



gipsa-lab

Controlling and learning in Robotics

The tradeoff between stability and performance ?

Transparents issus de la soutenance de thèse de E. Carvalho

Estéban Carvalho et **Nicolas Marchand**

gipsa-lab

2024-05-01



Outline

Introduction : Contexte, objectifs, modélisation, structures de contrôle

Contrôle et Apprentissage : utilisation d'un réseau de neurone pour améliorer le suivi de trajectoire

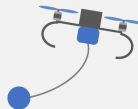
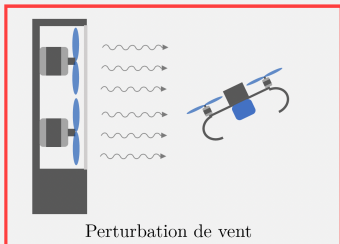
Vers une approche en ligne : un contrôle basé apprentissage sur évènement avec critères de stabilité et de performances

Ecole d'été

Objectif principal : Amélioration des performances de vol d'un drone quadrirotor

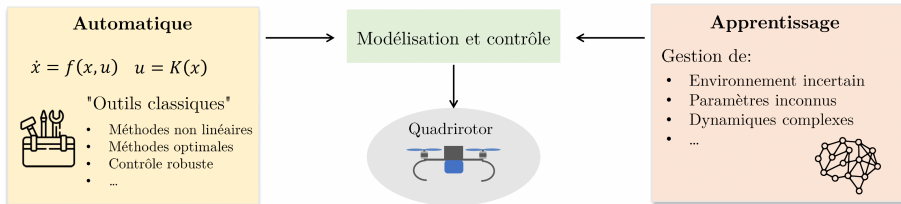
- **Performances ?** Vitesse de déplacement, erreur de suivi de trajectoire et robustesse aux perturbations.
- **Améliorer ?** Contrôle basé sur modèle : mauvaise modélisation, effets non modélisés, perturbations externes (vent), perturbations internes (masse inconnue), etc.

Exemple de cas étudiés dans mes travaux



Objectif principal : Amélioration des performances de vol d'un drone quadrirotor

- **Comment ?** En utilisant un mix des approches « automatique » et par « apprentissage ».
- **Pourquoi ?** Combiner les avantages des deux approches :
 - Bénéficier d'une base simple à paramétrer,
 - Utiliser la puissance des approches basées sur des données.



Etablir un modèle du quadrirotor

- Pourquoi ? ■ Connaissance générale du système
 ■ Etablir des lois de commande efficaces

Comment ? Bilan des forces et moments (Newton)

Modèle non linéaire du quadrirotor

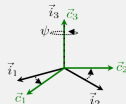
Modèle simplifié, le plus courant dans la littérature :

$$\begin{cases} \dot{\xi} &= v \\ m\dot{v} &= -mg\vec{e}_3 + R\vec{T}\vec{b}_3 \\ \dot{R} &= R\Omega_{\times} \\ R\dot{\Omega} &= -\Omega_{\times}J\Omega + \Gamma \end{cases}$$

► Les repères

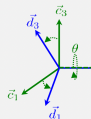
Formalisme Tait-Bryan (angles d'Euler)

Repère inertiel



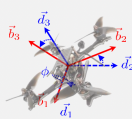
$R_z(\psi)$

Repère intermédiaire



$R_y(\theta)$

Repère du drone



$R_x(\phi)$

► Les notations

ξ	position	g	acc. normale de la pesanteur
v	vitesse	m	masse
R	matrice de rotation	J	matrice d'inertie
Ω	vitesse angulaires	Ω_{\times}	matrice antisymétrique
T	poussée		
Γ	moment		

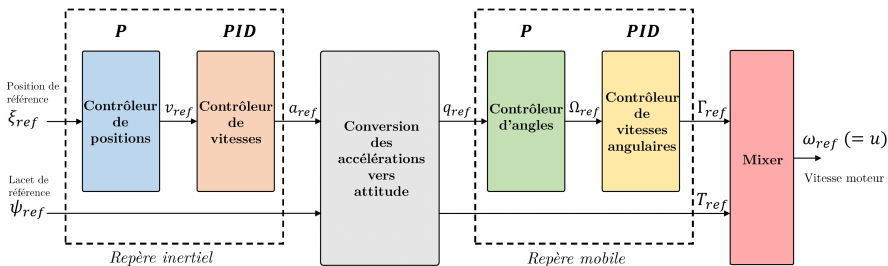
Modèle non linéaire du quadrirotor plus fidèle

$$\begin{cases}
 \ddot{\xi} &= v \\
 m\dot{v} &= -mg\vec{1}_3 + \alpha_g R \sum_{i=1}^4 R_i^{flap} T_i \vec{b}_3 - K_v |v|v + F_{ext} \\
 \dot{R} &= R\Omega_x \\
 R\dot{\Omega} &= -\Omega_x J\Omega + I_r(\Omega \times \vec{b}_3) \sum_{i=1}^4 \omega_i + \sum_{i=1}^4 (c_t \omega_i^2 R_i^{flap} \vec{b}_3 \times P_i) + \sum_{i=1}^4 (-1)^{i+1} c_q \omega_i^2 \vec{b}_3 + I_r \dot{\omega}_{res} \vec{b}_3 + \Gamma_{ext}
 \end{cases}$$

Poussée modifiée par effet de sol, la décharge de batterie et battement d'aile
Traînée
Force extérieure (inconnue)

Effet gyroscopique
Moment induit par la poussée
Couple de lacet
Contre couple inertiel
Moment extérieur (inconnu)

➔ Des effets complexes à modéliser et d'autres inconnus ...



Comment pilote-t-on un quadrirotor en pratique ?

- **Architecture en cascade :** boucles successives en position, vitesse, angles (« attitude ») et vitesses angulaires.
- **Régulateurs :** PID ~ 24 gains à paramétrer.

Quelques exemples de la littérature

Courses de drones IA pour détecter des portes

Elia Kaufmann et al. "Beauty and the Beast: Optimal Methods Meet Learning for Drone Racing." In: *2019 International Conference on Robotics and Automation (ICRA)*, IEEE, May 2019

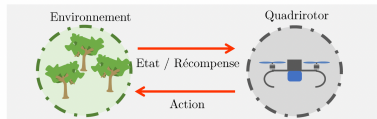
"En complément"



Contrôleur entièrement appris par renforcement

Nathan O. Lambert et al. "Low-Level Control of a Quadrotor With Deep Model-Based Reinforcement Learning." In: *IEEE Robotics and Automation Letters* 4.4, Oct. 2019

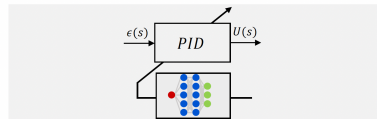
"En substitution"



Réglage de PID internes automatique

Yaonan Wang et al. "Fuzzy radial basis function neural network PID control system for a quadrotor UAV based on particle swarm optimization." In: *2015 IEEE International Conference on Information and Automation*. IEEE, Aug. 2015

"En parallèle"



Contrôle et Apprentissage

une combinaison compatible ? Mais...



Les avantages

- Amélioration des performances des systèmes
- Flexibilité permettant de gérer des systèmes complexes
- Valorisation des expériences passées



Les limites

- Complexité croissante des algorithmes
- Nécessité d'une grande quantité de données ...
- ... pertinentes pour l'apprentissage
- Temps d'apprentissage long
- Transfert complexe de la simulation vers l'expérimentation

Contrôle et Apprentissage

Approche adoptée

Pilotage de quadrirotor = architecture en cascade

- Maîtrise indépendante des boucles
- Simplicité de mise en place
- Simplicité de paramétrage (PID)



Solution proposée

Coupler l'architecture standard et l'apprentissage pour corriger le suivi de trajectoire

- Base simple de paramétrage
- Gérer des bases de données réduites
- Ne pas reposer la stabilité du système uniquement sur de l'apprentissage

"En parallèle"

Problématique

- ▶ Comment améliorer un contrôleur prédéfini afin d'obtenir les exigences fixées ?

Approche et méthodologie

- I. Synthèse** d'une commande sur le modèle linéarisé en vol stationnaire.
 - ↳ Comportement linéaire désiré
- II. Apprentissage** de l'erreur entre la dynamique linéaire et non linéaire.
 - ↳ Réseau de neurones profond (abr. RDN)
- III. Correction** de la commande par anticipation.

I. Spécification des performances

Choix des gains

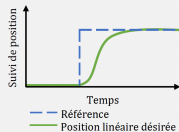
a. Choix d'un comportement désiré en boucle fermée :

$$u^* = -K(x - x_{ref}) + H\dot{x}_{ref}$$

Modèle linéaire
Dynamique désirée

$$\dot{x} = Ax + Bu$$

b. Spécification des performances sur un modèle linéaire



Contrôle et Apprentissage

Schéma de la méthodologie

I. Spécification des performances

Choix des gains

a. Choix d'un comportement désiré en boucle fermée :

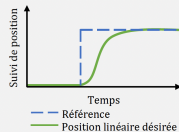
$$u^* = -K(x - x_{ref}) + H\dot{x}_{ref}$$

Modèle linéaire

Dynamique désirée

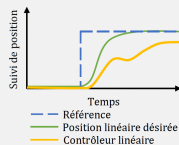
$$\dot{x} = Ax + Bu$$

b. Spécification des performances sur un modèle linéaire



II. Création d'une base de données

Résultat en suivi de trajectoire en utilisant le contrôleur linéaire



Contrôle et Apprentissage

Schéma de la méthodologie

I. Spécification des performances

Choix des gains

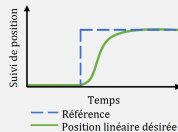
a. Choix d'un comportement désiré en boucle fermée :

$$u^* = -K(x - x_{ref}) + H\dot{x}_{ref}$$

Modèle linéaire
Dynamique désirée

$$\dot{x} = Ax + Bu$$

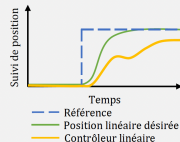
b. Spécification des performances sur un modèle linéaire



Quadrirrotor
Dynamique réelle



II. Création d'une base de données
Résultat en suivi de trajectoire en utilisant le contrôleur linéaire



III. Apprentissage d'une correction du vol

a. Erreur de dynamique

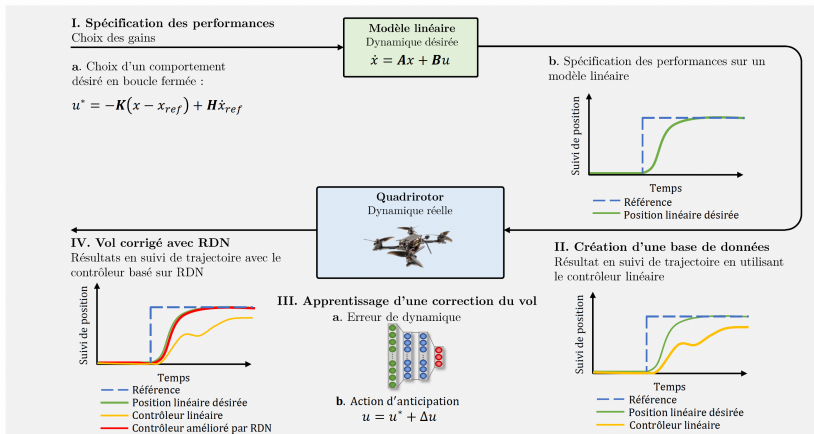


b. Action d'anticipation

$$u = u^* + \Delta u$$

Contrôle et Apprentissage

Schéma de la méthodologie



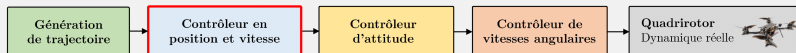


Schéma simplifié de l'architecture de contrôle du quadrirotor

Controler position/vitesse

- Modèle linéaire de l'erreur en position/vitesse

$$\dot{\eta} = \bar{A}\eta + \bar{B}(u_{\zeta} - B_{\zeta}^{-1}\dot{v}_{ref})$$

- Commande linéaire

$$u_{\zeta}^* = -K_{\eta}\eta + B_{\zeta}^{-1}\dot{v}_{ref}$$

~ équivalent à un régulateur proportionnel-dérivé

► Les notations

Vecteur d'état

$$\eta = \begin{bmatrix} \tilde{\xi} - \tilde{\xi}_{ref} \\ \tilde{v} - \tilde{v}_{ref} \end{bmatrix}$$

Commande

$$u_{\zeta} = \begin{bmatrix} \tilde{\phi} \\ \tilde{\theta} \\ \tilde{\tau} \end{bmatrix}$$

Matrice d'état

$$\bar{A} = \begin{bmatrix} \mathbf{0}_3 & I_3 \\ \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix}$$

Matrice de commande

$$\bar{B} = \begin{bmatrix} \mathbf{0}_3 \\ B_{\zeta} \end{bmatrix}$$

avec:

$$B_{\zeta} = \begin{bmatrix} 0 & g & 0 \\ -g & 0 & 0 \\ 0 & 0 & m^{-1} \end{bmatrix}$$

1. Etablir la boucle fermée

- Dynamique en position / vitesse

$$\dot{\eta} = \bar{A}\eta + \bar{B}(u_{\zeta} - B_{\zeta}^{-1}\dot{v}_{ref} + B_{\zeta}^{-1}\delta_t)$$

- Commande corrigée

$$u_{\zeta} = u_{\zeta}^* + u_{\zeta}^{RDN}$$

- Boucle fermée

$$\dot{\eta} = (\bar{A} - \bar{B}K_{\eta})\eta + \bar{B}(u_{\zeta}^{RDN} + B_{\zeta}^{-1}\delta_t)$$

Comportement
linéaire désiré

Comportement
non linéaire



δ_t = terme d'erreur de dynamique entre le comportement linéaire désiré et le comportement obtenu

2. Problème de minimisation

Pour obtenir le comportement désiré :

$$\min_{u_{\zeta}^{RDN}} \|u_{\zeta}^{RDN} + B_{\zeta}^{-1}\delta_t\|^2$$

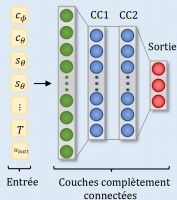
Solution de type : $u_{\zeta}^{RDN} = f(\delta_t)$

3. Connaître le terme d'erreur

Estimer δ_t ?

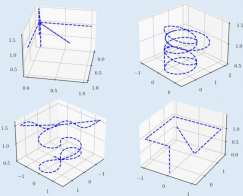
- choix de l'apprentissage

Réseau de neurones profonds



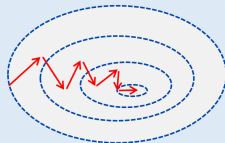
- Architecture multi-couches
- 12 entrées
- 2 couches cachées
- 64 unités par couche
- 3 sorties
- Activation = ReLU

Base de données



- Suivi dans l'espace : échelons, rampes, cercles, spirales, ...
- Plusieurs batteries (décharge)
- Environ 1h de données

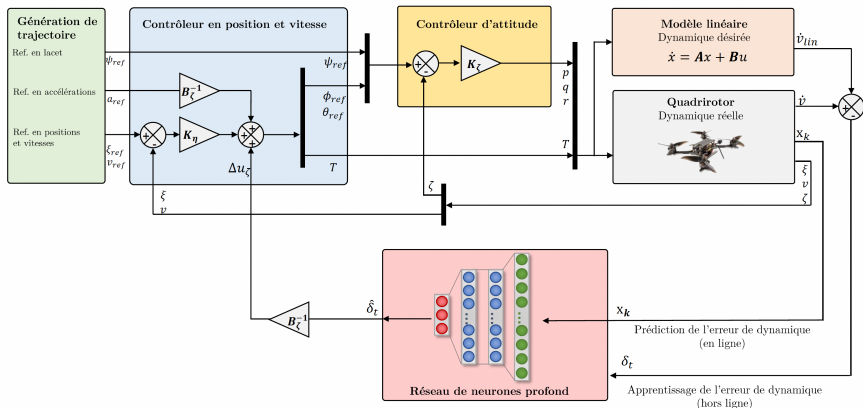
Apprentissage



- Optimisation type descente de gradient
- Algorithme NADAM
- Décroissance du pas d'apprentissage
- Ensemble de validation
- Choix du RDN

Contrôle et Apprentissage

Schéma de l'architecture de contrôle complète

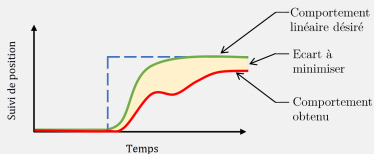


On souhaite valider expérimentalement l'approche

Quelle métrique pour l'évaluation ?

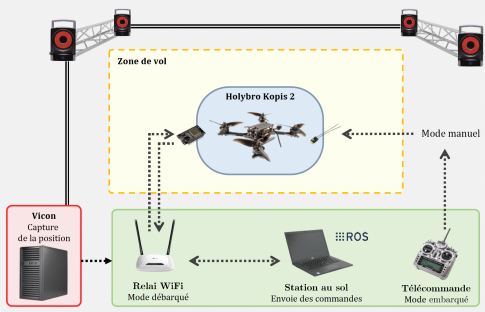
- ▶ Les erreurs quadratiques moyennes (MSE)
... au comportement linéaire désiré.

$$MSE = \frac{1}{N} \sum_{i=1}^N \|x^i - x_{lin}^i\|^2$$



- #### Quelles expériences ?
- Valider l'apprentissage d'erreurs liés à la modélisation, linéarisation, etc.
 - ▶ scénario d'échelons
 - Valider l'apprentissage d'erreurs liés à des perturbations externes
 - ▶ scénario proche du sol
 - ▶ scénario de vent

Quelle plateforme ?



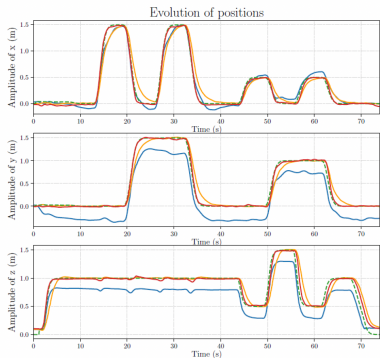
La salle de capture de mouvement du GIPSA-lab



Station au sol



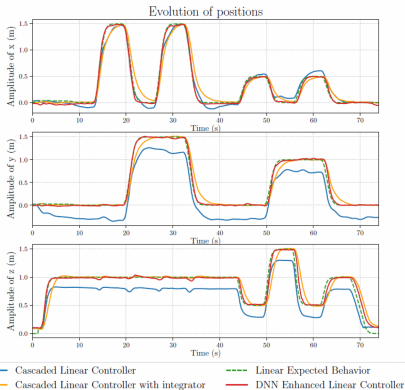
Zone de vols



Résultats expérimentaux - scénario test non vu
 = réponses à des échelons de références sur les 3 axes

	Contrôleur linéaire (sans intégrateur)	Contrôleur linéaire (avec intégrateur)	Contrôleur amélioré (RDN)
MSE axe x	0.0060 m ²	0.0150 m ²	0.0016 m ²
MSE axe y	0.0815 m ²	0.0125 m ²	0.0017 m ²
MSE axe z	0.0380 m ²	0.0198 m ²	0.0046 m ²

Table des erreurs quadratiques moyennes (MSE)
 Entre la réponse obtenue et la réponse attendue



Améliorations

- ✓ Correction des erreurs statiques
- ✓ Meilleure correspondance avec la dynamique linéaire désirée (temps de réponses et pentes)
- ✓ Apprentissage et compensation de la décharge de batterie

Effets non compensés

- ✗ Apparition de légères oscillations
- ✗ Couplages résistants
- ✗ Atterrissage et décollage absents de la base de données

Contrôle et Apprentissage

Vidéo



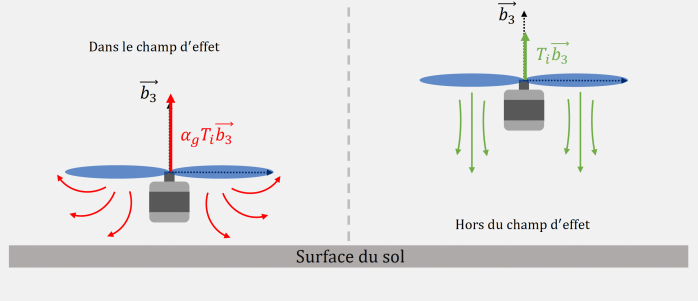
Contrôle et Apprentissage

Vidéo



Qu'est ce que l'effet de sol ?

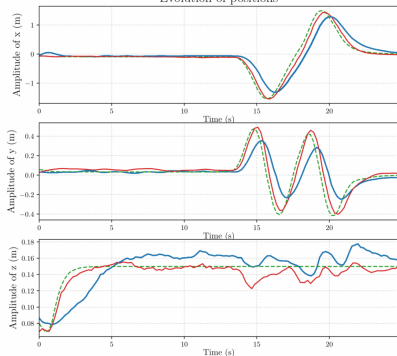
= Poussée amplifiée à proximité du sol



Contrôle et Apprentissage

Réponse à l'effet de sol

Evolution of positions



— Cascaded Linear Controller with integrator
 — DNN Enhanced Linear Controller
- - - Linear Expected Behavior

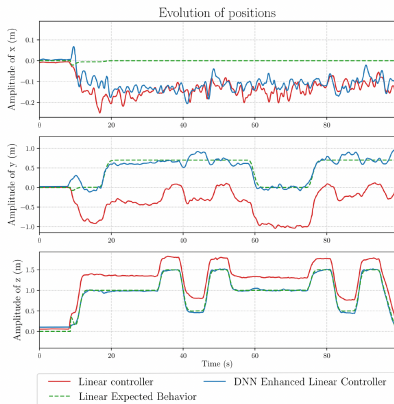
Améliorations

- ✓ Amélioration des erreurs statiques
- ✓ Meilleure correspondance avec la dynamique linéaire désirée (temps de réponses et pentes)
- ✓ Amélioration du comportement proche du sol
→ Intégration de z dans l'entrée du réseau de neurones

Mais ...

- ✗ Comportement améliorable

→ Augmenter la proportion de scénario proche du sol dans la base de données



Résultats expérimentaux - scénario de vent
= réponses à des échelons face au vent (direction y)

Améliorations

- ✓ Apprentissage du vent constant
- ✓ Meilleure correspondance avec la dynamique linéaire désirée (temps de réponses et pentes)

Robustesse à l'amplitude du vent

Attention

- ✗ Nécessité d'augmenter la base de données avec plusieurs plages d'intensité de vent

On souhaite répondre aux deux limites suivantes :

- l'apprentissage hors-ligne
- les garanties de stabilité

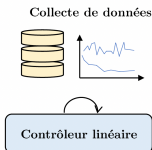
Approche et méthodologie

Apprentissage du réseau de neurones « **event-based** »

Evènements définis par deux critères : **stabilité** et **performances**

On souhaite répondre aux deux limites suivantes :

- l'apprentissage hors-ligne
- les garanties de stabilité



Approche et méthodologie

Apprentissage du réseau de neurones « **event-based** »

Evènements définis par deux critères : **stabilité** et **performances**

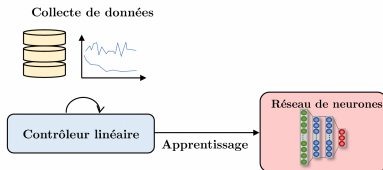
On souhaite répondre aux deux limites suivantes :

- l'apprentissage hors-ligne
- les garanties de stabilité

Approche et méthodologie

Apprentissage du réseau de neurones « **event-based** »

Evènements définis par deux critères : **stabilité** et **performances**



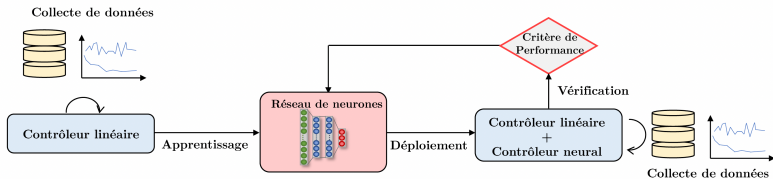
On souhaite répondre aux deux limites suivantes :

- l'apprentissage hors-ligne
- les garanties de stabilité

Approche et méthodologie

Apprentissage du réseau de neurones « event-based »

Evènements définis par deux critères : **stabilité** et **performances**



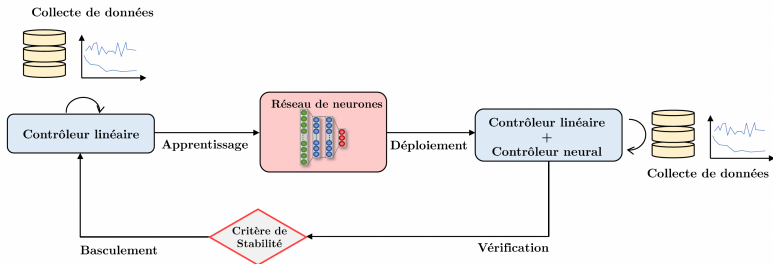
On souhaite répondre aux deux limites suivantes :

- l'apprentissage hors-ligne
- les garanties de stabilité

Approche et méthodologie

Apprentissage du réseau de neurones « event-based »

Evènements définis par deux critères : **stabilité** et **performances**



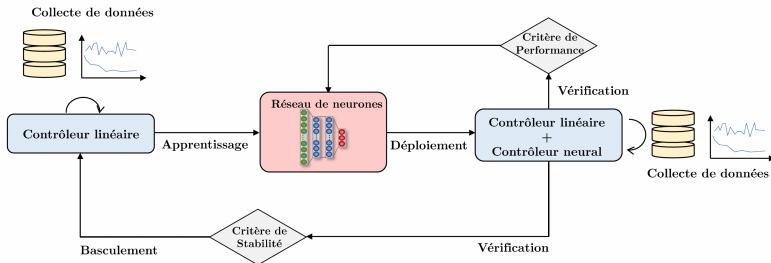
On souhaite répondre aux deux limites suivantes :

- l'apprentissage hors-ligne
- les garanties de stabilité

Approche et méthodologie

Apprentissage du réseau de neurones « event-based »

Evènements définis par deux critères : **stabilité** et **performances**



Contrôle et Apprentissage en ligne

Critère de stabilité

Que se passe-t-il si le RDN reçoit une donnée non vue, non apprise ?

- **Etape n°1:** Evaluer la stabilité du système bouclé avec le réseau.

Fonction de Lyapunov :

$$\begin{cases} \mathcal{V} := \eta^T \mathbf{P} \eta \\ \dot{\mathcal{V}} := -\alpha \omega^T \omega + 2\omega P^2 \frac{1}{2} \bar{B} B_{\zeta}^{-1} \varepsilon_{\eta} \end{cases}$$

- **Etape n°2:** Etablir le critère de stabilité, seuil sur l'erreur.

$$\|\varepsilon_{\eta}\| < f(\mathcal{V}) = \varepsilon_m$$

Solution: Si le critère est franchi, on bascule sur le contrôleur initial.

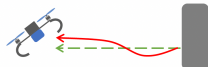


ε_{η} : erreur de prédiction

Contrôle et Apprentissage en ligne

Critère de performance

Que faire si le RDN ne corrige pas correctement le système, sans le déstabiliser ?



- **Etape n°1:** Quantifier la dégradation.

Erreur au comportement linéaire désiré :

$$\Sigma_{\eta} = \int \|\eta - \eta_{lin}\|^2$$

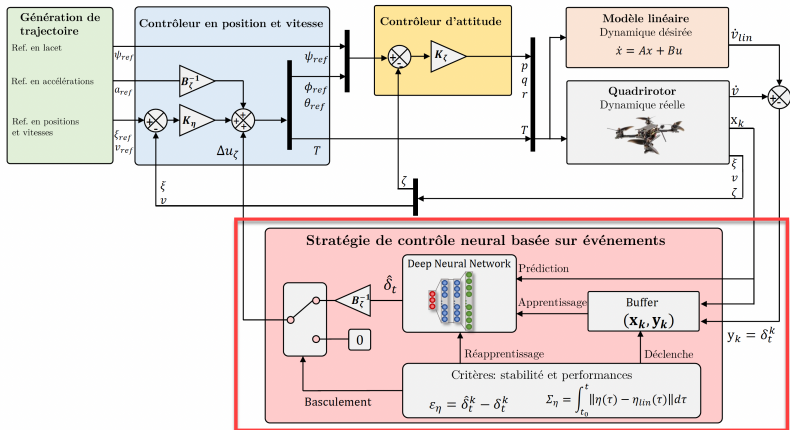
- **Etape n°2:** Etablir le critère de performance.

$$\Sigma_{\eta} > \Sigma_m$$

Solution: Si le critère est atteint et que l'on a collecté suffisamment de données, on déclenche un nouvel apprentissage à partir du précédent RDN.

Contrôle et Apprentissage en ligne

Schéma de l'architecture de contrôle complet





On souhaite valider en simulation les critères proposés

Quelles expériences ?

■ Valider le critère de performances

▶ scénario de cercles dans le plan x/y puis cercles rayon plus élevée dans le même plan

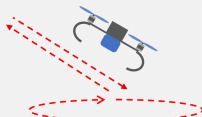
■ Valider le critère de stabilité

▶ scénario de cercles dans le plan x/y puis des échelons (step) dans le plan y/z

1



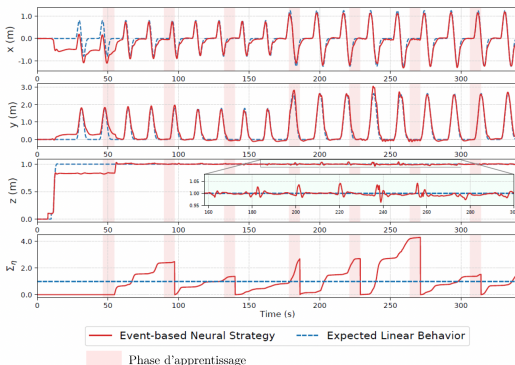
2



Contrôle et Apprentissage en ligne

Validation en simulation : critère de performance

Résultat en simulation (GAZEBO) – test du critère de performances
= réponse à des cercles dans le plan x/y



Analyses

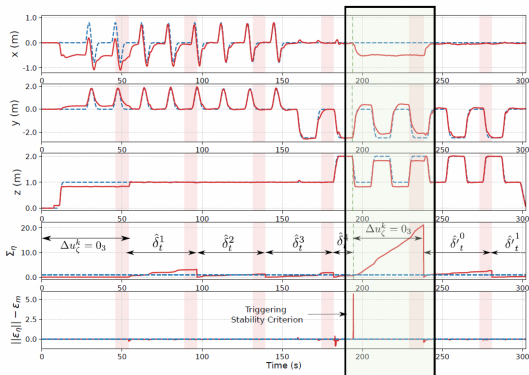
- Succession de phases d'apprentissage et de collecte
- Amélioration dès le 1^{er} apprentissage
- Amélioration progressive du comportement (/ comportement linéaire désiré)

◀ Evolution du critère de performances

Contrôle et Apprentissage en ligne

Validation en simulation : critère de stabilité

Résultat en simulation (GAZEBO) – test du critère de stabilité
= réponse à des cercles dans le plan x/y puis échelons sur y/z



Analyses

- Confirmation analyses précédentes
- Comportement « nouveau » non correctement appris basculement du contrôleur

◀ Evolution du critère de performances

◀ Evolution du critère de stabilité

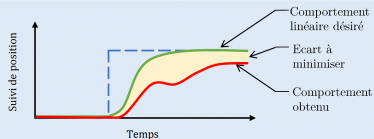


On souhaite valider expérimentalement l'approche proposée

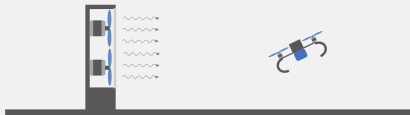
Quel métrique pour l'évaluation ?

► RMSE au comportement linéaire désiré

$$RMSE = \sqrt{MSE}$$



Quel expérience ? Des allers-retours face à un vent non constant = sinusoïde variant en amplitude et en moyenne



Contrôle et Apprentissage en ligne

Vidéo



MOBROB2024

Mobile robotic systems: from modelling and control to future challenges

45th International Summer School of Automatic Control
Grenoble, France

September 2 - September 6, 2024

MOBROB2024



<http://gipsacontrols Summerschool.org>

Mobile robotic systems: from modelling and control to future challenges

September 2 - September 6, 2024

- The scientific objective of this summer school is to establish the **fundamentals of mobile robotics**, to provide the modeling and control tools needed to meet the challenges of these complex systems, and to identify new areas of **research and applications** in this rapidly evolving field.
 - Scientific Chairs: Lara BRIÑÓN ARRANZ & Ahmad HABLY GIPSA-lab, Grenoble INP, Grenoble, France
 - Local Organizers: Mirko FIACCHINI & Matteo TACCHI GIPSA-lab, CNRS, Grenoble, France

Confirmed lecturers:

Barbara BAZZANA Univ. of Twente, Netherlands

Isabelle FANTONI CNRS, LS2N, Nantes, France

Antonio FRANCHI Univ. of Twente, Netherlands

Fabio RUGGERO Univ. of Naples Federico II, Italy

Martin SASKA Czech Technical University, Czech Republic

Bruno SICILIANO Univ. of Naples Federico II, Italy

Marco TOGNON INRIA Rennes-Bretagne Atlantique, France