

Apprentissage incrémental de l'opérateur de Koopman pour systèmes non-autonomes via prédiction conforme

Ben GAO^{1,2} Jordan PATRACONE^{1,2} Stéphane CHRÉTIEN³ Olivier ALATA²

¹Inria

²Université Jean Monnet Saint-Etienne, CNRS, Institut d'Optique Graduate School, Laboratoire Hubert Curien UMR 5516, F-42023, SAINT-ETIENNE, France

³Université Lyon 2, Laboratoire ERIC, UFR ASSP, 69676 Bron, France

Résumé – L'opérateur de Koopman permet de linéariser certaines dynamiques non linéaires via une action sur les observables, mais son extension au cas non-autonome reste un défi. Nous proposons une méthode incrémentale pour estimer cet opérateur dans un cadre non-autonome, en s'appuyant sur la prédiction conforme pour détecter automatiquement les changements de régime. Les paramètres du modèle sont ajustés uniquement lorsque la performance locale se dégrade, assurant des mises à jour parcimonieuses tout en conservant une précision comparable à l'apprentissage hors-ligne. Des expériences préliminaires sur le système de Duffing illustrent l'efficacité de notre approche.

Abstract – The Koopman operator enables the linearization of certain nonlinear dynamical systems by acting on observables, but its extension to the non-autonomous setting remains challenging. We propose an incremental method to estimate the Koopman operator in a non-autonomous context, leveraging conformal prediction to automatically detect regime changes. Model parameters are updated only when local performance deteriorates, ensuring sparse updates while maintaining accuracy comparable to off-line training. Preliminary experiments on the Duffing oscillator illustrate the effectiveness of our approach.

1 Introduction

La modélisation et la compréhension des systèmes dynamiques jouent un rôle central dans de nombreuses disciplines scientifiques. Lorsqu'ils sont non linéaires ou non-autonomes, leur analyse devient particulièrement complexe [11, 2].

Systèmes dynamiques. Nous considérons un espace métrique (X, d) sur lequel un système dynamique discret est défini par une règle d'évolution associant à chaque état $\mathbf{x}_t \in X$ un nouvel état \mathbf{x}_{t+1} . Dans le cas *autonome*, cette règle est donnée par une application $\mathbf{F} : X \rightarrow X$, de sorte que l'évolution s'écrit $\mathbf{x}_{t+1} = \mathbf{F}(\mathbf{x}_t)$. La trajectoire issue d'une condition initiale $\mathbf{x}_0 \in X$ est alors $\mathbf{x}_0, \mathbf{x}_1 = \mathbf{F}(\mathbf{x}_0), \mathbf{x}_2 = \mathbf{F}(\mathbf{x}_1), \dots$, soit $\mathbf{x}_t = \mathbf{F}^t(\mathbf{x}_0)$. En revanche, un système est dit *non-autonome* lorsque la règle d'évolution dépend explicitement du temps : $\mathbf{x}_{t+1} = \mathbf{F}_t(\mathbf{x}_t)$, où chaque $\mathbf{F}_t : X \rightarrow X$ peut varier avec t . Pour analyser ces trajectoires, il est souvent plus pertinent d'adopter une perspective fonctionnelle.

Approche fonctionnelle et opérateur de Koopman. L'étude des systèmes dynamiques peut s'aborder sous un angle fonctionnel, en analysant non plus l'évolution des états $\mathbf{x}_t \in X$, mais celle des *observables*, c'est-à-dire des fonctions $g : X \rightarrow \mathbb{C}$ appartenant à un espace fonctionnel $\mathcal{F}(X)$ — typiquement un espace de Hilbert tel que $L^2(X, \mu)$ (avec μ mesure de Borel), ou un RKHS. Ce changement de point de vue permet d'introduire l'opérateur de Koopman [11, 2], un opérateur linéaire qui encode la dynamique par composition.

Dans le cas *autonome*, l'opérateur de Koopman $\mathcal{K} : \mathcal{F}(X) \rightarrow \mathcal{F}(X)$ est défini par :

$$\mathcal{K}g := g \circ \mathbf{F}. \quad (1)$$

L'ensemble des valeurs propres de \mathcal{K} constitue le spectre ponctuel de \mathcal{K}

$$\sigma_p(\mathcal{K}) = \{\lambda \in \mathbb{C} : \mathcal{K} - \lambda I \text{ est non injectif}\}. \quad (2)$$

La trajectoire d'une observable initiale g_0 est alors donnée par $g_t = \mathcal{K}^t g_0$. L'opérateur \mathcal{K} est linéaire, ce qui permet une analyse spectrale : si $\mathcal{K}\varphi = \lambda\varphi$, alors $\varphi(\mathbf{x}_t) = \lambda^t \varphi(\mathbf{x}_0)$. Lorsque les fonctions propres $\{\varphi_1, \dots, \varphi_L\}$ définissent un plongement injectif de l'espace d'état, le système peut être linéarisé via le changement de coordonnées $\mathbf{x} \mapsto (\varphi_1(\mathbf{x}), \dots, \varphi_L(\mathbf{x}))$.

En revanche, dans le cas *non-autonome*, la dynamique dépend explicitement du temps. Le formalisme de Koopman [7] s'adapte en introduisant une famille d'opérateurs $(\mathcal{K}_t)_{t \in \mathbb{N}}$ définis par $\mathcal{K}_t g := g \circ \mathbf{F}_t$. La trajectoire d'une observable s'écrit alors récursivement : $g_{t+1} = \mathcal{K}_t g_t = \mathcal{K}_t \mathcal{K}_{t-1} \dots \mathcal{K}_0 g_0$. Comme l'opérateur \mathcal{K}_t change à chaque instant, les outils spectraux classiques ne s'appliquent pas directement. Il n'existe en général ni sous-espace invariant, ni base propre commune aux \mathcal{K}_t , ce qui empêche toute linéarisation globale via un unique changement de coordonnées. Notons que cette difficulté ne se limite pas au cas non-autonome : même dans un cadre autonome, une telle linéarisation n'est possible que si l'espace des observables choisies est invariant sous l'action de \mathcal{K} . Cette propriété dépend de la structure spectrale de l'opérateur, donc de la dynamique \mathbf{F} et de l'espace fonctionnel $\mathcal{F}(X)$. Par exemple, un changement de coordonnées peut exister pour des systèmes à attracteur global [4], mais reste inapplicable dans les régimes faiblement mélangés [11].

Approximation finie de l'opérateur de Koopman. L'opérateur de Koopman agissant sur un espace fonctionnel infini-dimensionnel, on cherche en pratique à l'approximer dans un

sous-espace de dimension finie engendré par un ensemble d'observables $\{f_1, \dots, f_L\} \subset \mathcal{F}(X)$. En définissant l'application de plongement $\Phi : X \rightarrow \mathbb{C}^L$ par $\Phi(\mathbf{x}) = (f_1(\mathbf{x}), \dots, f_L(\mathbf{x}))$, la dynamique est approchée par une matrice $\mathbf{K} \in \mathbb{C}^{L \times L}$ telle que :

$$\mathbf{z}_{t+1} = \mathbf{K}\mathbf{z}_t, \quad \text{avec } \mathbf{z}_t = \Phi(\mathbf{x}_t). \quad (3)$$

Dans le cas *autonome*, la *Dynamic Mode Decomposition* (DMD) [13, 12] a ouvert la voie à l'approximation de \mathcal{K} en supposant une base linéaire d'observables. Elle a été généralisée par l'*Extended DMD* (EDMD) [14], qui utilise un dictionnaire non-linéaire fixe. L'état de l'art repose sur les architectures auto-encodeur [6, 10], où le plongement Φ , le décodeur Ψ et la dynamique linéaire \mathbf{K} sont appris conjointement. Ces approches identifient un sous-espace approximativement invariant, chaque observable f_j s'écrivant comme combinaison linéaire de fonctions propres φ_ℓ :

$$f_j \approx \sum_{\ell=1}^L c_{j\ell} \varphi_\ell. \quad (4)$$

En revanche, le cas *non-autonome* reste plus délicat. L'opérateur de Koopman doit être approximé dynamiquement. Une stratégie courante consiste à supposer la dynamique localement stationnaire sur des fenêtres de taille w , i.e. $\mathbf{F}_\tau \approx \mathbf{F}_t$ pour $\tau \in [t, t+w]$ [3]. Ces méthodes sont efficaces tant que les variations sont lentes, mais échouent en cas de changements de régime brutaux [8]. Une autre approche s'applique aux dynamiques périodiques, pour lesquelles un changement de variable $\tilde{\mathbf{x}}_t = (\mathbf{x}_t, t)$ permet de revenir à un système autonome dans un espace augmenté, rendant applicable l'opérateur de Koopman standard [9].

Enfin, quelques travaux récents abordent explicitement l'apprentissage incrémental de l'opérateur de Koopman. En particulier, [15] propose une méthode efficace pour calculer la DMD en ligne, actualisant la matrice \mathbf{K} à chaque nouvelle observation via une mise à jour de rang 1, sans avoir à stocker les données passées. En intégrant un facteur d'oubli, la méthode s'adapte aux changements progressifs de la dynamique \mathbf{F}_t , mais reste limitée au cadre linéaire. Autrement, [5] étend cette idée à des représentations non linéaires via un auto-encodeur, mais sans garantie de convergence ni gestion explicite des ruptures.

Contributions. Nous abordons l'estimation de l'opérateur de Koopman en contexte non-autonome via un apprentissage incrémental guidé par la prédiction conforme. Cette approche permet d'identifier les régions temporelles où le modèle devient invalide, facilitant la détection des ruptures et l'adaptation dynamique. Les résultats numériques montrent (i) un gain computationnel, même en régime autonome, et (ii) une capacité à traiter des dynamiques non-autonomes sans hypothèse forte sur leur structure temporelle.

2 Méthodologie

Modèle. Suivant [6, 10], nous nous plaçons dans $\mathcal{F} = L^2(X, \mu)$ et nous approchons un sous-espace invariant de l'opérateur de Koopman à l'aide d'un encodeur $\Phi : X \rightarrow \mathbb{C}^L$, et représentons la dynamique dans cet espace par une matrice $\mathbf{K} \in \mathbb{C}^{L \times L}$. En supposant Φ injective, son application inverse

est approchée par un décodeur $\Psi : \mathbb{C}^L \rightarrow X$, permettant de reconstituer les trajectoires dans l'espace d'état. La composition $\Psi \circ \mathbf{K} \circ \Phi$ approxime alors la dynamique

$$\mathbf{x}_{t+1} \approx \Psi(\mathbf{K}\Phi(\mathbf{x}_t)). \quad (5)$$

Ce modèle dépend de paramètres regroupés sous la forme $\theta := (\theta_\Phi, \theta_\Psi, \mathbf{K})$, correspondant respectivement aux poids de l'encodeur, du décodeur, et à la matrice de transition dans l'espace latent.

Dans le cas "hors ligne", comme dénommé par la suite, les paramètres du modèle $\theta = (\theta_\Phi, \theta_\Psi, \mathbf{K})$ sont appris en minimisant une fonction de perte composée de trois termes :

- une erreur de consistance $c_t = \|\mathbf{K}\Phi(\mathbf{x}_t) - \Phi(\mathbf{x}_{t+1})\|^2$,
- une erreur de reconstruction $r_t = \|\mathbf{x}_t - \Psi(\Phi(\mathbf{x}_t))\|^2$,
- et une erreur de prédiction $e_t = \|\mathbf{x}_{t+1} - \Psi(\mathbf{K}\Phi(\mathbf{x}_t))\|^2$.

La fonction objectif s'écrit alors :

$$\min_{\theta} \left[\ell(\theta, \{(\mathbf{x}_t, \mathbf{x}_{t+1})\}_{0 \leq i \leq T}) := \frac{1}{T} \sum_{t=0}^{T-1} (c_t + r_t + e_t) \right]. \quad (6)$$

Dans ce travail, pour faire face à une dynamique non-autonome, nous proposons une méthode incrémentale où les paramètres deviennent temporellement indexés : $\theta_t = (\theta_{\Phi_t}, \theta_{\Psi_t}, \mathbf{K}_t)$, et seront mis à jour à chaque instant à partir des observations courantes via la méthode suivante.

Apprentissage incrémental guidé par prédiction conforme.

Nous proposons une stratégie d'apprentissage incrémental des paramètres du modèle $\theta_t = (\theta_{\Phi_t}, \theta_{\Psi_t}, \mathbf{K}_t)$, mise à jour à chaque instant t en fonction de l'information locale. Pour décider si une mise à jour est nécessaire, nous nous appuyons sur la prédiction conforme, et en particulier sur l'algorithme *Conformal PID Control* (abrégé *ConformalPID*) proposé dans [1]. Celui-ci permet de construire une séquence de seuils $\{q_t\}$ contrôlant l'erreur acceptable à chaque pas de temps, sans supposer d'indépendance entre les données. Ce mécanisme est bien adapté à notre contexte déterministe, car il ne nécessite pas de modélisation probabiliste.

Le score de conformité est défini comme la somme des trois termes d'erreur introduits précédemment :

$$s_t = c_t(\theta_t) + r_t(\theta_t) + e_t(\theta_t), \quad (7)$$

calculé uniquement à partir de la nouvelle observation $(\mathbf{x}_t, \mathbf{x}_{t+1})$. À chaque pas de temps, la mise à jour de θ_t n'est effectuée que si $s_t > q_t$, c'est-à-dire si le modèle actuel ne parvient plus à prédire correctement la dynamique observée.

Le seuil q_t est mis à jour via l'algorithme *Conformal PI*, une version sans terme dérivé du schéma PID, définie par :

$$q_{t+1} = q_t + \eta_{\text{PI}} g_t + h_t \left(\sum_{i=t_0}^t g_i \right), \quad (8)$$

où le coefficient $\eta_{\text{PI}} > 0$ contrôle la réactivité du seuil aux erreurs récentes, $g_t = \mathbb{1}_{s_t \leq q_t} - \alpha$, et $h_t(\cdot)$ est une fonction de saturation. Ce mécanisme permet de garantir que la proportion d'itérations couvertes $\frac{1}{T} \sum_{t=1}^{T-1} \mathbb{1}_{s_t \leq q_t}$ converge vers $1 - \alpha$, assurant ainsi un contrôle statistique de l'effort d'apprentissage. Dit autrement, le paramètre $\alpha \in (0, 1)$ contrôle le niveau de tolérance à l'erreur : plus α est grand, plus les mises à jour sont fréquentes. L'algorithme résultant est le suivant.

Algorithme 1 : ApprentissageIncrémental

```
1 si DétectionDeChangement( $q_\delta, \dots, q_t$ ) > 0 alors
2   réinitialiser  $\theta_t$  de manière aléatoire
3    $\delta = t - T_{\text{écart}}$ ; // mise à jour rupture
4   entraîner  $\theta_t$  sur  $\{(\mathbf{x}_i, \mathbf{x}_{i+1})\}_{\delta \leq i < t}$ 
5    $s_i = c_i(\theta_t) + r_i(\theta_t) + e_i(\theta_t)$  pour  $\delta \leq i \leq t$ 
6    $q_{t+1} = \text{quantile}(\{s_i\}_{\delta \leq i \leq t}, 1 - \alpha)$ 
7 fin
8 sinon
9    $s_t = c_t(\theta_t) + r_t(\theta_t) + e_t(\theta_t)$ 
10   $q_{t+1} = \text{ConformalPI}(s_t)$ 
11 fin
12 tant que  $s_t > q_t$  faire
13    $\theta_t = \theta_t - \eta \nabla \ell(\theta_t, \{(\mathbf{x}_i, \mathbf{x}_{i+1})\}_{\delta \leq i \leq t})$ 
14    $s_t = c_t(\theta_t) + r_t(\theta_t) + e_t(\theta_t)$ 
15 fin
16  $\theta_{t+1} = \theta_t$ 
```

Notons que la descente de gradient¹ avec pas $\eta > 0$ (ligne 14) est réalisée à l'aide de toutes les données observées depuis le dernier changement détecté, c'est-à-dire sur l'intervalle $\{(\mathbf{x}_i, \mathbf{x}_{i+1})\}_{\delta \leq i \leq t}$, où δ est le dernier instant où une rupture a été identifiée. Le score s_t est quant à lui local, calculé uniquement à partir de la paire courante $(\mathbf{x}_t, \mathbf{x}_{t+1})$. Ce mécanisme permet de déclencher les mises à jour uniquement lorsque la performance locale du modèle devient insuffisante.

Une nouvelle rupture est identifiée à partir de la séquence des seuils $\{q_\delta, \dots, q_t\}$. Notons que la rupture en temps $t - T_{\text{écart}}$ ne peut être détectée qu'en temps t , où $T_{\text{écart}}$ est le nombre d'observations postérieures utilisées pour analyser s'il y avait une rupture en temps $t - T_{\text{écart}}$

Algorithme 2 : DétectionDeChangement

```
1  $q_{\text{post}} = q_t$ ;  $q_{\text{eval}} = q_t - T_{\text{écart}}$ ;  $q_{\text{pre}} = q_t - 2 \times T_{\text{écart}}$ 
2  $d_{\text{gauche}} = q_{\text{eval}} - q_{\text{pre}}$ ;  $d_{\text{droite}} = q_{\text{post}} - q_{\text{eval}}$ 
3 si  $d_{\text{gauche}} > 0$  alors
4   retourner 0
5 fin
6 si  $d_{\text{droite}} > 0$  alors
7   retourner 1
8 fin
9 si  $(d_{\text{droite}} - d_{\text{gauche}}) > (\alpha \times q_{\text{eval}})$  alors
10  retourner 1
11 fin
12 sinon
13   retourner 0
14 fin
```

3 Expériences numériques

Nous rapportons ici des résultats préliminaires illustrant les performances de notre algorithme incrémental dans un cas non linéaire de référence. Nous considérons le même système dynamique que dans [14, 10], à savoir l'oscillateur de Duffing :

$$\ddot{x} = -c\dot{x} - x(b + ax^2), \quad (9)$$

¹En pratique, cette étape est remplacée par une mise à jour à l'aide d'Adam.

où, sauf mention contraire, les paramètres sont fixés à $a = 1$, $b = -1$ et $c = 0.5$. Ce système présente deux attracteurs spiraux stables aux points $(x, \dot{x}) = (\pm 1, 0)$, séparés par un point selle instable à l'origine $(0, 0)$. Nous simulons $N = 50$ trajectoires $\{\mathbf{x}_t^{(i)}\}_{t=0}^T$, pour $i = 1, \dots, N$, avec un horizon $T = 1800$ et un pas de temps fixé à $\Delta t = 1/180$. Les conditions initiales sont échantillonnées uniformément dans le carré $[-2, 2] \times [-2, 2]$ de l'espace d'état (x, \dot{x}) . Cette dispersion des trajectoires initiales est nécessaire pour couvrir les différents bassins d'attraction du système.

Les encodeurs Φ et décodeurs Ψ sont implémentés sous forme de réseaux de neurones entièrement connectés de tailles respectives $[2, 32, 16, 8, 3]$ et $[3, 8, 16, 32, 2]$, avec des activations Tanh à chaque couche cachée. L'optimisation est effectuée à l'aide de l'algorithme Adam avec un taux d'apprentissage fixé à $\eta = 10^{-3}$.

Cas autonome. Nous commençons par le cas autonome tel que décrit dans le cadre expérimental précédent. Cette première expérience vise à évaluer les performances de notre méthode d'apprentissage incrémental en comparaison avec une approche hors ligne classique. L'objectif est de vérifier que l'approche incrémentale n'induit aucune perte de performance, tout en réduisant le coût d'optimisation.

La Fig. 1a illustre une trajectoire typique reconstruite par l'approche hors ligne et par notre méthode (avec $\alpha = 0.1$, $T_{\text{écart}} = 100$ et η_{PI} est calculé de la même manière que dans [1]), comparées à la trajectoire réelle (oracle). Les trois versions coïncident visuellement, confirmant que notre méthode atteint une précision comparable à celle de l'apprentissage hors-ligne. Notons par ailleurs que sur l'ensemble des 50 trajectoires, l'erreur absolue finale reste inférieure à 10^{-4} . La Fig. 1b montre également l'impact du paramètre α : plus α est élevé, plus les mises à jour sont rares, ce qui réduit le coût d'optimisation sans compromettre la précision.

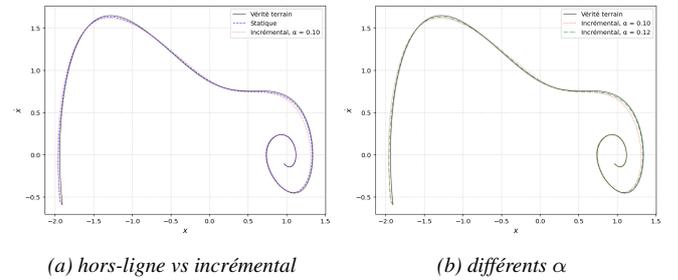


FIGURE 1 : **Cas autonome.** Comparaison des trajectoires : (a) l'apprentissage hors-ligne atteint une erreur de 7.25×10^{-5} après 10^4 itérations, tandis que notre méthode incrémentale obtient une erreur de 1.1×10^{-4} après seulement 2.58×10^3 itérations; (b) en réduisant le niveau de couverture avec $\alpha = 0.12$, l'erreur diminue à 3.62×10^{-5} en 5.72×10^3 itérations.

La Fig. 2a montre l'évolution temporelle du seuil q_t . Après une courte phase transitoire, q_t décroît progressivement et se stabilise autour de zéro à partir de $t \approx 400$. Cette convergence reflète le fait que le score de conformité s_t reste durablement inférieur au seuil : le modèle a appris une représentation suffisamment fidèle de la dynamique, et ses prédictions successives tombent dans l'intervalle de prédiction.

Enfin, la Fig. 2b compare le spectre estimé de la matrice K entre les deux approches. Les valeurs propres obtenues sont

quasi identiques, ce qui confirme que notre méthode restitue fidèlement la structure spectrale de l'opérateur de Koopman dans le cas autonome.

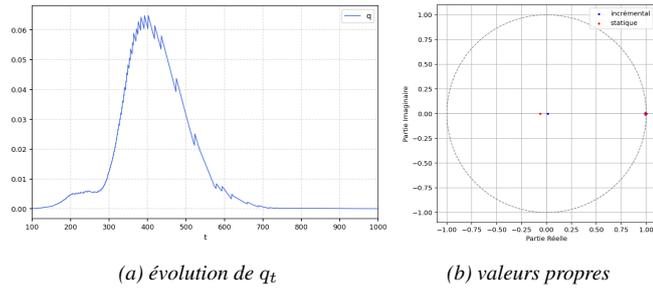


FIGURE 2 : **Cas autonome.** (a) Convergence de q_t et (b) comparaison des spectres.

Cas non-autonome. Nous considérons maintenant un cas non-autonome en faisant varier le paramètre c au cours du temps : $c = 3$ pour $0 \leq t < 600$, puis $c = 1$ pour $600 \leq t < 1200$, et enfin $c = 0,1$ pour $1200 \leq t \leq 1800$. Le système présente ainsi deux ruptures aux instants $t = 600$ et $t = 1200$. Cette seconde expérience a pour but d'évaluer la capacité de notre approche à détecter et s'adapter à ces changements de régime sans supervision explicite.

La Fig. 3a montre l'évolution du seuil q_t . En particulier, le changement brusque en $t = 600$ interrompt la convergence de q_t qui marque l'inadéquation du modèle appris avec les données du passé. Notre algorithme a détecté ce changement avec du retard en $t = 637$. La chute de q_t en $t = 737$ est dû à la réinitialisation et au ré-apprentissage du modèle après la détection. Après adaptation, q_t recommence à décroître, indiquant que le modèle s'est ajusté à la nouvelle dynamique. Le second changement en $t = 1200$, moins impactant localement, n'ont pas été détecté par notre algorithme, et q_t converge comme dans cas du système autonome.

Nous avons mis en œuvre un algorithme simple de détection basé sur les variations de q_t , qui permet d'estimer les points de rupture. Bien qu'élémentaire, cette procédure constitue une preuve de concept : elle démontre que notre cadre permet non seulement d'apprendre une dynamique non autonome, mais aussi d'identifier ses singularités. La Fig. 3b illustre les prédictions données par les deux modèles qui sont respectivement appris sur les données avant et après le changement détecté.

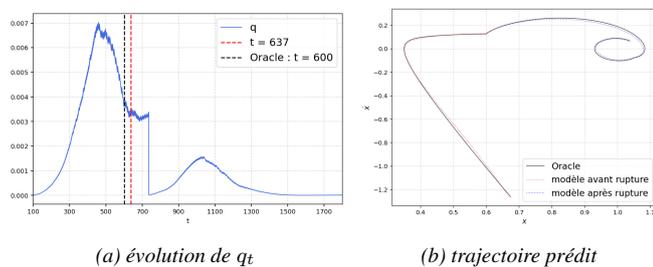


FIGURE 3 : **Cas non-autonome.** (a) l'évolution de q_t permet de détecter le changement (b) l'erreur est de $4,09 \times 10^{-5}$ après $5,58 \times 10^3$ itérations

4 Conclusion

Nous avons proposé une méthode incrémentale pour l'estimation de l'opérateur de Koopman dans un cadre non-autonome, en combinant apprentissage à la volée et prédiction conforme. Notre approche permet d'adapter dynamiquement le modèle tout en déclenchant les mises à jour de manière parcimonieuse. Les résultats préliminaires montrent une précision comparable à l'apprentissage hors-ligne et une bonne capacité à détecter les ruptures. Pour de futurs travaux, nous prévoyons d'améliorer le mécanisme de détection de changement et d'établir des garanties statistiques sur la performance du modèle.

Remerciements. Ces travaux ont bénéficié d'une subvention publique financée par la région Auvergne-Rhône-Alpes, Grenoble Alpes Métropole et BPIFrance, dans le cadre du projet I-Démo Région "Green AI".

Références

- [1] Anastasios ANGELOPOULOS, Emmanuel CANDES et Ryan J TIBSHIRANI : Conformal pid control for time series prediction. *In NeurIPS*, volume 36, pages 23047–23074, 2023.
- [2] Steven L. BRUNTON, Marko BUDIŠIĆ, Eurika KAISER et J. Nathan KUTZ : Modern koopman theory for dynamical systems. *SIAM Review*, 64(2):229–340, 2022.
- [3] J. Nathan KUTZ, Xing FU et Steven L. BRUNTON : Multiresolution dynamic mode decomposition. *SIAM Journal on Applied Dynamical Systems*, 15(2):713–735, 2016.
- [4] Yueheng LAN et Igor MEZIĆ : Linearization in the large of nonlinear systems and koopman operator spectrum. *Physica D : Nonlinear Phenomena*, 242(1):42–53, 2013.
- [5] Zhichao LIANG, Zixiang LUO, Keyin LIU, Jingwei QIU et Quanying LIU : Online learning koopman operator for closed-loop electrical neurostimulation in epilepsy. *IEEE Journal of Biomedical and Health Informatics*, 27(1):492–503, 2022.
- [6] Bethany LUSCH, J. Nathan KUTZ et Steven L. BRUNTON : Deep learning for universal linear embeddings of nonlinear dynamics. *Nature Communications*, 9(1), novembre 2018.
- [7] Senka MAČEŠIĆ et Nelida ČRNJARIĆ-ŽIĆ : *Koopman Operator Theory for Nonautonomous and Stochastic Systems*, pages 131–160. Cham, 2020.
- [8] Senka MAČEŠIĆ, Nelida Črnjarić ŽIĆ et Igor MEZIĆ : Koopman operator family spectrum for nonautonomous systems. *SIAM Journal on Applied Dynamical Systems*, 17(4):2478–2515, 2018.
- [9] Igor MEZIĆ et Amit SURANA : Koopman mode decomposition for periodic/quasi-periodic time dependence. *IFAC-PapersOnLine*, 49(18):690–697, 2016.
- [10] Samuel E. OTTO et Clarence W. ROWLEY : Linearly recurrent autoencoder networks for learning dynamics. *SIAM Journal on Applied Dynamical Systems*, 18(1):558–593, 2019.
- [11] Samuel E. OTTO et Clarence W. ROWLEY : Koopman operators for estimation and control of dynamical systems. *Annual Review of Control, Robotics, and Autonomous Systems*, 4(Volume 4, 2021):59–87, 2021.
- [12] Clarence ROWLEY, Igor MEZIĆ, Shervin BAGHERI, Philipp SCHLATTER et Dan HENNINGSON : Spectral analysis of nonlinear flows. *Journal of Fluid Mechanics*, 641:115–127, 2009.
- [13] Peter J. SCHMID : Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics*, 656:5–28, 2010.
- [14] Matthew O WILLIAMS, Ioannis G KEVREKIDIS et Clarence W ROWLEY : A data-driven approximation of the koopman operator : Extending dynamic mode decomposition. *Journal of Nonlinear Science*, 25:1307–1346, 2015.
- [15] Hao ZHANG, Clarence W. ROWLEY, Eric A. DEEM et Louis N. CATTAFESTA : Online dynamic mode decomposition for time-varying systems. *SIAM Journal on Applied Dynamical Systems*, 18(3):1586–1609, 2019.