dérivées, pointeur qui retourne les dérivées du système continu (Nous discuterons un peu plus loin de l'intégrateur utilisé pour obtenir l'évolution des flots continus). Si on souhaite détecter les orbites périodiques d'un "nouveau" modèle chaotique, il suffit d'ajouter la fonction correspondante s'il s'agit d'une application discrète ou encore d'ajouter ses dérivées s'il s'agit d'un système continu. On recompile le module et on peut se lancer à l'assaut des OPIs pour le nouveau système !

Le deuxième module s'occupe de faire les différents calculs du tableau 1.4. On appelle les fonctions du premier module au besoin (i.e. lors de l'évaluation des  $\delta\mathbf{w}$ , équation 1.30 ou 1.43). Le critère de précision choisi est :

$$\boxed{\begin{aligned} |\delta\mathbf{w}| &< |\mathbf{w}_n| * PRECISION + PRECISION \\ &\text{ou} \\ |\mathbf{G}| &< 10^{-15}, \end{aligned}} \quad (3.1)$$

avec  $PRECISION = 10^{-14}$ . Le deuxième critère qui est ajouté est surtout utile pour

l'algorithme *DL*. En effet, lors d'une bifurcation, il peut apparaître de nouvelles OPIs. Dans ces cas, certains éléments de la matrice jacobienne  $\mathbf{J}_G$  deviennent près de zéro et les corrections  $\delta\mathbf{w}_{DL}$  peuvent alors devenir inappropriées.

Mentionnons également qu'un ajout doit être fait pour les applications *modulo*  $\mathcal{M}$  telles l'Application Standard ( $\mathcal{M} = 1$ ) ou les différents Billards ( $\mathcal{M} = 2\pi$ ). En effet, ces applications ont la particularité suivante : Une composante  $v^{(d)} = 0$  d'un point de l'espace des phases est identique à  $v^{(d)} = \mathcal{M}$ . Lorsqu'on calcule  $\mathbf{G}$  (équation 1.21), on doit faire la différence entre  $\mathbf{F}^{(m)}(\mathbf{v}_n)$  et  $\mathbf{v}_n$ . Si la composante  $d$  de  $\mathbf{v}$  est près de 0 et que son retour après  $m$  itérations est près de  $\mathcal{M}$ , la différence  $G^{(d)}$  sera de l'ordre de  $\mathcal{M}$ . Pourtant, ces 2 points sont en réalité très près un de l'autre. On ajoute donc la condition suivante avant le calcul du pas  $\delta\mathbf{w}_{SD}$  ou  $\delta\mathbf{w}_{DL}$  :

$$\begin{aligned} \text{Si } G^{(d)} > \mathcal{M}/2 \text{ alors } G_{new}^{(d)} &= G_{old}^{(d)} - \mathcal{M}/2 \\ \text{Si } G^{(d)} < \mathcal{M}/2 \text{ alors } G_{new}^{(d)} &= G_{old}^{(d)} + \mathcal{M}/2. \end{aligned}$$

En plus des algorithmes du tableau 1.4, il faut, pour la recherche des orbites de période- $m$ , s'assurer que l'orbite qui est détectée est bien une orbite de période- $m$ . Rappelons que les algorithmes *SD* et *DL* convergent vers une solution  $\mathbf{v}^*$  de  $\mathbf{G}(\mathbf{v}^*) = 0$  (équation 1.21). Cette solution *peut* être la composante d'une orbite de période- $m$  mais peut aussi être, par exemple, une orbite de période-1. On vérifie que l'orbite trouvée est de période- $m$  en itérant  $\mathbf{v}^*$   $m - 1$  fois. En notant  $\mathbf{v}^*(k)$  le  $k^{ième}$  itéré de  $\mathbf{v}^*$ , on obtient la condition :

$$\begin{aligned} \text{Si } |\mathbf{v}^*(k) - \mathbf{v}^*| &> EPS_{REL}|\mathbf{v}^*| + EPS_{ABS} \text{ pour } \forall k < m, \\ \text{alors la composante trouvée est bien une composante de période-}m. \end{aligned} \tag{3.2}$$

avec  $EPS_{REL}$  et  $EPS_{ABS}$  de l'ordre de  $10^{-8}$ . Finalement, on doit aussi s'assurer que l'orbite trouvée n'a pas encore été détectée. Pour ce faire, on pose une autre condition pour accepter  $\mathbf{v}^*$ . En notant les  $\mathbf{v}_i^*$  les solutions qui ont déjà été repérées et acceptées

comme orbite de période- $m$ , on obtient la condition :

Si  $|\mathbf{v}_i^*(k) - \mathbf{v}^*| > EPS_{REL}|\mathbf{v}^*| + EPS_{ABS}$  pour  $\forall k \leq m$  et  $\forall i$ ,  
alors la composante trouvée est une nouvelle orbite.

(3.3)

### 3.1.2 Exemples numériques pour les flots continus

Nous incluons ici une courte section afin de donner quelques exemples numériques pour l'évolution des flots sur la Section de Poincaré. L'initiation au “monde” du numérique demande une bonne dose de patience et nous pensons que ces quelques exemples pourront faciliter l'apprentissage des personnes intéressées à ce domaine. L'intégrateur que nous utilisons pour les flots de Lorenz et DKP est bien décrit dans l'article de Meyer [60]. Le principe de ce schème d'intégration est très astucieux : On développe le flot  $\tilde{\mathbf{v}}(t)$  en série de Taylor autour d'une condition initiale  $\tilde{\mathbf{v}}(t = 0) = \tilde{\mathbf{v}}_0$ . On cherche à avancer cette condition  $\tilde{\mathbf{v}}_0$  d'un pas de temps  $h$  :

$$\tilde{\mathbf{v}}(t = h) = \tilde{\mathbf{v}}_0 + \frac{d\tilde{\mathbf{v}}}{dt}\Big|_{\tilde{\mathbf{v}}_0} h + \frac{1}{2} \frac{d^2\tilde{\mathbf{v}}}{dt^2}\Big|_{\tilde{\mathbf{v}}_0} h^2 + \dots$$

Le terme  $\frac{d\tilde{\mathbf{v}}}{dt}\Big|_{\tilde{\mathbf{v}}_0}$  est la fonction  $\tilde{\mathbf{f}}$  évaluée à  $\tilde{\mathbf{v}}_0$  ; ce sont les dérivées du système par rapport au temps (équation 2.31). Le prochain terme dans le développement de Taylor est la dérivée de  $\tilde{\mathbf{f}}$  par rapport au temps. Si  $\tilde{\mathbf{f}}$  a la forme d'un polynôme, on peut calculer cette dérivée facilement à l'aide de  $\tilde{\mathbf{v}}_0$  et  $\tilde{\mathbf{f}}(\tilde{\mathbf{v}}_0)$ . Les prochains termes s'obtiennent aussi par récurrence et on peut ainsi aller à l'ordre  $\mathcal{N}$ . Pour une précision  $\epsilon$  donnée, la performance de l'intégrateur est maximale pour  $\mathcal{N} \sim 2 - \log_{10} \epsilon$ . Le pas  $h$  effectué pour la précision demandée est  $h = (\epsilon \mathcal{N}!)^{1/\mathcal{N}} / z$  avec  $z = \max\{\frac{d^n\tilde{\mathbf{v}}}{dt^n}\Big|_{\tilde{\mathbf{v}}_0}\}$ ,  $n = \{\mathcal{N} - 2, \mathcal{N} - 1, \mathcal{N}\}$ . Pour l'intégrateur, nous choisissons une précision de  $\epsilon = 10^{-15}$ . Pour le système de Duffing, nous utilisons un intégrateur Runge-Kutta (coefficients Cash-Karp) à pas adaptatif [76]§16.2.

Système	Section de Poincaré $h(\tilde{\mathbf{v}}) = 0$	Condition initiale $\mathbf{v}_n$	Variable(s) éliminée(s) $\check{\mathbf{v}}_n$	Retour sur la Section $\mathbf{v}_{n+1}$	Matrice jacobienne $\mathbf{J}(\mathbf{v}_n)$
Système de Lorenz $\sigma = 16$ $b = 4$ $R = 45.92$	$z_n = R - 1$	$x_n = 15.0$ $y_n = 15.0$	$z_n = 44.92$	$x_{n+1} = 14.4878217965$ $y_{n+1} = 15.5200302938$	$J_{11} = 0.1815106250$ $J_{12} = 0.4946611603$ $J_{21} = 0.3581198156$ $J_{22} = 0.9778462507$
	$x_n + z_n = 40$	$x_n = 10.0$ $y_n = 20.0$	$z_n = 30.0$	$x_{n+1} = 14.6499599099$ $y_{n+1} = 26.1435201070$	$J_{11} = 0.9476067482$ $J_{12} = 0.7609816007$ $J_{21} = 2.3257174993$ $J_{22} = 1.8676872339$
Oscillateur de Duffing $\gamma = 0.2$ $\omega = 0.661$ $f = 36.0$	$z_n = 0$	$x_n = 4.0$ $y_n = 1.0$	$z_n = 0.0$	$x_{n+1} = 3.96185262375$ $y_{n+1} = -1.57734377023$	$J_{11} = -0.4575139316$ $J_{12} = 0.2452288160$ $J_{21} = -2.9543910117$ $J_{22} = 1.2570106335$
DKP $\epsilon = -0.1$	$\mu_n = 0$	$\nu_n = -0.8$ $p_{\nu_n} = -1.6$	$\mu_n = 0.0$ $p_{\mu_n} = 1.145426$	$\nu_{n+1} = 1.76839422438$ $p_{\nu_{n+1}} = 0.204166371694$	$J_{11} = -7.6103249267$ $J_{12} = -23.742730341$ $J_{21} = -9.2459160324$ $J_{22} = -28.976856210$

TAB. 3.1 – Quelques exemples numériques pour les flots continus.

On indique dans ce tableau le retour  $\mathbf{v}_{n+1}(\mathbf{v}_n)$  sur une Section de Poincaré pour quelques systèmes chaotiques. La matrice jacobienne sur la section est aussi donnée. Pour l'oscillateur de Duffing, on note que le déterminant de cette matrice est bien égal à  $e^{-2\pi\gamma/\omega}$  (équation 2.49). Pour *DKP*, le déterminant de  $\mathbf{J}$  est égal à 1.0 à une précision de  $10^{-13}$ .

### 3.1.3 Premier test : Détection d’OPIs chez Hénon

Afin de vérifier le bon fonctionnement des algorithmes, nous avons lancé une recherche d’OPIs sur un système où leur nombre est connu, soit le système de Hénon. Rappelons brièvement les quelques paramètres qui doivent être choisis pour l’utilisation de l’un ou l’autre des algorithmes :

- Choix de  $\lambda$  (Schmelcher-Diakonos) et  $\beta$  (Davichack-Lai)
- Choix des conditions initiales  $\mathbf{w}_0$  (DL suggèrent d’utiliser les composantes des orbites de période- $(m - 1)$  pour la recherche des OPIs de période- $m$ ).
- Choix du nombre maximal d’itérations  $N_{max}$  (DL suggèrent  $\sim 4 - 6$  fois  $\beta$ ).

Nous reviendrons régulièrement sur ces 3 points car ils peuvent influencer l’efficacité des algorithmes.

Comme premier test, nous recherchons les OPIs du système de Hénon jusqu’à la période-20 (ces OPIs ont été publiées dans [8]). Pour la détection, nous utilisons les algorithmes comme des “boîtes noires” en suivant directement les recommandations des auteurs. Pour *SD*, les conditions initiales ( $\sim 5000$ ) sont choisies aléatoirement sur l’attracteur. Le paramètre  $\lambda$  est diminué progressivement entre  $10^{-1}$  et  $10^{-5}$  selon la période recherchée. On choisit  $N_{max} \sim 50/\lambda$ . Pour l’algorithme *DL*, les conditions initiales sont les composantes des orbites de période  $m - 1$  pour  $m > 6$ . Pour  $m \leq 6$ , les conditions initiales sont choisies aléatoirement sur l’attracteur. *Les deux algorithmes parviennent à détecter l’ensemble complet d’OPIs dans Hénon.*

## 3.2 Dans le “coeur” des algorithmes

Bien que les premiers résultats sur Hénon soient concluants, nous voulons pousser l’étude un peu plus loin. Nous reprenons donc la recherche d’OPIs mais cette fois, sur

le système Ikéda. À l'origine, ce système fut utilisé par *SD* et *DL* pour illustrer le bon fonctionnement de leur algorithme. Nous reproduisons, dans un premier temps, les résultats de *SD* et *DL*. Nous faisons ensuite ressortir quelques différences entre les deux algorithmes.

### 3.2.1 Visualisation du fonctionnement des algorithmes

Avant d'aborder la détection des OPIs dans Ikéda, nous voulons donner une “image” du fonctionnement des deux méthodes. Cette image est utile pour bien comprendre ce que nous présentons un peu plus loin. La figure 3.1 montre comment évolue une condition initiale dans les systèmes : a) Ikéda libre, b) Ikéda sous itération *SD* (période-1) et c) Ikéda sous itération *DL* (période-1). Nous avons choisi une condition initiale aléatoirement dans l'espace des phases de Ikéda. En itérant cette condition initiale dans le système “libre”, on obtient évidemment l'attracteur présenté au chapitre 2 : On ne peut converger vers le point fixe car il est instable. On reprend ensuite cette même condition initiale, mais cette fois, on l'itère avec l'algorithme *SD* en demandant une période-1 (avec la matrice  $C_1, \lambda = 0.1$ ). La dynamique est ici totalement différente. Le système évolue graduellement vers le point fixe d'Ikéda. Enfin, la figure 3.1c montre la dynamique du système avec l'algorithme *DL*.

La figure suivante illustre l'évolution de quelques conditions initiales sous l'influence des algorithmes *SD* (fig. 3.2a) et *DL* (fig. 3.2b) mais cette fois, sur le système de Hénon. On recherche dans ce scénario des OPIs de période-8 en utilisant les composantes des OPIs de période-7 comme conditions initiales. La matrice choisie est  $C_1$ . L'algorithme *SD* n'arrive pas à retrouver toutes les OPIs de Hénon avec seulement cette matrice. Il faut faire appel aux autres matrices pour retrouver les 3 OPIs manquantes (La détection de ces 3 OPIs n'est pas illustrée sur la figure). L'algorithme *DL* repère les 7 OPIs de période-8 avec seulement la matrice  $C_1$ .

On remarque sur cette figure, tout comme la précédente, que les pas  $DL$  sont beaucoup plus grands que ceux de  $SD$ , et ce, même *loin* des OPIs. Pour quantifier d'avantage cette différence, nous avons tracé sur la figure 3.3 la grandeur moyenne des pas  $SD$  et  $DL$  (respectivement  $|\delta\mathbf{w}_{SD}|$  et  $|\delta\mathbf{w}_{DL}|$ ) pour les trajectoires qui convergent vers le point fixe d'Ikéda en fonction du résidu  $g$ . Rappelons que le résidu absolu  $g$  est défini par  $|\mathbf{G}|$  (réf. équation 1.43) et que  $\mathbf{G} \equiv \mathbf{F}^{(m)}(\mathbf{x}_n) - \mathbf{x}_n$ . Le module de cette fonction donne une indication de la distance à laquelle on se trouve de l'OPI. Si  $g = 0$ , on est directement sur l'OPI. Ce graphique donne ainsi une idée de la grandeur des corrections en fonction de la distance entre l'état du système et la position d'une OPI. Dans le cas  $SD$  et pour  $C_k = C_1$ , les corrections sont  $|\delta\mathbf{w}_{SD}| = \lambda g$ ; sur une échelle log – log, la pente est de 1 et l'ordonnée à l'origine correspond à  $\log \lambda$ . Pour  $DL$ , on ne peut établir une relation aussi simple. On remarque cependant que loin de l'OPI, la grandeur des pas est relativement constante (plateau horizontal). Les pas  $DL$  sont donnés par l'équation 1.43, que l'on réécrit ici :

$$\delta\mathbf{w}_n^{DL} = [\mathbf{1}\beta g(\mathbf{w}_n) - \mathbf{C}_k \mathbf{J}_G(\mathbf{w}_n)]^{-1} \mathbf{C}_k \mathbf{G}(\mathbf{w}_n).$$

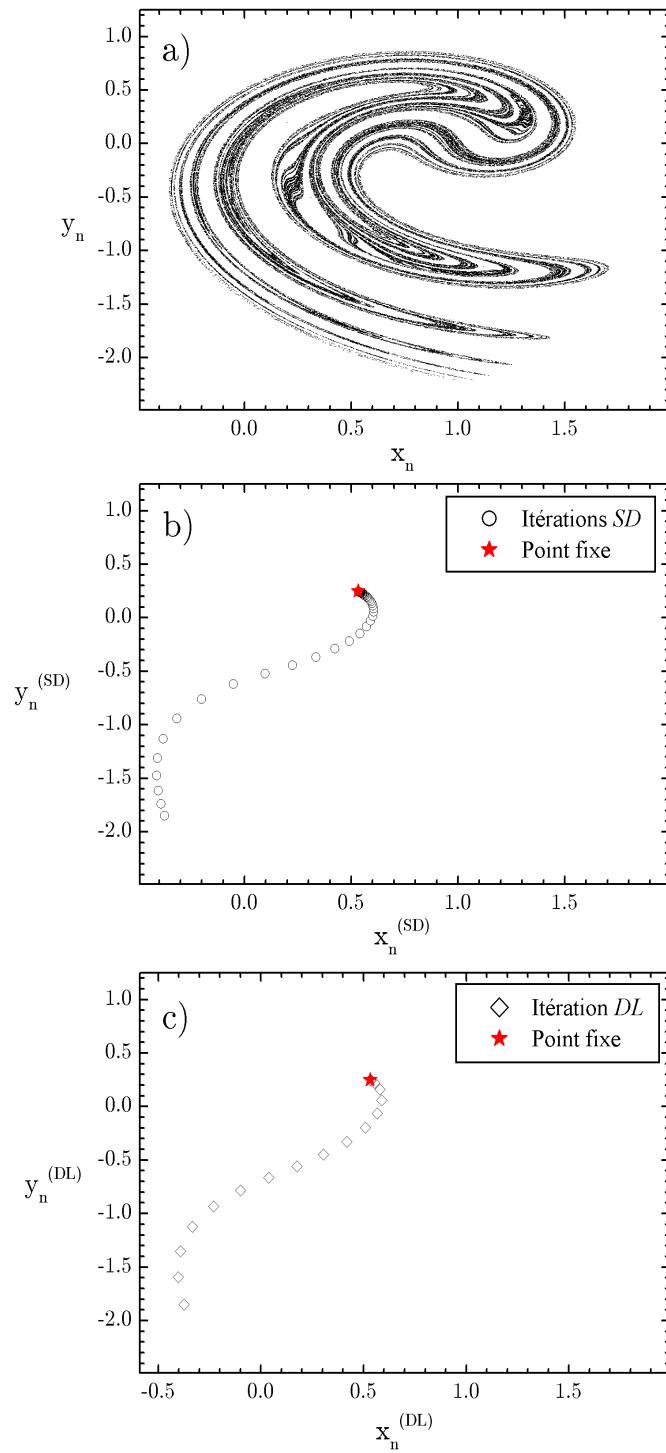
Loin de l'OPI, le terme  $g(\mathbf{w}_n)$  au dénominateur est plus imposant que  $\mathbf{J}_G$  (du moins pour ce point fixe) et la grandeur des pas est approximativement constante :

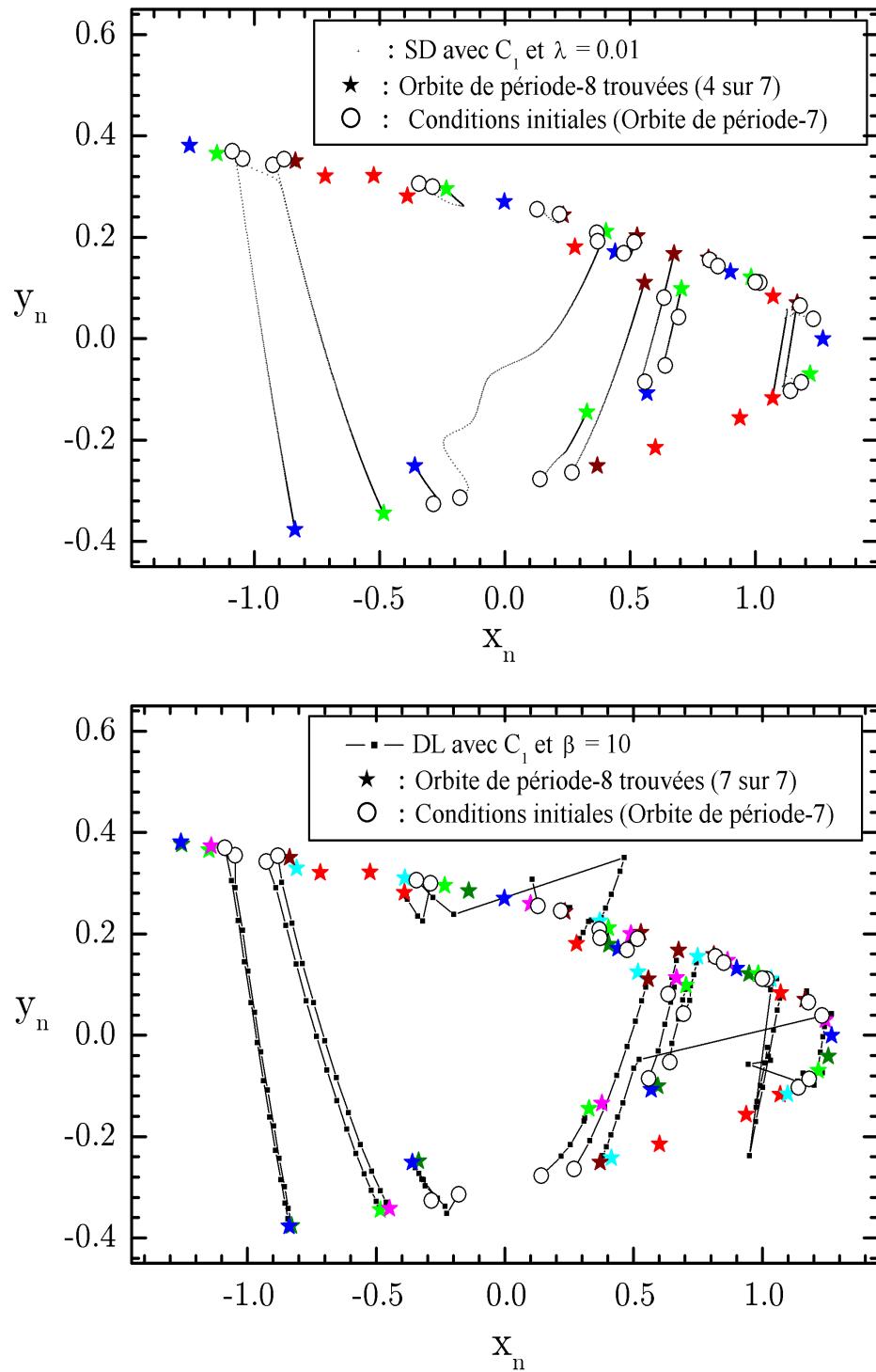
$$\begin{aligned} |\delta\mathbf{w}_n^{DL}| &\sim \left| \frac{\mathbf{C}_k \mathbf{G}(\mathbf{w}_n)}{\beta g(\mathbf{w}_n)} \right| \\ &\sim \frac{1}{\beta} = \text{constante} \end{aligned}$$

Lorsque  $\beta g$  devient moins imposant, la “partie” Newton-Raphson de l'algorithme  $SD$  se met en opération. On observe alors une “cassure” dans le plateau.

### 3.2.2 Ikéda comme banc d'essais

Nous allons maintenant comparer le rendement des deux algorithmes sur le système Ikéda. Le rendement qui nous intéresse ici comprend deux aspects : Le premier concerne le nombre total d'OPIs détectées et le second, le nombre d'itérations nécessaires pour

FIG. 3.1 – Dynamique des algorithmes de *SD* et *DL* sur le système d’Ikéda.

FIG. 3.2 – Dynamique des algorithmes de *SD* et *DL* sur l'attracteur de Hénon.

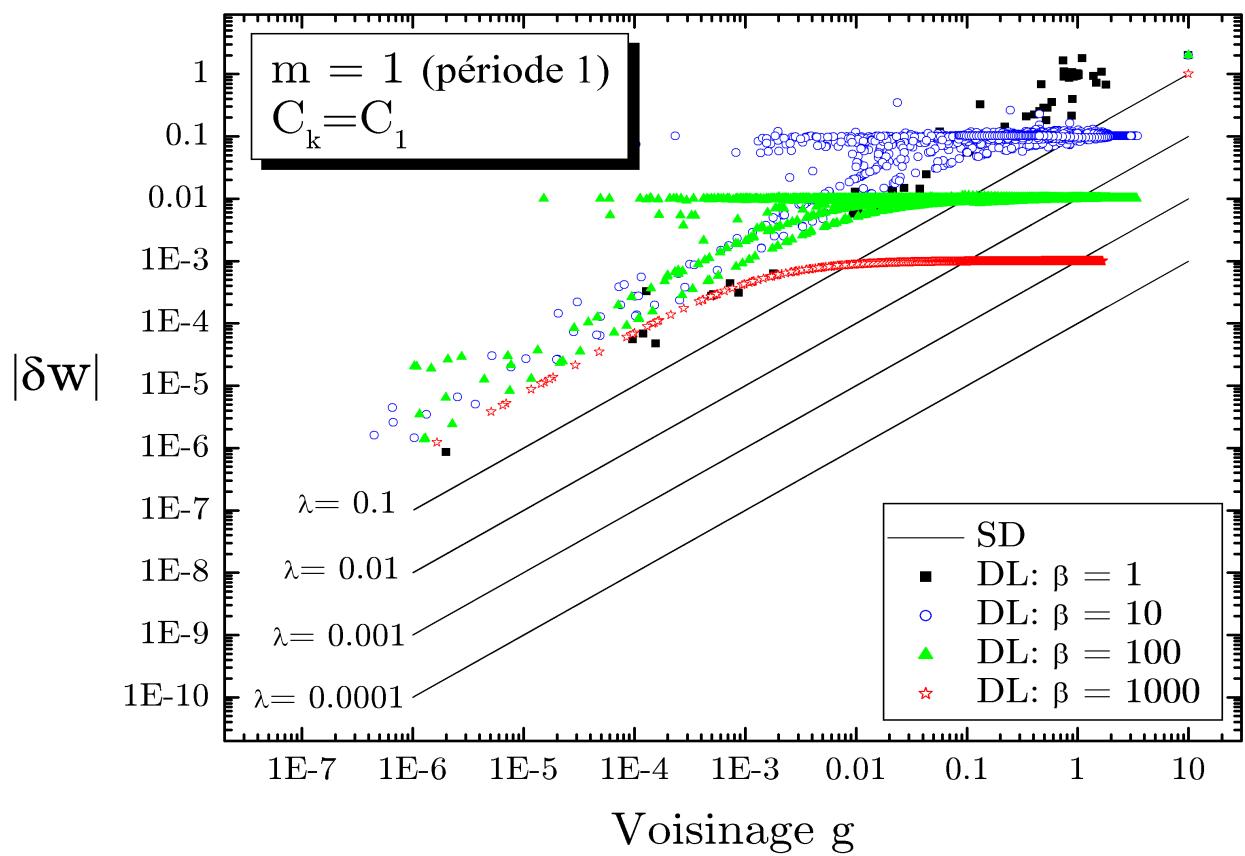
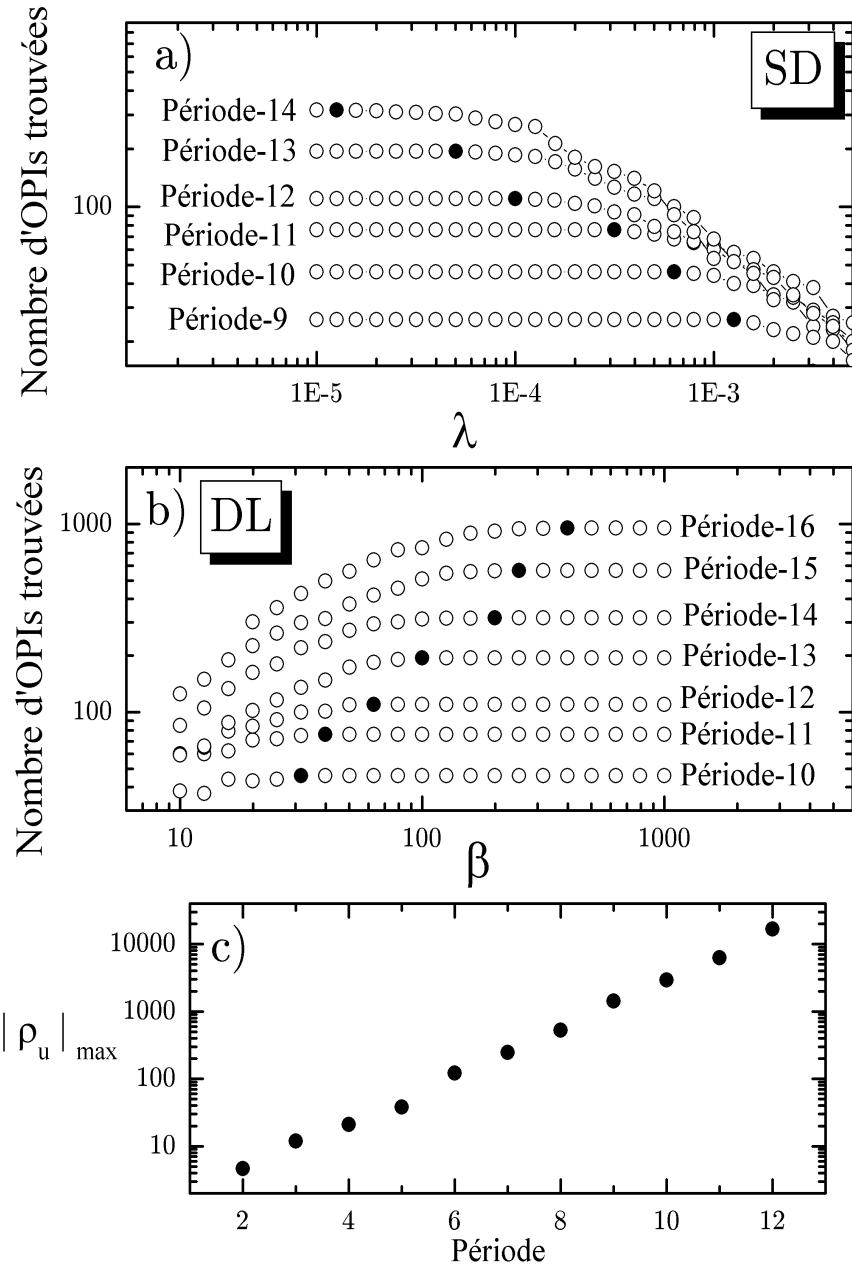


FIG. 3.3 – Grandeur des pas  $\delta w_{SD}$  et  $\delta w_{DL}$  en fonction du voisinage  $g$  (Système Ikéda).

y arriver. Les OPIs pour les périodes 1 à 2 sont détectées en choisissant des conditions initiales aléatoirement sur Ikéda. Nous utilisons ensuite, pour les deux algorithmes, les composantes des OPIs de période- $(m - 1)$  comme conditions initiales. Pour chaque période recherchée, on essaie plusieurs valeurs de  $\lambda$  (resp.  $\beta$ ). Le paramètre  $N_{max}$  est fixé à  $50/\lambda$  pour  $SD$  et  $10\beta$  pour  $DL$ . La figure 3.4 montre le nombre d’OPIs détectées en fonction des paramètres  $\lambda$  (resp.  $\beta$ ). Nous indiquons par un point noir la valeur de  $\lambda_c$  critique (resp.  $\beta_c$ ) à partir de laquelle le nombre total d’OPIs trouvées ne change plus. Ceci *pourrait* être pris comme un premier critère pour affirmer que l’ensemble des OPIs est complet, bien que ce ne soit pas un critère très rigoureux. Nous insistons ici sur le “pourrait” car nous verrons plus loin que ceci est plutôt une particularité du système Ikéda. On remarque aussi que la position de  $\lambda_c$  (resp.  $\beta_c$ ) semble suivre une relation exponentielle de la forme  $\lambda_c \propto e^{-\Lambda m}$  (resp.  $\beta_c \propto e^{\alpha m}$ ). Pour  $SD$ , cette dépendance exponentielle s’explique assez bien puisque la valeur de  $\lambda_c$  est directement reliée à la valeur propre de la matrice jacobienne de l’OPI la plus instable (réf. équation 1.37). On donne à la figure 3.4c le module de cette valeur propre pour chaque période d’Ikéda ( $|\rho_u|_{max}$ ). Ce module dépend aussi de la période de façon exponentielle. Pour  $DL$ , il semble difficile d’établir une relation autre qu’empirique entre  $\beta_c$  et la période.

Le graphique suivant donne le nombre d’itérations fait par l’algorithme  $SD$  par rapport à  $DL$  pour chaque période recherchée. Pour la détection complète des périodes- $m$ , un certain nombre de conditions initiales devaient être itérées (soit les composantes des périodes  $m - 1$ ). Pour itérer toutes ces conditions initiales jusqu’à convergence ou jusqu’à ce que le critère d’arrêt  $N_{max}$  soit atteint, il faut faire  $N_{SD}$  (resp.  $N_{DL}$ ) itérations. Le graphique montre donc le rapport  $N_{SD}/N_{DL}$  en fonction de la période et il a été fait pour  $\lambda = \lambda_c$  (resp.  $\beta = \beta_c$ ). On remarque que plus la période est élevée, plus l’algorithme  $DL$  devient performant par rapport à  $SD$ .

La comparaison est ici légèrement biaisée puisqu’une itération  $SD$  ne demande pas le même “effort” qu’une itération  $DL$ . Nous avons vu aux équations 1.39 et 1.46 qu’un certain nombre d’opérations doivent être effectuées pour accomplir un “pas”  $SD$

FIG. 3.4 – Nombre d’OPIs trouvées en fonction de  $\lambda$  et  $\beta$ .

Cette figure donne le nombre d’OPIs détectées par *SD* et *DL* en fonction de  $\lambda$  et  $\beta$  pour le système Ikéda. La figure 3.4c donne le module maximal des valeurs propres de la matrice jacobienne pour chaque période.

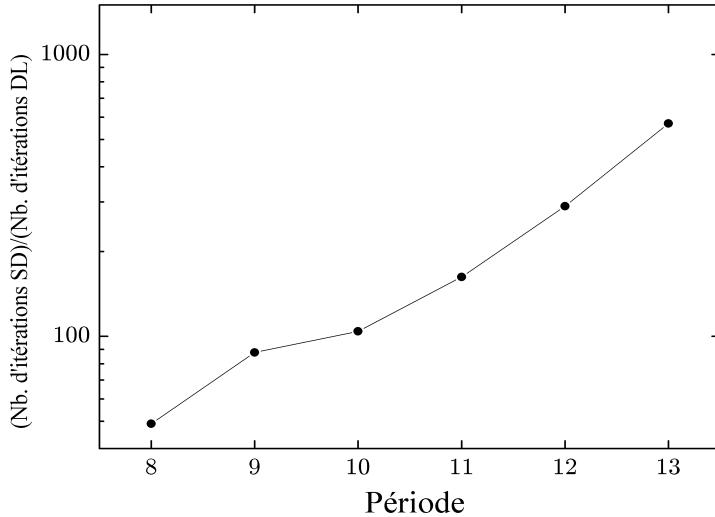


FIG. 3.5 – Nombre d’itérations  $SD$  par rapport  $DL$  en fonction de la période recherchée (Système Ikéda).

ou  $DL$ . Ce qui limite ici la rapidité des algorithmes est principalement l’évaluation de  $\mathbf{G}$  (et l’évaluation de  $\mathbf{J}_G$  dans le cas de  $DL$ ). En effet, la fonction  $\mathbf{G}$  est égale à  $\mathbf{F}^{(m)}(\mathbf{w}_n) - \mathbf{w}_n$ . Il faut donc effectuer  $m$  itérations de la fonction qui décrit l’évolution du système chaotique ( $\mathbf{F}$ ) pour une évaluation de  $\mathbf{G}$ . Avec  $m$  qui augmente, cette évaluation devient beaucoup plus imposante que les quelques opérations d’additions et de multiplications des matrices  $\mathbf{C}_k$ . La seule différence importante qui existe entre un pas  $SD$  et un pas  $DL$  est l’évaluation de la matrice jacobienne  $\mathbf{J}_G$  (qui doit être calculée dans le cas de l’algorithme  $DL$ ). Pour un système discret comme l’application d’Ikéda, cette évaluation est très peu coûteuse (soit environ du même ordre de grandeur que celle de  $\mathbf{G}$ ). Même en tenant compte de ces considérations, l’algorithme  $DL$  demeure toujours plus performant par au moins un ordre de grandeur.

Il faut cependant penser que pour les flots continus, l’évaluation de la matrice jacobienne peut devenir limitante puisqu’on doit intégrer les équations variationnelles. Pour un systèmes continus à  $N$  variables, ces dernières amènent  $N^2$  nouvelles équations. Pour évaluer  $\mathbf{G}$  et  $\mathbf{J}_G$ , il faut alors intégrer  $N + N^2$  équations pour  $m$  retour sur la Section de Poincaré. Avec ces considérations, on déduit que pour les flots, un pas  $DL$  correspond

à environ  $(N + 1)$  pas  $SD$ , ce qui peut devenir non négligeable selon la période- $m$  recherchée. Ceci nous motive à pousser l'analyse un peu plus loin afin d'amener certaines améliorations à la méthode  $SD$ .

### 3.2.3 Optimisation de l'algorithme $SD$

Nous avons vu à la section précédente que l'algorithme  $SD$  souffre d'un taux de convergence relativement lent. Pour des périodes élevées, ceci le rend moins performant que l'algorithme  $DL$ . Nous avons jusqu'ici utilisé l'algorithme  $SD$  comme une boîte noire. Nous allons maintenant essayer de le décortiquer davantage afin d'en tirer profit au maximum. Pour ce faire, nous mettons de côté le système Ikéda et revenons à l'application logistique. Ce système étant à une dimension, il est légèrement plus simple à analyser et permettra d'exposer nos idées plus facilement. Nous reviendrons ensuite aux systèmes à plusieurs dimensions et traiterons du cas d'Ikéda.

Nous avons mentionné à la section 1.3.2.3 que l'algorithme  $SD$  converge linéairement. En une dimension, on peut estimer le nombre d'itérations nécessaires pour atteindre un résidu  $g$ . Avec l'algorithme  $SD$ , les corrections de  $w_n$  sont, en une dimension,  $\delta w_n = \pm \lambda G(w_n)$ . Le signe  $\pm$  provient des deux “matrices”  $C_k$  soit  $C_1 = 1$  et  $C_2 = -1$ . On traite ici le cas de  $C_1, C_2$  étant pratiquement identique. Après une itération,  $G(w_{n+1})$  est donné par :

$$\begin{aligned} G(w_{n+1}) &= G(w_n + \delta w_n) \\ &\sim G(w_n) + \frac{dG}{dw}\Big|_{w_n} \delta w_n \\ &\sim G(w_n) + \frac{dG}{dw}\Big|_{w_n} \lambda G(w_n) \\ &\sim (1 + \lambda \frac{dG}{dw}\Big|_{w_n}) G(w_n). \end{aligned} \tag{3.4}$$

La dérivée  $\frac{dG}{dw}$  a déjà été rencontrée au chapitre 1 (équation 1.27) et nous l'avions notée par  $J_G$  (en plusieurs dimensions, il s'agit de la matrice jacobienne  $\mathbf{J}_G$ ). Le résidu

$g_n = |G(w_n)|$  après  $N_{SD}$  itérations est alors donné par :

$$g_{n+N_{SD}} \sim g_n \left| 1 + \lambda J_G|_{OPI^{(-)}} \right|^{N_{SD}}, \quad (3.5)$$

où nous avons simplifié légèrement en supposant que  $J$  sur l'OPI $^{(-)}$  est le même que sur les  $w_n$ , ce qui est plausible car le voisinage est petit. Nous avons placé l'exposant “ $(-)$ ” après  $OPI$  pour indiquer que seul les OPIs à Jacobien négatif seront stabilisées (il faut utiliser la matrice  $C_2$  pour les OPIs à Jacobien positif). Le terme  $(1 + \lambda J_G)$  correspond simplement au Jacobien dans le système transformé (chapitre 1, équation 1.31). Sous la transformation  $SD$ , les orbites périodiques sont stables : À chaque itération, le système en une dimension se rapproche d'un facteur équivalent au Jacobien  $|1 + \lambda J_G|$ . En moyenne, on doit donc s'attendre que le nombre d'itérations pour atteindre un résidu  $g = |G|$  donné, varie comme :

$$1D : N_{SD} \sim \frac{\log(g)}{\log \left| 1 + \lambda J_G|_{OPI^{(-)}} \right|}.$$

(3.6)

On montre à la figure 3.6a le nombre moyen d'itérations qui ont été faites pour atteindre un résidu  $g$  lors de la recherche d'un point fixe de l'application logistique avec  $R = 4$  (Cette valeur de  $R$  fait en sorte que  $J_G|_{OPI^{(-)}} = -2^m - 1$ ). Sur une échelle semi-log, la pente du graphique doit être égale à  $|1 + \lambda J_G|_{OPI^{(-)}}|^{-1}$  : elle varie ainsi avec  $\lambda$ . C'est ce que nous observons en pratique. Sur cette même figure, nous indiquons dans l'encadré les valeurs mesurées des pentes, valeurs qui sont pratiquement identiques à celles de référence (données entre parenthèses). Les approximations que nous avons faites sont donc relativement justes.

À ce point, il est possible d'aborder un des problèmes que nous avons énoncés au début de la section concernant le nombre d'itérations maximales “ $N_{max}$ ” qui devraient être essayées avant de passer à une autre condition initiale. L'équation 3.6 donne le nombre d'itérations, en moyenne, pour atteindre un résidu  $g$ . Si on veut  $g = PRECISION$ , on doit alors poser :

$$1D : N_{SD_{max}} = \frac{\log(PRECISION)}{\log \left| 1 + \lambda J_G|_{OPI^{(-)}} \right|}.$$

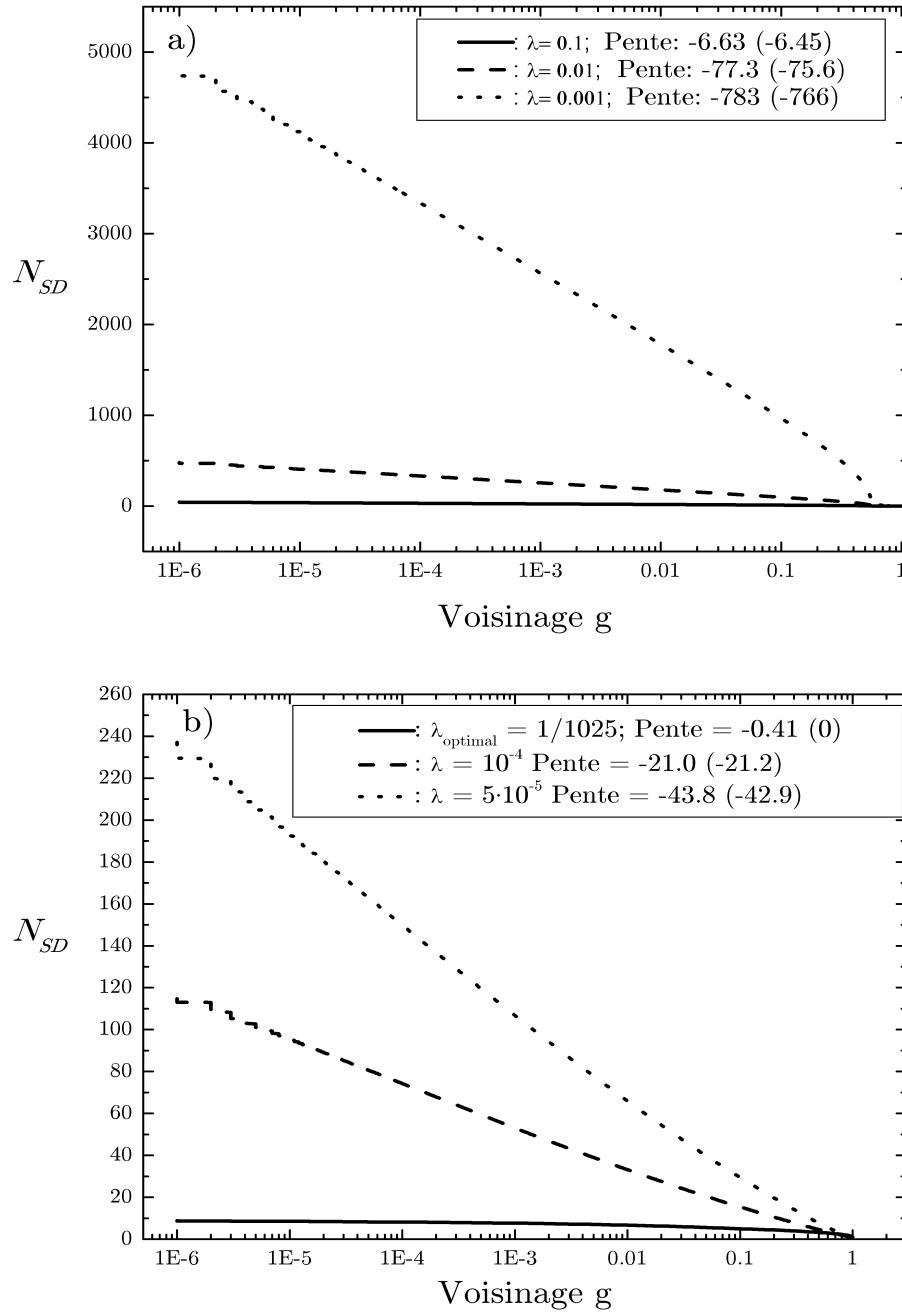


FIG. 3.6 – Nombre moyen de pas  $SD$  pour atteindre un résidu  $g \equiv |\mathbf{G}(\mathbf{w})|$ .

Ces 2 figures montrent le nombre de pas  $SD$  pour atteindre un résidu  $g$  avec différentes valeurs de  $\lambda$  ( $C_k = C_1$ , figure 3.6a :  $m = 1$ , figure 3.6b :  $m = 10$ ). Les valeurs mesurées des pentes sont indiquées en fonction de  $\lambda$ . Les valeurs entre parenthèses sont les valeurs de pentes données par l'équation 3.6.

On ouvre ici une petite parenthèse pour faire remarquer qu'en 1D, on peut minimiser le nombre d'itérations pour atteindre la précision demandée en choisissant une valeur de  $\lambda$  telle que  $1 + \lambda J_G|_{OPI(-)} = 0$ . Par exemple, pour l'application logistique avec  $R = 4$  et  $J_G|_{OPI(-)} = -2^m - 1$ , on prend :

$$\begin{aligned}\lambda_{optimal} &\sim \frac{-1}{J_G|_{OPI(-)}} \\ &\sim \frac{1}{2^m + 1}.\end{aligned}$$

Avec ce choix, on devrait converger rapidement (i.e. de façon quadratique) vers les OPIs à Jacobien négatif. On montre à la figure 3.6b le nombre moyen d'itérations nécessaires pour atteindre un résidu  $g$  pour une période  $m = 10$ , et ce, pour plusieurs valeurs de  $\lambda$ . Sur l'échelle semi-log, la pente du graphique devient presque égale à 0 pour  $\lambda = \lambda_{optimal}$ , ce qui indique que très peu d'itérations sont nécessaires pour atteindre la précision demandée ( $g = 10^{-14}$ ). En pratique, on ne connaît pas  $J_G$  et c'est plutôt ce que l'on cherche ! Il est donc difficile de choisir  $\lambda$  de façon optimale. Aussi, comme en général  $J_G$  varie d'une OPI à l'autre, il y a plusieurs  $\lambda_{optimal}$  (l'application logistique avec  $R = 4$  est un cas très particulier). On peut toutefois envisager d'imposer une nombre maximal d'itérations  $N_{SD_{max}}$  relativement faible ( $\sim 50$ ) et d'augmenter très progressivement la valeur de  $\lambda$ . Si  $\lambda$  devient près de la valeur optimale pour une OPI, le système convergera rapidement vers celle-ci s'il est dans son bassin d'attraction. Comme démontré plus loin, lorsque la dimension du système est supérieure à 1, on ne peut trouver un  $\lambda$  qui annule la pente du graphique  $N$  fonction de  $\log(g)$ .

Si on revient à l'équation 3.6, on remarque une chose importante : le nombre d'itérations pour faire varier le résidu de  $\Delta g$  ne dépend pas de  $g$  ! Ce qui veut dire qu'il en coûte aussi cher pour passer de  $g = 1.0$  à  $g = 0.1$  que pour passer de  $g = 10^{-13}$  à  $g = 10^{-14}$ . Ceci nous permet de suggérer une amélioration possible. En effet, il semble inutile de continuer à itérer le système selon  $SD$  à partir d'un résidu de  $g \approx 10^{-3} - 10^{-5}$ . On pourrait ainsi passer à des corrections Newton-Raphson  $\delta\mathbf{w}_{NR}$  (équation 1.27) lorsque ce résidu est atteint, en prenant soin de mettre une “protection” qui vérifie

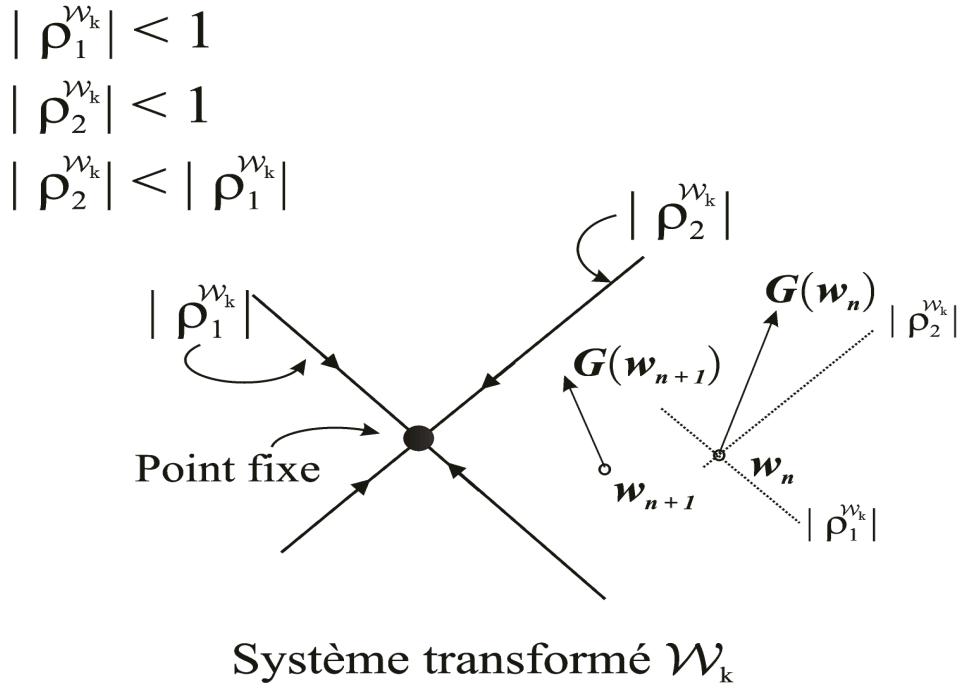
que  $g$  diminue lorsque les corrections  $\delta\mathbf{w}_{NR}$  sont enclenchées.

L'analyse que nous avons faite en une dimension nous a permis de dégager deux éléments importants. Le premier donne une estimation du nombre de pas  $SD$  qui doivent être faits pour atteindre un résidu  $g$  (équation 3.6). Le deuxième concerne l'ajout de corrections Newton-Raphson à partir d'un certain résidu. Nous allons maintenant essayer de transposer ces éléments pour des systèmes à  $D$  dimensions.

On commence avec le nombre d'itérations  $N_{SD}$  pour obtenir un résidu  $g$ . On cherche à amener le résidu  $g \equiv |\mathbf{G}|$  à une certaine précision. Les corrections  $\delta\mathbf{w}_n$  sont données par  $\lambda\mathbf{C}_k\mathbf{G}$ . La fonction  $\mathbf{G}$  après une itération  $SD$  est :

$$\begin{aligned}\mathbf{G}(\mathbf{w}_{n+1}) &= \mathbf{G}(\mathbf{w}_n + \delta\mathbf{w}_n) \\ &\sim \mathbf{G}(\mathbf{w}_n) + \mathbf{J}_G \Big|_{\mathbf{w}_n} \delta\mathbf{w}_n \\ &\sim \mathbf{G}(\mathbf{w}_n) + \mathbf{J}_G \Big|_{\mathbf{w}_n} \lambda\mathbf{C}_k\mathbf{G}(\mathbf{w}_n) \\ &\sim (\mathbf{1} + \lambda\mathbf{J}_G\mathbf{C}_k \Big|_{\mathbf{w}_n})\mathbf{G}(\mathbf{w}_n).\end{aligned}\quad (3.7)$$

Le terme  $(\mathbf{1} + \lambda\mathbf{J}_G\mathbf{C}_k)$  correspond simplement à la matrice Jacobienne dans le système transformé (chapitre 1, équation 1.31). Sur les OPIs, cette matrice possède donc des valeurs propres qui sont de module plus petit que 1. Les valeurs propres du système transformé sont notées  $\rho_d^{\mathcal{W}_k}$  (chapitre 1, équation 1.32). Parmi l'ensemble des  $\rho_d^{\mathcal{W}_k}$ , celle qui a le module le plus grand est celle qui *retarde* la convergence. La figure 3.7 illustre ce que nous venons de mentionner. Un point fixe est représenté en  $2D$  avec les deux directions données par les vecteurs propres dans le système  $\mathcal{W}_k$ . Nous avons indiqué le vecteur  $\mathbf{G}(\mathbf{w}_n)$  et  $\mathbf{G}(\mathbf{w}_{n+1})$  (équation 3.7). En supposant que la matrice  $\mathbf{J}_G$  est constante dans le voisinage du point fixe, on déduit que les deux composantes de  $\mathbf{G}(\mathbf{w}_n)$ , qui sont selon les deux directions propres, changent à chaque itération par des facteurs différents. L'effet global est d'amener  $\mathbf{G}(\mathbf{w}_{n+1})$  progressivement près de la droite correspondant à la valeur propre la plus grande (en valeur absolue). Après quelques itérations, le vecteur  $\mathbf{G}(\mathbf{w}_n)$  se retrouve près de cette droite. Le résidu  $g = |\mathbf{G}|$  change alors comme  $g_{n+1} = (\rho_{max}^{\mathcal{W}_k})g_n$ .

FIG. 3.7 – Dynamique autour du point fixe dans le système transformé  $\mathcal{W}_k$ .

De façon générale, on ne peut écrire les valeurs propres  $\rho_d^{\mathcal{W}_k}$  directement en fonction de  $\lambda$  car elles dépendent des coefficients de la matrice  $\mathbf{J}_G$ . Cependant, pour  $C_k = C_1$ , on estime la valeur propre la plus grande  $\rho_{max}^{\mathcal{W}_k}$  comme :

$$\rho_{max}^{\mathcal{W}_k} \sim 1 - \lambda.$$

Cette dernière équation est justifiée car les valeurs propres de  $\mathbf{J}_G$  sont de la forme  $\rho_d - 1$  où les  $\rho_d$  sont les  $d$  valeurs propres dans le système  $\mathbf{F}^{(m)}$  initial. En général pour les OPIs hyperboliques, une des valeurs propres  $\rho_d$  est près de 0 lorsque  $m$  augmente et c'est ce qui conduit à  $\rho_{max}^{\mathcal{W}_k} \sim 1 - \lambda$ .

On peut maintenant écrire le résidu  $g_n = |\mathbf{G}(\mathbf{w}_n)|$  après  $N_{SD}$  itérations pour la matrice  $C_1$  :

$$g_{n+N_{SD}} \sim g_n (1 - \lambda)^{N_{SD}}. \quad (3.8)$$

Ici, nous avons itéré l'équation 3.7  $N_{SD}$  fois avec la matrice  $C_1$  en supposant que la plus grande valeur propre de  $\mathbf{1} + \lambda \mathbf{J}_G|_{\mathbf{w}_n}$  est toujours de l'ordre de  $1 - \lambda$ . On doit donc

s'attendre à ce que le nombre d'itérations  $N_{SD}$  pour obtenir un résidu  $g$  avec la matrice  $C_k = C_1$  varie comme :

$$N_{SD} \sim \frac{\log(g)}{\log(1 - \lambda)}. \quad (3.9)$$

La figure 3.8 montre le nombre d'itérations  $SD$  qui ont été effectuées en moyenne pour atteindre un résidu  $g$  dans le système Ikéda (échelle semi-log). Nous avons fait ce calcul avec les matrices  $\mathbf{C}_1$  (traits pleins —) et  $\mathbf{C}_3$  (traits pointillés - - -) et ce, pour des périodes 3 et 5. Pour la matrice  $C_1$ , la pente du graphique ne dépend pas de la période. Sa valeur est très semblable à la valeur de référence (Nous avons aussi indiqué la pente de référence  $1 / \log(1 - \lambda)$ , équation 3.9). Pour la matrice  $C_3$ , on ne peut tirer de conclusion générale sur la pente du graphique. La valeur propre  $\rho_{max}^{\mathcal{W}_k}$  qui détermine en quelque sorte la valeur de cette pente dépend des éléments de la matrice  $\mathbf{J}_G$  lorsque  $k \neq 1$ . Comme ces éléments changent d'une OPIs à l'autre et donc d'une période à l'autre, la pente du graphique change également et il devient difficile de faire une prédiction théorique.

L'équation 3.9 nous permet cependant d'établir un ordre de grandeur quant au nombre d'itérations nécessaires pour faire varier le résidu d'une quantité  $\Delta g$ . Tout comme le cas 1D, ce nombre d'itérations est indépendant de  $g$ . La suggestion d'utiliser une correction Newton-Raphson à partir d'un certain résidu est donc toujours justifiée en  $D$  dimensions. Si on se propose d'atteindre un résidu  $g_{NR}$  avant d'enclencher les corrections Newton-Raphson, alors le nombre maximal d'itérations  $SD$  devrait être choisi comme :

$$D > 1 : N_{SD_{max}} \sim \frac{\log(g_{NR})}{\log(1 - \lambda)} . \quad (3.10)$$

Nous résumons ici les 2 principales modifications qui devraient être apportées à l'algorithme  $SD$  afin d'augmenter son efficacité :

1. Il faut choisir un résidu  $g_{NR}$  à partir duquel on essaie des corrections de type Newton-Raphson. Typiquement  $g_{NR} \sim 10^{-2} - 10^{-4}$  selon la période recherchée.
2. Lorsque  $g < g_{NR}$ , on passe aux corrections  $\delta \mathbf{w}_n^{NR}$  (équation 1.27).

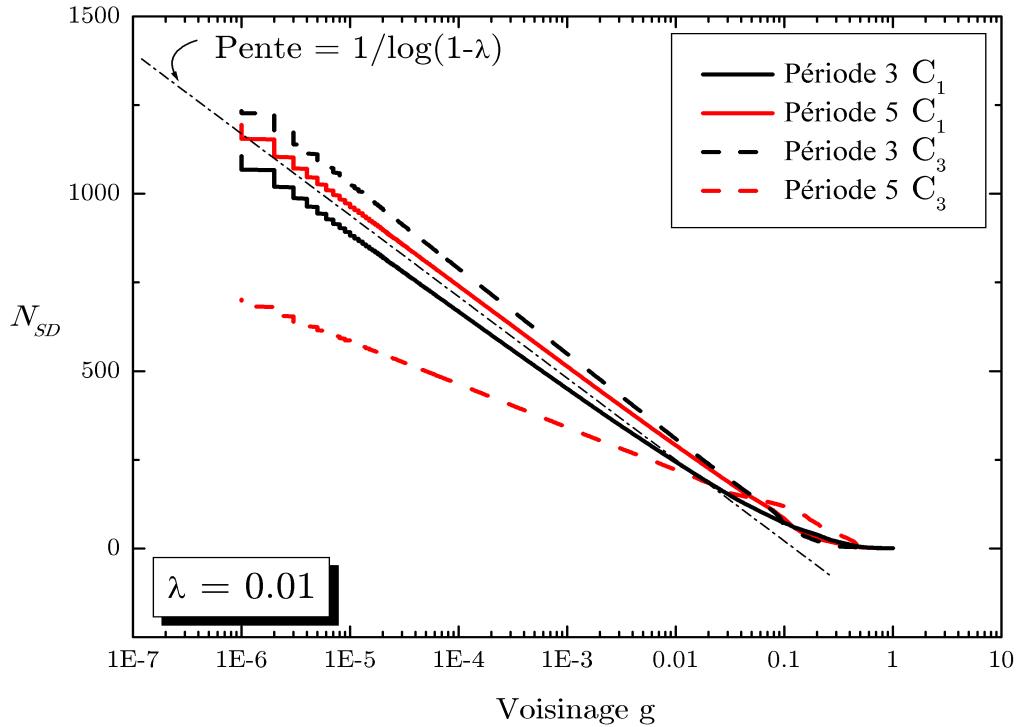


FIG. 3.8 – Nombre moyen de pas  $SD$  pour atteindre un résidu  $g \equiv |\mathbf{G}(\mathbf{w})|$  dans Ikéda avec  $C_k = C_1, C_3$ .

3. Le nombre d’itérations maximales à faire avant de passer à une autre condition initiale devrait être  $N_{max} \sim \log(g_{NR}) / \log(1 - \lambda)$ .

### 3.2.4 Ikéda revisité

Nous reprenons ici la détection d’OPIs chez Ikéda avec un algorithme  $SD$  modifié. À partir d’un résidu  $g_{NR}$ , on remplace les corrections  $SD$  (équation 1.30) par des corrections Newton-Raphson (équation 1.27). On montre à la figure 3.9a le nombre d’OPIs détectées en fonction de  $\lambda$  avec ce nouvel algorithme. La position des  $\lambda_c$  (indiquée par un point plein) n’est pas modifiée par rapport à l’ancien algorithme (symboles  $\triangle$ ). Ceci était à prévoir puisque la valeur de  $\lambda_c$  dans le cas du nouvel algorithme est principalement dictée par la stabilité des OPIs. Nous disons ici “principalement” car le

fait d'employer des corrections Newton-Raphson à partir d'un certain voisinage brise le lien entre la stabilité des OPIs et la valeur de  $\lambda_c$ . Si par un pur hasard on choisit une condition initiale donnant un résidu inférieur à  $g_{NR}$ , on aura probablement convergence vers l'OPIs grâce aux corrections  $NR$ . Toutefois, les conditions initiales que nous choisissons sont les composantes des OPIs de période  $(m - 1)$ , il y a donc peu de chance d'avoir initialement un résidu inférieur à  $g_{NR}$  lors de la recherche d'une période  $m$ . C'est pourquoi, les valeurs de  $\lambda_c$  ne sont pas affectées par l'ajout des corrections  $NR$ .

À la figure 3.9b, on compare le nombre d'itérations de ce nouvel algorithme  $SD$  aux nombres d'itérations  $DL$  pour la détection des OPIs de différentes périodes. Même avec les corrections que nous avons apportées,  $DL$  demeure nettement plus efficace. Ceci s'explique relativement bien. Nous avons démontré que le nombre d'itérations  $SD$  pour atteindre un résidu  $g$  varie environ comme  $N_{SD} \sim \log(g)/\log(1 - \lambda)$ . Pour amener le résidu aux environs de  $g_{NR}$ , il faut faire environ  $N_{SD} \sim -1/\log(1 - \lambda)$  itérations. Ceci est une limite inférieure bien optimiste. Il s'agit d'une situation où les conditions initiales doivent être itérées sur seulement un ordre de grandeur avant d'atteindre le résidu  $g_{NR}$ . Pour ce qui est de  $DL$ , on ne peut déterminer ce nombre facilement mais dans le pire des cas, on itère  $N_{DL} \sim 10\beta$  qui est nombre maximal d'itérations  $DL$  possible fixé dans l'algorithme. Ce critère, bien qu'un peu arbitraire, semble donner de bons résultats. Ainsi, on doit s'attendre à ce que pour la détection des OPIs, le rapport  $N_{SD}/N_{DL}$  soit, dans le meilleur de cas, de l'ordre de :

$$\frac{N_{SD}}{N_{DL}} \sim \frac{-1}{10\beta \log(1 - \lambda)}. \quad (3.11)$$

En mettant les valeurs de  $\lambda_c$  et  $\beta_c$  dans cette dernière relation, on arrive à estimer le rapport  $N_{SD}/N_{DL}$  “optimal” pour le système Ikéda. Nous avons calculé ce rapport et l'avons aussi placé sur la figure 3.9. Même dans ces conditions bien optimistes, l'algorithme  $SD$  est toujours plus lent que  $DL$  et l'écart se creuse avec la période. Pour  $SD$ , si on souhaite détecter l'ensemble des OPIs, la valeur de  $\lambda_c$  doit nécessairement diminuer avec  $m$  qui augmente. La tendance qui se dégage ici pour Ikéda se manifestera

aussi pour d'autres systèmes chaotiques.

### 3.2.5 Quelques nuances dans l'algorithme *DL*

L'algorithme *DL* semble plus performant que *SD* pour une détection rapide de l'ensemble des OPIs d'un système chaotique. Cependant, il faut garder à l'esprit que cette méthode est légèrement plus difficile à implémenter que *SD*. L'efficacité et la rapidité de *DL* sont en partie dues à la présence de la matrice jacobienne dans l'évaluation des corrections  $\delta\mathbf{w}^{DL}$ . Cette matrice doit donc être évaluée à chaque pas et de façon relativement précise. Pour les applications discrètes, la matrice jacobienne est généralement disponible sous la forme analytique et son calcul ne pose pas vraiment de problème. Dans les cas des flot continus, il faut intégrer minutieusement les équations variationnelles comme expliqué au chapitre 2. Si ce travail est mal fait, les performances de *SD* seront compromises.

Nous soulignons ici l'importance de la matrice jacobienne dans l'algorithme *SD*. Nous avons relancé la détection d'OPIs chez Ikéda en ajoutant un certain seuil de bruit blanc sur les éléments de cette matrice. La figure 3.10a montre comment se déplace  $\beta_c$  pour la détection d'une période-13 pour différents niveaux de bruit (rappelons que  $\beta_c$  est la valeur de  $\beta$  minimale pour détecter l'ensemble des OPIs ; elle est indiquée par les symboles pleins sur le graphique). Lorsque la matrice jacobienne est exacte,  $\beta_c$  est de l'ordre de 100 pour une période 13. Avec 0.5% de bruit,  $\beta_c$  devient de l'ordre de 2000 ! Sur la figure 3.10b, on indique la valeur de  $\beta_c$  en fonction de la période pour deux niveaux de bruit ( $\beta_c$  est alors normalisé par rapport à sa valeur lorsque la matrice jacobienne est exacte). Avec la période qui augmente, l'efficacité de *DL* se trouve nettement compromise si la matrice jacobienne fait défaut. On compare à la figure 3.11 le nombre d'itérations *SD* par rapport à *DL* pour différents niveaux de bruit sur les éléments de la matrice (Ce graphique a été présenté à la figure 3.5 dans le cas où  $\mathbf{J}_G$  est connue exactement). On remarque qu'avec 0.5% d'erreur, l'efficacité de

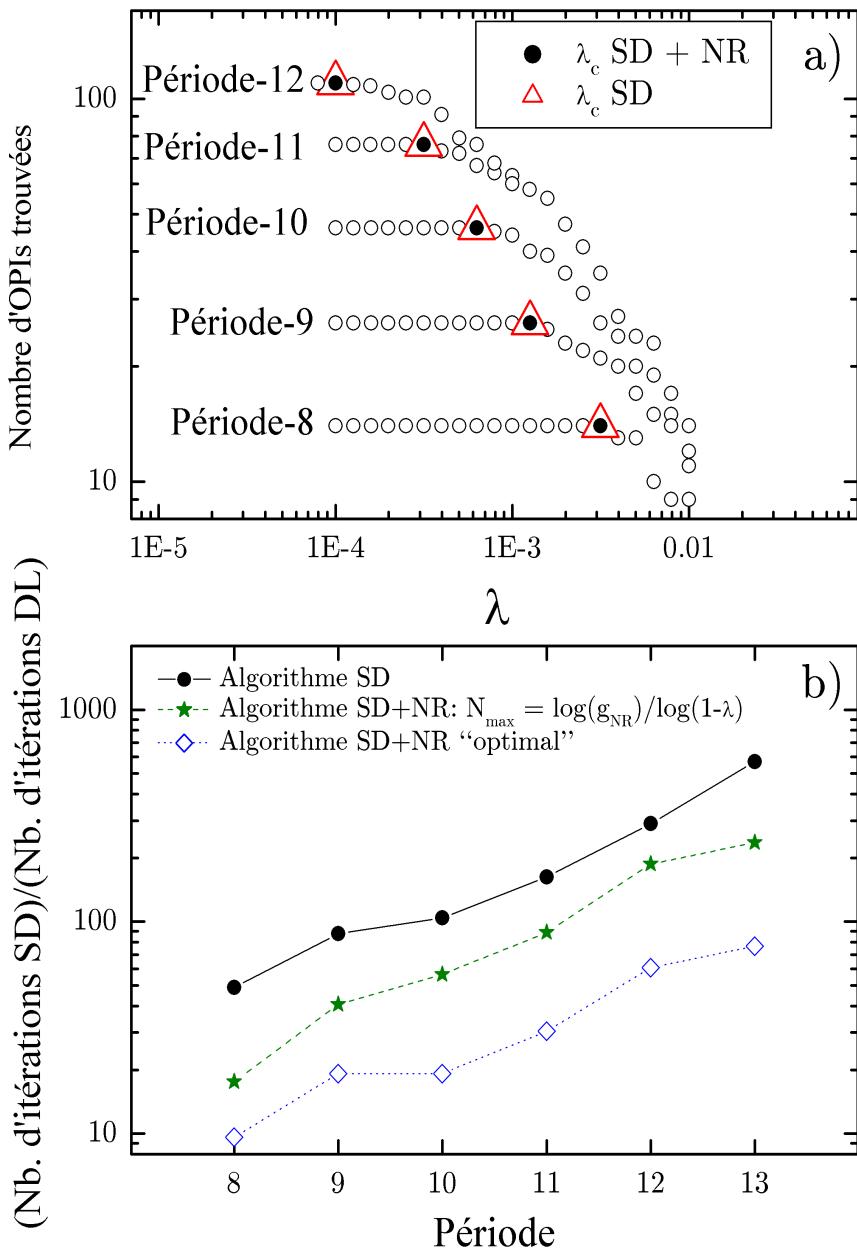


FIG. 3.9 – Algorithme *SD* avec l’ajout de corrections Newton-Raphson.

Sur la figure 3.9a, on donne le nombre d’OPIs détectées dans Ikéda en fonction de  $\lambda$ . La position des  $\lambda_c$  n’est pas affectée par l’ajout de corrections Newton-Raphson. En 3.9b, on donne le nombre d’itérations fait avec ce nouvel algorithme pour différentes périodes. Ce nombre est normalisé en fonction du nombre d’itérations fait par l’algorithme *DL*.

$DL$  chute considérablement. Par exemple, il faut environ 1000 fois moins d’itérations  $DL$  que d’itération  $SD$  pour parvenir à la détection des OPIs de période-13 lorsque  $\mathbf{J}_G$  est connue exactement. Avec 0.5% d’erreur sur  $\mathbf{J}_G$ , le nombre d’itérations  $DL$  se rapproche nettement de  $SD$  (seulement 10 fois moins).

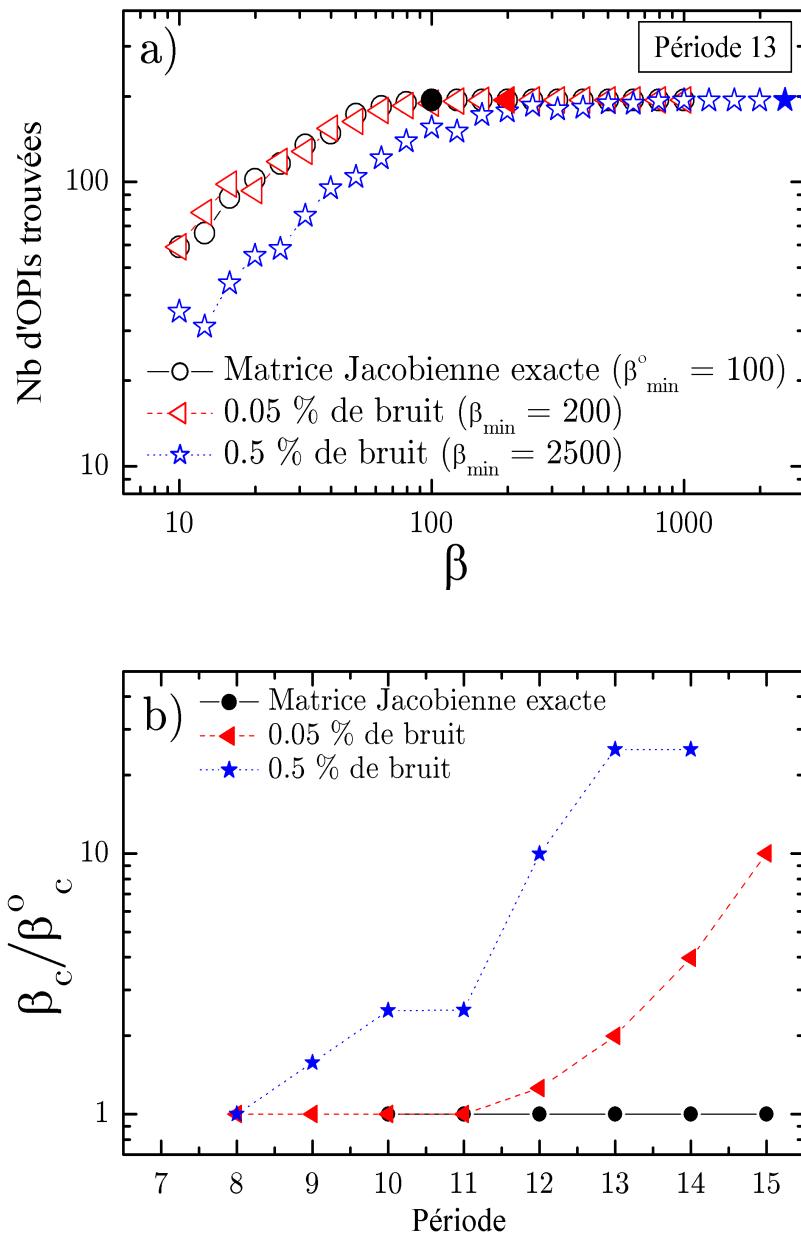
Un autre point que nous voulons soulever ici avant de quitter le système Ikéda est le choix des conditions initiales. Pour la détection des OPIs de période- $m$ , nous avons jusqu’à maintenant utilisé les composantes des OPIs de période- $(m-1)$ . Est-ce vraiment justifié ? Il semble difficile d’établir le lien entre les bassins d’attraction des périodes  $m$  et la position des OPIs de période- $(m-1)$ . Nous avons refait une recherche d’OPI à l’aide de l’algorithme  $DL$  mais cette fois, en utilisant le même nombre des conditions initiales réparties aléatoirement sur l’attracteur. Le graphique 3.12 montre comment évolue  $\beta_c$  avec la période lorsque les conditions initiales sont choisies de cette façon. Nous avons indiqué par le symbole  $\triangle$  la position de  $\beta_c$  lorsque les OPIs de périodes  $(m-1)$  sont utilisées comme conditions initiales. On remarque qu’il y a une légère différence lorsque  $m$  augmente. On explique difficilement cette différence.

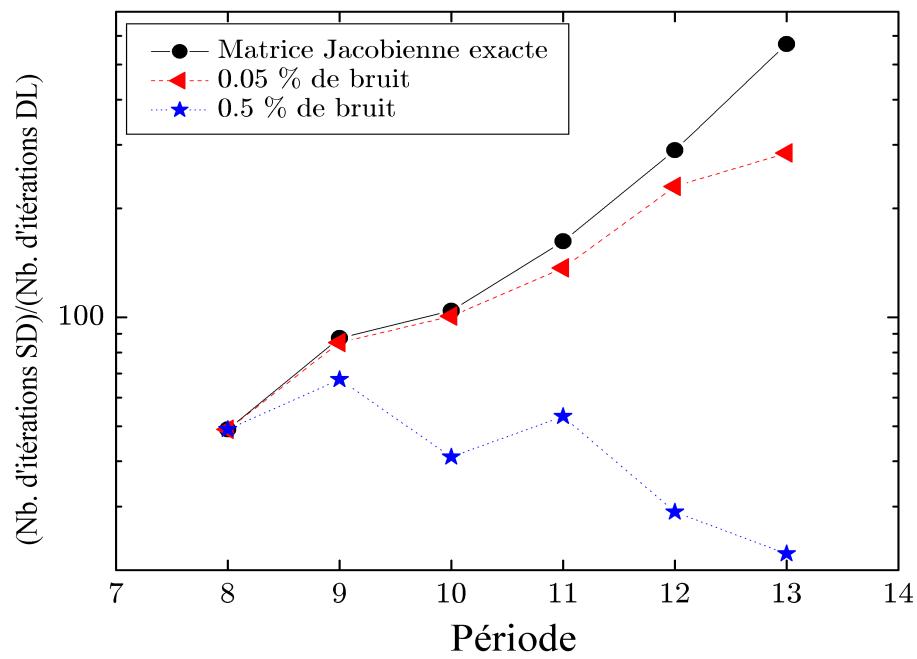
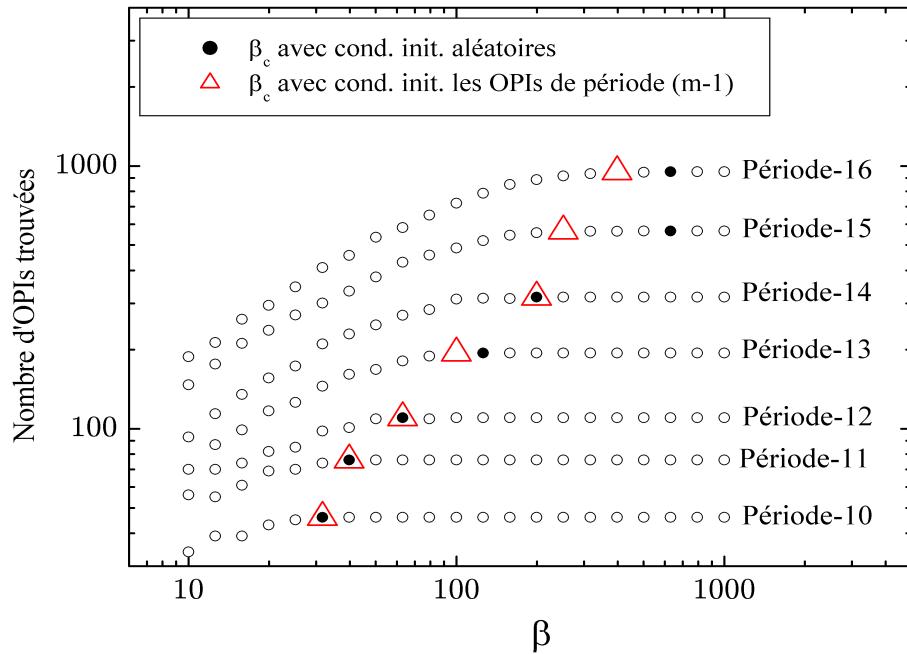
### 3.3 Détection d’OPIs pour d’autres systèmes

Nous montrons ici quelques résultats obtenus avec l’algorithme  $DL$  sur d’autres systèmes chaotiques.

#### 3.3.1 Systèmes conservatifs

Nous avons fait une recherche d’OPIs sur l’Application Standard avec  $K = 1.25$ . Rappelons que cette application présente une dynamique mixte (région chaotique et îlots de stabilité) pour cette valeur de  $K$ . Comme nous l’avons mentionné au chapitre 1, les méthodes de détection par récurrence sont ici moins performantes dû au phénomène

FIG. 3.10 – Algorithme *DL* avec bruit sur la matrice jacobienne.

FIG. 3.11 – Rapport  $N_{SD}/N_{DL}$  pour différents niveaux bruit sur la matrice jacobienne.FIG. 3.12 – Nombre d’OPIs détectées par  $DL$  en fonction de  $\beta$  pour des conditions initiales choisies aléatoirement sur l’attracteur.

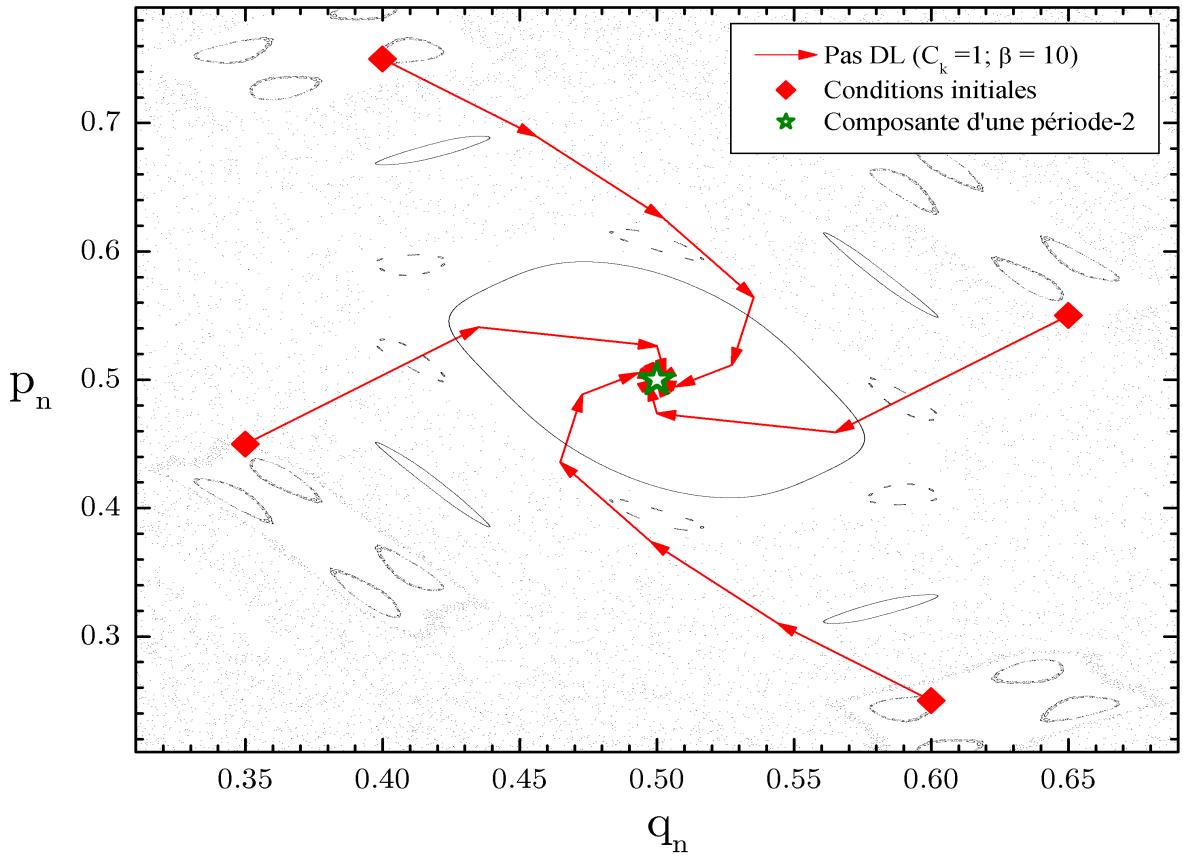


FIG. 3.13 – Dynamique de l'algorithme *DL* sur l'Application Standard ( $C_k = C_1$ ;  $\beta = 10$ ).

de piégeage (*sticking*) autour des îlots. L'algorithme *DL* (tout comme *SD*) est complètement “insensible” à la présence de ces derniers. En effet, les OPIs qui se cachent à l'intérieur des îlots sont de type elliptique dans le système original ( $|\rho_{1,2}| = 1$ ,  $|Im(\rho_{1,2})| > 0$ , tableau 1.3). Sous transformation *SD*, ces points fixes deviennent des foyers stables avec ( $|\rho_{1,2}^{W_k}| < 1$ ,  $|Im(\rho_{1,2}^{W_k})| > 0$ ) et le système converge alors vers ceux-ci. On montre à la figure 3.13 l'évolution de quatre conditions initiales sous itération *DL* ( $C_k = C_1, \beta = 10$ ). Avec ces quatre conditions, on a convergence vers le centre d'un îlot où se cache la composante d'une période-2. On voit clairement la forme de la spirale qui caractérise les foyers stables.

Pour la recherche des OPIs, nous avons essayé plusieurs valeurs de  $\beta$  et ce, pour chacune des périodes recherchées. Pour les périodes  $m \leq 6$ , nous utilisons les conditions initiales réparties uniformément dans l'espace des phases. Pour les périodes  $m > 6$ , les conditions initiales sont les composantes des OPIs de périodes  $m - 1$ . La figure 3.14a montre le nombre d'OPIs trouvées en fonction de la valeur de  $\beta$ . Ici, la tendance est complètement différente du cas Ikéda. La valeur de  $\beta_c$  n'augmente pas vraiment avec la période. Étonnamment, des valeurs de  $\beta \sim 1$  donnent de bons résultats (il faut alors moins de 10 itérations pour converger!). On indique sur la figure 3.14b le module moyen et la valeur maximale des valeurs propres instables (resp.  $\langle |\rho_u| \rangle$  et  $|\rho_u|_{max}$ ) pour les différentes périodes. Contrairement à Ikéda, il ne semble pas y avoir de relation empirique entre  $\beta_c$  et  $|\rho_u|_{max}$ .

On remarque une très grande fluctuation du nombre d'OPIs de période-11 en fonction de  $\beta$ . L'Application Standard possède une dynamique extrêmement complexe. Sous une variation de  $K$ , de nombreux îlots peuvent se former. Autour de  $K = 1.25$ , il semble y avoir une très grande quantité de minuscules îlots cachant des périodes-11. On localise ces OPIs sur la figure 3.15. Les îlots sont trop petits pour être distinguables. Mentionnons qu'il existe d'autres OPIs de période-11 facilement détectables ailleurs dans l'espace des phases. Nous ne les avons pas placées afin de ne pas surcharger la figure.

Nous avons augmenté légèrement la valeur de  $K$  pour essayer de diminuer le nombre ces petits îlots de période-11 et augmenter leur taille. Autour de  $K = 1.7$  on arrive à bien distinguer les îlots (figure 3.16). Il ne reste alors que deux OPIs elliptiques différentes associées à ceux-ci. Les symboles  $\star$  et  $\bullet$  indiquent les OPIs elliptiques et les symboles  $\triangleleft$  et  $\triangleright$  sont réservés pour les orbites hyperboliques dans cette même région. Sans avoir fait une étude très détaillée de ce phénomène de bifurcation, on constate numériquement que cette “explosion” de périodes-11 se situe autour de  $K = 1.16917$ . Après  $K > 1.88825$ , il n'y a plus de période-11 elliptique. Évidemment, les périodes-11 ne sont pas uniques et d'autres périodes subissent le même sort.

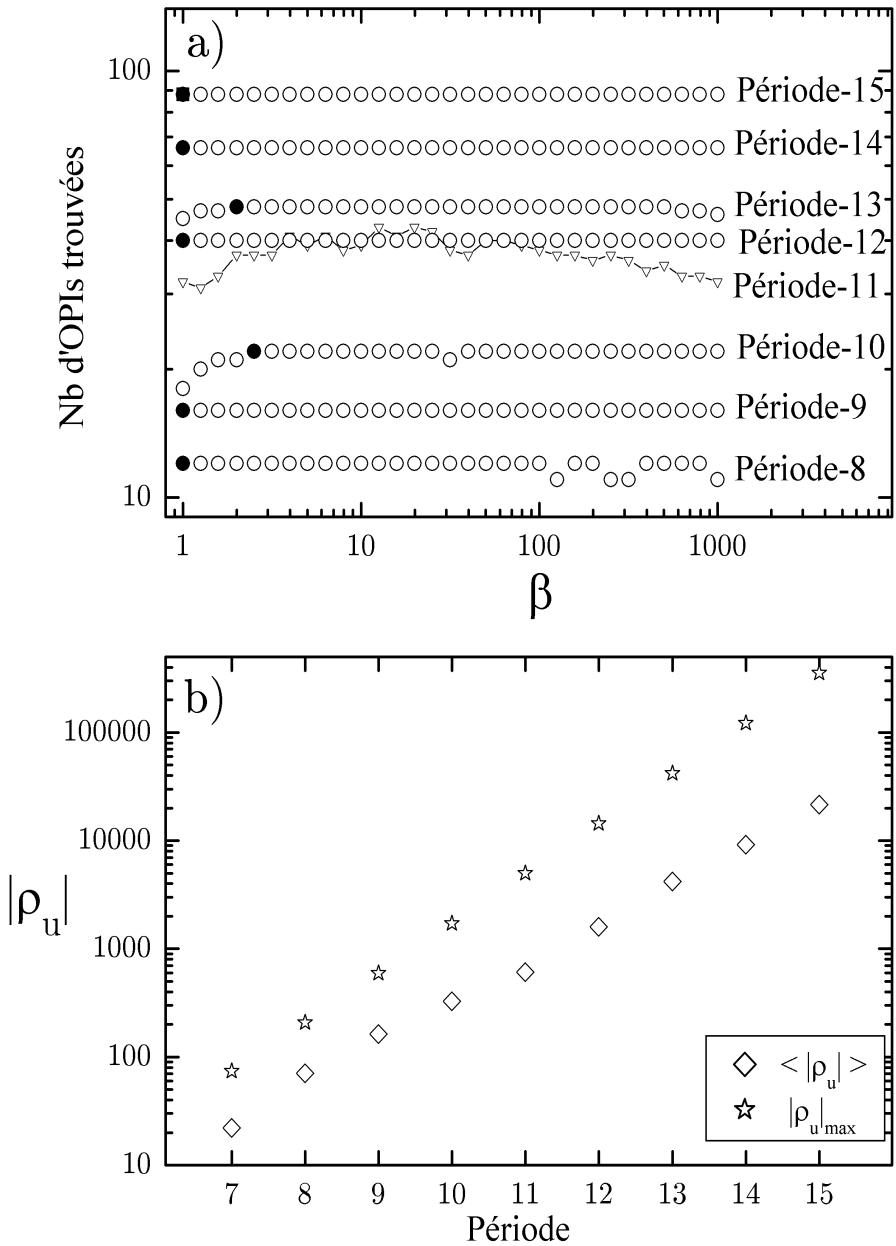
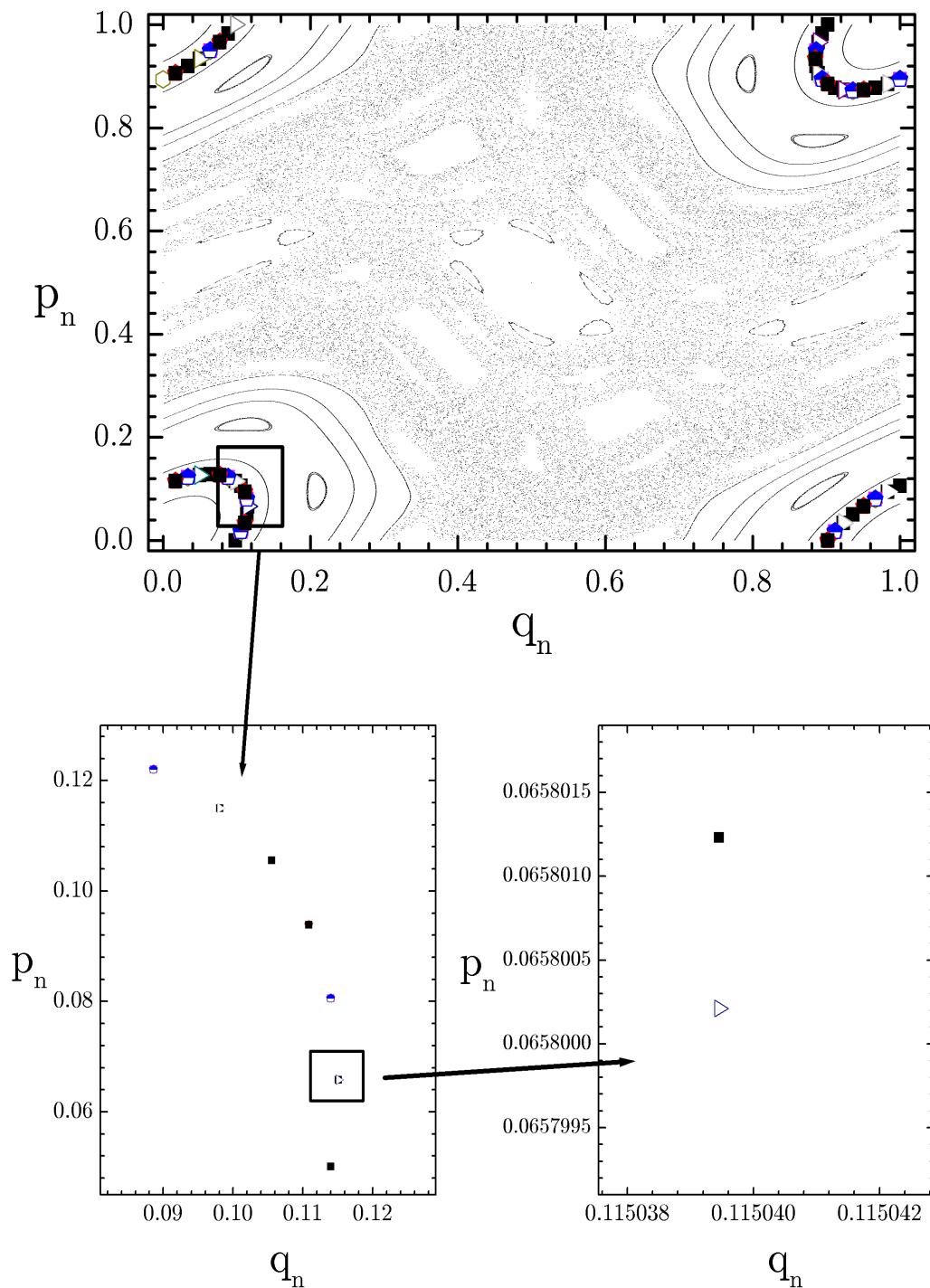
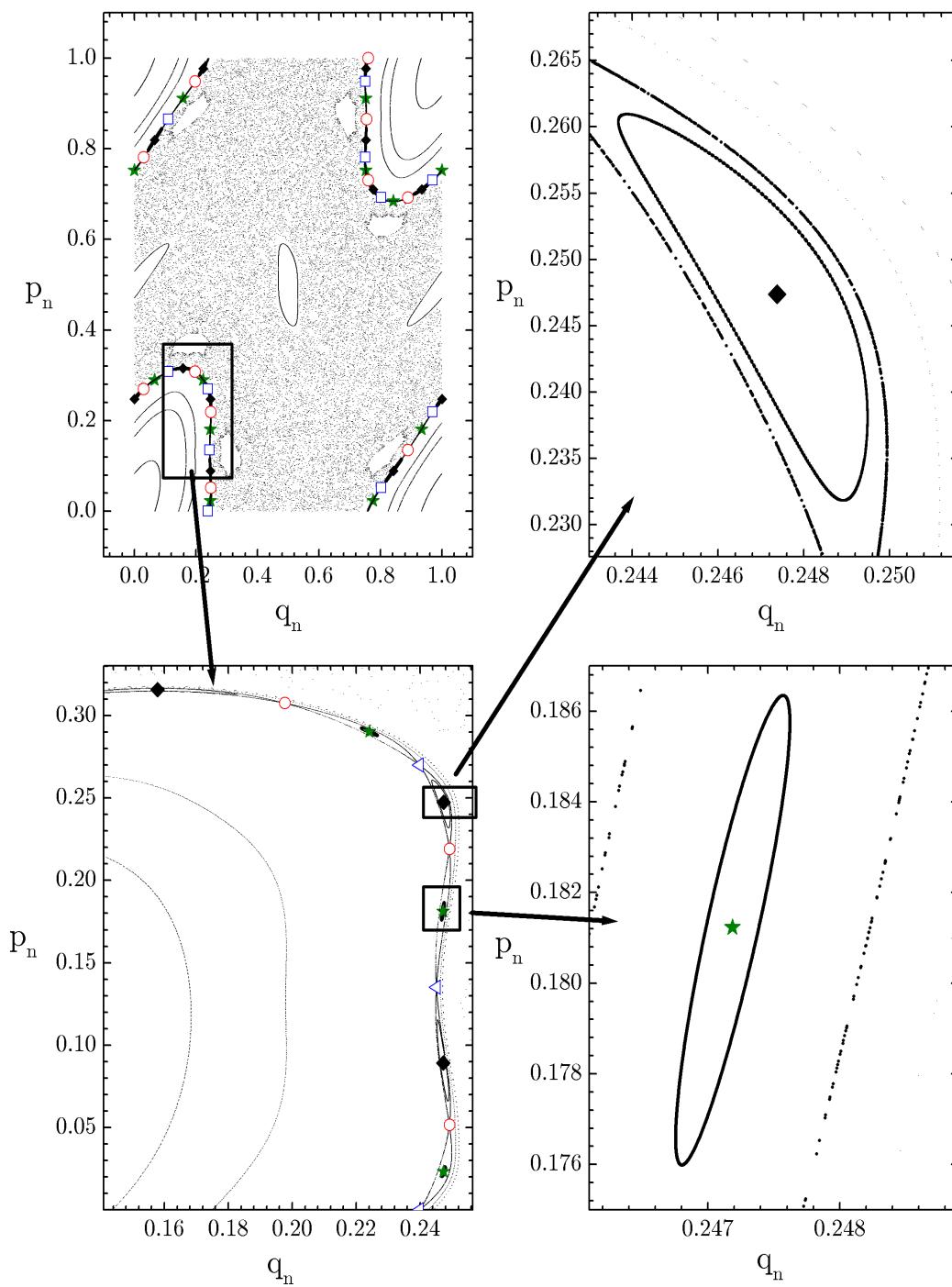


FIG. 3.14 – Recherche des OPIs de l'Application Standard.

La figure 3.14a donne le nombre d'OPIs trouvées en fonction de  $\beta$ . Il ne semble pas y avoir de relation entre  $\beta_c$  et les valeurs propres (fig. 3.14b) de la matrice jacobienne.

FIG. 3.15 – OPIs elliptiques de période-11 dans l’Application Standard ( $K = 1.25$ ).

FIG. 3.16 – OPIs elliptiques de période-11 dans l’Application Standard ( $K = 1.7$ ).

On donne dans le tableau 3.2 le nombre d’OPIs trouvées pour l’Application Standard et ce, pour différentes périodes. On indique aussi dans ce tableau quelques informations relatives aux OPIs.

Période	Nombre d’OPIs total	Nombre d’OPIs elliptiques	$\langle  \rho_u  \rangle$	$ \rho_u _{max}$
1	2	1	1.95293	2.90587
2	2	1	2.15191	3.30383
3	4	2	2.53142	4.06285
4	4	0	3.78972	6.14000
5	8	2	6.41305	11.81018
6	6	1	17.24747	27.78047
7	16	2	22.02053	73.67616
8	12	3	70.43891	206.5665
9	22	4	162.51796	592.276
10	22	4	326.56414	1712.65559
11	54?	?	609.4975	4967.89425
12	40	2	1594.43485	14426.75493
13	48	5	4190.45735	41912.52813
14	66	0	9179.01539	121782.145
15	88	8	21525.06574	353872.34131

TAB. 3.2 – Les OPIs de l’Application Standard ( $K = 1.25$ ).

### 3.3.2 Systèmes de dimensions 3 et 4

Nous incluons ici une courte section pour discuter de la recherche de cycles périodiques dans des systèmes de dimensions plus élevées. Nous avons d’abord lancé une recherche d’OPIs sur le système Hénon-3D. L’algorithme utilisé est *DL* avec la même stratégie que précédemment. Les OPIs de périodes-1 à 6 sont détectées en utilisant un certain nombre de conditions initiales choisies aléatoirement. Par la suite, les composantes des périodes  $m - 1$  deviennent les conditions initiales pour la recherche des périodes- $m$ . Tout comme pour l’Application Standard, l’évolution de  $\beta_c$  avec la période ne semble pas suivre de règle précise (figure 3.17). Et tout comme pour l’Application Standard, il

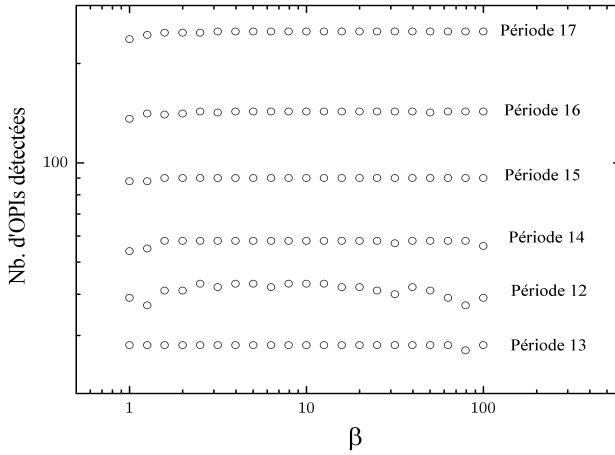


FIG. 3.17 – Nombre d’OPIs détectées en fonction de  $\beta$  pour le système Hénon-3D ( $A = 1.76, B = 0.1$ ).

est plus ou moins justifié de parler de “ $\beta_c$ ” puisque le nombre d’OPIs détectées ne reste pas toujours constant après avoir atteint un maximum. C’est pourquoi nous croyons que l’utilisation des composantes de période  $m - 1$  comme conditions initiales, tout comme la position de  $\beta_c$ , devraient plutôt être pris comme point de départ pour une recherche d’OPIs. On pourrait ainsi dégager une valeur de  $\beta$  qui semble bien fonctionner. Par la suite, on devrait choisir un bon nombre de conditions initiales (comme par exemple les composantes des orbites  $m - 2, m - 3$ , etc). L’ensemble de ces conditions initiales pourrait être divisé en une dizaine de sous-ensembles. Il suffirait alors d’itérer chacun des sous-ensembles avec la valeur de  $\beta$  qui semble optimale. Si le nombre d’OPIs ne change plus d’un sous-ensemble à l’autre, alors on peut être espérer un nombre d’OPIs complet. En fait, il faut bien comprendre qu’à la base, il n’y a pas de critère rigoureux pour s’assurer que le nombre d’OPIs est bien complet.

Nous avons également fait une recherche d’OPIs sur le Pendule Double, qui est un système dynamique en dimension 4. Ici, on semble entrevoir une limite pour les algorithmes *SD* et *DL*. Le Pendule Double possède 36 points fixes [84]. Parmi ces points, quatre sont situés dans des régions peu visitées de l’attracteur. Les bassins d’attraction

de ces quatre points fixes sont très petits. En plaçant une condition initiale à l'intérieur d'un de ces bassins, on converge évidemment vers l'OPI. Par contre, si on opte pour des conditions initiales choisies aléatoirement sur l'attracteur, alors il en faut un nombre considérable ( $> 10^6$ ) avant d'en trouver une qui soit à l'intérieur du bassin. Ce qui nous ramène à une des questions importantes reliées à ces algorithmes : Comment choisir les conditions initiales ? C'est une question à laquelle nous n'arrivons pas à répondre de façon complètement satisfaisante et générale.

### 3.3.3 Systèmes continus

Nous terminons avec quelques résultats pour les systèmes continus. Les systèmes de Lorenz (frot dissipatif) et DKP (frot conservatif) ont été utilisés comme banc d'essais. Encore une fois ici, les systèmes continus ne posent pas vraiment de problème *si* tout le travail pour obtenir numériquement le retour sur la Section de Poincaré et la construction de la matrice jacobienne a été fait correctement. Pour Lorenz, nous avons rapidement détecté les OPIs de périodes 1 à 4 en utilisant quelques conditions initiales choisies aléatoirement parmi les états de Lorenz sur la Section de Poincaré. Pour les autres périodes, nous utilisons toujours les composantes des OPIs de période  $m - 1$ . Ici, le nombre d'OPIs varie peu avec  $\beta$ .

On indique dans le tableau 3.3 les OPIs pour Lorenz jusqu'à la période-15 (Section de Poincaré :  $z_n = 44.92$ ,  $\frac{dz}{dt} > 0$ ).<sup>1</sup> On donne dans ce tableau le module moyen ainsi que le module maximal des valeurs propres instables (resp.  $\langle |\rho_u| \rangle$  et  $|\rho_u|_{max}$ ). La valeur moyenne des temps  $\tau$  pour effectuer  $m$  retours est aussi indiquée, tout comme les valeurs extrêmes de  $\tau$  (resp.  $\tau_{min}$  et  $\tau_{max}$ ). Mentionnons que le nombre d'OPIs trouvées pour les périodes 2 à 13 correspond au nombre donné dans la publication récente de Pingel & al. [72]. Pour la période-14, nous avons repéré une période de

---

<sup>1</sup>La période se définit ici comme le nombre de croisements positifs avec la Section de Poincaré. Pour la section que nous avons choisie, il n'y a pas de période-1.

plus. Le nombre de période-15 n'est pas donné dans leur article. Il faut noter que ces auteurs utilisent l'algorithme *SD* en y ajoutant des corrections Newton-Raphson et que le calcul de leur matrice jacobienne se fait par une technique légèrement différente de la nôtre. Finalement, ils choisissent les conditions initiales aléatoirement sur l'attracteur pour toutes les périodes ; il leur faut plus de conditions initiales pour repérer les OPIs de Lorenz que ce que nous avons utilisé (soit les composantes des OPIs de période  $(m - 1)$ ).

Pour *DKP*, on travaille avec les variables  $(\nu_n, p_{\nu_n})$  sur la Section de Poincaré  $\mu_n = 0$ . La variable  $p_{\mu_n}$  est éliminée par la contrainte sur l'énergie, comme expliqué à l'annexe C. Pour la section que nous avons choisie, les équations différentielles sont invariantes sous l'action des réflexions  $\nu \mapsto -\nu$  et  $p_\nu \mapsto -p_\nu$ . Ainsi, on détecte les OPIs de période-1 et 2 à l'aide de conditions initiales réparties uniformément dans le premier quadrant de l'espace des phases. Avec les OPIs trouvées (majoritairement dans le premier quadrant), on déduit la position des OPIs situées dans les autres quadrants. Pour les autres périodes  $(m > 2)$ , on utilise les composantes des périodes  $m - 1$  comme conditions initiales.

On indique dans le tableau 3.4 les OPIs de *DKP* jusqu'à la période 5. Ici, nos résultats sont complètement en désaccord avec ceux de Pingel & al. Le nombre d'OPIs que nous avons détectées est supérieur par au moins un facteur 10 à ce qui est donné dans leur article. Cette différence est peut être due au fait que ces auteurs ont choisi des valeurs de  $\lambda \sim 0.005$ . Comme ils utilisent l'algorithme *SD*, ils peuvent détecter les orbites avec des valeurs propres instables de module inférieur à  $\sim 400$  (ref. équation 1.37). Comme on le constate dans le tableau, il existe des orbites avec des valeurs propres bien au-delà de cette limite.

Période	Nombre d'OPIs total	$\langle  \rho_u  \rangle$	$ \rho_u _{max}$	$\langle \tau \rangle$	$\tau_{min}$	$\tau_{max}$
2	1	4.41038	4.41038	0.94082	0.94082	0.94082
3	2	8.67619	8.67619	1.39449	1.39449	1.39449
4	3	16.75789	18.26482	1.84339	1.83249	1.86521
5	6	34.09438	38.69379	2.30452	2.26227	2.33808
6	9	66.72287	79.59853	2.75605	2.68678	2.80301
7	18	136.77478	171.03043	3.21916	3.10754	3.27946
8	30	273.85346	350.1281	3.67641	3.52543	3.74317
9	56	552.71802	754.67562	4.13587	3.94101	4.2204
10	99	1113.214	1543.27273	4.59492	4.59492	4.68384
11	186	2250.56535	3328.77327	5.05483	5.05483	5.16125
12	335	4534.43045	6805.50084	5.51403	5.17699	5.62463
13	630	9153.58684	14681.53388	5.97364	5.58601	6.10208
14	1161	18454.30964	30013.95198	6.43304	5.99374	6.56545
15	2182	37234.34779	64751.57663	6.89255	6.40028	7.04291

TAB. 3.3 – Les OPIs du système de Lorenz.

On donne dans ce tableau les OPIs du système de Lorenz ( $\sigma = 16, b = 4, R = 45.92$ ) pour différentes périodes. La Section de Poincaré est placée en

$$(z_n = R - 1 = 44.92, \frac{dz}{dt} \Big|_{\tau_n} > 0).$$

Période	Nombre d'OPIs total	$\langle  \rho_u  \rangle$	$ \rho_u _{max}$	$\langle \tau \rangle$	$\tau_{min}$	$\tau_{max}$
1	323	844.5	2157	10.8	4.85	14.0
2	1077	37983.1	272399.0	17.4	7.20	27.9
3	1232	45935.2	925450.4	17.4	8.39	40.7
4	2550	131826.2	9343920	20.3	9.50	28.3
5	2932	206057.0	11529700	22.2	10.2	32.5

TAB. 3.4 – Les OPIs du système *DKP*.

On donne dans ce tableau les OPIs du système *DKP* ( $\epsilon = -0.1$ ) pour différentes périodes. La Section de Poincaré est placée en ( $\mu_n = 0, \frac{d\mu}{dt} \Big|_{\tau_n} > 0$ ).

## 3.4 Discussion

Nous voulons maintenant prendre un peu de recul et essayer de revenir sur les éléments importants de ce chapitre. À la base, un de nos buts était de comparer l'efficacité des algorithmes *SD* et *DL*. Nous voulions aussi vérifier leur bon fonctionnement sur différents systèmes chaotiques, dont notamment les flots continus conservatifs. Comme mentionné dans l'introduction de cette thèse, Pourbohloul [75] n'avait pu détecter des OPIs de période supérieure à 1 avec la méthode *SD* pour les flots conservatifs.

Chaque algorithme possède 3 paramètres qui doivent être choisis par l'utilisateur. Il s'agit :

- Choix de  $\lambda$  (Schmelcher-Diakonos) et  $\beta$  (Davichack-Lai)
- Choix des conditions initiales  $\mathbf{w}_0$
- Choix du nombre maximal d'itération  $N_{max}$

Nous nous sommes d'abord attardé sur le troisième point qui concerne le nombre d'itérations à essayer avant de passer à une autre condition initiale. Pour *SD*, nous avons dégagé un ordre de grandeur pour  $N_{max}$  en une dimension et en plusieurs dimensions. Pour faire varier le résidu de  $\Delta g$ , il faut faire un nombre d'itération  $N_{SD}$  de l'ordre de :

$$\begin{aligned} 1D : \quad N_{SD} &\sim \frac{\log(\Delta g)}{\log|1 + \lambda J_G|_{OPI}} \\ D > 1 : \quad N_{SD} &\sim \frac{\log(\Delta g)}{\log(1 - \lambda)}. \end{aligned}$$

Le résultat pour  $D > 1$  est valide avec la matrice  $\mathbf{C}_k = \mathbf{C}_1$  et pour les OPIs hyperboliques.

En une dimension, on remarque que le nombre d'itération  $N_{SD}$  devient minimal lorsque  $\lambda = \lambda_{optimal} \equiv 1/J_G|_{OPI}$ . Comme  $J_G|_{OPI}$  n'est pas connu, on suggère d'imposer un nombre d'itérations  $N_{SD_{max}} \sim 50$  et de diminuer progressivement la valeur de  $\lambda$  lors de la recherche des OPIs. La convergence devient presque quadratique lorsque  $\lambda$  est

près de la valeur optimale d'une OPI donnée. L'avantage de cette technique est qu'en  $1D$ , il devient inutile d'ajouter des corrections Newton-Raphson. Pour des systèmes continus quasi  $1D$  tel Lorenz, on évite ainsi d'intégrer les équations variationnelles.

Lorsque la dimension du système est supérieure à 1, nous avons suggéré de combiner l'algorithme  $SD$  à des corrections Newton-Raphson à partir de l'atteinte d'un certain résidu  $g_{NR}$ . Comme nous l'avons démontré, il faut faire, en moyenne, un nombre d'itérations  $N_{SD_{max}}$  de l'ordre de :

$$D > 1 : N_{SD_{max}} = \frac{\log(g_{NR})}{\log(1 - \lambda)},$$

pour atteindre le résidu  $g_{NR}$ . Comme les OPIs hyperboliques sont bien courantes dans les systèmes chaotiques et que la matrice  $\mathbf{C}_k = \mathbf{C}_1$  occupe une place importante parmi les matrices qui stabilisent les OPIs du système initial (voit tableau 1.3), on se doit d'imposer ce choix pour  $N_{SD_{max}}$ . Pour ce qui est de l'algorithme  $DL$ , il semble difficile d'établir un critère rigoureux pour  $N_{DL_{max}}$ . Nous avons utilisé  $N_{DL_{max}} = 10\beta$ . Avec ces choix pour  $N_{max}$ , il s'avère que  $DL$  converge toujours plus rapidement vers les OPIs et l'écart se creuse lorsque que ces dernières deviennent de plus en plus instables (i.e. lorsque  $m$  augmente, réf. fig. 3.9b).

Il faut insister sur le fait qu'en ajoutant des corrections Newton-Raphson à  $SD$  on règle le problème de la convergence linéaire près de l'OPI (i.e. pour  $g < g_{NR} \sim 10^{-3} - 10^{-4}$ ). Par contre, il reste à faire le bout de chemin jusqu'à l'atteinte du résidu  $g_{NR}$ ! Et c'est là tout le problème. L'algorithme  $SD$  converge linéairement avec un facteur  $1 - \lambda$  à chaque itération. Pour avancer d'un ordre de grandeur vers l'OPIs avec  $\lambda = 10^{-5}$ , la convergence globale de  $SD$  se paie à coup de  $2 \times 10^5$  itérations! Aussi, en ajoutant des corrections de type Newton-Raphson, on perd le lien qui existe entre  $\lambda$  et  $|\rho_u|_{max}$  tout comme ceux qui existent entre les matrices  $\mathbf{C}_k$  et le type d'OPI détectée. De plus, on se retrouve avec un problème supplémentaire soit celui de choisir un résidu  $g_{NR}$  à partir duquel on applique les corrections Newton-Raphson. Pour ces différentes raisons, nous ne recommandons pas l'ajout de telles corrections à l'algorithme  $SD$ . Il

est plutôt avantageux de choisir la méthode  $DL$  si une détection complète des OPIs est souhaitée.

Il est bon ici de nuancer quelque peu sur *détection complète des OPIs*. On s'intéresse aux OPIs principalement pour évaluer des quantités globales de la dynamique, comme mentionné au chapitre 1. Ces quantités se calculent par des sommes infinies de termes dont l'importance est directement reliée à la stabilité des cycles. Les termes les plus importants sont ceux associés aux OPIs les moins instables (réf. équation 1.19). Il est donc essentiel de repérer les OPIs les moins instables d'abord. Par exemple pour  $DKP$ , nous avons détecté 323 cycles de période-1. Plusieurs de ces cycles sont plus instables que les OPIs de période-5. Leur contribution à la somme infinie est donc plus négligeable que certains cycles de période-5. C'est en voulant repérer les cycles très instables que nous avons mis de côté l'algorithme  $SD$ . Pour détecter ces cycles, on utilise alors des  $\lambda$  qui rendent la méthode complètement inefficace. Le choix de l'algorithme devrait se faire en fonction des besoins en OPIs et selon la convergence de la somme qui décrit la propriété dynamique étudiée.

Si on est principalement motivé par la détection de l'ensemble des OPIs pour  $D > 1$ , on devrait opter pour l'algorithme  $DL$ . On pourrait aussi penser à un schème différent de celui de  $DL$  pour solutionner l'équation 1.40. Rappelons brièvement que l'algorithme  $DL$  est une méthode semi-implicite pour obtenir les points stationnaires de l'équation 1.40. D'autres schèmes numériques peuvent être envisagés. Quelques tests ont été réalisés avec des méthodes explicites telles *Runge-Kutta* et *Bulirsch-Stoer* [76]§16. Toutefois, ces dernières souffrent du même problème que l'algorithme  $SD$  (qui est la méthode d'Euler explicite appliquée à 1.40), soit un pas qui devient trop petit lorsque la période augmente. Comme mentionné dans le chapitre 1, l'équation 1.40 se classe parmi les équations associées aux problèmes raides et il est préférable d'explorer des schèmes numériques implicites pour la solutionner.

Cependant, même avec une autre méthode que celle de  $DL$ , on reste au prise avec

le problème du choix des conditions initiales, qui s'avère un obstacle beaucoup plus important que le taux de convergence vers les solutions stationnaires de 1.40. Il semble avantageux d'utiliser les composantes des OPIs de période- $(m - 1)$  comme conditions initiales. Évidemment, si le nombre d'OPIs de période- $m$  pour un système chaotique de dimension  $D$  est supérieur à  $2^D D!$  fois le nombre d'OPIs de période- $(m - 1)$ , alors ces périodes  $m$  ne pourront être toutes détectées. En fait, pour une détection complète des OPIs, on devrait toujours essayer plusieurs valeurs de  $\beta$  pour l'ensemble des conditions initiales. Si on obtient un nombre d'OPIs relativement constant pour quelques valeurs de  $\beta$ , on peut espérer que l'ensemble est complet. Si ce plateau n'est pas rencontré, alors on devrait ajouter d'autres conditions initiales, choisies soit aléatoirement sur l'attracteur soit parmi les composantes des périodes  $(m - 2), (m - 3), \dots$

En fait, le choix des conditions initiales est un problème majeur, problème auquel nous n'avons pu répondre complètement. Avec la dimension du système qui augmente, ce point devient important étant donné la taille limitée des bassins d'attraction des différentes OPIs. Aussi, la quantité de matrice  $\mathbf{C}_k$  peut devenir considérable avec  $D$  qui augmente. Pourrait-on restreindre davantage ce groupe de matrices ? Cette dernière question, tout comme celle des conditions initiales, devraient être considérées en premier lieu afin améliorer la méthode que nous avons présentée.

# Chapitre 4

## Ciblage à l'aveuglette

La plupart des méthodes de contrôle du chaos demande de connaître la position de l'OPI à stabiliser ainsi que sa matrice jacobienne. La localisation de l'OPI peut se faire de différentes façons, notamment par les algorithmes de *SD* ou *DL* présentés au chapitre 1. Pour certains systèmes chaotiques, on doit ajouter un algorithme de ciblage (*targeting*) au contrôle afin de les amener plus rapidement vers l'OPI à stabiliser. Nous présentons dans ce chapitre une nouvelle approche qui fusionne ciblage et contrôle. Cette approche utilise les propriétés de convergence globale de l'algorithme *SD*. De plus, la méthode que nous présentons ne demande pas de connaître l'OPI à contrôler. Elle permet ainsi d'amener un système évoluant de façon chaotique sur une OPI qui n'est *pas* connue *a priori*. Nous appelons donc cette procédure : *ciblage à l'aveuglette*.

## 4.1 La méthode du ciblage à l'aveuglette

### 4.1.1 Utilité d'une méthode de ciblage en général

La majorité des méthodes de contrôle du chaos publiées depuis le début des années '90 suivent le même grand principe général : elles appliquent de petites perturbations à un paramètre du système, paramètre qui doit être accessible à l'expérimentateur ([67], [46], [77],[12],[27],[28]). Ces perturbations sont choisies judicieusement et appliquées à des moments stratégiques afin de maintenir le système chaotique près d'une de ses orbites périodiques. Afin d'éviter les interventions brusques qui pourraient modifier complètement le régime dynamique, on limite la grandeur des perturbations appliquées (celle-ci est généralement proportionnelle à la distance entre l'état du système et l'OPI). En pratique, on doit donc attendre que le système entre de lui-même dans le voisinage de l'OPI avant d'intervenir.

L'ergodicité des systèmes chaotiques nous assure que l'OPI sera visitée à un moment donné [96]. Toutefois, pour certains systèmes, le temps d'attente peut devenir trop long pour que la procédure de contrôle s'enclenche efficacement. Comme nous en avons déjà discuté au chapitre 1, les systèmes conservatifs présentent parfois un temps d'attente particulièrement long. Le *ciblage* réfère donc à une méthode particulière qui a pour but de diminuer les temps de transition vers les OPIs, en forçant le système à se diriger vers les régions d'intérêt sur l'attracteur. La technique que nous présentons s'ajoute ainsi aux méthodes déjà publiées dans les dernières années ([95],[92],[93],[94],[51]).

### 4.1.2 Description de l'algorithme du ciblage à l'aveuglette

Soit un système dynamique en dimension  $D$ , donné par l'application suivante :

$$\mathbf{v}_{n+1} = \mathbf{F}(\mathbf{v}_n, \mathbf{p}) . \quad (4.1)$$

Le symbole  $\mathbf{p}$  représente les différents paramètres accessibles du système. L'application dépend directement de ces paramètres et c'est pourquoi nous incluons une dépendance en  $\mathbf{p}$  dans la fonction  $\mathbf{F}$ . Il est important de noter qu'il doit y avoir *au moins* un paramètre  $p$  accessible, car nous prévoyons intervenir sur le système en programmant de petites perturbations  $\delta\mathbf{p}$  autour de la valeur nominale  $\mathbf{p}_0$  de ce paramètre. Nous gardons cependant l'éventuelle possibilité d'avoir plusieurs paramètres accessibles et notons l'ensemble de ces paramètres par le vecteur  $\mathbf{p}$ . Nous supposons que le système évolue dans un régime chaotique pour  $\mathbf{p} \approx \mathbf{p}_0$ . Ce régime offre un nombre élevé d'OPIs (un ensemble infini) qui peuvent en principe être toutes stabilisées.

Afin d'amener le système chaotique d'une position quelconque de l'espace des phases vers une de ses OPIs, nous allons lui prescrire de façon régulière des *cibles locales* qui seront choisies judicieusement. Nous appliquerons les perturbations  $\delta\mathbf{p}$  afin de "viser" ces *cibles locales*, sans trop se soucier de l'endroit où nous allons aboutir ! Le choix des *cibles locales* est évidemment la clé de la réussite de cette procédure puisqu'elles doivent conduire à coup sûr vers une OPI du système.

Les cibles en question sont obtenues par la transformation proposée par Schelcher et Diakonos et présentée au chapitre 1 (équation 1.29). Rappelons que cette transformation permet de localiser les orbites de période- $m$  de  $\mathbf{F}$ . Elle est donnée par :

$$\mathcal{W}_k : \mathbf{w}_{i+1} = \mathbf{w}_i + \lambda \mathbf{C}_k [\mathbf{F}^{(m)}(\mathbf{w}_i, \mathbf{p}_0) - \mathbf{w}_i]. \quad (4.2)$$

Cette dernière équation est presque identique à l'équation 1.29. Nous avons cependant ajouté la dépendance de  $\mathbf{F}$  aux paramètres externes  $\mathbf{p}$ . Ce schème itératif est utilisé pour repérer les orbites de période- $m$  de  $\mathbf{F}$  évoluant sous les paramètres nominaux  $\mathbf{p}_0$ . Rappelons que  $0 < \lambda \ll 1$  et que  $\mathbf{C}_k$  est une matrice  $D \times D$  orthogonale, avec des éléments  $c_{ij} \in \{0, \pm 1\}$  et un seul élément par ligne ou par colonne est différent de 0. En  $D$ -dimensions, il existe au maximum  $2^D D!$  de ces matrices. Leur choix influence le type d'orbite périodique qui peut être repéré (réf. tableau 1.3). La valeur de  $\lambda$  détermine le degré maximal d'instabilité de l'orbite qui peut être localisée (réf. section 1.3.2.2).

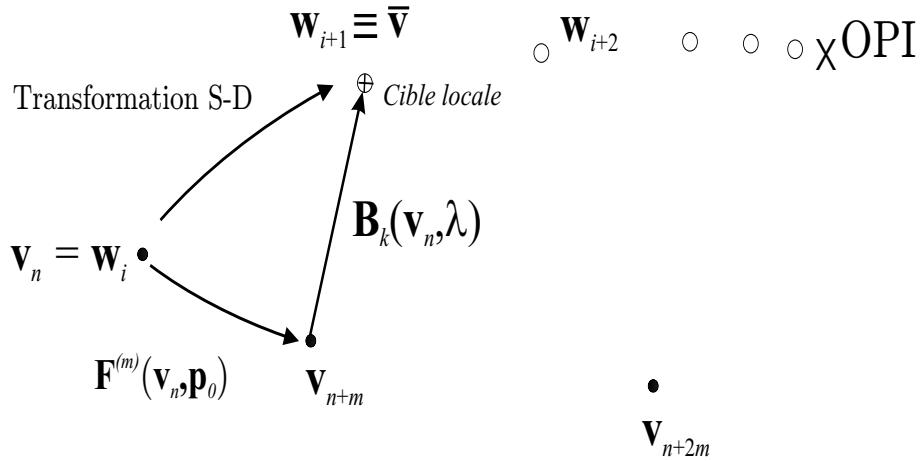
Puisque l'équation 4.2 peut éventuellement conduire à une orbite de période- $m$  de  $\mathbf{F}$ , il semble justifié de choisir comme *cible locale* la position  $\mathbf{w}_{i+1}$ , qui correspond à la position du système après une itération dans l'espace  $\mathcal{W}_k$ . Ainsi, pour une position  $\mathbf{v}_n$  dans l'espace des phases, la *cible locale* (symbolisée par  $\bar{\mathbf{v}}$ ) est donnée par :

$$\bar{\mathbf{v}} \equiv \mathbf{w}_{i+1} = \mathbf{v}_n + \lambda \mathbf{C}_k [\mathbf{F}^{(m)}(\mathbf{v}_n, \mathbf{p}_0) - \mathbf{v}_n]. \quad (4.3)$$

Nous schématisons ces différents éléments sur la figure 4.1, qui représente l'évolution du système  $\mathbf{F}(\mathbf{v}_n, \mathbf{p}_0)$  dans l'espace des phases (trajectoire représentée par les symboles  $\bullet$ ). On retrouve aussi sur cette figure la trajectoire que suivrait le système dans l'espace  $\mathcal{W}_k$  si on itérait l'équation 4.2 à partir de  $\mathbf{w}_i = \mathbf{v}_n$  (symboles  $\diamond$  sur la figure). Cette dernière trajectoire peut conduire à une orbite périodique. La *cible locale* que nous choisissons ( $\bar{\mathbf{v}}$ ) est le point  $\mathbf{w}_{i+1}$ , qui correspond à la position après un pas dans le système transformé. On utilise des indices différents ( $n$  et  $i$ ) car ils s'appliquent à deux systèmes différents (resp.  $\mathcal{V}$  et  $\mathcal{W}_k$ ).

Afin d'aboutir sur une période- $m$ , nous devons ajouter, pour les  $m$  prochaines itérations, une perturbation  $\delta\mathbf{p}$  aux paramètres  $\mathbf{p} = \mathbf{p}_0$  pour faire en sorte que  $\mathbf{v}_{n+m}$  tombe sur (ou le plus près possible de)  $\bar{\mathbf{v}}$ . À chaque  $m$  itérations, il faut recalculer la position de la cible  $\bar{\mathbf{v}}$  au moyen de l'équation 4.3 et choisir à nouveau la perturbation  $\delta\mathbf{p}$ .

Nous allons maintenant obtenir une expression quantitative pour la perturbation  $\delta\mathbf{p}$ . Deux cas types doivent cependant être distingués. Le premier est celui où le nombre de paramètres accessibles est au moins égal à la dimension du système ; on peut ainsi exiger que l'itéré  $\mathbf{v}_{n+m}$  tombe directement sur la *cible locale*  $\bar{\mathbf{v}}$ . Le deuxième est celui où le nombre de paramètres accessibles est inférieur à la dimension du système ; on impose alors à l'itéré  $\mathbf{v}_{n+m}$  de tomber le plus près possible de  $\bar{\mathbf{v}}$  et le ciblage peut se faire à un seul paramètre.

FIG. 4.1 – Évolutions de  $\mathbf{v}_n$  (libre et sous la transformation de SD)

#### 4.1.2.1 Ciblage à plusieurs paramètres

Nous supposons ici que le nombre de paramètres accessibles est plus grand ou égal à la dimension  $D$  du système. Pour de faibles perturbations  $\delta\mathbf{p}$ , on obtient la position de  $\mathbf{v}_{n+m}$  en linéarisant l'équation 4.1 autour de  $\mathbf{v}_n$  et de  $\mathbf{p}_0$  :

$$\mathbf{v}_{n+m}(\mathbf{p}_0 + \delta\mathbf{p}) = \mathbf{F}^{(m)}(\mathbf{v}_n, \mathbf{p}_0 + \delta\mathbf{p}) \sim \mathbf{F}^{(m)}(\mathbf{v}_n, \mathbf{p}_0) + \sum_{i=1}^D \mathbf{V}_n^{(i)} \delta p^{(i)}, \quad (4.4)$$

où le vecteur  $\mathbf{V}_n^{(i)}$  correspond à la variation de l'application  $\mathbf{F}^{(m)}$  par rapport au  $i^{ième}$  paramètre au point  $\mathbf{v}_n$  i.e.  $\mathbf{V}_n^{(i)} \equiv D_{p^{(i)}} \mathbf{F}^{(m)}(\mathbf{v}_n, \mathbf{p}_0)$ . Pour écrire la linéarisation, nous utilisons  $D$  paramètres accessibles et  $p^{(i)}$  correspond au  $i^{ième}$  paramètre utilisé. Le symbole  $\delta p^{(i)}$  est donc la perturbation appliquée au paramètre  $p^{(i)}$  pour  $m$  itérations.

Le critère de ciblage que nous pouvons utiliser ici est  $\|\bar{\mathbf{v}} - \mathbf{v}_{n+m}(\mathbf{p}_0 + \delta\mathbf{p})\| = 0$ . En effet, comme il y au moins  $D$  paramètres accessibles, on peut en général tomber sur le point  $\bar{\mathbf{v}}$  (qui a  $D$  coordonnées). Chaque perturbation  $\delta p^{(i)}$  déplace le point  $\mathbf{v}_{n+m}$  dans la direction du vecteur  $\mathbf{V}_n^{(i)}$ . Si on possède  $D$  vecteurs  $\mathbf{V}^{(i)}$  indépendants, on peut toujours trouver une combinaison de  $\delta\mathbf{p}$  qui déplace le point  $\mathbf{v}_{n+m}$  de sa position normale (i.e. sous évolution libre) sur la cible locale  $\bar{\mathbf{v}}$ . Le choix de la perturbation  $\delta\mathbf{p}$  s'obtient donc

en imposant :

$$\|\bar{\mathbf{v}} - \mathbf{v}_{n+m}(\mathbf{p}_0 + \boldsymbol{\delta p})\| = 0. \quad (4.5)$$

En utilisant les équations 4.3 et 4.4, cette dernière équation devient :

$$\|\bar{\mathbf{v}} - \mathbf{v}_{n+m}(\mathbf{p}_0 + \boldsymbol{\delta p})\| \sim \|\mathbf{B}_k(\mathbf{v}_n, \lambda) - \sum_{i=1}^D \mathbf{V}_n^{(i)} \delta p^{(i)}\| = 0$$

(4.6)

avec  $\mathbf{B}_k(\mathbf{v}_n, \lambda) \equiv [\lambda \mathbf{C}_k - \mathbf{1}] [\mathbf{F}^{(m)}(\mathbf{v}_n, p_0) - \mathbf{v}_n]$ . Le vecteur  $\mathbf{B}_k$  dépend du choix de  $\mathbf{C}_k$  et de  $\lambda$ , ces deux derniers paramètres provenant de la transformation de Schmelcher-Diakonos. Ce vecteur correspond en fait à la différence entre deux positions soit la *cible locale* ( $\bar{\mathbf{v}}$ ) et l'état du système après  $m$  évolutions libres à partir de  $\mathbf{v}_n$  (i.e.  $\mathbf{F}^{(m)}(\mathbf{v}_n, \mathbf{p}_0)$ ). Nous l'avons aussi représenté sur la figure 4.1.

En principe, il est possible de trouver une solution pour les  $\delta p^{(i)}$  qui satisfait l'équation 4.6. Il s'agit en effet d'un système linéaire à  $D$  équations,  $D$  inconnus. Si *tous* les  $\delta p^{(i)}$  obtenus sont inférieurs aux valeurs maximales applicables  $\delta p_{max}^{(i)}$ , on applique la perturbation  $\boldsymbol{\delta p}$  pour les  $m$  prochaines itérations. Sinon, on laisse le système libre pour une itération et on reprend le calcul du  $\boldsymbol{\delta p}$  pour une nouvelle *cible locale*.

#### 4.1.2.2 Ciblage à un paramètre

Lorsque le nombre de paramètres accessibles est inférieur à la dimension  $D$  du système, l'équation 4.6 n'a pas de solution en général. Nous allons ainsi "adoucir" le critère de ciblage (équation 4.5) afin de pouvoir appliquer la méthode en tout temps. Nous imposons maintenant que l'itéré  $\mathbf{v}_{n+m}$  tombe le plus près possible de la *cible locale*  $\bar{\mathbf{v}}$ . Aussi, afin d'obtenir une solution simple pour ce nouveau critère, nous utilisons un seul paramètre de contrôle. En linéarisant l'équation 4.1 autour de  $\mathbf{v}_n$  et de  $p_0$  (i.e. avec seul paramètre accessible), on obtient :

$$\mathbf{v}_{n+m}(p_0 + \delta p) = \mathbf{F}^{(m)}(\mathbf{v}_n, p_0 + \delta p) \sim \mathbf{F}^{(m)}(\mathbf{v}_n, p_0) + \mathbf{V}_n \delta p. \quad (4.7)$$

On laisse tomber l'indice  $(i)$  sur le vecteur  $\mathbf{V}_n \equiv D_p \mathbf{F}^{(m)}(\mathbf{v}_n, p_0)$  ainsi que sur la perturbation  $\delta p$  puisqu'il n'y a plus qu'un paramètre accessible. Notre critère de ciblage combiné à cette dernière linéarisation conduit alors à :

$$\|\bar{\mathbf{v}} - \mathbf{v}_{n+m}(p_0 + \delta p)\| \sim \|\mathbf{B}_k(\mathbf{v}_n, \lambda) - \mathbf{V}_n \delta p\| = \min \quad . \quad (4.8)$$

En minimisant cette équation par rapport à  $\delta p$ , on obtient directement une expression pour la perturbation à appliquer :

$$\boxed{\delta p = \mathbf{B}_k(\mathbf{v}_n, \lambda) \cdot \mathbf{V}_n / \|V_n\|^2} \quad . \quad (4.9)$$

Encore une fois, cette dernière perturbation sera appliquée pour les  $m$  prochaines itérations si elle est inférieure à une valeur maximale admissible  $\delta p_{max}$ . Sinon, on laisse le système évoluer librement pour une itération et on calcule à nouveau la perturbation pour une nouvelle cible locale.

Il est intéressant de noter que l'expression obtenue pour la perturbation  $\delta p$  dans le cas du ciblage à un paramètre se ramène à des expressions connues pour le contrôle lorsque le point  $\mathbf{v}_n$  est dans le voisinage d'une OPI. En une dimension, lorsque le système est dans le voisinage d'une OPI, la perturbation 4.9 est identique à celle prescrite par la méthode connue sous l'acronyme *OPF* (Occasional Proportional Feedback [46]). Pour des dimensions supérieures à 1, la perturbation 4.9 est similaire à celle prescrite par le contrôle de type MED [79] une fois dans le voisinage de l'OPI (MED : Minimal Expected Deviation). Le ciblage à l'aveuglette est en quelque sorte une extension de certaines méthodes de contrôle publiées dans les dernières années ([67], [26], [84], [46], [77], [71], [83], [63]).

Il faut aussi mentionner que les expressions 4.6 et 4.9 pour les perturbations contiennent une quantité qui en principe n'est pas connue par l'expérimentateur. Dans le vecteur  $\mathbf{B}_k(\mathbf{v}_n, \lambda)$  se cache l'expression  $\mathbf{F}^{(m)}(\mathbf{v}_n, \mathbf{p}_0)$ , qui n'est pas connue exactement dans un contexte expérimental. Il s'agit en effet de la position du système dans le futur.

Nous traiterons de cette particularité un peu plus loin dans ce chapitre. Pour l'instant, mentionnons qu'il sera possible d'estimer  $\mathbf{F}^{(m)}(\mathbf{v}_n, \mathbf{p}_0)$  à l'aide d'un prédicteur construit à partir d'une série temporelle préalablement enregistrée.

En terminant, il faut souligner que cette méthode de ciblage peut aussi bien nous conduire sur des orbites périodiques stables ou instables. En effet, nos *cibles locales* sont obtenues à partir de la transformation de Schmelcher-Diakonos, qui elle, peut aboutir sur toutes orbites périodiques. Dans le contexte des systèmes Hamiltoniens (avec une dynamique mixte [56]), on peut donc espérer accéder à des régions de régularité, où se cachent les orbites stables (elliptiques), à partir des bandes stochastiques. Cette approche pourrait ainsi permettre de traverser les barrières entre les zones régulières et stochastiques (tores de KAM).

#### 4.1.3 Résumé de l'algorithme

Nous résumons dans le tableau 4.1 l'algorithme pour le ciblage à l'aveuglette d'une orbite de période- $m$ . En fonction du système chaotique en jeu, on peut choisir entre un ciblage à 1 ou  $D$  paramètres.

## 4.2 Quelques résultats

Dans cette section, nous présentons des scénarios de ciblage obtenus numériquement sur une vaste gamme de systèmes dynamiques. Nous voulons ainsi faire ressortir le côté général de la méthode. Les exemples choisis pourront ainsi varier entre des applications discrètes ou des flots continus, tant dissipatifs que conservatifs. Tous les systèmes chaotiques que l'on retrouve dans cette section ont été présentés au chapitre 1.

**0.** Choisir un  $\mathbf{p}_0$  pour lequel le système à  $D$  dimensions  $\mathbf{F}$  est chaotique.

**1.** Choisir une condition initiale  $\mathbf{v}_0$

**2.** Choisir une valeur de  $\lambda$

**3.** Choisir une matrice  $\mathbf{C}_k$

**4.** Faire jusqu'à ce qu'une OPI soit stabilisée

**i)** Calculer la ou les perturbations :

Pour 1 paramètre :

À partir de  $\delta p = \mathbf{B}_k(\mathbf{v}_n, \lambda) \cdot \mathbf{V}_n / \|V_n\|^2$  (équation 4.9)

Pour  $D$  paramètres :

En solutionnant  $\|\mathbf{B}_k(\mathbf{v}_n, \lambda) - \sum_{i=1}^D \mathbf{V}_n^{(i)} \delta p_i\| = 0$  (équation 4.6)

**ii)** IF ( $\delta p_i < \delta p_{i_{max}}$ ) pour tout  $i$  :

⇒ Appliquer les  $\delta p_i$  pour  $m$  itérations. GOTO **4i**)

ELSE :

⇒ Faire une itération. GOTO **4i**)

TAB. 4.1 – Résumé de l'algorithme du ciblage à l'aveuglette

### 4.2.1 Ciblage à $D$ paramètres

Les premiers scénarios se font sur une application connue analytiquement (système Ikéda), semi-analytique (*Billards Magnétiques*) et un flot continu (système de Lorenz). Pour l'ensemble de ces simulations, on calcule la perturbation selon l'équation 4.6 avec  $\mathbf{C}_k = \mathbf{C}_1$  et  $\lambda = 0.05$ .

La figure 4.2 montre un scénario de ciblage avec les paramètres  $B$  et  $\eta$  de l'application Ikéda ( $\delta\mathbf{p}_{max} = 0.01\mathbf{p}_0$ ). On demande de stabiliser une période 1 et 2 pendant  $10^4$  itérations. Avant d'enclencher le contrôle, on laisse le système évoluer librement pendant  $10^4$  itérations. Les bandes blanches et noires au bas de la figure indiquent si le système évolue librement ou s'il subit des perturbations. La position et la stabilité des OPIs ciblées sur Ikéda sont indiquées sur la figure 4.3 (Le symbole  $\rho_u$  réfère à la plus grande valeur propre (en valeur absolue) de la matrice jacobienne de l'OPI).

Un scénario différent est présenté à la figure 4.4 où cette fois, on essaie de stabiliser  $10^4$  itérations d'une période 3 et d'une période 4 dans un *Billard Magnétique Ovalé*. Les perturbations sont appliquées sur  $e$  (la forme de la cavité) et  $B$  (champ magnétique) ( $\delta\mathbf{p}_{max} = 0.05\mathbf{p}_0$ ). À la figure 4.5, on indique la position et la stabilité des orbites ciblées.

Le dernier scénario est illustré sur la figure 4.6. Ici, on demande le contrôle de période 1 et 2 sur l'attracteur de Lorenz. Les paramètres  $B$  et  $R$  sont utilisés pour appliquer les perturbations ( $\delta\mathbf{p}_{max} = 0.05\mathbf{p}_0$ ). On obtient une application discrète à partir du flot de Lorenz en choisissant les temps  $\tau_n$  tels que  $x(\tau_n) \equiv 0$  et  $\frac{dx}{dt}\Big|_{\tau_n} > 0$ . La position et la stabilité des OPIs ciblées pour Lorenz sont données à la figure 4.7.

Le tableau 4.2 regroupe l'ensemble des informations nécessaires pour reproduire ces différents scénarios. Pour chacun d'entre eux, on indique par une flèche ( $\Rightarrow$ ) les paramètres utilisés pour le ciblage. Pour tous les scénarios, nous présentons deux figures.

La première illustre une des variables en fonction du nombre d'itérations (par exemple fig. 4.2) et la seconde donne la position, dans l'espace des phases, des OPIs qui ont été stabilisées (exemple fig. 4.3).

Scénario	Nom du système	Itérations à contrôler	Itérations libres	Paramètres nominaux	$\delta p_{max}$	Conditions initiales
Périodes 1 et 2 (fig. 4.2 et 4.3)	<i>Ikéda</i> $(D = 2)$	$10^4$	$10^4$	$\Rightarrow B = 0.9$ $\Rightarrow \eta = 6.0$ $k = 0.4$	1% de $\mathbf{p}_0$	$x_0 = 0.5$ $y_0 = 0.2$
Périodes 3 et 4 (fig. 4.4 et 4.5)	<i>Billards Magnétiques</i> $(D = 2)$	$10^4$	$10^4$	$\Rightarrow e = 0.15$ $\Rightarrow B = 0.25$	5% de $\mathbf{p}_0$	$\varphi_0 = 1.8$ $\alpha_0 = 0.2$
Périodes 1 et 2 (fig. 4.6 et 4.7)	<i>Lorenz</i> $(D = 2)$	$10^4$	$10^4$	$\Rightarrow B = 4.0$ $\Rightarrow R = 45.92$ $\sigma = 16$	5% de $\mathbf{p}_0$	$x_0 = 1.0$ $y_0 = 1.0$ $z_0 = 1.0$ $x(\tau_n) \equiv 0$

TAB. 4.2 – Résumé des scénarios de ciblage à  $D$  paramètres

### 4.2.2 Ciblage à 1 paramètre

Nous présentons ici des exemples où le ciblage se fait à 1 paramètre. L'équation 4.9 est maintenant celle qui dicte la perturbation à appliquer (avec  $\mathbf{C}_k = \mathbf{C}_1$  et  $\lambda = 0.05$ ). L'ensemble des paramètres utilisés pour les simulations sont regroupés le tableau 4.3

Le premier scénario est la stabilisation des périodes 1, 2, 4 et 7 dans Hénon. Le paramètre  $A$  est utilisé pour le contrôle ( $\delta p_{max} = 0.05A$ ). La figure 4.8 illustre cette première simulation et la figure 4.9 donne la position et la stabilité des OPIs ciblées dans Hénon.

La deuxième simulation (figures 4.10 et 4.11) est appliquée sur le *Billard Oval*

*Libre.* Le ciblage est effectué en modifiant la forme de la cavité (paramètre  $e$ ). Cette fois, on demande le contrôle d'une période 3 et 4. On remarque que le système aboutit sur une orbite *stable*; lorsqu'on enlève le contrôle à la fin du scénario, le système reste sur l'orbite de période-4. Le billard est donc passé de la région chaotique (la période-3 stabilisée étant située dans cette région) à une bande régulière. L'orbite de période-4 (sur laquelle le système termine sa course) est localisée dans un îlot de stabilité.

Un scénario différent est ensuite essayé sur le système DKP (figures 4.12 et 4.13). Il faut maintenant stabiliser successivement une période-2 et une période-3 en modifiant le paramètre  $\epsilon$  de l'Hamiltonien. Le flot DKP est discrétisé en choisissant les temps  $\tau_n$  tels que  $\mu(\tau_n) = 0$ ,  $\frac{d\mu}{dt}\Big|_{\tau_n} > 0$  et en ajustant la variable  $p_\mu$  pour respecter la contrainte sur l'énergie, comme expliqué au chapitre 2. Une fois encore, le système traverse de nouveau dans un îlot de régularité; l'orbite de période-3 ciblée est stable.

La dernière simulation est plus audacieuse, on applique le ciblage à l'aveuglette sur une dynamique avec une dimension plus élevée que celles des systèmes présentés jusqu'à maintenant. On demande ainsi de cibler une période-1 dans le Pendule Double (Dimension de Lyapunov = 2.8). Bien que le temps d'attente soit assez considérable, le ciblage à l'aveuglette arrive tout de même à stabiliser un des points fixes de ce pendule (figure 4.14). On remarque cependant que la technique ne parvient pas à amener le système précisément sur le point fixe. En effet, le “trait” qui indique que le système reste sur la même position entre  $n = 4 \cdot 10^5$  et  $5 \cdot 10^5$  est relativement large ( $\omega^1$  oscille entre 1.68 et 1.75). Pour les dimensions plus élevées, comme celle de Pendule Double, on est quelque peu limité avec le ciblage à un paramètre. Rappelons que le seul critère qui est appliqué pour prescrire les  $\delta p_n$  est de minimiser la distance entre les cibles locales  $\bar{\mathbf{v}}$  et le prochain itéré  $\mathbf{v}_n$ . Pour un point fixes en dimension 4 comme celui du Pendule Double, le critère ne permet pas de choisir convenablement  $\delta p_n$  pour tomber sur le point fixe. Il faudrait ainsi introduire trois autres paramètres externes (comme par exemple une force sur la 1<sup>ière</sup> tige et la possibilité de varier l'angle entre les forces appliquées et la verticale (réf. figure 2.5)). On aurait alors quatre paramètres et il serait

possible de tomber précisément sur la cible locale et éventuellement le point fixe.

Scénario	Nom du système	Itérations à contrôler	Itérations libres	Paramètres nominaux	$\delta p_{max}$	Conditions initiales
Périodes 1, 2 4 et 7 (fig. 4.8 et 4.9)	<i>Hénon</i> $(D = 2)$	$10^4$	$10^4$	$\Rightarrow A = 1.4$ $B = 0.3$	1% de $A$	$x_0 = 0.2$ $y_0 = 0.2$
Périodes 3 et 4 (fig. 4.10 et 4.11)	<i>Billards</i> <i>Libres</i> $(D = 2)$	$10^4$	$10^4$	$\Rightarrow e = 0.15$	5% de $e$	$\varphi_0 = 1.3$ $\alpha_0 = 1.2$
Périodes 2 et 3 (fig. 4.12 et 4.13)	<i>DKP</i> $(D = 2)$	$10^4$	$10^4$	$\Rightarrow \epsilon = -0.2$	0.5% de $\epsilon$	$\mu_0 = 0$ $\nu_0 = -0.8$ $p_\mu = \text{ajustée}$ $p_\nu = -1.6$ $(\mu(\tau_n) = 0)$
Période 1 (fig. 4.14 et 4.15)	<i>Pendule Double</i> $(D = 4)$	$10^5$	0	$\Rightarrow f_0 = 9.0$	2% de $f_0$	$\theta_0^1 = 1.0$ $\theta_0^2 = 1.0$ $\omega_0^1 = 0.1$ $\omega_0^2 = 0.1$

TAB. 4.3 – Résumé des scénarios de ciblage à 1 paramètre

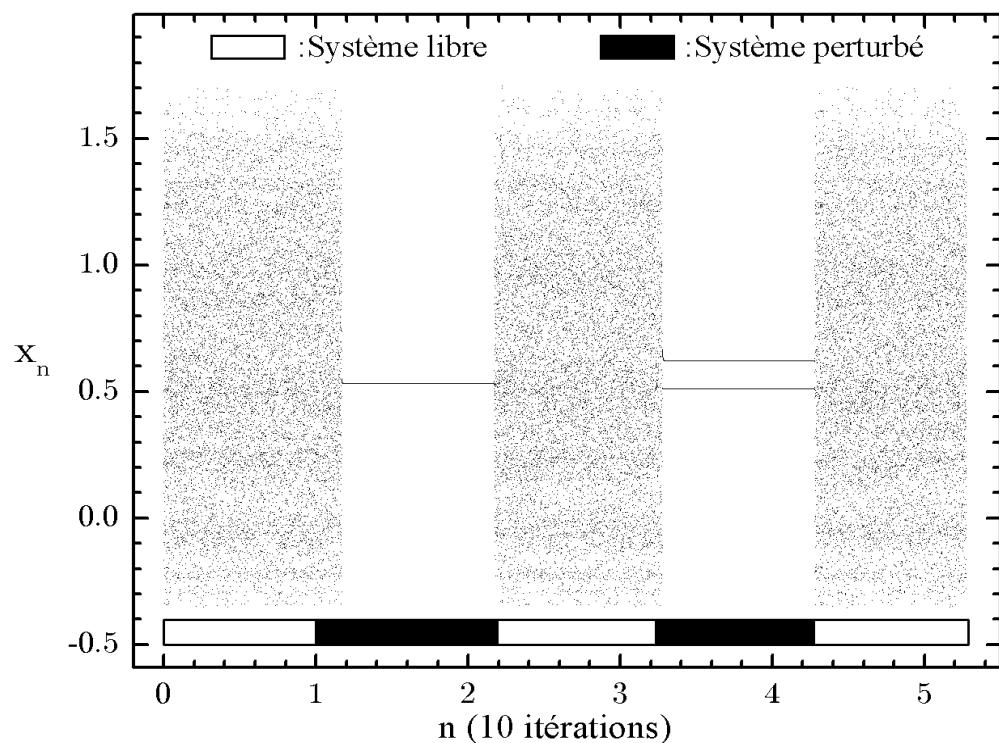


FIG. 4.2 – Ciblage de période 1 et 2 sur Ikéda

Le ciblage se fait en perturbant les paramètres  $B$  et  $\eta$  (1% max) du système Ikéda. Les bandes horizontales au bas de la figure indiquent une évolution libre du système (bandes blanches) ou contrainte à des perturbations (bandes noires).

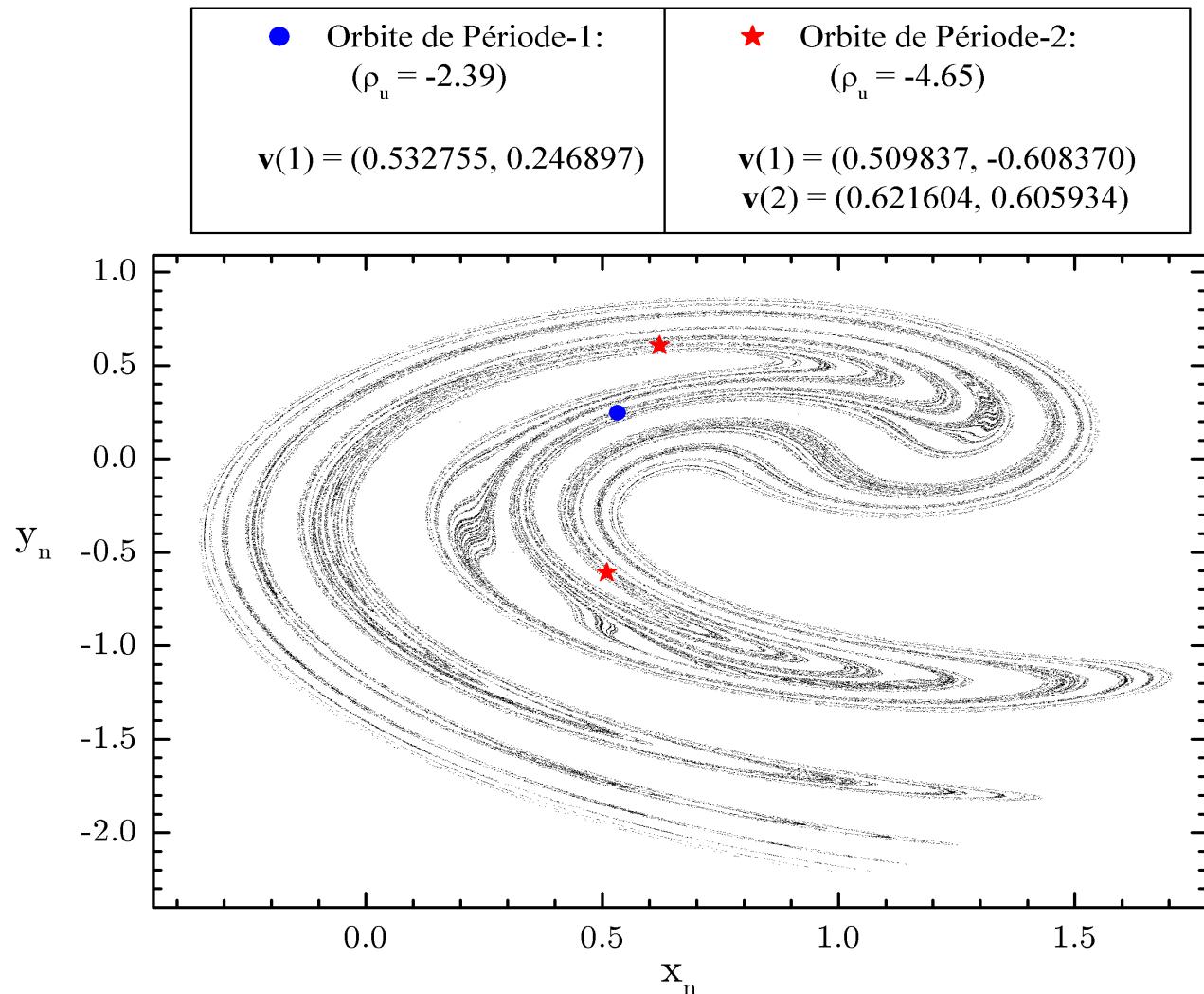
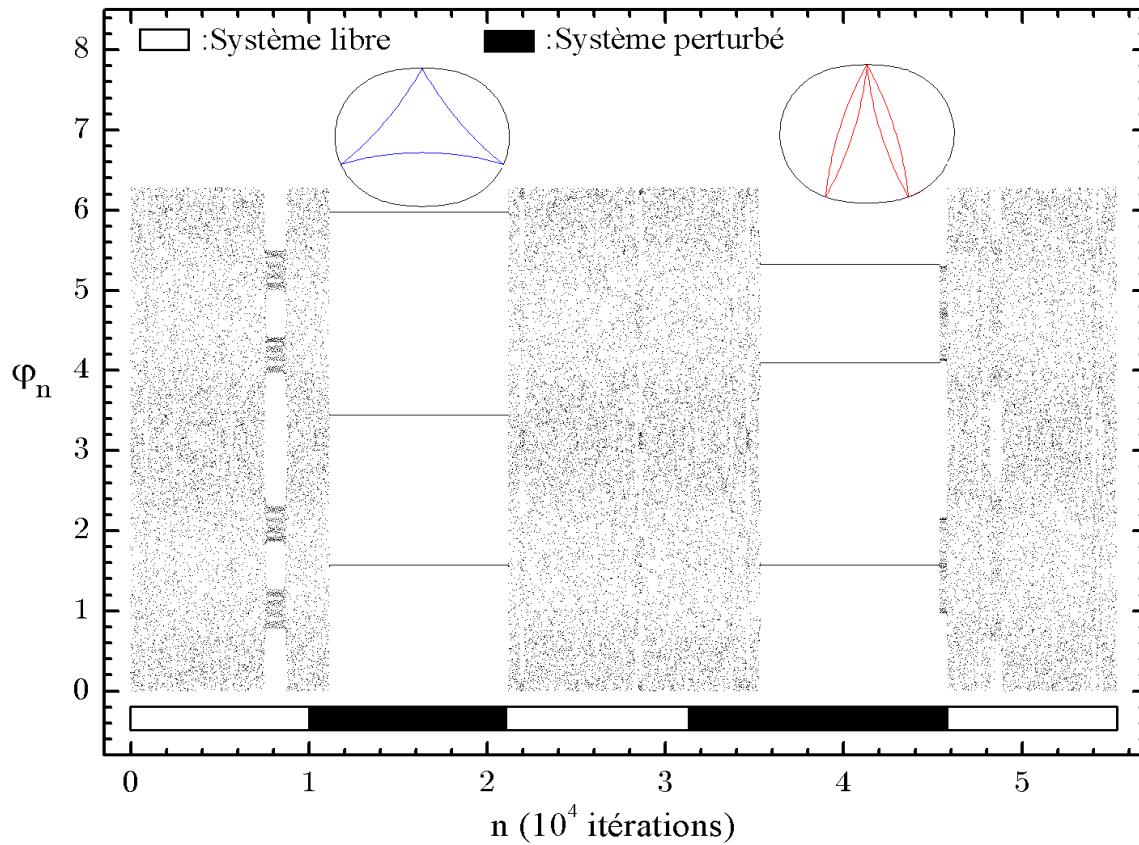
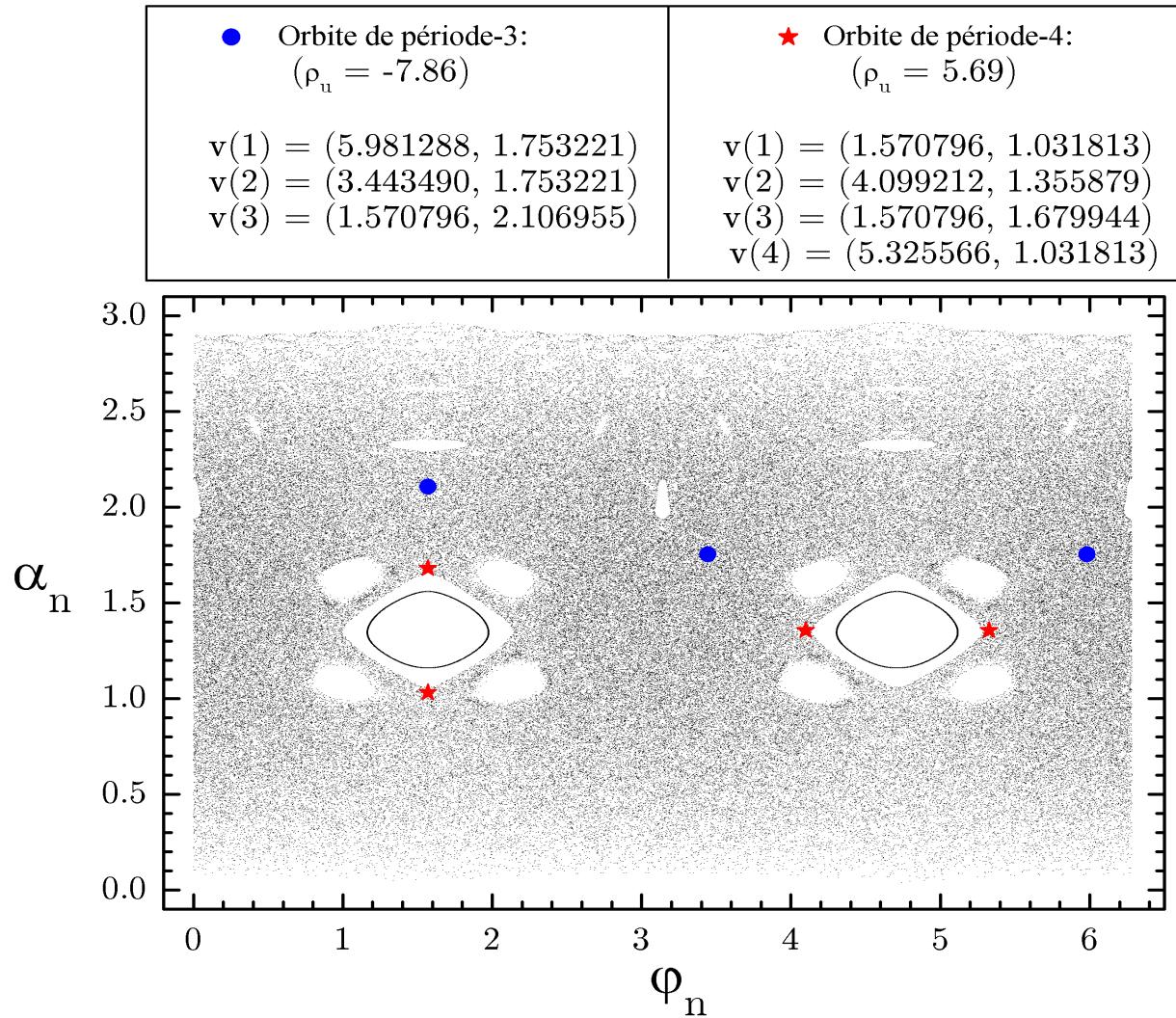


FIG. 4.3 – Orbites de période 1 et 2 sur Ikéda

Cette figure illustre la position des OPIs stabilisées par la technique du ciblage à l'aveuglette (réf. figure 4.2).

FIG. 4.4 – Ciblage de période 3 et 4 sur le *Billard Magnétique*

Le ciblage se fait en perturbant les paramètres  $e$  (forme de la cavité) et  $B$  (champ magnétique)(5% max). La forme des orbites de période 3 et 4 ainsi stabilisées est aussi représentée. On remarque un peu de “sticking” autour d’îlots de stabilité vers la 8000<sup>*ième*</sup> itération.

FIG. 4.5 – Orbites de période 3 et 4 dans un *Billard Magnétique*

Cette figure illustre le diagramme des phases du *Billard Magnétique* ainsi que la position des OPIs stabilisées par la technique du ciblage à l'aveuglette (réf. figure 4.4).

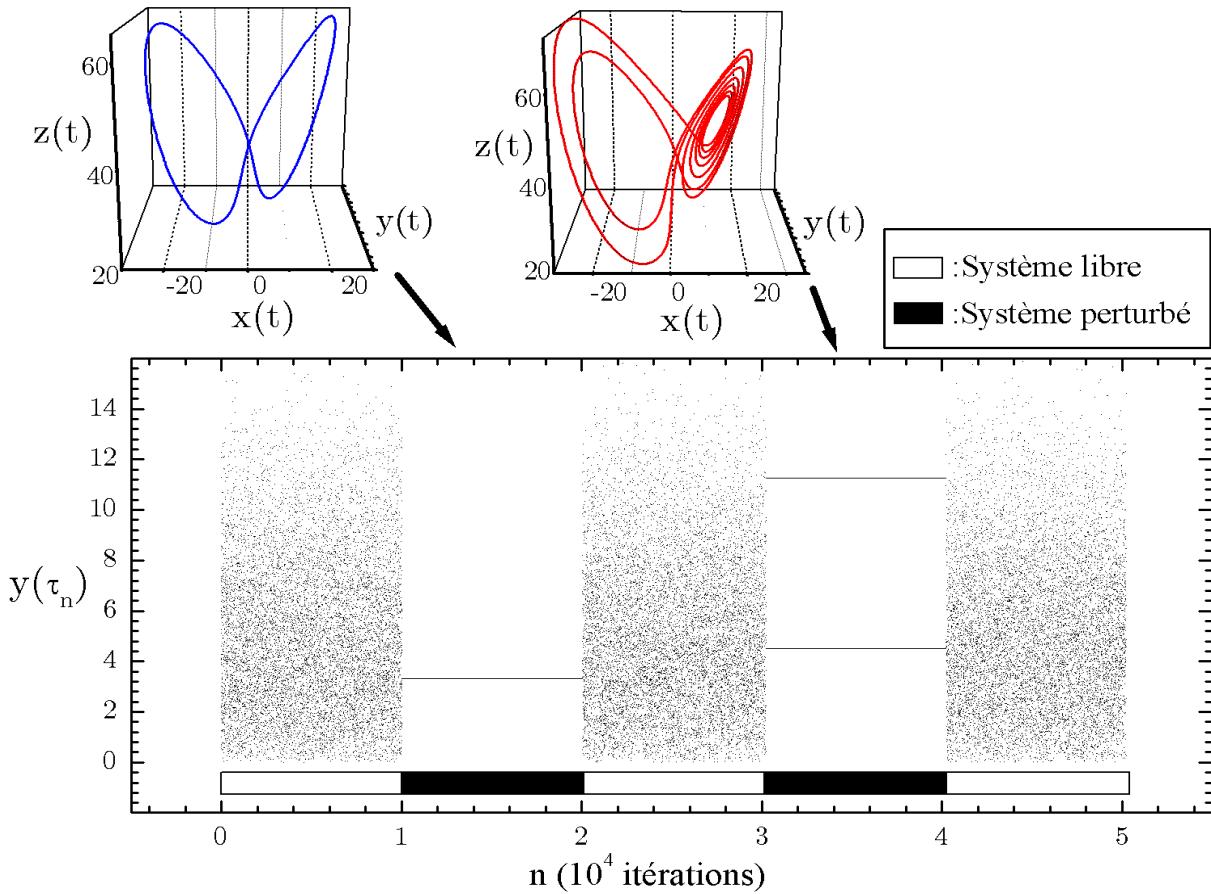


FIG. 4.6 – Ciblage de période 1 et 2 sur le système de Lorenz

Le ciblage se fait en perturbant les paramètres  $b$  et  $R$  (5% max). L’application discrète pour stabiliser le flot de Lorenz est obtenue en choisissant les temps  $\tau_n$  tels que  $x(\tau_n) = 0, \frac{dx}{dt}\Big|_{\tau_n} > 0$ . Les trajectoires ainsi contrôlées sont aussi illustrées.

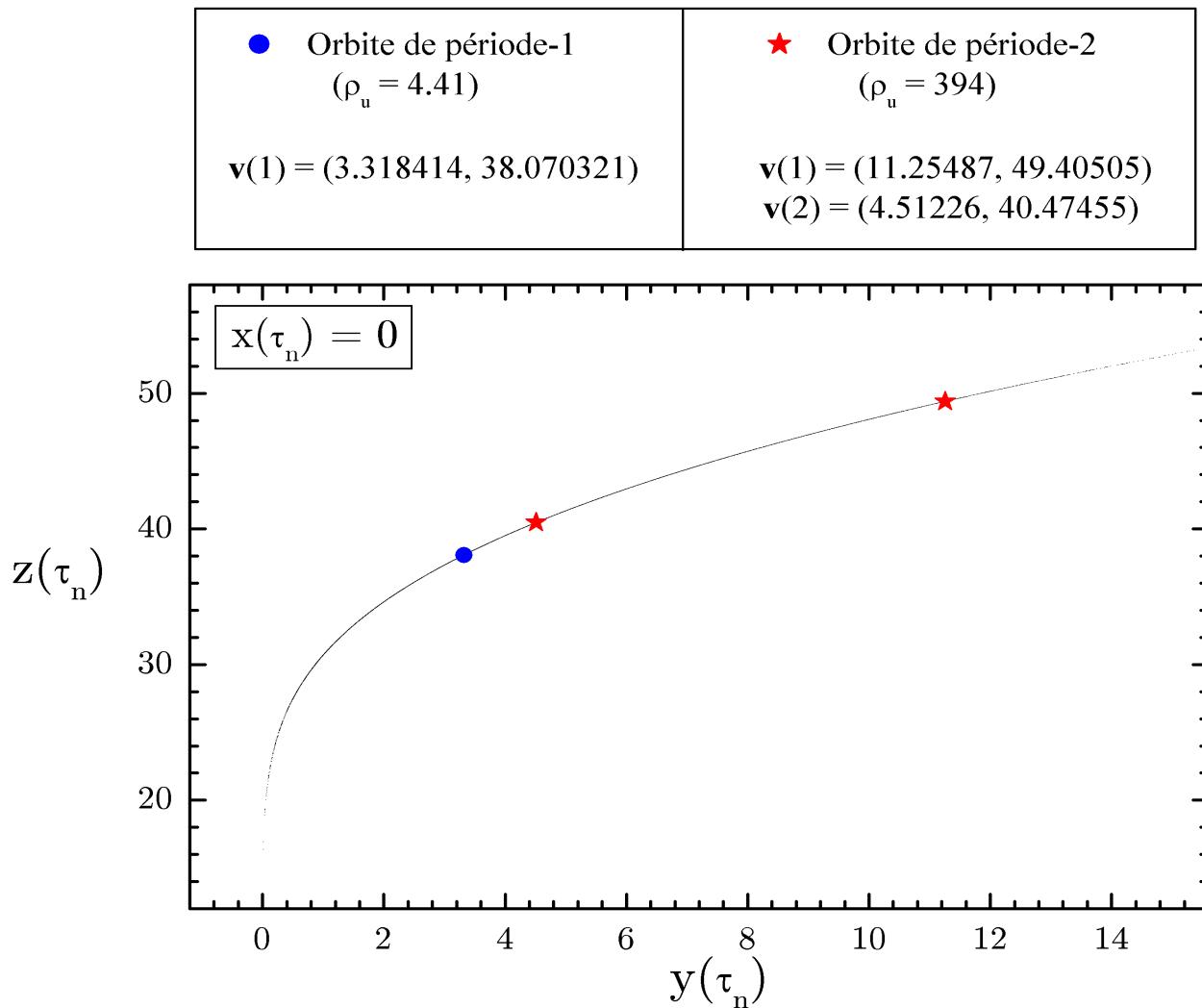


FIG. 4.7 – Orbites de période 3 et 4 de Lorenz

Cette figure illustre le diagramme des phases de l'attracteur de Lorenz sur la Section de Poincaré  $x(\tau_n) = 0$ ,  $\frac{dx}{dt}\Big|_{\tau_n} > 0$  ainsi que la position des OPIs stabilisées par la technique du ciblage à l'aveuglette (réf. figure 4.6).

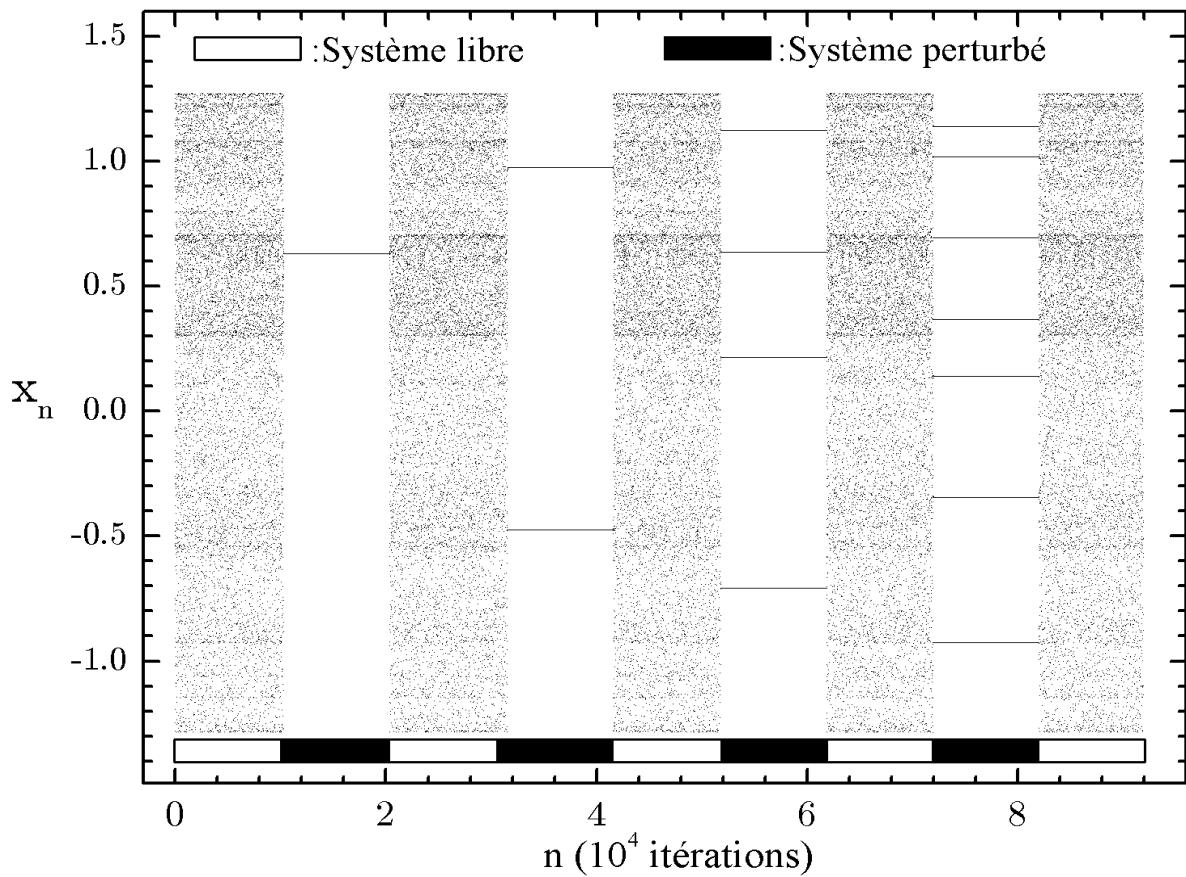


FIG. 4.8 – Ciblage de période 1, 2, 4 et 7 sur Hénon

Le ciblage se fait en perturbant le paramètre  $A$  (5% max) du système de Hénon.

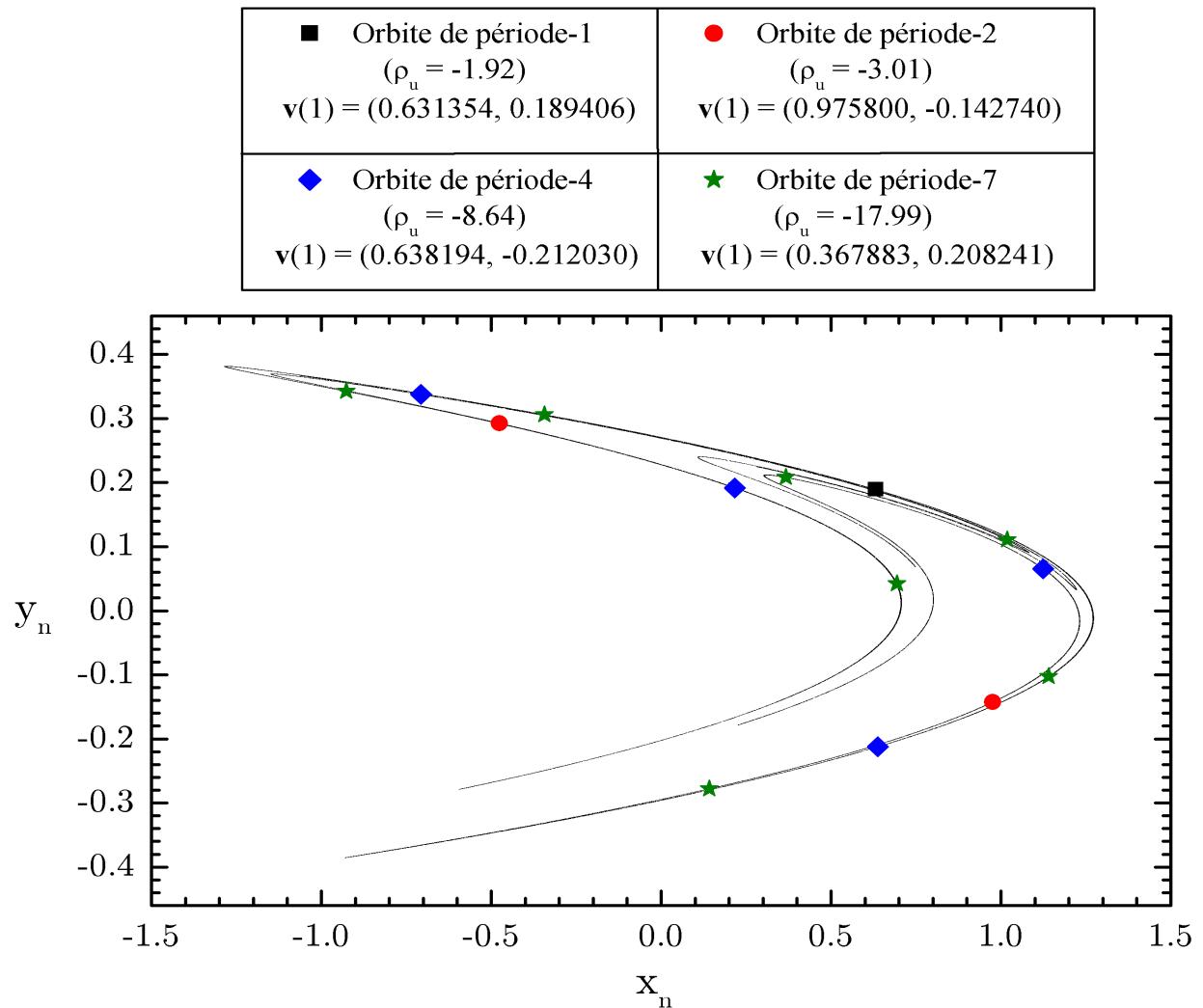


FIG. 4.9 – Orbites de période 1, 2, 4 et 7 dans le système de Hénon

Cette figure illustre l'attracteur de Hénon ainsi que la position des OPIs stabilisées par la technique du ciblage à l'aveuglette (réf. figure 4.8). La stabilité et la position de la première composante pour chaque OPI sont aussi indiquées.

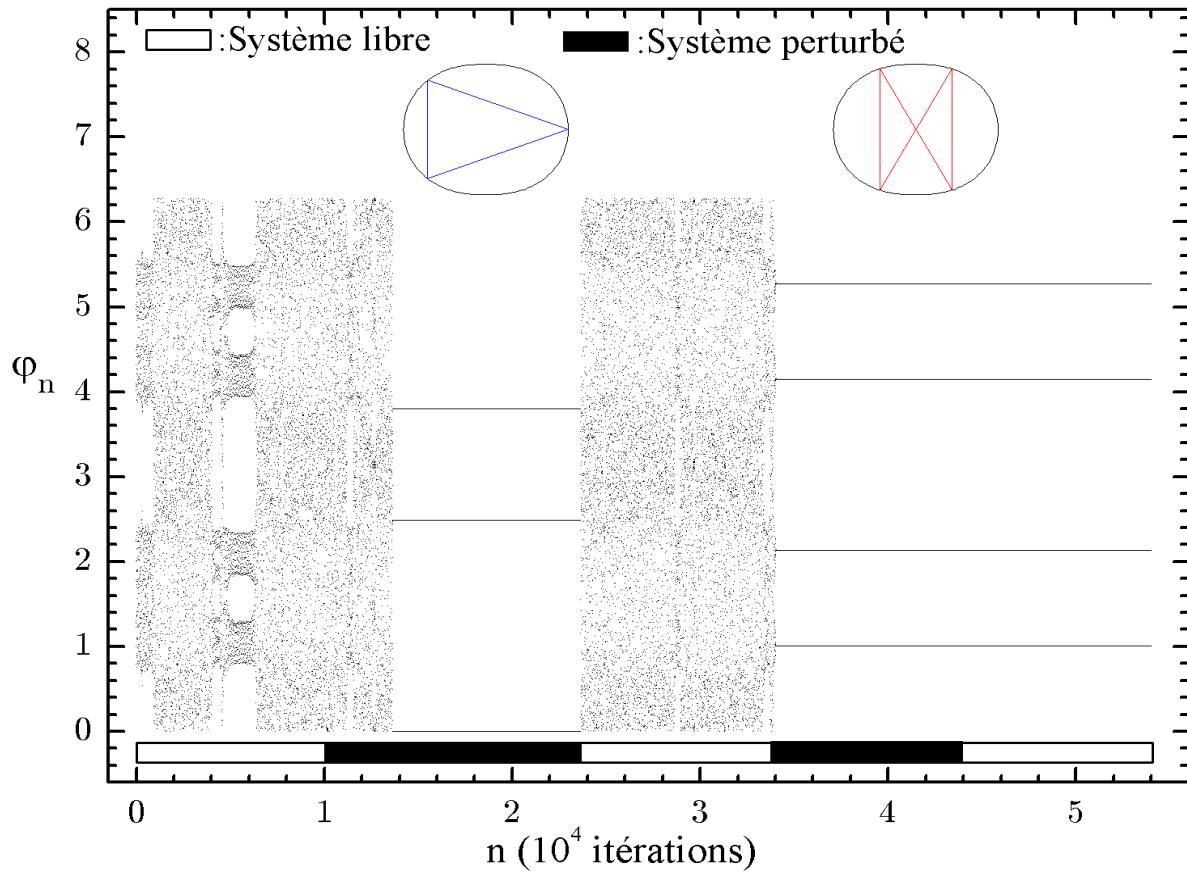
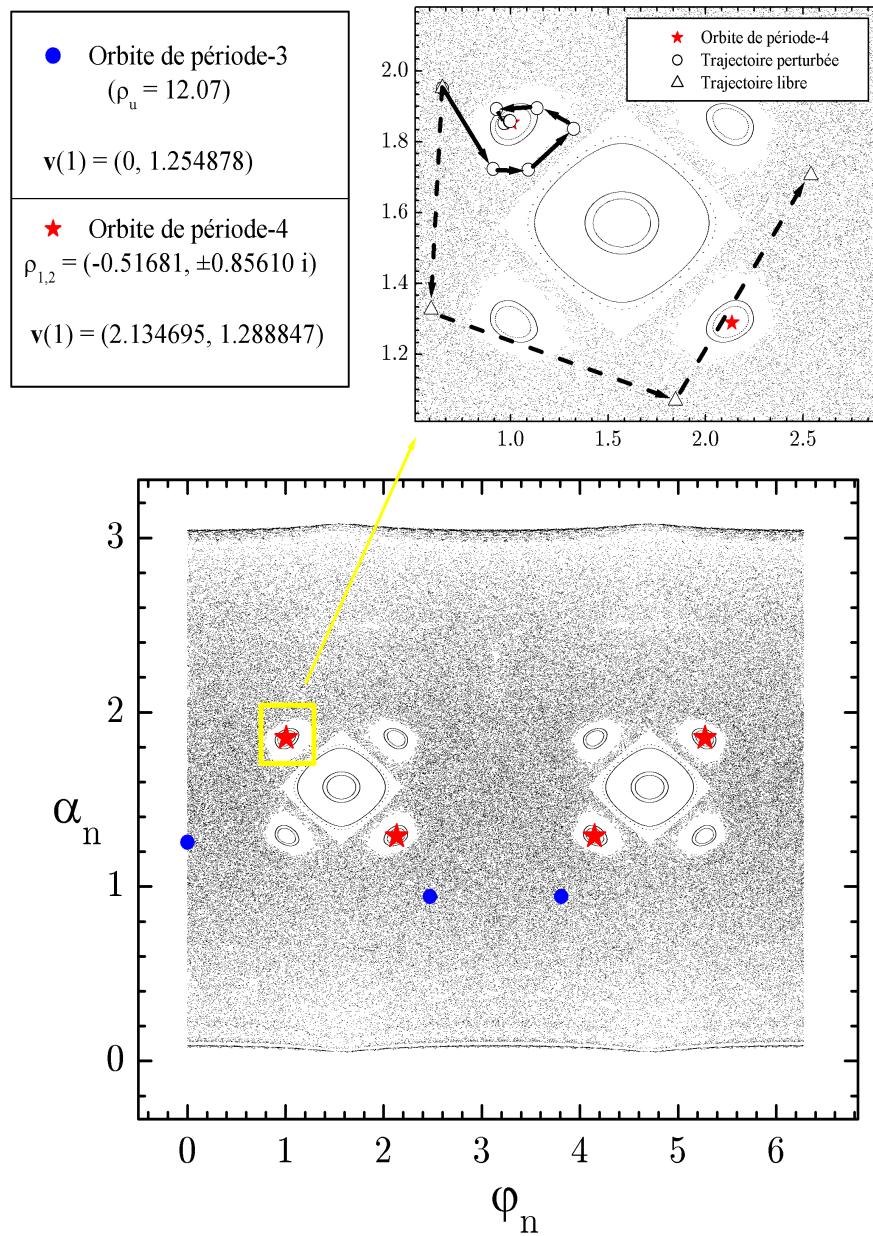


FIG. 4.10 – Ciblage de période 3 et 4 sur le *Billard Ovale Libre*

Le ciblage se fait en perturbant le paramètre  $e$  (5% max), qui correspond ici à la forme de la cavité. On remarque que la période-4 est une orbite stable. Même sans contrôle, le système se maintient sur celle-ci. La particule dans le billard est passée de la bande stochastique à une région de stabilité. Tout comme le Billard Magnétique, on remarque un peu de “sticking” autour d’îlots de stabilité vers la  $4000^{i\text{ème}}$  itération..

FIG. 4.11 – Orbites de période 3 et 4 dans un *Billard Ovale Libre*

Cette figure illustre le diagramme des phases d'un *Billard Ovale Libre* ainsi que la position des OPIs stabilisées par la technique du ciblage à l'aveuglette (réf. figure 4.10). La stabilité et la position de la première composante pour chaque OPI sont aussi indiquées. L'encadré en haut à droite montre comment le système a été amené sur l'orbite de période-4 elliptique. Les symboles  $\circ$  correspondent au système sous l'effet des perturbations. Les symboles  $\triangle$  indiquent où le système aurait été sous évolution libre.

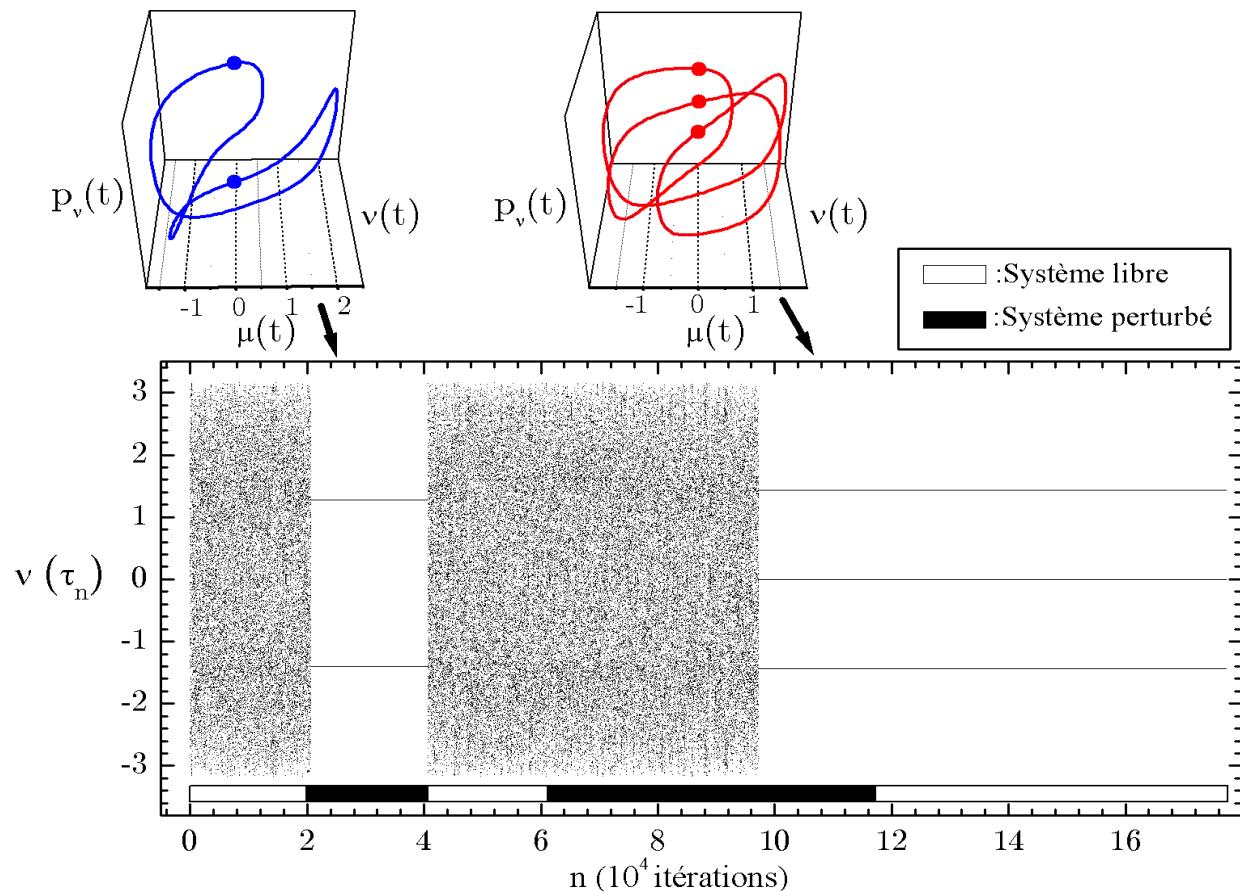


FIG. 4.12 – Ciblage de période 2 et 3 sur le système de DKP

Le ciblage se fait en perturbant les paramètres  $\epsilon$  (0.5% max) du système DKP. L'application discrète pour le flot DKP est obtenue en plaçant la Section de Poincaré en  $\mu(\tau_n) = 0$ ,  $\frac{d\mu}{dt}\Big|_{\tau_n} > 0$ . Les trajectoires ainsi contrôlées sont aussi illustrées. Tout comme pour le Billard Libre, le système termine sa course dans un minuscule îlot de stabilité (période-3, réf. figure 4.13)

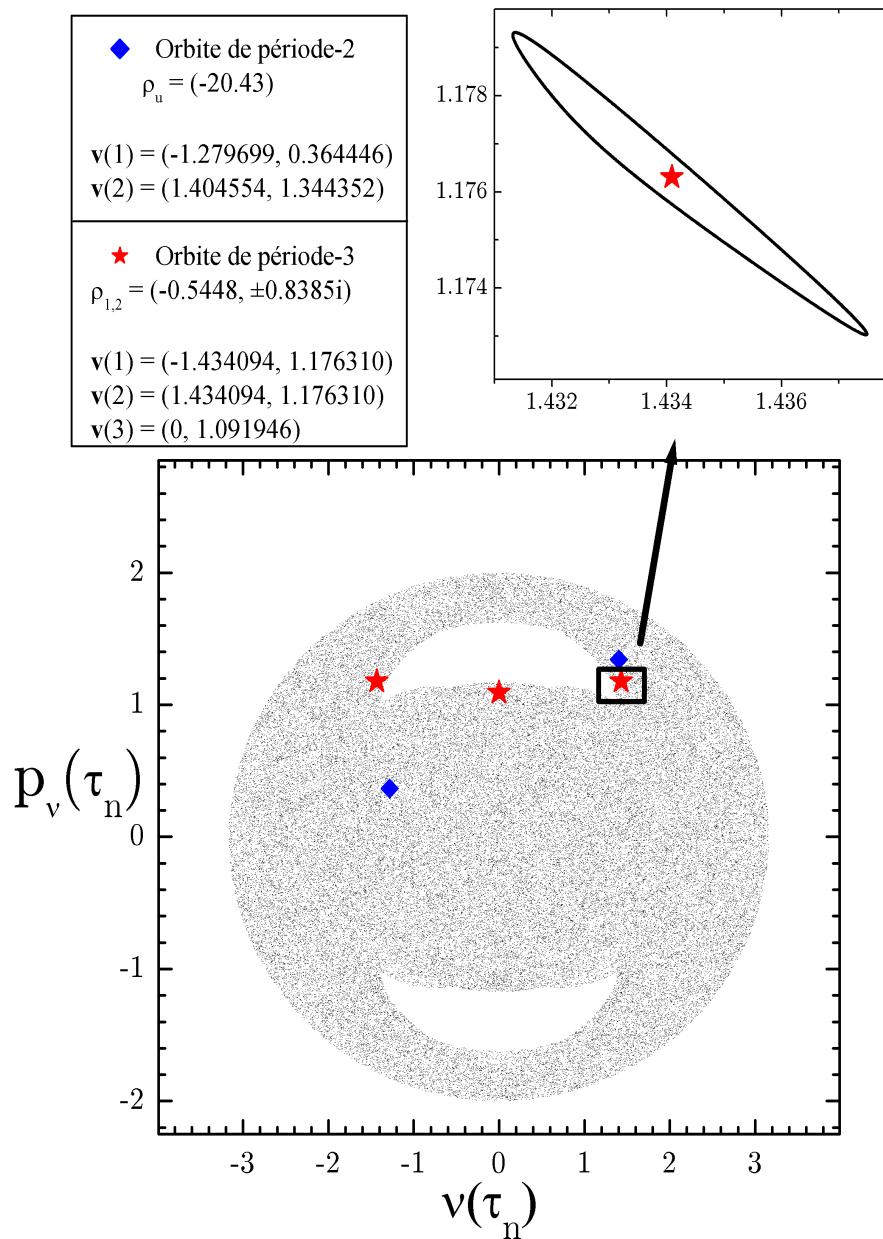


FIG. 4.13 – Orbites de période 2 et 3 dans le système DKP

Cette figure illustre le diagramme des phases du système DKP (sur la Section de Poincaré  $\mu(\tau_n) = 0$ ,  $\frac{d\mu}{dt}\Big|_{\tau_n} > 0$ ) ainsi que la position de OPIs stabilisées par la technique du ciblage à l'aveuglette (réf. figure 4.12). La stabilité et les positions des OPIs sont aussi inscrites dans l'encadré.

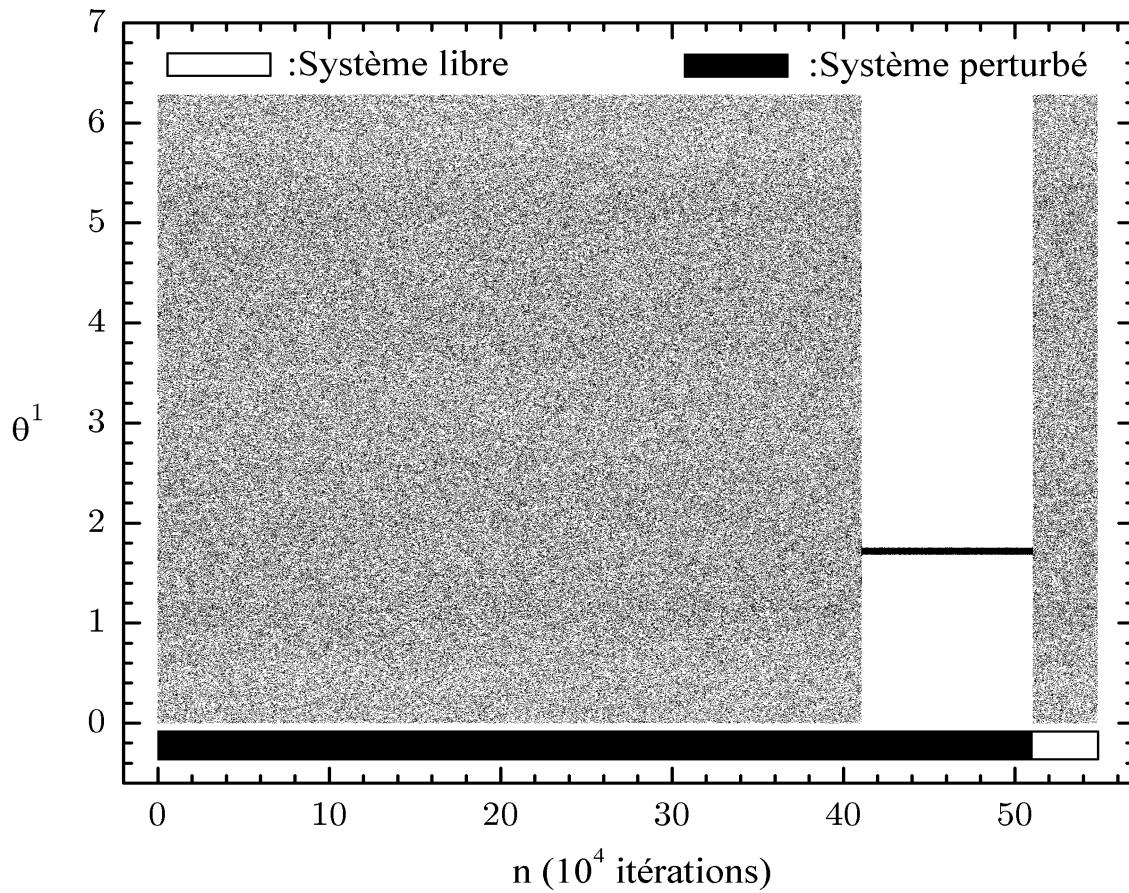


FIG. 4.14 – Ciblage de période 1 pour le Pendule Double

Le ciblage se fait en perturbant le paramètre  $f$  (2% max). La méthode de ciblage n’arrive pas à stabiliser *parfaitement* l’OPI.

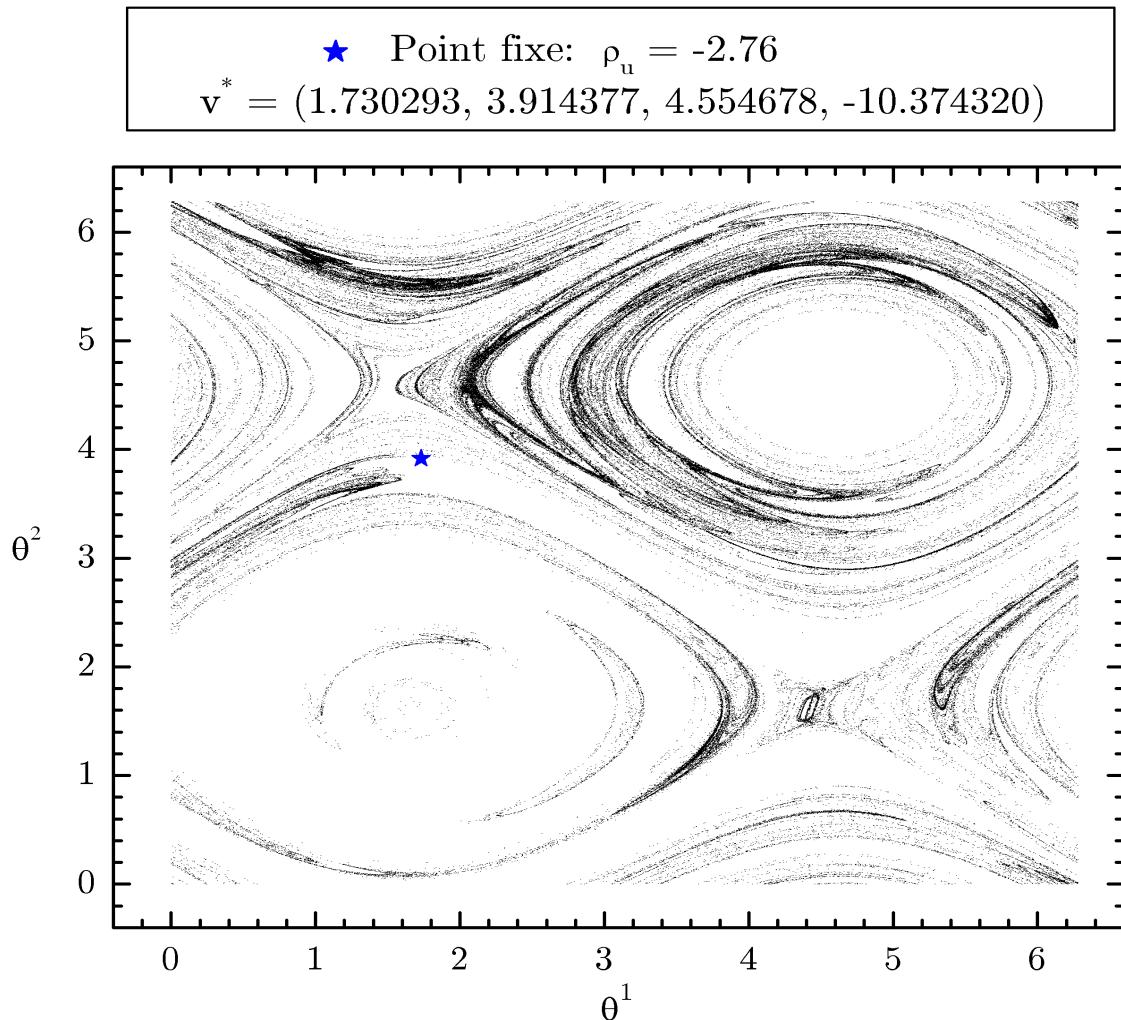


FIG. 4.15 – Point fixe du Pendule Double

Cette figure illustre une projection de l'attracteur du Pendule Double ainsi que la position de l'OPI stabilisée par la technique du ciblage à l'aveuglette (réf. figure 4.14). Le diagramme est obtenu en collectant les points de l'application regroupés dans une bande autour du point fixe  $v^*$ . Cette bande est définie par  $|\mathbf{K} \cdot (v - v^*)| < w$ ,  $\mathbf{K} = (0, 0, 1, 1)$ ,  $w = 10^{-2}$ . La stabilité et la position du point fixe sont aussi inscrites dans l'encadré.

## 4.3 Ciblage dans un contexte expérimental

Les simulations présentées jusqu'à maintenant ont été réalisées en considérant que l'état du système  $\mathbf{v}_n$  était entièrement connu et accessible. En pratique, cette hypothèse de travail n'est pas applicable; l'expérimentateur étant souvent limité à une mesure scalaire de cet état. Pour les systèmes à plus d'une dimension, on doit alors utiliser les techniques standards de reconstruction pour retrouver une équivalence "dynamique" entre le système et les mesures effectuées sur celui-ci [100], [29], [88].

Nous allons maintenant voir comment il est possible, par le biais de ces méthodes de reconstruction, de simuler des scénarios de ciblage qui se rapprochent d'un contexte plus expérimental. Ici, nous n'utilisons qu'une mesure scalaire sur les systèmes. L'état  $\mathbf{v}_n$  est reconstruit par les coordonnées à délais sur la mesure. De plus, l'application  $\mathbf{F}(\mathbf{v}_n)$ , qui est nécessaire au calcul de la perturbation (équation 4.6 ou 4.9) est estimée à l'aide d'un prédicteur.

Dans cette section, nous présentons deux prédicteurs relativement simples. Nous appliquons ensuite le ciblage à l'aveuglette sur un système dynamique reconstruit. Nous terminons par l'étude d'un cas particulièrement intéressant soit celui du ciblage sur un système hamiltonien reconstruit.

### 4.3.1 Construction d'un prédicteur

Il est possible de construire un prédicteur pour estimer l'application  $\mathbf{F}^{(m)}(\mathbf{v}, \mathbf{p}_0)$  lors du processus de ciblage. Deux types prédicteurs sont ici présentés : le premier s'inspire d'un article classique de Lorenz [55] et le second, un peu plus raffiné, repose sur les idées de Farmer & Sidorowich [31], Casdagli [13] et de Sauer [68].

### 4.3.1.1 Prédicteur de Lorenz

Si on dispose d'une banque de données suffisamment longue, on peut évaluer  $\mathbf{F}^{(m)}(\mathbf{v}_n, \mathbf{p}_0)$  en repérant d'abord dans cette banque les plus proches voisins de  $\mathbf{v}_n$  et leur  $m^{ième}$  itéré. Nous notons le  $k^{ième}$  voisin de  $\mathbf{v}_n$  par  $\mathbf{v}^{(k)}$  et son  $m^{ième}$  itéré par  $\mathbf{v}_{n+m}^{(k)}$ . On estime  $\mathbf{v}_{n+m} = \mathbf{F}^{(m)}(\mathbf{v}_n, \mathbf{p}_0)$  comme la moyenne des  $\mathbf{v}_{n+m}^{(k)}$ . Cette prédiction sera d'autant meilleur si le voisinage est petit.

### 4.3.1.2 Prédicteur linéaire

On peut aussi construire un modèle linéaire pour  $\mathbf{F}^{(m)}(\mathbf{v}_n, \mathbf{p}_0)$ . Il s'agit de développer  $\mathbf{F}^{(m)}$  autour de plusieurs points dans l'espace des phases et de garder en mémoire les coefficients de ces développements. On évalue  $\mathbf{F}^{(m)}(\mathbf{v}_n, \mathbf{p}_0)$  en recherchant parmi les points pour lesquels un développement en série a été fait, celui qui est le plus près de  $\mathbf{v}_n$ . On utilise ensuite les coefficients associés à ce point particulier pour estimer la position de  $\mathbf{v}_{n+m}$ . La construction d'un tel prédicteur passe par les étapes suivantes :

1. À partir d'une banque de données contenant l'évolution de  $\mathbf{F}$  sous  $\mathbf{p}_0$ , on regroupe les plus proches voisins pour former des amas. La position moyenne du  $i^{ième}$  amas est notée  $\mathbf{v}^{(i)}$ . Chaque amas contient  $K_i$  points voisins, avec  $K_i \geq D + 1$  (pour des raisons qui deviendront évidentes un peu plus loin). Rappelons que  $D$  est la dimension du système. On représente le  $k^{ième}$  voisin du  $i^{ième}$  amas par  $\mathbf{v}^{(i,k)}$ .
2. Autour du  $i^{ième}$  amas, on développe la fonction  $\mathbf{F}^{(m)}(\mathbf{v}, \mathbf{p}_0)$  en série :

$$\mathbf{F}^{(m)}(\mathbf{v}, \mathbf{p}_0) \sim \mathbf{J}^{(i)} \delta \mathbf{v}^{(i)} + \mathbf{b}^{(i)} \quad (\text{pour } \delta \mathbf{v}^{(i)} \ll 1),$$

où nous avons posé  $\delta \mathbf{v}^{(i)} \equiv [\mathbf{v} - \mathbf{v}^{(i)}]$ . Les coefficients de la matrice Jacobienne  $\mathbf{J}^{(i)}$  ( $D \times D$  coefficients) et du vecteur  $\mathbf{b}^{(i)}$  ( $D$  coefficients) doivent maintenant être évalués.

3. On utilise les différents voisins de  $\mathbf{v}^{(i)}$  pour ajuster les coefficients de  $\mathbf{J}^{(i)}$  et  $\mathbf{b}^{(i)}$ .

Dans l'approximation linéaire, chaque voisin  $\mathbf{v}^{(i,k)}$  doit obéir à l'équation :

$$\mathbf{F}^{(m)}(\mathbf{v}^{(i,k)}, \mathbf{p}_0) \sim \mathbf{J}^{(i)} \delta \mathbf{v}^{(i,k)} + \mathbf{b}^{(i)},$$

avec  $\delta \mathbf{v}^{(i,k)} \equiv [\mathbf{v}^{(i,k)} - \mathbf{v}^{(i)}]$ . On peut écrire cette dernière équation explicitement comme :

$$\begin{pmatrix} F_1^{(m)}(\mathbf{v}^{(i,k)}, \mathbf{p}_0) \\ \vdots \\ F_D^{(m)}(\mathbf{v}^{(i,k)}, \mathbf{p}_0) \end{pmatrix} \sim \begin{pmatrix} J_{11}^{(i)} & \cdots & J_{1D}^{(i)} \\ \vdots & \ddots & \vdots \\ J_{D1}^{(i)} & \cdots & J_{DD}^{(i)} \end{pmatrix} \begin{pmatrix} \delta v_1^{(i,k)} \\ \vdots \\ \delta v_D^{(i,k)} \end{pmatrix} + \begin{pmatrix} b_1^{(i)} \\ \vdots \\ b_D^{(i)} \end{pmatrix}. \quad (4.10)$$

Exceptionnellement ici, on note les composantes des vecteurs  $\mathbf{F}^{(m)}$ ,  $\delta \mathbf{v}$ , et  $\mathbf{b}$  par  $F_{1,\dots,D}^{(m)}$ ,  $\delta v_{1,\dots,D}$  et  $b_{1,\dots,D}$  respectivement. Comme l'amas  $i$  possède  $K_i$  points voisins, avec  $K_i \geq D + 1$ , on peut trouver une solution pour les coefficients de la matrice  $\mathbf{J}^{(i)}$  et du vecteur  $\mathbf{b}^{(i)}$  par la méthode des moindres carrés. Par exemple, on obtient les éléments de la  $d^{\text{ième}}$  ligne de  $\mathbf{J}^{(i)}$  ainsi que la  $d^{\text{ième}}$  composante de  $\mathbf{b}^{(i)}$  en écrivant d'abord les  $K_i$  équations linéaires associées à ces quantités et en solutionnant (au sens des moindres carrés) ce système d'équations. À partir de l'équation 4.10 et de l'ensemble des  $K_i$  voisins, on écrit, pour la composante  $d$  de  $\mathbf{F}^{(m)}(\mathbf{v}^{(i,k)}, \mathbf{p}_0)$  :

$$\begin{pmatrix} F_d^{(m)}(\mathbf{v}^{(i,1)}, \mathbf{p}_0) \\ \vdots \\ F_d^{(m)}(\mathbf{v}^{(i,K_i)}, \mathbf{p}_0) \end{pmatrix} \sim \begin{pmatrix} 1 & \delta v_1^{(i,1)} & \cdots & \delta v_D^{(i,1)} \\ \vdots & & \ddots & \vdots \\ 1 & \delta v_1^{(i,K_i)} & \cdots & \delta v_D^{(i,K_i)} \end{pmatrix} \begin{pmatrix} b_d^{(i)} \\ J_{d1}^{(i)} \\ \vdots \\ J_{dD}^{(i)} \end{pmatrix}. \quad (4.11)$$

On voit, sous cette dernière forme, que  $D + 1$  inconnus doivent être évalués (soit les  $D$  éléments  $J_{d1}^{(i)} \dots J_{dD}^{(i)}$  et la composante  $b_d^{(i)}$ ) à partir de  $K_i$  équations. Cette évaluation peut entre autre se faire à l'aide d'une décomposition en valeur singulière (*Singular Value Decomposition*, [76], [53]).

4. Lorsque les  $\mathbf{J}^{(i)}$  et  $\mathbf{b}^{(i)}$  ont été évalués pour tous les amas, le modèle de  $\mathbf{F}^{(m)}$  est complet. On doit garder en mémoire l'ensemble de ces coefficients. Pour un  $\mathbf{v}$

donné, on évalue  $\mathbf{F}^{(m)}(\mathbf{v}, \mathbf{p}_0)$  en recherchant d'abord parmi l'ensemble des amas, celui qui est le plus près de  $\mathbf{v}$  et en calculant ensuite :

$$\mathbf{F}^{(m)}(\mathbf{v}, \mathbf{p}_0) \sim \mathbf{J}^{(\bar{i})} \delta \mathbf{v}^{(\bar{i})} + \mathbf{b}^{(\bar{i})}.$$

Le symbole  $\bar{i}$  est utilisé pour indiquer l'amas qui est le plus près de  $\mathbf{v}$ .

Mentionnons en terminant que la précision des deux prédicteurs que nous venons de présenter dépend du nombre de données utilisées pour la phase d'apprentissage ( $N_{app}$ ). Si le nombre de ces données est relativement grand ( $N_{app} > 1000$ ), on peut ajouter un algorithme pour la recherche des voisins [91] et rendre ainsi la procédure rapide et efficace.

La fonction  $\mathbf{F}^{(m)}(\mathbf{v}, \mathbf{p}_0)$  nécessaire au calcul de la perturbation peut être estimée à partir de l'un ou l'autre de ces prédicteurs. Le vecteur  $\mathbf{V}_n$  qui représente la variation paramétrique de  $\mathbf{F}^{(m)}$  au point  $\mathbf{v}_n$  et qui est aussi essentiel dans la méthode de ciblage proposée, peut être estimé par :

$$\mathbf{V}_n \equiv D_p \mathbf{F}^{(m)}(\mathbf{v}_n, p_0) \approx \frac{\mathbf{F}^{(m)}(\mathbf{v}_n, p_0 + \delta p) - \mathbf{F}^{(m)}(\mathbf{v}_n, p_0)}{\delta p}. \quad (4.12)$$

En plus d'une banque de données contenant l'évolution du système sous le paramètre  $p_0$ , il faut avoir une séquence d'états  $\{\mathbf{v}_n\}$  sous le paramètre  $p_0 + \delta p$  afin de modéliser  $\mathbf{F}^{(m)}(\mathbf{v}_n, p_0 + \delta p)$ .

On présente à la figure 4.16 l'erreur moyenne ( $E_{moy}(1)$ ) des deux types de prédicteur en fonction du nombre de données initiales utilisées pour la construction des modèles, et ce pour trois systèmes dynamiques (Hénon, Ikéda et Hénon 3 – D). Cette erreur est normalisée par rapport à la déviation standard moyenne de la variable  $v_1$  de ces différents systèmes :

$$E_{moy}(m) = \frac{\sqrt{<(F_{v_1 \text{prédicteur}}^{(m)} - F_{v_1}^{(m)})^2>}}{\sqrt{<(v_1) - <v_1>)^2>}} \equiv \frac{\sigma_\Delta(m)}{\sigma_{v_1}}$$

Ainsi, on sait que le prédicteur n'est plus d'aucune utilité lorsque  $E_{moy} \rightarrow 1$ . Dans ce cas particulier, l'application  $\mathbf{F}$  peut être simplement estimée par l'écart quadratique moyen de la variable  $v_1$ .

De plus, nous avons aussi représenté sur la figure 4.16 le seuil d'erreur au-delà duquel le ciblage à l'aveuglette n'est plus possible. Ce seuil est indiqué par un symbole en noir sur la figure. On remarque qu'il est toujours de l'ordre de  $10^{-2}$  et ce, peu importe le système dynamique. Par contre, le nombre de données  $N_{app}$  nécessaires pour l'atteindre augmente avec la complexité du système. Par exemple, à l'aide du prédicteur linéaire, on arrive à stabiliser une période  $m = 1$  dans Hénon avec seulement  $N_{app} = 316$  points dans la phase d'apprentissage. Il faut cependant  $N_{app} = 10000$  pour réussir un ciblage dans le cas du système Hénon-3D !

### 4.3.2 Ciblage à l'aveuglette sur un système dynamique reconstruit

Nous allons ici supposer que la seule information disponible sur le système dynamique consiste en une mesure scalaire  $\mathcal{M}(t)$  de l'état du système qui évolue de façon continue. En utilisant un délai  $\kappa$  et une dimension de recouvrement  $N$ , on obtient un vecteur de dimension  $N$  dans l'espace de recouvrement [88] :

$$\tilde{\mathbf{v}}(t) = (\mathcal{M}(t), \mathcal{M}(t - \kappa), \dots, \mathcal{M}(t - (N - 1)\kappa)).$$

On construit ensuite une application discrète de dimension  $D = N - 1$  à partir de ce flot continu en éliminant la première composante de ce vecteur à délais par le biais d'une Section de Poincaré placée en  $\tilde{v}^1(\tau_n) \equiv \mathcal{M}(\tau_n) = c = \text{constante}$ . On génère ainsi une suite d'états  $\mathbf{v}_n$ , qui représentent maintenant un système discret reconstruit à partir d'une mesure :

$$\mathbf{v}_n = (\mathcal{M}(\tau_n - \kappa), \dots, \mathcal{M}(\tau_n - D\kappa)).$$

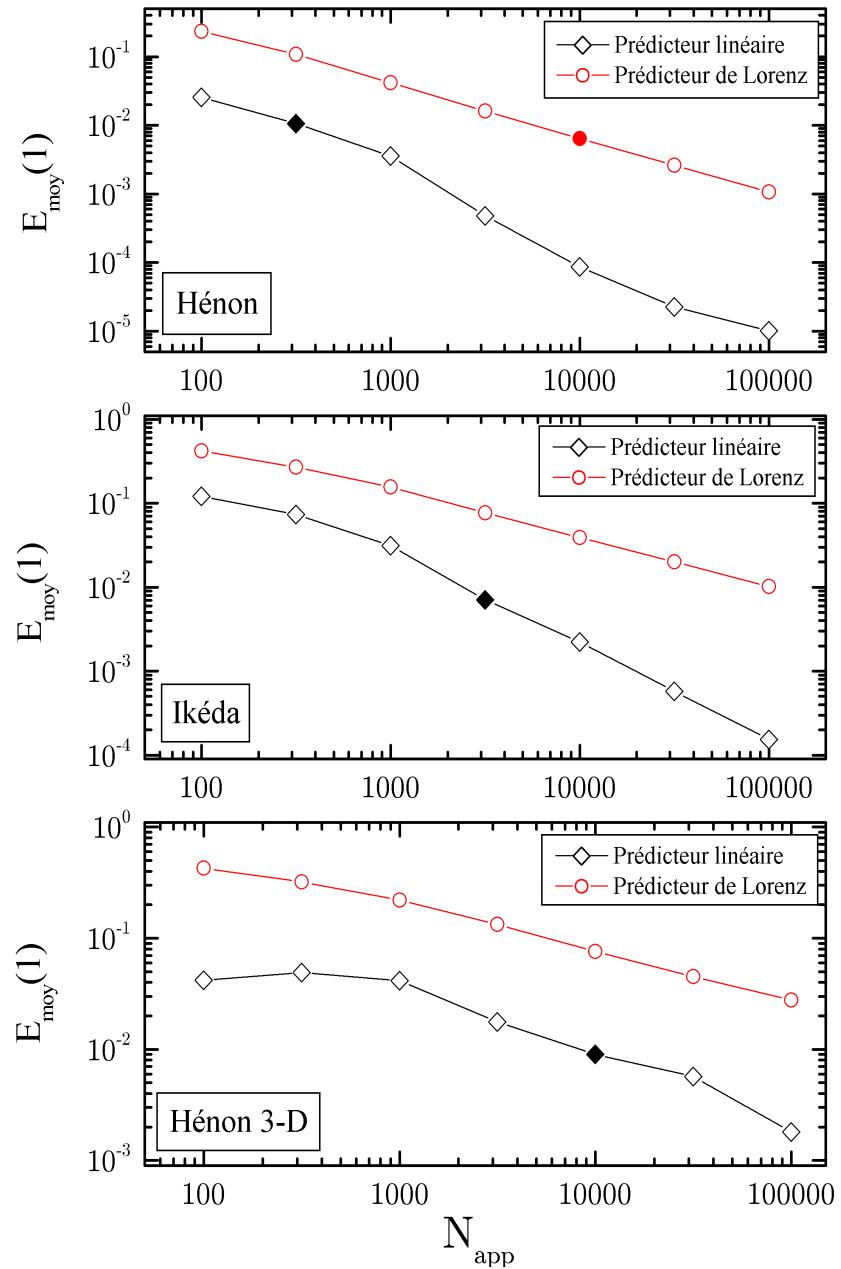


FIG. 4.16 – Erreur moyenne normalisée des prédicteurs en fonction de  $N_{app}$

Ce graphique représente l'erreur normalisée des prédicteurs de Lorenz ( $\circ$ ) et linéaire ( $\diamond$ ) en fonction du nombre de données initiales pour construire le modèle ( $N_{app}$ ). On remarque que pour un nombre de données quelconque, l'erreur augmente avec la complexité et la dimension du système. On indique aussi par les symboles pleins le nombre de données initiales minimales pour réussir le ciblage à l'aveuglette.

À partir de l'évolution de l'état  $\mathbf{v}_n$  ainsi reconstruit, on peut modéliser l'application  $\mathbf{F}$  (par un prédicteur linéaire) et appliquer la technique de ciblage à l'aveuglette pour stabiliser les OPIs d'un système donné.

Le système choisi est celui de l'oscillateur de Duffing, présenté à la section 2.4.3. Historiquement, l'oscillateur de Duffing a été utilisé par Dressler et Nitsche afin de généraliser les techniques de contrôle du chaos aux dynamiques reconstruites [26]. Les paramètres de l'oscillateur sont  $d = 0.2$ ,  $f = 36.0$  et  $\omega = 0.661$ . On prend comme mesure  $\mathcal{M}(t) \equiv x(t)$ . Le délai pour la reconstruction est  $\kappa = \frac{1}{4}T$  avec  $T = \frac{2\pi}{\omega}$  qui correspond à la période de la force excitatrice. On utilise une dimension de recouvrement  $N = 3$  et on choisit la Section de Poincaré en  $\mathcal{M}(\tau_n) = 1$  avec  $\frac{d\mathcal{M}}{dt}\Big|_{\tau_n} > 0$ . La figure 4.17 montre l'attracteur de Duffing dans l'espace réel (a) et reconstruit (b). En plaçant une Section de Poincaré en  $\mathcal{M}(\tau_n) = 1$  sur l'attracteur reconstruit, on obtient l'attracteur illustré sur la figure 4.18. C'est à partir de points collectés sur cette surface que l'on construit la suite d'états  $\{\mathbf{v}_n\}$ .

Il est possible de localiser avec une bonne précision les points fixes  $\mathbf{v}^*$  de l'attracteur reconstruit sur la Section de Poincaré. Par récurrence, on collecte d'abord quelques points qui reviennent dans un voisinage  $\epsilon$  après une itération sur cette section. Les points ainsi amassés sont tous situés dans le voisinage d'un point fixe. On estime la position du point fixe en prenant d'abord le centre de masse de l'amas (noté  $\mathbf{v}_{C.M.}$ ). À l'aide de tous les points de l'amas, on modélise ensuite l'application  $\mathbf{F}(\mathbf{v})$  autour de  $\mathbf{v}_{C.M.}$  avec le même modèle linéaire décrit à la section 4.3.1 :

$$\mathbf{F}(\mathbf{v}) \sim \mathbf{J}[\mathbf{v} - \mathbf{v}_{C.M.}] + \mathbf{b} \quad (\text{pour } \mathbf{v} - \mathbf{v}_{C.M.} \ll 1).$$

Le point fixe  $\mathbf{v}^*$  devrait maintenant satisfaire l'équation :

$$[\mathbf{v}^* - \mathbf{v}_{C.M.}] \sim \mathbf{J}[\mathbf{v}^* - \mathbf{v}_{C.M.}] + \mathbf{b},$$

d'où l'on tire :

$$\mathbf{v}^* \sim \mathbf{v}_{C.M.} + (\mathbf{1} - \mathbf{J})^{(-1)}\mathbf{b}. \quad (4.13)$$

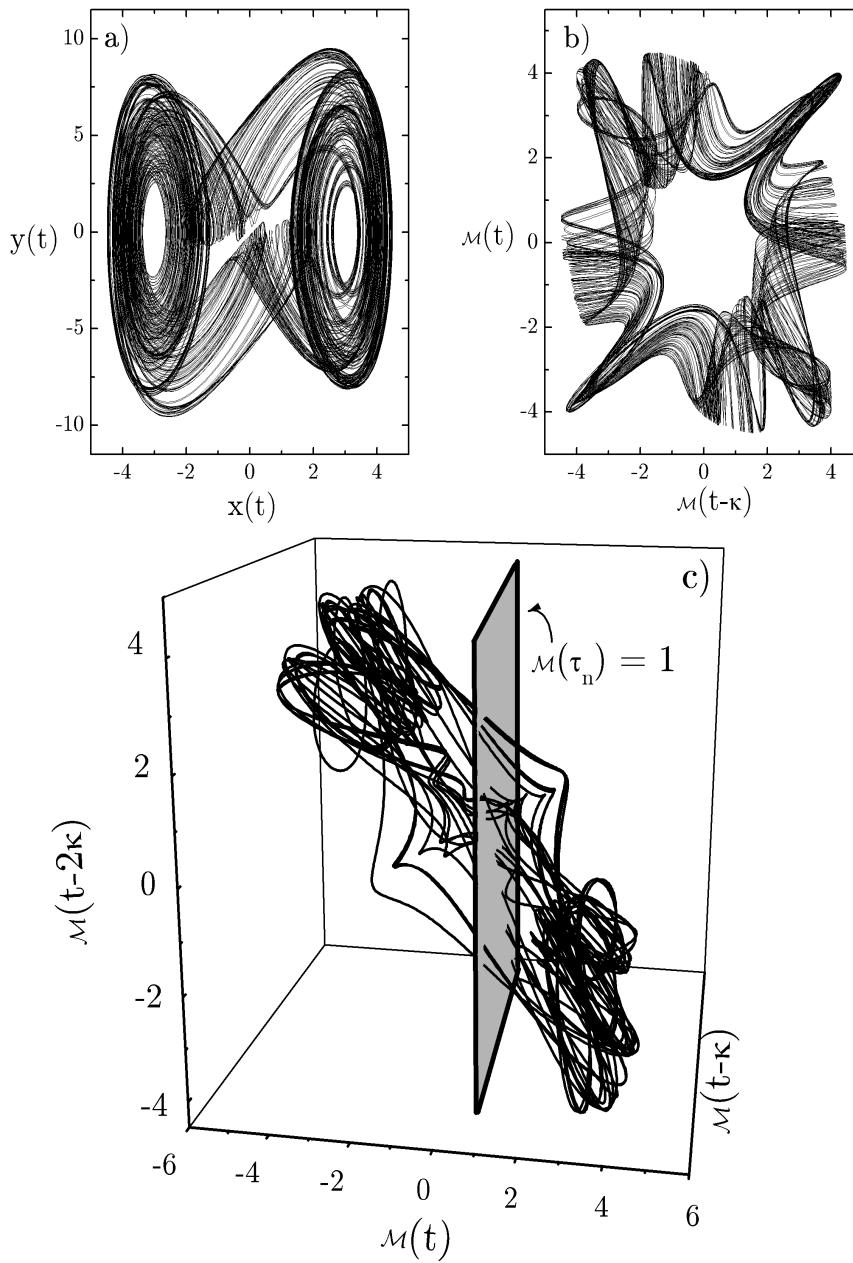


FIG. 4.17 – Reconstruction de l'attracteur de Duffing

On utilise la mesure  $\mathcal{M}(t) \equiv x(t)$  sur l'oscillateur de Duffing ( $d = 0.2$ ,  $f = 36.0$  et  $\omega = 0.661$ ; figure a) pour obtenir une reconstruction de la dynamique de cet oscillateur ( $\kappa = \frac{\pi}{2\omega}$ ) dans un espace de recouvrement de dimension  $N = 2$  (figure b) et  $N = 3$  (figure c). En plaçant une Section de Poincaré en  $\mathcal{M}(\tau_n) = 1$ ,  $\frac{d\mathcal{M}}{dt}\Big|_{\tau_n}$ , on obtient une application discrète (figure 4.18).

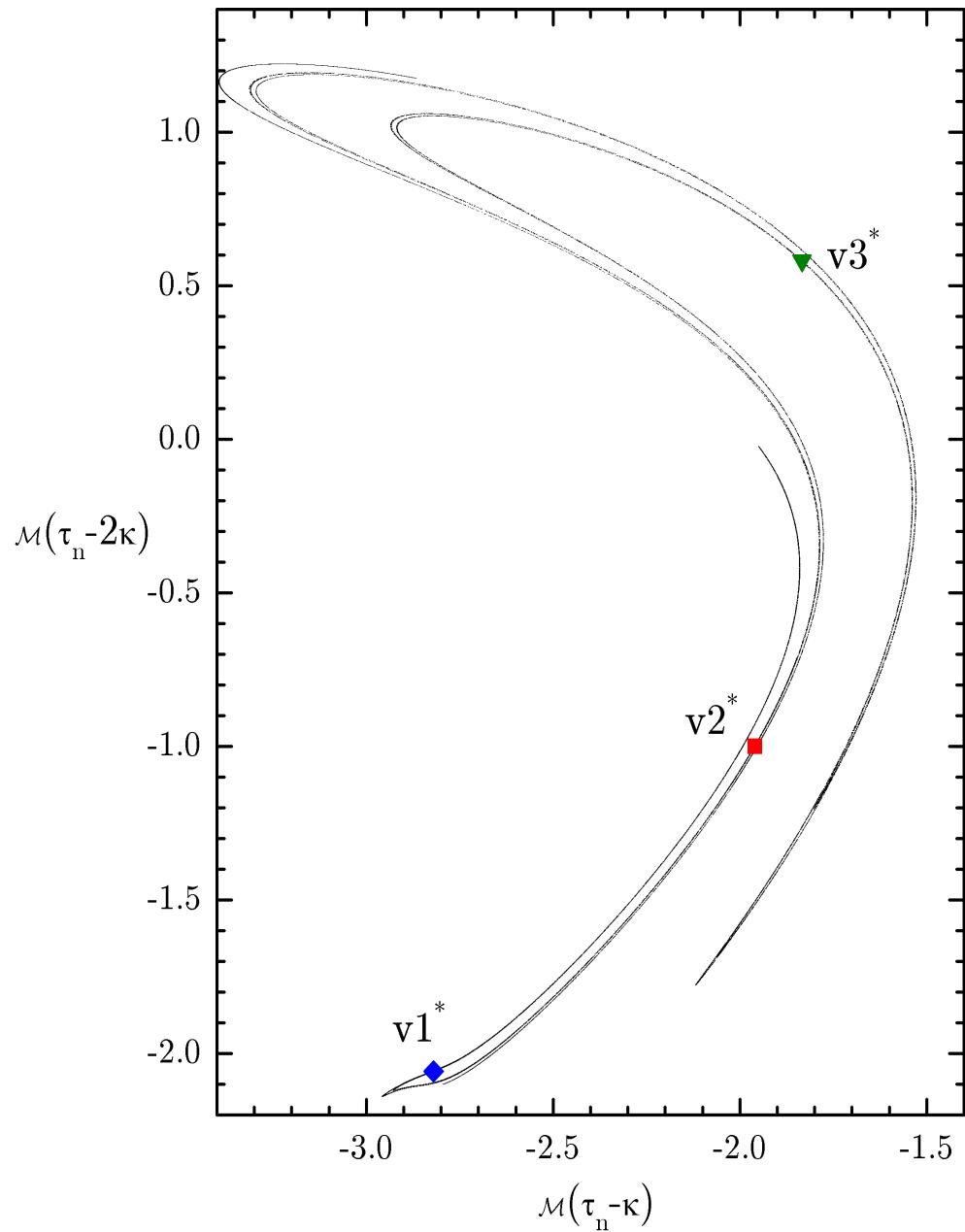


FIG. 4.18 – L'attracteur de Duffing reconstruit sur la Section de Poincaré  $\mathcal{M}(\tau_n) = 1$ ,  
 $\frac{d\mathcal{M}}{dt}\Big|_{\tau_n} > 0$

Point fixe	Coordonnées dans l'espace reconstruit	Matrice jacobienne	Valeurs propres
$\mathbf{v1}^*$	$\mathcal{M}(\tau_n - \kappa) = -2.819824$ $\mathcal{M}(\tau_n - 2\kappa) = -2.058675$	$J_{11} = -2.7161$ $J_{12} = 1.8200$ $J_{21} = -1.2453$ $J_{22} = 0.7792$	$\rho_u = -1.86$ $\rho_s = -0.08$
$\mathbf{v2}^*$	$\mathcal{M}(\tau_n - \kappa) = -1.959798$ $\mathcal{M}(\tau_n - 2\kappa) = -1.000024$	$J_{11} = -0.1338$ $J_{12} = -0.3447$ $J_{21} = -1.2453$ $J_{22} = 5.0421$	$\rho_u = 4.88$ $\rho_s = 0.03$
$\mathbf{v3}^*$	$\mathcal{M}(\tau_n - \kappa) = -1.834001$ $\mathcal{M}(\tau_n - 2\kappa) = 0.582268$	$J_{11} = -0.6859$ $J_{12} = 1.0928$ $J_{21} = 0.6580$ $J_{22} = -1.2656$	$\rho_u = -1.87$ $\rho_s = -0.08$

TAB. 4.4 – Points fixes de l'oscillateur de Duffing dans un espace reconstruit

Ce tableau contient les trois points fixes de l'oscillateur de Duffing ( $d = 0.2$ ,  $f = 36.0$  et  $\omega = 0.661$ ) obtenus par une reconstruction dynamique sur une mesure  $\mathcal{M}(t)$  choisie comme  $\mathcal{M}(t) \equiv x(t)$ . La dimension de recouvrement est  $N = 3$  et le délai  $\kappa = \frac{\pi}{2\omega}$ . La Section de Poincaré est placée en  $\mathcal{M}(\tau_n) = 1$ ,  $\frac{d\mathcal{M}}{dt}\Big|_{\tau_n} > 0$ .

Trois points fixes ont ainsi été localisés et sont illustrés sur figure 4.18. Le tableau 4.4 donne la position de chacun de ces points sur la Section de Poincaré dans l'espace reconstruit. Ce tableau donne aussi les valeurs propres de la matrice jacobienne de l'application autour de chacun de ces points fixes. Cette matrice est directement donnée par la matrice  $\mathbf{J}$  de l'équation 4.13.

Sachant que trois points fixes différents sont potentiellement contrôlables, nous lançons le scénario de ciblage suivant : il faut stabiliser un de ces trois points fixes pour au moins  $10^4$  itérations. Aussitôt que cette tâche est accomplie, on demande de stabiliser un point fixe différent. Pour ce faire, on interdit toute perturbation lorsque le système se trouve dans le voisinage du point fixe déjà contrôlé. Les perturbations appliquées sont calculées à l'aide de l'équation 4.9 (ciblage à un paramètre) en utilisant encore une fois la matrice  $C_k = C_1$ . Le résultat de ce scénario est présenté à la figure 4.19. On remarque qu'un seul des points fixes a été stabilisé ( $\mathbf{v}3^*$ ).

Le ciblage à l'aveuglette souffre donc des mêmes limites que la méthode de contrôle OGY dans sa première version [67]. Ces limites avaient été clairement identifiées par Dressler et Nitsche [26] [65] et ont ainsi inspiré d'autres méthodes de contrôle [83]. Les principales conclusions de Dressler & al. sont les suivantes : dans un espace reconstruit, l'application  $\mathbf{F}$  qui décrit le  $n^{ième}$  retour sur la Section de Poincaré dépend de la valeur des paramètres  $\mathbf{p}_0$  lors du  $n^{ième}$  retour ainsi que de la valeur de ces paramètres lors des itérations précédentes. Il faut donc écrire, pour les systèmes reconstruits :

$$\mathbf{v}_{n+1} = \mathbf{F}(\mathbf{v}_n, \mathbf{p}_n, \mathbf{p}_{n-1}),$$

où  $\mathbf{p}_n$  est la valeur des paramètres lors de la  $n^{ième}$  itération. De façon générale, il faut tenir compte de la valeur des paramètres  $\mathbf{p}_{n-2}, \mathbf{p}_{n-3}, \dots$ . Nous ne considérons dans ce qui suit que  $\mathbf{p}_n$  et  $\mathbf{p}_{n-1}$ . La linéarisation utilisée pour la formulation du ciblage à l'aveuglette (équation 4.14 dans le cas du ciblage à un paramètre) doit maintenant être modifiée afin de tenir compte de la variation du paramètre  $p$  d'une itération à l'autre :

$$\mathbf{F}^{(m)}(\mathbf{v}_n, p_0 + \delta p_n, p_0 + \delta p_{n-m}) \sim \mathbf{F}^{(m)}(\mathbf{v}_n, p_0, p_0) + \mathbf{U}_n \delta p_n + \mathbf{W}_n \delta p_{n-m}, \quad (4.14)$$

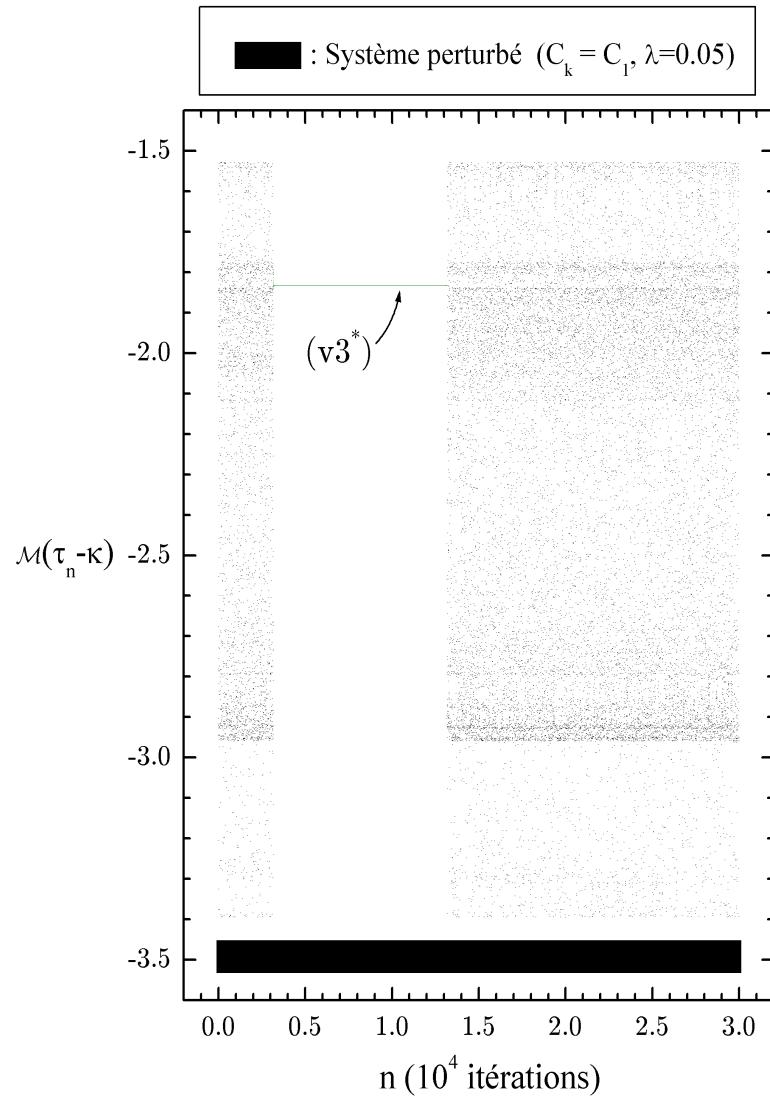


FIG. 4.19 – Premier scénario de ciblage sur l'oscillateur de Duffing

En utilisant le paramètre  $f$  de l'oscillateur de Duffing, on essaie de cibler et stabiliser ses points fixes. On utilise l'équation 4.9 pour le calcul de la perturbation avec  $C_k = C_1$ ,  $\lambda = 0.05$  et  $\delta p_{max} = 1\%$ . On modélise les applications  $\mathbf{F}$  à l'aide d'un prédicteur linéaire ( $N_{app} = 60000$ ). Seul le point fixe  $\mathbf{v}3^*$  arrive à être stabilisé.

avec :

$$U_n \equiv D_{p_n} \mathbf{F}^{(m)}(\mathbf{v}_n, p_n, p_{n-m}) \Big|_{p_n=p_{n-m}=p_0}$$

et

$$W_n \equiv D_{p_{n-m}} \mathbf{F}^{(m)}(\mathbf{v}_n, p_n, p_{n-m}) \Big|_{p_n=p_{n-m}=p_0}$$

En utilisant cette dernière linéarisation, on peut déterminer la valeur de la perturbation à appliquer pour respecter le critère de ciblage (équation 4.8) :

$$\delta p_n = [\mathbf{B}_k(\mathbf{v}_n, \lambda) - \mathbf{W} \delta p_{n-m}] \cdot \mathbf{U}_n / \|U_n\|^2.$$

(4.15)

Les vecteurs  $\mathbf{U}_n$  et  $\mathbf{W}_n$  doivent maintenant être évalués à partir d'une banque de données sur l'évolution de  $\mathbf{v}_n$  obtenues en alternant la valeur de  $p$  entre  $p_0$  pour  $m$  itérations et  $p_0 + \delta p$  pour les  $m$  itérations suivantes. À l'aide de cette banque, il est possible de modéliser  $\mathbf{F}^{(m)}(\mathbf{v}, p_0 + \delta p, p_0)$  et  $\mathbf{F}^{(m)}(\mathbf{v}, p_0, p_0 + \delta p)$  et ainsi évaluer  $\mathbf{U}_n$  et  $\mathbf{W}_n$  comme nous l'avons fait pour  $\mathbf{V}_n$  (équation 4.12). On réfère encore une fois aux articles de Dressler & al. pour plus de détails [26] [65].

Avec cette modification (équation 4.15), on relance le scénario de ciblage avec encore une fois comme objectif de stabiliser les trois points fixes de l'oscillateur de Duffing. Le résultat de cette simulation est présenté à la figure 4.20. Pour les  $3 \times 10^4$  premières itérations, on utilise la première formulation du ciblage à l'aveuglette (équation 4.9). Comme précédemment, seul le point fixe  $\mathbf{v}3^*$  est stabilisé. Après  $3 \times 10^4$  itérations, on modifie le programme afin de tenir compte de la perturbation  $\delta p_{n-1}$  (équation 4.15). Cette fois, on arrive à stabiliser deux points fixes :  $\mathbf{v}3^*$  et  $\mathbf{v}1^*$ .

Malgré la correction apportée au ciblage, le point fixe  $\mathbf{v}2^*$  ne peut être stabilisé. On peut comprendre pourquoi en revenant au tableau 4.4, qui donne la position des points fixes et leur stabilité (valeurs de  $\rho$ ). Rappelons que les scénarios de ciblage ont été réalisés en utilisant la matrice  $\mathbf{C}_k = \mathbf{C}_1$  (équations 4.9 et 4.15). Or, nous avons vu au chapitre 1.3.2.2 que le choix de la matrice  $\mathbf{C}_k$  a un rôle important à jouer quant au type d'orbites périodiques qui peuvent être détectées par la transformation de Schmelcher

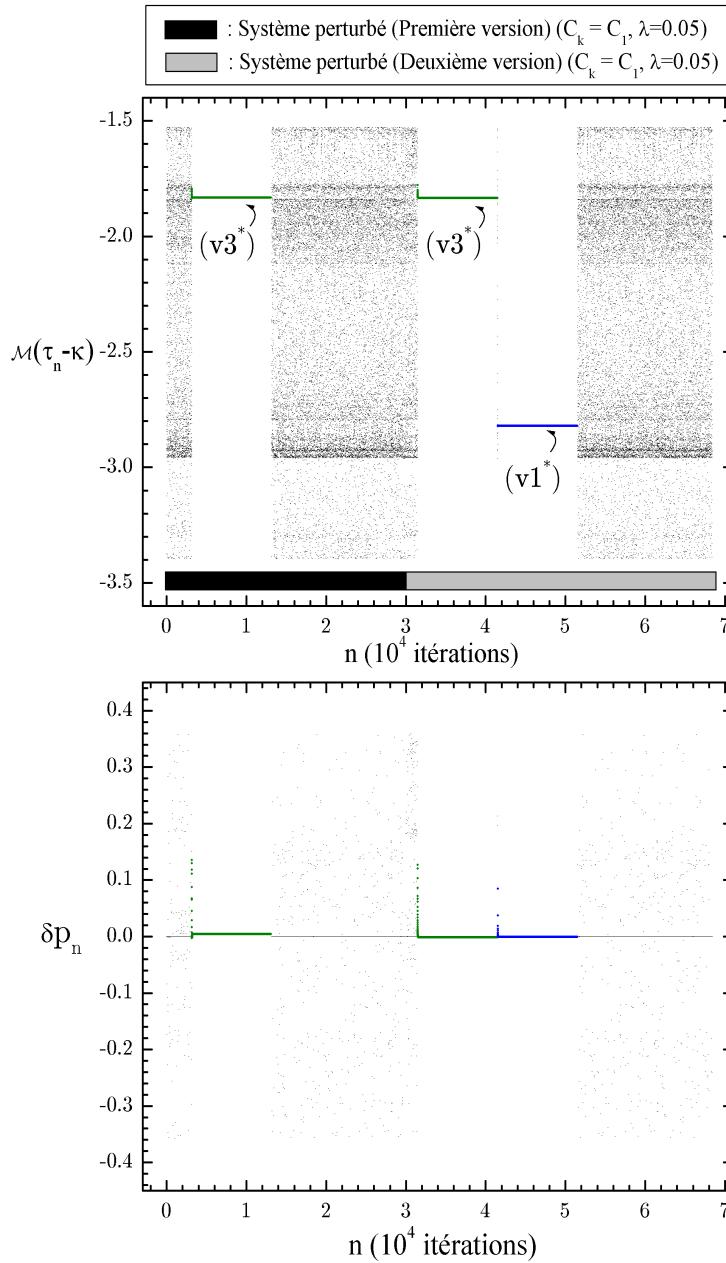


FIG. 4.20 – Deuxième scénario de ciblage sur l'oscillateur de Duffing

Le calcul de la perturbation se fait selon l'équation 4.9 (première version) pour les  $3 \times 10^4$  premières itérations. On change ensuite le programme pour tenir compte de l'historique des perturbations (Deuxième version : équation 4.15). Pour ces deux versions,  $C_k = C_1$ ,  $\lambda = 0.05$  et  $\delta p_{max} = 1\%$ . On modélise les applications  $\mathbf{F}$  à l'aide d'un prédicteur linéaire ( $N_{app} = 60000$ ). Avec la deuxième version, on arrive à stabiliser  $\mathbf{v1}^*$  et  $\mathbf{v3}^*$ . On montre aussi, sur la figure du bas, l'évolution de la perturbation ( $\delta p_n$ ).

et Diakonos. Comme cette transformation est la base même du ciblage à l'aveuglette, le choix de la matrice  $\mathbf{C}_k$  influence aussi le type d'orbite qui peut être stabilisée.

La matrice jacobienne du point fixe  $\mathbf{v}2^*$  a les valeurs propres  $\rho_u = 4.87$  et  $\rho_s = 0.03$  (point hyperbolique sans réflexion) avec  $J_{11} < J_{22}$  (réf. tableau 4.4). Nous avons vu, à la section 1.3.2.2, que ce type de point fixe peut être stabilisé en utilisant la matrice  $\mathbf{C}_k = \mathbf{C}_3$  (réf. tableau 1.3). Il faut donc utiliser cette matrice afin d'amener le système sur le point fixe  $\mathbf{v}2^*$ . La simulation présentée à la figure 4.21 utilise la matrice  $\mathbf{C}_1$  pour les  $2 \times 10^4$  premières itérations (en interdisant le contrôle des points  $\mathbf{v}1^*$  et  $\mathbf{v}3^*$  qui ont déjà été stabilisés lors du scénario précédent). Après ces  $2 \times 10^4$  itérations, on change pour  $\mathbf{C}_k = \mathbf{C}_3$  et le système se stabilise après une centaine d'itérations sur le point fixe  $\mathbf{v}2^*$  ce qui confirme l'importance du choix des matrices  $\mathbf{C}_k$  dans le processus de ciblage !

### 4.3.3 Cas particulier : ciblage sur un système conservatif reconstruit

Il peut être intéressant d'appliquer la méthode du ciblage à l'aveuglette sur un système avec une dynamique mixte (chaotique et régulière). Un tel système, laissé à lui-même, évolue soit dans la bande chaotique, soit dans un îlot de stabilité. Les tores de KAM sont des frontières qui interdisent le passage d'une région vers une autre. Le système qui est initialement situé dans la région chaotique n'a pas accès, sous évolution libre, aux bandes régulières.

Le prochain scénario de ciblage (figures 4.22 à 4.24) propose donc de stabiliser deux périodes-6 dans un système à dynamique mixte : le *Billard Gravitationnel* (réf. section 2.3.3). La mesure scalaire  $\mathcal{M}_n$  sur ce système est choisie comme  $\mathcal{M}_n \equiv x_n$  et la reconstruction se fait avec une dimension de recouvrement  $N = D = 2$ . L'état du

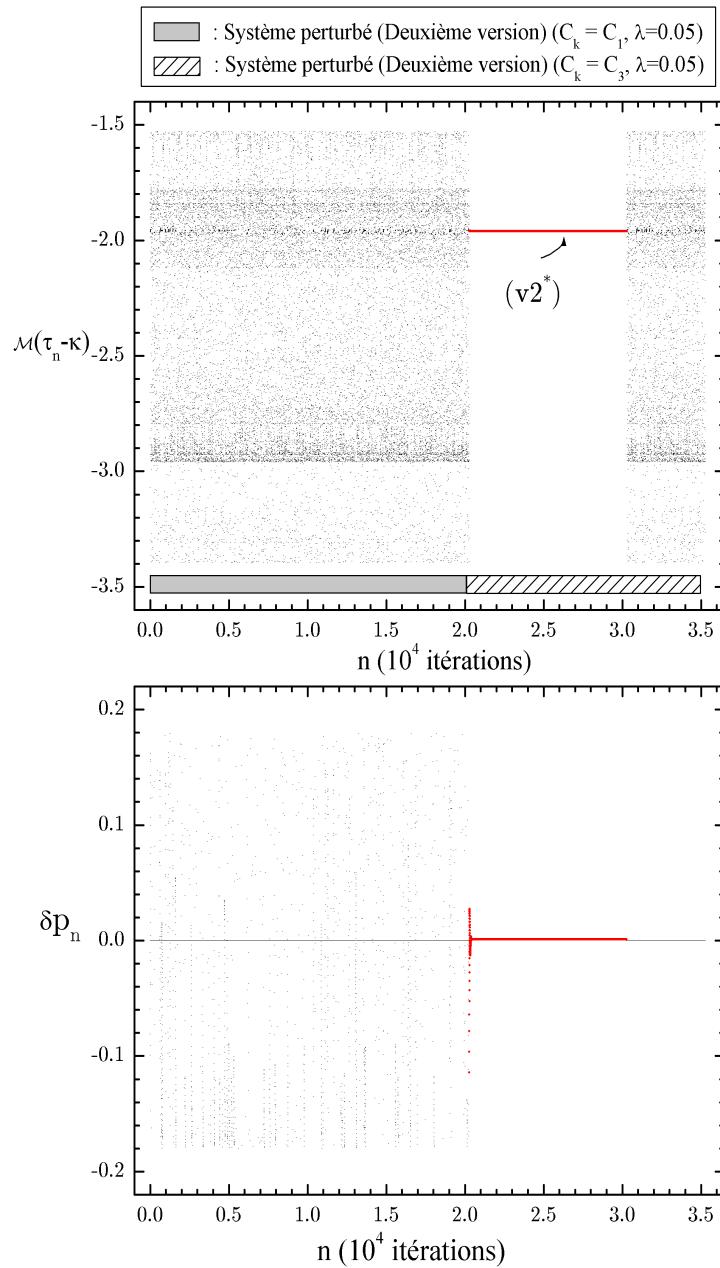


FIG. 4.21 – Troisième scénario de ciblage sur l'oscillateur de Duffing

Le calcul de la perturbation se fait selon l'équation 4.15 (Ciblage à un paramètre, Deuxième Version) avec  $\lambda = 0.05$  et  $\delta p_{max} = 1\%$ . Pour les  $2 \times 10^4$  premières itérations, on choisit  $\mathbf{C}_k = \mathbf{C}_1$  et pour le reste,  $\mathbf{C}_k = \mathbf{C}_3$ . On modélise les applications  $\mathbf{F}$  à l'aide d'un prédicteur linéaire ( $N_{app} = 60000$ ). Le point fixe  $\mathbf{v}2^*$  peut être stabilisé par un choix de matrice approprié ( $\mathbf{C}_k = \mathbf{C}_3$ )

système dans les coordonnées à délais  $\mathbf{v}_n$  est donné explicitement par :

$$\mathbf{v}_n = (\mathcal{M}_n, \mathcal{M}_{n-1}) \equiv (x_n, x_{n-1})$$

On utilise encore une fois un prédicteur linéaire pour modéliser l'application  $\mathbf{F}(\mathbf{v}_n, \mathbf{p}_0)$ . On demande le contrôle pour  $10^4$  itérations de deux périodes-6. Le ciblage se fait en utilisant l'équation 4.9 (i.e. on ne tient pas compte de perturbations appliquées dans le passé  $\delta p_{n-m}$ ) et la matrice  $\mathbf{C}_k = \mathbf{C}_1$ . La première orbite stabilisée est située dans la bande stochastique. Lorsqu'on relâche le contrôle, le système redevient chaotique. La deuxième orbite ciblée est localisée dans un îlot de stabilité. Les perturbations appliquées ont permis au système de franchir les frontières de KAM. Si on enlève le contrôle, le système demeure dans l'îlot.

On remarque cependant qu'à la fin de sa course, le système n'est pas parfaitement situé sur l'orbite stable de période-6. Comme les modélisations de l'application  $\mathbf{F}$  et du vecteur  $\mathbf{V}$  ont été réalisées à partir d'une séquence de  $\mathbf{v}_n$  prise dans la *bande chaotique*, ce modèle est incomplet et ne couvre pas les régions de stabilité. Il serait évidemment possible de refaire un modèle de  $\mathbf{F}$  à partir de l'évolution du système dans l'îlot, et l'amener graduellement sur l'orbite stable.

## 4.4 Efficacité du ciblage à l'aveuglette

Dans les sections précédentes, il a été démontré que la technique du ciblage à l'aveuglette permet de stabiliser des orbites périodiques, stables ou instables, pour une vaste gamme de systèmes chaotiques. Nous voulons maintenant vérifier si cette méthode est *efficace* en terme de ciblage. Pour ce faire, il faut mesurer le temps de transition moyen  $\tau$ , i.e. le nombre d'itérations nécessaires pour amener et stabiliser le système sur une de ses orbites périodiques. Ce temps  $\tau$  peut être ensuite comparé au temps nécessaire pour réussir le contrôle dans le cas où on laisse le système entrer de lui-même dans le

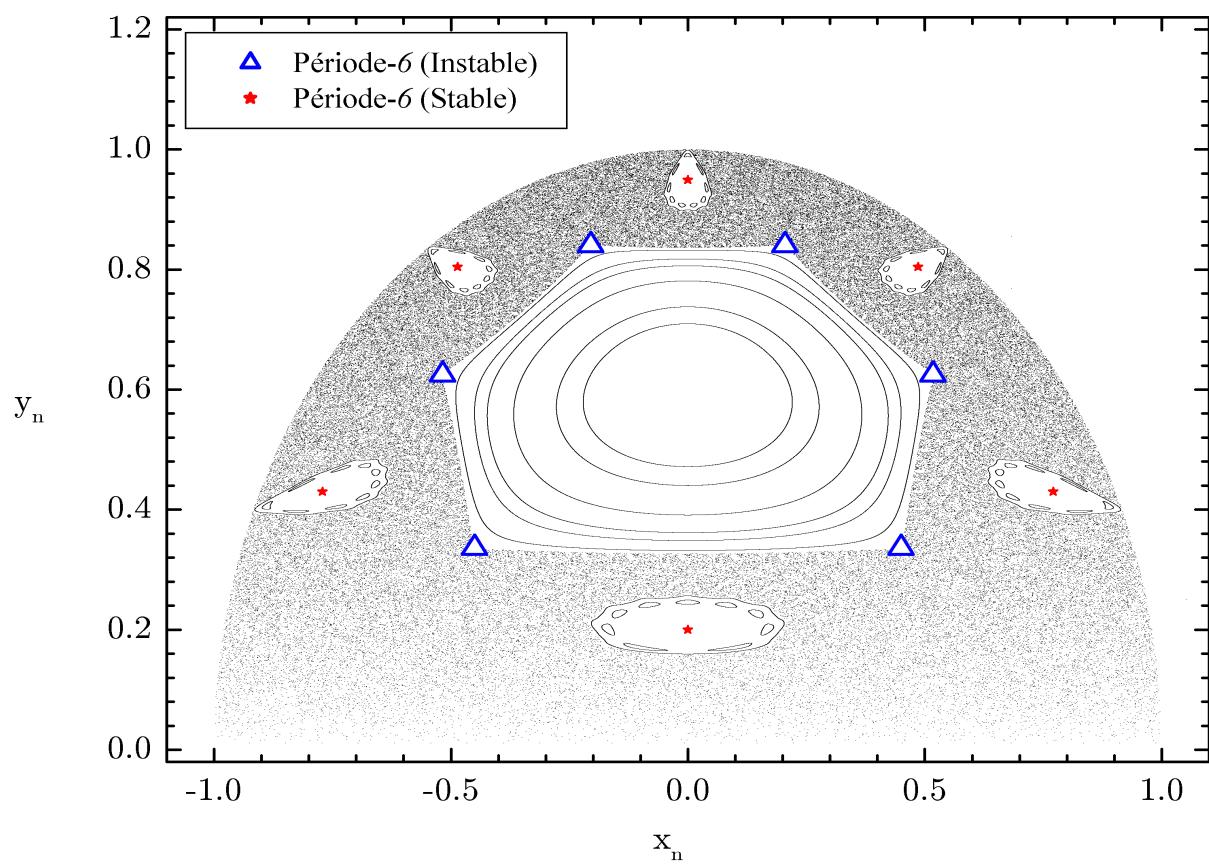
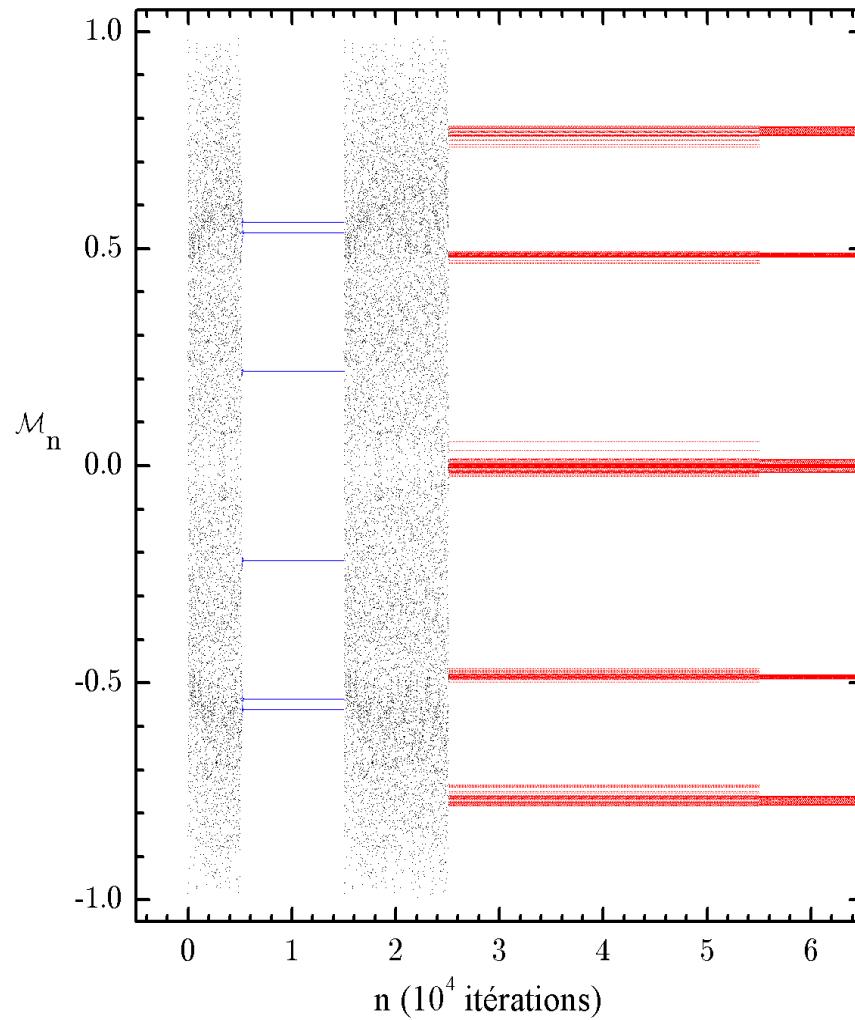
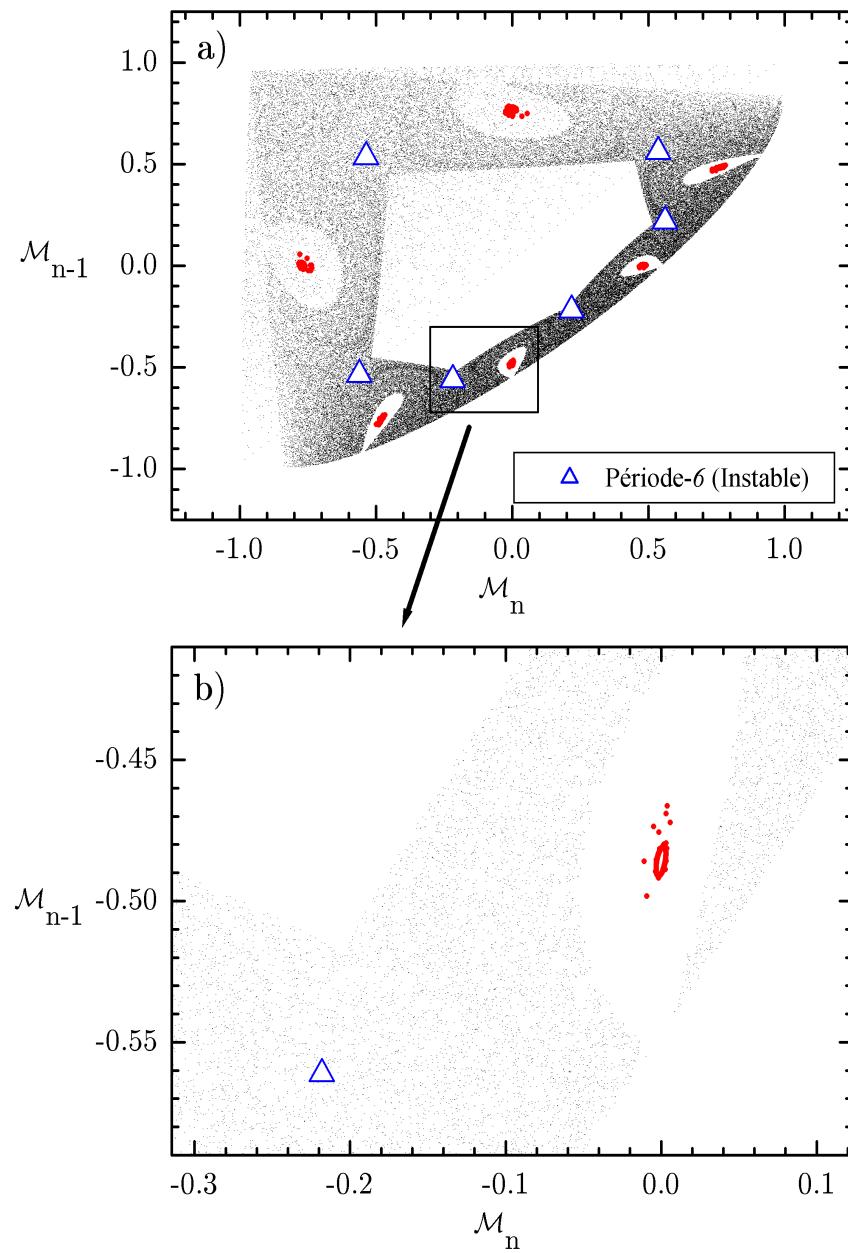


FIG. 4.22 – Diagramme des phases et orbites de période-6 pour le Billard Gravitationnel

FIG. 4.23 – Ciblage à l'aveuglette sur un *Billard Gravitationnel*

À partir d'une reconstruction dans le *billard gravitationnel* (Mesure  $\mathcal{M}_n \equiv x_n$ ,  $\kappa = 1$  et dimension de recouvrement  $N = D = 2$ ), on modélise l'application  $\mathbf{F}$  (Prédicteur linéaire,  $N_{app} = 60000$ ) et on calcule les perturbations selon l'équation 4.9 (ciblage à un paramètre), le paramètre de contrôle est  $\theta$ , soit l'angle formé par le coin du *Billard Gravitationnel*). Deux orbites de période-6 ont été demandées. La deuxième orbite ciblée est située dans un îlot de stabilité (réf. figure 4.24).

FIG. 4.24 – Reconstruction dynamique du *Billard Gravitationnel*

Cette figure montre le diagramme des phases du *Billard Gravitationnel* dans l'espace de recouvrement. Les symboles  $\Delta$  indiquent la position de l'orbite de période-6 instable qui a été contrôlée dans le scénario de ciblage de la figure 4.23. Les points  $\bullet$  sont les positions visitées par le système sous l'effet des perturbations du ciblage. À la fin du scénario, le système évolue dans les îlots

voisinage d'une OPI. Une fois dans le voisinage, on applique des perturbations dictées par des techniques de contrôle standards, de type OGY ou MED.

Nous avons lancé 1000 simulations à partir de conditions initiales choisies aléatoirement dans l'espace des phases et ce, pour différents systèmes. Ces 1000 simulations ont été lancées pour trois cas distincts (I, II et III). Dans le premier (I), on laisse le système évoluer librement jusqu'à ce qu'il visite le voisinage  $\epsilon$  d'une OPI. Une fois dans le voisinage, on applique un contrôle de type OGY ([67], [84]). On calcule ensuite le nombre d'itérations qui ont été nécessaires pour *réussir le contrôle* et on fait la moyenne sur les 1000 simulations. Le contrôle est considéré comme réussi si le système reste dans le voisinage de l'OPI pendant 100 itérations consécutives. Le second cas (II) est presque identique au premier ; on utilise cependant un contrôle de type MED [83]. Pour le troisième et dernier cas (III), on applique la technique du ciblage à l'aveuglette (*CA*). Le système peut être, pour ce dernier cas, perturbé n'importe où dans l'espace des phases. Rappelons que pour le ciblage à l'aveuglette, la position de l'OPI n'est pas connue *a priori*. Le concept de voisinage n'a donc plus de sens pour cette méthode particulière. Le seul critère ici est que la perturbation ne dépasse jamais un certain seuil  $\delta p_{max}$ .

On présente à la figure 4.25 les temps moyens  $\tau$  obtenus pour les 1000 simulations en fonction de la grandeur de la perturbation seuil  $\delta p_{max}$  et ce pour les trois cas (I, II et III). Il est bon ici de préciser que pour les cas I et II, on doit attendre que le système entre de lui-même dans le voisinage de l'OPI. Ce voisinage  $\epsilon$  doit être choisi de sorte que  $\delta p_n = \delta p_{max}$  pour  $\mathbf{v}_n \approx \mathbf{v}(i) + \epsilon$ . Autrement dit, si l'état du système ( $\mathbf{v}_n$ ) se trouve à la limite du voisinage choisi, la perturbation appliquée  $\delta p_n$  sera à peu près égale à la perturbation seuil.

Nous avons fait ces simulations pour des systèmes chaotiques à complexité croissante : Hénon, Ikéda, Hénon-3D et le Pendule Double (réf. chapitre 2). Pour ces quatre systèmes, on cherchait à stabiliser une période-1. Mentionnons que le Pendule double a

36 périodes-1 de sorte que ce test comparatif est quelque peu biaisé pour ce système : le ciblage à l'aveuglette part avec un léger avantage puisqu'il dispose de 36 cibles finales possibles (pour les 2 autres méthodes, on spécifie une seule OPI du Pendule Double).

On constate au premier coup d'oeil que la technique du ciblage à l'aveuglette prend, en général, plus de temps à amener les systèmes sur une OPI. Cette technique semble cependant plus efficace dans le cas du système Hénon-3D. Cette particularité est due au fait que le ciblage s'est fait à l'aide du paramètre  $A$  pour ce système. Dans le cas de Hénon-3D on a  $\mathbf{V} \equiv D_A \mathbf{F}^{(m)}(\mathbf{v}_n, A, B) = \text{constante}$ . Comme le vecteur  $\mathbf{V}$  est constant, le ciblage à l'aveuglette devient particulièrement efficace. Il s'agit cependant d'un cas isolé.

Il faut aussi remarquer que le ciblage à l'aveuglette devient inefficace pour les systèmes plus complexes (Pendule Double). Le critère de minimisation utilisé dans le cas du ciblage (équation 4.8) s'avère insuffisant pour stabiliser efficacement la dynamique. Ce critère est aussi celui du contrôle de type MED et cette méthode s'avère aussi moins efficace pour le Pendule Double. Le contrôle de type OGY se démarque nettement dans cette situation car il utilise toute l'information sur la dynamique locale autour du point fixe : les perturbations sont choisies afin de projeter le système sur la variété stable.

Il est possible de regarder les simulations précédentes d'une autre façon. La figure 4.26 montre le rayon du voisinage efficace moyen ( $\epsilon_{eff}$ ) en fonction de la perturbation seuil ( $\delta p_{max}$ ). Si le système entre dans le voisinage efficace de l'OPI, il sera amené et stabilisé sur celle-ci. Le but de tout ciblage est d'obtenir un voisinage efficace plus grand que celui obtenu par de simples techniques de contrôle. Il s'avère que le ciblage à l'aveuglette *diminue* le voisinage efficace autour de l'OPI.

Il faut ici bien comprendre que les cibles locales que nous prescrivons sont dictées par une transformation (équation 4.2) qui a une dynamique complètement différente de celle

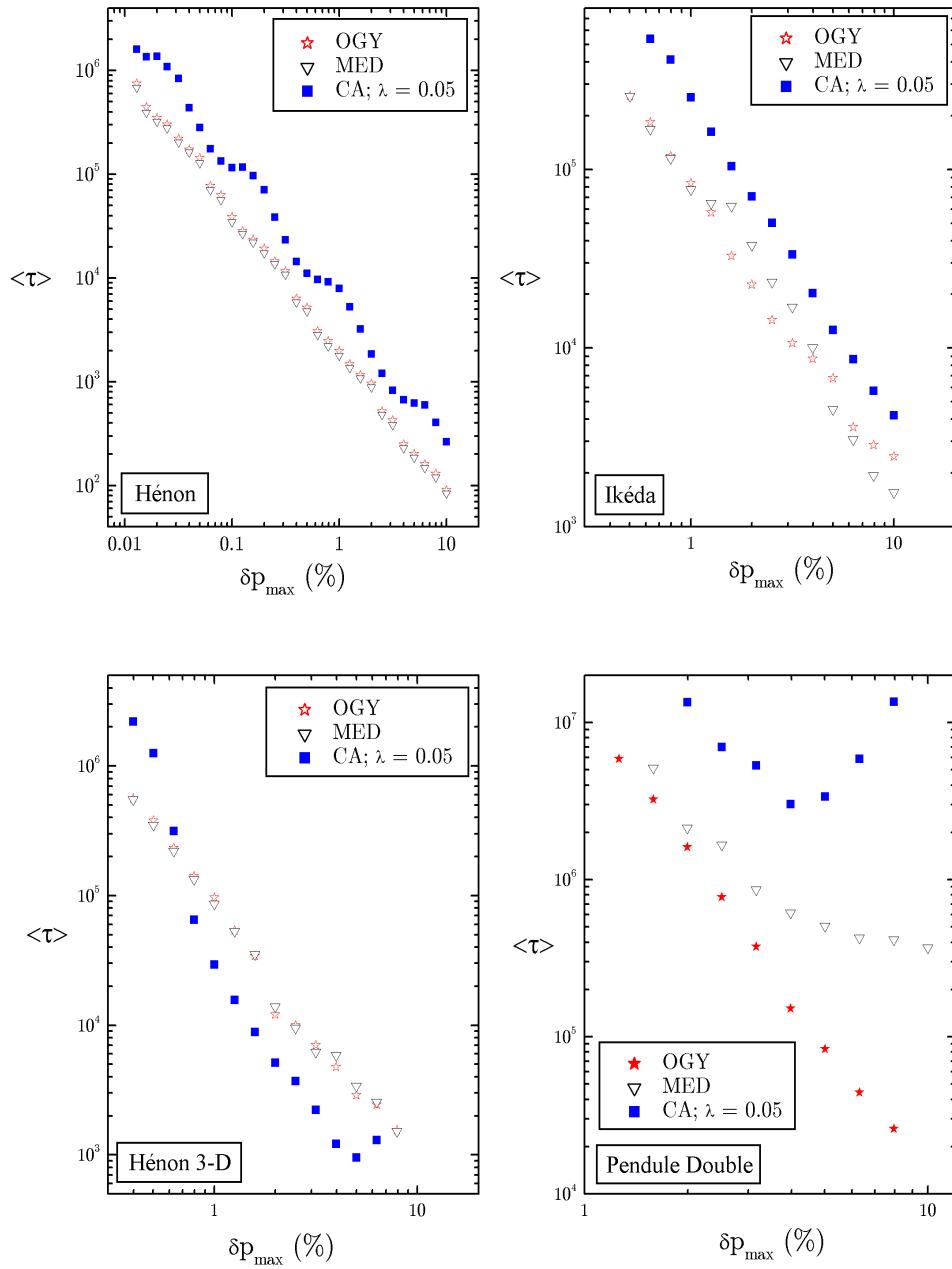
du système initial. La figure 3.2 présentée au chapitre 3 illustre bien la différence entre ces deux dynamiques. La dynamique de  $\mathbf{F}(\mathbf{v}_n, \mathbf{p}_0)$  n'arrive pas, en général, à suivre celle de la transformation de Schmelcher-Diakonos et les *cibles locales* ne peuvent pas être atteintes par de faibles perturbations sur  $\mathbf{p}_0$ . Dans le voisinage de l'orbite périodique, les cibles deviennent plus accessibles et le contrôle peut s'enclencher.

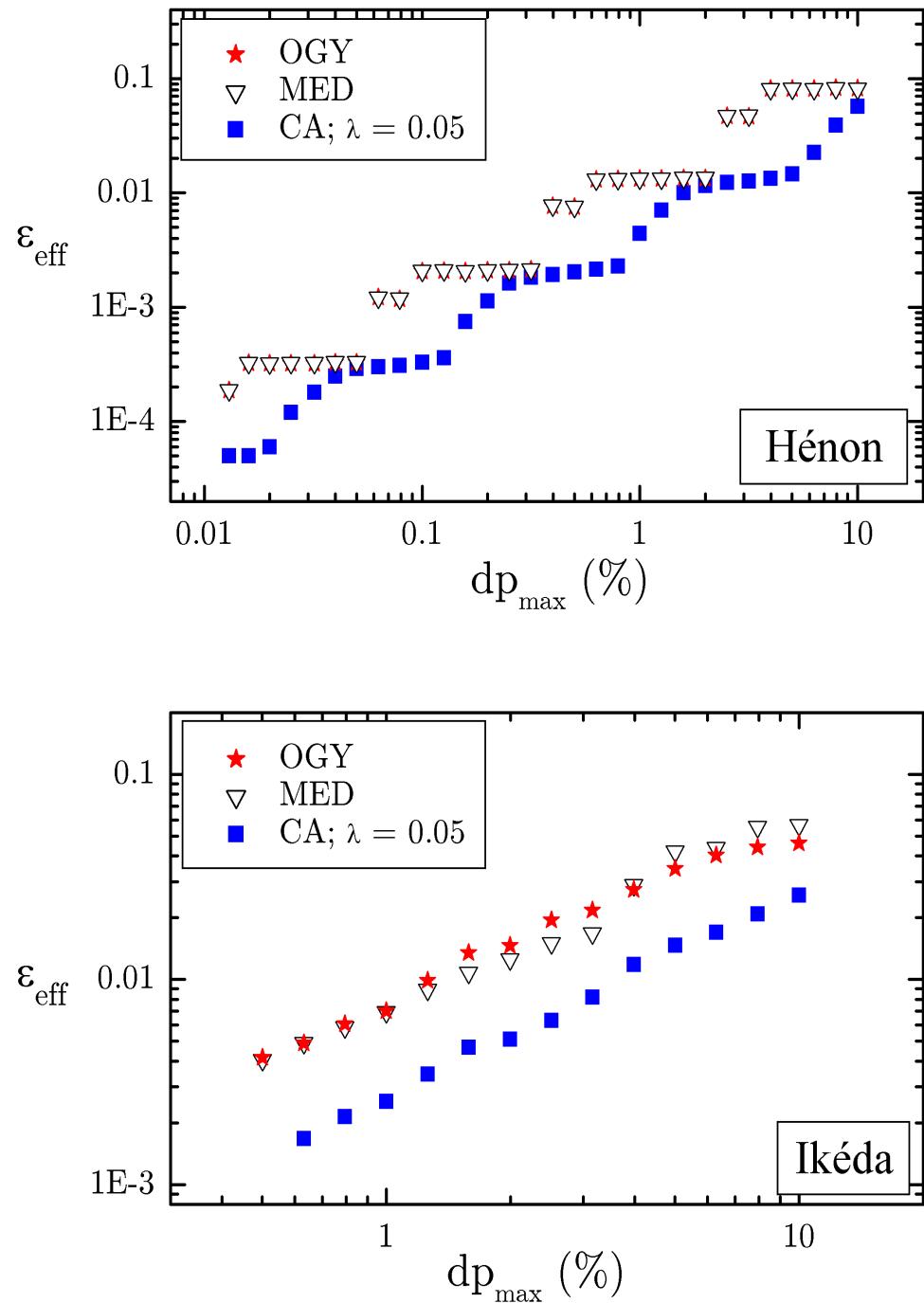
On peut alors se demander si le ciblage à l'aveuglette serait plus efficace en utilisant  $D$  paramètres de contrôle plutôt qu'un seul. En utilisant  $D$  paramètres, on peut en effet se permettre de “viser” directement les cibles locales (équation 4.6). On reprend les simulations précédentes en utilisant cette fois, l'équation 4.6 (ciblage à  $D$  paramètres) pour le calcul de la perturbation. Les résultats sont présentés à la figure 4.27 pour les systèmes de Hénon et Ikéda ( $D = 2$ ). Comme le montre clairement ces courbes, le ciblage à plusieurs paramètres n'a pas vraiment d'influence sur les temps de transition moyens.

## 4.5 Discussion

Nous avons proposé dans ce chapitre une méthode de ciblage et de contrôle d'orbites périodiques qui utilise les propriétés de convergence de la transformation de Schmelcher-Diakonos. Un aspect élégant de cette méthode est qu'elle ne demande pas de connaître la position des orbites périodiques *a priori*. La technique du ciblage a été utilisée sur plusieurs systèmes dynamiques (applications discrètes et flots continus, tant dissipatifs que conservatifs). Au premier coup d'œil, les résultats sont satisfaisants puisqu'avec un critère relativement simple (équation 4.9), on arrive à stabiliser des systèmes assez complexes. Le Pendule Double (Dimension de Lyapunov de 2.8) est difficilement stabilisable avec notre approche.

Le ciblage à l'aveuglette devait, en principe, permettre d'amener rapidement un système dynamique sur une de ses orbites périodiques. Nous avons fait une mesure des

FIG. 4.25 – Temps de transition moyen en fonction de la perturbation seuil  $\delta p_{max}$

FIG. 4.26 – Voisinage efficace  $\epsilon_{eff}$  en fonction de la perturbation seuil  $\delta p_{max}$

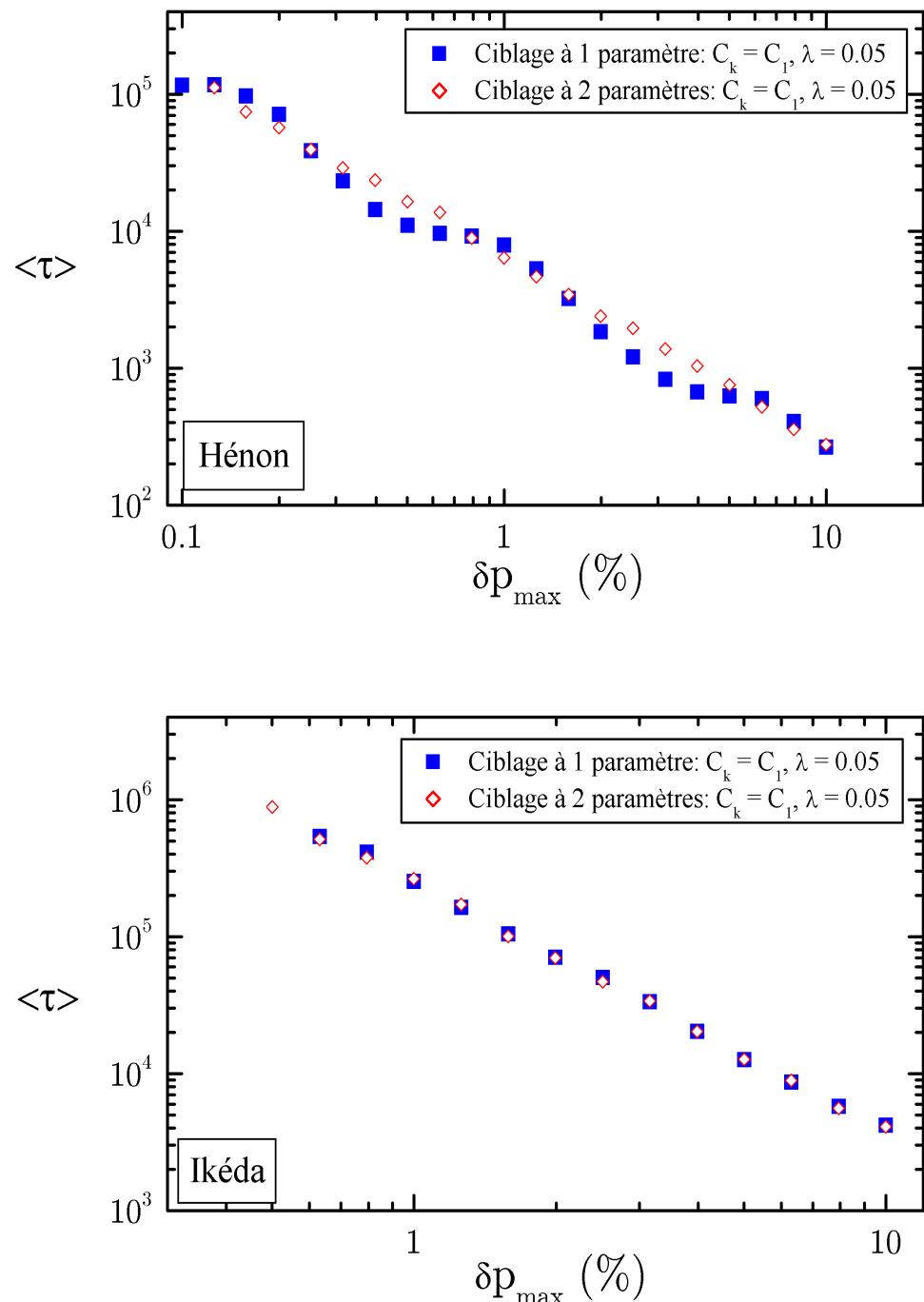


FIG. 4.27 – Temps de transition moyen pour un ciblage à 1 et 2 paramètres

temps de transitions moyens ( $\tau_{moy}$ ) sur plusieurs systèmes. Il s'avère que le ciblage à l'aveuglette prend en général *plus* de temps pour stabiliser la dynamique qu'une technique standard de contrôle (i.e. une évolution libre et un contrôle appliqué seulement dans le voisinage de l'OPI).

Ce dernier résultat est compréhensible puisque la méthode de ciblage est une approche qui utilise des petites perturbations pour amener graduellement l'état du système vers une orbite périodique. Ces perturbations sont calculées afin de “viser” des *cibles locales* prescrites par la transformation de *SD*. Or, cette transformation évolue avec une dynamique qui est trop différente de la dynamique originale (figure 3.2) pour que les cibles locales soient accessibles par de petites perturbations. Ces cibles deviennent atteignables seulement dans le voisinage de l'OPI. Seulement, une fois dans le voisinage de l'OPI, il est préférable de “viser” directement celle-ci plutôt que d'essayer d'atteindre une cible locale.

Malgré la faible efficacité de ciblage de notre approche, elle demeure toutefois intéressante pour une situation bien particulière : le ciblage et le contrôle expérimental d'un système conservatif. Ce scénario a été simulé à la section 4.3.3. Comme notre méthode de ciblage peut aboutir sur des orbites stables ou instables, les bandes régulières deviennent alors aussi accessibles. Le scénario que nous avons réalisé (réf. figure 4.23) présente une situation où l'état du système passe de la bande chaotique à une région régulière. L'orbite de période-6 dans l'îlot de stabilité ne serait pas accessible par une technique de contrôle standard. En effet, pour ces techniques standards, il faut connaître la position de l'OPI ainsi que sa matrice jacobienne pour intervenir. Si on ne dispose que d'une série de mesures expérimentales, prises dans la région chaotique, l'orbite de période-6 dans l'îlot ne sera pas repérée et sa matrice jacobienne encore moins évaluée.

En terminant, mentionnons que la position des cibles locales ( $\bar{\mathbf{v}}$ ) qui est calculée à partir de l'équation 4.3 pourrait aussi se faire en utilisant la transformation de Davidchack et Lai. Comme cette transformation converge plus rapidement vers l'OPI, l'ef-

ficacité du ciblage se verrait légèrement augmentée. Par contre, on perdrait un côté intéressant soit celui de pouvoir décider du type d'orbite qui sera ciblée, aspect qui est propre à la transformation de Schmelcher et Diakonos.

# Conclusion

Ce travail de recherche nous a permis d'aborder le problème fondamental et complexe de la détection des orbites périodiques instables dans les systèmes chaotiques. L'aspect fondamental repose sur le lien étroit qui existe entre la stabilité des orbites périodiques et la caractérisation de leurs propriétés dynamiques. Une première approche pour étudier les propriétés dynamiques d'un système physique est d'ajouter de faibles perturbations à un Hamiltonien “approximatif”, Hamiltonien dont il est possible d'obtenir la solution analytiquement. Cependant, plusieurs systèmes non linéaires manifestent des comportements asymptotiques beaucoup plus riches que ce que l'on peut entrevoir à partir d'une première approximation analytique. Les orbites périodiques offrent alors une autre forme d'approximation de la dynamique puisqu'elles forment un *squelette rigide* qui décrit relativement bien le comportement à long terme. La localisation de ces cycles si importants s'avère donc essentielle, d'où l'intérêt d'approfondir et améliorer les méthodes connues pour leur détection.

Le premier de nos objectifs était de comparer la performance de deux algorithmes récents conçus pour réaliser cette tâche. Nous voulions également approfondir ces deux méthodes afin de les utiliser sur différents systèmes chaotiques dont des flots continus dissipatifs et conservatifs. Ce dernier point, bien que plus technique, n'avait jamais été abordé.

Au premier chapitre, nous avons d'abord essayé de faire ressortir le plus simplement

et le plus rigoureusement possible le rôle fondamental des OPIs, tout en soulignant la problématique associée à leur détection. Ce chapitre a aussi permis de présenter en détails les deux algorithmes en question, soit celui de Schmelcher-Diakonos (*SD*) et celui de Davidchack-Lai (*DL*).

Le chapitre suivant présentait quelques modèles chaotiques de dimensions et de complexité variées. Les algorithmes *SD* et *DL* ont été historiquement publiés pour des applications discrètes. Une attention particulière a donc été portée aux systèmes continus afin d'expliquer comment obtenir numériquement la dynamique discrète et la matrice jacobienne à partir du flot.

Nous avons présenté une analyse du taux de convergence des deux méthodes au troisième chapitre. Pour les systèmes unidimensionnels, nous avons fait ressortir l'existence, pour l'algorithme *SD*, d'une valeur de  $\lambda = \lambda_{optimal}$  qui permet une convergence quadratique. La performance de cet algorithme peut donc être améliorée en limitant le nombre maximal d'itérations ( $N_{SD_{max}}$ ) à environ  $\sim 50$  et en variant progressivement la valeur de  $\lambda$ . Les systèmes continus quasi  $1D$  (tel Lorenz) pourraient bénéficier de cette approche puisqu'on évite alors d'intégrer les équations variationnelles.

Pour les systèmes de dimensions supérieures à  $1$ , nous avons dégagé une limite importante quant à la convergence de l'algorithme *SD*. Lorsqu'on applique cette méthode à des systèmes de dimensions supérieures à  $1$ , le nombre d'itérations nécessaires pour converger vers la solution est de l'ordre de  $\sim 1/\log(1 - \lambda)$ . Comme la valeur de  $\lambda$  doit être très petite ( $\sim 10^{-4} - 10^{-5}$ ) pour la détection complète des OPIs, l'algorithme *SD* devient inefficace pour les cycles très instables. Même jumelé à une méthode Newton-Raphson, il demeure peu performant. Pour les systèmes continus dont la dimension sur la Section de Poincaré est supérieure à  $1$ , il est donc plus avantageux d'utiliser l'algorithme *DL*, même si ce dernier nécessite l'intégration des équations variationnelles. Un problème important qu'il reste à résoudre ici est le choix des conditions initiales pour les 2 algorithmes. Avec la dimension qui augmente, le nombre de matrices  $\mathbf{C}_k$

devient imposant et les conditions initiales devraient être choisies plus judicieusement. On pourrait envisager de contourner ce problème en restreignant davantage le nombre de ces matrices.

D'autre part, l'étude des deux algorithmes a permis de proposer une nouvelle méthode de ciblage d'orbites périodiques. Cette technique utilise la transformation *SD* afin de choisir convenablement les perturbations à appliquer aux paramètres accessibles du système pour cibler et stabiliser un cycle périodique. Nous avons démontré qu'il était possible de stabiliser un tel cycle sans connaître sa position *a priori*. L'analyse approfondie de notre méthode a toutefois fait ressortir qu'elle possède une faible capacité de ciblage étant donné les grandes différences qui existent entre la dynamique du système chaotique et celle de la transformation *SD*. Dans le cas des systèmes conservatifs, notre méthode peut être intéressante puisqu'elle permet d'accéder aux régions régulières à partir de la bande chaotique.

Un aspect important et élégant des méthodes de détection et de ciblage que nous avons présentées dans cette thèse est qu'elles demandent peu d'information sur la dynamique du système d'intérêt. Les algorithmes de détection peuvent donc être utilisés rapidement comme premier outil pour une recherche d'orbites périodiques. Nous nous sommes limités dans cette thèse à illustrer leur applicabilité sur quelques systèmes chaotiques relativement simples. Il faut toutefois insister sur le fait que ces algorithmes peuvent aussi être utilisés sur des systèmes plus complexes. Par exemple, une dynamique fort intrigante, qui n'a malheureusement pu être abordée dans ce travail, est celle des longues chaînes moléculaires (cristaux liquides) en interaction avec la lumière d'un faisceau laser. Un modèle théorique a été développé en collaboration avec E. Brasselet afin de décrire le comportement de ces cristaux en présence d'une onde électromagnétique (les résumés des articles que nous avons publiés sur le sujet est disponible à l'annexe D). Ce modèle est sensiblement plus complexe que ceux qui ont été présentés dans cette thèse étant donné qu'il s'agit d'un système aux dérivées partielles. Malgré la complexité du modèle, l'algorithme *SD* a pu être utilisé afin de repérer les états stationnaires du

système.

Dans un autre ordre d'idée, il faut aussi garder à l'esprit que l'ensemble complet des orbites périodiques d'un système chaotique est utilisé pour obtenir des informations *quantitatives* sur ses propriétés dynamiques et physiques. Il faut donc avoir une certaine idée de ce que l'on veut faire avec cet ensemble. Un travail important a été réalisé dans les dernières années afin d'approximer de façon optimale les propriétés dynamiques d'un système chaotique à partir de ses cycles périodiques. Même si cet aspect n'a pas vraiment été abordé dans cette recherche, il y a encore place à amélioration. Pourrait-on optimiser davantage la convergence des fonctions qui décrivent les propriétés dynamiques d'un système de sorte que ces dernières soient peu affectées lorsque l'ensemble d'OPIs est incomplet ? Aussi, nous avons vu que les méthodes de détection s'avèrent moins efficaces lorsque la dimension du système augmente. Pourrait-on envisager de jumeler une approche par la dynamique symbolique afin de choisir plus judicieusement les conditions initiales utilisées par les algorithmes de détection que nous avons présentés ? Cette dynamique symbolique est difficile à construire. Cependant, même si elle est incomplète, pourrait-elle servir de guide pour choisir plus convenablement les conditions initiales ? Ce sont là des voies explorables pour un travail futur, voies qui pourraient conduire à de meilleures prédictions quant aux propriétés dynamiques et physiques des systèmes chaotiques.

## Annexe A

# Démonstrations de l'algorithme de $SD$ en $2D$

Au chapitre 1.3, nous avons présenté l'algorithme de  $SD$  pour la recherche d'OPIs, algorithme qui nécessite une transformation linéaire du système chaotique initial. Cette transformation utilise une classe de matrices  $\mathbf{C}_k$  particulière. Chaque matrice  $C_k$  a la propriété de détecter une famillle d'OPIs donnée. Dans cette annexe, nous démontrons cette dernière affirmation en reprenant chacun des points du tableau 1.3 de la section 1.3.2.2.

Nous avons vu à la section 1.3.2.1 que la transformation  $SD$  modifie la stabilité des OPIs. La condition pour obtenir des OPIs stables dans le système transformé  $\mathcal{W}_k$  est donnée par :

$$\text{Condition de stabilité : } \boxed{\text{Re}[\rho_d^{A_k}] < 0 \quad [\forall d \in \{1, \dots, D\}]} \quad (\text{A.1})$$

où les  $\rho_d^{A_k}$  sont les valeurs propres de la matrice  $\mathbf{A}_k$ . Rappelons que cette matrice  $\mathbf{A}_k$  est définie par :

$$\mathbf{A}_k \equiv \mathbf{C}_k(\mathbf{J}^{(m)} - \mathbf{1}),$$

où  $\mathbf{J}^{(m)}$  est la matrice jacobienne du système initial.

On peut exprimer la condition de stabilité sous une autre forme. En effet, comme  $\mathbf{A}_k$  est diagonalisable par hypothèse, on écrit de façon générale :

$$\begin{aligned} \text{Det}[\mathbf{A}_k] &= \rho_1^{A_k} \rho_2^{A_k} \\ \text{Tr}[\mathbf{A}_k] &= \rho_1^{A_k} + \rho_2^{A_k}. \end{aligned}$$

En dimension 2 et pour  $\mathbf{A}_k$  réelle (où les valeurs propres sont soit réelles, soit complexes conjuguées), la condition A.1 se ramène à :

Conditions de stabilité en 2D :

$$\text{Det}[\mathbf{A}_k] > 0 \quad (\text{A.2})$$

$$\text{Tr}[\mathbf{A}_k] < 0. \quad (\text{A.3})$$

Pour démontrer les conditions (i) à (iv) du tableau 1.3 de la section 1.3.2.2, nous pouvons donc utiliser la condition A.1 ou les conditions A.2 et A.3 combinées. Nous reprenons maintenant les affirmations du tableau 1.3.

### Démonstration de (i) :

On a une orbite périodique avec  $\rho_1 < -1$  et  $-1 < \rho_2 < 0$ . Ces  $\rho_d$  sont les valeurs propres de la matrice jacobienne  $\mathbf{J}^{(m)}$  évaluée sur le cycle (voir équation 1.35). On veut maintenant démontrer que la matrice  $\mathbf{C}_1 = \mathbf{1}$  est celle qui stabilise l'orbite par le biais de la transformation 1.29. Si on choisit  $\mathbf{C}_k = \mathbf{C}_1 = \mathbf{1}$ , alors la relation entre les valeurs propres de  $\mathbf{A}_k = \mathbf{A}_1$  et de  $\mathbf{J}^{(m)}$  devient simplement :

$$\begin{aligned} \rho_1^{A_1} &= \rho_1 - 1 \Rightarrow \text{Re}\{\rho_1^{A_1}\} < 0 \\ \rho_2^{A_1} &= \rho_2 - 1 \Rightarrow \text{Re}\{\rho_2^{A_1}\} < 0. \end{aligned} \quad (\text{A.4})$$

Les dernières inégalités proviennent du fait que  $\rho_1$  et  $\rho_2$  sont toutes deux inférieures à 1 (par hypothèse). La condition A.1 étant respectée, l'orbite transformée est stable.

**Démonstration de (ii) :**

On procède exactement comme ce qui a été fait pour (i). On arrive rapidement au choix de  $\mathbf{C}_k$  soit  $\mathbf{C}_k = \mathbf{C}_1 = \mathbf{1}$ .

**Démonstration de (iii) :**

L'orbite a comme valeurs propres  $\rho_1 > 1$  et  $-1 < \rho_2 < 1$ . On utilise les conditions A.2 et A.3 pour cette démonstration. On se concentre d'abord sur la première condition :  $\text{Det}[\mathbf{A}_k] > 0$ . On peut écrire ce déterminant en fonction de celui de  $\mathbf{C}_k$  et de  $\mathbf{J}^{(m)}$  :

$$\begin{aligned}\text{Det}[\mathbf{A}_k] &= \text{Det}[\mathbf{C}_k]\text{Det}[\mathbf{J}^{(m)} - \mathbf{1}] \\ &= \text{Det}[\mathbf{C}_k](\rho_1 - 1)(\rho_2 - 1).\end{aligned}$$

Comme  $\rho_1 > 1$  et  $\rho_2 < 1$  on a que  $\text{Det}[\mathbf{J}^{(m)} - \mathbf{1}] < 0$ . Afin de respecter A.2, il faut donc choisir  $\mathbf{C}_k$  telle que  $\text{Det}[\mathbf{C}_k] < 0$ . Les seules matrices  $\mathbf{C}_k$  possibles avec  $\text{Det}(\mathbf{C}_k) < 0$  sont :

$$\begin{array}{ll}\mathbf{C}_2 = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} & \mathbf{C}_4 = \begin{pmatrix} 0 & -1 \\ -1 & 0 \end{pmatrix} \\ \mathbf{C}_3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} & \mathbf{C}_5 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.\end{array} \quad (\text{A.5})$$

On utilise maintenant l'autre condition sur  $\mathbf{A}_k$ , soit  $\text{Tr}[\mathbf{A}_k] < 0$ . Si on choisit  $\mathbf{C}_2$ , on a alors que :

$$\begin{aligned}\mathbf{A}_2 &= \mathbf{C}_2(\mathbf{J}^{(m)} - \mathbf{1}) \\ &= \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{J}_{11}^{(m)} - 1 & \mathbf{J}_{12}^{(m)} \\ \mathbf{J}_{21}^{(m)} & \mathbf{J}_{22}^{(m)} - 1 \end{pmatrix} \\ &= \begin{pmatrix} 1 - \mathbf{J}_{11}^{(m)} & -\mathbf{J}_{12}^{(m)} \\ \mathbf{J}_{21}^{(m)} & \mathbf{J}_{22}^{(m)} - 1 \end{pmatrix}.\end{aligned}$$

On obtient dans ce cas  $Tr[\mathbf{A}_2] = \mathbf{J}_{22}^{(m)} - \mathbf{J}_{11}^{(m)}$ , qui est négatif seulement si  $\mathbf{J}_{11}^{(m)} > \mathbf{J}_{22}^{(m)}$ . Autrement dit, la condition A.3 est respectée avec  $\mathbf{C}_k = \mathbf{C}_2$ , si et seulement si  $\mathbf{J}_{11}^{(m)} > \mathbf{J}_{22}^{(m)}$ . De façon similaire, on démontre que  $\mathbf{C}_3$  est une matrice qui respecte les conditions A.2 et A.3 si et seulement si  $\mathbf{J}_{11}^{(m)} < \mathbf{J}_{22}^{(m)}$ .

Pour le cas limite  $J_{11}^{(m)} = J_{22}^{(m)}$ , on choisit  $\mathbf{C}_k = \mathbf{C}_4$  ou  $\mathbf{C}_5$  et  $\mathbf{A}_k$  prend la forme :

$$\begin{aligned}\mathbf{A}_k &= \mathbf{C}_{4;5}(\mathbf{J}^{(m)} - \mathbf{1}) \\ &= \begin{pmatrix} \pm\mathbf{J}_{12}^{(m)} & \pm(\mathbf{J}_{11}^{(m)} - 1) \\ \pm(\mathbf{J}_{22}^{(m)} - 1) & \pm\mathbf{J}_{21}^{(m)} \end{pmatrix},\end{aligned}$$

le signe + est associé à  $\mathbf{C}_k = \mathbf{C}_5$  et le signe - à  $\mathbf{C}_k = \mathbf{C}_4$ . Si  $\mathbf{J}_{12}^{(m)} + \mathbf{J}_{21}^{(m)} > 0$ , on choisit  $\mathbf{C}_k = \mathbf{C}_4$  afin de respecter le critère  $Tr[\mathbf{A}_k] < 0$ . Si  $\mathbf{J}_{12}^{(m)} + \mathbf{J}_{21}^{(m)} < 0$ , on choisit  $\mathbf{C}_k = \mathbf{C}_5$  de sorte que  $Tr[\mathbf{A}_k] < 0$ .

#### Démonstration de (iv) :

Cette dernière démonstration concerne maintenant les points fixes elliptiques avec  $|\rho_{1,2}| = 1$  et  $Im\{\rho_{1,2}\} \neq 0$ . Ce dernier cas est particulièrement intéressant puisqu'il se retrouve couramment dans les systèmes conservatifs (ou hamiltoniens). Pour ce type d'orbite, les valeurs propres sont complexes et se situent sur le cercle de rayon unitaire. Elles viennent par paires de nombres complexes conjugués. Si  $\rho_{1c}$  et  $\rho_{2c}$  sont les valeurs propres complexes du  $\mathbf{J}^{(m)}$  (l'indice  $c$  indiquant que ces valeurs propres sont complexes), alors on peut écrire :

$$\begin{aligned}\rho_{1c} &= \cos \alpha + i \sin \alpha \\ \rho_{2c} &= \cos \alpha - i \sin \alpha,\end{aligned}\tag{A.6}$$

avec  $0 \leq \alpha \leq \pi$ . On applique maintenant la transformation *SD*. Les équations A.2 et A.3 expriment les contraintes imposées à  $\mathbf{A}_k$  afin de stabiliser un point fixe.

Écrivons le déterminant de  $\mathbf{A}_k$  en fonction des valeurs propres de  $\rho_{1c}$  et  $\rho_{2c}$  (équation

A.6). On obtient :

$$\begin{aligned}
 \text{Det}[\mathbf{A}_k] &= \text{Det}[\mathbf{C}_k] \text{Det}[\mathbf{J}^{(m)} - \mathbf{1}] \\
 &= \text{Det}[\mathbf{C}_k](\rho_{1c} - 1)(\rho_{2c} - 1) \\
 &= 2 * \text{Det}[\mathbf{C}_k](1 - \cos \alpha).
 \end{aligned} \tag{A.7}$$

Sous cette dernière forme, il est clair que pour  $\alpha \neq 0$ , l'équation A.2 est respectée si et seulement si  $\text{Det}[\mathbf{C}_k] > 0$ , ce qui permet de restreindre le choix des matrices  $\mathbf{C}_k$ . Choisissons  $\mathbf{C}_k = \mathbf{C}_1 = \mathbf{1}$ . Avec ce choix particulier, la trace de  $\mathbf{A}_1$  devient :

$$\begin{aligned}
 \text{Tr}[\mathbf{A}_1] &= \text{Tr}[\mathbf{C}_1(\mathbf{J}^{(m)} - \mathbf{1})] \\
 &= \text{Tr}[\mathbf{J}^{(m)} - \mathbf{1}] \\
 &= \rho_{1c} + \rho_{2c} - 2 \\
 &= 2(\cos \alpha - 1).
 \end{aligned} \tag{A.8}$$

Sous cette dernière forme, on remarque que la  $\text{Tr}[\mathbf{A}_1]$  est inférieure à 0 pour  $\alpha \neq 0$ . Le deuxième critère (équation A.3) est ainsi respecté. On conclut donc que, à elle seule, la matrice  $\mathbf{C}_k = \mathbf{C}_1 = \mathbf{1}$  permet de stabiliser toutes les orbites elliptiques des systèmes conservatifs.

Il faut noté que pour le point (iv), on exclue le cas  $|Im\{\rho_{1,2}\}| = 0$  (orbites paraboliques) qui correspond à  $\alpha = 0$  dans la dernière démonstration. Pour ce cas particulier, on remarque que l'équation A.7 donne zéro, peu importe le choix de  $\mathbf{C}_k$ . Ceci implique qu'une valeur propre de  $\mathbf{A}_k$  est zéro. Et si une valeur propre de  $\mathbf{A}_k$  est zéro, le cycle transformé par *SD* reste non-hyperbolique (voir équation 1.32).

Ceci complète la démonstration des points (i) à (iv). Mentionnons que les démonstrations (i) et (ii) pourraient être faites de façon générale soit comme (iii), i.e. en écrivant d'abord les deux conditions sur la matrice  $\mathbf{A}_k$  et en cherchant ensuite les matrices  $\mathbf{C}_k$  qui satisfont ces conditions. On obtient alors que  $\text{Det}[\mathbf{C}_k] > 1$ . Quatre matrices sont possibles, mais  $\mathbf{C}_k = \mathbf{C}_1 = \mathbf{1}$  permet de stabiliser toutes les orbites avec  $\rho_1 < -1$  et  $-1 < \rho_2 < 1$ .

# Annexe B

## Billards : méthode de propagation

Dans cette annexe, nous présentons la description des algorithmes utilisés pour obtenir l'évolution d'une particule dans les Billards Libres et Magnétiques.

Dans un premier temps, nous allons développer les équations nécessaires à cette tâche. Nous discutons aussi des points plus techniques pour l'algorithme du Billard Magnétique.

### B.1 Billards Libres

L'algorithme que nous décrivons ici est celui de Korsch et Jodl [50]. Il permet d'obtenir de façon semi-analytique le comportement dynamique de la particule à l'intérieur du Billard Libre (figue B.1). Les angles  $\varphi$  et  $\alpha$  servent à la description de cette dynamique. L'angle  $\varphi$  est la position de la particule dans la cavité avec  $\varphi \in [0, 2\pi]$ . La variable  $\alpha$  mesure l'angle entre la direction de la vitesse et la tangente positive au point d'impact avec  $\alpha \in ]0, \pi[$ . Si on connaît la position angulaire  $\varphi_n$  de la particule sur le contour ainsi que l'angle d'incidence  $\alpha_n$ , on peut calculer le prochain point dans l'espace des phases  $(\varphi_{n+1}, \alpha_{n+1})$ . Pour ce faire, on calcule d'abord l'angle  $\vartheta_n$  entre la

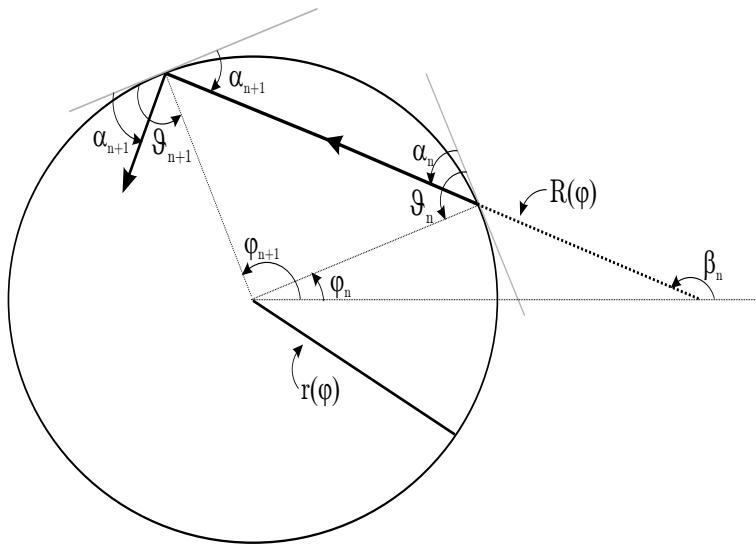


FIG. B.1 – Définition des variables du Billard Libre.

direction positive de la tangente et la direction radiale au point d'impact  $n$  :

$$\tan \vartheta_n = \left. \frac{r(\varphi)}{dr/d\varphi} \right|_n. \quad (\text{B.1})$$

Le mouvement dans le sens anti-horaire correspond à  $\alpha_n < \vartheta_n$  et le sens horaire à  $\alpha_n > \vartheta_n$ . On définit ensuite l'angle  $\beta_n$  comme l'angle entre la trajectoire rectiligne au  $n^{i\text{ème}}$  rebond et la direction  $\varphi = 0$  :

$$\beta_n = \pi + \varphi_n + \alpha_n - \vartheta_n. \quad (\text{B.2})$$

En coordonnées polaires, cette trajectoire est donnée par :

$$R(\varphi) = r(\varphi_n) \frac{\sin(\beta_n - \varphi_n)}{\sin(\beta_n - \varphi)}. \quad (\text{B.3})$$

Le prochain impact a lieu à l'endroit où la trajectoire  $R(\varphi)$  devient égale au contour de la cavité ( $r(\varphi)$ ). Il faut donc résoudre numériquement :

$$R(\varphi) - r(\varphi) = 0. \quad (\text{B.4})$$

Le prochain rebond  $\varphi_{n+1}$  se fait au zéro (autre que  $\varphi_n$ ) de cette équation. Le prochain angle d'incidence est donné par

$$\alpha_{n+1} = \varphi_{n+1} - \varphi_n + \vartheta_n - \vartheta_{n+1} - \alpha_n. \quad (\text{B.5})$$

Cette méthode est rapide car il n'y a pas d'intégration numérique à faire pour obtenir la dynamique. Mentionnons aussi que l'équation B.4 peut se résoudre par un algorithme de Newton-Raphson borné : la solution doit être comprise entre 0 et  $2\pi$  et ne peut être  $\varphi_n$  [76][§9.4].

## B.2 Billards Magnétiques

### B.2.1 Techniques numériques

Nous voulons ici présenter une méthode afin d'obtenir l'évolution d'une particule chargée dans un billard magnétique. Rappelons que la dynamique du billard est décrite par deux variables soit  $\varphi_n$ , la position angulaire de la particule lors du  $n^{ième}$  rebond, et  $\alpha_n$ , l'angle de réflexion mesuré par rapport à la tangente de la cavité au  $n^{ième}$  rebond. Le but est de trouver  $\varphi_{n+1}$  et  $\alpha_{n+1}$  en fonction de  $\varphi_n$  et  $\alpha_n$ .

Il faut modifier quelque peu l'algorithme utilisé dans le cas du Billard Libre. La particule dans le Billard Magnétique décrit, non plus une ligne droite entre chaque rebond, mais bien un arc de cercle de rayon  $b$ , correspondant au rayon de Larmor :

$$b = \frac{mv}{q\mathcal{B}}, \quad (\text{B.6})$$

où  $m$  est la masse de la particule,  $v$  sa vitesse,  $q$  sa charge et  $\mathcal{B}$ , le champ magnétique externe. Ce champ est uniforme et perpendiculaire à la surface du billard. Par convention, le sens positif du champ magnétique “entre” dans la surface. Le rayon de Larmor est normalisé par rapport au rayon de la cavité.

La courbe limitant le billard, donnée par  $r(\varphi)$ , est connue de façon analytique. La position de l'impact  $n$  aura lieu à l'intersection de cette courbe avec la trajectoire circulaire de rayon de Larmor  $R(\varphi)$  (réf. figure B.2). Si on peut exprimer cette trajectoire de façon analytique, il sera ensuite possible de trouver le point d'impact en solutionnant

l'équation suivante pour  $\varphi$  :

$$R(\varphi) - r(\varphi) = 0. \quad (\text{B.7})$$

Pour l'instant, nous allons d'abord trouver une façon d'écrire la trajectoire circulaire  $R(\varphi)$  en fonction des paramètres connus. Ces paramètres sont  $\varphi_n$ ,  $\alpha_n$  et  $\beta_n$  (équation B.2). Sur la figure B.2, nous avons tracé la trajectoire juste avant l'impact (---) ainsi que celle juste après celui-ci (—). Le centre de la trajectoire circulaire après l'impact  $n$  (noté  $C_{\text{après}}^{(n)}$ ) est donné en coordonnées cartésiennes par :

$$\begin{aligned} X_{\text{après}}^{(n)} &= r(\varphi_n) \cos(\varphi_n) - b \sin(\beta_n) \\ Y_{\text{après}}^{(n)} &= r(\varphi_n) \sin(\varphi_n) + b \cos(\beta_n) \end{aligned} \quad (\text{B.8})$$

Nous aurons besoin un peu plus loin de la position du centre de la trajectoire circulaire avant l'impact. Cette position est :

$$\begin{aligned} X_{\text{avant}}^{(n)} &= r(\varphi_n) \cos(\varphi_n) - b \sin(\beta_n - 2\alpha_n) \\ Y_{\text{avant}}^{(n)} &= r(\varphi_n) \sin(\varphi_n) + b \cos(\beta_n - 2\alpha_n) \end{aligned} \quad (\text{B.9})$$

L'équation de la trajectoire circulaire après l'impact est donnée par

$$(X - X_{\text{après}}^{(n)})^2 + (Y - Y_{\text{après}}^{(n)})^2 = b^2. \quad (\text{B.10})$$

En coordonnées polaires, cette dernière équation devient :

$$R^2(\varphi) - 2R(\varphi) [X_{\text{après}}^{(n)} \cos(\varphi) + Y_{\text{après}}^{(n)} \sin(\varphi)] + [X_{\text{après}}^{(n)}]^2 + [Y_{\text{après}}^{(n)}]^2 - b^2 = 0. \quad (\text{B.11})$$

En posant  $B \equiv X_{\text{après}}^{(n)} \cos(\varphi) + Y_{\text{après}}^{(n)} \sin(\varphi)$  et  $C \equiv [X_{\text{après}}^{(n)}]^2 + [Y_{\text{après}}^{(n)}]^2 - b^2$ , on obtient finalement une expression de la trajectoire de la particule en fonction de  $\varphi$  :

$$R(\varphi) = -B \pm \sqrt{B^2 - 4C}. \quad (\text{B.12})$$

La position de l'impact ( $\varphi_{n+1}$ ) s'obtient en solutionnant l'équation B.7 pour une cavité donnée ( $r(\varphi)$ ) et une trajectoire circulaire  $R(\varphi)$  (éq. B.12). Un algorithme de type Newton-Raphson peut être efficace pour cette dernière tâche. Il faut cependant avoir

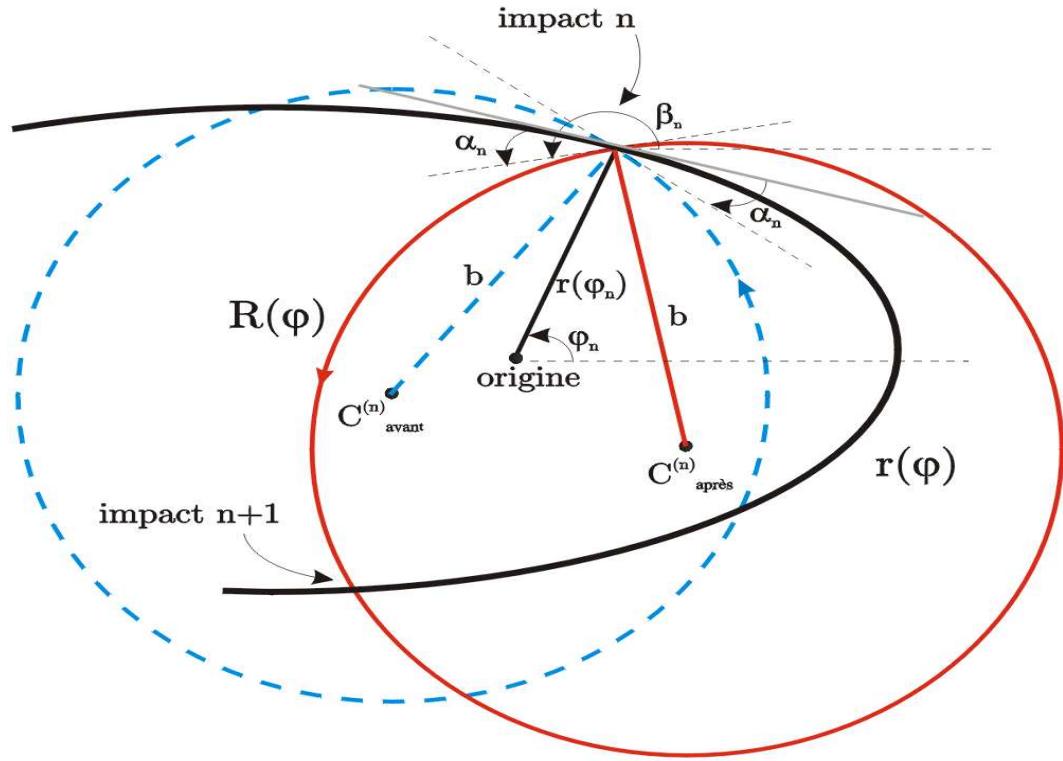


FIG. B.2 – Billard magnétique en deux dimensions.

en main une expression pour  $\frac{dR}{d\varphi} - \frac{dr}{d\varphi}$ . En dérivant l'équation B.11 implicitement on trouve :

$$\frac{dR}{d\varphi} = \frac{R \left[ Y_{après}^{(n)} \cos(\varphi) - X_{après}^{(n)} \sin(\varphi) \right]}{R - \left[ X_{après}^{(n)} \cos(\varphi) + Y_{après}^{(n)} \sin(\varphi) \right]}. \quad (\text{B.13})$$

Il faut maintenant déterminer  $\alpha_{n+1}$  en fonction des paramètres connus. On sait que la position du centre de la trajectoire circulaire après l'impact  $n$  est la même que celle avant l'impact  $(n+1)$ . Clairement, on peut écrire que  $C_{après}^{(n)} = C_{avant}^{(n+1)}$ . En utilisant les équations B.9 et B.8 on obtient :

$$\begin{aligned} X_{après}^{(n)} &= r(\varphi_{n+1}) \cos(\varphi_{n+1}) - b \sin(\beta_{n+1} - 2\alpha_{n+1}) \\ Y_{après}^{(n)} &= r(\varphi_{n+1}) \sin(\varphi_{n+1}) + b \cos(\beta_{n+1} - 2\alpha_{n+1}) \end{aligned} . \quad (\text{B.14})$$

Ces dernières équations font intervenir l'angle recherché soit  $\alpha_{n+1}$ . L'angle  $\beta_{n+1}$  que

l'on retrouve dans ces équations dépend aussi de  $\alpha_{n+1}$  (éq. B.2). On obtient finalement :

$$\alpha_{n+1} = \pi + \varphi_{n+1} - \vartheta_{n+1} - \arctan \left( \frac{r \cos(\varphi_{n+1}) - X_{\text{après}}^{(n)}}{Y_{\text{après}}^{(n)} - r \sin(\varphi_{n+1})} \right). \quad (\text{B.15})$$

### B.2.2 Un mot sur l'algorithme

À la section précédente, nous avons présenté les équations nécessaires pour obtenir la suite d'itérés  $(\varphi_n, \alpha_n)$ . L'implémentation de ces équations se fait sans difficulté. Il faut cependant être prudent lors du calcul de  $\varphi_{n+1}$  (éq. B.7). Ce calcul se fait à l'aide d'un algorithme de Newton-Raphson borné ([76, page 366]). Les bornes déterminent l'intervalle des valeurs de  $\varphi$  à l'intérieur duquel se trouve la solution. Deux cas peuvent se présenter (réf. figure B.3) :

I- L'origine de la cavité est à l'intérieur du “ cercle magnétique ”. On cherche alors  $\varphi_{n+1}$  entre  $\varphi_n + \delta$  et  $\varphi_n - \delta + 2\pi$ . La solution recherchée ( $\varphi_{n+1}$ ) doit être comprise dans le cercle et ne peut évidemment pas être  $\varphi_n$ . On ajoute donc un  $\delta$  ( $\sim 10^{-6}$ ) de part et d'autre de  $\varphi_n$  afin d'éviter de retrouver cette solution.

II- L'origine de la cavité est à l'extérieur du “ cercle magnétique ”. La solution doit maintenant être cherchée entre  $\varphi_{\min}$  et  $\varphi_{\max}$ , le cercle magnétique n'étant pas défini pour les autres valeurs de  $\varphi$ .

Ce dernier cas doit être détaillé davantage. En particulier, il faut exprimer les valeurs de  $\varphi_{\min}$  et  $\varphi_{\max}$  en fonction des paramètres connus. En utilisant la figure B.3 comme référence, on obtient directement l'expression pour ces deux angles :

- Pour B positif (entrant dans la feuille) :

$$\begin{aligned} \varphi_{\max} &= \varphi_n - \delta \\ \varphi_{\min} &= \arctan \left( \frac{Y_{\text{après}}^{(n)}}{X_{\text{après}}^{(n)}} \right) - \xi + \delta \end{aligned} \quad (\text{B.16})$$

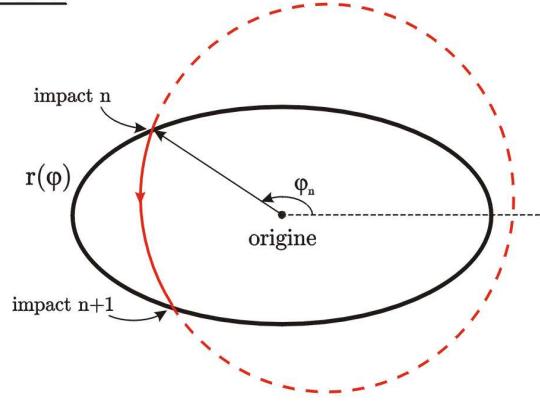
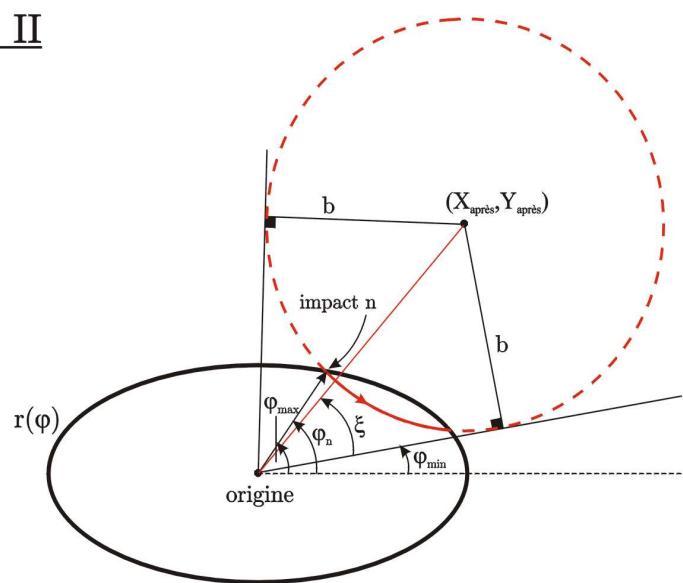
avec  $\xi \equiv \arcsin \left( b / \sqrt{X_{\text{après}}^2 + Y_{\text{après}}^2} \right)$ . Ceci correspond au cas où la particule est déviée à sa gauche par le champ magnétique (C'est le cas représenté sur la figure B.3 - Cas II).

- Pour  $B$  négatif (sortant de la feuille) :

$$\begin{aligned}\varphi_{min} &= \varphi_n + \delta \\ \varphi_{max} &= \arctan \left( \frac{Y_{\text{après}}^{(n)}}{X_{\text{après}}^{(n)}} \right) + \xi - \delta\end{aligned}. \quad (\text{B.17})$$

Ceci correspond au cas où la particule est déviée à sa droite par le champ magnétique.

On ajoute un  $\pm \delta$  aux expressions  $\arctan \left( \frac{Y_{\text{après}}^{(n)}}{X_{\text{après}}^{(n)}} \right) \mp \xi$  afin d'éviter des problèmes numériques.

Cas ICas IIFIG. B.3 – Recherche de  $\varphi_{n+1}$  : géométrie du problème.

## Annexe C

# Application discrète à partir du flot DKP

Nous décrivons dans cette annexe comment implémenter la procédure décrite à la section 2.4.1 pour le flot DKP. L'ensemble de la démarche permet d'obtenir une application discrète  $\mathbf{F}(\mathbf{v}_n)$  de dimension  $D = 2$  ainsi que la matrice jacobienne de cette application. Un aide-mémoire sous forme de tableau est aussi donné à la fin de l'annexe.

Les équations de la dynamique pour DKP sont :

$$\frac{d\tilde{\mathbf{v}}}{dt} = \tilde{\mathbf{f}}(\tilde{\mathbf{v}})$$

soit :

$$\text{DKP : } \frac{d\tilde{\mathbf{v}}}{dt} = \begin{pmatrix} \frac{d\tilde{v}^1}{dt} \\ \frac{d\tilde{v}^2}{dt} \\ \frac{d\tilde{v}^3}{dt} \\ \frac{d\tilde{v}^4}{dt} \end{pmatrix} \equiv \begin{pmatrix} \frac{d\nu}{dt} \\ \frac{dp_\nu}{dt} \\ \frac{d\mu}{dt} \\ \frac{dp_\mu}{dt} \end{pmatrix} = \begin{pmatrix} p_\nu \\ \nu(2\epsilon - \frac{1}{4}\mu^4 - \frac{1}{2}\nu^2\mu^2) \\ p_\mu \\ \mu(2\epsilon - \frac{1}{4}\nu^4 - \frac{1}{2}\nu^2\mu^2) \end{pmatrix}. \quad (\text{C.1})$$

avec les équations variationnelles données par :

$$\frac{d\tilde{\mathbf{J}}_{\tilde{\phi}}}{dt} = D_{\tilde{\mathbf{v}}}\tilde{\mathbf{f}}(\tilde{\mathbf{v}}) \tilde{\mathbf{J}}_{\tilde{\phi}}(t, \tilde{\mathbf{v}}),$$

soit :

$$\frac{d\tilde{\mathbf{J}}_{\tilde{\phi}}}{dt} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 2\epsilon - \frac{1}{4}\mu^4 - \frac{3}{2}\mu^2\nu^2 & 0 & -\mu\nu(\mu^2 + \nu^2) & 0 \\ 0 & 0 & 0 & 1 \\ -\mu\nu(\mu^2 + \nu^2) & 0 & 2\epsilon - \frac{1}{4}\nu^4 - \frac{3}{2}\mu^2\nu^2 & 0 \end{pmatrix} \tilde{\mathbf{J}}_{\tilde{\phi}}(t, \tilde{\mathbf{v}}).$$

Ces dernières dérivées ajoutent 16 équations différentielles au système initial. En utilisant la notation :

$$\begin{aligned} \hat{\nu}^i &= \frac{\partial \nu}{\partial \tilde{v}^i} & \hat{p}_\nu^i &= \frac{\partial p_\nu}{\partial \tilde{v}^i} \\ \hat{\mu}^i &= \frac{\partial \mu}{\partial \tilde{v}^i} & \hat{p}_\mu^i &= \frac{\partial p_\mu}{\partial \tilde{v}^i} \end{aligned},$$

les 16 équations variationnelles s'écrivent :

$$\begin{aligned} \frac{d\hat{\nu}^i}{dt} &= \hat{p}_\nu^i \\ \frac{d\hat{p}_\nu^i}{dt} &= (2\epsilon - \frac{1}{4}\mu^4 - \frac{3}{2}\mu^2\nu^2)\hat{\nu}^i - \mu\nu(\mu^2 + \nu^2)\hat{\mu}^i \\ \frac{d\hat{\mu}^i}{dt} &= \hat{p}_\mu^i \\ \frac{d\hat{p}_\mu^i}{dt} &= -\mu\nu(\mu^2 + \nu^2)\hat{\nu}^i + (2\epsilon - \frac{1}{4}\nu^4 - \frac{3}{2}\mu^2\nu^2)\hat{\mu}^i \end{aligned}, \quad (\text{C.2})$$

avec  $i = \{1, \dots, 4\}$ . Il faut donc intégrer les équations C.1 et C.2 simultanément (au total, 20 équations) à partir d'un  $\tilde{\mathbf{v}}_n$  donné. Les conditions initiales pour les équations variationnelles sont :

$$\tilde{\mathbf{J}}_{\tilde{\phi}}(t = 0, \tilde{\mathbf{v}} = \tilde{\mathbf{v}}_n) = \mathbf{1},$$

soit :

$$\hat{\nu}^i = \delta_{i1} \quad \hat{p}_\nu^i = \delta_{i2}$$

$$\hat{\mu}^i = \delta_{i3} \quad \hat{p}_\mu^i = \delta_{i4}$$

avec  $\delta_{ij} = 1$  si  $i = j$  et  $\delta_{ij} = 0$  sinon.

Nous devons maintenant choisir la Section de Poincaré, qui est définie par la fonction  $h(\tilde{\mathbf{v}}(t))$  (équation 2.33). Une section “naturelle” pour le système DKP est de prendre

$\mu = 0$  ou  $\nu = 0$ . Nous choisissons  $\mu = 0$  ce qui impose  $\tilde{\xi}^T = (0, 0, 1, 0)$  et un point  $\tilde{\mathbf{v}}_0$  sur la section donné par  $\tilde{\mathbf{v}}_0^T = (\nu_0, p_{\nu_0}, 0, p_{\mu_0})$  :

$$\begin{aligned} h(\tilde{\mathbf{v}}(t)) &= \tilde{\xi}^T (\tilde{\mathbf{v}}(t) - \tilde{\mathbf{v}}_0) \\ h(\mu, \nu, p_\mu, p_\nu) &= (0, 0, 1, 0) \begin{pmatrix} \nu(t) - \nu_0 \\ p_\nu(t) - p_{\nu_0} \\ \mu(t) - 0 \\ p_\mu(t) - p_{\mu_0} \end{pmatrix} \\ h(\mu(t)) &= \mu(t). \end{aligned}$$

Sur la Section de Poincaré, la fonction  $h$  est égale à zéro et on reconnaît alors l'équation du plan choisi au départ ( $\mu_n = 0$ ). Ceci nous amène une première équation de contrainte :

$$\text{Première contrainte : } \mu_n = 0 \quad (\text{Section de Poincaré}). \quad (\text{C.3})$$

De plus, on sait que pour le flot, l'énergie est conservée. Cette énergie est donnée par le (pseudo)-Hamiltonien (réf. équation 2.50) et nous procure une deuxième équation de contrainte :

$$\text{Deuxième contrainte : } \hat{h} \equiv 2 = \frac{1}{2}(p_\mu^2 + p_\nu^2) + V(\mu, \nu) \quad (\text{Conservation de l'énergie}),$$

où  $V(\mu, \nu) = \epsilon(\mu^2 + \nu^2) + \frac{1}{8}\mu^2\nu^2(\mu^2 + \nu^2)$ . Avec cette contrainte, nous choisissons d'éliminer la variable  $p_\mu$  :

$$p_\mu = \sqrt{2(\hat{h} - V(\mu, \nu)) - p_\nu},$$

et sur la Section de Poincaré nous avons :

$$p_{\mu n} = \sqrt{2(\hat{h} - V(\mu_n = 0, \nu_n)) - p_{\nu n}}. \quad (\text{C.4})$$

Pour reprendre les notations que nous avons utilisées à la section 2.4.1, pour le flot DKP, nous avons  $N_c = 2$  équations de contraintes. Ces équations permettent d'exprimer les variables que l'on élimine ( $\check{\mathbf{v}}_n = \{\mu_n, p_{\mu n}\}$ ) en fonction des variables que l'on garde (les  $\mathbf{v}_n = \{\nu_n, p_{\nu n}\}$ ). Ce sont les équations C.3 et C.4. Les variables que l'on élimine doivent être exprimées uniquement en fonction des variables que l'on garde.

Le système discret que nous obtenons se compose ainsi des variables  $\nu_n$  et  $p_{\nu_n}$ . Mentionnons que ce couple de variables, sur la section que nous avons choisie, conserve les aires : la matrice jacobienne de la dynamique discrète doit avoir un déterminant égal à 1 [38][§7] (ceci n'est pas nécessairement vrai pour une autre Section de Poincaré). Le vecteur de dimension 4  $\tilde{\mathbf{v}}_n$  qui se compose des deux sous-vecteurs  $\mathbf{v}_n$  et  $\check{\mathbf{v}}_n$  s'écrit donc :

$$\tilde{\mathbf{v}}_n = (\mathbf{v}_n, \check{\mathbf{v}}_n)^T = \begin{pmatrix} v_n^1 \\ v_n^2 \\ \check{v}_n^1(\mathbf{v}_n) \\ \check{v}_n^2(\mathbf{v}_n) \end{pmatrix} \equiv \begin{pmatrix} \nu_n \\ p_{\nu_n} \\ \mu_n \\ p_{\mu_n}(\nu_n, p_{\nu_n}) \end{pmatrix}.$$

La prochaine étape consiste à obtenir l'application discrète sur la Section de Poincaré. On utilise la méthode de Hénon. La variable  $u \equiv h(\tilde{\mathbf{v}}(t))$  est donnée par  $u = \mu(t)$  et :

$$\frac{du(t)}{dt} = \tilde{f}^3(\tilde{\mathbf{v}}) = p_\mu. \quad (\text{C.5})$$

C'est en fait l'équation 2.35 avec notre choix de  $\xi^T = (0, 0, 1, 0)$ . Les équations de la dynamique en fonction de cette nouvelle variable (réf. équation 2.36) sont :

$$\frac{d\tilde{\mathbf{v}}^+}{du} = \frac{d\tilde{\mathbf{v}}^+}{dt} \frac{dt}{du} = \begin{pmatrix} \frac{d\nu}{du} \\ \frac{dp_\nu}{du} \\ \frac{d\mu}{du} \\ \frac{dp_\mu}{du} \\ \frac{dt}{du} \end{pmatrix} = \begin{pmatrix} \frac{p_\nu}{p_\mu} \\ \frac{\nu(2\epsilon - \frac{1}{4}\mu^4 - \frac{1}{2}\nu^2\mu^2)}{p_\mu} \\ 1 \\ \frac{\mu(2\epsilon - \frac{1}{4}\nu^4 - \frac{1}{2}\nu^2\mu^2)}{p_\mu} \\ \frac{1}{p_\mu} \end{pmatrix}. \quad (\text{C.6})$$

et les équations variationnelles s'écrivent en fonction de  $u = \mu(t)$  :

$$\begin{aligned} \frac{d\hat{\nu}^i}{du} &= \frac{d\hat{\nu}^i}{dt} \frac{dt}{du} = \frac{\hat{p}_\nu^i}{p_\mu} \\ \frac{d\hat{p}_\nu^i}{du} &= \frac{d\hat{p}_\nu^i}{dt} \frac{dt}{du} = \frac{(2\epsilon - \frac{1}{4}\mu^4 - \frac{3}{2}\mu^2\nu^2)\hat{\nu}^i - \mu\nu(\mu^2 + \nu^2)\hat{\mu}^i}{p_\mu} \\ \frac{d\hat{\mu}^i}{du} &= \frac{d\hat{\mu}^i}{dt} \frac{dt}{du} = \frac{\hat{p}_\mu^i}{p_\mu} \\ \frac{d\hat{p}_\mu^i}{du} &= \frac{d\hat{p}_\mu^i}{dt} \frac{dt}{du} = \frac{-\mu\nu(\mu^2 + \nu^2)\hat{\nu}^i + (2\epsilon - \frac{1}{4}\nu^4 - \frac{3}{2}\mu^2\nu^2)\hat{\mu}^i}{p_\mu} \end{aligned} . \quad (\text{C.7})$$

Pour obtenir numériquement la prochaine itération  $\tilde{\mathbf{v}}_{n+1}$ , on procède alors comme suit :

1. On choisit une condition initiale  $(\nu_n, p_{\nu_n})^T$ . À partir de nos équations de contraintes C.3 et C.4, on déduit les  $\check{\mathbf{v}} = (\mu_n, p_{\mu_n})^T$ . On intègre ensuite simultanément les équations C.1 et C.2 (au total, 20 équations) jusqu'à ce que la fonction  $h = \mu(t)$  change de signe 2 fois (passe d'abord de positif à négatif en traversant la section dans le sens négatif et ensuite de négatif à positif lorsque le flot traverse cette section dans le sens positif). On arrête d'intégrer *juste après* avoir traversé la section dans le sens positif.
2. À ce point on est en  $\tilde{\mathbf{v}}(t^*)$  avec  $t^* > \tau_n$  et  $u = h(\tilde{\mathbf{v}}(t^*))$  positif. Rappelons que  $u = 0$  sur la section de Poincaré. On change maintenant de variable d'intégration. On utilise  $u = \mu(t)$  à la place de  $t$  (équation C.6 et C.7). Il faut intégrer de  $\mu = \mu(t^*)$  à  $\mu = 0$ .
3. On propose un pas  $du = d\mu = -\mu(t^*)$  à l'intégrateur. Si le critère de précision est respecté, alors le système sera amené d'un seul pas à  $u = \mu = 0$  et à  $\tilde{\mathbf{v}}_{n+1} = (\nu_{n+1}, p_{\nu_{n+1}})^T$ . Le temps  $dt^*$  correspondant au pas  $du$  est donné par la 5*ième* variable du système d'équations C.1. Ainsi,  $\tau_n = t^* + dt^*$ . Si le critère de précision n'est pas respecté, on diminue simplement la grandeur du pas et on effectue alors plus d'un pas pour tomber sur  $\mu = 0$ .

Nous avons maintenant en main une condition  $\tilde{\mathbf{v}}_n$ , son itéré sur la section de Poincaré  $\tilde{\mathbf{v}}_{n+1}$  ainsi que le temps d'intégration  $\tau_n$  entre ces deux itérés. Nous passons au calcul de la matrice jacobienne. On se reporte à l'équation 2.45 que l'on réécrit ici :

$$\mathbf{J}(\mathbf{v}_n) = D_{\mathbf{v}}\phi\Big|_{\tilde{\mathbf{v}}_n, \tau_n} + [\mathbf{f}(\tilde{\phi})D_{\mathbf{v}}\tau]_{\tilde{\mathbf{v}}_n, \tau_n} + \sum_{k=1}^{N_c} \left\{ \frac{\partial \phi}{\partial \check{v}^k} + \mathbf{f}(\tilde{\phi}) \frac{\partial \tau}{\partial \check{v}^k} \right\}_{\tilde{\mathbf{v}}_n, \tau_n} D_{\mathbf{v}}\check{v}^k\Big|_{\mathbf{v}_n}.$$

On connaît déjà le premier terme  $D_{\mathbf{v}}\phi\Big|_{\tilde{\mathbf{v}}_n, \tau_n}$ ; il s'agit de 4 éléments  $\{\hat{\nu}^1, \hat{\nu}^2, \hat{p}_{\mu}^1, \hat{p}_{\mu}^2\}$  de la matrice  $4 \times 4$  que nous avons intégrée (réf. équation C.2) pendant un temps  $\tau_n$  :

$$D_{\mathbf{v}}\phi\Big|_{\tilde{\mathbf{v}}_n, \tau_n} \equiv \begin{pmatrix} \frac{\partial \tilde{\phi}^1}{\partial \tilde{v}^1} & \frac{\partial \tilde{\phi}^1}{\partial \tilde{v}^2} \\ \frac{\partial \tilde{\phi}^2}{\partial \tilde{v}^1} & \frac{\partial \tilde{\phi}^2}{\partial \tilde{v}^2} \end{pmatrix}_{\tilde{\mathbf{v}}_n, \tau_n} \equiv \begin{pmatrix} \hat{\nu}^1 & \hat{\nu}^2 \\ \hat{p}_{\mu}^1 & \hat{p}_{\mu}^2 \end{pmatrix}_{\tilde{\mathbf{v}}_n, \tau_n}.$$

On passe ensuite au deuxième terme :

$$\mathbf{f}(\tilde{\phi})|_{\mathbf{v}_n, \tau_n} = \mathbf{f}(\tilde{\mathbf{v}}_{n+1}) \equiv \begin{pmatrix} f^1(\mathbf{v}_{n+1}) \\ f^2(\mathbf{v}_{n+1}) \end{pmatrix},$$

et on utilise directement C.1 pour l'évaluer. Les dérivées  $D_{\tilde{\mathbf{v}}}\tau|_{\tilde{\mathbf{v}}_n, \tau_n}$  se calculent à l'aide de l'équation 2.40 (ici  $\tilde{\boldsymbol{\xi}}^T = (0, 0, 1, 0)$ ) :

$$\begin{aligned} D_{\tilde{\mathbf{v}}}\tau|_{\tilde{\mathbf{v}}_n, \tau_n} &\equiv \left( \frac{\partial \tau}{\partial \nu} \quad \frac{\partial \tau}{\partial p_\nu} \quad \frac{\partial \tau}{\partial \mu} \quad \frac{\partial \tau}{\partial p_\mu} \right)_{\tilde{\mathbf{v}}_n, \tau_n} \\ &= - \left[ \frac{\tilde{\boldsymbol{\xi}}^T D_{\tilde{\mathbf{v}}}\tilde{\phi}}{\tilde{\boldsymbol{\xi}}^T \tilde{\mathbf{f}}(\tilde{\phi})} \right]_{\tilde{\mathbf{v}}_n, \tau_n} \quad (\text{équation 2.40}) \\ &= \frac{(0, 0, 1, 0) \begin{pmatrix} \hat{\nu}^1 & \hat{\nu}^2 & \hat{\nu}^3 & \hat{\nu}^4 \\ \hat{p}_\nu^1 & \hat{p}_\nu^2 & \hat{p}_\nu^3 & \hat{p}_\nu^4 \\ \hat{\mu}^1 & \hat{\mu}^2 & \hat{\mu}^3 & \hat{\mu}^4 \\ \hat{p}_\mu^1 & \hat{p}_\mu^2 & \hat{p}_\mu^3 & \hat{p}_\mu^4 \end{pmatrix}_{\tilde{\mathbf{v}}_n, \tau_n}}{(0, 0, 1, 0) \begin{pmatrix} \tilde{f}^1(\tilde{\mathbf{v}}_{n+1}) \\ \tilde{f}^2(\tilde{\mathbf{v}}_{n+1}) \\ \tilde{f}^3(\tilde{\mathbf{v}}_{n+1}) \\ \tilde{f}^4(\tilde{\mathbf{v}}_{n+1}) \end{pmatrix}} \\ &= \frac{1}{\tilde{f}^3(\tilde{\mathbf{v}}_{n+1})} \begin{pmatrix} \hat{\mu}^1 & \hat{\mu}^2 & \hat{\mu}^3 & \hat{\mu}^4 \end{pmatrix}_{\tilde{\mathbf{v}}_n, \tau_n} \\ &= \frac{1}{p_{\mu_{n+1}}} \begin{pmatrix} \hat{\mu}^1 & \hat{\mu}^2 & \hat{\mu}^3 & \hat{\mu}^4 \end{pmatrix}_{\tilde{\mathbf{v}}_n, \tau_n}. \end{aligned} \quad (\text{C.8})$$

Le vecteur  $D_{\mathbf{v}}\tau|_{\tilde{\mathbf{v}}_n, \tau_n}$  qui compose le deuxième terme de la matrice jacobienne est un sous vecteur de cette dernière équation :

$$D_{\mathbf{v}}\tau|_{\tilde{\mathbf{v}}_n, \tau_n} \equiv \left( \frac{\partial \tau}{\partial \nu} \quad \frac{\partial \tau}{\partial p_\nu} \right)_{\tilde{\mathbf{v}}_n, \tau_n} = \frac{1}{p_{\mu_{n+1}}} \begin{pmatrix} \hat{\mu}^1 & \hat{\mu}^2 \end{pmatrix}_{\tilde{\mathbf{v}}_n, \tau_n}. \quad (\text{C.9})$$

Finalement, il nous reste à évaluer le terme de  $\sum_k$  de l'équation donnant la matrice **J**. Ici nous avons deux équations de contraintes (C.3 et C.4) :

$$\begin{aligned} \check{v}_n^1 &\equiv \mu_n = 0 \\ \check{v}_n^2 &\equiv p_{\mu n} = \sqrt{2(\hat{h} - V(\mu_n = 0, \nu_n)) - p_{\nu n}} \end{aligned}.$$

La première variable éliminée par la contrainte  $\check{v}_n^1 \equiv \mu_n$  ne dépend pas des  $\mathbf{v}_n = (\nu_n, p_{\nu_n})^T$  et nous avons :

$$D_{\mathbf{v}} \check{v}_n^1 \Big|_{\mathbf{v}_n} \equiv D_{\mathbf{v}} \mu_n \Big|_{\mathbf{v}_n} = 0.$$

ce qui implique que le premier terme de la somme sur  $k$  est 0. Ceci est vrai en général si on élimine une variable par une Section de Poincaré qui est “perpendiculaire” à cette variable (soit  $\check{v}_n^1 = \text{constante}$ ). Pour le deuxième terme de la somme nous avons :

$$\begin{aligned} D_{\mathbf{v}} \check{v}_n^2 \Big|_{\mathbf{v}_n} &\equiv D_{\mathbf{v}} p_{\mu} \Big|_{\mathbf{v}_n} \\ &= \left( \frac{\partial p_{\mu}}{\partial \nu} \quad \frac{\partial p_{\mu}}{\partial p_{\nu}} \right)_{\mathbf{v}_n} \\ &= \left[ \frac{1}{p_{\mu}} \left( -\frac{\partial V}{\partial \nu} \quad -p_{\nu} \right) \right]_{\mathbf{v}_n} \\ &= \frac{1}{p_{\mu_n}} \left( \tilde{f}^2([\mu = 0], \nu_n) \quad -p_{\nu_n} \right), \end{aligned} \quad (\text{C.10})$$

où nous avons utilisé une des propriétés des systèmes hamiltoniens soit  $\tilde{f}^2 \equiv \frac{dp_{\nu}}{dt} = -\frac{\partial V}{\partial \nu}$ . Il faut aussi noter que ces dernières dérivées sont évaluées *au début* de l’intégration (soit en  $\mathbf{v}_n$ ). Le dernier terme de la matrice jacobienne  $\mathbf{J}$  devient :

$$\begin{aligned} \sum_{k=1}^{N_c} \left\{ \frac{\partial \phi}{\partial \check{v}^k} + \mathbf{f}(\tilde{\phi}) \frac{\partial \tau}{\partial \check{v}^k} \right\}_{\tilde{\mathbf{v}}_n, \tau_n} D_{\mathbf{v}} \check{v}^k \Big|_{\mathbf{v}_n} &= \left( \begin{array}{c} \frac{\partial \phi^1}{\partial \check{v}^2} + f^1(\tilde{\phi}) \frac{\partial \tau}{\partial \check{v}^2} \\ \frac{\partial \phi^2}{\partial \check{v}^2} + f^2(\tilde{\phi}) \frac{\partial \tau}{\partial \check{v}^2} \end{array} \right)_{\tilde{\mathbf{v}}_n, \tau_n} \left( \begin{array}{cc} \frac{\partial \check{v}^2}{\partial v^1} & \frac{\partial \check{v}^2}{\partial v^2} \end{array} \right)_{\mathbf{v}_n} \\ &= \left( \begin{array}{c} \hat{v}^4 + f^1(\tilde{\phi}) \frac{\partial \tau}{\partial p_{\mu}} \\ \hat{p}_{\nu}^4 + f^2(\tilde{\phi}) \frac{\partial \tau}{\partial p_{\mu}} \end{array} \right)_{\tilde{\mathbf{v}}_n, \tau_n} \left( \begin{array}{cc} \frac{\partial p_{\mu}}{\partial \nu} & \frac{\partial p_{\mu}}{\partial p_{\nu}} \end{array} \right)_{\mathbf{v}_n}. \end{aligned}$$

Nous n’écrivons pas explicitement le produit entre ces quantités mais mentionnons qu’il s’agit d’un produit entre un vecteur colonne et un vecteur ligne, ce qui donne une matrice  $2 \times 2$ . Chaque terme de ce produit a déjà été calculé : les  $\{\hat{v}^4, \hat{p}_{\nu}^4\}$  sont des éléments de la matrice variationnelle (équation C.2),  $\frac{\partial \tau}{\partial p_{\mu}}$  est la 4<sup>ième</sup> composante du vecteur  $D_{\tilde{\mathbf{v}}} \tau \Big|_{\tilde{\mathbf{v}}_n, \tau_n}$  de l’équation C.8 et les dérivées  $\{\frac{\partial p_{\mu}}{\partial \nu}, \frac{\partial p_{\mu}}{\partial p_{\nu}}\}$  sont données par l’équation C.10.

Symbole	Forme explicite	Remarques	Exemple <i>DKP</i>
$\tilde{\phi}(t, \tilde{\mathbf{v}})$ aussi noté $\tilde{\mathbf{v}}(t)$ ou $\tilde{\phi}_t(\tilde{\mathbf{v}})$	$\begin{pmatrix} \tilde{\phi}_t^1(\tilde{\mathbf{v}}) \\ \vdots \\ \tilde{\phi}_t^D(\tilde{\mathbf{v}}) \\ \tilde{\phi}_t^{N_c}(\tilde{\mathbf{v}}) \end{pmatrix} \equiv \begin{pmatrix} \phi_t^1(\tilde{\mathbf{v}}) \\ \vdots \\ \phi_t^D(\tilde{\mathbf{v}}) \\ \check{\phi}_t^1(\tilde{\mathbf{v}}) \\ \vdots \\ \check{\phi}_t^{N_c}(\tilde{\mathbf{v}}) \end{pmatrix}$	État du système continu au temps t à partir de la condition $\tilde{\mathbf{v}}$ . Dimension N ( $N = D + N_c$ )	$\begin{pmatrix} \nu(t) \\ p_\nu(t) \\ \mu(t) \\ p_\mu(t) \end{pmatrix}$
$\tilde{\mathbf{v}}_n$	$\begin{pmatrix} \tilde{v}_n^1 \\ \vdots \\ \tilde{v}_n^D \\ \check{v}_n^1 \\ \vdots \\ \check{v}_n^{N_c} \end{pmatrix} \equiv \begin{pmatrix} v_n^1 \\ \vdots \\ v_n^D \\ \check{v}_n^1 \\ \vdots \\ \check{v}_n^{N_c} \end{pmatrix}$	État du système discret sur la Section de Poincaré Dimension N ( $N = D + N_c$ )	$\begin{pmatrix} \nu_n \\ p_{\nu_n} \\ \mu_n \\ p_{\mu_n} \end{pmatrix}$
$\mathbf{v}_n$	$\begin{pmatrix} v_n^1 \\ \vdots \\ v_n^D \end{pmatrix}$	État du sous-système discret sur la Section de Poincaré (Dimension D)	$\begin{pmatrix} \nu_n \\ p_{\nu_n} \end{pmatrix}$
$\check{\mathbf{v}}_n$	$\begin{pmatrix} \check{v}_n^1 \\ \vdots \\ \check{v}_n^{N_c} \end{pmatrix}$	État des variables éliminées sur la Section de Poincaré ( $N_c$ variables)	$\begin{pmatrix} \mu_n \\ p_{\mu_n} \end{pmatrix}$
$\tilde{\mathbf{f}}(\tilde{\phi}) \Big _{\mathbf{v}_n, \tau_n}$	$\begin{pmatrix} \tilde{f}^1 \\ \vdots \\ \tilde{f}^N \end{pmatrix}_{\tilde{\mathbf{v}}_{n+1}} \equiv \begin{pmatrix} f^1 \\ \vdots \\ f^D \\ \check{f}^1 \\ \vdots \\ \check{f}^{N_c} \end{pmatrix}_{\tilde{\mathbf{v}}_{n+1}}$	Dérivées du système par rapport au temps (Équation 2.31) évaluées à $\tilde{\phi}(\tilde{\mathbf{v}}_n, \tau_n) = \tilde{\mathbf{v}}_{n+1}$ (Dimension N)	$\begin{pmatrix} p_\nu \\ \nu(2\epsilon - \frac{1}{4}\mu^4 - \frac{1}{2}\nu^2\mu^2) \\ p_\mu \\ \mu(2\epsilon - \frac{1}{4}\nu^4 - \frac{1}{2}\nu^2\mu^2) \end{pmatrix}_{\tilde{\mathbf{v}}_{n+1}}$ (équation C.1)
$\mathbf{f}(\tilde{\phi}) \Big _{\mathbf{v}_n, \tau_n}$	$\begin{pmatrix} f^1(\tilde{\phi}) \\ \vdots \\ f^D(\tilde{\phi}) \end{pmatrix}_{\tilde{\mathbf{v}}_{n+1}}$	Sous vecteur de $\tilde{\mathbf{f}}(\tilde{\phi})$ (Dimension D) On prend les D premières composantes de $\tilde{\mathbf{f}}(\tilde{\phi}) \Big _{\mathbf{v}_n, \tau_n}$ .	$\begin{pmatrix} p_\nu \\ \nu(2\epsilon - \frac{1}{4}\mu^4 - \frac{1}{2}\nu^2\mu^2) \end{pmatrix}_{\tilde{\mathbf{v}}_{n+1}}$ (équation C.1)

TAB. C.1 – Aide-mémoire pour les notations du chapitre 2.

Symbole	Forme explicite	Remarques	Exemple <i>DKP</i>
$\tilde{\mathbf{J}}_{\tilde{\phi}}(t, \tilde{\mathbf{v}})$ $\equiv$ $D_{\tilde{\mathbf{v}}} \tilde{\phi}(t, \tilde{\mathbf{v}})$	$\begin{pmatrix} \frac{\partial \tilde{\phi}_t^1}{\partial \tilde{v}^1} & \dots & \frac{\partial \tilde{\phi}_t^1}{\partial \tilde{v}^N} \\ \vdots & \ddots & \vdots \\ \frac{\partial \tilde{\phi}_t^N}{\partial \tilde{v}^1} & \dots & \frac{\partial \tilde{\phi}_t^N}{\partial \tilde{v}^N} \end{pmatrix}$	Matrice des variations (équations 2.37) ( $N \times N$ )	$\begin{pmatrix} \hat{\nu}^1 & \hat{\nu}^2 & \hat{\nu}^3 & \hat{\nu}^4 \\ \hat{p}_{\nu}^1 & \hat{p}_{\nu}^2 & \hat{p}_{\nu}^3 & \hat{p}_{\nu}^4 \\ \hat{\mu}^1 & \hat{\mu}^2 & \hat{\mu}^3 & \hat{\mu}^4 \\ \hat{p}_{\mu}^1 & \hat{p}_{\mu}^2 & \hat{p}_{\mu}^3 & \hat{p}_{\mu}^4 \end{pmatrix}$ (équation C.2)
$D_{\mathbf{v}} \phi _{\tilde{\mathbf{v}}_n, \tau_n}$	$\begin{pmatrix} \frac{\partial \phi^1}{\partial v^1} & \dots & \frac{\partial \phi^1}{\partial v^D} \\ \vdots & \ddots & \vdots \\ \frac{\partial \phi^D}{\partial v^1} & \dots & \frac{\partial \phi^D}{\partial v^D} \end{pmatrix}_{\tilde{\mathbf{v}}_n, \tau_n}$ $\equiv$ $\begin{pmatrix} \frac{\partial \tilde{\phi}^1}{\partial \tilde{v}^1} & \dots & \frac{\partial \tilde{\phi}^1}{\partial \tilde{v}^D} \\ \vdots & \ddots & \vdots \\ \frac{\partial \tilde{\phi}^D}{\partial \tilde{v}^1} & \dots & \frac{\partial \tilde{\phi}^D}{\partial \tilde{v}^D} \end{pmatrix}_{\tilde{\mathbf{v}}_n, \tau_n}$	Sous matrice $D \times D$ de $\tilde{\mathbf{J}}_{\tilde{\phi}}(t, \tilde{\mathbf{v}})$ On intègre 2.37 pendant un temps $\tau_n$ à partir de $\tilde{\mathbf{v}}_n$ et on prend les $D$ premières lignes et les $D$ premières colonnes de $\tilde{\mathbf{J}}_{\tilde{\phi}}(t = \tau_n, \tilde{\mathbf{v}} = \tilde{\mathbf{v}}_n)$ .	$\begin{pmatrix} \hat{\nu}^1 & \hat{\nu}^2 \\ \hat{p}_{\mu}^1 & \hat{p}_{\mu}^2 \end{pmatrix}_{\tilde{\mathbf{v}}_n, \tau_n}$ (équations C.2)
$D_{\tilde{\mathbf{v}}} \tau _{\tilde{\mathbf{v}}_n, \tau_n}$	$\left( \frac{\partial \tau}{\partial \tilde{v}^1} \dots \frac{\partial \tau}{\partial \tilde{v}^N} \right)_{\tilde{\mathbf{v}}_n, \tau_n}$	Variations du temps $\tau$ d'intégration par rapport à la position (Dimension $N$ ) (équation 2.40)	$\left( \frac{\partial \tau}{\partial \nu} \frac{\partial \tau}{\partial p_{\nu}} \frac{\partial \tau}{\partial \mu} \frac{\partial \tau}{\partial p_{\mu}} \right)_{\tilde{\mathbf{v}}_n, \tau_n}$ (équation C.8)
$D_{\mathbf{v}} \tau _{\tilde{\mathbf{v}}_n, \tau_n}$	$\left( \frac{\partial \tau}{\partial v^1} \dots \frac{\partial \tau}{\partial v^D} \right)_{\tilde{\mathbf{v}}_n, \tau_n}$ $\equiv$ $\left( \frac{\partial \tau}{\partial \tilde{v}^1} \dots \frac{\partial \tau}{\partial \tilde{v}^D} \right)_{\tilde{\mathbf{v}}_n, \tau_n}$	Sous vecteur de $D_{\tilde{\mathbf{v}}} \tau$ (Dimension $D$ ) On prend les $D$ premières composantes de $D_{\tilde{\mathbf{v}}} \tau _{\tilde{\mathbf{v}}_n, \tau_n}$ .	$\left( \frac{\partial \tau}{\partial \nu} \frac{\partial \tau}{\partial p_{\nu}} \right)_{\tilde{\mathbf{v}}_n, \tau_n}$ (équation C.8)

TAB. C.1 – Aide-mémoire pour les notations du chapitre 2 (suite).

Symbole	Forme explicite	Remarques	Exemple <i>DKP</i>
$\frac{\partial \phi}{\partial \check{v}^k}$	$\begin{pmatrix} \frac{\partial \phi^1}{\partial \check{v}^k} \\ \vdots \\ \frac{\partial \phi^D}{\partial \check{v}^k} \end{pmatrix}_{\tilde{\mathbf{v}}_n, \tau_n} \equiv \begin{pmatrix} \frac{\partial \tilde{\phi}^1}{\partial \check{v}^{D+k}} \\ \vdots \\ \frac{\partial \tilde{\phi}^D}{\partial \check{v}^{D+k}} \end{pmatrix}_{\tilde{\mathbf{v}}_n, \tau_n}$	<p>Sous vecteur de la matrice des variations  <math>\tilde{\mathbf{J}}_{\tilde{\phi}}(t, \tilde{\mathbf{v}})</math>      On prend la <math>(D+k)^{ième}</math> colonne de la matrice  <math>\tilde{\mathbf{J}}_{\tilde{\phi}}(t = \tau_n, \tilde{\mathbf{v}} = \tilde{\mathbf{v}}_n)</math></p>	<p>Pour <math>k = 2</math></p> $\begin{pmatrix} \hat{\nu}^4 \\ \hat{p}_\nu^4 \end{pmatrix}_{\tilde{\mathbf{v}}_n, \tau_n}$
$D_{\mathbf{v}} \check{v}^k _{\mathbf{v}_n}$	$\begin{pmatrix} \frac{\partial \check{v}^k}{\partial v^1} & \dots & \frac{\partial \check{v}^k}{\partial v^D} \end{pmatrix}_{\mathbf{v}_n} \equiv \begin{pmatrix} \frac{\partial \check{v}^{D+k}}{\partial \tilde{v}^1} & \dots & \frac{\partial \check{v}^{D+k}}{\partial \tilde{v}^D} \end{pmatrix}_{\mathbf{v}_n}$	<p>Variation des contraintes par rapport à <math>\mathbf{v}</math>      Les <math>\check{v}^k</math> sont exprimés <u>uniquement</u> en fonction des <math>\mathbf{v}</math></p>	$\begin{pmatrix} \frac{\partial p_\mu}{\partial \nu} & \frac{\partial p_\mu}{\partial p_\nu} \end{pmatrix}_{\mathbf{v}_n}$ (équation C.10)

TAB. C.1 – Aide-mémoire pour les notations du chapitre 2 (suite).

## Annexe D

# Publications associées à cette thèse

Nous présentons dans cette annexe la liste des publications réalisées pendant cette recherche. Les deux premières sont directement reliées à cette thèse. Les deux autres ont été produites en collaboration avec E. Brasselet (ancien étudiant gradué du département de Physique de l'Université Laval qui poursuit maintenant ses études post-doctorales à l'École Normale Supérieure de Cachan à Paris), ainsi que T.V. Galstian, (professeur au département de physique de l'Université Laval).

- B. DOYON AND L. J. DUBÉ *Targeting Unknown and Unstable Periodic Orbits* Phys. Rev. E **65**, 037202 (1-4) (2002)

We present a new method to target and subsequently control (if necessary) orbits of specified period but otherwise unknown stability and position. For complex systems where the dynamics is often mixed ( e.g. coexistence of regular and chaotic regions in area-preserving (Hamiltonian) systems), this targeting algorithm offers a novel way to, not only, gently bring the system from the chaotic domain to an unstable periodic orbit (where control is applied), but also to access *stable* regions of phase space (where control is not necessary) from *within* the stochastic regions. The technique is quite general and applies equally well to dissipative or conservative discrete maps and continuous flows.

- L. J. DUBÉ, P. DESPRÉS, B. DOYON, AND B. POURBOHLOUL *The Control of Hamiltonian Chaos* PiC – Special Issue on Nonlinear Dynamics – **57**, 77-82

(2001)

We demonstrate the control of the chaotic dynamics of Hamiltonian systems. This ability offers the possibility to select (stabilize) at will regular behavior(s) *within* the chaotic regime and to make efficient use of the richness and diversity of chaos.

- E. BRASSELET, B. DOYON, T.V. GALSTIAN, AND L.J. DUBÉ *New Laser Induced Spatio-Temporal Transition In Nematics* Phys. Lett. A **299**, 212-216 (2002)

We report the observation of a new light-induced spatio-temporal transition in homeotropically aligned nematic liquid crystals under the action of circularly polarized light. We believe that the competition of symmetric and asymmetric excitation modes is at the origin of this abrupt transition. A theoretical model explaining the main features of the observation is proposed.

- E. BRASSELET, B. DOYON, T.V. GALSTIAN, AND L.J. DUBÉ *Optically Induced Dynamics In Nematic Liquid Crystals : The Role of Twist Deformation and Asymmetry* (Submitted Phys. Rev. E)

We present a theoretical study of optically induced dynamics in a homeotropic nematic liquid crystal excited at normal incidence. By retaining the first symmetric and anti-symmetric reorientation modes, the dynamical equations are reduced to a four dimensional problem. The main advantage of this *minimal* approach is to emphasize the role of twisted mode and asymmetry of the light-induced molecular reorientation in a manner suitable for a clear physical interpretation. Theoretical results are compared with experiments in the particular case of circularly polarized light beams to show the physical origin of mode competition and of the breakdown of chiral and longitudinal symmetry. The model confirms previous experimental studies such as time dependent three dimensional molecular dynamics, light-induced stabilized helical reorientation in achiral nematics and in-plane precession regime. While a recent experiment in the case of a single circularly polarized excitation has revealed a new spatio-temporal transition, the model succeeds to describe all the features of such a bifurcation and points out the importance of asymmetry. Finally, the first quantitative description of the appearance of a giant mirrorless intrinsic optical bistability when twisted reorientation modes are excited is demonstrated. Particular care is exercised to obtain a qualitative physical interpretation for all these phenomena.

# Bibliographie

- [1] R. ARTUSO, E. AURELL, AND P. CVITANOVIC, *Recycling of strange sets : I. Cycle expansions*, Nonlinearity, **3** (1990), pp. 325–360.
- [2] ——, *Recycling of strange sets : II. Applications*, Nonlinearity, **3** (1990), pp. 325–360.
- [3] D. AUERBACH, P. CVITANOVIC, J. P. ECKMANN, G. GUNARATNE, AND I. PROCACCIA, *Exploring chaotic motion through periodic orbits*, Phys. Rev. Lett., **58** (1987), pp. 2387–2389.
- [4] E. BARRETO, E. J. KOSTELICH, C. GREBOGI, E. OTT, AND J. A. YORKE, *Efficient switching between controlled unstable periodic orbits in higher dimensional chaotic systems*, Phys. Rev. E, **51** (1995), pp. 4169–4171.
- [5] M. V. BERRY, *Regular and irregular motion*, AIP Conf. Proc., **46** (1978), pp. 16–120.
- [6] M. V. BERRY, *Regularity and chaos in classical mechanics, illustrated by three deformations of a circular ‘billiard’*, Eur. J. of Phys., **2** (1981), pp. 90–102.
- [7] O. BIHAM AND M. KVALE, *Unstable periodic orbits in the stadium billiard*, Phys. Rev. A, **46** (1992), pp. 6334–6339.
- [8] O. BIHAM AND W. WENZEL, *Unstable periodic orbits and the symbolic dynamics of the complex Hénon map*, Phys. Rev. A, **42** (1990), pp. 4629–4646.
- [9] G. D. BIRKHOFF, *Proof of the ergodic theorem*, Proc. Nat. Acad. Sci. U.S.A., **17** (1931), pp. 656–660.

- [10] J. BLASCHKE AND M. BRACK, *Periodic orbit theory of a circular billiard in homogeneous magnetic fields*, Phys. Rev. A, **56** (1997), pp. 182–194.
- [11] R. BLÜMEL AND W. P. REINHARDT, *Chaos in Atomic Physics*, Cambridge University Press, (1997).
- [12] S. BOCCALETTI, C. GREBOGI, Y.-C. LAI, H. MANCINI, AND D. MAZA, *The control theory of chaos : theory and applications*, Phys. Rep., **329** (2000), pp. 103–197.
- [13] M. CASDAGLI, *Nonlinear prediction of chaotic time series*, Physica D, **35** (1989), pp. 335–356.
- [14] F. CHRISTIANSEN AND P. CVITANOVIĆ, *Periodic orbit quantization of the anisotropic Kepler problem*, CHAOS, **2** (1992), pp. 61–69.
- [15] P. CVITANOVIĆ, *Invariant Measurement of Strange Sets in Terms of Cycles*, Phys. Rev. Lett., **61** (1988), pp. 2729–2732.
- [16] ——, *Periodic orbit theory in classical and quantum mechanics*, Chaos, **2** (1992), pp. 1–4.
- [17] ——, *Dynamical averaging in terms of periodic orbits*, Physica D, **83** (1995), pp. 109–123.
- [18] P. CVITANOVIĆ, R. ARTUSO, P. DAHLQVIST, R. MAINIERI, G. TANNER, G. VATTAY, N. WHELAN, AND A. WIRZBA, *Classical and Quantum Chaos*, www.nbi.dk/ChaosBook, (2000).
- [19] R. L. DAVIDCHACK AND Y. C. LAI, *Efficient algorithm for detecting unstable periodic orbits in chaotic systems*, Phys. Rev. E., **60** (1999), pp. 6172–6175.
- [20] R. L. DAVIDCHACK, Y. C. LAI, A. KLEBANOFF, AND E. BOLLT, *Towards complete detection of unstable periodic orbits in chaotic systems*, Phys. Lett. A, **287** (2001), pp. 99–104.
- [21] J. B. DELOS, S. K. KNUDSON, AND D. W. NOID, *Trajectories of an atomic electron in a magnetic field*, Phys. Rev. A, **30** (1984), pp. 1208–18.

- [22] V. DER POL, *Forced oscillations in a circuit with non-linear resistance*, Phil. Mag. Ser, **2** (1926), pp. 978–992.
- [23] P. DESPRÉS, *Détection et contrôle du chaos*, Master's thesis, Université Laval, mai 2000.
- [24] F. K. DIAKONOS, P. SCHMELCHER, AND O. BIHAM, *Systematic computation of the least unstable periodic orbits in chaotic attractors*, Phys. Rev. Lett., **81** (1998), pp. 4349–4352.
- [25] B. DOYON AND L. J. DUBÉ, *Targeting unknown and unstable periodic orbits*, Phys. Rev. E, **65** (2002), pp. 37202–1–4.
- [26] U. DRESSLER AND G. NITSCHE, *Controlling chaos using time delay coordinates*, Phys. Rev. Lett., **68** (1992), pp. 1–4.
- [27] L. DUBÉ, P. DESPRÉS, B. DOYON, AND B. POURBOHLOUL, *The Control of Hamiltonian Chaos*, Physics in Canada, **57** (2001), pp. 77–82.
- [28] L. J. DUBÉ AND P. DESPRÉS, *The Control of Dynamical Systems—Recovering Order from Chaos*, in The Physics of Electronic and Atomic Collisions, XXI International Conference, vol. **500**, AIP CONF. PROC., 2000, pp. 551–570.
- [29] J. P. ECKMANN AND D. RUELLE, *Ergodic theory of chaos and strange attractors*, Rev. Mod. Phys., **57** (1985), pp. 617–656.
- [30] J. D. FARMER, E. OTT, AND J. A. YORKE, *The dimension of chaotic attractors*, Physica D, **7** (1983), pp. 153–180.
- [31] J. D. FARMER AND J. J. SIDOROWICH, *Predicting chaotic time series*, Phys. Rev. Lett., **59** (1987), pp. 845–848.
- [32] M. FEIGENBAUM, *Quantitative universality for a class of nonlinear transformations*, J. Stat. Phys., **21** (1978), pp. 25–52.
- [33] H. FRIEDRICH AND D. WINTGEN, *The hydrogen atom in a uniform magnetic field - an example of chaos*, Physics Reports, **183** (1989), pp. 37–79.
- [34] J. GLEICK, *CHAOS : Making a New Science*, Viking, (1987).

- [35] P. GRASSBERGER, H. KANTZ, AND U. MOENIG, *On the symbolic dynamics of the Hénon map*, J. Phys. A., **22** (1989), pp. 5217–5230.
- [36] C. GREBOGI, E. KOSTELICH, E. OTT, AND J. A. YORKE, *Multi-dimensional intertwined basin boundaries : Basin structure of the kicked double rotator*, Physica D, **25** (1987), pp. 347–360.
- [37] C. GREBOGI, E. OTT, AND J. A. YORKE, *Unstable periodic orbits and the dimensions of multifractal chaotic attractors*, Phys. Rev. A, **37** (1988), pp. 1711–1724.
- [38] M. C. GUTZWILLER, *Chaos in Classical and Quantum Mechanics*, Springer - Verlag, (1990).
- [39] H. HADAMARD, *Les surfaces courbes opposées et leur lignes géodésiques*, J. Math. Pures et Appl., **1** (1898), pp. 27–73.
- [40] S. M. HAMMEL, C. K. R. T. JONES, AND J. V. MOLONEY, *Global dynamical behavior of the optical fields in a ring cavity*, J. Opt. Soc. Am. B, **2** (1985), pp. 552–564.
- [41] E. HELLER, M. CROMMIE, C. LUTZ, AND D. EIGLER, *Mesoscopic Systems and Quantum Corrals*, in The Physics of Electronic and Atomic Collisions, XIX International Conference, vol. **360**, AIP CONF. PROC., 1995, pp. 3–17.
- [42] M. HÉNON, *A two-dimensional mapping with a strange attractor*, Comm. Math. Phys., **50** (1976), pp. 69–77.
- [43] M. HÉNON, *On numerical computation of Poincaré maps*, Physica D, **5** (1982), pp. 412–414.
- [44] K. HORNBERGER AND U. SMILANSKY, *Spectral cross correlations of magnetic edge states*, Phys. Rev. Lett., **88** (2002), p. 24101.
- [45] K. HUANG, *Statistical Mechanics*, Wiley, 1963.
- [46] E. R. HUNT, *Stabilizing high-period orbits in a chaotic system : The diode resonator*, Phys. Rev. Lett., **67** (1991), pp. 1953–1955.

- [47] K. IKEDA, *Multiple-valued stationary state and its instability of the transmitted light by a ring cavity system*, Opt. Commun., **30** (1979), pp. 257–261.
- [48] E. A. JACKSON, *Perspectives of Nonlinear Dynamics*, Cambridge University Press, (1989).
- [49] A. KLEBANOFF AND E. BOLLT, *Convergence analysis of Davidchack and Lai's algorithm for finding periodic orbits*, Chaos, Solitons and Fractals, **12** (2001), pp. 1305–1322.
- [50] H. J. KORSCH AND H.-J. JODL, *Chaos, A program Collection for the PC*, Springer-Verlag, (1994).
- [51] E. KOSTELICH, C. GREBOGI, E. OTT, AND J. A. YORKE, *Higher-dimensional targeting*, Phys. Rev. E, **47** (1993), pp. 305–310.
- [52] D. LATHROP AND E. J. KOSTELICH, *Characterization of an experimental strange attractor by periodic orbits*, Phys. Rev. A, **40** (1989), pp. 4028–4031.
- [53] C. LAWSON AND R. HANSON, *Solving Least Squares Problems*, Prentice-Hall, (1974).
- [54] H. E. LEHTIHET AND B. N. MILLER, *Numerical study of a billiard in a gravitational field*, Physica D, **21** (1986), pp. 93–104.
- [55] E. N. LORENZ, *Deterministic nonperiodic flow*, J. Atmos. Sci., **20** (1963), pp. 130–141.
- [56] R. S. MACKAY AND J. D. MEISS, *Hamiltonian Dynamical Systems*, Adam Hilger, Bristol, (1987).
- [57] R. S. MACKAY, J. D. MEISS, AND I. C. PERCIVAL, *Transport in Hamiltonian systems*, Physica D, **13** (1984), pp. 55–81.
- [58] R. MARCINEK AND D. DELANDE, *Semiclassical photoionization cross section of the hydrogen atom in a magnetic field*, Phys. Rev. A, **62** (2000), pp. 1–12.
- [59] R. MAY, *Simple Mathematical Models with Very Complicated Dynamics*, Nature, **261** (1976), pp. 459–467.

- [60] H. MEYER, *Theory of the Liapunov exponents of Hamiltonian systems and a numerical study of the transition from regular to irregular classical motion*, J. Chem. Phys., **84** (1986), pp. 3147–3161.
- [61] V. MILNER, J. HANSSEN, W. CAMPBELL, AND M. RAIZEN, *Optical Billiards for Atoms*, Phys. Rev. Lett., **86** (2001), pp. 1514–1517.
- [62] C. MIRA, *Chaotic Dynamics. From the One-Dimensional Endomorphism to the Two-Dimensional Diffeomorphism*, World Scientific, (1987).
- [63] K. MYNNENI, A. T. BARR, N. J. CORRON, AND S. D. PETHEL, *New method for the control of fast chaotic oscillations*, Phys. Rev. Lett., **83** (1999), pp. 2175–2178.
- [64] N. NAKAMURA AND H. THOMAS, *Quantum billiard in a magnetic field : Chaos and diamagnetism*, Phys. Rev. Lett., **61** (1988), pp. 247–250.
- [65] G. NITSCHE AND U. DRESSLER, *Controlling chaotic dynamical systems using time delay coordinates*, Physica D, **58** (1992), pp. 153–164.
- [66] E. OTT, *Chaos in Dynamical Systems*, Cambridge University Press, (1993).
- [67] E. OTT, C. GREBOGI, AND J. A. YORKE, *Controlling chaos*, Phys. Rev. Lett., **64** (1990), pp. 1196–1199.
- [68] E. OTT, T. SAUER, AND J. A. YORKE, *Coping with Chaos*, Wiley, (1994).
- [69] T. S. PARKER AND L. O. CHUA, *Practical Numerical Algorithms for Chaotic Systems*, Springer-Verlag, 1989.
- [70] U. PARLITZ AND W. LAUTERBORN, *Superstructure in the bifurcation set of the Duffing equation*, Phys. Lett. A, **107** (1985), pp. 351–355.
- [71] B. PENG, V. PETROV, AND K. SHOWALTER, *Controlling low-dimensional chaos by proportionnal feedback*, Physica A, **188** (1992), pp. 210–216.
- [72] D. PINGEL, P. SCHMELCHER, AND F. DIAKONOS, *Detecting unstable periodic orbits in chaotic continuous-time dynamical systems*, Phys. Rev. E, **64** (2001), pp. 26214–1–6.

- [73] D. PINGEL, P. SCHMELCHER, F. DIAKONOS, AND O. BIHAM, *Theory and applications of the systematic detection of unstable periodic orbits in dynamical systems*, Phys. Rev. E, **62** (2000), pp. 2119–2134.
- [74] H. POINCARÉ, *Les Méthodes Nouvelles de la Mécanique Céleste*, Dover, 1957.
- [75] B. POURBOHLOUL, *Control and Tracking of Chaos in Hamiltonian Systems*, PhD thesis, Université Laval, 1999.
- [76] W. H. PRESS, S. A. TEUKOLSKY, W. T. VETTERLING, AND B. P. FLANNERY, *Numerical recipes in C*, Cambridge University Press, (1996).
- [77] K. PYRAGAS, *Continuous control of chaos by self-controlling feedback*, Physics Letters A, **170** (1992), pp. 421–428.
- [78] J. W. RAYLEIGH, *The Theory of Sound*, Dover, (1945).
- [79] C. REYL, L. FLEPP, R. ADDI, AND E. BRUN, *Control of NMR-laser chaos in high-dimensional embedding space*, Phys. Rev. E, **47** (1993), pp. 267–272.
- [80] P. RICHTER, H. SCHOLZ, AND A. WITTEK, *A breathing chaos*, Nonlinearity, **3** (1990), pp. 45–67.
- [81] M. ROBNICK, *Classical dynamics of a family of billiards with analytic boundaries*, J. Phys. A, **16** (1983), pp. 3971–3986.
- [82] M. ROBNICK AND M. V. BERRY, *Classical billiards in magnetic fields*, J. Phys. A, **18** (1984), pp. 1361–1378.
- [83] R. W. ROLLINS, P. PARMANANDA, AND P. SHERARD, *Controlling chaos in highly dissipative systems : A simple recursive algorithm*, Phys. Rev. E, **47** (1993), pp. 780–783.
- [84] F. J. ROMEIRAS, C. GREBOGI, E. OTT, AND W. P. DAYAWANSA, *Controlling chaotic dynamical systems*, Physica D, **58** (1992), pp. 165–192.
- [85] M. RUDERMAN, *Matter and magnetospheres in superstrong magnetic fields*, Ann. NY Acad. Sci., **257** (1975), pp. 127–140.
- [86] D. RUETTE, *Thermodynamic Formalism*, Addison - Wesley, (1978).

- [87] B. SALTZMAN, *Finite amplitude convection as an initial value problem I.*, J. Atmos. Sci., **19** (1962), pp. 329–341.
- [88] T. SAUER, J. YORKE, AND M. CASDAGLI, *Embedology*, J. Stat. Phys., **65** (1991), pp. 579–616.
- [89] P. SCHMELCHER AND F. K. DIAKONOS, *Detecting unstable periodic orbits of chaotic dynamical systems*, Phys. Rev. Lett., **78** (1997), pp. 4733–36.
- [90] P. SCHMELCHER AND F. K. DIAKONOS, *General approach to the localization of unstable periodic orbits in chaotic dynamical systems*, Phys. Rev. E, **57** (1998), pp. 2739–2746.
- [91] T. SCHREIBER, *Efficient neighbor searching in nonlinear time series analysis*, Int. J. Bifurcation and Chaos, **5** (1995), pp. 349–358.
- [92] T. SHINBROT, W. DITTO, C. GREBOGI, E. OTT, M. SPANO, AND J. A. YORKE, *Using the sensitive dependence of chaos (the 'butterfly effect') to direct trajectories in an experimental chaotic system*, Phys. Rev. Lett., **68** (1992), pp. 2863–2866.
- [93] T. SHINBROT, C. GREBOGI, E. OTT, AND J. A. YORKE, *Using chaos to target stationary states of flows*, Phys. Lett. A, **165** (1992), pp. 349–354.
- [94] T. SHINBROT, E. OTT, C. GREBOGI, AND J. A. YORKE, *Using chaos to direct orbits to targets in systems described by a one-dimensional map*, Phys. Rev. A, **45** (1992), pp. 4165–4168.
- [95] ——, *Using chaos to direct trajectories to targets*, Phys. Rev. Lett., **65** (1993), pp. 3215–3218.
- [96] Y. G. SINAI, *Topics in Ergodic Theory*, Princeton University Press, (1994).
- [97] D. SPEHNER, R. NAREVICH, AND E. AKKERMANS, *Semiclassical spectrum of integrable systems in a magnetic field*, J. Phys. A, **31** (1998), pp. 6531–6545.
- [98] T. SZEREDI AND D. GOODINGS, *Classical and quantum chaos of the wedge billiard. I. Classical mechanics*, Phys. Rev. E, **48** (1993), pp. 3518–3528.

- [99] M. TABOR, *Chaos and Integrability in Nonlinear Dynamics*, John Wiley & Sons, (1989).
- [100] F. TAKENS, *Detecting strange attractors in turbulence*, in Lecture Notes in Math., vol. **898**, Springer-Verlag, 1981.
- [101] G. TANNER, K. HANSEN, AND J. MAIN, *The semiclassical resonance spectrum of hydrogen in a constant magnetic field*, *Nonlinearity*, **9** (1996), pp. 1641–1670.
- [102] G. TANNER, P. SCHERER, E. BOGOMONLY, B. ECKHARDT, AND D. WINTGEN, *Quantum Eigenvalues from Classical Periodic Orbits*, *Phys. Rev. Lett.*, **67** (1991), pp. 2410–2412.
- [103] G. TANNER AND D. WINTGEN, *Quantization of chaotic systems*, *CHAOS*, **2** (1992), pp. 53–59.
- [104] D. ULLMO, K. RICHTER, AND R. A. JALABERT, *Orbital magnetism in ensembles of ballistic billiards*, *Phys. Rev. Lett.*, **74** (1995), pp. 383–386.
- [105] J. VON NEUMANN, *Proof of the quasi-ergodic hypothesis*, *Proc. Nat. Acad. Sci. U.S.A.*, **18** (1932), pp. 70–82.
- [106] W. WENZEL, O. BIHAM, AND C. JAYAPRAKASH, *Periodic orbits in the dissipative standard map*, *Phys. Rev. A*, **43** (1991), pp. 6550–6557.
- [107] D. WINTGEN, K. RICHTER, AND G. TANNER, *The semiclassical helium atom*, *CHAOS*, **2** (1992), pp. 19–32.