

UNIVERSITÉ DE SHERBROOKE
Faculté de génie
Département de génie mécanique

Asservissement d'un système robotique
sous-actué à un capteur neuromorphique

Mémoire de maîtrise
Spécialité : génie mécanique

Mathieu Bergeron

Sherbrooke (Québec) Canada

Août 2022

MEMBRES DU JURY

Julien Sylvestre

Directeur

Fabien Alibert

Évaluateur

Abderraouf Boucherif

Évaluateur

RÉSUMÉ

Le contrôle de systèmes physiques par boucle de rétroaction est intégré dans la technologie d'aujourd'hui. Ceux-ci dépendent de capteurs pour quantifier leurs interactions avec l'environnement et d'ordinateurs pour intégrer et calculer la réponse à envoyer aux actuateurs. Ce projet de recherche a pour but de prouver qu'il est possible de contrôler un système sous-actué lorsque ces composants ont été remplacés par un seul capteur neuromorphique. Pour ce faire, deux problèmes classiques de contrôle sont attaqués ; le chariot-tige et le pendule de Furuta. Une approche incrémentale est empruntée pour développer une méthode de contrôle à l'aide de "reservoir computer", une méthode d'apprentissage machine à laquelle le capteur neuromorphique appartient. Le problème du chariot-tige est tout d'abord attaqué en simulation en utilisant une variante numérique d'un réservoir computer, les Echo State Network (ESN). Dans ce cadre, une méthode de recherche de réservoir est choisie et différentes configurations d'ESN ont été testées. La structure qui s'est montrée la plus efficace en simulation est un ESN simple avec une rétroaction de la dernière commande générée, arrivant à basculer le pendule et à le maintenir droit sur son axe (à l'intérieur d'une marge de $\pm 15^\circ$) pendant en moyenne 81.2 % du temps total d'un essai. En comparaison, le contrôleur de référence maintient le pendule droit sur son axe en moyenne 85.2 % du temps d'un essai. Suite à ce succès, le contrôleur ESN est implémenté sur le montage expérimental du pendule de Furuta. Pour des essais d'une durée de 25 secondes, le contrôleur arrive à maintenir le pendule entre 15° et -15° pour une durée de 31.5 % du temps total, par rapport à 62.7 % pour le contrôleur de référence utilisé sur des essais d'une même durée. Ces performances ont été atteintes avec un contrôleur ESN utilisant une rétroaction et ayant plusieurs régressions, qui correspondent à différents régimes du pendule. Finalement, ce qui a été appris est ensuite transposé au processeur et au capteur neuromorphique, pour faire des tests de contrôle avec des réservoirs physiques. En conservant la même structure de régression, le capteur neuromorphique arrive à maintenir le pendule sur son axe pendant 18.4% du temps d'un essai.

Mots-clés : Physical reservoir computing, Reservoir computing, Capteur Neuromorphique, MEMS, Accéléromètre, Oscillateur, Pendule de Furuta, Contrôle, Systèmes sous-actuels

Ce qui rend le processus de la maîtrise et de l'écriture d'un mémoire agréable est la possibilité de diverger et d'explorer de nouveaux concepts, chose qui se fait beaucoup mieux à plusieurs. C'est pour cette raison que je souhaite remercier Guillaume Dion, Laurent Chiasson-Poirier, Anthony Haman et Xavier Trahan pour les discussions enrichissantes et leurs nombreuses contributions aux différents codes utilisés dans ce projet.

TABLE DES MATIÈRES

1	Introduction	1
1.1	Spécificités du problème	2
1.2	Contributions	3
2	État de l'art	5
2.1	Pendule de Furuta	5
2.1.1	Dynamique du système	6
2.1.2	Méthodes de contrôle	7
2.1.3	Évaluation des performances	10
2.2	Reservoir Computing	11
2.2.1	Réservoirs	12
2.2.2	Étage de sortie	15
2.2.3	Caractérisation des réservoirs	15
2.3	Capteur neuromorphique (CN)	18
2.3.1	Accéléromètres capacitifs	19
2.3.2	Oscillateur de Duffing	21
2.3.3	Physical reservoir computing (PRC)	22
3	Simulation et montages expérimentaux	25
3.1	Simulation	25
3.2	Montage du pendule de Furuta	30
3.3	Banc de test du CN	33
4	Echo State Networks	37
4.1	Simulation	37
4.1.1	Entraînement hors-ligne	38
4.1.2	Entraînement par lots	42
4.2	Montage expérimental	47
4.2.1	Entraînement par lots	50
4.2.2	Entraînement en ligne et structure de régressions	51
5	Physical reservoir computing (PRC)	59
5.1	Intégration	59
5.2	Réservoir physique	60
5.3	Processeur Neuromorphique	62
5.4	Capteur Neuromorphique	65
6	Discussion	71
6.1	Retour sur l'approche générale	71
6.1.1	Recherche de réservoir	71
6.1.2	Méthode d'entraînement	72

6.2	Contrôle de systèmes sous-actués	72
6.2.1	Contrôle par ESN	72
6.2.2	Contrôle par PRC	74
6.3	Stabilité des contrôleurs	75
7	Conclusion	79
	LISTE DES RÉFÉRENCES	81

LISTE DES FIGURES

1.1	Schéma du pendule de Furuta avec le placement du capteur neuromorphique. Un moteur actue la rotation du bras sur lequel le pendule est lié. Celui-ci est libre de tourné sur son axe selon les forces qu'il subit. Tel qu'illustré, le capteur neuromorphique est positioné sur la masse au bout du pendule et y subit seulement l'accélération de l'axe dans lequel il est orienté.	2
2.1	Schéma d'une pendule de Furuta au repos. Les axes de rotation ainsi que les principales quantités y sont indiqués.	5
2.2	Schéma général d'un contrôleur à boucle de rétroaction	8
2.3	Schéma d'un RC. L'étage d'entrée est illustré en bleu, le réservoir est en rouge et l'étage de régression est en vert.	12
2.4	Schéma d'un RC composé d'un seul noeud couplé dans le temps. L'étage d'entrée est illustré en bleu, le réservoir est en rouge et l'étage de régression est en vert. Le temps associé à la mesure de chaque noeud est θ et le temps d'une itération du réservoir est τ [1].	14
2.5	Image au microscope optique d'un accéléromètre neuromorphique. 1) Masse inertielle suspendue. 2) Ressorts. 3) Arrêt ; préviens les trop grand déplacements. 4) Oscillateur de Duffing. 5) Jauges piezorésistives.	19
2.6	Image au microscope électronique d'un oscillateur de Duffing (rouge) piloté par une électrode (jaune). Les jauges piézorésistives sont surlignées en vert et l'électrode de mesure est en bleu [2].	21
2.7	Mesure de l'amplitude du signal d'oscillation (en haut) et comparaison entre un modèle numérique (en bas) en fonction de l'amplitude du voltage d'alimentation V_0 [3]	23
3.1	Influence du pas de temps sur la trajectoire calculée pour le pendule. Lorsque de grandes vitesses sont atteintes, un plus petit pas de temps est requis pour conserver une précision minimale. Les valeurs des constantes physiques de la simulation sont présentées au tableau 3.2.	27
3.2	Erreur relative à l'énergie initiale du système dans le temps pour différents pas de temps. La trajectoire de référence utilisée est une simulation avec un pas de temps de $78.125 \mu s$. Les trajectoires associées aux courbes d'erreur sont illustrées à la figure 3.1. Les paramètres de la simulation sont présentés au tableau 3.2	27
3.3	Exemple de différentes trajectoires générées par simulation. Le point de départ de celles-ci est choisi près du sommet du pendule pour bien illustrer les deux attracteurs localisés à -180° et 180° , qui représentent le pendule à son état de repos.	28

3.4	Exemple de trajectoires actuées où la commande de balancement vers le sommet est calculée à partir du contrôleur proposé par Gordillo et al. [4] et la commande d'équilibre est calculée avec un contrôleur LQR. Les X rouges représentent les conditions initiales de chaque trajectoire.	28
3.5	À gauche - Vue latérale du pendule de Furuta. À droite - Identification des différents composants localisés sur le pendule de Furuta ainsi qu'en rouge les circuits d'acquisition des données.	31
3.6	Ensemble de 10 trajectoires du pendule de Furuta sélectionnées aléatoirement parmi les tests effectués avec le contrôleur de référence. Le pendule est initialement au repos, avant que le moteur du bras force une impulsion de 0.154 Nm pendant 0.5 seconde. Les croix rouges représentent les conditions initiales du pendule alors que les croix bleues représentent la dernière position du pendule avant que le contrôleur soit interrompu, 25 secondes plus tard.	32
3.7	Variation de la proportion de temps passé au-dessus de 15° pour un contrôleur idéal. Chaque essai est d'une durée de 25 secondes. Entre chacun d'eux, le pendule est forcé à sa position initiale, soit $\theta_1 = 0, \dot{\theta}_1 = 0$	33
3.8	Schéma bloc du banc de test du capteur neuromorphique.	34
3.9	Fonctionnement général de chacun des blocs qui composent la boucle de rétroaction électronique du système.	35
3.10	Schéma de bloc des deux actionneurs qui peuvent être utilisés pour reproduire des accélérations sur le banc de test.	35
4.1	Axes des accélérations utilisées par l'ESN.	37
4.2	À gauche - Progression des performances après chaque nouvel entraînement sur un ensemble de données de test de 50 essais. À droite - même graphique où le NRMSE minimal a été soustrait sur une échelle semi-log. Ce graphique permet de représenter visuellement la progression par rapport au NRMSE minimal atteint.	39
4.3	À gauche - Recherche de réservoir optimal à l'aide de la méthode de Monte-Carlo. Au centre - Recherche de réservoir optimal par grille. À droite - Recherche de réservoir optimal à l'aide d'optimisation bayésienne.	40
4.4	En haut à gauche - Comparaison de la série de commandes idéales aux commandes générées par ESN préentraîné et aux commandes de référence instantanées. La série de commandes idéales correspond aux commandes générées pour la même situation initiale lorsque le contrôleur de référence effectue entièrement le contrôle. La commande de référence instantanée est la commande générée par le contrôleur de référence à chaque itération lorsque l'ESN contrôle la simulation. En haut à droite - Comparaison entre les trajectoires empruntées par les contrôleurs de référence et ESN préentraînés pour la même condition initiale. En bas - Comparaison des performances des deux contrôleurs pour 100 essais avec les mêmes conditions initiales. . .	41

4.5	À gauche - Comparaison du NRMSE entre la commande de référence instantanée et la commande générée par l'ESN dans chacun des deux méthodes d'entraînement. À droite - Comparaison des performances des 3 contrôleurs pour 100 essais. Dans les deux méthodes d'évaluations, une amélioration des performances de l'ESN est observée lorsque l'entraînement par lot est utilisé.	43
4.6	En haut - Illustration de l'angle du pendule dans le temps. Les deux lignes rouges représentent les angles de -15 et 15 degrés. En bas - Signal de 10 noeuds choisis aléatoirement par rapport au temps. L'amplitude des variations des noeuds est beaucoup plus petite lorsque le pendule est près du sommet de sa trajectoire.	43
4.7	Schéma d'un RC à sortie composée. Un nombre arbitraire de régressions peut-être sélectionné pour discrétiser l'espace de la fonction à reproduire. Le choix de régression peut être effectué à l'aide d'un RC préentraîné tel qu'illustré, ou bien à partir de mesures prises sur le système dynamique tel que l'angle du pendule. $X(t)$, $u(t)$ et $a(t)$ représentent respectivement les états du réservoir, la commande générée par la régression et l'accélération du pendule du Furuta.	45
4.8	Comparaison des performances des 4 contrôleurs pour 100 essais. Une augmentation de la proportion du temps total passé près du point d'équilibre est observée pour le contrôleur ESN entraîné par lot avec une sortie composée. En moyenne, le contrôleur maintient le pendule près de son point d'équilibre (59.1 ± 5.6) % du temps total d'un essai.	45
4.9	Exemple de trajectoire empruntée par un contrôleur composé en simulation. En haut, la position angulaire du pendule en fonction du temps. En bas, la trajectoire empruntée par le pendule sur son diagramme de phase. Il semble y avoir un biais qui a été introduit dans l'entraînement, de sorte que son point de stabilisation est près de 5°	46
4.10	48
4.11	Comparaison des performances des 5 contrôleurs pour 100 essais. Une amélioration notable est obtenue par le contrôleur avec rétroaction par rapport aux autres structures d'ESN. Ses performances sont même similaires à celles du contrôleur de référence.	48
4.12	Angle du pendule dans le temps et représentation des noeuds du réseau. . .	49
4.13	Trajectoires empruntées par la simulation lorsqu'un contrôleur ESN avec rétroaction entraîné est utilisé.	49
4.14	Progression de la recherche selon le nombre de réservoir évalué avec un entraînement hors-ligne et une régression simple.	50
4.15	Historique de l'erreur entre la commande générée et la commande de référence des différents essais lors de l'entraînement. Chaque nouvel essai utilise une régression qui est calculée à partir des données récupérées aux essais précédents pour effectuer le contrôle.	51

4.16	Comparaison des trajectoires empruntées lors des 10 premiers essais du réseau. Le début de chaque trajectoire est marqué d'un "X" rouge. Chaque nouvel essai utilise une régression qui est calculée à partir des données récupérées aux essais précédant. La commande de contrôle est ensuite calculée à chaque instant à partir de la nouvelle régression. L'entièreté des trajectoires est présentée pour illustrer la perte de contrôle du pendule lors des premiers essais.	52
4.17	Trajectoires empruntées par le pendule de Furuta dans son espace de phase lors des 10 derniers essais de son entraînement.	52
4.18	Temps d'inversion d'une matrice selon ses dimensions. 100 matrices aléatoires de chaque dimension ont été générées puis inversées. Chaque point sur le graphique correspond à la moyenne de ces 100 temps d'inversion et les barres d'erreurs correspondent à un écart-type (1σ). La ligne horizontale rouge représente le seuil de 5 [ms], qui ne doit en aucun cas être dépassé.	54
4.19	Comparaison de l'évolution du NRMSE pendant l'entraînement selon la méthode utilisée. Les deux méthodes convergent plus ou moins vers la même valeur, avec une légère différence. L'apprentissage initial est plus rapide avec l'entraînement en ligne, mais le NRMSE vers lequel ce type d'entraînement converge semble légèrement plus haut.	54
4.20	Exemples de trajectoires au début et à la toute fin de l'entraînement. Les "X" rouges représentent le début des trajectoires. À gauche - 10 premières trajectoires obtenues lors de l'entraînement en ligne. À droite - 10 dernières trajectoires obtenues lors de l'entraînement en ligne.	55
4.21	Performances et exemples de trajectoires du contrôleur avec une structure à double régression.	56
4.22	Performances et exemples de trajectoires d'un contrôleur ESN avec rétroaction et une structure à trois régressions.	58
5.1	Schéma de bloc du code utilisé pour interfacer avec le capteur neuromorphique. Les blocs de gestion du code sont encadrés en vert. Les sous-programmes sont identifiés en bleu et les modules physiques d'interfaçage avec le capteur neuromorphique sont en rouge.	60
5.2	Schéma du code qui permet de lier les montages du LCSM au pendule de Furuta en temps réel. Les blocs de gestion de code sont en vert. Ceux-ci servent à démarrer les sous-programmes et prédéfinir les blocs mémoires à partager. Les sous-programmes qui accomplissent des tâches spécifiques sont identifiés en bleu puis les composantes physiques d'interfaçage par lecture/écriture de voltage sont identifiées en rouge. Les communications sont effectuées via UDP par deux des sous-programmes sur chaque montage expérimental. L'ordinateur du pendule est asservi à l'ordinateur du LCSM via un script powershell, qui est initialement démarré par une commande SSH.	61

5.3	NRMSE de différents réservoirs testés suite à leur entraînement hors-ligne. L'entraînement est effectué sur 5 essais préenregistrés où le contrôleur de référence bascule le pendule vers le point d'équilibre. La mesure de NRMSE est effectuée sur 3 autres essais préenregistrés où les signaux du réservoir sont combinés avec la régression linéaire calculée à l'entraînement puis comparés à la commande de référence de chaque cas.	64
5.4	Progression de l'entraînement par lots en utilisant un processeur neuromorphique intégré avec un pendule de Furuta. Une structure à trois régressions est utilisée pour générer la commande transmise à l'actuateur. La commande de référence est générée par le contrôleur par modulation d'énergie lors du balancement vers le haut, puis par un contrôleur LQR pour maintenir l'équilibre. Le NRMSE minimal obtenu lors d'un essai expérimental est de 0.6978, la ligne rouge correspond à ce minimum. Les coupures de la courbe du NRMSE sont dues à des ratés dans la communication entre les montages expérimentaux. Ces ratés ne sont pas utilisés lors de l'entraînement.	64
5.5	Comparaison des performances obtenues dans différentes situations. L'histogramme en orange représente les performances obtenues lors de l'entraînement par lots avec des hyperparamètres fixes, où entre chaque essai la régression était mise à jour. La distribution en vert représente les performances de la structure de régression utilisée à l'essai #48 de l'entraînement par lots. Elle a été sélectionnée puisque c'est la structure de régression avec laquelle les meilleures performances (proportion de 29.10 % du temps passé à 15° du point d'équilibre) avaient été obtenues.	66
5.6	Trajectoire emprunté par le pendule de Furuta dans son espace de phase lorsque le contrôle est effectué à l'aide d'un capteur neuromorphique. L'essai dont la trajectoire est illustré passe 26% de son temps sous un angle de 15°. Le contrôleur de référence est initialement utilisé. Le contrôle est ensuite repris au point de transition, indiquer par un "X" rouge, par le processeur neuromorphique et sa structure de contrôle.	66
5.7	Commandes associées à la trajectoire présentée à la figure 5.6. La commande "Référence" est celle générée par le contrôleur standard, la commande "Génération" est celle générée par le processeur neuromorphique et sa structure de régression. La commande "Initialisation" est la commande utilisée pour initialiser le pendule de Furuta dans un état aléatoire. Il y a des différences entre la commande "Référence" et la commande "Initialisation" puisque la première est générée localement et que la seconde est générée au LCSM. Il y a donc de légers délais entre les données utilisées par le contrôleur dans les deux cas. La seconde différence est la première demi-seconde, qui est une application forcée de moment par le moteur pour démarrer le balancement. Durant les essais, les commandes "Initialisation" puis "Génération" sont appliquées sur le pendule.	67
5.8	Progression de la recherche de réservoir. Le réservoir ayant démontré les meilleures performances en termes de NRMSE est sélectionné pour l'entraînement par lots.	67
5.10	Comparaison de la distribution de performances de différents contrôleurs.	68

5.9	Progression de l'entraînement par lots du capteur neuromorphique. La ligne en rouge représente le NRMSE minimal obtenu lors de la recherche de réservoir par entraînement hors-ligne.	69
6.1	Diagramme de phase du meilleur essai obtenu à l'aide d'un contrôleur ESN. Ce contrôleur utilise les accélérations du pendule dans 3 axes orthogonaux et un étage de sortie à 3 régressions.	73
6.2	Trajectoire angulaire du pendule pour le même essai que celui présenté à la figure 6.1. Les zones délimitées par les lignes rouges représentent les angles de 15° et -15° près du point d'équilibre. La proportion de temps passé sous 15° du pendule lors de cet essai était de 48.2 %.	74
6.3	Distribution de performance obtenue lors de l'entraînement en ligne par un ESN à une accélération, avec une rétroaction et une structure à 3 régressions.	75
6.4	Distribution des propriétés de différents réservoirs. 3000 ESNs aléatoires ont été testés et 100 réservoirs physiques ont été testés avec le capteur neuromorphique. Les deux types de réservoirs n'occupent pas les mêmes régions dans l'espace des hyperpropriétés. Des points de donnée additionnels sont nécessaires avec les réservoirs physiques pour pouvoir mieux comparer les limitations de chaque type de réservoir.	76
6.5	Comparaison des écarts-types relatifs des différentes distributions selon le type de réseau utilisé et la moyenne de leur performance.	77

LISTE DES TABLEAUX

2.1	Paramètres physiques associés au modèle du pendule de Furuta	6
2.2	Propriétés de l'oscillateur de Duffing microfabriqué [3]	23
2.3	Propriétés de l'accéléromètre [3]	24
3.1	Paramètres des contrôleurs utilisés	29
3.2	Constantes utilisés dans la simulation du chariot-tige	29
3.3	Constantes physiques du pendule de Furuta et leur valeurs	30
3.4	Paramètres des contrôleurs utilisés	31
4.1	Paramètres fixes pour chaque essai d'entraînement hors-ligne	38
4.2	Bornes des hyperparamètres de réservoirs optimisés par les différents type de recherche.	38
4.3	48
4.4	Bornes de la recherche par optimisation bayésienne et hyperparamètres du réservoir trouvé par optimisation bayésienne et utilisé pour le contrôle du montage expérimental. Il est à noter que différents réservoirs ont été testés suite à des recherches avec différentes bornes. Les valeurs choisies présentées au tableau sont celles avec lesquelles le plus de résultats ont été obtenus.	50
5.1	Noms des hyperparamètres et sommaire description de chacun d'eux.	61
5.2	Hyperparamètres du meilleur essai de la recherche par optimisation bayésienne. Le NRMSE obtenu entre la commande de référence et la commande générée pour cet essai est de 0.6781	63
5.3	Comparaison numérique de tous les résultats obtenus. Les spécificités du type d'entraînement, de la structure de régression utilisée et de la recherche sont affichées. Il est possible d'y voir rapidement quels changements ont permis d'augmenter significativement les performances dans chaque cas. L'incertitude présentée sur la moyenne est l'écart type de la distribution. L'abréviation «O.B.» représente dans ce cas-ci «Optimisation Bayésienne». Il est à noter que seuls les réservoirs les plus performants de chaque recherche sont considérés lors des tests.	70
6.1	Bornes de l'exploration des hyperpropriétés des différents types de réservoirs. Les propriétés de l'ESN ont été explorées en utilisant des distributions uniformes aléatoires entre les bornes présentées. Pour le réservoir physique, les propriétés ont été explorées en utilisant une exploration par grille, comme le but initial de ces essais était d'isoler différents régimes d'opération du substrat.	76

LEXIQUE

Mot ou expression	Abbréviation
Reservoir computer	RC
Physical reservoir computing	PRC
Capteur neuromorphique	CN
Recurrent Neural Networks	RNN
MicroSystèmes ÉlectroMécanique	MEMS
Full State Feedback	FSF
Linear Quadratic Regulator	LQR
Non-Linear AutoRegressive Moving Average	NARMA

LISTE DES SYMBOLES

Paramètres	Symbole	Unité
Constante gravitationnelle	g	$[m/s^2]$
Masse du pendule	m_1	$[kg]$
Masse cumulative (bras & pendule)	m_0	$[kg]$
Longueur du pendule	L_1	$[m]$
Longueur du bras rotatif	L_0	$[m]$
Inertie de l'axe de rotation vertical	I_0	$[kgm^2]$
Intertie du pendule	I_1	$[kgm^2]$
Moment de force appliqué à l'axe vertical	u	$[Nm]$
Constante d'amortissement du pendule	α_1	$[kgm^2/s]$
Constante d'amortissement de l'axe vertical de rotation	α_0	$[kgm^2/s]$
Dérivée par rapport au temps	$\dot{\quad}$	$[1/s]$
Voltage	V	$[V]$
Énergie potentielle	U	$[J]$
Énergie cinétique	K	$[J]$
Commande envoyée par un contrôleur	u	N/A
Taux de succès	τ_s	\square
Poids du réservoir	W	\square
Poids de l'étage de sortie du réservoir	W_{out}	\square
Matrice de régularisation	Γ	\square
Masse d'un système masse-ressort	M_{eff}	$[kg]$
Force	F	$[N]$
Constante de ressort	K_{eff}	$[N/m]$
Rigidité d'un système masse ressort (atténuation)	ζ_{eff}	$[Ns/m]$
Fréquence de résonance angulaire	ω_0	$[rad/s]$
Largeur de la poutre	w	$[\mu m]$
Épaisseur de la poutre	t	$[\mu m]$
Longueur de la poutre	l	$[\mu m]$
Module d'Young du silicone	E	$[Pa]$
Constante de non-linéarité	β	$[Hz^2/m^2]$
Fréquence de résonance	f_0	$[Hz]$
Facteur de qualité	Q	\square
Distance entre la poutre et la masses inertielle au repos	d	$[\mu m]$
Dimension de la masse inertielle	w_{acc}	$[\mu m^2]$
Dimension des trous dans la masse inertielle	t_{acc}	$[\mu m^2]$
Masse	m_{acc}	$[ng]$
Distance entre la masse inertielle et son arrêt	d_{gap}	$[\mu m]$
Largeur des branches des ressorts	$w_{ressort}$	$[\mu m]$
Épaisseur des branches des ressorts	$t_{ressort}$	$[\mu m]$
Longueur des petites branches des ressorts	L_s	$[\mu m]$
Longueur des grandes branches des ressorts	L_l	$[\mu m]$
Fréquence de résonance de la masse inertielle	f_{acc}	$[kHz]$

CHAPITRE 1

Introduction

Par définition, un capteur est un dispositif permettant de détecter une grandeur physique et de la restituer sous la forme d'un signal utilisable. C'est l'une des unités de base des technologies modernes puisqu'ils permettent de mesurer des caractéristiques de leur environnement et de transmettre ces données à un système externe. Pour cette raison, pratiquement toutes les technologies modernes en font usage. Cette demande a mené au développement de capteurs MEMS, des microsystèmes électromécaniques fabriqués à même les circuits intégrés en utilisant des technologies de microfabrication. Ils sont petits, efficaces et très peu énergivores. Cependant, pour les systèmes complexes comprenant plusieurs capteurs ou une dynamique complexe, ils doivent toujours être liés à un ordinateur pour interpréter et distribuer les informations. Dû à leur généralité, les ordinateurs ne sont pas optimisés pour ce type de problème, ils sont souvent trop puissants et consomment plus d'énergie que nécessaire. La prochaine étape à franchir pour une nouvelle génération de systèmes intelligents serait d'intégrer une plus grande capacité de calcul directement sur les capteurs, leur permettant de générer directement des signaux exploitables.

Les capteurs neuromorphiques (CN) sont justement un type de capteurs qui a été développé pour intégrer une capacité de traitement à la détection de leur environnement. Un capteur neuromorphique est une combinaison d'un processeur neuromorphique avec un capteur. Le processeur neuromorphique utilise des analogues matériels à des neurones pour créer un réseau qui apprendra à faire des opérations. Un CN est obtenu lorsque les neurones peuvent être influencés par un capteur sans passer par une interface numérique. La méthode la plus prometteuse pour mettre en place un capteur neuromorphique est le "physical reservoir computing" (PRC)[5]. Cette méthode utilise les interactions entre des systèmes dynamiques complexes pour accomplir une variété de tâches. L'un de ces dispositifs est un accéléromètre neuromorphique MEMS, développés au 3IT dans le laboratoire de Julien Sylvestre. L'accéléromètre neuromorphique a résolu des problèmes tests tels que la prédiction de fonction de parité ou bien de séries chaotiques telle que NARMA-10 [3]. L'étape qui suit naturellement est de démontrer la capacité du capteur neuromorphique à remplir les requis d'applications réelles. Deux problèmes de contrôle classiques sont le chariot-tige et le pendule de Furuta. Ces deux systèmes sont sous-actués, c'est-à-dire que leur nombre de degrés de liberté est plus grand que leur nombre d'actuateurs (qui sont

respectivement de 2 et de 1 dans les deux cas). Dans le but de démontrer l'usabilité du capteur neuromorphique, la question de recherche abordée dans ce mémoire est la suivante :

Est-il possible d'asservir un système robotique sous-actué à un capteur neuromorphique ?

L'objectif primaire de cette thèse est de démontrer qu'il est possible d'asservir un système sous-actué à un capteur neuromorphique. De façon plus spécifique, l'hypothèse sera testée avec un problème de référence, le pendule de Furuta. Ce problème sert de porte d'entrée des CN vers de nombreuses applications du domaine du contrôle. Le bagage nécessaire à la compréhension des domaines abordés dans ce travail est présenté dans la section de l'état de l'art (chapitre 2).

1.1 Spécificités du problème

Le problème qu'on souhaite résoudre est le contrôle du pendule de Furuta. Cependant, puisqu'on désire utiliser le capteur neuromorphique, certains détails doivent être explicités. Premièrement, le capteur neuromorphique utilisé est sensible à l'accélération dans un seul axe. Pour qu'un signal qui dépend de tous les mouvements du système soit pris en compte par le capteur neuromorphique, il faut donc qu'il soit placé au bout du pendule, tel qu'illustré à la figure 1.1

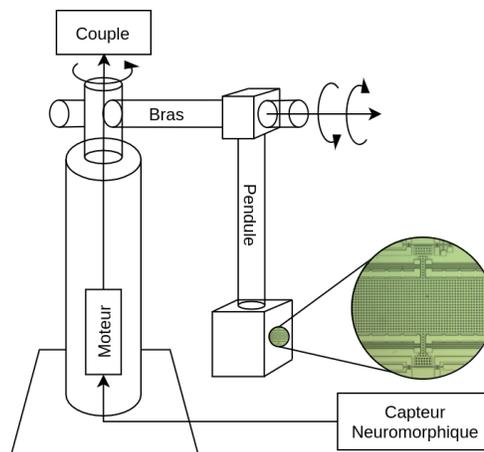


FIGURE 1.1 Schéma du pendule de Furuta avec le placement du capteur neuromorphique. Un moteur actue la rotation du bras sur lequel le pendule est lié. Celui-ci est libre de tourner sur son axe selon les forces qu'il subit. Tel qu'illustré, le capteur neuromorphique est positionné sur la masse au bout du pendule et y subit seulement l'accélération de l'axe dans lequel il est orienté.

Le capteur neuromorphique n'utilise pas une méthode standard de calcul, mais fait usage du "réservoir computing" pour arriver à reproduire un signal de référence. Comme de fait, les défis associés au réservoir computing sont rencontrés dans ce projet. Premièrement, la recherche d'un réservoir qui génère des signaux assez diversifiés pour reproduire un signal de sortie du même type que celui désiré. Par la suite, différentes topologies peuvent être explorées, tel que l'ajout de rétroaction de la sortie, qui ont un impact sur les bornes des paramètres de la recherche. Puis dernièrement, la méthode d'entraînement, qui influence le calcul de poids de sorti, et donc la performance finale.

1.2 Contributions

Ce mémoire présente les progrès faits jusqu'à maintenant pour prouver qu'il est possible d'effectuer le contrôle d'un système sous-actué à l'aide d'un capteur neuromorphique.

Des capteurs neuromorphiques n'ont encore jamais été utilisés dans des applications de contrôle. Il n'existe donc pas encore de méthode fiable pour enseigner une commande et obtenir des résultats. Le problème sera donc approché de façon incrémentale. Une grande quantité de simplifications seront initialement effectuées, puis elles seront progressivement éliminées au cours du projet. Tout d'abord, le problème du chariot-tige sera abordé dans une simulation, introduite à la section 3.1. Comme la simulation permet de tester des méthodes rapidement, une méthode de recherche de réservoir a été sélectionnée pour le restant du projet, soit l'optimisation bayésienne. Dans le même cadre de simulation, différentes méthodes d'apprentissage et topologie de réservoirs sont testées. En simulation, seuls des tests avec des "Echo State Networks" (ESN) sont effectués (voir la section 4.1). À partir des résultats obtenus à l'étape précédente, un passage est effectué au montage expérimental du pendule de Furuta (voir section 3.2). Le contrôle du pendule de Furuta est tout d'abord tenté avec des ESNs (voir section 4.1). Des méthodes d'entraînement et différentes topologies de réservoir y sont validées. Par la suite, un processeur neuromorphique (voir section 5.3) puis finalement un capteur neuromorphique (voir section 5.4) sont intégrés avec un lien temps réel. La topologie ayant donné les meilleurs résultats avec un ESN est récupérée puis les performances comme contrôleur d'un processeur et d'un capteur neuromorphique sont mesurées expérimentalement. Un retour sur l'approche générale empruntée est par la suite effectuée (voir chapitre 6).

CHAPITRE 2

État de l'art

2.1 Pendule de Furuta

Le problème de pendule inverse est un problème classique pour démontrer des applications de contrôle. Le but du problème est d'arriver à maintenir un pendule en équilibre au-dessus de son axe de rotation ($\theta_1 = 0^\circ$) à partir d'une position au repos ($\theta_1 = 180^\circ$). La seule façon d'influencer le système est via une force latérale appliquée à la base de l'axe de rotation du pendule. Un système de ce type est aussi qualifié de sous-actué étant donné que le nombre d'actuateurs est inférieur à son nombre de degrés de liberté, dans ce cas-ci l'axe de déplacement horizontal et l'axe de rotation du pendule. Le pendule de Furuta est une variante de ce type de problème. Dans celle-ci, la base du pendule est fixée avec un pivot sur une tige horizontale perpendiculaire à un second axe de rotation central. Le degré de liberté associé au déplacement horizontal est donc remplacé par un axe de rotation vertical. Un schéma du pendule de Furuta est présenté à la figure 2.1.

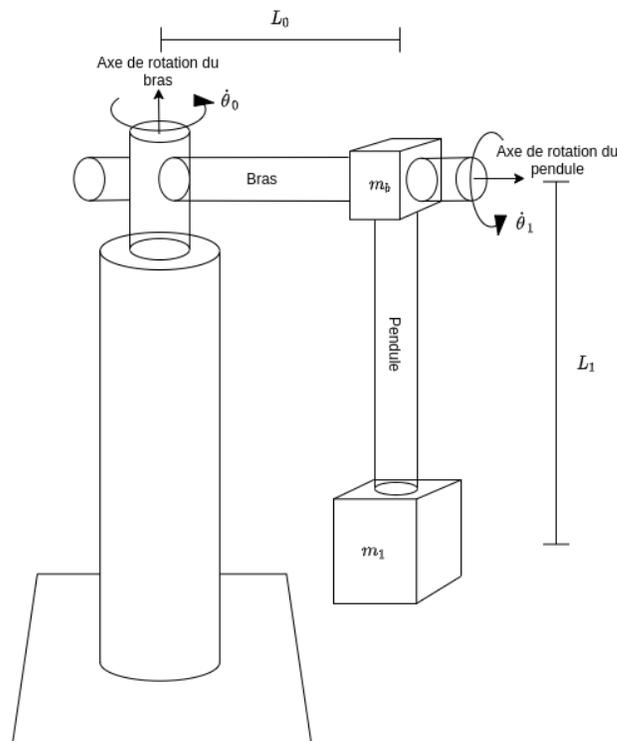


FIGURE 2.1 Schéma d'une pendule de Furuta au repos. Les axes de rotation ainsi que les principales quantités y sont indiqués.

La base du pendule est actuée en appliquant un couple sur l'axe de rotation du bras. Cette modification permet d'affranchir l'axe horizontal des limites physiques des déplacements latéraux. Cependant, des forces supplémentaires doivent être considérées étant donné que le pendule est maintenant dans un référentiel non inertiel. Initialement introduit en 1991 par K. Furuta [6], le problème est depuis devenu un étalon pour tester des méthodes de contrôle émergentes.

2.1.1 Dynamique du système

Dans le but de dériver les équations du système, une approche Lagrangienne est prise. Le tableau 2.1 liste les constantes pertinentes.

Paramètres	Symbole	Unité
Constante gravitationnelle	g	$[m/s^2]$
Masse du pendule	m_1	$[kg]$
Masse cumulative (bras & pendule)	m_0	$[kg]$
Longueur du pendule	L_1	$[m]$
Longueur du bras rotatif	L_0	$[m]$
Inertie de l'axe de rotation vertical	I_0	$[kgm^2]$
Inertie du pendule	I_1	$[kgm^2]$
Moment de force appliqué à l'axe vertical	u	$[Nm]$
Constante d'amortissement du pendule	α_1	$[kgm^2/s]$
Constante d'amortissement de l'axe vertical de rotation	α_0	$[kgm^2/s]$

TABLEAU 2.1 Paramètres physiques associés au modèle du pendule de Furuta

Le système peut être décrit au complet par deux coordonnées, θ_0 et θ_1 , où chaque variable représente respectivement l'angle entre le bras et son origine, et l'angle du pendule par rapport à sa position verticale. L'énergie potentielle du pendule lorsqu'il est à son sommet est fixée à 0. Pour simplifier le modèle, il est approximé que $m_1 \approx m_0$ et que cette l'entièreté de cette masse est fixée au bout du pendule. Les énergies potentielles et cinétiques du système sont respectivement décrites comme suit :

$$U = m_1 g L_1 (\cos(\theta_1) - 1) \quad (2.1)$$

$$T = \frac{1}{2} (m_0 (L_0^2 \dot{\theta}_0^2 + L_1^2 \dot{\theta}_1^2 + 2L_0 L_1 \cos(\theta_1) \dot{\theta}_0 \dot{\theta}_1 + L_1^2 \sin(\theta_1)^2 \dot{\theta}_0^2) + I_0 \dot{\theta}_0^2 + I_1 \dot{\theta}_1^2) \quad (2.2)$$

En appliquant les équations d'Euler-Lagrange et en ajoutant le terme correspondant au moment de force appliqué par l'actuateur (u) ainsi que les termes d'amortissement (α_0, α_1), le système d'équations non-linéaires suivant est dérivé :

$$\begin{aligned}
& \begin{bmatrix} m_0L_0^2 + I_0 + m_0L_1^2\sin(\theta_1)^2 & m_0L_1L_0\cos(\theta_1) \\ m_0L_0L_1\cos(\theta_1) & m_0L_1^2 + I_1 \end{bmatrix} \begin{bmatrix} \ddot{\theta}_0 \\ \ddot{\theta}_1 \end{bmatrix} + \\
& \begin{bmatrix} 2m_0L_1^2\cos(\theta_1)\sin(\theta_1)\dot{\theta}_1 & -m_0L_1L_0\sin(\theta_1)\dot{\theta}_1 \\ -m_0L_1^2\cos(\theta_1)\sin(\theta_1)\dot{\theta}_0 & 0 \end{bmatrix} \begin{bmatrix} \dot{\theta}_0 \\ \dot{\theta}_1 \end{bmatrix} + \\
& \begin{bmatrix} \alpha_0 & 0 \\ 0 & \alpha_1 \end{bmatrix} \begin{bmatrix} \dot{\theta}_0 \\ \dot{\theta}_1 \end{bmatrix} + \begin{bmatrix} 0 \\ -m_1gL_1\sin(\theta_1) \end{bmatrix} + \begin{bmatrix} -u \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}
\end{aligned} \tag{2.3}$$

L'équation 2.3 décrit le système d'équations qu'un contrôleur doit anticiper pour arriver à garder le pendule en équilibre. Comme le problème à un certain âge, il est facile de valider les équations à l'aide de sources externes [7, 8]. Dans le cas où le pendule est près de son sommet ($\theta_1 \approx 0$) et que la vitesse de rotation du bras du pendule se rapproche de 0 ($\dot{\theta}_0 \approx 0$), le problème peut être réduit sous la forme suivante :

$$\begin{aligned}
& \begin{bmatrix} m_0L_0^2 + I_0 & m_0L_1L_0 \\ m_0L_0L_1 & m_0L_1^2 + I_1 \end{bmatrix} \begin{bmatrix} \ddot{\theta}_0 \\ \ddot{\theta}_1 \end{bmatrix} + \\
& \begin{bmatrix} 0 & 0 \\ 0 & \alpha_1 \end{bmatrix} \begin{bmatrix} \dot{\theta}_0 \\ \dot{\theta}_1 \end{bmatrix} + \begin{bmatrix} 0 \\ -m_1gL_1 \end{bmatrix} \begin{bmatrix} \theta_0 \\ \theta_1 \end{bmatrix} + \begin{bmatrix} -u \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}
\end{aligned} \tag{2.4}$$

Ces simplifications permettent l'utilisation d'algorithmes de contrôle linéaires lorsque le pendule de Furuta est près de son point d'équilibre [9].

2.1.2 Méthodes de contrôle

À l'aide des équations d'un système dynamique, il est possible de prédire les corrections à apporter à l'aide d'actuateurs pour que le système atteigne l'état désiré. Pour calculer les corrections, des contrôleurs dits *en boucle fermée* sont utilisés [10]. La commande envoyée au système est calculée à partir de l'état (ou d'une quantité dérivée de l'état) du système et d'un objectif du contrôleur. La forme générale d'un contrôleur à boucle de rétroaction est illustrée à la figure 2.2.

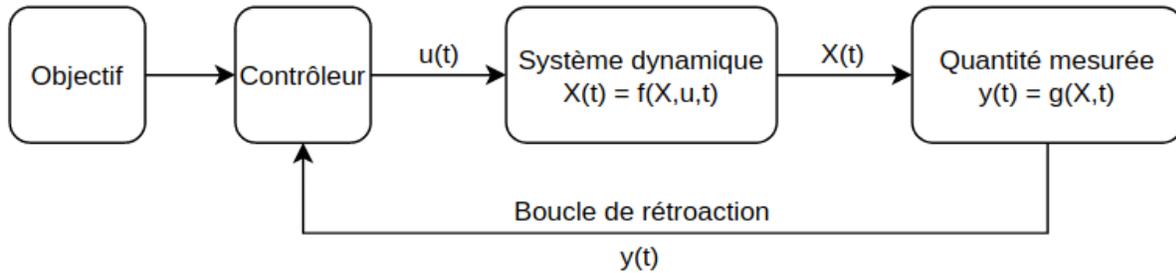


FIGURE 2.2 Schéma général d'un contrôleur à boucle de rétroaction

Le contrôleur envoie une commande $u(t)$, où $u \in \mathbb{R}^{p \times 1}$ et p est le nombre d'actuateurs du système. L'état du système dynamique est influencé par cette commande. Il est décrit par le vecteur $X(t)$, où $X \in \mathbb{R}^{n \times 1}$ et n est le nombre de variables nécessaires à la description de l'état. Des capteurs mesurent ensuite une quantité dérivée de l'état (ou l'état), nommée $y(t)$, où $y \in \mathbb{R}^{q \times 1}$ et q est le nombre de capteurs dans la boucle de rétroaction du système. Les données pertinentes au contrôle sont ensuite renvoyées au contrôleur, qui calcule une nouvelle commande et continue le cycle.

Il existe différentes méthodes pour calculer une commande appropriée à appliquer [11, 12]. Le choix de la bonne méthode dépend grandement du système dynamique qui est sujet au contrôle.

Full State Feedback

Un contrôleur "Full State Feedback" est approprié pour des systèmes dynamiques linéaires où les équations le décrivant prennent la forme :

$$\frac{dX}{dt} = AX + Bu \quad (2.5)$$

$$y = X \quad (2.6)$$

où $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times p}$, $C \in \mathbb{R}^{q \times n}$, $D \in \mathbb{R}^{q \times p}$. Le concept est d'insérer une transformation linéaire choisie pour changer la dynamique du système de façon à le diriger vers un point d'équilibre désiré. La transformation linéaire prend la forme d'une matrice, K . Dans ce cas spécifique, le contrôleur doit recevoir comme entrée l'état du système dynamique. La commande envoyée au système peut donc être décrite par :

$$u = -KX \quad (2.7)$$

où $K \in \mathbb{R}^{p \times q}$ est la matrice de contrôle. Le système dynamique est influencé par le contrôleur d'une façon linéaire. L'entièreté du système, incluant la boucle de rétroaction, est décrit par :

$$\frac{dX}{dt} = (A - BH)X \quad (2.8)$$

Le problème de contrôle revient donc à un *placement de pôles*. La matrice de contrôle, H , doit être choisie de sorte que les points d'équilibre de l'entièreté du système soient aux endroits désirés. Les deux méthodes principales pour le calcul de cette matrice est le placement de pôles et le LQR ("Linear quadratic regression") [10]. Ce type de contrôleur est entre autres utilisé ici pour balancer le pendule de Furuta lorsqu'il est suffisamment près de son point d'équilibre avec succès [13, 14].

Contrôle par modulation d'énergie

Le principe du contrôle par modulation d'énergie est relativement simple, mais difficile à appliquer en pratique. Celui-ci vise à créer un champ de potentiel virtuel en fonction de l'état du système. En choisissant une forme adéquate, des minima peuvent être placés de façon arbitraire à différents états du système. Ce potentiel virtuel est ensuite répliqué en utilisant les actionneurs du système. Un exemple de ce type de contrôleur est celui dérivé par F. Gordillo pour résoudre le problème du pendule inverse [4] avec succès. Cette méthode est plus complexe que le Full State Feedback, mais elle permet de résoudre des problèmes non linéaires, tels que le "swing-up" du pendule inverse ou du pendule de Furuta. L'objectif à atteindre doit être intégré dans le potentiel virtuel calculé d'avance, de sorte que toutes les trajectoires qui peuvent être empruntées par le système dynamique mènent à l'état désiré. Une commande qui peut être physiquement appliquée par les actionneurs du système doit aussi être dérivable à partir de ce potentiel.

Apprentissage par renforcement

Ce type de méthode consiste à utiliser un agent qui est programmé pour apprendre à générer la commande appropriée au contrôle sans avoir de signal de référence. Le principal avantage de ce type de méthode est qu'aucune connaissance *à priori* n'est requise. Une instance impressionnante de ce type de contrôleur est celui développé par H. Kim et al. [15]. Leur contrôleur arrive à piloter un hélicoptère et à effectuer des manoeuvres de compétition de pilotage. Le contrôleur qu'ils utilisent est un réseau neuronal ayant une

architecture spécifique choisie pour optimiser les capacités de contrôle. Il est entraîné en utilisant une méthode similaire au Q-Learning, la méthode *Pegasus* [16] dérivée par leur groupe de recherche.

2.1.3 Évaluation des performances

Les métriques de performances existantes dans le domaine du contrôle dépendent beaucoup du problème à résoudre. Pour certains problèmes de contrôle, par exemple un contrôle en température d'un four, il est désirable de ne jamais dépasser l'objectif. Il est donc dit qu'il ne faut pas, dans ce cas-là, "overshoot". Cette mesure devient une métrique du contrôleur. Dans le cas du pendule inverse ou du pendule de Furuta, les paramètres évalués varient en fonction de l'auteur qui effectue l'étude. Dans certains cas, la stabilité du contrôleur proposé est prouvée mathématiquement à l'aide d'une analyse de Lyapunov [17]. Dans différentes études, la comparaison des trajectoires empruntées par différents contrôleurs (créés localement) est utilisée pour prouver la supériorité de l'un d'eux [18]. Il est donc pertinent d'identifier un cadre d'évaluation local des performances comme il n'existe pas de standard de comparaison. Dans le cadre de ce travail, comme la méthode que nous souhaitons utiliser est expérimentale, des métriques plus rudimentaires seront utilisées. Premièrement, la proportion de temps passé à l'équilibre, puis la comparaison de commande.

Proportion de temps à l'équilibre

La proportion de temps passé à l'équilibre est définie comme le ratio entre le temps passé aux points d'équilibre et le temps total de l'essai, tel que décrit à l'équation suivante :

$$P_{eq} = \frac{T_{eq}}{T_{total}} \quad (2.9)$$

avec T_{eq} représentant le temps passé à un angle inférieur à 15 degrés, et T_{total} représente le temps total d'un essai. Dans le cadre de ces travaux, le temps d'un essai est fixé à 10 secondes en simulation et 25 secondes sur le montage expérimental. La mesure est prise à partir du moment où l'essai ou la simulation est démarré. Toutes les commandes sont arrêtées dès que le temps est écoulé. Dans le contexte où un contrôleur général est désiré à la fin de ce projet, cette métrique permet de rapidement évaluer un contrôleur, de valider la répétabilité des essais et de comparer les performances de différents contrôleurs. Lorsque la proportion de temps à l'équilibre est très basse ($< 10\%$), le contrôleur n'arrive pas à ajouter de l'énergie au système. Lorsqu'elle est faible ($< 25\%$), le contrôleur n'amortit pas le pendule au sommet de sa trajectoire.

Comparaison de commandes

La seconde métrique qui a été choisie est la comparaison de commande générée par le contrôleur avec une référence à l'aide de la racine de la moyenne de l'erreur au carré normalisée (NRMSE). Celle-ci est définie comme suit :

$$NRMSE = \frac{1}{\bar{S}_{exp}} \sqrt{\frac{\sum_{i=0}^N (S_{ref} - S_{exp})^2}{N}} \quad (2.10)$$

avec S_{ref} le signal de référence, S_{exp} le signal expérimental, N le nombre de points du signal et \bar{S}_{exp} la moyenne du signal expérimental. Puisque les contrôleurs utilisés sont entraînés avec une commande de référence comme enseignant, cette métrique est particulièrement pertinente pour évaluer la capacité qu'à un contrôleur à reproduire une commande de référence.

2.2 Reservoir Computing

Le "reservoir computing" est une méthode d'apprentissage machine qui a été développée en parallèle par deux groupes de recherches [19, 20]. Inspirés des "Recurrent Neural Networks", cette méthode a une dépendance temporelle qui permet de conserver des liens entre une entrée vue dans le passé et le signal qui est généré. De plus, sa structure simple permet un entraînement rapide ; seule une régression linéaire est nécessaire.

Le principe d'opération d'un reservoir computer (RC) est de faire ressortir une grande variété de patrons répétitifs à partir d'un signal temporel, puis d'utiliser ceux-ci pour calculer une quantité désirée. C'est un concept qui est très prometteur ; il a été démontré qu'il est possible de contrôler un bras robotisé à l'aide d'un RC [21]. Plus récemment, une démonstration de contrôle de déplacement utilisant une régression sur les données des différents senseurs à même les éléments non linéaires du corps d'un robot a été accomplie [22]. Il est aussi envisageable d'utiliser des microsystemes non linéaires pour jouer le rôle de réservoir [23] du RC. Ce type de dispositif a été testé pour remplir des tâches de prédiction de série chaotique (NARMA-10) [2].

En plus de présenter une capacité d'adaptabilité à différents problèmes, le "reservoir computing" est une méthode qui est simple à appliquer. Ce type de réseau est composé d'un étage d'entrée, d'un réservoir et d'une régression. Le schéma du RC le plus simple est illustré à la figure 2.3.

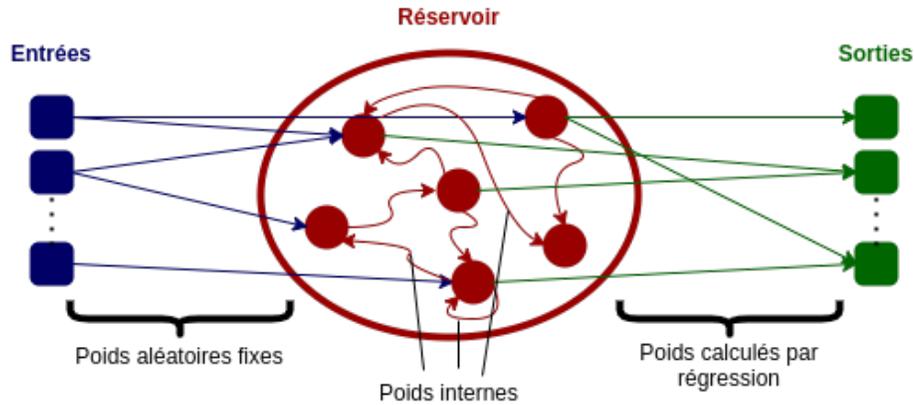


FIGURE 2.3 Schéma d'un RC. L'étage d'entrée est illustré en bleu, le réservoir est en rouge et l'étage de régression est en vert.

L'étage d'entrée lie un nombre choisi par l'utilisateur de données d'entrées à chaque nœud du réservoir. Chaque lien de dépendance des nœuds aux entrées est choisi aléatoirement dû au grand nombre de connexion. Les paramètres des distributions aléatoires sont choisis pour éviter, autant que possible de saturer les nœuds.

Le réservoir peut être vu comme un système dynamique quasi-chaotique. Celui-ci est composé d'un certain nombre de nœuds inter-reliés qui peuvent chacun avoir des états distincts. Chacune de ces interconnexions est fixée aléatoirement à la création du réservoir. À chaque requête faite au RC, le réservoir prend un vecteur d'entrée qui influence les nœuds d'une "façon" différente. L'ancienne valeur du nœud n'est cependant pas complètement remplacée ; une combinaison non linéaire de l'ancienne valeur du nœud, des dépendances aux autres nœuds du réservoir et aux entrées sont effectuées. Ceci assure que les états précédents du réservoir continuent à influencer son état présent, jusqu'à un certain point.

Si le réservoir est approprié au problème, des répétitions de motifs variés émergeront entre différents nœuds. L'étage de régression exploite ces motifs et les lie à une sortie qui correspond au signal désiré à l'aide d'une régression linéaire.

2.2.1 Réservoirs

Le réservoir doit, à chaque incrémentation, générer un ensemble de valeurs de nœuds, nommé état. Ces états sont référés par le symbole $X \in \mathbb{R}^{N \times 1}$, où X est un vecteur représentant la concaténation des valeurs des N nœuds qui composent le réservoir. Il doit aussi avoir certaines propriétés pour arriver à fournir assez d'information à l'étage de sortie. Voici une description sommaire des principales conditions de fonctionnement. Celles-ci sont explorées plus en détail dans un article écrit par M. Lukoševičius [24].

1. Les états du réservoir doivent être récurrents. Des suites d'entrées similaires doivent faire converger le réservoir vers des états similaires. La capacité de généralisation du réservoir en dépend.
2. La mémoire d'une entrée doit décliner avec le temps. Une vieille entrée doit éventuellement cesser d'avoir une influence sur l'état du réservoir.
3. Le réservoir doit être suffisamment chaotique. Il doit y avoir une grande variété d'états récurrents.

Les requis concernant chacune de ces propriétés dépendent du problème à résoudre puisque la succession d'états générés dans le temps dépend des entrées. Pour chaque problème, il est requis de retrouver les hyperparamètres optimaux du réservoir pour se rapprocher des contraintes minimales requises. Différents types de réservoirs remplissent ces critères et peuvent être utilisés avec succès dans un RC.

Echo State networks (ESNs)

Un ESN est un type de RC numérique initialement développé par H. Jaeger [19]. Les 3 étages du RC y sont représentés comme des matrices. $W_{in} \in \mathbb{R}^{N \times N_{in}}$, $W \in \mathbb{R}^{N \times N}$ et $W_{out} \in \mathbb{R}^{N_{out} \times N}$ correspondent respectivement aux étages d'entrée, au réservoir et à la régression. N , N_{in} et N_{out} correspondent respectivement au nombre de noeuds du réservoir, au nombre d'entrées et au nombre de sorties. Dans le cas de l'ESN, chaque élément de la matrice du réservoir W représente un lien entre deux noeuds. L'état du réservoir à chaque incrément de temps se calcule comme suit :

$$X_{n+1} = (1 - \alpha)X_n + \alpha \tanh(W_{in}u_{n+1} + WX_n) \quad (2.11)$$

où $\alpha \in [0, 1]$ est un facteur appelé le taux de fuite, qui influence la capacité de mémoire du RC. Ce facteur a été intégré pour améliorer les performances obtenues [25]. La sortie du réservoir est calculée en multipliant l'état à la matrice de sortie,

$$y_n = W_{out}X_n \quad (2.12)$$

avec $y_n \in N_{out} \times 1$ qui représente la sortie d'un réservoir.

Liquid State Machines (LSMs)

Une LSM est un type de réseau qui est composé d'une façon similaire à un ESN. Il contient un étage d'entrée ainsi qu'un étage de sortie et un réservoir. Cependant, la non-linéarité associée au calcul des valeurs des noeuds est différente. À la place d'utiliser une sigmoïde, une équation différentielle est utilisée pour calculer la valeur de chaque noeud. Conceptuellement, chaque noeud accumule une "charge", jusqu'à atteindre un seuil. Lorsque celui-ci est atteint, le noeud se décharge dans ses voisins, qui accumulent cette charge. Il existe plusieurs modèles de neurones qui représentent ce comportement. L'un des plus utilisés est le modèle "Integrate & Fire", qui a initialement été proposé pour imiter le fonctionnement de neurones biologiques[26]. L'idée a été implémentée en premier par W. Maass et al. [20] sous forme de RC et continue d'être utilisée [27, 28].

Réservoir à rétroaction retardée

Ce concept, proposé et prouvé expérimentalement par L. Appeltant et al. [1], montre qu'il est possible de répliquer un réservoir complet en utilisant un seul noeud. Pour ce faire, le noeud est temporellement couplé avec lui-même. Un nombre de "noeuds virtuels totaux" du réservoir, N , est prédéfini à l'avance, ainsi qu'un temps fixe θ associé à chaque noeud. Une itération complète du réservoir prend donc un temps $\tau = N\theta$. Les entrées sont traitées en les multipliant avec une matrice de poids aléatoires fixe, qui est dans ce cas-ci nommée "masque". Cependant, chaque donnée du vecteur résultant est associée à un noeud virtuel précis et à la fenêtre de temps correspondant pour être injecté dans le réseau. Ce procédé est illustré à la figure 2.4.

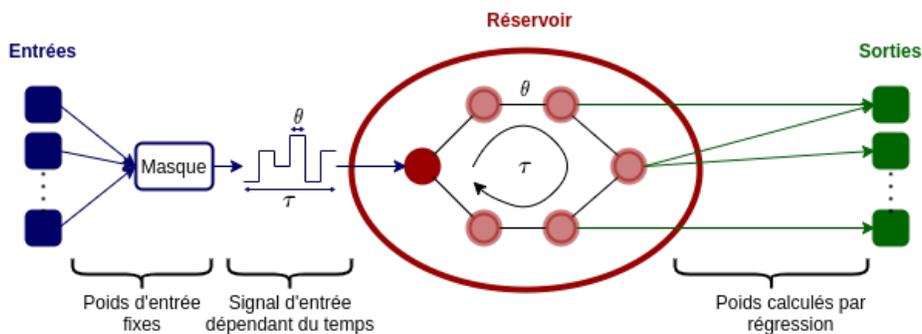


FIGURE 2.4 Schéma d'un RC composé d'un seul noeud couplé dans le temps. L'étage d'entrée est illustré en bleu, le réservoir est en rouge et l'étage de régression est en vert. Le temps associé à la mesure de chaque noeud est θ et le temps d'une itération du réservoir est τ [1].

Pour assurer que le réseau ait une mémoire, la valeur du noeud mesurée à sa dernière itération lui est partiellement réinjectée. Sa dernière valeur est combinée linéairement

avec le signal d'entrée. L'énorme avantage mis de l'avant par cet article est qu'il est possible de concevoir un réservoir fonctionnel à l'aide d'un seul noeud physique. Ceci allège grandement les difficultés de conception et de fabrication de certains types de réservoirs mécaniques, par exemple des réseaux d'oscillateurs mécaniques non linéaires [23].

2.2.2 Étage de sortie

L'étage de sorti du RC est la seule portion du réservoir qui subi un entraînement [24]. Deux paramètres peuvent être contrôlés à cet étage ; le moment du calcul des poids ainsi que la méthode utilisée pour leur calcul. Ces paramètres sont explorés en fonction du problème à résoudre. Le moment du calcul des poids (entraînement "offline" vs "online") est souvent choisi en fonction du type de problème qui est attaqué [29].

Un estimateur de régression linéaire est utilisé pour calculer la matrice de poids $W_{[out]}$. Celui-ci minimise l'erreur au carré entre le signal généré et le signal de référence. Dans le cas d'un RC, une régularisation de Tikhonov est la méthode de choix pour calculer les poids de la sortie [24]. Elle prend la forme suivante :

$$W_{out} = y_{ref} X^T (X X^T + \lambda \Gamma \Gamma^T)^{-1} \quad (2.13)$$

où $X \in \mathbb{R}^{N \times R}$ est l'ensemble des états du réservoir dans le temps, $\Gamma \in \mathbb{R}^{N \times N}$ est une matrice identité multipliée par un paramètre de régularisation λ et $y_{ref} \in \mathbb{R}^{N_{out} \times R}$ est le signal de référence que le RC doit reproduire. Le paramètre R correspond à la taille de la séquence d'entraînement. Cependant, tel qu'il est possible de le noter à l'équation 2.13, ce calcul des poids requiert une inversion matricielle, où la matrice à inverser a une dimension de $N \times N$. Dans le cas où l'inversion doit être effectuée à chaque itération, il est parfois avantageux d'utiliser des méthodes alternatives. Il en existe plusieurs, basées sur les "recursive least square" (RLS) qui permettent d'obtenir un résultat similaire sans inversion matricielle[30]. Ces méthodes sont donc moins lourdes numériquement. Elles sont utilisées pour des problèmes qui requièrent de plus gros réservoirs et où le RC est dans la boucle de rétroaction du problème [29].

2.2.3 Caractérisation des réservoirs

Comme un réservoir peut prendre plusieurs formes, tel que vu à la sous-section 2.2.1, la question suivante peut être posée : comment comparer les différents réservoirs entre eux ?

Plusieurs moyens ont été proposés[31][32], la plupart impliquant la résolution de problèmes étalons conçus pour l'évaluation des propriétés des réservoirs.

L'un de ces problèmes étalons est la résolution de fonctions de parité. Un signal d'entrée $u(t)$ qui correspond à une suite binaire aléatoire, où chaque instance séparée par un délai fixe τ est choisie parmi 1, -1, est fourni au réservoir. Avec ce signal d'entrée, le but est que le réservoir reproduise une série de fonctions, qui ont des dépendances variables au temps. Ces fonctions sont décrites par :

$$P_{N,\delta}(t) = \prod_{i=0}^{N-1} u(t - (i + \delta)\tau) \quad , \quad (2.14)$$

où $P_{N,\delta}(t)$ est la fonction de parité d'ordre N avec un délai fixe de δ , τ est le délai temporel entre deux points consécutifs. Le résultat peut ensuite être évalué en calculant le NRMSE entre le signal généré par le réservoir suite à un entraînement et le signal $P_{n,\delta}(t)$ [2]. Les résultats obtenus à ce test révèlent la capacité du réservoir à retenir de l'information dans le temps et à effectuer des opérations non-linéaires.

Quatre autres propriétés peuvent être utilisées pour évaluer les réservoirs : le Kernel Rank (KR), le Generalisation Rank (GR), la Memory Capacity (MC) et l'Information Processing Capacity (IPC). Les trois premières métriques sont extensivement définies par Dale et al. [31] alors que la dernière est introduite par Dambre et al. [32].

Le KR quantifie la capacité du réservoir à séparer des entrées distinctes. Plus le KR est élevé, plus les fonctions générées sont indépendantes les unes des autres. Cette mesure est calculée en fournissant un signal aléatoire à une dimension à un réservoir. Le KR correspond au rang de la matrice d'états résultante du réservoir. Comme le rang maximal d'une matrice correspond sa dimension minimale, le KR peut au maximum avoir comme valeur le nombre de noeuds du réservoir.

Le GR sert à quantifier l'aptitude d'un réservoir à généraliser lorsque différentes entrées similaires lui sont fournies. Un GR bas signifie une bonne habilité à généraliser des entrées similaires à des états similaires. Pour calculer le GR, la même entrée que celle du KR est utilisée, sauf qu'un bruit y est ajouté. La mesure est exactement la même que le KR, soit le rang de la matrice d'états résultante.

La MC est la mémoire linéaire à court terme. Elle représente la dépendance linéaire de l'état du réservoir aux entrées précédentes. La MC se mesure en sommant les capacités du réservoir à reconstituer différentes entrées passées. La MC est calculée à l'aide de l'équation 2.15.

$$MC = \sum_{k=1}^{2N} MC_k = \sum_{k=1}^{2N} \frac{\text{cov}^2(u(t-k), y_k(t))}{\sigma^2(u(t))\sigma^2(y_k(t))} \quad , \quad (2.15)$$

où N est le nombre de noeuds du réservoir, $u(t)$ est l'entrée qui a été utilisée et $y_k(t)$ la sortie du réservoir entraînée à reproduire l'entrée d'il y a k pas de temps. σ est utilisé pour représenter l'écart-type. Une entrée typique correspond à un nombre de t d'échantillons entre -1 et 1 choisis aléatoirement selon une distribution uniforme. Cette caractérisation est très similaire au test de fonction de parité avec un délai ($\delta \geq 1$) et lorsque seule une fonction linéaire est considérée ($N = 0$).

L'IPC est une mesure similaire à la MC, mais étendue à plus que des fonctions linéaires. L'IPC correspond à la capacité d'un réservoir à reproduire une base de différentes fonctions indépendantes. L'entrée utilisée est la même que pour la MC, soit t échantillons choisis selon une distribution uniforme entre -1 et 1. Des fonctions orthogonales de cette entrée aléatoire (dans le cas de [32], des polynômes de Legendre) sont calculées comme signaux enseignants. Les erreurs entre le signal généré par le RC lorsqu'entraîné et les différentes fonctions orthogonales sont sommées, pour obtenir au final le nombre de fonctions indépendantes générées par le réservoir. Comme le nombre de signaux générés par le réservoir correspond au nombre de noeuds, la valeur maximale de la mesure est le nombre de noeuds du réservoir. La capacité d'un réservoir à reproduire une fonction est définie par

$$C_T = 1 - \frac{\min_{W_i}(MSE_T[y, y_T])}{\langle y_T^2 \rangle} \quad , \quad (2.16)$$

où C_T est la capacité à reproduire la fonction de référence, y_T est la fonction enseignante (Teacher), y est la fonction générée par le RC, $MSE_T[y, y_T] = \frac{1}{T} \sum_{t=1}^T (y(t) - y_T(t))^2$ représente l'erreur moyenne au carré entre la fonction de référence et la fonction générée par le RC, et $\langle y_T^2 \rangle$ représente la moyenne de la fonction de référence au carré. Si C_T est sous un seuil ϵ , la valeur est mise à 0. Ce seuil est différent pour différents estimateurs de fonctions. Selon [32], ce seuil est de 1.7×10^{-4} pour les ESN. Ce processus est répété pour chaque fonction de la base orthogonale, puis toutes ces capacités sont sommées, pour obtenir l'IPC du réservoir.

$$IPC = \lim_{T \rightarrow N} \sum_{l=1}^T C_T \quad (2.17)$$

La valeur maximum de l'IPC d'un réservoir est le nombre total de noeuds du réservoir, si les fonctions générées par chaque noeud sont indépendantes. Comme il est impossible de faire cette sommation pour un nombre infini de combinaisons (degré des fonctions et délais dans le temps), des espaces plus restreints sont mesurés. Par exemple, Dambre et al [32] mesurent les IPC jusqu'au polynômes de neuvième degré.

2.3 Capteur neuromorphique (CN)

Beaucoup de systèmes physiques sont assez non-linéaires pour être utilisés comme réservoir d'un RC [5]. Ceci est prouvé par la grande variété de réservoirs physiques ayant été testés, allant de nanotubes de carbones [33] jusqu'à un corps mou macroscopique [34]. Une fraction de ces systèmes ont le potentiel d'être utilisés comme capteur neuromorphique. Un CN est un type de capteur qui intègre une capacité de traitement de données à la détection des conditions d'un environnement. Pour ce faire, un réservoir physique est couplé physiquement avec un mécanisme influencé par l'environnement.

Dans le cadre de ce travail, le capteur neuromorphique utilisé est un neuroaccéléromètre. Pour arriver à un concept fonctionnel de CN, deux microsystèmes électromécaniques (MEMS) ont été intégrés. Le premier est un accéléromètre capacitif qui consiste en une masse inertielle suspendue et joue le rôle de couche d'entrée. Le second, un oscillateur de Duffing, sert de réservoir. Les deux composantes sont microfabriquées à une distance de $8 \mu\text{m}$ et couplés électrostatiquement. Lorsque le capteur subit une accélération, la masse inertielle se déplace et influence la vibration de l'oscillateur. L'amplitude de ses oscillations modifie la résistance de jauges piézorésistives. Un voltage est lu par une carte d'acquisition externe et sert de noeud. Comme le dispositif ne contient qu'un seul oscillateur, une architecture de réservoir à rétroaction retardée (voir section 2.2.1) est choisie. À partir des valeurs des noeuds, il est ensuite possible d'utiliser un étage de sortie externe pour calculer un signal lié aux accélérations imposées au CN. Une image du CN est présentée à la figure 2.5.

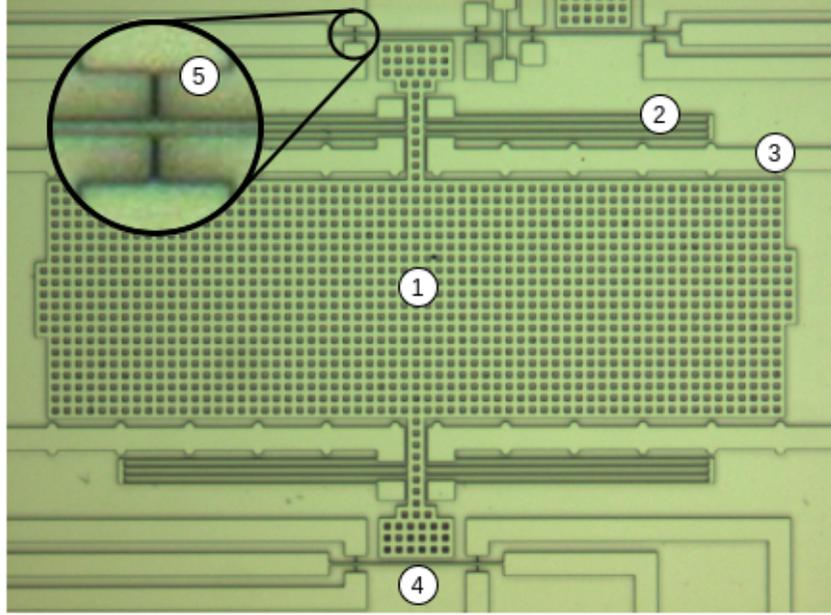


FIGURE 2.5 Image au microscope optique d'un accéléromètre neuromorphique. 1) Masse inertielle suspendue. 2) Ressorts. 3) Arrêt ; prévient les trop grand déplacements. 4) Oscillateur de Duffing. 5) Jauges piezorésistives.

2.3.1 Accéléromètres capacitifs

Un accéléromètre capacitif est composé d'une masse inertielle suspendue par des ressorts qui sont ancrés dans un substrat. Des électrodes sont placées à proximité des extrémités opposées de la masse inertielle (à la place des poutres localisées en 4 à la figure 2.5). Le déplacement de la masse peut être approximé en utilisant un système masse-ressort-amortisseur décrit par :

$$M_{eff}\ddot{x} + \zeta_{eff}\dot{x} + K_{eff}x = F(t) \quad (2.18)$$

où x est le déplacement, M_{eff} est la masse effective, K_{eff} est la rigidité effective, ζ_{eff} est l'amortissement effectif du système et $F(t)$ la force subie. La sensibilité statique de l'accéléromètre, définie par le ratio de son déplacement et l'accélération subie, est donnée par l'équation suivante :

$$\frac{x}{a} = \frac{M_{eff}}{K_{eff}} = \frac{1}{\omega_0^2} \quad (2.19)$$

avec a comme accélération et ω_0 comme fréquence angulaire naturelle. Les systèmes masse-ressort-amortisseur sont bien étudiés, leur fréquence de résonance est donnée par :

$$f_0 = \frac{\omega_0}{2\pi} = \sqrt{\frac{K_{eff}}{M_{eff}}} \quad (2.20)$$

La constante de rigidité du type de ressort utilisé pour les accéléromètres neuromorphiques (voir la figure 2.5) est approximée comme suit[3] :

$$K_{eff} = \frac{4Ewt^3}{L_l^3 + L_c^3} \quad (2.21)$$

où E est le module d'Young du matériau constituant les ressorts dans la direction de déplacement, w est la hauteur du ressort, t est l'épaisseur de chaque branche du ressort, L_l et L_c sont respectivement les longueurs minimales et maximales de branches des ressorts. Une constante additionnelle Q , nommé facteur de qualité, est aussi définie pour décrire les pertes d'énergie à chaque cycle complet d'oscillation. Celle-ci est décrite par l'équation :

$$Q = 2\pi \frac{E_{resonance}}{E_{perdue}} = \omega_0 \frac{M_{eff}}{\zeta_{eff}} \quad (2.22)$$

où $E_{resonance}$ est l'énergie emmagasinée dans l'oscillateur et E_{perdue} est l'énergie perdue à chaque cycle d'oscillation. Un facteur de qualité relativement bas est désiré pour la masse inertielle. Dissiper l'énergie rapidement permet de réduire les oscillations et d'atteindre un état d'équilibre plus rapidement.

Dans le cas où le dispositif serait purement utilisé comme accéléromètre, la mesure de l'accélération est prise à partir de mesures capacitives. Un potentiel constant est appliqué entre la masse inertielle et des électrodes localisées à ses extrémités. Le déplacement de la masse inertielle fait varier la capacité des condensateurs formés par celle-ci et l'électrode. Les variations de capacité sont très petites, de l'ordre de ≈ 50 fF. Un circuit de détection spécialisé [35] doit donc être intégré. Il est à noter que le concept présenté à la figure 2.5 n'est pas du tout optimisé pour être utilisé purement comme accéléromètre. Des concepts plus optimaux sont présentés dans différents ouvrages [36, 35].

2.3.2 Oscillateur de Duffing

Un oscillateur de Duffing est un type d'oscillateur non linéaire qui présente des propriétés intéressantes. Son amplitude d'oscillation change abruptement lorsqu'il oscille près de sa fréquence de résonance, rendant son mouvement chaotique. L'équation utilisée pour modéliser son déplacement est la suivante :

$$\ddot{x} + \frac{\omega_0}{Q}\dot{x} + \omega_0^2 x + \beta x^3 = \frac{F(t)}{M_{eff}} \quad (2.23)$$

où x est l'amplitude du déplacement, ω_0 est la fréquence de résonance de l'oscillateur, Q est le facteur de qualité de la poutre (défini à l'équation 2.22), M_{eff} est la masse effective, $F(t)$ la force appliquée dans le sens du mouvement et β est le paramètre de non-linéarité.

De tels oscillateurs ayant une longueur de l'ordre du millimètre ont déjà été fabriqués par différents laboratoires à une petite échelle, en utilisant des technologies MEMS. Ceux-ci prennent la forme d'une poutre fixée à ses extrémités. L'amplitude des oscillations est mesurée en temps réel avec des jauges piézorésistives [37]. Un exemple de ce type d'oscillateur microfabriqué au 3IT [2] est présenté à la figure 2.6.

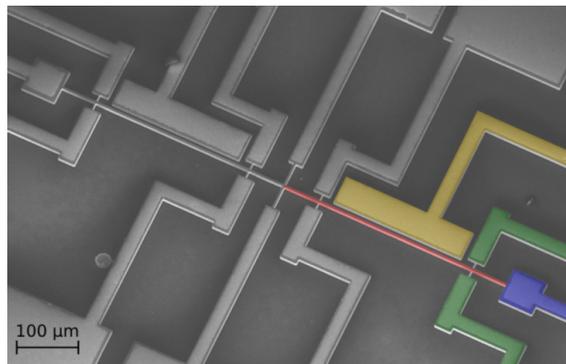


FIGURE 2.6 Image au microscope électronique d'un oscillateur de Duffing (rouge) piloté par une électrode (jaune). Les jauges piézorésistives sont surlignées en vert et l'électrode de mesure est en bleu [2].

Les oscillateurs de Duffing peuvent être amenés à leur résonance en utilisant un couplage électrostatique. Une électrode est microfabriquée à proximité de l'oscillateur, puis un potentiel électrique est appliqué entre les deux composants. Une force est subie par chacun d'eux [38]. Celle-ci est décrite par l'équation suivante :

$$F_e = \frac{-V^2}{2} \frac{\delta C}{\delta d} \quad (2.24)$$

où F_e est la force électrostatique, V est la différence de potentiel, d représente la distance et C est la capacité du condensateur formé par la poutre et l'électrode.

2.3.3 Physical reservoir computing (PRC)

Tel que dit précédemment, l'oscillateur jouera le rôle de réservoir dans le RC physique. Celui-ci est mis à la terre. Un potentiel AC à la fréquence de résonance de l'oscillateur est appliqué sur la masse inertielle au travers de ses ancrages. Celui-ci est modulé selon la formule suivante [2] :

$$V(t) = V_0[m(t) + \alpha(t)((1 - LR)X_{t-1} + LR * X_t)] \cos(2\pi f_d t) \quad (2.25)$$

où V_0 est l'amplitude du signal envoyé, $m(t)$ est le masque appliqué, α est la matrice de dépendances entre les noeuds, LR est le taux de fuite du réservoir, f_d est la fréquence d'alimentation et t est le temps. Cette façon de calculer le signal d'entrée permet d'assurer une continuité entre les valeurs d'un même noeud à différents points dans le temps, en plus de forcer des liens entre les noeuds d'un réseau à l'aide de la matrice α . Le taux de fuite est ajouté pour augmenter la mémoire du réservoir [25]. La fréquence d'alimentation est décrite par $f_d = \frac{f_0}{2}$ où f_0 est la fréquence de résonance de la poutre. Cette relation est obtenue comme la force appliquée sur la poutre est proportionnelle au voltage au carré, et $\cos(2\pi f_0 t)^2 = \frac{1}{2}(\cos(4\pi f_0 t) + 1) = \frac{1}{2}(\cos(2\pi f_d t) + 1)$. La réponse de la courbe à différentes amplitudes de signaux d'alimentation est présentée à la figure 2.7.

L'amplitude des oscillations de la poutre varie de façon non linéaire en fonction de la fréquence et de l'amplitude du signal d'entrée. Il est possible de remarquer sur la courbe brune à la figure 2.7 que lorsque la fréquence est trop élevée, une hystérésis apparaît. Celle-ci ajoute une grande dépendance entre les noeuds voisins d'un réseau. Cependant, il est encore inconnu si sa présence est désirable ; le CN a été testé dans un régime où elle n'était pas présente, soit celui représenté par la courbe rouge de la figure 2.7. Les propriétés physiques de l'oscillateur de Duffing qui a déjà été fabriqué sont présentées dans le tableau 2.2.

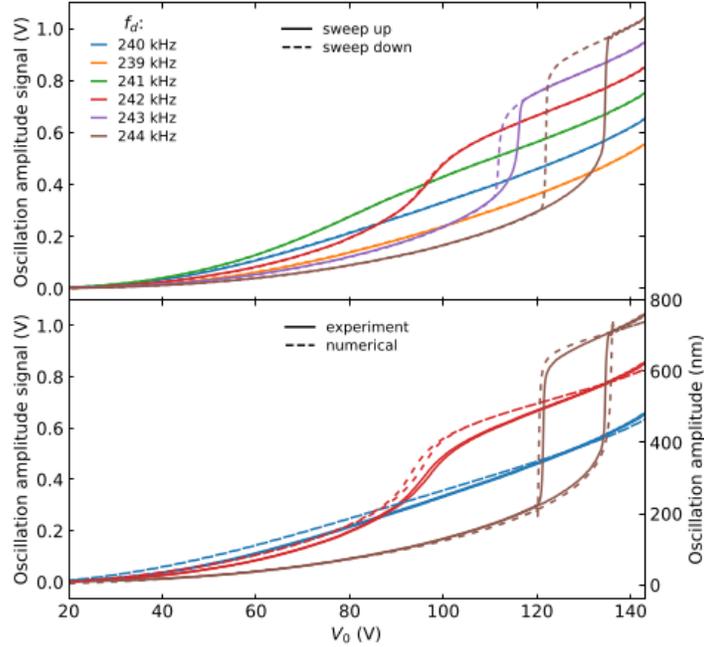


FIGURE 2.7 Mesure de l'amplitude du signal d'oscillation (en haut) et comparaison entre un modèle numérique (en bas) en fonction de l'amplitude du voltage d'alimentation V_0 [3]

Paramètres	Symbole	Valeur
Largeur	w	50 [μm]
Épaisseur	t	3 [μm]
Longueur	l	280 [μm]
Module d'Young du silicone	E	125 [GPa]
Constante de non-linéarité	β	$1.0 \times 10^{24} Hz^2/m^2$
Fréquence de résonance	f_0	482.2 [Hz]
Facteur de qualité	Q	145
Distance entre la poutre et la masses inertielle au repos	d	8 [μm]

TABLEAU 2.2 Propriétés de l'oscillateur de Duffing microfabriqué [3]

Jusqu'à maintenant, un oscillateur de ce type a été utilisé avec succès pour remplir les tâches de prédiction de fonctions de parités (P_2 : 100 %, P_3 : 100 %, P_4 : 100 %, P_5 : 95%, P_6 : 88 %) et de classification de mots énoncés tirés de la base de données TI-46 (taux de succès de 78 %) [2]. Le réservoir physique faisant usage de l'oscillateur a donc une mémoire et performe des transformations non linéaires de ses entrées. La fonctionnalité du couplage avec l'accéléromètre a aussi été validée. Pour ce faire, des signaux numériques ont été convertis en signaux analogues qui actionnent un pôt vibrant. Les tâches remplies

de cette façon sont encore une fois la fonction de parité ($P_2 : 100 \%$, $P_3 : 97.4 \%$, $P_4 : 82 \%$, $P_5 : 70\%$, $P_6 : 63 \%$) et l'émulation d'un modèle de moyenne mobile autorégressive (NARMA) [3]. Les propriétés de l'accéléromètre sur le CN testé sont présentées au tableau 2.3.

Paramètres	Symbole	Valeur
Dimension de la masse inertielle	w_{acc}	590 x 410 [μm^2]
Dimension des trous dans la masse inertielle	t_{acc}	10 x 10 [μm^2]
Masse	m_{acc}	28.8 [ng]
Distance entre la masse inertielle et son arrêt	d_{gap}	5 [μm]
Largeur des branches des ressorts	$w_{ressort}$	50 [μm]
Épaisseur des branches des ressorts	$t_{ressort}$	2 [μm]
Longueur des petites branches des ressorts	L_s	366 [μm]
Longueur des grandes branches des ressorts	L_l	410 [μm]
Fréquence de résonance de la masse inertielle	f_{acc}	1.9 [kHz]

TABLEAU 2.3 Propriétés de l'accéléromètre [3]

Le CN développé au laboratoire est une nouvelle technologie qui présente beaucoup d'inconnues. Malgré le fait que le CN et sa régression soient suffisants pour accomplir des tâches de classification de signaux, des tâches de génération de signaux n'ont pas encore été accomplies. De façon analogue à J. Degraeve et al. [39], un succès sera difficile à atteindre avec une régression simple due à la complexité du signal de contrôle. Il est par contre possible d'intégrer des couches de sorties plus complexes qu'une régression. Des mécanismes de rétroaction peuvent aussi être intégrés en modulant le signal d'entrée par des paramètres additionnels à ceux énoncés dans l'équation 2.25. Aussi, la sensibilité de l'accéléromètre peut être adaptée au problème en modifiant la masse inertielle et ses ressorts. L'ajustabilité de tous ces paramètres, ainsi que les succès rencontrés dans le domaine du contrôle en PRC [22] et en apprentissage machine [15] me donnent bon espoir quant à la faisabilité du projet.

CHAPITRE 3

Simulation et montages expérimentaux

Une approche incrémentale est choisie pour démontrer qu'il est possible d'asservir un système sous-actué à un CN. Des versions "plus faciles" du problème seront tout d'abord résolues, avant de faire une transition vers des problèmes progressivement plus difficiles. Le concept de "facilité" peut être séparé en deux portions : le nombre de sources de bruit présentes dans le système ainsi que la quantité d'information disponible pour générer le signal de contrôle. Certaines contraintes sont aussi ajoutées à chaque itération du problème pour refléter les contraintes du capteur qui a été développé au laboratoire. Par exemple, puisque les entrées du CN sont limitées aux accélérations, seuls les *accélérations ressenties*, c'est-à-dire les accélérations subies par le bout du pendule, seront considérées comme signaux d'entrée pour chacune des variantes du problème. Un seul réservoir sera également utilisé dans les méthodes de contrôle explorées pour que les méthodes développées soient transférables au premier capteur neuromorphique miniaturisé.

3.1 Simulation

La première variante du problème considéré est une simulation d'un chariot-tige (cart-pole). Ce problème est le plus facile puisque le bruit est absent (à moins d'en ajouter) et chaque essai est entièrement reproductible. Aussi, le référentiel du pendule est translationnel plutôt que rotatif, ce qui diminue le nombre de forces qui influencent l'accélération subie par celui-ci. La méthode utilisée pour faire la simulation est celle d'Euler Explicite [40]. Ce choix a été fait puisqu'une discrétisation des équations du problème par différence finie est facile à obtenir et que cette simplicité permet une intégration rapide de différents types de contrôleurs. Le système d'équations qui représente le chariot-tige est obtenu en empruntant une approche Lagrangienne [41] du problème. Les constantes suivantes sont utilisées : m_0 représente la masse du chariot, m_1 la masse du pendule, L_1 la longueur du bras du pendule et I_1 l'inertie du pendule. Sous forme matricielle, les équations qui représentent ce problème ont la forme :

$$\begin{aligned} & \begin{bmatrix} m_1 + m_0 & m_1 L_1 \cos(\theta_1) \\ m_1 L_1 \cos(\theta_1) & I_1 \end{bmatrix} \begin{bmatrix} \ddot{q} \\ \ddot{\theta}_1 \end{bmatrix} + \\ & \begin{bmatrix} \alpha_0 & -m_1 L_1 \sin(\theta_1) \dot{\theta}_1 \\ 0 & \alpha_1 \end{bmatrix} \begin{bmatrix} \dot{q} \\ \dot{\theta}_1 \end{bmatrix} + \begin{bmatrix} 0 \\ -m_1 g L_1 \cos(\theta_1) \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix}. \end{aligned} \quad (3.1)$$

La variable q est utilisée pour représenter la position du chariot sur son axe de déplacement horizontal. La variable θ_1 est la position angulaire du pendule, suivant la même convention que celle introduite pour le pendule de Furuta. f représente la force appliquée sur le chariot et les constantes α_i sont les coefficients d'amortissement ou de friction. Chaque matrice peut être substituée par une variable la représentant. L'équation peut donc être ré-écrite comme

$$M \begin{bmatrix} \ddot{q} \\ \ddot{\theta}_1 \end{bmatrix} + N \begin{bmatrix} \dot{q} \\ \dot{\theta}_1 \end{bmatrix} + P = F. \quad (3.2)$$

Ce système peut ensuite être représenté de façon équivalente par

$$\begin{bmatrix} \ddot{q} \\ \ddot{\theta}_1 \end{bmatrix} = M^{-1} \left(-N \begin{bmatrix} \dot{q} \\ \dot{\theta}_1 \end{bmatrix} - P + F \right). \quad (3.3)$$

En connaissant la position du chariot-tige ainsi que sa vitesse, il est possible de calculer les accélérations angulaire et linéaire correspondantes. À partir de ces nouvelles données, la position et la vitesse au prochain incrément de temps peuvent être calculées approximativement à partir des règles d'actualisation de la méthode d'Euler-Explicite.

Il est possible de constater à la figure 3.1 que lorsque le pas de temps est trop élevé, des erreurs commencent à s'accumuler sur les trajectoires empruntées. Pour choisir un pas de temps suffisamment petit, un seuil d'erreur tolérable sur l'énergie totale du système sans actuation est fixé à 5% de l'énergie initiale du pendule. Ce seuil est respecté lorsque le pas de temps est fixé à 1.125 ms, tel qu'il est possible de l'observer à la figure 3.2.

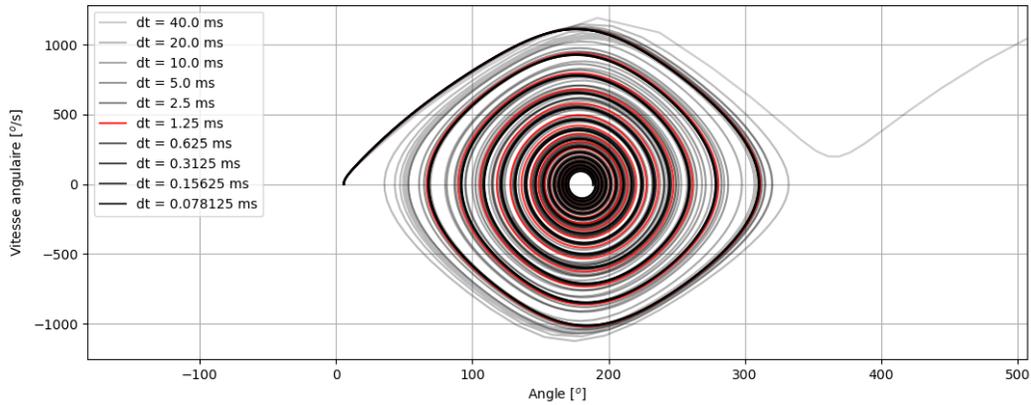


FIGURE 3.1 Influence du pas de temps sur la trajectoire calculée pour le pendule. Lorsque de grandes vitesses sont atteintes, un plus petit pas de temps est requis pour conserver une précision minimale. Les valeurs des constantes physiques de la simulation sont présentées au tableau 3.2.

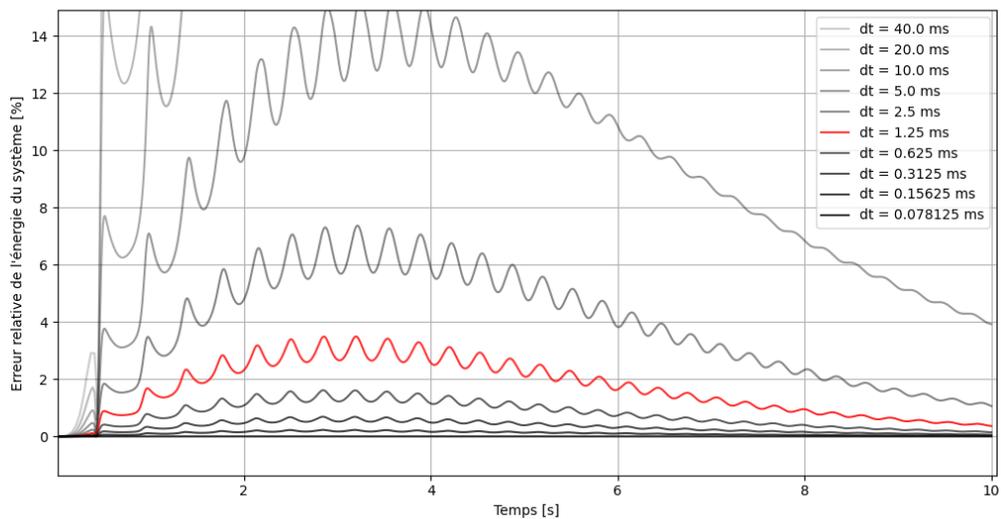


FIGURE 3.2 Erreur relative à l'énergie initiale du système dans le temps pour différents pas de temps. La trajectoire de référence utilisée est une simulation avec un pas de temps de $78.125 \mu s$. Les trajectoires associées aux courbes d'erreur sont illustrées à la figure 3.1. Les paramètres de la simulation sont présentés au tableau 3.2

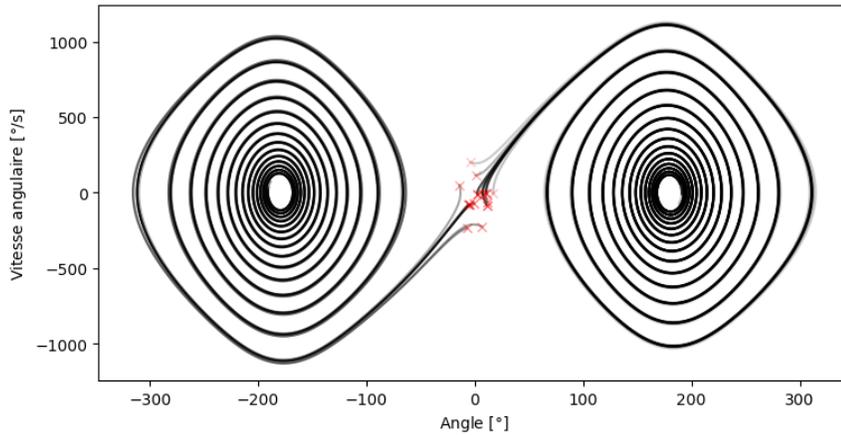


FIGURE 3.3 Exemple de différentes trajectoires générées par simulation. Le point de départ de celles-ci est choisi près du sommet du pendule pour bien illustrer les deux attracteurs localisés à -180° et 180° , qui représentent le pendule à son état de repos.

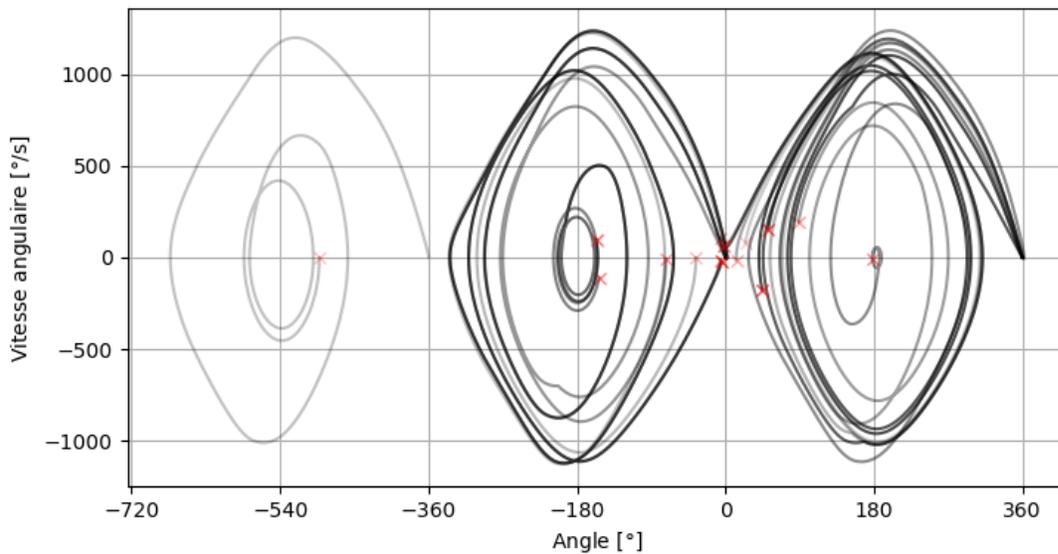


FIGURE 3.4 Exemple de trajectoires actives où la commande de balancement vers le sommet est calculée à partir du contrôleur proposé par Gordillo et al. [4] et la commande d'équilibre est calculée avec un contrôleur LQR. Les X rouges représentent les conditions initiales de chaque trajectoire.

Des trajectoires du pendule sans application de force externe avec un pas de temps $\delta t = 1.125$ ms sont illustrées à la figure 3.3.

Un contrôleur de référence a été programmé pour le pendule simulé. Celui-ci utilise une méthode de Lyapunov. Un potentiel d'énergie virtuel est reproduit par un actuateur externe pour amener le pendule vers un point d'équilibre qui est au minimum de la somme des potentiels gravitationnels et virtuels. Cette approche est présentée plus en détail dans Gordillo et al. [4]. Pour simuler les contraintes du montage expérimental, les commandes ne sont changées qu'à chaque 5 ms, donc à chaque 4 itérations. Aussi, un contrôleur LQR [10] est aussi utilisé pour atteindre l'équilibre. Celui-ci a été ajouté pour reproduire la structure de contrôleur du montage expérimental. Des exemples de trajectoires empruntées pour différentes conditions initiales avec l'ajout du contrôleur sont présentés à la figure 3.4. Les propriétés du contrôleur utilisé ainsi que les paramètres de la simulation sont présentées aux tableaux 3.1 et 3.2.

Constantes de l'algorithme de Gordillo		Symbole	Valeur
Paramètre d'amplitude du potentiel virtuel		A	1.4
Paramètre du terme de damping		K	6.5
Paramètre d'ajustement de l'espacement entre l'énergie réelle et H^*		ϵ_1	0.8
Paramètre d'ajustement des glissement du contrôleurs		ϵ_2	0.5
Paramètre pour réduire les déplacements horizontaux		λ	0.5
Constantes du contrôleur LQR		Symbole	Valeur
Gain de θ_0		K_3	0
Gain de θ_1		K_1	23.36602578
Gain de $\dot{\theta}_0$		K_4	1.99998984
Gain de $\dot{\theta}_1$		K_2	3.039907

TABLEAU 3.1 Paramètres des contrôleurs utilisés

Paramètre	Symbole	Valeur	Unité
Masse cumulative (bras & pendule)	m_0	1.364	[kg]
Constante d'amortissement du déplacement horizontal	α_0	1	[kg · m/s]
Masse du pendule	m_1	0.202	[kg]
Longueur du pendule	L_1	10.8	[cm]
Inertie du pendule	I_1	0.00233	[kg · m ²]
Constante d'amortissement du pendule	α_1	0.001	[kg · m ² /s]
Force maximale appliquée à l'axe de déplacement horizontal	u	6.7	[N]
Incrément de temps	δt	1.125	[ms]

TABLEAU 3.2 Constantes utilisés dans la simulation du chariot-tige

3.2 Montage du pendule de Furuta

Dans le but de tester le CN, un montage physique du pendule de Furuta a été construit par des stagiaires avant le début de cette maîtrise. Un servomoteur est utilisé pour contrôler l'axe de rotation du bras. L'axe du pendule est laissé libre. Pour envoyer des données à l'ordinateur, quatre différents microcontrôleurs sont utilisés. Le premier récupère l'angle du bras à l'aide d'un encodeur optique lié à l'axe du moteur et mesure θ_0 . Le second est attaché à une batterie sur le bras du pendule. Il interface avec un encodeur optique AEDB-9340-B13C qui mesure l'angle du pendule θ_1 . Les deux autres microcontrôleurs sont chacun liés à des accéléromètres Adafruit ADXL345 dans le but de mesurer les accélérations dans les 3 axes cartésiens orthogonaux de leur référentiel. Ils sont positionnés au bout du pendule, à la position qui sera occupée par le CN. Des photos du pendule de Furuta sont présentés à la figure 3.5 où les principaux composants sont identifiés.

Les accélérations mesurées seront utilisées lors de l'intégration d'un ESN au montage expérimental. Toutes les informations récupérées par les microcontrôleurs sont transmises à l'ordinateur central par le réseau wifi d'un routeur dédié. Les propriétés physiques du pendule de Furuta sont présentées au tableau 3.3.

Paramètre	Symbole	Valeur	Unité
Masse cumulative (bras & pendule)	m_0	1.364	[kg]
Longueur du bras rotatif	L_0	7.8	[cm]
Inertie de l'axe de rotation vertical	I_0	0.005	[kg · m ²]
Constante d'amortissement de l'axe vertical de rotation	α_0	0	[kg · m ² /s]
Masse du pendule	m_1	0.202	[kg]
Longueur du pendule	L_1	10.8	[cm]
Inertie du pendule	I_1	0.0109	[kg · m ²]
Constante d'amortissement du pendule	α_1	0.000317	[kg · m ² /s]
Moment de force appliqué à l'axe vertical	u	0.154	[N · m]

TABLEAU 3.3 Constantes physiques du pendule de Furuta et leur valeurs

Le contrôleur choisi pour le pendule de Furuta est similaire à celui présenté dans le cadre de la simulation de façon intentionnelle. La commande est calculée à partir de l'état du pendule. Lors du balancement vers le haut, le contrôleur a été dérivé avec une méthode de Lyapunov [4]. Au sommet de la trajectoire, les équations peuvent être linéarisées. Un contrôleur LQR est utilisé lorsque l'angle du pendule est sous 15 degrés. Les paramètres des deux contrôleurs sont présentés au tableau 3.4.

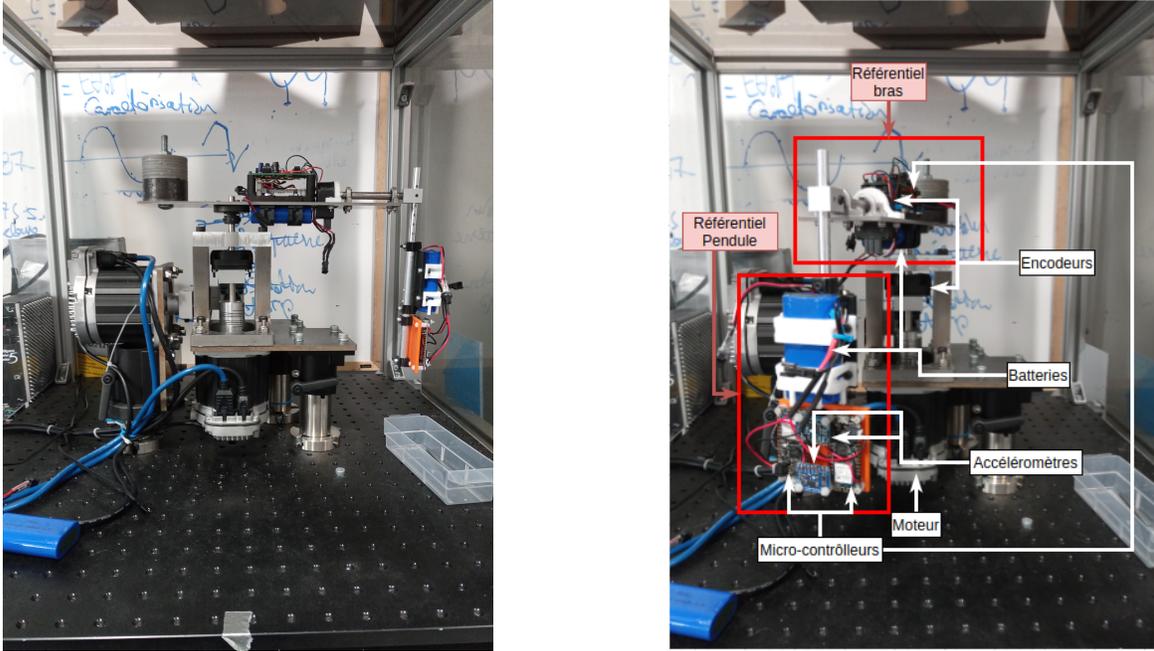


FIGURE 3.5 À gauche - Vue latérale du pendule de Furuta. À droite - Identification des différents composants localisés sur le pendule de Furuta ainsi qu'en rouge les circuits d'acquisition des données.

Constantes de l'algorithme de Gordillo	Symbole	Valeur	Unités
Paramètre d'amplitude du potentiel virtuel	A	1.1	
Paramètre du terme de damping	K	12.3	
Paramètre d'ajustement de l'espacement entre l'énergie réelle et H^*	ϵ_1	1.7	
Paramètre d'ajustement des glissement du contrôleurs	ϵ_2	0.001	
Paramètre pour réduire les déplacements horizontaux	λ	0.5	
Facteur de normalisation de la vitesse	N_v	0.1081	$\left[\frac{s}{m}\right]$
Angle de changement de contrôleur	θ_{switch}	15	$[^\circ]$
Constante du LQR	Symbole	Valeur	
Gain de θ_0	K_3	0	$\left[\frac{1}{rad}\right]$
Gain de θ_1	K_1	190	$\left[\frac{1}{rad}\right]$
Gain de $\dot{\theta}_0$	K_4	1.5	$\left[\frac{s}{rad}\right]$
Gain de $\dot{\theta}_1$	K_2	10	$\left[\frac{s}{rad}\right]$

TABLEAU 3.4 Paramètres des contrôleurs utilisés

Comme le montage du pendule est physique, celui-ci présente une certaine stochasticité. Bien que ces différences soient d'abord mineures à chaque incrément de temps, elles s'accumulent et influencent la génération de commande de contrôle. Différents essais sur le même montage expérimental aux mêmes conditions initiales n'emprunteront pas toujours la même trajectoire. Ceci est illustré à la figure 3.6, où un ensemble de trajectoires dans l'espace de phase du pendule partageant les même conditions initiales sont présentées. Tel

qu'il est possible de le constater, les trajectoires empruntées varient, et parfois plus d'une rotation complète est effectuée.

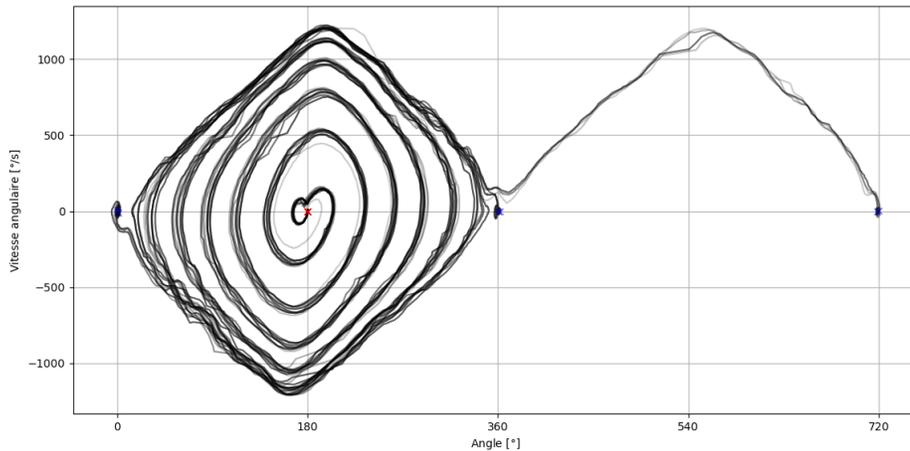


FIGURE 3.6 Ensemble de 10 trajectoires du pendule de Furuta sélectionnées aléatoirement parmi les tests effectués avec le contrôleur de référence. Le pendule est initialement au repos, avant que le moteur du bras force une impulsion de 0.154 Nm pendant 0.5 seconde. Les croix rouges représentent les conditions initiales du pendule alors que les croix bleues représentent la dernière position du pendule avant que le contrôleur soit interrompu, 25 secondes plus tard.

Pour mesurer la performance d'un contrôleur qui arrive partiellement à atteindre l'équilibre, une métrique doit être établie, soit la proportion de temps total passé au-dessus de 15° . Il est à noter que le temps total de chaque essai et le couple maximal du moteur influenceront grandement cette métrique. Pour le premier de ces deux facteurs, si le contrôleur arrive à maintenir le pendule à l'équilibre, le balancement du pendule vers la position verticale occupera de moins en moins de temps. Pour le second, une plus grande puissance peut être utilisée pour faire le balancement du pendule vers le haut, réduisant ainsi le temps minimal de balancement. Pour éviter des variations de la métrique indépendants du contrôleur, les deux variables sont fixés à des valeurs constantes sur le montage expérimental, soit 25 secondes à partir du moment où l'essai est démarré pour le temps total et le couple maximal est fixé à 2% du couple maximal du servomoteur.

Une bonne représentation des performances d'un contrôleur est illustrée à la figure 3.7. Les performances de 100 différents essais avec le contrôleur de référence sont présentées sous forme d'histogramme, donnant ainsi une "distribution de performance" pour ce contrôleur.

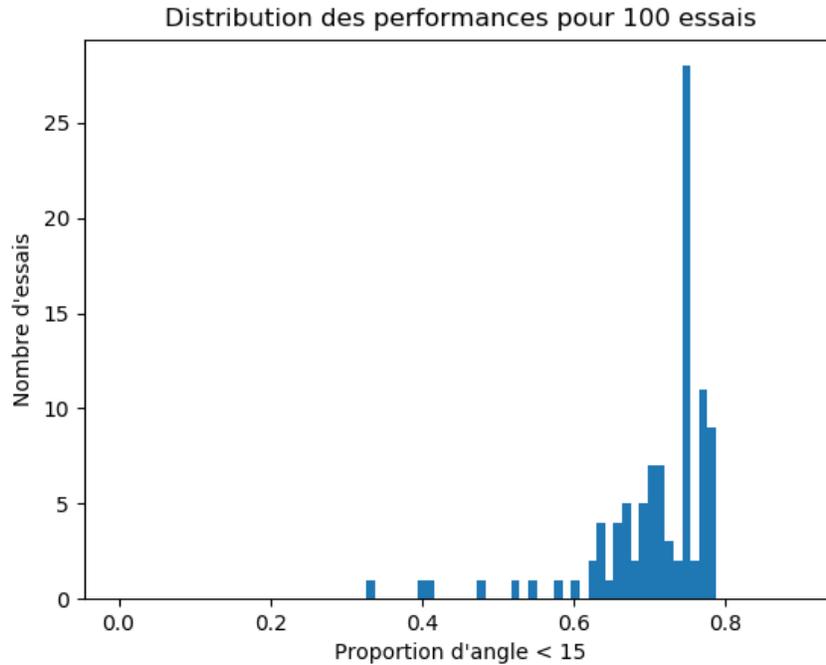


FIGURE 3.7 Variation de la proportion de temps passé au-dessus de 15° pour un contrôleur idéal. Chaque essai est d'une durée de 25 secondes. Entre chacun d'eux, le pendule est forcé à sa position initiale, soit $\theta_1 = 0, \dot{\theta}_1 = 0$.

La variation dans la performance du contrôleur est attribuée à certains effets non linéaires qui ne sont pas compensés par le contrôleur, comme la friction sèche des différents joints. L'essai moyen passe $(62.7 \pm 5.8) \%$ du temps total dans une position où le pendule est à un angle de moins de 15 degrés ($\theta_1 < 15^\circ$), attestant que le contrôleur de référence utilisé arrive à faire le balancement du pendule et à le maintenir droit sur son axe pour des durées prolongées.

3.3 Banc de test du CN

Pour caractériser et interpréter les données générées par le capteur neuromorphique, un montage expérimental est nécessaire. Celui-ci est un assemblage de blocs qui prennent chacun un rôle différent au niveau de la génération et du traitement des données. Un schéma générique est présenté à la figure 3.8. Tel qu'il est possible de le constater, il y a deux "circuits" en parallèle. Le premier est la boucle de rétraction du RC et le second est le circuit de l'actionneur. Dans le premier circuit, on retrouve la génération et l'interprétation des données liées au reservoir computing. Dans le second, on retrouve tout ce qui est lié à la génération des données d'accélération qui seront ressenties par le capteur neuromorphique.

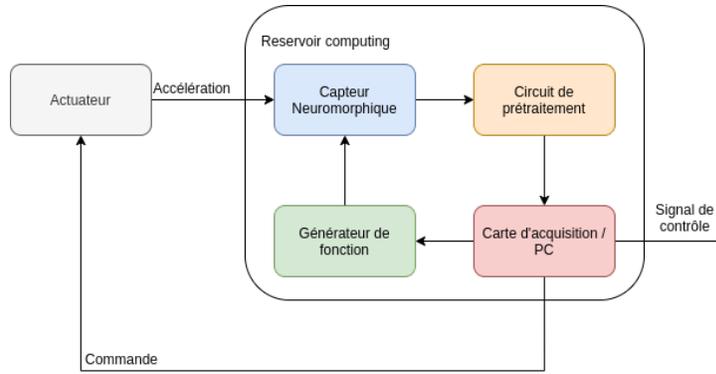


FIGURE 3.8 Schéma bloc du banc de test du capteur neuromorphique.

Le premier bloc est le circuit de prétraitement. Comme la mesure d’amplitude des oscillations de la poutre provient des piezo-résistances du MEMS, le signal est très petit et assez bruité. Le circuit a donc été conçu pour amplifier la mesure tout en atténuant le bruit autant que possible. Il est composé de multiples étages. Dans l’ordre et tel qu’illustré à la figure 3.9, ceux-ci sont : un pont de Wheatstone avec les deux jauges piezorésistives, un amplificateur différentiel, un filtre passe-bande, un préamplificateur, un détecteur d’enveloppe puis un dernier amplificateur. Le voltage de sortie est ensuite mesuré périodiquement par l’ADC d’une carte d’acquisition National Instrument PCIe-6374. Les mesures sont ensuite regroupées et transférées à l’ordinateur. Sur ces données, l’ordinateur applique le masque, le taux de fuite et la rétroaction et les renvoie au temps approprié sur le DAC de la carte d’acquisition. Ces données sont finalement multipliées au signal d’alimentation par un générateur de fonction RIGOL DG4162, puis amplifiées par un amplificateur haut-voltage avant d’être envoyées au capteur neuromorphique.

Au niveau de l’acquisition des données sur l’ordinateur, la réception des données de l’ADC est accompagnée par un incrément de temps qui correspond à la fréquence de mesure. La nouvelle valeur du temps est utilisée pour choisir la valeur d’accélération planifiée à cet instant. Cette valeur est convertie à un voltage correspondant par un DAC de la carte d’acquisition. Ce voltage est ensuite lu par l’actuateur et l’accélération est physiquement reproduite par un actuateur. AU laboratoire, deux actuateurs sont disponibles pour effectuer des tests avec le capteur neuromorphique ; un pot vibrant et un moteur linéaire. Les schémas de bloc associés à ces deux actuateurs sont légèrement différents, comme le pot vibrant est asservi en accélération, alors que le moteur linéaire est asservi en position. Ceux-ci sont présentés à la figure 3.10.

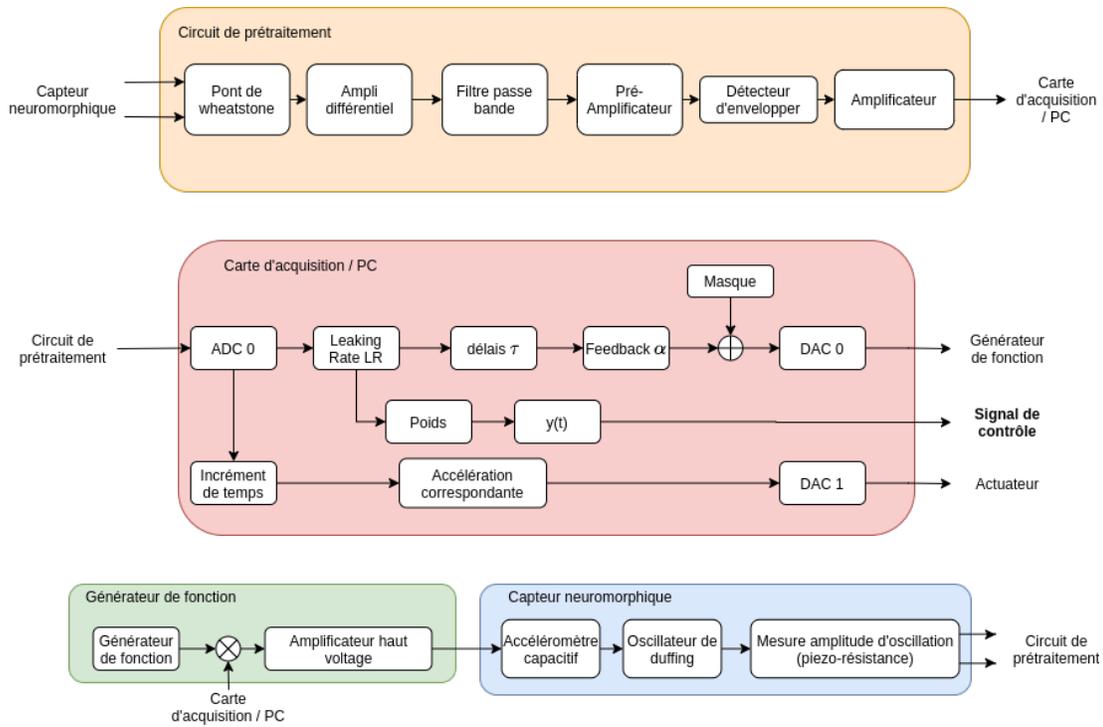


FIGURE 3.9 Fonctionnement général de chacun des blocs qui composent la boucle de rétroaction électronique du système.

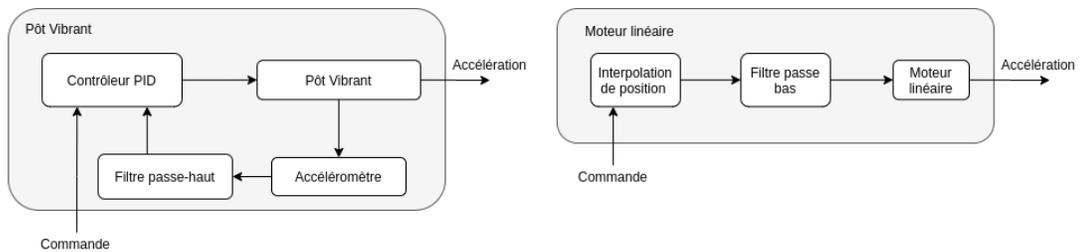


FIGURE 3.10 Schéma de bloc des deux actuateurs qui peuvent être utilisés pour reproduire des accélérations sur le banc de test.

CHAPITRE 4

Echo State Networks

Dans le but de vérifier qu'il est possible de créer un contrôleur fonctionnel avec un RC, une version numérique plus flexible que le CN, un ESN, est initialement utilisée. Ces réseaux seront utilisés pour confirmer que l'accélération peut être utilisée pour générer une commande de contrôle appropriée avec des ESN. Par le fait même, les conditions au contrôle pourront être déterminées.

Différents environnements seront utilisés pour qu'une progression soit observable avec différentes méthodes. Tout d'abord, la simulation sera utilisée, puis le montage expérimental. Dans ces deux cas, différents types d'entraînements ainsi que différentes structures de contrôleurs seront évalués selon les difficultés rencontrées.

4.1 Simulation

Les premiers tests utilisant des ESN comme contrôleur sont effectués à l'aide de la simulation du cart-pole dont les paramètres sont détaillés à la section 3.1. Les tests initiaux ont tous été effectués avec des ESN qui prennent comme entrée les accélérations dans le référentiel du pendule selon les trois axes orthogonaux, illustrés à la figure 4.1.

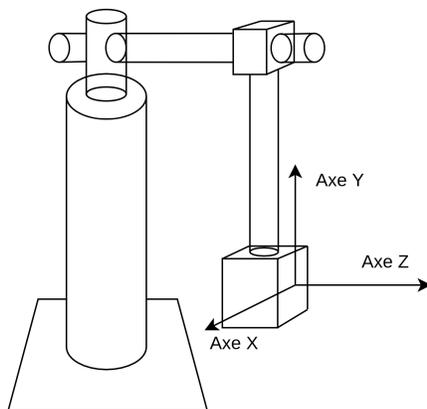


FIGURE 4.1 Axes des accélérations utilisées par l'ESN.

Comme le système évalué en simulation est en fait le problème du chariot-tige (un pendule inverse sur un qui se déplace horizontalement), seuls deux axes d'accélérations seront utilisés comme entrée de l'ESN : les axes X et Y.

4.1.1 Entraînement hors-ligne

Les premiers tests ont été effectués très simplement. 500 essais de référence sont tout d’abord générés avec le contrôleur présenté à la section 3.1. Pour chacun de ces essais, les coordonnées, les vitesses, les accélérations, le temps et les commandes générées par le contrôleur sont enregistrés. Ceux-ci sont par la suite séparés en deux groupes selon une proportion 70-30, soit le groupe d’entraînement et le groupe de test. Le contrôleur ESN est par la suite entraîné à reproduire les commandes de référence avec le groupe d’entraînement. Il est par la suite rapidement évalué en comparant la commande qu’il génère en réponse aux entrées des essais du groupe de test à la sortie de référence. Une métrique de performance, la racine de l’erreur au carrée normalisée (NRMSE) moyenne entre le signal de contrôle généré et le signal de contrôle de référence pour tous les essais du groupe de test sert de point de comparaison.

Paramètres	Valeur
Entrée(s)	2 accélérations orthogonales
Sortie(s)	1 Force
Rétroaction	Aucune
Temps d’update	5 ms
Durée d’un essai	10 s

TABLEAU 4.1 Paramètres fixes pour chaque essai d’entraînement hors-ligne

Cette procédure d’entraînement et d’évaluation est répétée pour un grand nombre de réservoirs, avec des hyperparamètres dans l’espace présenté au tableau 4.2.

Hyperparamètre	Minimum	Maximum
Taux de fuite	0	1
Rayon spectral	0.1	1
Parcimonie des poids	0.01	0.5
Échelle d’entrée	0.5	1.5
Échelle du biais	0.5	1.5
Parcimonie d’entrée	0.01	0.5
β	1e-9	1e-1

TABLEAU 4.2 Bornes des hyperparamètres de réservoirs optimisés par les différents type de recherche.

Tel que discuté à la section 2.2.1, les performances sont grandement influencées par les paramètres du réservoirs. Les différentes combinaisons d’hyperparamètres sont comparées

sur la base du NRMSE moyen obtenu par l'ESN suite à un entraînement. À partir de cette métrique, il est aussi possible de déterminer le nombre d'essais optimal. Ce nombre est déterminé empiriquement en observant le point d'inflexion de la courbe de performance. Tel qu'illustré à la figure 4.2, après un certain nombre d'essais, l'entraînement cesse d'avoir un impact significatif.

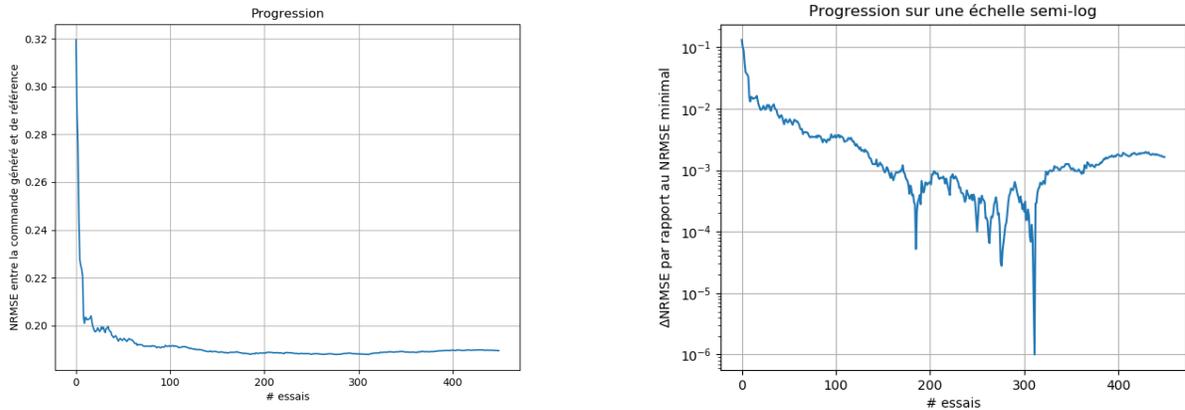


FIGURE 4.2 À gauche - Progression des performances après chaque nouvel entraînement sur un ensemble de données de test de 50 essais. À droite - même graphique où le NRMSE minimal a été soustrait sur une échelle semi-log. Ce graphique permet de représenter visuellement la progression par rapport au NRMSE minimal atteint.

Le nombre d'essais d'entraînement idéal est entre 150 et 200 essais d'entraînements pour le problème du pendule inverse. Pour la suite des entraînements hors lignes, 180 essais d'entraînements et 20 essais de validation seront utilisés pour évaluer un réservoir.

Au niveau de la sélection des réservoirs, trois méthodes d'exploration ont été utilisées : exploration par grille, exploration Monte-Carlo et exploration par optimisation bayésienne (à l'aide d'une librairie python, Optuna). Un exemple de chaque type d'entraînement est présenté à la figure 4.3.

La méthode pour converger non aléatoirement vers des hyperparamètres et fournir le plus grand nombre de réservoirs optimaux est l'optimisation bayésienne. Celle-ci sera donc utilisée à chaque fois qu'une recherche de réservoir est requise. Pour aider à réduire les temps de recherche, les bornes des différents hyperparamètres sont rapprochées au fil des différentes itérations de recherche pour permettre aux algorithmes de recherche de converger plus rapidement.

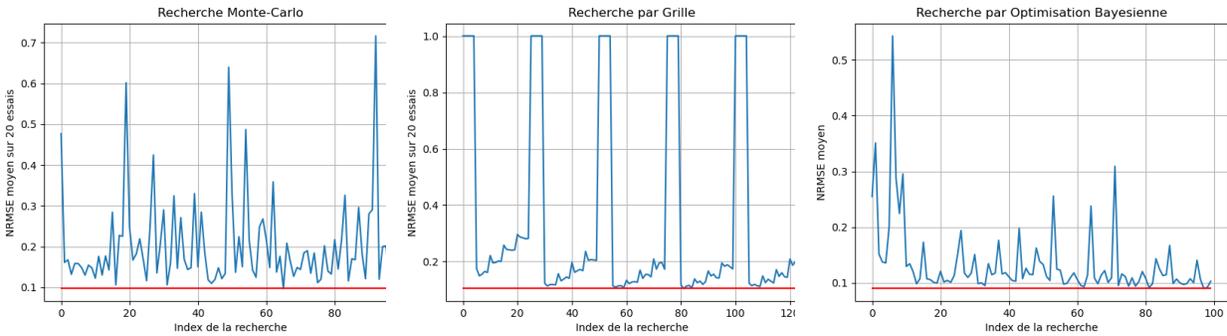


FIGURE 4.3 À gauche - Recherche de réservoir optimal à l'aide de la méthode de Monte-Carlo. Au centre - Recherche de réservoir optimal par grille. À droite - Recherche de réservoir optimal à l'aide d'optimisation bayésienne.

À la fin du processus de sélection de réservoir, l'ESN le plus performant selon la métrique de NRMSE est enregistré puis utilisé de concert avec la simulation. Celui-ci prend la place du contrôleur et génère à chaque 5 ms une commande pour la simulation. Un exemple de trajectoire et de commande résultante est présenté à la figure 4.4. Dans le but de comparer visuellement les performances des différents types de contrôleurs sans avoir à visualiser chacune des trajectoires, une métrique de proportion de temps de l'essai passé sous un angle de 15° (introduite à la section 2.1.3) est utilisée. La proportion de chacune des trajectoires empruntées lorsque le contrôleur ESN génère les commandes est compilée pour former un histogramme. Celui-ci permet de facilement comparer les performances de différents contrôleurs.

Le contrôleur ESN n'arrive jamais à stabiliser le pendule au sommet de sa trajectoire. Tel qu'il est possible de le constater à la figure 4.4, la proportion de temps où l'angle entre le pendule et son axe est sous un angle de 15 degrés est nettement inférieure lorsque le contrôleur ESN préentraîné est utilisé. En moyenne, la proportion de temps passé près de son point d'équilibre est de $18.4 \pm 2.6 \%$. Aussi, lorsque les diagrammes de phase sont examinés (voir figure 4.4), il est possible de constater que malgré le fait que l'ESN arrive à effectuer le balancement vers sa position verticale ou bien à ajouter de l'énergie au système, le pendule n'est jamais ralenti puis mis à l'équilibre. Une hypothèse est posée pour continuer de progresser ; l'instabilité du contrôleur ESN est due à l'atteinte d'états du réservoir qui n'ont pas été rencontrés pendant l'entraînement. Puisque l'ESN est dans la boucle de rétroaction, les erreurs par rapport à la commande du contrôleur de référence s'accumulent dans le système dynamique du "cartpole". L'ESN atteint alors des états du réservoir (et du système dynamique) qui sortent du contexte qui lui a été enseigné. Pour remédier à ce problème, une méthode d'entraînement alternative est explorée.

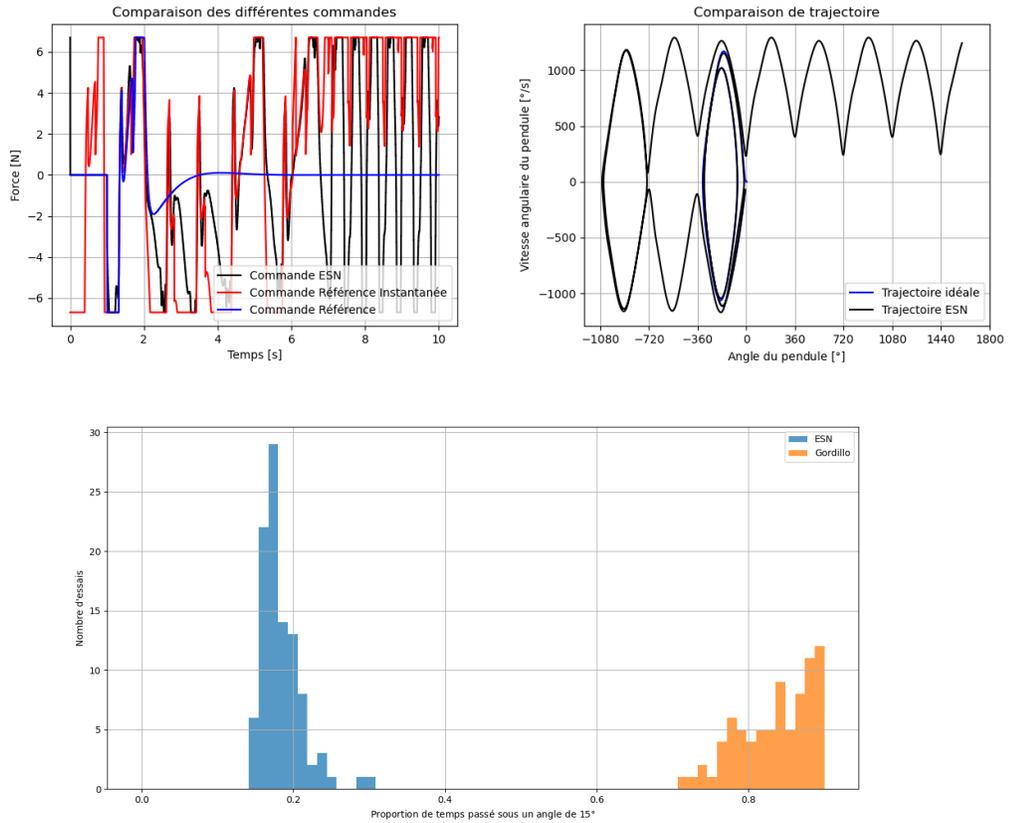


FIGURE 4.4 En haut à gauche - Comparaison de la série de commandes idéales aux commandes générées par ESN préentraîné et aux commandes de référence instantanées. La série de commandes idéales correspond aux commandes générées pour la même situation initiale lorsque le contrôleur de référence effectue entièrement le contrôle. La commande de référence instantanée est la commande générée par le contrôleur de référence à chaque itération lorsque l'ESN contrôle la simulation. En haut à droite - Comparaison entre les trajectoires empruntées par les contrôleurs de référence et ESN préentraînés pour la même condition initiale. En bas - Comparaison des performances des deux contrôleurs pour 100 essais avec les mêmes conditions initiales.

4.1.2 Entraînement par lots

L'entraînement par lots remplace l'entraînement hors-ligne une fois qu'un réservoir a été choisi. Le concept est le suivant : l'ESN est mis en contrôle de la simulation dès le départ sans être entraîné a priori. À chaque fois que la fin d'un essai est atteinte, l'ESN est entraîné sur les données, c'est-à-dire que la couche de sortie est mise à jour avec les nouvelles données du dernier essai. La nouvelle couche de sortie est ensuite utilisée à l'essai suivant. Les données générées sont ensuite utilisées pour l'entraînement subséquent, et ainsi de suite jusqu'à ce qu'un certain nombre d'essais aient été accomplis. Ce processus permet au contrôleur ESN de commettre des erreurs et d'apprendre à compenser celles-ci selon le contexte. Le contrôleur de référence est encore utilisé comme enseignant. Les paramètres de chaque essai en simulation sont présentés au tableau 4.1.

Les méthodes d'entraînement qui incluent le réseau dans la boucle de rétroaction du système dynamique nécessitent plus de temps d'entraînement ; lors d'un changement de réservoir, chaque essai en simulation doit être recommencé, comme les états de réservoir obtenus à chaque itération seront différents. Les ensembles de données ne peuvent donc plus être préenregistrés. Pour sauver du temps au niveau de la recherche de réservoir, une recherche avec un entraînement hors-ligne qui minimise le NRMSE est effectuée avant d'entraîner un réservoir par lots. Les réservoirs qui permettent de générer des commandes ayant un NRMSE minimal par rapport aux essais de tests sont ensuite sélectionnés pour être réentraînés avec cette méthode.

Sortie simple

Les premiers essais qui utilisent l'entraînement par lots ont été effectués avec une seule sortie. La structure de sortie est donc dite "simple". Le réservoir ayant démontré la meilleure performance lors de la recherche effectuée à la section 4.1.1 est sélectionné et la couche de sortie est réentraînée avec un entraînement par lots.

Tel qu'il est possible de voir à la figure 4.5 il y a une amélioration par rapport aux contrôleurs ESN entraînés hors-ligne. Cette nouvelle méthode d'entraînement permet d'obtenir des performances moyennes de (50.8 ± 7.6) % avec le même type de contrôleur. Malgré le fait que cette amélioration est significative, elle n'est pas suffisante pour obtenir une distribution d'essais similaire à celle du contrôleur de référence, qui a des performances de (85.2 ± 5.4) % selon la même métrique de proportion de temps passé à 15° du point d'équilibre. Pour expliquer la différence avec le contrôleur de référence, les noeuds du réseau ainsi que l'angle du pendule dans le temps sont examinés.

En observant les différents états du réseau de l'ESN dans le temps (voir figure 4.6), il est possible de remarquer que les noeuds du réservoir ont deux régimes. Le premier se pro-

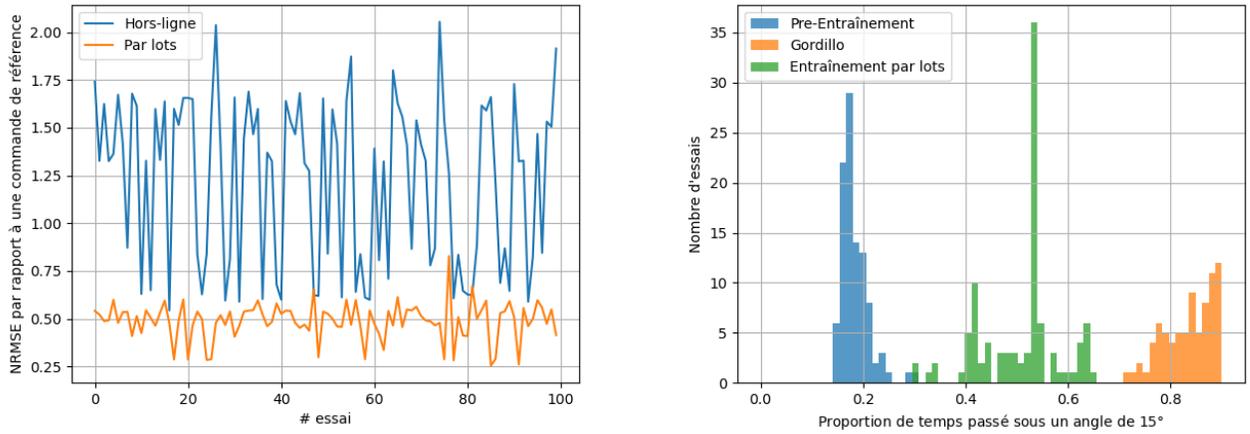


FIGURE 4.5 À gauche - Comparaison du NRMSE entre la commande de référence instantanée et la commande générée par l'ESN dans chacun des deux méthodes d'entraînement. À droite - Comparaison des performances des 3 contrôleurs pour 100 essais. Dans les deux méthodes d'évaluations, une amélioration des performances de l'ESN est observée lorsque l'entraînement par lot est utilisé.

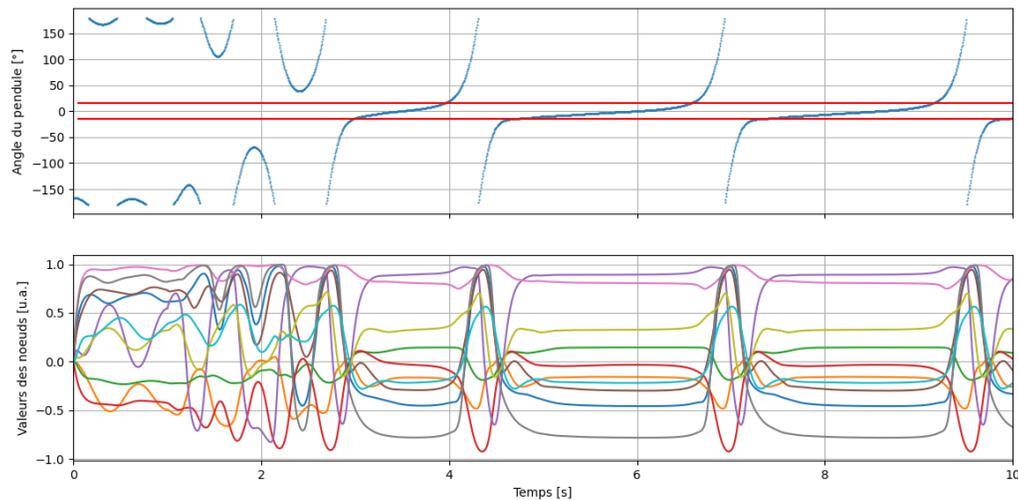


FIGURE 4.6 En haut - Illustration de l'angle du pendule dans le temps. Les deux lignes rouges représentent les angles de -15 et 15 degrés. En bas - Signal de 10 noeuds choisis aléatoirement par rapport au temps. L'amplitude des variations des noeuds est beaucoup plus petite lorsque le pendule est près du sommet de sa trajectoire.

duit pendant le balancement du pendule vers son point d'équilibre et montre une grande amplitude de variation dans les états du réservoir. Le second régime est atteint lorsque le pendule est près de son apex. L'amplitude des oscillations des noeuds est alors beaucoup plus petite. En s'inspirant d'un "tiling" de l'espace pour exploiter ces différences de régimes, une approche d'ESN avec une sortie composée est développée, tout en conservant l'utilisation d'un seul réservoir.

Sortie composée

Tel que mentionné en conclusion de la section 4.1.2, il existe plusieurs régimes des noeuds du réservoir, ainsi que plusieurs régimes de la fonction de sortie. En assignant une régression différente à chacun de ces régimes, la fonction générée par le réseau dans chacun des différents régimes reproduit mieux chacun d'eux et permet d'exclure des dépendances à certains noeuds qui nuisent à la reproduction de fonction de référence dans un régime, mais qui sont bénéfiques aux autres. Comme il y a plusieurs sorties (ou plusieurs régressions), une sélection doit être effectuée parmi celles-ci à chaque moment. Pour résoudre ce problème, une régression de classification peut être entraînée. Un schéma de la structure d'un tel RC est présenté à la figure 4.7. Cependant, dans le cadre de ce mémoire, seule une approche de sélection basée sur l'angle du pendule est prise pour simplifier l'entraînement.

Au niveau de l'entraînement, la même méthodologie que précédemment est utilisée. Le meilleur réservoir d'une recherche "hors-ligne" est sélectionné, puis entraîné par lots. Comme une sortie composée utilise deux régressions, le signal enseignant ainsi que les valeurs des noeuds à chaque incrément sont utilisés pour l'entraînement exclusif de l'une ou l'autre des régressions. Le réservoir présenté au tableau 4.2 est réutilisé. Le RC à sortie composée testé dans cette section a seulement deux régressions ; chacune d'elle est associée à un régime du réservoir. Pour simplifier la tâche, le choix de régression est basé sur l'angle atteint par le pendule.

Il est possible d'observer à la figure 4.8 qu'il y a effectivement une amélioration de la performance moyenne de la tâche de contrôle par rapport aux deux contrôleurs ESN précédents. L'équilibre est atteint lors de certains essais, même s'il reste précaire. Le contrôleur finit toujours par échapper le pendule après un temps suffisamment long. Un exemple de trajectoire de ce type est présenté à la figure 4.9.

Cependant, les performances du contrôleur ESN ne rejoignent toujours pas celle du contrôleur de référence. À ce point-ci, l'hypothèse est posée que le réseau manque d'information pour générer une fonction de sortie similaire à celle du contrôleur de référence. Une solution à ce problème serait d'ajouter en entrée une donnée autre que l'accélération. Par contre il n'est pas possible de faire cela en respectant les contraintes du capteur neuromorphique.

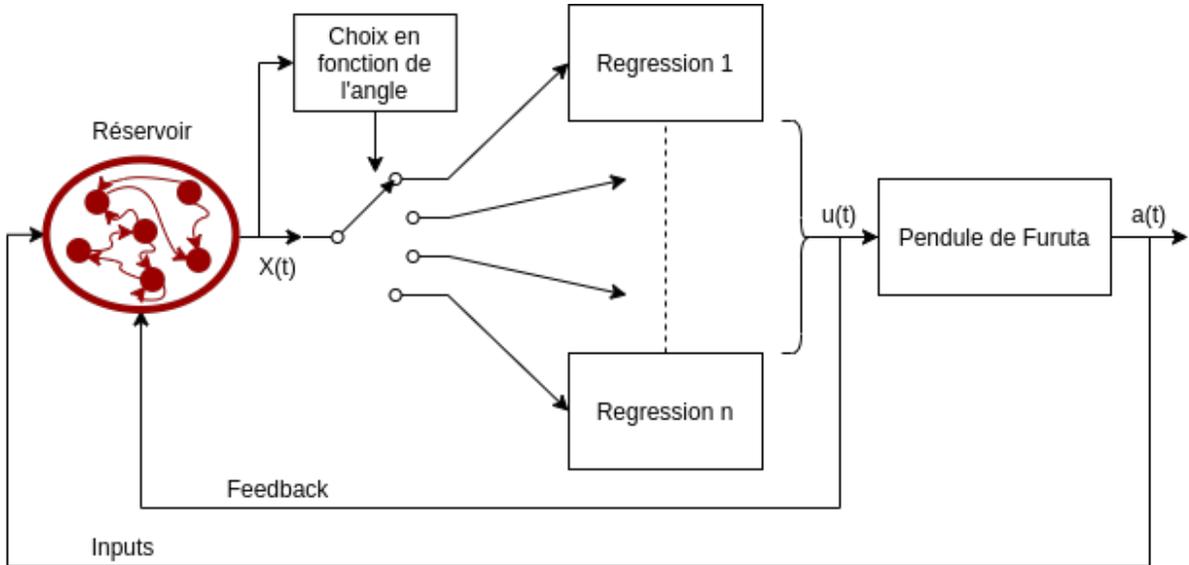


FIGURE 4.7 Schéma d'un RC à sortie composée. Un nombre arbitraire de régressions peut-être sélectionné pour discrétiser l'espace de la fonction à reproduire. Le choix de régression peut être effectué à l'aide d'un RC préentraîné tel qu'illustré, ou bien à partir de mesures prises sur le système dynamique tel que l'angle du pendule. $X(t)$, $u(t)$ et $a(t)$ représentent respectivement les états du réservoir, la commande générée par la régression et l'accélération du pendule du Furuta.

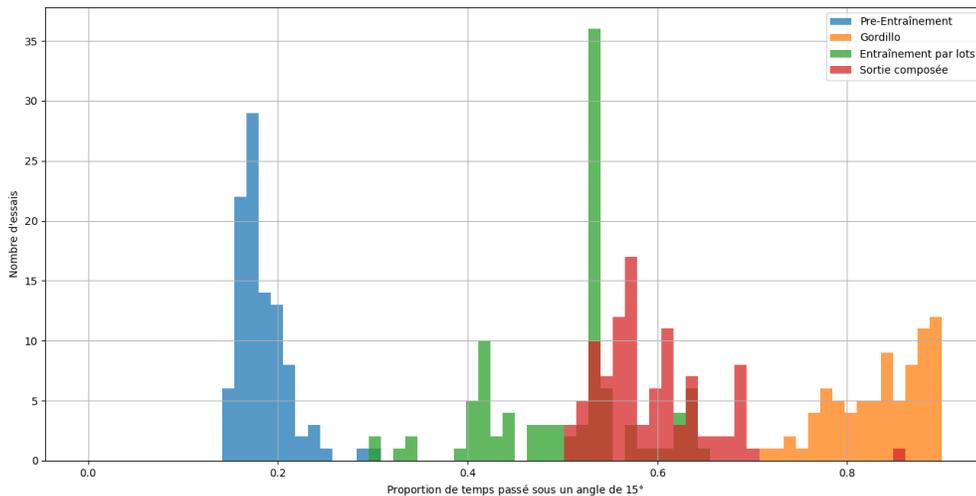


FIGURE 4.8 Comparaison des performances des 4 contrôleurs pour 100 essais. Une augmentation de la proportion du temps total passé près du point d'équilibre est observée pour le contrôleur ESN entraîné par lot avec une sortie composée. En moyenne, le contrôleur maintient le pendule près de son point d'équilibre (59.1 ± 5.6) % du temps total d'un essai.

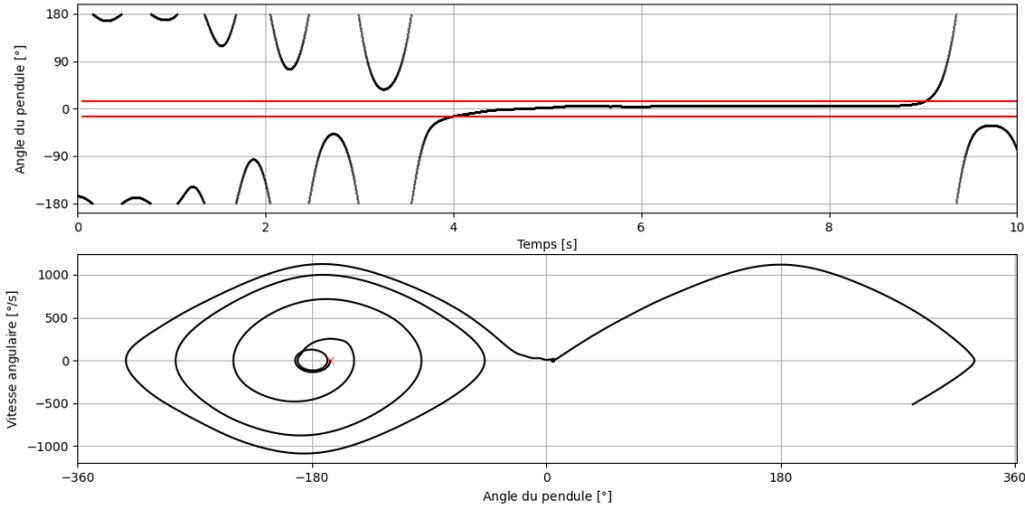


FIGURE 4.9 Exemple de trajectoire empruntée par un contrôleur composé en simulation. En haut, la position angulaire du pendule en fonction du temps. En bas, la trajectoire empruntée par le pendule sur son diagramme de phase. Il semble y avoir un biais qui a été introduit dans l’entraînement, de sorte que son point de stabilisation est près de 5 °.

L’ESN génère toutefois à chaque itération une commande. Cette commande est accessible en tout temps, incluant sur le capteur neuromorphique. Le concept de rétroaction, initialement présenté dans [29] est donc intégré. Afin de tester cette nouvelle modification indépendamment des autres, une structure à une seule sortie est choisie pour les premiers tests. Le contrôleur résultant est un ESN avec une rétroaction.

Rétroaction

La rétroaction consiste à ajouter comme entrée la donnée générée par l’ESN à sa dernière itération. Ceci permet d’ajouter de l’information sur l’état du système dans le réservoir sans nécessiter de capteur additionnel. Pour appliquer ce principe, l’équation 2.11 pour calculer les états d’un ESN est légèrement modifiée :

$$X_{n+1} = (1 - \alpha)X_n + \alpha \tanh(W_{in}u_{n+1} + WX_n + W_{fb}Y_n) \quad (4.1)$$

L’ajout du terme W_{fb} permet d’ajuster la dépendance du réseau à la sortie Y de l’itération précédente sans affecter les autres paramètres. Cet ajout vient toutefois avec son lot de modifications ; comme une nouvelle entrée est ajoutée au réseau de façon effective, il est nécessaire d’effectuer une nouvelle recherche de réservoir. Des modifications doivent

toutefois être effectuées pour la recherche. En plus des nouveaux paramètres qui ont trait à la matrice de rétroaction, un signal doit être dirigé vers celle-ci à chaque itération de la simulation. Dans ce cas-ci, la commande de référence précédente est utilisée dans le but d'imiter le cas où l'ESN arrive à parfaitement reproduire le contrôleur de référence.

Malgré une baisse notable des performances lors de la recherche de réservoir (NRMSE 0.93 avec rétroaction vs 0.12 hors-ligne), une amélioration est observable lorsque l'entraînement par lots est effectué avec le meilleur réservoir. Avec la rétroaction, le pendule passe en moyenne (81.2 ± 6.4) % de son temps à 15° du point point d'équilibre, par rapport à (59.1 ± 5.6) % avec une sortie composée et (85.2 ± 5.4) % pour le contrôleur de référence.

Tel qu'il est possible de voir à la figure 4.11 il y a une amélioration par rapport aux performances des entraînements hors-lignes sans rétroaction. Au niveau des noeuds de ce type d'ESN, les deux régimes qui ont justifié le contrôleur composé sont encore observables à la figure 4.12. Les diagrammes de phase de différents essais sont présentés à la figure 4.13. Le contrôleur par rétroaction arrive à ajouter de l'énergie au pendule puis à le maintenir à l'équilibre.

Avec ce dernier succès en simulation, un passage est effectué au montage expérimental. La méthode retenue pour les premiers essais avec le montage expérimental est l'entraînement par lots couplé avec un ESN qui utilise une rétroaction.

4.2 Montage expérimental

Avant de commencer à tester différents ESNs sur le pendule de Furuta, un cadre expérimental est défini. Pour chaque essai individuel, des conditions initiales aléatoires sont atteintes à l'aide du contrôleur de référence. Un temps aléatoire entre 0.5 et 12 secondes est sélectionné selon une distribution uniforme au début de chaque essai. La commande de référence est appliquée pour cette durée de temps laissant ainsi le montage expérimental avec des coordonnées $(\theta_1, \dot{\theta}_1, \ddot{\theta}_1)$ variées. L'angle θ_0 est choisi aléatoirement. Cette valeur n'a aucun impact dû à la symétrie radiale du pendule de Furuta autour de l'axe de rotation du bras. Chaque essai a une durée totale de 25 secondes, incluant le temps aléatoire choisi initialement. Le temps aléatoire biaise la métrique de proportion lorsque le contrôleur de référence a le contrôle pour des durées prolongées de l'initialisation. Pour le contrôleur de référence, le temps de mise à l'équilibre moyen est de 9.3 secondes. Certains essais seront donc biaisés positivement comme ils atteindront et maintiendront l'équilibre temporairement. Ce biais est cependant appliqué à tous les contrôleurs ESN. À la fin de chaque essai, un temps de relaxation de 10 secondes est ajouté pour permettre au pendule de reprendre sa position au repos ($\theta_1 = 180^\circ$, $\dot{\theta}_0 = 0 \text{ rad/s}$, $\ddot{\theta}_1 = 0 \text{ rad/s}$).

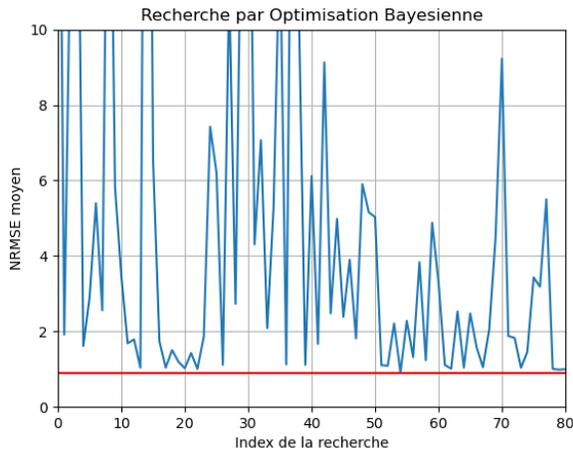


FIGURE 4.10

À gauche, figure 4.10 : Recherche par optimisation bayesienne pour le NRMSE. Pour simuler la rétroaction, la commande enseignant de l'itération précédente est fournie à la place de la commande générée par l'ESN.

À droite, tableau 4.3 : Hyper-paramètres du réservoir ayant les meilleures performances.

Hyper-paramètre	Valeur
Taux de fuite	0.2288
Rayon spectral	0.3896
Parcimonie des poids	0.0382
Échelle d'entrée	0.1361
Échelle du biais	0.1362
Parcimonie d'entrée	0.0478
Échelle de rétroaction	0.3268
Parcimonie de rétroaction	0.3595
β	0.1

TABLEAU 4.3

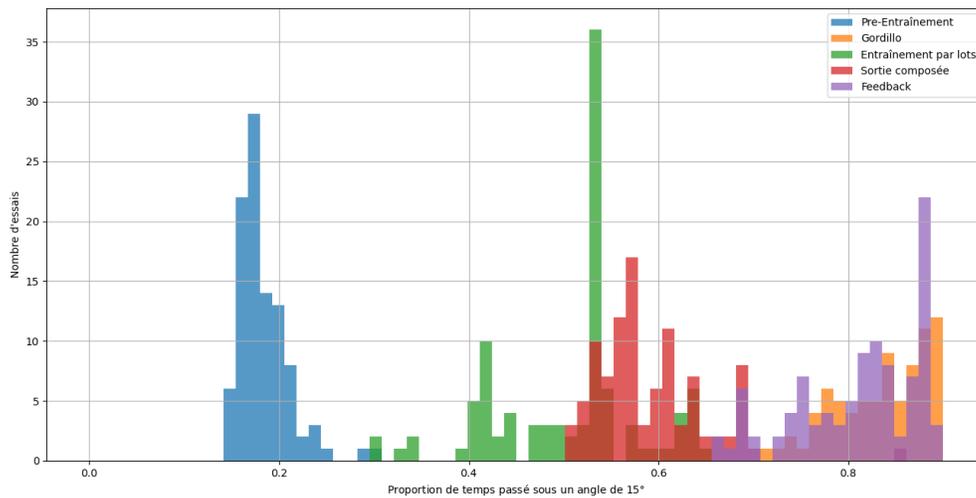


FIGURE 4.11 Comparaison des performances des 5 contrôleurs pour 100 essais. Une amélioration notable est obtenue par le contrôleur avec rétroaction par rapport aux autres structures d'ESN. Ses performances sont même similaires à celles du contrôleur de référence.

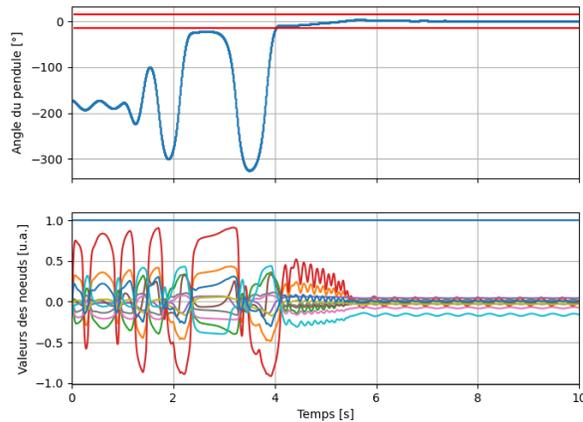


FIGURE 4.12 Angle du pendule dans le temps et représentation des noeuds du réseau.

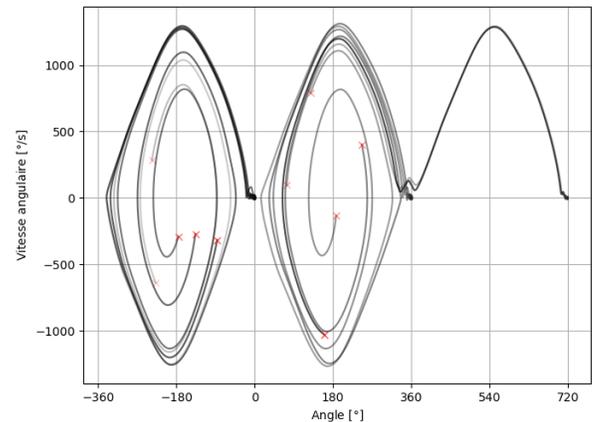


FIGURE 4.13 Trajectoires empruntées par la simulation lorsqu'un contrôleur ESN avec rétroaction entraîné est utilisé.

Le réservoir qui effectue le contrôle est choisi d'une façon similaire à la simulation. Un ensemble d'essais d'une durée de 25 secondes utilisant seulement le contrôleur de référence sont effectués. Lors de ceux-ci, les accélérations, les angles et les vitesses du pendule de Furuta sont enregistrés. Ces données sont ensuite utilisées pour faire une recherche hors-ligne d'ESN avec rétroaction par optimisation bayésienne. Le réservoir le plus performant en termes de NRMSE entre la commande de référence et la commande générée suite à l'entraînement hors-ligne est sélectionné pour être entraîné par lots. Les bornes de recherche pour chaque hyperparamètre sont présentées au tableau 4.4. Les paramètres liés à la rétroaction (Échelle et parcimonie de rétroaction) sont intégrés à la recherche dès le départ. La progression de la recherche au fil de ses différents essais de réservoir est présentée à la figure 4.14.

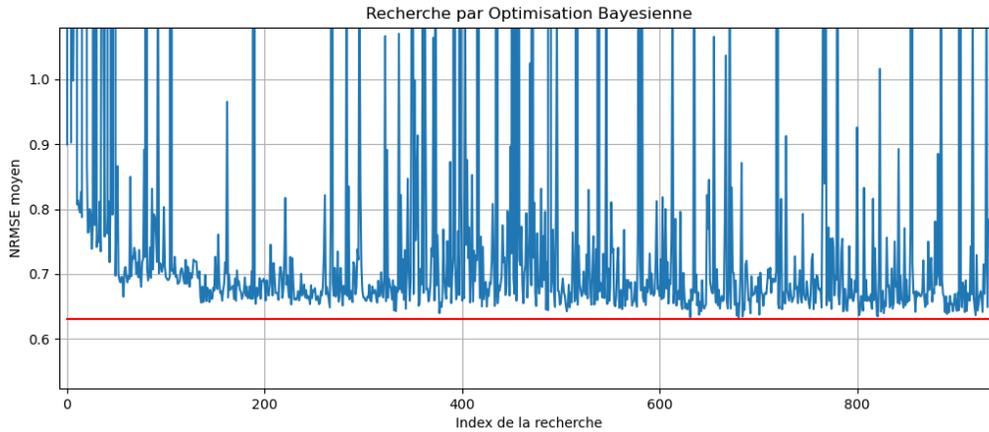


FIGURE 4.14 Progression de la recherche selon le nombre de réservoir évalué avec un entraînement hors-ligne et une régression simple.

Hyperparamètre	Borne minimale	Borne Maximale	Valeurs choisies
Taux de fuite	0.001	0.9	0.255
Rayon spectral	0.1	1.1	0.732
Parcimonie des poids	0	0.5	0.407
Échelle d'entrée	0	0.5	0.063
Échelle du biais	0	0.6	0.098
Parcimonie d'entrée	0	1	0.718
Échelle de rétroaction	0	1	0.016
Parcimonie de rétroaction	0	1	0.655

TABLEAU 4.4 Bornes de la recherche par optimisation bayésienne et hyperparamètres du réservoir trouvé par optimisation bayésienne et utilisé pour le contrôle du montage expérimental. Il est à noter que différents réservoirs ont été testés suite à des recherches avec différentes bornes. Les valeurs choisies présentées au tableau sont celles avec lesquelles le plus de résultats ont été obtenus.

4.2.1 Entraînement par lots

Lorsqu'un réservoir est sélectionné suite à un entraînement hors-ligne, une session d'entraînement par lots est débutée. Chaque session d'entraînement est composée d'un nombre fixe d'essais, où l'étage de sortie est mis à jour à la fin de chacun d'eux. Pour déterminer le nombre d'essais nécessaire, une session d'entraînement de 200 essais est effectuée. Tel qu'il est possible de le voir à la figure 4.15, une progression significative des performances, évaluées selon le NRMSE, est observable jusqu'à environ ≈ 50 essais. Pour permettre la reproduction de résultats, les poids de sortie sont également enregistrés à la fin de chacun des essais.

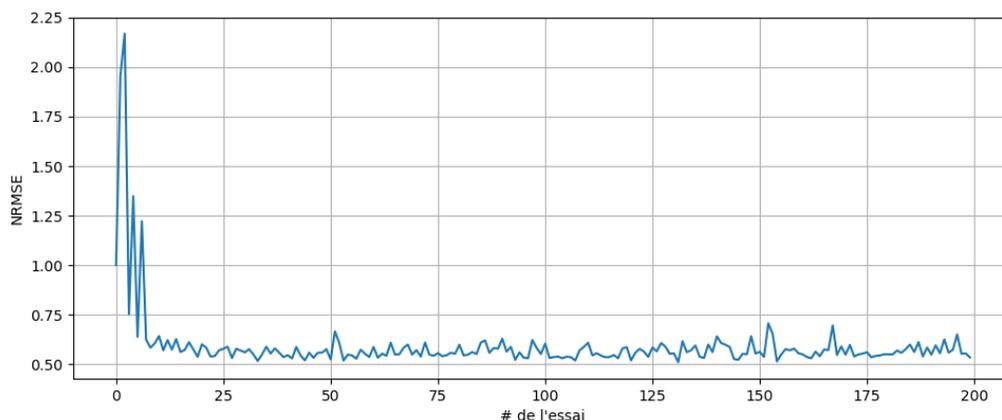


FIGURE 4.15 Historique de l'erreur entre la commande générée et la commande de référence des différents essais lors de l'entraînement. Chaque nouvel essai utilise une régression qui est calculée à partir des données récupérées aux essais précédents pour effectuer le contrôle.

Quelques essais sont nécessaires avant d'observer un apprentissage réel de la fonction de contrôle. Par exemple, lors du premier essai, la régression est nulle. Les commandes générées sont donc nulles pour l'entièreté de l'essai. Les essais qui suivent immédiatement ne font qu'injecter de l'énergie dans le pendule de Furuta. Les 10 premières trajectoires empruntées lors de l'entraînement par lot sont présentées à la figure 4.16.

L'injection d'énergie en continu est observable en notant le décalage des courbes dans l'axe des "y", qui représente la vitesse de rotation du pendule. Ce type d'erreur s'améliore, et le pendule apprend à partiellement amortir le système. Cependant, les commandes générées par l'ESN entraîné par lot ne maintiennent jamais le pendule à l'équilibre. Les 10 dernières trajectoires des essais d'entraînement sont présentées à la figure 4.17. Les trajectoires se ressemblent plus et il n'y a pas d'injection d'énergie en continu.

Comme les performances stagnaient à la fin des essais et que le pendule n'atteint pas son point d'équilibre, et ce malgré l'utilisation de l'entraînement par lots avec une rétroaction, une autre méthode d'entraînement sera explorée.

4.2.2 Entraînement en ligne et structure de régressions

L'entraînement en ligne consiste à recalculer les poids de la régression à chaque pas de temps. De cette façon, les erreurs ont tendance à être corrigées plus rapidement comme le signal de référence a une influence instantanée sur le réseau. Le désavantage de cette méthode est le fait qu'elle limite la taille du réseau ; l'inversion d'une matrice est requise à chaque pas de temps pour calculer le nouvel étage de sortie. Comme le temps attendu

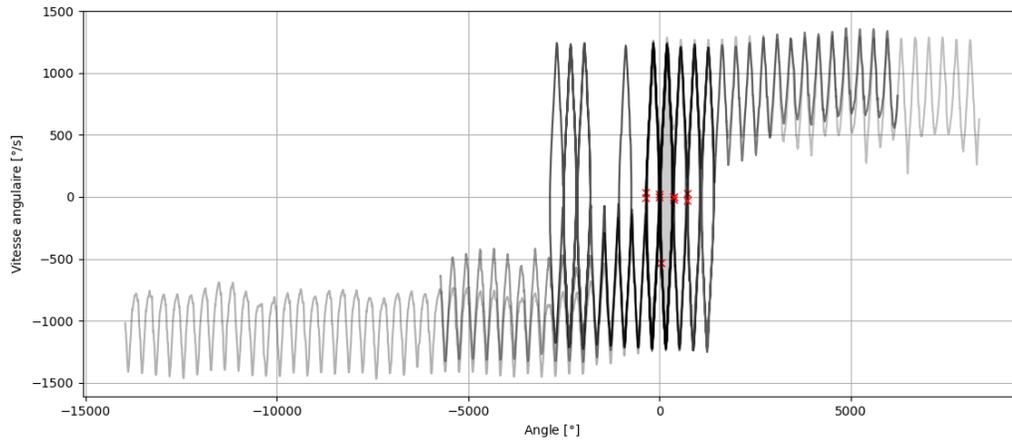


FIGURE 4.16 Comparaison des trajectoires empruntées lors des 10 premiers essais du réseau. Le début de chaque trajectoire est marqué d'un "X" rouge. Chaque nouvel essai utilise une régression qui est calculée à partir des données récupérées aux essais précédant. La commande de contrôle est ensuite calculée à chaque instant à partir de la nouvelle régression. L'entièreté des trajectoires est présentée pour illustrer la perte de contrôle du pendule lors des premiers essais.

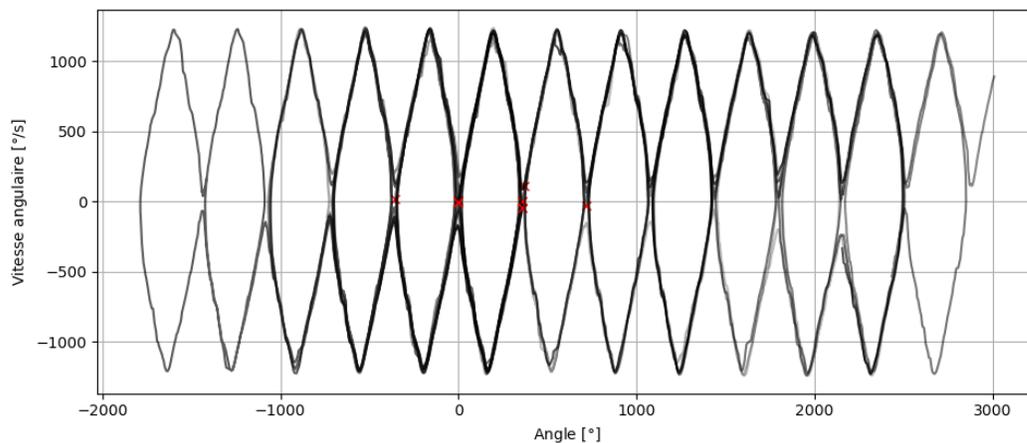


FIGURE 4.17 Trajectoires empruntées par le pendule de Furuta dans son espace de phase lors des 10 derniers essais de son entraînement.

entre chaque commande est de 5 ms, l'inversion matricielle doit être effectuée à l'intérieur de cet intervalle de temps. Puisque la dimension de cette matrice est le nombre de noeuds du réservoir, la taille du réservoir doit être limitée pour respecter cette contrainte. Une mesure du temps d'inversion des matrices en fonction de leur taille est donc effectuée sur l'ordinateur de contrôle du pendule pour trouver le nombre de noeuds maximal pour ce type d'entraînement. Cette courbe est illustrée à la figure 4.18.

Pour éviter d'être victime d'une baisse spontanée de puissance de calcul, le nombre de noeuds d'un réservoir est limité à 175, où l'inversion de la matrice prend environ 2.5 ms. Comme seule la méthode d'entraînement a changé (et qu'en réalité, l'analyse du temps d'inversion a été effectuée préalablement à l'entraînement par lots), seuls les meilleurs réservoirs résultants de la recherche hors-ligne présentés au tableau 4.4 sont utilisés. Dans le but de comparer avec les résultats, un premier entraînement en ligne d'un ESN avec rétroaction est effectué. Les courbes de progression des deux types d'entraînements sont présentées à la figure 4.19. On peut constater que l'entraînement par lot est moins rapide que l'entraînement en ligne, mais que les deux types d'entraînements convergent plus ou moins vers le même NRMSE.

À des fins de comparaison, les trajectoires empruntées par les 10 premiers et les 10 derniers essais sont présentées à la figure 4.20. Tel qu'il est possible de le voir, l'entraînement en ligne permet d'éviter complètement les essais où le pendule n'avait appris que l'injection d'énergie.

Comme l'entraînement par lot ou en ligne en forçant une rétroaction n'est pas suffisant pour obtenir un ESN qui arrive à atteindre l'équilibre, une méthode essayée en simulation est reprise ; les structures de régressions. Le processus d'entraînement en ligne est donc recommencé avec une sortie composée de deux régressions. La sélection de la régression utilisée pour la génération de commande est effectuée à partir de l'angle mesuré par l'encodeur du pendule. Lorsque celui-ci est sous 6 degrés, le passage d'une régression à la seconde est effectué et vice-versa.

Malgré la légère amélioration de NRMSE moyen (voir figure 4.21a) suite à l'ajout d'une seconde régression, le contrôleur ESN n'atteint pas l'état d'équilibre désiré et est moins performant que l'entraînement par lot selon cette métrique. Au niveau de la proportion de temps passé à l'équilibre, il n'y a pas de différence notable (voir la figure 4.21b). Pour améliorer ses performances, une régression additionnelle est ajoutée à la sortie du réservoir et les points de transition entre chaque régression sont modifiés. Les trois régressions auront les rôles suivants : la première régression sert à *ajouter* de l'énergie au système

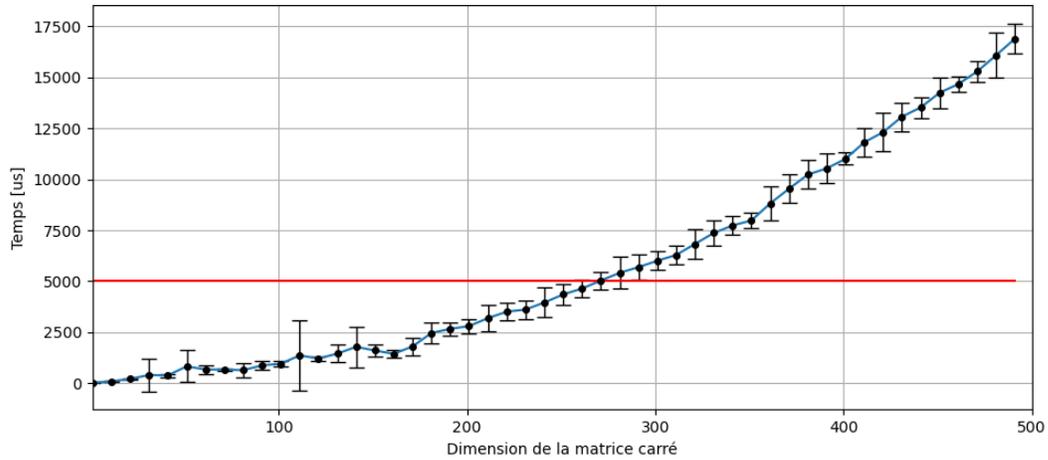


FIGURE 4.18 Temps d'inversion d'une matrice selon ses dimensions. 100 matrices aléatoires de chaque dimension ont été générées puis inversées. Chaque point sur le graphique correspond à la moyenne de ces 100 temps d'inversion et les barres d'erreurs correspondent à un écart-type (1σ). La ligne horizontale rouge représente le seuil de 5 [ms], qui ne doit en aucun cas être dépassé.

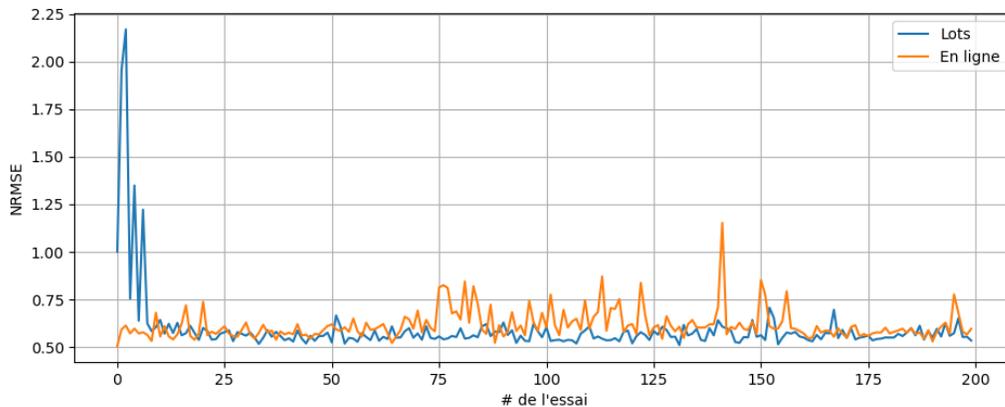


FIGURE 4.19 Comparaison de l'évolution du NRMSE pendant l'entraînement selon la méthode utilisée. Les deux méthodes convergent plus ou moins vers la même valeur, avec une légère différence. L'apprentissage initial est plus rapide avec l'entraînement en ligne, mais le NRMSE vers lequel ce type d'entraînement converge semble légèrement plus haut.

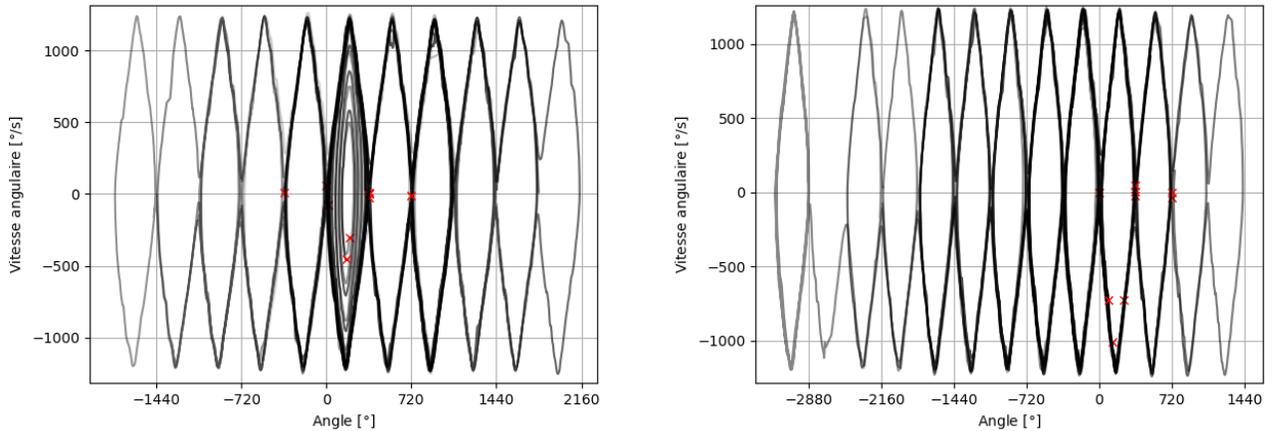
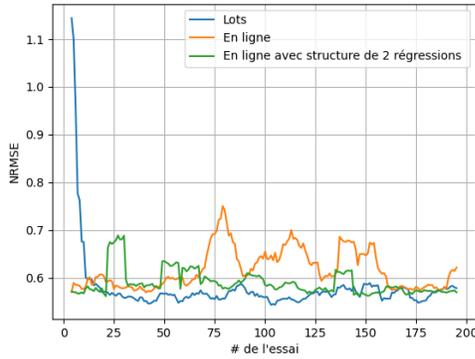


FIGURE 4.20 Exemples de trajectoires au début et à la toute fin de l'entraînement. Les "X" rouges représentent le début des trajectoires. À gauche - 10 premières trajectoires obtenues lors de l'entraînement en ligne. À droite - 10 dernières trajectoires obtenues lors de l'entraînement en ligne.

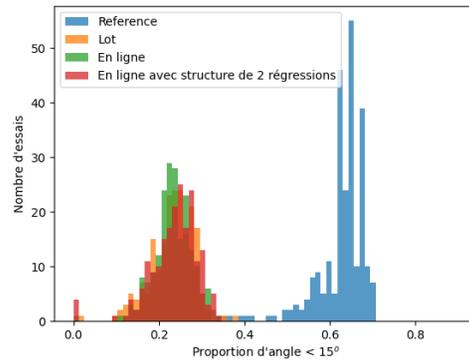
("pumping"), la seconde à *enlever* de l'énergie ("damping"), et la troisième à maintenir l'équilibre. Cet ordre de séparation est basé sur le fonctionnement du contrôleur de référence, qui comprend deux contrôleurs, Gordillo pour le balancement vers le haut et LQR pour l'équilibre. L'algorithme du contrôleur énergétique [4] est lui-même séparé en deux cas, qui correspondent au "pumping" et "damping" basées sur l'énergie du pendule. Au niveau des points de transition d'une régression à l'autre, ils sont spécifiés par :

$$\text{Régression} = \begin{cases} \text{Pumping} & \text{si } \theta_1 \geq 15^\circ \\ \text{Damping} & \text{sinon si } \theta_1 \leq 15^\circ \ \& \ \dot{\theta}_1 \geq \ \& \ \text{Régression} \neq \text{Equilibrium} \\ \text{Equilibrium} & \text{sinon si } \theta_1 \leq 4^\circ \end{cases}$$

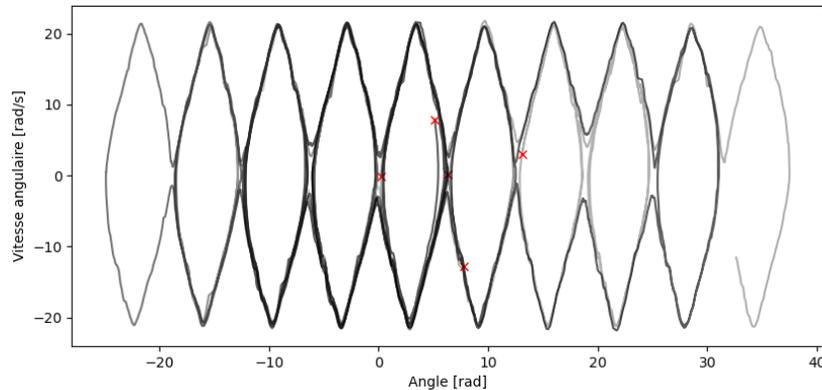
Toujours en conservant les valeurs d'hyperparamètres présentées au tableau 4.4, une nouvelle série d'entraînements est effectuée avec une structure de régression triple. Une comparaison des performances des différentes méthodes d'entraînement et des contrôleurs résultants est présentée à la figure 4.22b. Tel qu'il est possible de le voir, malgré l'amélioration obtenue lors de la composition des réservoirs, les performances du contrôleur de référence ne sont pas répliquées. Cependant, dans plusieurs cas l'ESN arrive de façon effective à balancer le pendule sur son axe. Ce type de trajectoire est présenté à la figure 4.22d.



(a) Comparaison des NRMSE en fonction de la progression de l'entraînement. Un filtre passe-bas a été appliqué (moyenne mobile à 9 termes) pour aider à la comparaison du progrès entre les différentes méthodes.



(b) Comparaison des distribution de la proportion de temps que le pendule a passé à un angle plus faible que 15° (en fonction de la progression de l'entraînement).

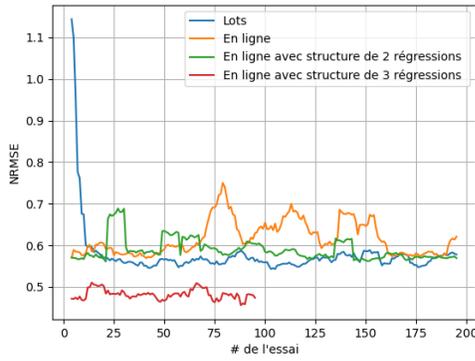


(c) Exemples de trajectoires empruntées par le pendule de Furuta avec un contrôleur RC avec rétroaction, entraîné en ligne et ayant une structure de régression double.

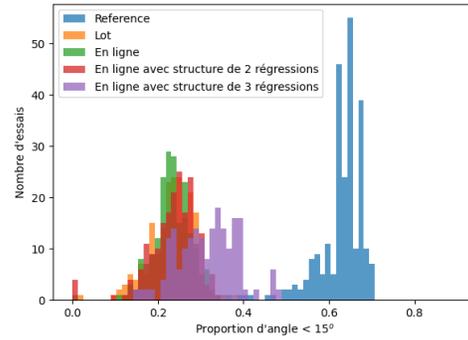
FIGURE 4.21 Performances et exemples de trajectoires du contrôleur avec une structure à double régression.

L'équilibre est atteint par le pendule de Furuta pour des durées de temps prolongées lors de certains essais. Par exemple, l'un deux a passé 48.3 % du temps à 15° du point d'équilibre. Cependant, la stabilité du contrôleur laisse à désirer. Effectivement, un ESN avec rétroaction et une structure à trois régressions passe en moyenne (31.5 ± 6.8) % à 15° de leur point d'équilibre, ce qui est loin du (62.7 ± 5.8) % du contrôleur de référence. L'instabilité sera discutée plus en détail dans la discussion (voir section 6.3) de ce mémoire.

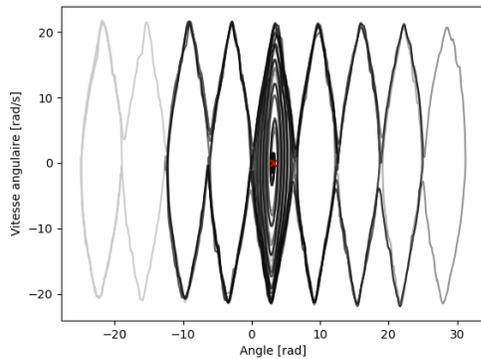
À ce point-ci, dû aux contraintes de temps du projet, il a été choisi de passer directement à l'utilisation de réservoirs physiques.



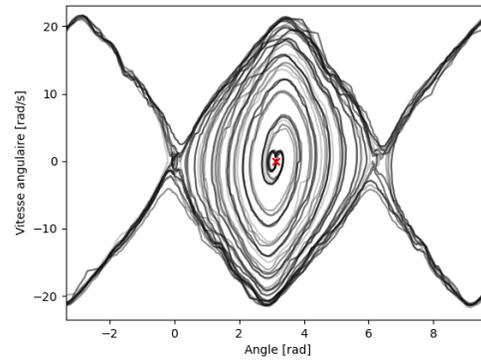
(a) Comparaison des NRMSE en fonction de la progression de l'entraînement pour différentes méthodes d'entraînement. Un filtre passe-bas a été appliqué (moyenne mobile à 9 termes) pour aider à la comparaison du progrès entre les différentes méthodes.



(b) Comparaison des distributions de la proportion de temps que le pendule a passé à un angle plus faible que 15° en fonction de la progression de l'entraînement.



(c) Exemple de trajectoires empruntées par le pendule de Furuta avec un contrôleur RC entraîné en ligne ayant une structure de régression triple.



(d) Agrandissement du diagramme de phase présenté en 4.22c. Beaucoup des trajectoires oscillent autour des points $(0,0)$ et $(2\pi,0)$, indiquant qu'un équilibre est atteint et maintenu pour de courtes durées.

FIGURE 4.22 Performances et exemples de trajectoires d'un contrôleur ESN avec rétroaction et une structure à trois régressions.

CHAPITRE 5

Physical reservoir computing (PRC)

Dans ce chapitre, les requis pour le passage à l'utilisation du capteur neuromorphique ainsi que les résultats obtenus avec celui-ci sont documentés.

5.1 Intégration

Le code développé précédemment au laboratoire ne supportait pas l'entraînement en ligne ainsi que des mesures plus rapides que 100 [us] par noeud. Ce second facteur en particulier limitait le nombre de noeuds d'un réservoir à 50, dû aux contraintes de temps minimal entre deux commandes (5000 us). Pour s'affranchir de ces deux limitations, une nouvelle librairie d'interfaçage avec le réservoir physique nommée "MultiDAQ" a été développée. Elle comporte deux avantages : des temps de mesure aussi rapide que 25 us peuvent être atteints et un signal d'entrée peut être reproduit en temps réel. Cette amélioration provient de la distribution des tâches d'interfaçage avec la carte NI en parallèle sur plusieurs coeurs de l'ordinateur. Un schéma global du fonctionnement du code et de la distribution des tâches est présenté à la figure 5.1.

Par la suite, il était désirable de coupler le montage du capteur neuromorphique au pendule de Furuta pour permettre de faire des tests ainsi que des entraînements par lots ou bien en ligne. Une chaîne de communication a été établie entre les deux montages expérimentaux. Le temps de communication aller-retour devait être moins de 5 ms pour respecter les contraintes de temps entre chaque commande générée. Pour ce faire, une architecture de réception et d'envoi des données en parallèle a été choisie sur chacun des ordinateurs. Un schéma global donnant une idée générale de la structure des programmes de chaque montage est présenté à la figure 5.2.

Pour accélérer et rendre plus fiable les communications, les deux ordinateurs sont connectés sur le réseau filaire du 3IT (3 ms aller-retour). Alternativement, il aurait été possible d'utiliser le wi-fi local. Des tests préliminaires ont toutefois mis en lumière que la communication wi-fi était moins fiable et plus lente (5 ms aller-retour).

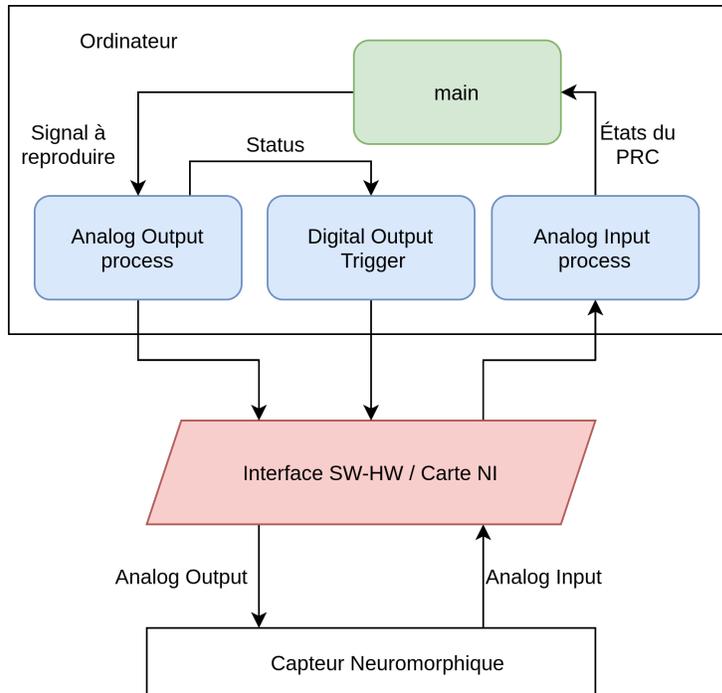


FIGURE 5.1 Schéma de bloc du code utilisé pour interfacer avec le capteur neuromorphique. Les blocs de gestion du code sont encadrés en vert. Les sous-programmes sont identifiés en bleu et les modules physiques d'interfaçage avec le capteur neuromorphique sont en rouge.

5.2 Réservoir physique

Les réservoirs physiques utilisés sont des réseaux d'oscillateurs de Duffing microfabriqués, décrits plus en détail à la section 2.3.3. Pour simplifier la microfabrication, le concept de réservoir à rétroaction retardée est utilisé (voir section 2.2.1). Cette méthode permet d'utiliser un seul oscillateur de Duffing pour générer un réseau entier. Le type de réservoir utilisé est décrit plus en détail dans l'article écrit par Barazani et al.[3]. Puisque le type des réservoirs est différent des ESNs utilisés précédemment, les hyperparamètres le sont aussi. Ces nouveaux hyperparamètres à optimiser sont présentés au tableau 5.1 avec une courte description.

Il est à noter que les capteurs neuromorphiques étudiés dans ce mémoire ne sont sensibles qu'à l'accélération dans un seul axe. Un paramètre est donc ajouté à la recherche pour optimiser l'orientation du capteur neuromorphique. Ce paramètre est nommé "l'angle" et correspond à l'orientation du capteur neuromorphique sur le plan X-Y du pendule, où un angle de 0° représente l'orientation où les déplacements de l'accéléromètre se feraient selon l'axe des X. Pour reproduire l'effet de l'orientation du CN sur le montage expérimental,

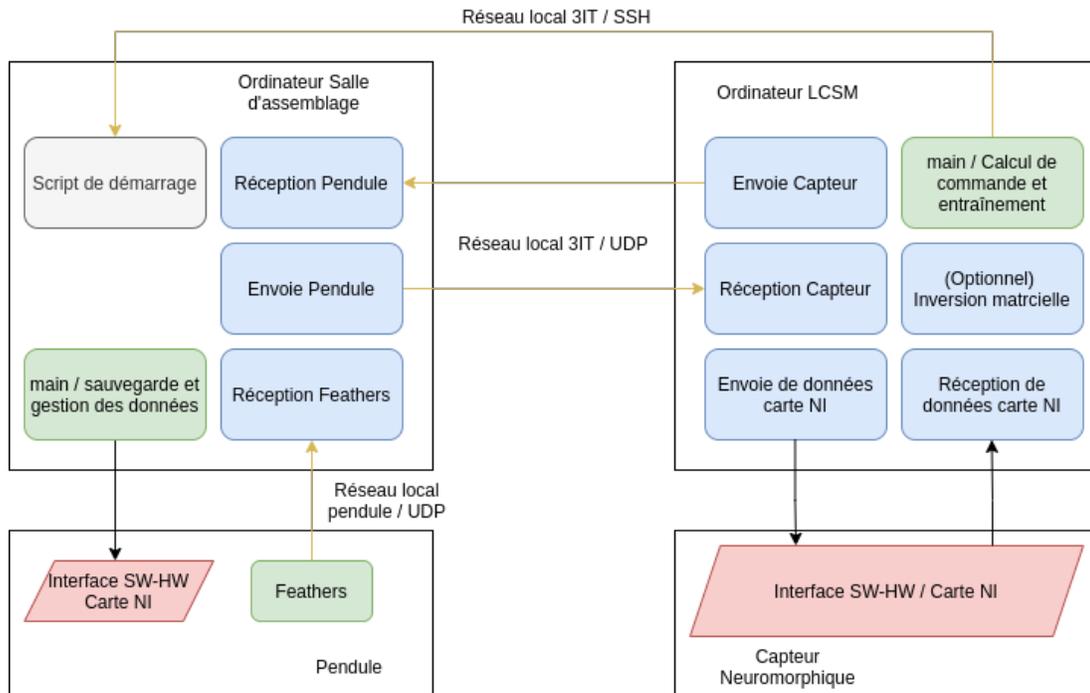


FIGURE 5.2 Schéma du code qui permet de lier les montages du LCSM au pendule de Furuta en temps réel. Les blocs de gestion de code sont en vert. Ceux-ci servent à démarrer les sous-programmes et prédéfinir les blocs mémoires à partager. Les sous-programmes qui accomplissent des tâches spécifiques sont identifiés en bleu puis les composants physiques d'interfaçage par lecture/écriture de voltage sont identifiées en rouge. Les communications sont effectuées via UDP par deux des sous-programmes sur chaque montage expérimental. L'ordinateur du pendule est asservi à l'ordinateur du LCSM via un script powershell, qui est initialement démarré par une commande SSH.

Hyperparamètre	Description
Fréquence de Pompe	Fréquence du signal d'alimentation de l'oscillateur
Amplitude de Pompe	Amplitude du signal d'alimentation de l'oscillateur
Masque minimal	Modulation minimale du signal d'alimentation
Masque maximal	Modulation maximale du signal d'alimentation
α	Taux de réinjection de l'ancienne valeur de chaque noeud
Taux de fuite	Taux de fuite des noeuds du réseau
Échelle d'entrée	Échelle multiplicative de l'entrée
Angle	Orientation du PRC

TABLEAU 5.1 Noms des hyperparamètres et sommaire description de chacun d'eux.

une projection des accélérations dans l'axe choisi est effectuée préalablement à la recherche de réservoir et à l'entraînement.

Comme c'est un nouveau type de réservoir, il est nécessaire d'effectuer une recherche hors-ligne pour sélectionner des hyperparamètres optimaux pour la tâche. Par contre, dû aux contraintes de temps entre chaque commande, certains d'entre eux sont fixés. Le nombre de noeud est fixé à $N = 100$ et le temps de mesure associé à chacun d'eux est fixé en fonction de la valeur de la fréquence de pompe au multiple entier le plus près de $\theta = 50$ us ($\theta = \text{int}(\frac{1}{\text{Fréquence de pompe}} * \frac{1}{50\mu\text{s}})$). Une échelle de temps plus petite que celle utilisée par Barazani et al. [3] est forcée pour générer des commandes à l'intérieur d'un délai de 5 ms avec un plus grand nombre de noeud. Ce temps de mesure est aussi plus court que le temps caractéristique de notre oscillateur, donné par

$$T = 2Q/\omega_0 \approx 130 \text{ us} \quad (5.1)$$

Ce critère permet de vérifier qu'aucun des noeuds ne sature en régime permanent [1].

5.3 Processeur Neuromorphique

Il est possible d'utiliser le capteur neuromorphique purement comme processeur neuromorphique. Tel que présenté par Dion et al. [2], l'équation 2.25 d'alimentation de l'oscillateur de Duffing peut être remplacée par l'expression suivante,

$$V(t) = V_0[m(t)s(t) + \alpha(t)((1 - L_R)X_{t-1} + L_RX_t)] \cos(2\pi f_d t), \quad (5.2)$$

où le seul changement par rapport à l'équation du capteur neuromorphique (voir équation 2.25) est l'ajout de $s(t)$. Ce terme correspond au signal d'entrée qui change avec le temps. La multiplication du signal d'entrée avec le masque module l'alimentation de la poutre dans le temps. De façon analogue au déplacement de l'accéléromètre du capteur neuromorphique, cette modulation change le signal d'alimentation et fait varier l'amplitude d'oscillation de la poutre non linéairement. Dans ce cas-ci, le signal d'entrée sera un signal d'accélération projeté dans une direction puis normalisé.

La progression de la recherche de réservoir est présentée à la figure 5.3. Les hyperparamètres étaient choisis par optimisation Bayésienne, de la même façon qu'à la section 4.1.1. Les paramètres sélectionnés sont ensuite évalués. Lors du processus d'évaluation, une régression simple est utilisée pour générer un signal de contrôle à partir des noeuds du réservoir. L'entraînement et l'évaluation sont effectués à partir de 8 signaux d'accélération

pré-enregistrés ; les 5 premiers sont utilisés pour l’entraînement et les 3 derniers pour l’évaluation. Ce nombre est petit, mais il permet d’évaluer rapidement différentes combinaisons d’hyperparamètres. Le NRMSE présenté à la figure 5.3 pour chaque essai est la moyenne de ces 3 NRMSE. En se basant sur ces résultats, les paramètres choisis pour les tests sont présentés au tableau 5.2.

Hyperparamètre	Valeur
Fréquence de Pompe	250 300 Hz
Amplitude de Pompe	8.9947 V
Masque Minimal	1.3244
Masque Maximal	2.8282
α	0.8991
Taux de fuite	0.00571
Échelle d’entrée	10.
Angle	0.07012 °
θ	4.7942e-05 s

TABLEAU 5.2 Hyperparamètres du meilleur essai de la recherche par optimisation bayésienne. Le NRMSE obtenu entre la commande de référence et la commande générée pour cet essai est de 0.6781 .

La structure de régression la plus performante trouvée avec les ESNs est reprise avec un entraînement par lots pour être testée avec le processeur neuromorphique. Le code d’intégration des deux montages présentés à la section 5.1 est utilisé pour synchroniser les deux bancs de tests ensemble et communiquer les commandes et les accélérations en temps réel. Pour accélérer l’entraînement des régressions près des points d’équilibre, un temps de transition est choisi selon une distribution uniforme entre 0.5 et 10 secondes. Lors de chaque essai d’entraînement, de 0.5 secondes au temps de transition, le contrôleur de référence génère la commande. Lorsque le temps de transition est atteint, le contrôle est laissé au processeur neuromorphique. La progression de l’entraînement par lots pour chaque essai est présentée à la figure 5.4.

La structure de régression ayant obtenu les meilleurs résultats est celle obtenue à l’essai # 48, avec une proportion de temps passé sous un angle de 15° de 29.10 % et un NRMSE de 0.80827. Cette régression et les hyperparamètres associés sont repris puis testés de façon répétée pour obtenir une distribution de performance du contrôleur. La distribution de proportion de temps passé sous 15° de ce contrôleur est présentée à la figure 5.5. Un exemple de trajectoire empruntée lorsque le moteur est asservi au processeur neuromor-

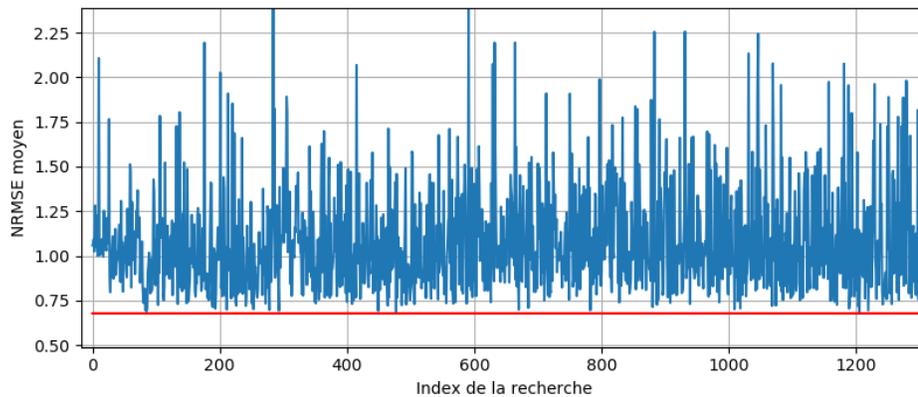


FIGURE 5.3 NRMSE de différents réservoirs testés suite à leur entraînement hors-ligne. L'entraînement est effectué sur 5 essais préenregistrés où le contrôleur de référence bascule le pendule vers le point d'équilibre. La mesure de NRMSE est effectuée sur 3 autres essais préenregistrés où les signaux du réservoir sont combinés avec la régression linéaire calculée à l'entraînement puis comparés à la commande de référence de chaque cas.

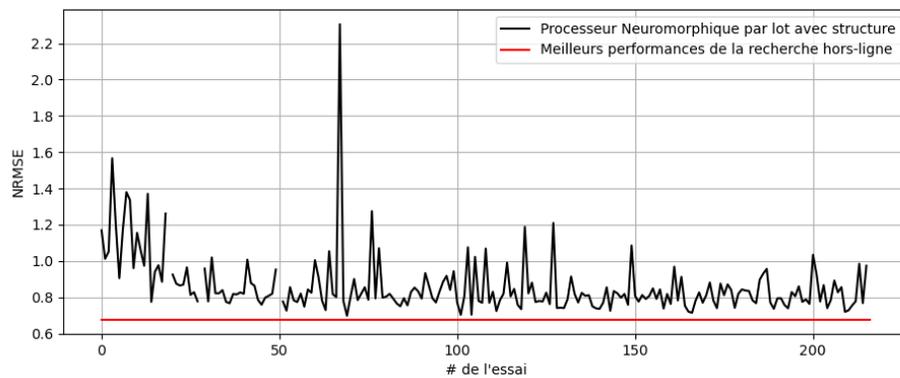


FIGURE 5.4 Progression de l'entraînement par lots en utilisant un processeur neuromorphique intégré avec un pendule de Furuta. Une structure à trois régressions est utilisée pour générer la commande transmise à l'actuateur. La commande de référence est générée par le contrôleur par modulation d'énergie lors du balancement vers le haut, puis par un contrôleur LQR pour maintenir l'équilibre. Le NRMSE minimal obtenu lors d'un essai expérimental est de 0.6978, la ligne rouge correspond à ce minimum. Les coupures de la courbe du NRMSE sont dues à des ratés dans la communication entre les montages expérimentaux. Ces ratés ne sont pas utilisés lors de l'entraînement.

phique avec ce contrôleur est présenté à la figure 5.6 et la commande associée est présentée à la figure 5.7.

Une amélioration significative était attendue lors du test d'une structure de régression entraînée par rapport aux distributions de performances obtenues lors de l'entraînement par lots. Puisque les régressions sont déjà entraînées, le contrôleur ne devrait pas commettre les erreurs de départ. Par exemple, lors des premiers essais de l'entraînement par lots, la structure de régressions n'est pas entraînée, elle génère donc des commandes nulles ou hors contexte. Or, tel qu'il est possible de le constater à la figure 5.5, les performances du contrôleur sont variables ; la distribution obtenue lorsque le processeur est exploité est seulement légèrement plus performante que celle de l'entraînement, sans qu'il y ait de différences notables. Malgré ces performances insuffisantes, un passage est effectué vers l'utilisation du capteur neuromorphique. L'étalement des performances ainsi que la reproductibilité des résultats seront discutés plus en détail dans la section 6.

5.4 Capteur Neuromorphique

La différence entre le processeur neuromorphique et le capteur neuromorphique est la façon de transmettre l'information. Dans le cas du capteur neuromorphique, plutôt que de multiplier l'accélération au masque comme lors de la rétroaction dans le cas du processeur neuromorphique, il doit subir l'accélération. Pour ce faire, le signal d'accélération projeté dans un axe est envoyé à un pot vibrant sur lequel le capteur neuromorphique est installé. De la même façon que pour le processeur neuromorphique, les accélérations sont mises à jour en temps réel pour simuler le montage du pendule lors des entraînements par lot.

Encore une fois, une recherche par optimisation Bayésienne pour minimiser le NRMSE entre la commande générée par le capteur neuromorphique et la commande de référence est effectuée pour trouver une bonne combinaison d'hyperparamètres. Pour chaque nouvelle combinaison d'hyperparamètres, le capteur neuromorphique subit les accélérations de 8 essais de 25 secondes préenregistrés où le contrôleur de référence était aux commandes. Les 5 premiers servent à l'entraînement et les 3 derniers à évaluer le réseau. Les hyperparamètres de masque et d'échelle d'entrée prennent un nouveau sens par rapport au processeur neuromorphique. L'échelle d'entrée est maintenant un facteur d'amplification pour l'actuateur linéaire. Comme le masque n'est maintenant plus multiplié par le signal d'entrée, celui-ci peut être interprété comme une sélection du point d'opération de l'oscillateur de Duffing pour chacun des noeuds. La progression de cette recherche hors-ligne est présentée à la figure 5.8.

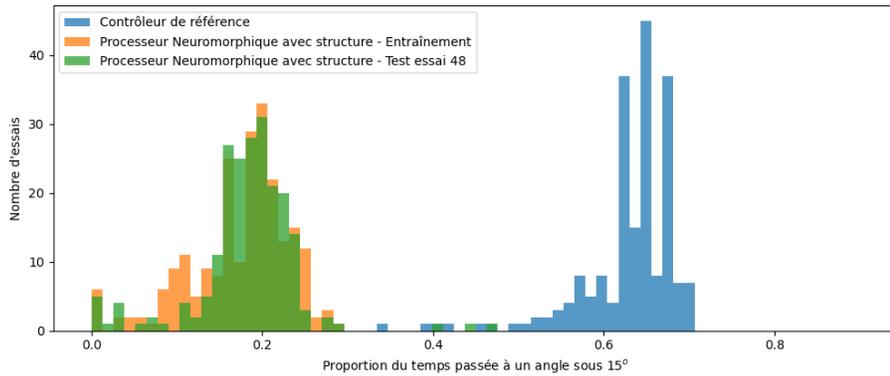


FIGURE 5.5 Comparaison des performances obtenues dans différentes situations. L'histogramme en orange représente les performances obtenues lors de l'entraînement par lots avec des hyperparamètres fixes, où entre chaque essai la régression était mise à jour. La distribution en vert représente les performances de la structure de régression utilisée à l'essai #48 de l'entraînement par lots. Elle a été sélectionnée puisque c'est la structure de régression avec laquelle les meilleures performances (proportion de 29.10 % du temps passé à 15° du point d'équilibre) avaient été obtenues.

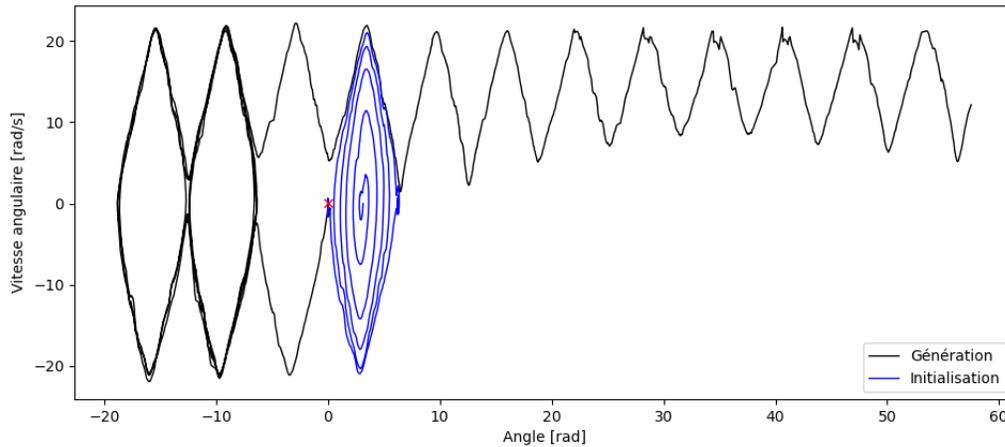


FIGURE 5.6 Trajectoire emprunté par le pendule de Furuta dans son espace de phase lorsque le contrôle est effectué à l'aide d'un capteur neuromorphique. L'essai dont la trajectoire est illustré passe 26% de son temps sous un angle de 15°. Le contrôleur de référence est initialement utilisé. Le contrôle est ensuite repris au point de transition, indiqué par un "X" rouge, par le processeur neuromorphique et sa structure de contrôle.

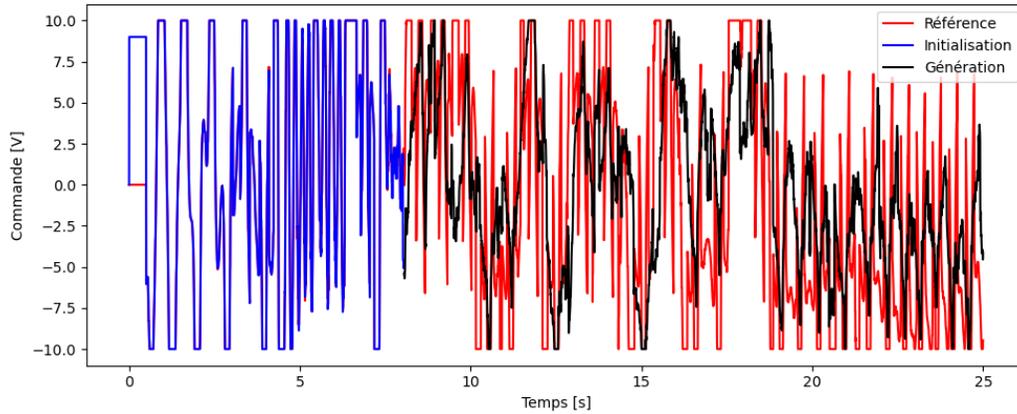


FIGURE 5.7 Commandes associées à la trajectoire présentée à la figure 5.6. La commande "Référence" est celle générée par le contrôleur standard, la commande "Génération" est celle générée par le processeur neuromorphique et sa structure de régression. La commande "Initialisation" est la commande utilisée pour initialiser le pendule de Furuta dans un état aléatoire. Il y a des différences entre la commande "Référence" et la commande "Initialisation" puisque la première est générée localement et que la seconde est générée au LCSM. Il y a donc de légers délais entre les données utilisées par le contrôleur dans les deux cas. La seconde différence est la première demi-seconde, qui est une application forcée de moment par le moteur pour démarrer le balancement. Durant les essais, les commandes "Initialisation" puis "Génération" sont appliquées sur le pendule.

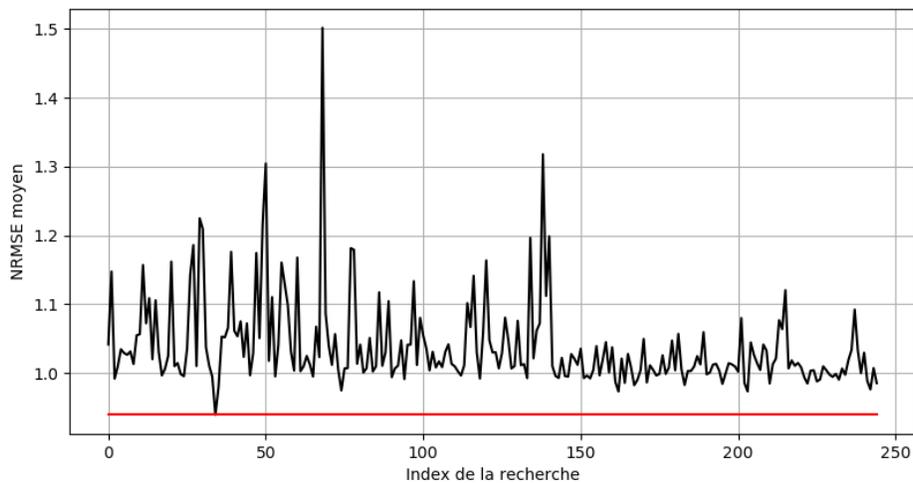


FIGURE 5.8 Progression de la recherche de réservoir. Le réservoir ayant démontré les meilleures performances en termes de NRMSE est sélectionné pour l'entraînement par lots.

Par la suite, la même structure de contrôle à 3 régressions développée avec les ESN (voir section 4.2.2) est réutilisée pour effectuer un entraînement par lots. La progression de l’entraînement est présentée à la figure 5.9. De la même façon que le processeur neuromorphique, la structure de régressions ayant maintenu le pendule sous un angle de 15° pendant le plus de temps lors de l’entraînement par lots sont récupérées. Le contrôleur ainsi obtenu est ensuite évalué à répétition pour déterminer sa distribution de performance. Pour éviter des changements de paramètres dus à des conditions variables du MEMS ou du circuit de mesure de la poutre, les étapes de recherche, d’entraînement par lots et d’évaluation sont effectuées à la chaîne. Finalement, la distribution de performance de l’exploitation du capteur neuromorphique est présentée à la figure 5.10. À des fins de comparaison, la distribution de performance de l’entraînement par lots et la distribution d’exploitation pour le processeur neuromorphique sont aussi superposées.

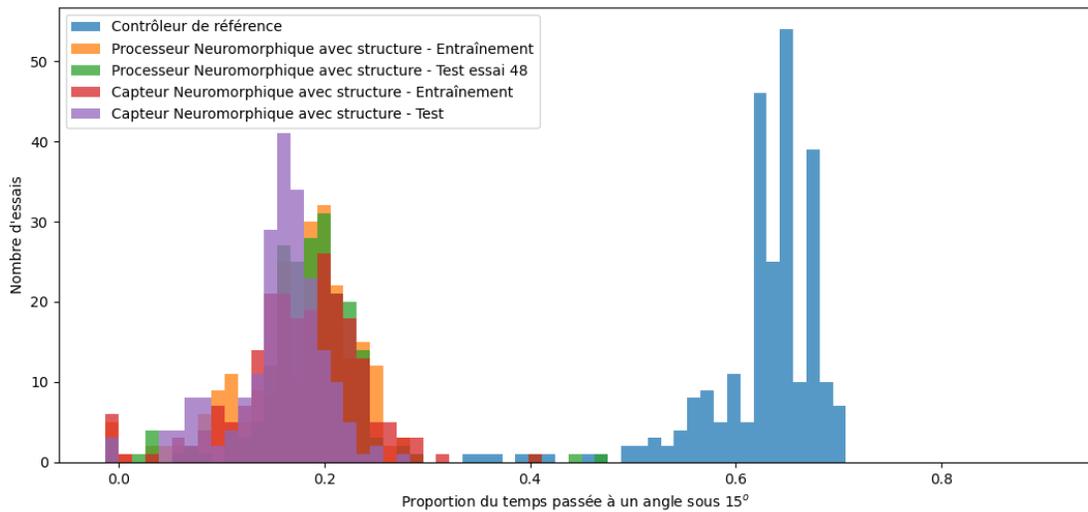


FIGURE 5.10 Comparaison de la distribution de performances de différents contrôleurs.

De la même façon que pour le processeur neuromorphique, une amélioration significative de la proportion de temps passé à l’équilibre était attendue entre la distribution de l’entraînement et la distribution de l’exploitation du capteur neuromorphique. En comparant les performances, la distribution d’exploitation est moins performante (en moyenne) que la distribution d’entraînement.

Une comparaison numérique de tous les résultats obtenus aux différents stages de ce projet est présentée au tableau 5.3. On y voit que les résultats obtenus à l’aide du réservoir physique sont similaires aux résultats obtenus en simulation avec un entraînement hors-

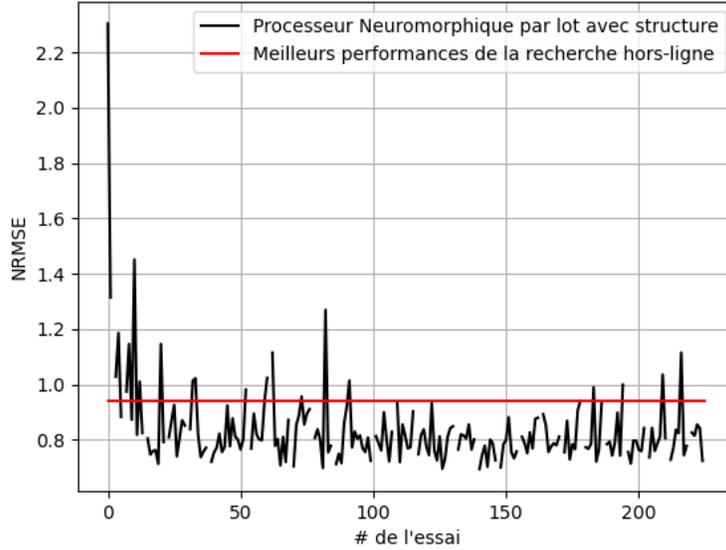


FIGURE 5.9 Progression de l'entraînement par lots du capteur neuromorphique. La ligne en rouge représente le NRMSE minimal obtenu lors de la recherche de réservoir par entraînement hors-ligne.

ligne. Des hypothèses sur les raisons de ces performances sont élaborées à la section 6, ainsi que des améliorations ou modifications possibles qui pourraient mener à une amélioration des performances.

	Type d'entraînement		Structure de régression			Réservoir		Performance - Proportion		
	Hors-ligne	Par lot	En ligne	Simple	Double	Triple	O.B.	Rétroaction	Moyenne	Maximale
Référence - Simulation									0.852± 0.054	0.949
Simulation (voir section 4.1)	X			X			X		0.184± 0.026	0.296
		X		X			X		0.508±0.076	0.643
		X			X		X		0.591± 0.056	0.853
		X		X			X	X	0.812± 0.064	0.899
Référence - Expérimentale								0.627± 0.058	0.701	
ESN expérimental (voir section 4.2)	X			X			X	X	0.231± 0.049	0.385
	X				X		X	X	0.232± 0.045	0.344
			X	X			X	X	0.235± 0.057	0.341
			X			X	X	X	0.315± 0.068	0.483
Processeur Neuromorphique (voir section 5.3)	X					X	X		0.174± 0.057	0.291
	X					X	X		0.184± 0.058	0.464
Capteur Neuromorphique (voir section 5.4)	X					X	X		0.175± 0.058	0.400
	X					X	X		0.154± 0.046	0.273

TABLEAU 5.3 Comparaison numérique de tous les résultats obtenus. Les spécificités du type d'entraînement, de la structure de régression utilisée et de la recherche sont affichées. Il est possible d'y voir rapidement quels changements ont permis d'augmenter significativement les performances dans chaque cas. L'incertitude présentée sur la moyenne est l'écart type de la distribution. L'abréviation «O.B.» représente dans ce cas-ci «Optimisation Bayésienne». Il est à noter que seuls les réservoirs les plus performants de chaque recherche sont considérés lors des tests.

CHAPITRE 6

Discussion

Dans cette section, une revue critique des résultats présentés dans le mémoire sera effectuée. Dans un premier temps, la méthodologie appliquée sera revue, puis les résultats, pour finir avec un saut vers la stabilité des contrôleurs obtenus.

6.1 Retour sur l'approche générale

L'entraînement d'un contrôleur est effectué en trois étapes indépendamment de la variante du problème à résoudre, que ce soit en simulation, sur le montage, avec un ESN ou bien l'oscillateur de Duffing. Celles-ci sont, dans l'ordre : la recherche de réservoir, la création d'une structure de régression et l'entraînement final.

6.1.1 Recherche de réservoir

La recherche de réservoir est à la base du reservoir computing. Dès le début du projet, il a été posé que la métrique à minimiser serait le NRMSE entre la commande de référence et une commande générée par le réseau suite à un entraînement hors-ligne. L'avantage de la méthode qui a été développée est d'évaluer une grande quantité de réservoirs rapidement. Cependant, en rétrospective, ce ne sont pas toujours les réservoirs qui avaient le plus petit NRMSE en entraînement hors-ligne qui ont le mieux performé. Par exemple, dans le cadre des simulations, le réservoir qui arrive le mieux à remplir la tâche est celui présenté au tableau 4.3. Il remplit la tâche avec une régression simple et l'ajout d'une rétroaction de sa sortie. Pourtant, le NRMSE qu'il a obtenu en recherche hors-ligne est de 0.93 par rapport à des NRMSE de l'ordre de 0.12 pour ses contreparties utilisées lors des essais précédents en simulation, qui remplissent moins bien la tâche.

Il est donc possible que le NRMSE ne soit pas le meilleur critère à minimiser pour la recherche de réservoir. Comme introduit à la section 2.2.3, des méthodes pour mesurer les propriétés des réservoirs, telles que le Kernel Rank, le Generalisation Rank, la Memory Capacity et l'IPC sont documentés [31][32]. Ces mesures pourraient représenter des critères alternatifs de sélection de réservoir. Par exemple, comme démontré à la section 4.2.2 pour le pendule de Furuta, un ESN avait des performances acceptables. En mesurant les propriétés du réservoir utilisé selon ces quatre métriques, il pourrait être possible de quantifier les requis d'un réservoir pour arriver à résoudre le problème. Par la suite, ces requis pourraient

être utilisés comme critères de sélections, qui sont eux indépendants du type de réservoir utilisé.

6.1.2 Méthode d'entraînement

Dans ce travail, trois méthodes d'entraînement distinctes ont été testées, l'entraînement hors-ligne, par lots, puis en ligne. Ces trois méthodes utilisent une régression de Tikhonov pour calculer les poids résultants. Les principales différences entre chaque méthode sont le moment du calcul des nouveaux poids et le moment d'application de la nouvelle régression. Pour les entraînements par lots et en ligne, ce moment est respectivement à la fin d'un essai ou à chaque commande générée pendant un essai. Dans le cas de l'entraînement hors ligne, le calcul des poids est effectué à la fin d'un entraînement complet où le signal de référence est appliqué sur le système. Le gain obtenu en utilisant un entraînement par lot ou en ligne est que le réseau s'adapte aux erreurs qu'il commet puisque la commande qu'il génère influence le système dynamique duquel il prend ses entrées. Il apprend donc les corrections à effectuer, en plus du signal de référence.

Il existe cependant une autre méthode qui est spécifiquement adaptée à l'entraînement de réseaux imbriqués dans des boucles de rétroaction nommée l'apprentissage FORCE [42]. Selon les auteurs, leur approche permet d'éviter qu'une commande générée erronée influence trop la dynamique des noeuds, ce qui changerait la régression résultante. Il serait intéressant de reprendre notre problématique avec l'approche FORCE pour vérifier si des améliorations des performances sont atteignables.

6.2 Contrôle de systèmes sous-actués

Le défi du projet avait deux aspects principaux. Premièrement, il fallait créer un contrôleur de systèmes sous-actués avec un RC. Cette notion a déjà été démontrée par T. Waegeman dans sa thèse [43]. Le second aspect, qui permet à ce travail de se démarquer, est le type de signaux utilisés pour calculer la loi de contrôle. Contrairement aux contrôleurs standard du problème de pendule inverse, les accélérations prises dans le référentiel du pendule ont été utilisées comme entrée. Cette contrainte est appliquée dans le but de faire la transition au capteur neuromorphique. Les étapes qui précèdent cette transition sont des outils pour apprendre les conditions dans lesquels l'utilisation du capteur neuromorphique serait possible.

6.2.1 Contrôle par ESN

Le contrôle par ESN du pendule inverse utilisant les accélérations comme entrées a tout d'abord été un succès en simulation ; une distribution de proportion avec une moyenne de $(81.1 \pm 6.4) \%$ est atteinte, par rapport à $(85.2 \pm 5.4) \%$ pour le contrôleur de référence

selon la même métrique (section 4.1). Cette réussite est cependant sous certaines conditions. Tout d'abord, le problème de contrôle est légèrement différent comme le problème simulé est celui du chariot-tige. Deuxièmement, deux accélérations sont utilisées comme entrées plutôt qu'une seule dans le cas du capteur neuromorphique. Une rétroaction de la commande générée à la dernière itération est nécessaire et le contrôleur doit également être entraîné par lots.

Suite à ce premier succès, un ESN a été intégré au montage expérimental. La méthode d'entraînement et la structure les plus performantes en simulation ont servi de point de départ. La distribution de proportion de ce contrôleur a une moyenne de 0.231 ± 0.049 (section 4.2). Quelques itérations de contrôleurs sont effectuées jusqu'à ce qu'un contrôleur ESN arrive à remplir la tâche dans certains cas. Un exemple d'une telle trajectoire est présenté aux figures 6.1 et 6.2, où le contrôleur a passé 48.2 % du temps total de l'essai sous un angle de 15° . La moyenne de la distribution de performance du même contrôleur est de 0.315 ± 0.069 par rapport à 0.626 ± 0.058 pour le contrôleur de référence. Pour obtenir cette distribution, le contrôleur emploie une structure à 3 régressions ainsi qu'une rétroaction de la commande générée à l'itération précédente, en plus d'être entraîné en ligne.

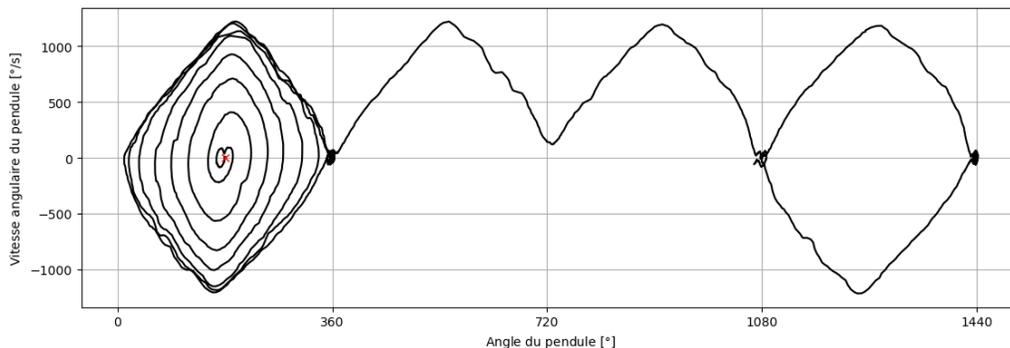


FIGURE 6.1 Diagramme de phase du meilleur essai obtenu à l'aide d'un contrôleur ESN. Ce contrôleur utilise les accélérations du pendule dans 3 axes orthogonaux et un étage de sortie à 3 régressions.

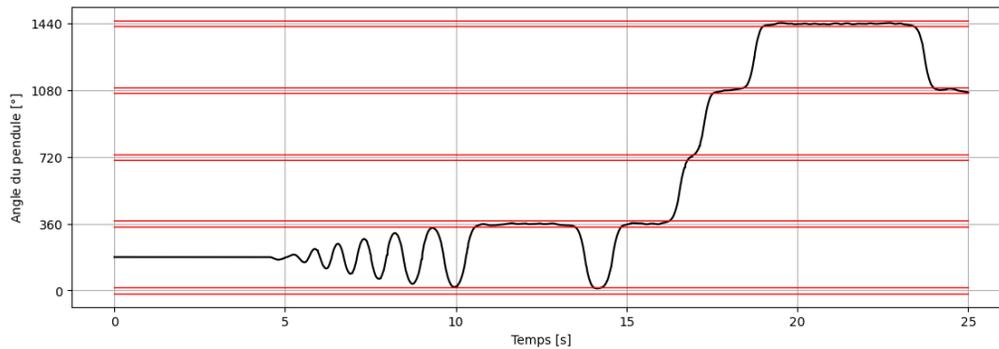


FIGURE 6.2 Trajectoire angulaire du pendule pour le même essai que celui présenté à la figure 6.1. Les zones délimitées par les lignes rouges représentent les angles de 15° et -15° près du point d'équilibre. La proportion de temps passé sous 15° du pendule lors de cet essai était de 48.2% .

Les succès en simulation et sur le montage expérimental démontrent qu'il est possible qu'un ESN émule une loi de contrôle à partir des accélérations subies pour un pendule (rotatif) inverse.

6.2.2 Contrôle par PRC

Avec des contrôleurs utilisant des réservoirs physiques (processeur et capteur neuromorphique), les performances laissent à désirer. La moyenne de la distribution de proportion du processeur et du capteur neuromorphique sont respectivement de 0.183 ± 0.058 et de 0.154 ± 0.046 . Ces performances ont été obtenues avec une structure de sortie à 3 régressions et un entraînement par lots.

Il n'y a pas de facteur unique auquel attribuer cette grande baisse de performance par rapport aux résultats obtenus avec les ESNs. Moins d'information est disponible pour le capteur ou le processeur neuromorphique puisque l'accélération dans une seule direction est utilisée comme entrée au réservoir physique. Cette hypothèse a été partiellement validée avec un ESN. Des tests préliminaires ont été effectués avec un réservoir qui utilise la rétroaction, une structure à trois régressions et un entraînement en ligne. La distribution de performances résultante est présentée à la figure 6.3. En moyenne, le contrôleur passe (15.3 ± 3.1) % du temps près de son point d'équilibre. Ces données appuient l'hypothèse que le manque d'information contribue aux faibles performances. Il faudrait toutefois répéter cette expérience avec un grand nombre de réservoirs pour la valider de façon définitive.

D'autres facteurs influencent aussi les performances. Dans le cas du PRC, le signal d'accélération est distordu par les montages expérimentaux. Les actionneurs utilisés pour reproduire les accélérations au LCSM ont une course limitée (5 mm pour le pot vibrant) et

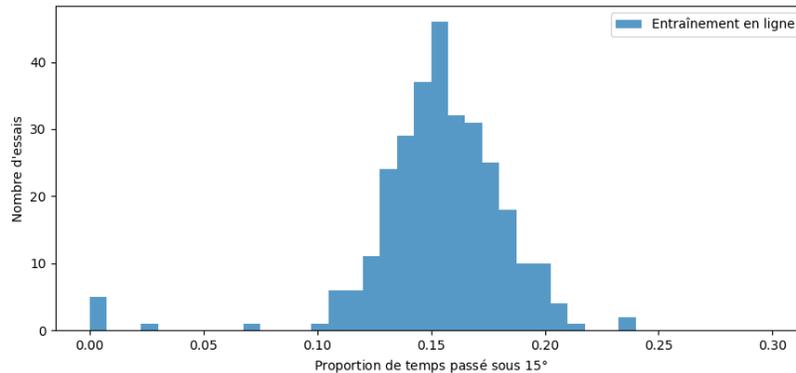


FIGURE 6.3 Distribution de performance obtenue lors de l’entraînement en ligne par un ESN à une accélération, avec une rétroaction et une structure à 3 régressions.

appliquent un filtre mécanique sur le signal. Les basses fréquences de l’accélération sont donc coupées et les accélérations à différentes fréquences sont amplifiées avec un gain variable, ce qui influence le résultat. Aussi, les propriétés (voir section 2.2.3) du réservoir physiques ne peuvent pas être ajustées sur une plage aussi large que celle des ESNs. En effet, tel qu’il est possible de le voir à la figure 6.4, les mesures de KR et de GR pour chaque type de réseau ne sont pas contenues dans les mêmes régions d’un espace formé par les propriétés. Il est donc possible que les signaux générés par les réservoirs physiques testés jusqu’à maintenant n’aient simplement pas été assez riches ou diversifiés pour reproduire une commande de contrôle précise à tout moment.

Les hyperparamètres des ESNs testés à la figure 6.4 sont choisis aléatoirement parmi les étendues présentées au tableau 6.1. Seuls deux hyperparamètres du substrat physique ont été explorés pour ce test. Il serait pertinent de refaire cette caractérisation pour une plus grande étendue de paramètres. Il est aussi à noter que toutes les données du réservoir physique ont été prises sur le même substrat, ce qui pourrait influencer les résultats.

6.3 Stabilité des contrôleurs

Le contrôleur de référence développé par Gordillo et al. [4] utilise une loi de contrôle trouvée par modulation d’énergie. Cette loi de contrôle est stable, dans le sens que la commande est en tout temps calculée pour que le pendule se dirige vers le point d’équilibre à son sommet, pour chaque point de l’espace de phase du pendule de Furuta. La même chose peut être dite pour la portion LQR du contrôleur dans un espace restreint où les équations peuvent être linéarisées. Le gain sur l’état du pendule a été choisi pour que la commande

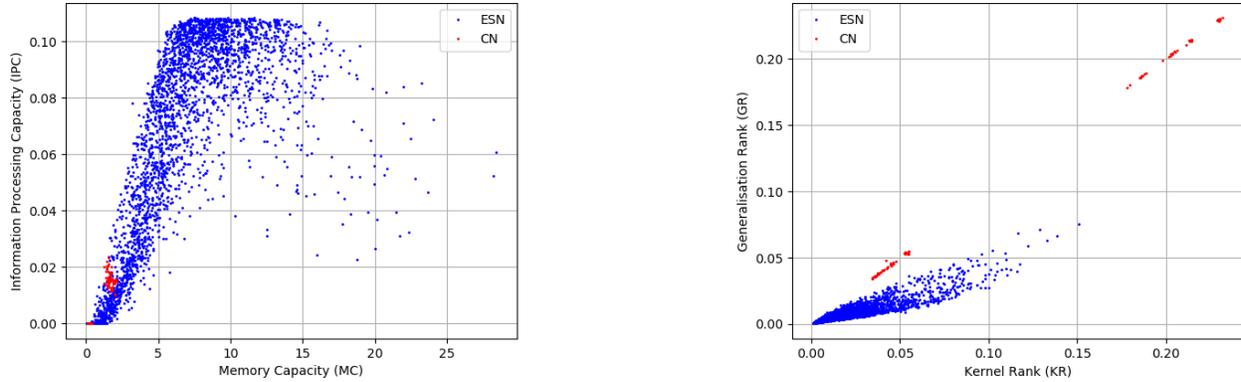


FIGURE 6.4 Distribution des propriétés de différents réservoirs. 3000 ESNs aléatoires ont été testés et 100 réservoirs physiques ont été testés avec le capteur neuromorphique. Les deux types de réservoirs n’occupent pas les mêmes régions dans l’espace des hyperpropriétés. Des points de donnée additionnels sont nécessaires avec les réservoirs physiques pour pouvoir mieux comparer les limitations de chaque type de réservoir.

Hyperparamètre	Valeur Minimale	Valeur Maximale
ESN		
Taux de Fuite	0	1
Rayon Spectral	0.3	1.2
Parcimonie du réservoir	0.001	0.8
Échelle d’entrée	0.5	3
Parcimonie d’entrée	0.1	1
PRC		
Temps de mesure d’un noeud [μs]	50	100
Fréquence d’alimentation [kHz]	246.8	248.4

TABLEAU 6.1 Bornes de l’exploration des hyperpropriétés des différents types de réservoirs. Les propriétés de l’ESN ont été explorées en utilisant des distributions uniformes aléatoires entre les bornes présentées. Pour le réservoir physique, les propriétés ont été explorées en utilisant une exploration par grille, comme le but initial de ces essais était d’isoler différents régimes d’opération du substrat.

appliquée par l'actuateur fasse converger le pendule vers un point d'équilibre au sommet de sa trajectoire.

Or, la même chose ne peut pas être dite pour un contrôleur RC. La commande est calculée à partir d'un réservoir, qui est en lui-même un système dynamique qui a une dépendance à ses anciens états. La commande calculée au temps présent dépend non seulement de l'état présent du pendule, mais aussi de la trajectoire qui a été empruntée jusqu'à maintenant. Cette différence rend difficile de prouver qu'un contrôleur est stable et c'est pourquoi des distributions sont utilisées pour caractériser les performances des contrôleurs. Cependant, tel qu'il est possible de le voir au tableau 5.3, l'écart-type absolu associé aux distributions de contrôleur RC sur le montage expérimental est toujours entre 4 et 7% de la proportion de temps passé près du point d'équilibre. Comme indicateur, la distribution de performance du contrôleur de référence a un écart-type de 5.8 % de la proportion de temps passé près du point d'équilibre.

À première vue, cette mince différence pourrait être interprétée comme preuve que les distributions des contrôleurs RC ont la même incertitude que la distribution du contrôleur de référence. Cette incertitude sur les distributions proviendrait donc plutôt du pendule de Furuta puisqu'il a été utilisé avec tous les types d'entraînement et de réservoir. D'un autre côté, en calculant les écarts-types relatifs plutôt qu'absolus de chaque distribution, une seconde interprétation peut être effectuée. L'écart-type relatif est obtenu en divisant l'écart-type absolu de chaque distribution par sa moyenne respective. Comme illustré à la figure 6.5, les écarts-types relatifs des deux types de réservoirs sont regroupés en strates selon le type de réseau. Il est donc possible que l'écart-type des distributions soit plutôt lié au type de réservoir utilisé et à sa dynamique. Plus d'essais avec des réservoirs variés sont nécessaires pour valider cette hypothèse.

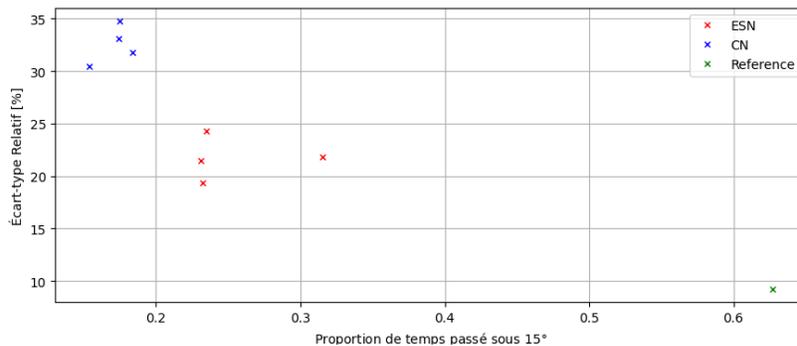


FIGURE 6.5 Comparaison des écarts-types relatifs des différentes distributions selon le type de réseau utilisé et la moyenne de leur performance.

CHAPITRE 7

Conclusion

L'objectif principal de cette maîtrise était de démontrer qu'un pendule de Furuta pouvait être asservi à un capteur neuromorphique (CN). Une contrainte additionnelle a dû être respectée ; le CN développé par le laboratoire est seulement sensible à l'accélération dans un axe. Ceci complique la démonstration, comme le problème attaqué est classiquement résolu à partir de l'état du pendule de Furuta, c'est-à-dire une mesure d'angle et de vitesse angulaire pour le pendule et pour le bras. Pour compenser cette difficulté ajoutée, une approche incrémentale a été choisie pour arriver à des résultats. Une simulation ainsi que deux montages expérimentaux, le pendule de Furuta et le banc de test de CN, ont été utilisés pour arriver aux résultats présentés dans ce mémoire.

Pour récapituler, les premiers tests ont été effectués avec des analogues numériques des capteurs neuromorphiques, soit des ESNs. Ceux-ci ont plusieurs avantages concernant leur adaptabilité par rapport à leur contrepartie physique ; il est facile de changer chacun de leurs paramètres et ils sont faciles à intégrer expérimentalement. Ces facteurs ont fait des ESN un outil indispensable pour les tests initiaux de contrôleurs et pour choisir les différentes structures et méthodes d'entraînement qui ont par la suite été implémentées avec le CN. En utilisant les ESNs, le contrôle d'un pendule inverse sur un chariot a tout d'abord été accompli en simulation où en moyenne 81.1 % du temps total de chaque essai était passé près d'un point d'équilibre (section 4.1). À partir des apprentissages faits en simulation, des améliorations ont été apportées et un ESN est arrivé à effectuer un contrôle partiel du pendule de Furuta. La moyenne de proportion de temps passé près d'un point d'équilibre avec ce contrôleur est de 31.5 %, avec des performances maximales de 48.2 % ayant été atteintes lors d'un essai expérimental (section 4.2). Il est à noter que dans ces deux cas, les accélérations dans trois axes ont été utilisées comme entrées pour l'ESN. Ces premiers résultats démontrent donc qu'il est possible d'effectuer le contrôle de systèmes sous-actuels à l'aide d'un ESN sans avoir d'information directe sur l'angle du pendule, autant en simulation qu'expérimentalement. Il est à noter que des contrôleurs utilisant les accélérations comme entrées pour la résolution du problème de pendule inverse ne semblent pas être discutés dans la littérature.

Finalement, un passage au CN a été effectué. Comme un circuit intégré n'a pas encore été développé pour le CN, un lien temps réel entre le pendule de Furuta et le banc de

test du CN a été conçu pour relayer les commandes générées (section 5.1). Tout d'abord, un processeur neuromorphique a été entraîné pour effectuer le contrôle puis un capteur neuromorphique. Dans ces deux cas, la moyenne de proportion de temps passé près de leur point d'équilibre est respectivement de 18.3 % et de 15.4 % lors de tests de contrôleurs (sections 5.3 et 5.4). En observant les diagrammes de phase (voir figure 5.6), on peut conclure que ces deux méthodes peuvent être utilisées pour ajouter de l'énergie à un système sous-actué comme le pendule de Furuta, mais elle n'arrive pas à le stabiliser.

Pour arriver à prouver qu'un système sous-actué peut être contrôlé à l'aide d'un capteur neuromorphique, des travaux subséquents seront nécessaires. Des modifications à la méthodologie utilisée pourraient permettre de trouver des solutions aux problèmes rencontrés. Par exemple, utiliser des métriques pour quantifier les propriétés des réservoirs pourrait permettre de trouver des réservoirs plus appropriés au problème (section 6.2.2). Un recadrage du problème pourrait aussi être bénéfique. Comme le contrôleur arrive à faire basculer le pendule, il pourrait être bénéfique de morceler le problème et de travailler uniquement sur le maintien de l'équilibre du pendule, de façon analogue à l'environnement d'entraînement présenté par Brockman et al. [44].

LISTE DES RÉFÉRENCES

- [1] L. Appeltant, M. C. Soriano, G. Van der Sande, J. Danckaert, S. Massar, J. Dambre, B. Schrauwen, C. R. Mirasso, and I. Fischer. Information processing using a single dynamical node as complex system. *Nature Communications*, 2(11) :1–6, Sep 2011.
- [2] Guillaume Dion, Salim Mejaouri, and Julien Sylvestre. Reservoir computing with a single delay-coupled non-linear mechanical oscillator. *Journal of Applied Physics*, 124(15) :152132, Oct 2018.
- [3] Bruno Barazani, Guillaume Dion, Jean-Francois Morissette, Louis Beaudoin, and Julien Sylvestre. Microfabricated neuroaccelerometer : Integrating sensing and reservoir computing in mems. *Journal of Microelectromechanical Systems*, 29(3) :338–347, Jun 2020.
- [4] F. Gordillo and J. Aracil. A new controller for the inverted pendulum on a cart. *International Journal of Robust and Nonlinear Control*, 18(17) :1607–1621, Nov 2008.
- [5] Kohei Nakajima. Physical reservoir computing—an introductory perspective. *Japanese Journal of Applied Physics*, 59(6) :060501, Jun 2020.
- [6] K. Furuta, M. Yamakita, and S. Kobayashi. Swing up control of inverted pendulum. In *Proceedings IECON '91 : 1991 International Conference on Industrial Electronics, Control and Instrumentation*, page 2193–2198. IEEE, 1991.
- [7] Benjamin Seth Cazzolato and Zebb Prime. On the dynamics of the furuta pendulum. *Journal of Control Science and Engineering*, 2011 :1–8, 2011.
- [8] Isabelle Fantoni and Rogelio Lozano. Stabilization of the furuta pendulum around its homoclinic orbit. *IFAC Proceedings Volumes*, 34(6) :807–812, Jul 2001.
- [9] K. Furuta, M. Yamakita, S. Kobayashi, and M. Nishimura. A new inverted pendulum apparatus for education. *IFAC Proceedings Volumes*, 25(12) :133–138, Jun 1991.
- [10] Karl J. Astrom and Richard M. Murray. *Feedback systems : an introduction for scientists and engineers*. Princeton University Press, 2008.
- [11] Graham C Goodwin, Stefan F Graebe, Mario E Salgado, et al. *Control system design*. Upper Saddle River, NJ : Prentice Hall,, 2001.
- [12] F. L. Lewis, D. Vrabie, and K. G. Vamvoudakis. Reinforcement learning and feedback control : Using natural decision methods to design optimal adaptive controllers. *IEEE Control Systems Magazine*, 32(6) :76–105, 2012.
- [13] F. Gordillo, J. A. Acosta, and J. Aracil. A new swing-up law for the furuta pendulum. *International Journal of Control*, 76(8) :836–844, Jan 2003.
- [14] Pedro X. La Hera, Leonid B. Freidovich, Anton S. Shiriaev, and Uwe Mettin. New approach for swinging up the furuta pendulum : Theory and experiments. *Mechatronics*, 19(8) :1240–1250, Dec 2009.
- [15] H Kim, Michael Jordan, Shankar Sastry, and Andrew Ng. Autonomous helicopter flight via reinforcement learning. *Advances in neural information processing systems*, 16 :799–806, 2003.

- [16] Andrew Y Ng and Michael I Jordan. Pegasus : A policy search method for large mdps and pomdps. *arXiv preprint arXiv :1301.3878*, 2013.
- [17] Mun-Soo Park and Dongkyoung Chwa. Swing-up and stabilization control of inverted-pendulum systems via coupled sliding-mode control method. *IEEE Transactions on Industrial Electronics*, 56(9) :3541–3555, Sep 2009.
- [18] N. Muskinja and B. Tovornik. Swinging up and stabilization of a real inverted pendulum. *IEEE Transactions on Industrial Electronics*, 53(2) :631–639, Apr 2006.
- [19] Herbert Jaeger. The”echo state”approach to analysing and training recurrent neural networks. In *N/A*, 2001.
- [20] Wolfgang Maass, Thomas Natschläger, and Henry Markram. Real-time computing without stable states : A new framework for neural computation based on perturbations. *Neural Computation*, 14(11) :2531–2560, Nov 2002.
- [21] Prashant Joshi and Wolfgang Maass. Movement generation with circuits of spiking neurons. *Neural Computation*, 17(8) :1715–1738, Aug 2005.
- [22] K. Caluwaerts, M. D’Haene, D. Verstraeten, and B. Schrauwen. Locomotion without a brain : Physical reservoir computing in tensegrity structures. *Artificial Life*, 19(1) :35–66, Nov 2012.
- [23] Jean C. Coulombe, Mark C. A. York, and Julien Sylvestre. Computing with networks of nonlinear mechanical oscillators. *PLOS ONE*, 12(6) :e0178663, Jun 2017.
- [24] Mantas Lukoševičius. *A Practical Guide to Applying Echo State Networks*, volume 7700 of *Lecture Notes in Computer Science*, page 659–686. Springer Berlin Heidelberg, 2012.
- [25] Herbert Jaeger, Mantas Lukoševičius, Dan Popovici, and Udo Siewert. Optimization and applications of echo state networks with leaky- integrator neurons. *Neural Networks*, 20(3) :335–352, Apr 2007.
- [26] A. L. Hodgkin and A. F. Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of Physiology*, 117(4) :500–544, 1952.
- [27] Prashant Joshi and Wolfgang Maass. *Movement Generation and Control with Generic Neural Microcircuits*, volume 3141 of *Lecture Notes in Computer Science*, page 258–273. Springer Berlin Heidelberg, 2004.
- [28] Marc-Antoine Moïnnereau, Thomas Brienne, Simon Brodeur, Jean Rouat, Kevin Whittingstall, and Eric Plourde. Classification of auditory stimuli from eeg signals with a regulated recurrent neural network reservoir. *arXiv :1804.10322 [cs, eess]*, Apr 2018. arXiv : 1804.10322.
- [29] Mantas Lukoševičius and Herbert Jaeger. Reservoir computing approaches to recurrent neural network training. *Computer Science Review*, 3(3) :127–149, Aug 2009.
- [30] A. U. Kucukemre. Echo state networks for adaptive filtering, 2006.
- [31] Matthew Dale, Julian F. Miller, Susan Stepney, and Martin A. Trefzer. A substrate-independent framework to characterize reservoir computers. *Proceedings of the Royal Society A : Mathematical, Physical and Engineering Sciences*, 475(2226) :20180723, Jun 2019.

- [32] Joni Dambre, David Verstraeten, Benjamin Schrauwen, and Serge Massar. Information processing capacity of dynamical systems. *Scientific reports*, 2(1) :1–7, 2012.
- [33] Matthew Dale, Julian F. Miller, and Susan Stepney. *Reservoir Computing as a Model for In-Materio Computing*, page 533–571. Emergence, Complexity and Computation. Springer International Publishing, 2017.
- [34] Kohei Nakajima, Helmut Hauser, Tao Li, and Rolf Pfeifer. Information processing via physical soft body. *Scientific Reports*, 5(1) :10487, Sep 2015.
- [35] Minhang Bao. *Analysis and design principles of MEMS devices*. Elsevier, 1. ed edition, 2005.
- [36] N. Yazdi, F. Ayazi, and K. Najafi. Micromachined inertial sensors. *Proceedings of the IEEE*, 86(8) :1640–1659, Aug 1998.
- [37] Marc Sansa, Guillaume Goulat, Guillaume Jourdan, Marc Gely, Patrick Villard, Gilles Sicard, and Sébastien Hentz. Compact heterodyne nems oscillator for sensing applications. *Solid-State Electronics*, 125 :214–219, Nov 2016.
- [38] Reza Abdolvand, Behraad Bahreyni, Joshua E.-Y. Lee, and Frederic Nabki. Micro-machined resonators : A review. *Micromachines*, 7(99) :160, Sep 2016.
- [39] J. Degraeve, K. Caluwaerts, J. Dambre, and F. Wyffels. Developing an embodied gait on a compliant quadrupedal robot. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, page 4486–4491, Sep 2015.
- [40] André Fortin. *Équations différentielles*, chapter 7 - Équations différentielles. Presses Internationales Polytechnique, 2012.
- [41] John R Taylor. *Équations de Lagrange*, chapter 7 - Équations de Lagrange. de boeck, 2005.
- [42] David Sussillo and Larry F Abbott. Generating coherent patterns of activity from chaotic neural networks. *Neuron*, 63(4) :544–557, 2009.
- [43] Tim Waegeman. *Adaptive control of compliant robots with Reservoir Computing*. PhD thesis, Ghent University, 2013.
- [44] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv :1606.01540*, 2016.

