

ControlSysLab - Projet Systèmes Asservis

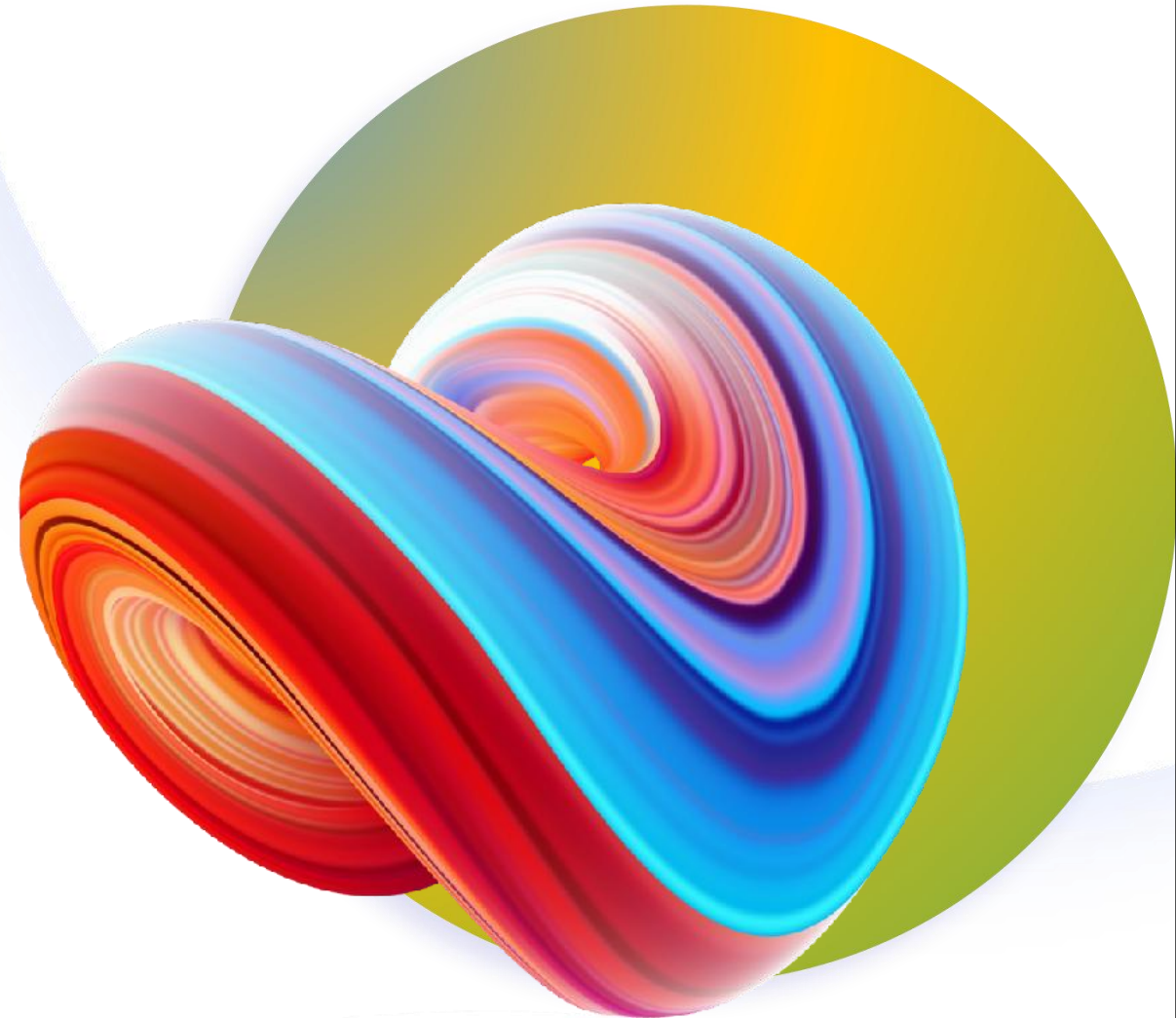
# Application interactive pour l'analyse et la commande de systèmes

Par : BADIBANGA BADIBANGA 1ICE EE

DZAPILI MITANO 1ICE EE

LOOTA BETOKO 1ICE EE

MARHEGANE BITATI 1ICE IN



# Plan

- **Introduction**
- **Objectifs spécifiques**
- **Technologies utilisées**
- **Fonctionnalités**
- **Images d'illustration**
- **Perspectives**

# Introduction

---

- **Contexte : besoin d'un outil pédagogique pour l'analyse et la conception de systèmes de contrôle.**
- **Objectif général : combiner théorie et pratique via une application interactive.**

# Objectifs spécifiques

---

- Analyse d'un système linéaire à partir de son équation d'état.
- Vérification de la contrôlabilité et observabilité.
- Commande par retour d'état et retour de sortie.
- Étude de systèmes non linéaires.
- Synthèse de régulateurs PID.

# Technologies utilisées

## Architecture MVC

- Backend : Python (NumPy, SciPy, SymPy, Control)
- Interface graphique : PyQt5
- Simulation : Matplotlib, Plotly.
- Exportation : PDF / CSV des résultats.

# Fonctionnalités

# Analyse d'État

---

**Entrée : matrices (A, B, C, D).**

- **Sorties : pôles, zéros, stabilité.**
- **Visualisation : réponses temporelles, diagrammes de Bode.**
- **(Ajouter une illustration de graphe ou schéma du manuel)**

# **Contrôlabilité & Observabilité**

**Calcul des matrices  $W_c$  et  $W_o$ .**

**Vérification automatique (contrôlable / observable).**

**Visualisation des états accessibles.**

**Exemple :  $\text{rang}(W_c) = \text{rang}(W_o) = 2 \rightarrow$  système contrôlable et observable.**



# Commande d'État

---

**Placement de pôles désirés.**

**Calcul automatique du gain  $K$ .**

**Simulation boucle fermée (avec et sans contrôle).**

**Exemple : pôles choisis  $(-4, -5) \rightarrow$  gain  $K$  généré.**

# Commande de Sortie

---

**Observateur de Luenberger.**

**Mesure partielle ou complète.**

**Comparaison boucle complète vs partielle.**

**Exemple :  $y = x_1$ , pôles observateur =  $(-6, -7)$ .**

# **Systemes Non Linéaires**

---

**Éditeur d'équations différentielles.**

**Linéarisation automatique (Jacobienne).**

**Méthode de Lyapunov.**

**Affichage : portraits de phase, surfaces de stabilité.**

# Régulateurs PID

---

**Réglages manuels et automatiques (Ziegler-Nichols, Cohen-Coon).**

**Comparaison des performances (réponses temporelles, Nyquist).**

**Exemple :  $G(s)$  défini  $\rightarrow$  PID simulé avec gain ajusté.**

# Résultats du Design

## ControlSysLab

## Analyse d'État

Contrôlabilité/Observab

Commande d'État

Commande de Sortie

Systèmes Non-Linéaires

Régulateurs PID

Thème Clair

## Analyse d'État

## Définition du Système

Nombre d'états (n): 2 Entrées (m): 1 Sorties (p): 1

 $\dot{x}_2$  -2.0... -3.0... $\dot{x}_2$  1.0000

## C (p×n) - Matrice de Sortie

	x1	x2
y1	1.0000	0.0000

## D (p×m) - Transmission Directe

	u1
y1	0.0000

Générer Aléatoire

Exemples

Importer CSV

Effacer

Valider

## Paramètres de Simulation

Temps final (s) 10.00

Pas d'échantillonnage 0.010

## Type d'Entrée

Échelon unitaire

Analyser le Système

ANALYSE DU SYSTÈME

Description...

Nouveau

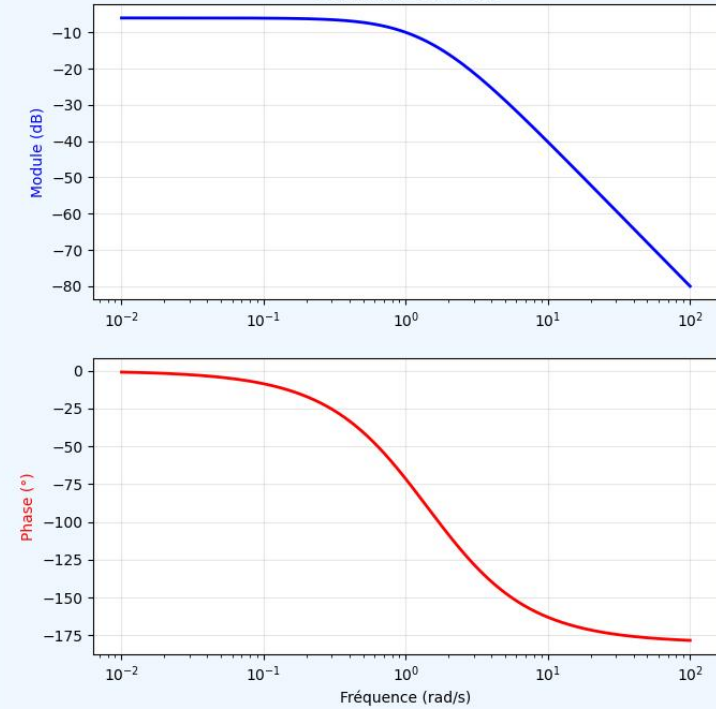
Ouvrir

Enregistrer

Exporter

Capturer

## Diagramme de Bode



Effacer

Sauvegarder

Grille

Réponse Temporelle

Diagramme de Bode

Pôles/Zéros

Réponse Impulsionnelle

### Type d'Entrée

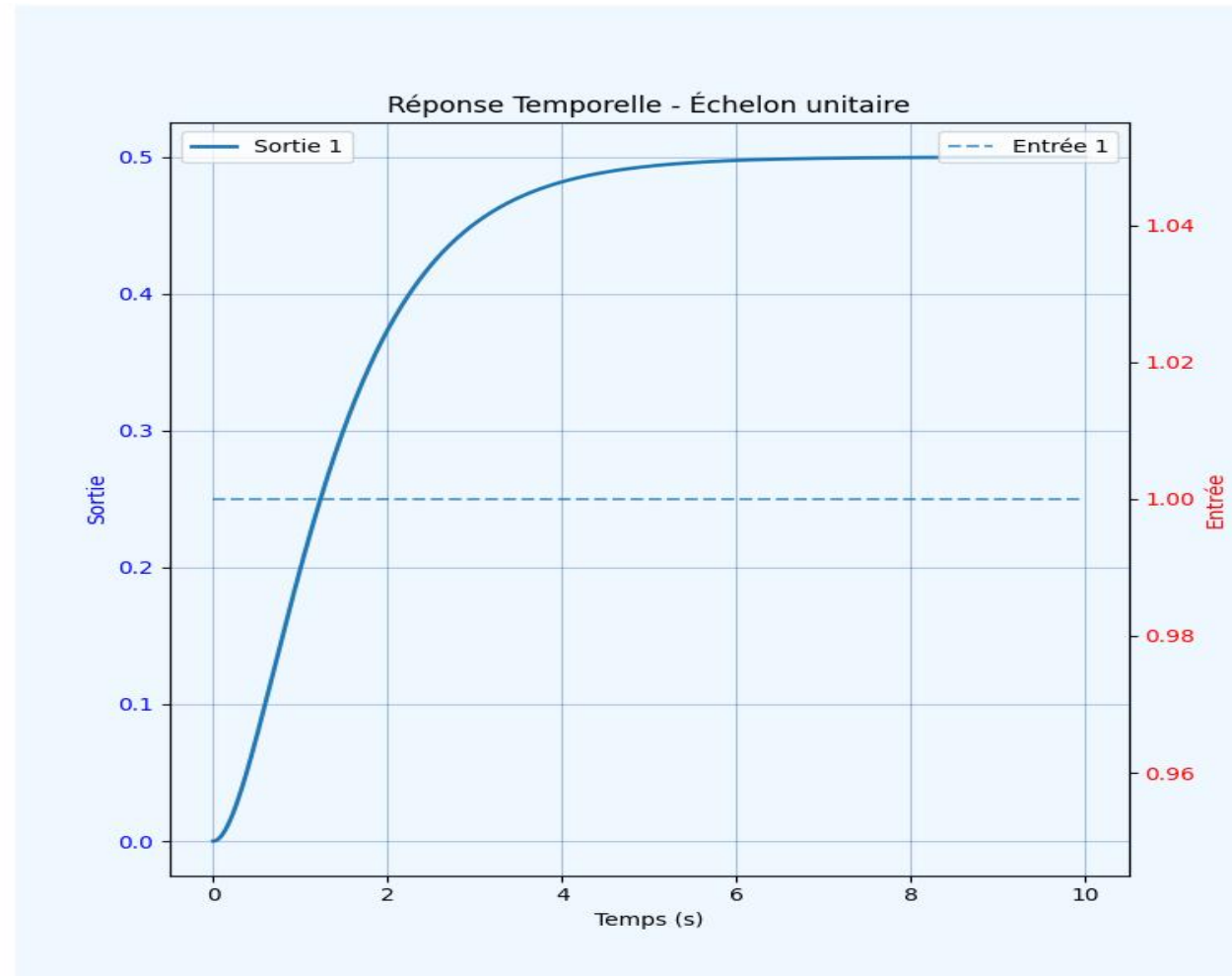
Échelon unitaire

Analyser le Système

Poles du systeme:

-1.0000

-2.0000



 Effacer

 Sauvegarder

 Grille

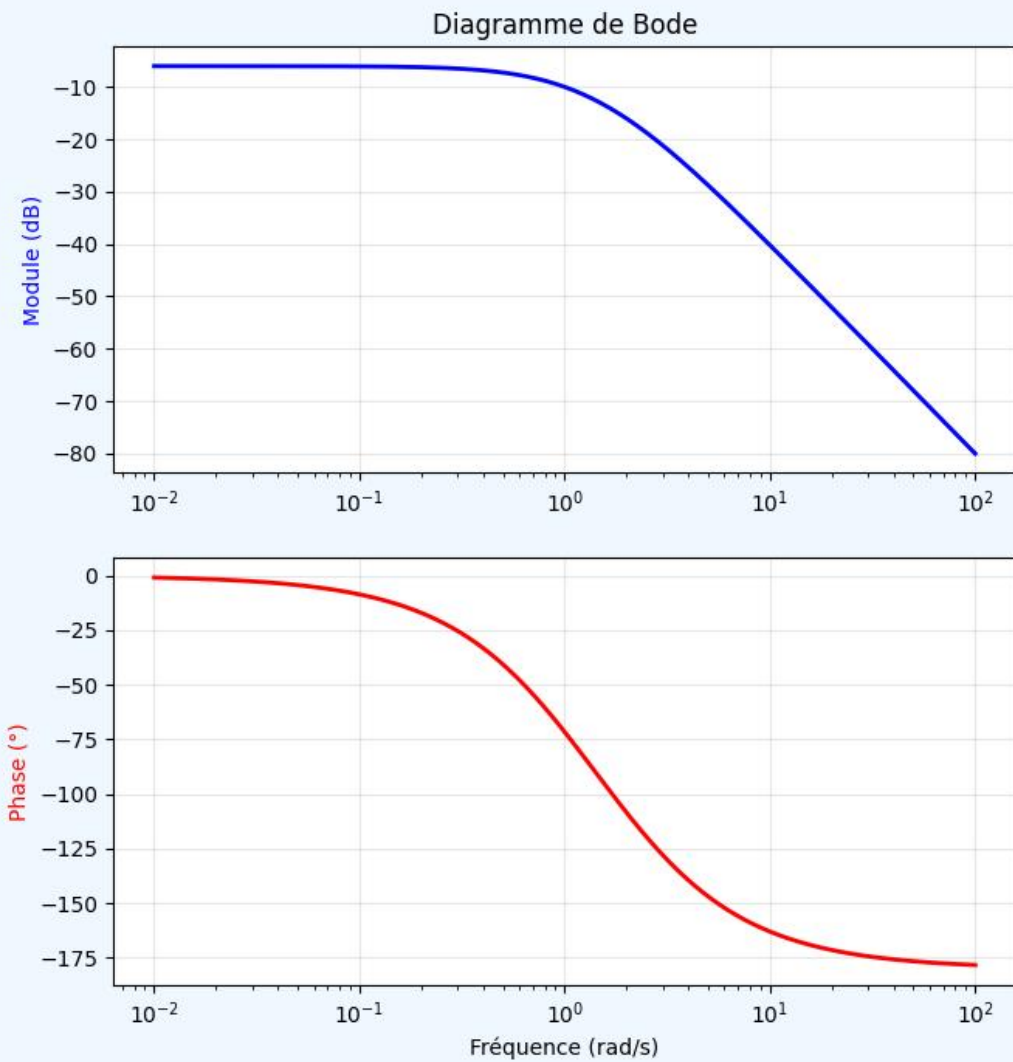
Réponse Temporelle

Diagramme de Bode

Pôles/Zéros

Réponse Impulsionnelle





Effacer

Sauvegarder

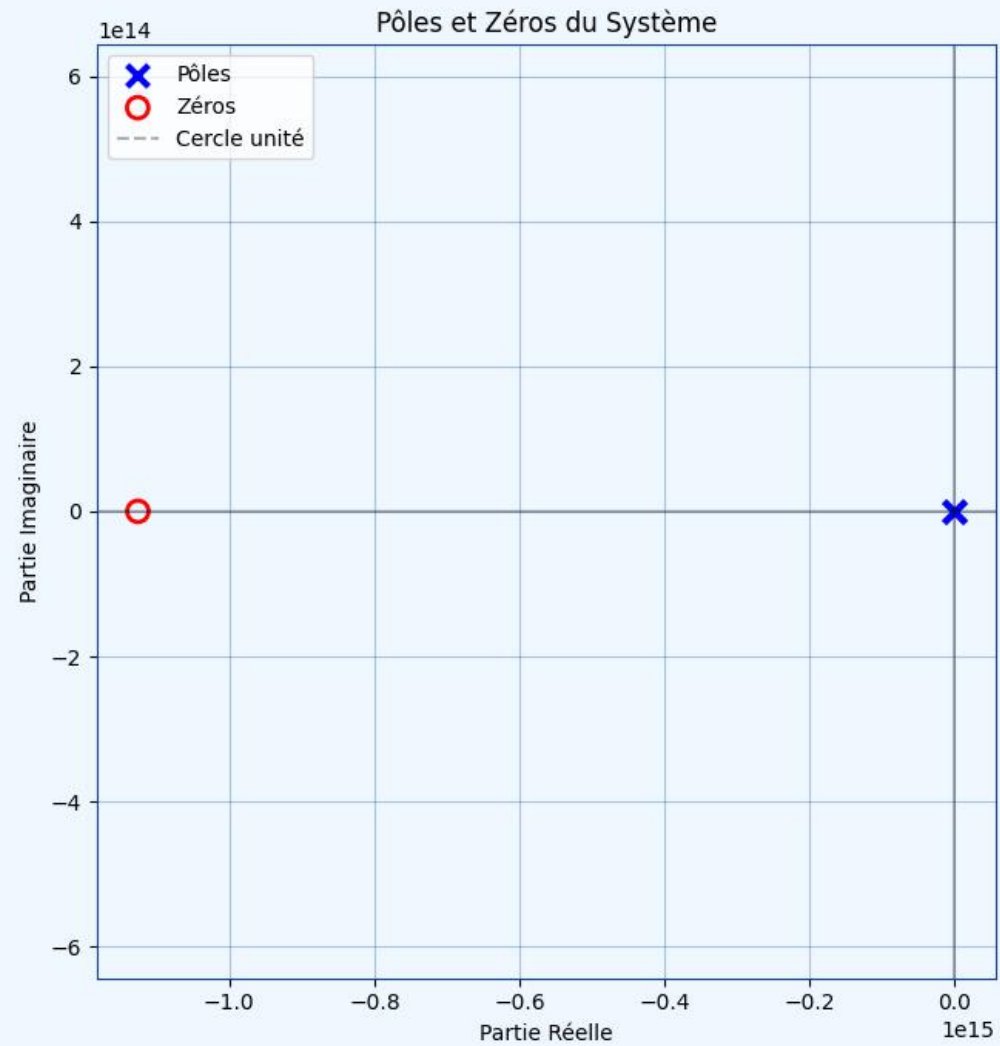
Grille

Réponse Temporelle

Diagramme de Bode

Pôles/Zéros

Réponse Impulsionnelle



Effacer

Sauvegarder

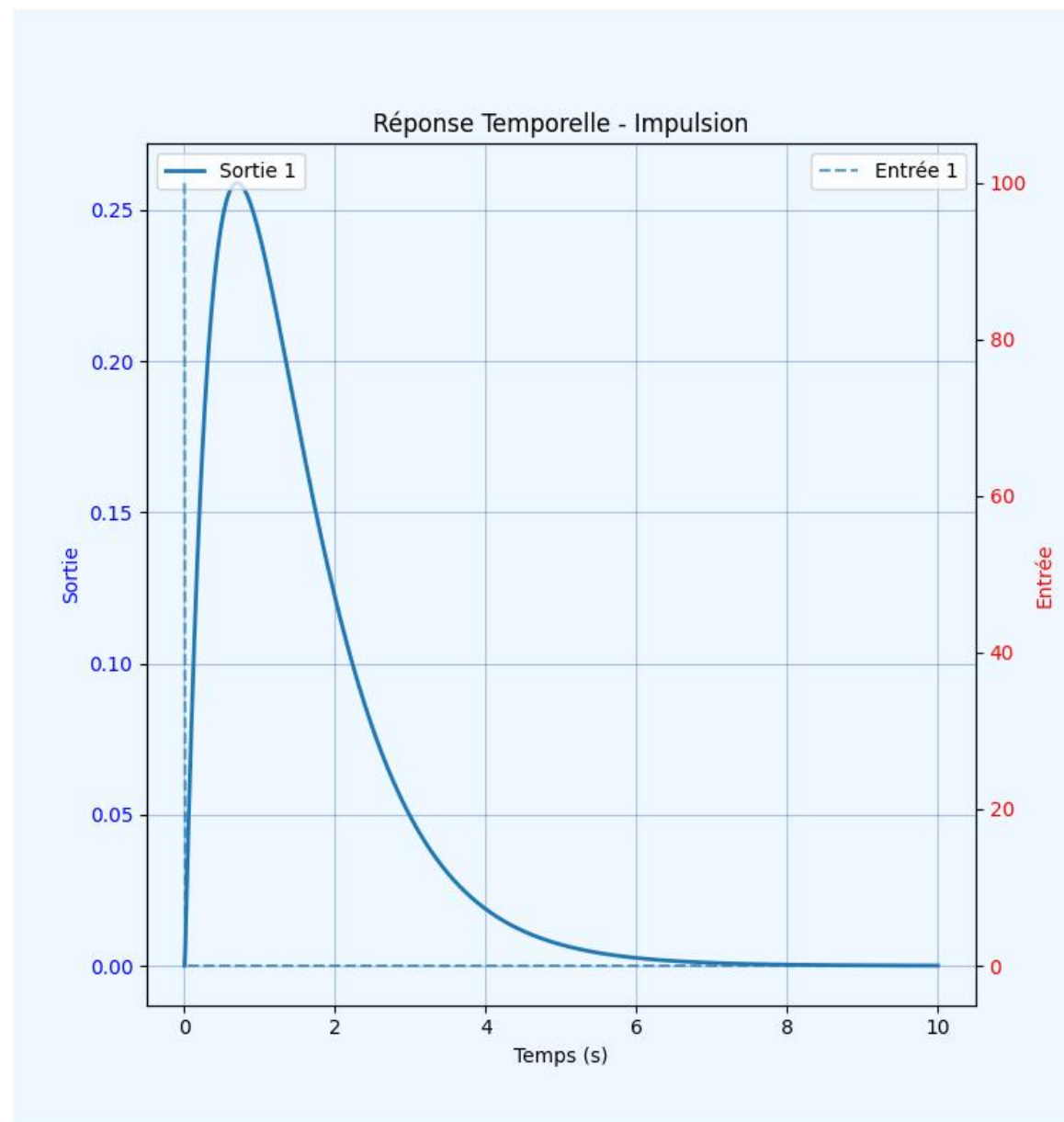
Grille

Réponse Temporelle

Diagramme de Bode

Pôles/Zéros

Réponse Impulsionnelle



 Effacer

 Sauvegarder

 Grille

Réponse Temporelle

Diagramme de Bode

Pôles/Zéros

Réponse Impulsionnelle

## ControlSysLab

Analyse d'État

Contrôlabilité/Observabilité

Commande d'État

Commande de Sortie

Systèmes Non-Linéaires

Régulateurs PID

Thème Clair

## Contrôlabilité/Observabilité

Nouveau

Ouvrir

Enregistrer

Exporter

Capturer

## Système à Analyser

Nombre d'états (n):

2

Entrées (m):

1

Sorties (p):

1

## A (n×n) - Matrice d'État

	x1	x2
x1	-0.1...	-0.0...
x2	-0.3...	-0.4...

## B (n×m) - Matrice d'Entrée

	u1
x1	0.6720
x2	0.9446

## C (p×n) - Matrice de Sortie

|--|--|

## D (p×m) - Transmission Directe

|--|--|

Générer Aléatoirement

Exemples

Importer CSV

Effacer

Valider

## Analyse

Analyser Contrôlabilité

Analyser Observabilité

Analyse Complète

Conditionnement: 5.69e+00

Effacer

Sauvegarder

Grille

Matrice Wc

Matrice Wo

Valeurs Singulières

États Accessibles

## ControlSysLab

Analyse d'État

Contrôlabilité/Observabilité

Commande d'État

Commande de Sortie

Systèmes Non-Linéaires

Régulateurs PID

## Contrôlabilité/Observabilité

Nouveau

Ouvrir

Enregistrer

Exporter

Capturer

## Système à Analyser

Nombre d'états (n):

2

Entrées (m):

1

Sorties (p):

1

## A (n×n) - Matrice d'État

	x1	x2
x1	0.0000	1.0000
x2	-2.0...	-3.0...

## B (n×m) - Matrice d'Entrée

	u1
x1	0.0000
x2	1. ...

## C (p×n) - Matrice de Sortie

Générer Aléatoirement	Exemples

## D (p×m) - Transmission Directe

Importer CSV	Effacer
Valider	

## Analyse

Analyser Contrôlabilité

Analyser Observabilité

Analyse Complète

Thème Clair

Interprétation:

Le système est complètement contrôlable. Tous les états peuvent être atteints.

Effacer

Sauvegarder

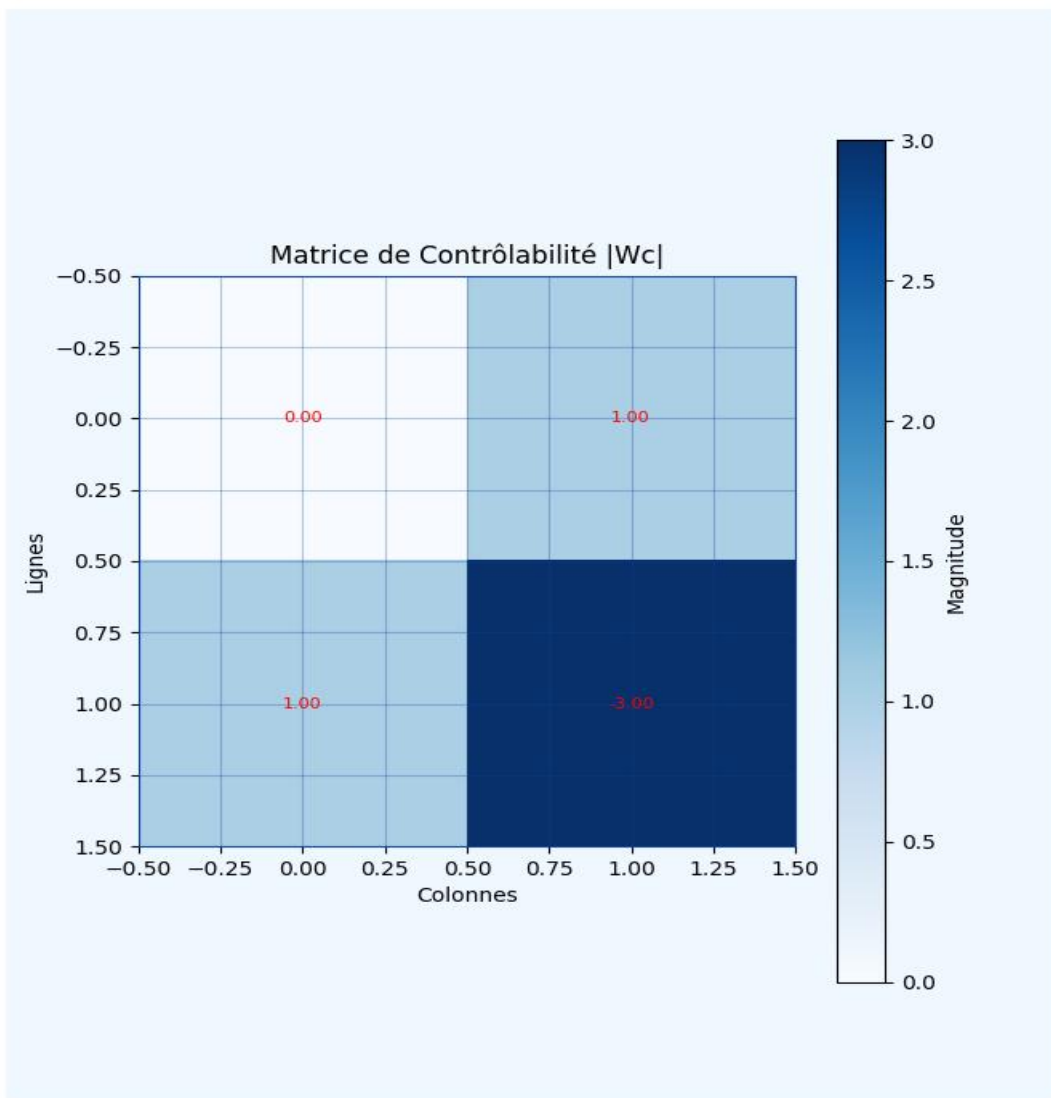
Grille

Matrice Wc

Matrice Wo

Valeurs Singulières

États Accessibles



Effacer

Sauvegarder

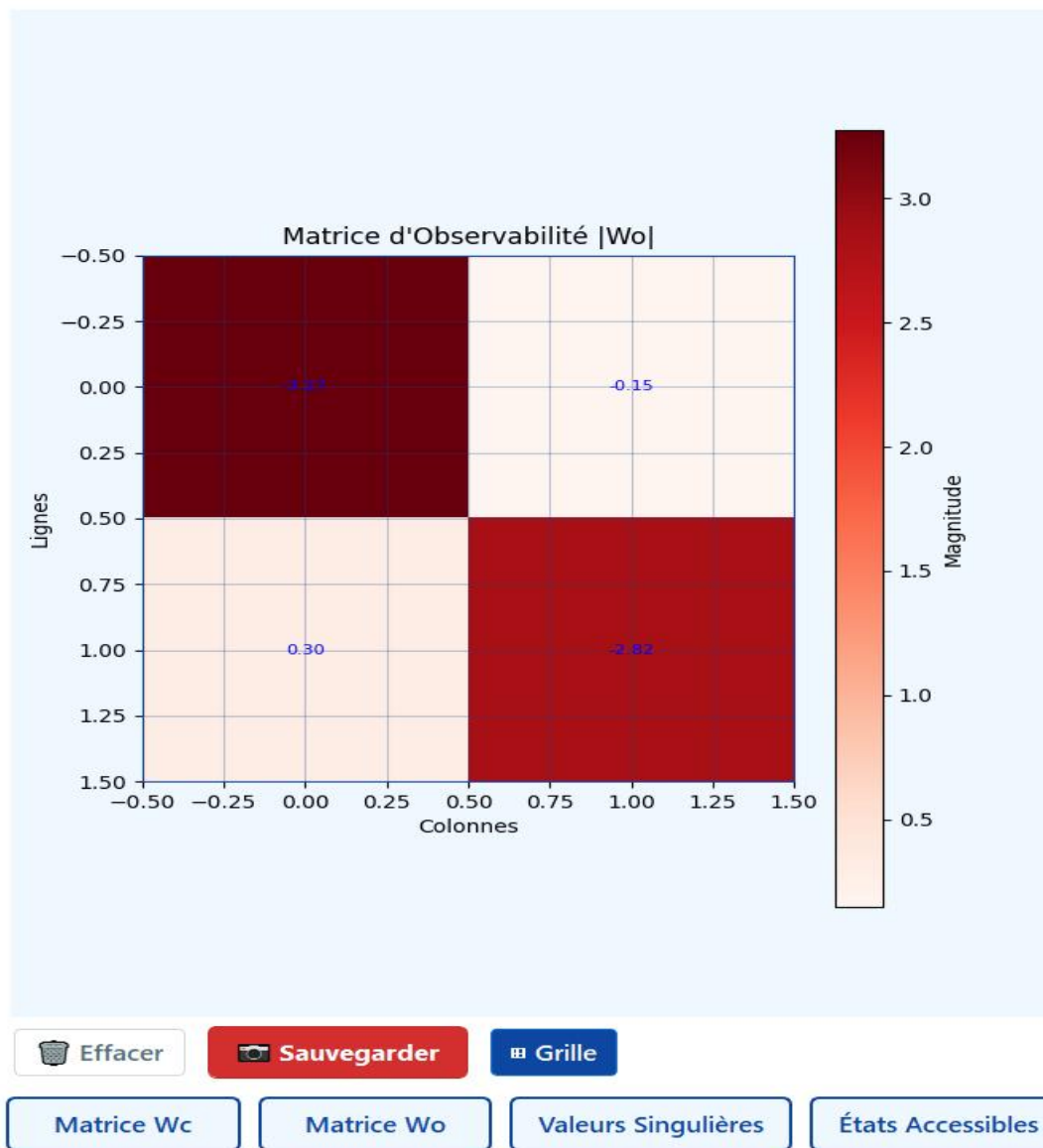
Grille

Matrice  $W_c$

Matrice  $W_o$

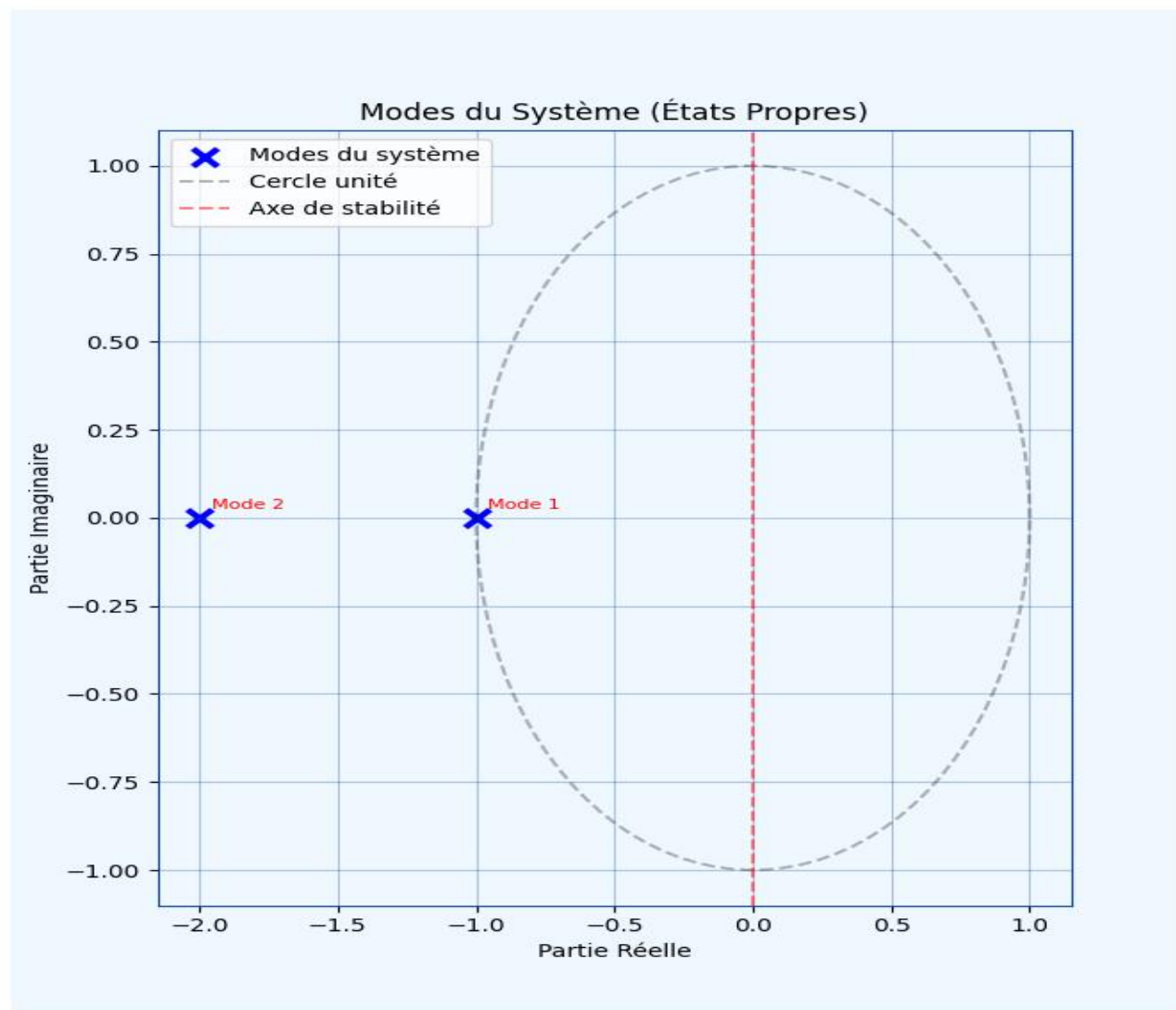
Valeurs Singulières

États Accessibles









Effacer

Sauvegarder

Grille

Matrice  $W_c$

Matrice  $W_o$

Valeurs Singulières

États Accessibles

## Commande d'État

### Système à Contrôler

Nombre d'états (n):  Entrées (m):  Sorties (p):

$\dot{x}_2$  0.6719 0.3997

$\dot{x}_2$  0.5174

Générer Aléatoirement

Exemples

Importer CSV

Effacer

Valider

$C (p \times n)$  - Matrice de Sortie

$D (p \times m)$  - Transmission Directe

### Spécifications de Performance

Méthode:

#### Spécifications Temporelles

Temps de réponse (s):  Dépassement (%):

Calculer Pôles Désirés

### Conception du Régulateur

Vérifier Contrôlabilité

Concevoir Gain K

Simuler Boucle Fermée

Paramètres calculés:

- Amortissement  $\zeta$ : 0.690
- Pulsation naturelle  $\omega_n$ : 2.898 rad/s

ControlSysLab - Analyse et Conception de Systèmes de Contrôle

ControlSysLab

Analyse d'État

Contrôlabilité/Observab

Commande d'État

Commande de Sortie

Systèmes Non-Linéaires

Régulateurs PID

Thème Clair

Commande de Sortie

Nouveau

Ouvrir

Enregistrer

Exporter

Capturer

Système à Contrôler

$\dot{x}_2$

-0.2...

0.0141

$\dot{x}_2$

-2.1...

Générer Aléatoire

Exemples

Importer CS

Effacer

Valider

Configuration de l'Observateur

Sorties mesurées: Toutes les sorties

Vitesse observateur (facteur): 5,0x plus rapide

☐ Ajouter bruit de mesure 0,010 %

Configuration du Contrôleur

Type de contrôleur: Régulateur PI basé observation

Temps de réponse (s): 2,00

Dépassement (%): 5,0

Conception

Vérifier Propriétés Système

Concevoir Observateur

Concevoir Contrôleur

Simuler Système Complet

VÉRIFICATION DES PROPRIÉTÉS DU SYSTÈME

Effacer

Sauvegarder

Grille

Réponse Complète

Erreur d'Estimation

États vs Estimés

Performance Contrôleur

04:57

## Commande de Sortie

### Système à Contrôler

$\dot{x}_2$	-0.2...	0.0141		$\dot{x}_2$	-2.1...	
Générer Aléatoire		Exemples	Importer CS'	Effacer	Valider	

### Configuration de l'Observateur

Sorties mesurées: Toutes les sorties

Vitesse observateur (facteur): 5,0× plus rapide

☐ Ajouter bruit de mesure 0,010 %

### Configuration du Contrôleur

Type de contrôleur: Régulateur PI basé observation

Temps de réponse (s): 2,00 Dépassement (%): 5,0

### Conception

Vérifier Propriétés Système

Concevoir Observateur

Concevoir Contrôleur

Simuler Système Complet

## Systèmes Non-Linéaires

### Définition du Système Non-Linéaire

Système prédéfini:

Équations différentielles (format Python/SymPy):

Point d'équilibre [x1, x2]:

Valider Système

Linéariser

Simuler

### Analyse de Linéarisation

☒ Linéarisation automatique ☐ Afficher valeurs propres

### Analyse de Stabilité de Lyapunov

Méthode:

Fonction V(x) personnalisée (optionnel):

Analyser Stabilité Lyapunov

### Paramètres de Simulation

Temps final

Pas de temps

Plage x1

Plage x2

ControlSysLab

Analyse d'État

Contrôlabilité/Observabilité

Commande d'État

Commande de Sortie

Systèmes Non-Linéaires

Régulateurs PID

Thème Clair

Régulateurs PID

Nouveau

Ouvrir

Enregistrer

Exporter

Capturer

Système à Réguler

A (n×n) - Matrice d'État

B (n×m) - Matrice d'Entrée

gérer Aléatoire

Exemples

Importer CS

Effacer

Valider

Configuration du Régulateur PID

Type de régulateur: PID standard

Réglage Manuel

Anti-windup activé Limite: 10,0

Méthodes de Réglage Automatique

Ziegler-Nichols

Cohen-Coon

Réglage Optimisé

Simuler Régulation PID

Paramètres de Simulation

Temps final

20.0 s

Amplitude consigne

1.00

Temps consigne

1.00 s

Fonction de transfert:  
 $G(s) = (0.0417s^2 + 0.2143s - 0.1849) / (s^2 - 0.8572s + 0.2615)$



Effacer

Sauvegarder

Grille

Réponse PID

Comparaison Méthodes

Signal de Commande

Analyse Fréquentielle

SIMULATION PID TERMINÉE

Configuration:

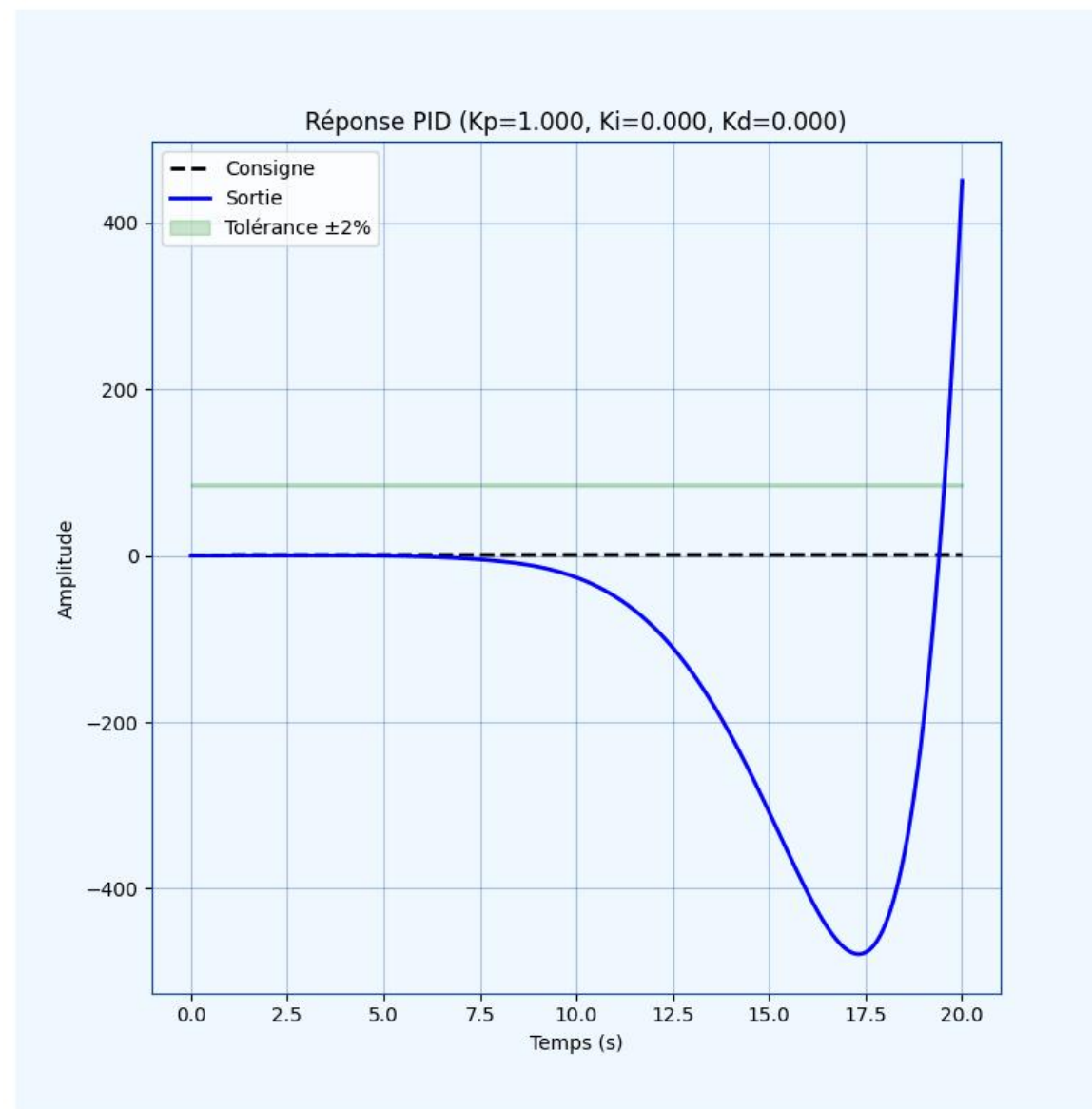
- $K_p = 1.0000$
- $K_i = 0.0000$

- $K_i = 0.0000$
- $K_d = 0.0000$
- Anti-windup: Activé

Performances obtenues:

- Temps de montée: 0.110 s
- Dépassement: 429.9 %

- Temps d'établissement: 19.550 s
- Erreur statique: 84.0594
- ISE (Intégrale erreur<sup>2</sup>): 838742.336043



Effacer

Sauvegarder

Grille

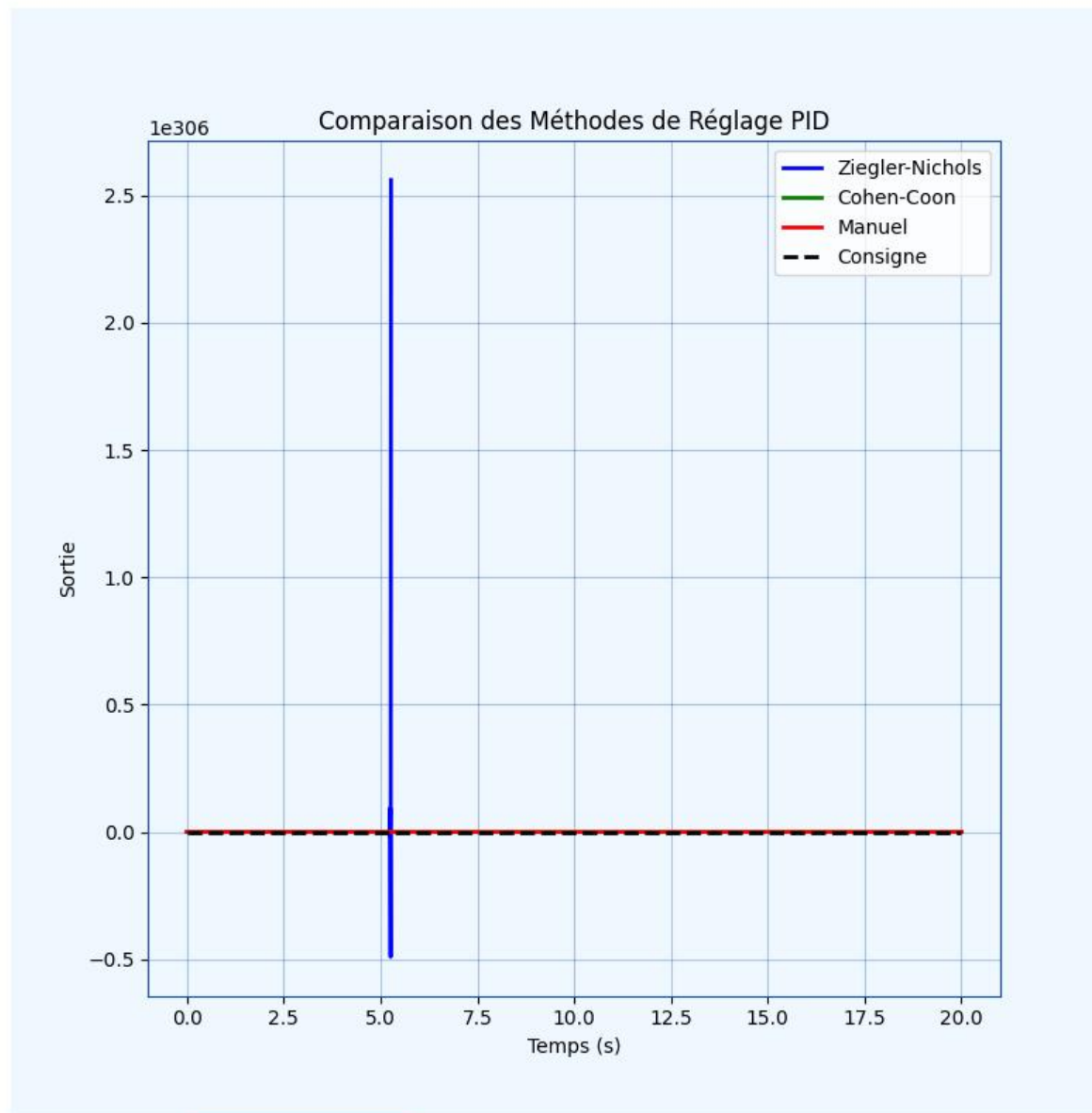
Réponse PID

Comparaison Méthodes

Signal de Commande

Analyse Fréquentielle





Effacer

Sauvegarder

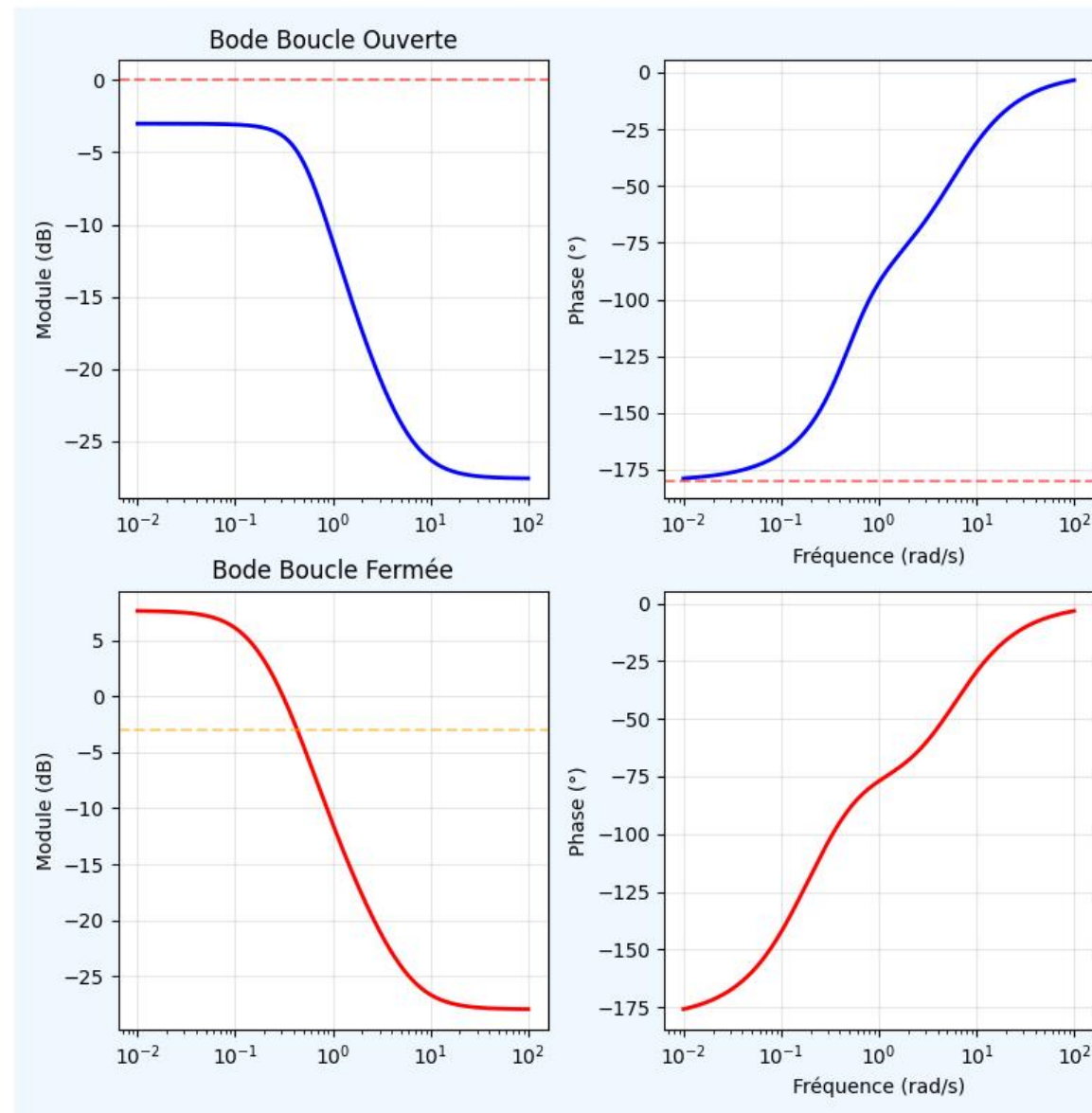
Grille

Réponse PID

Comparaison Méthodes

Signal de Commande

Analyse Fréquentielle



# Perspectives



**Implémenter une interface web (front-end web)**

**Concevoir une version app mobile qui soit simple d'utilisation**

**Déployer sur toutes les plateformes de téléchargement**

**Merci !**