

Université de Kinshasa/Faculté polytechnique
Département de Génie Électrique et informatique



Manuel ControlSysLab

Application pédagogique pour l'analyse et la
commande des systèmes dynamiques

Auteurs :

BADIBANGA BADIBANGA
DZAPILI MITANO
LOOTA BETOKO
MARHEGANE BITATI

Encadrants :

Pr.Dr.Ir. WALONGO GUY
Ass.Ir. MAZUBA LIONEL

Année académique 2024-2025

Table des matières

Introduction	2
1 Analyse d'État	3
2 Contrôlabilité et Observabilité	10
3 Commande d'État	17
4 Commande de Sortie	21
5 Systèmes Non Linéaires	24
6 Régulateurs PID	27
Conclusion	33

Introduction

ControlSysLab est une application pédagogique et pratique conçue pour l'analyse et la conception de systèmes asservis. Elle intègre plusieurs fonctionnalités allant de l'analyse d'état à la synthèse de régulateurs PID. Ce manuel enrichi explique comment utiliser l'application étape par étape, fournit des exemples visuels et présente l'architecture logicielle du projet.

Les principales fonctionnalités de ControlSysLab sont :

1. Analyse d'état
2. Contrôlabilité et Observabilité
3. Commande d'état
4. Commande de sortie
5. Systèmes non-linéaires
6. Régulateurs PID

Chapitre 1

Analyse d'État

Cette fonctionnalité permet d'analyser la dynamique interne du système. On peut étudier la réponse temporelle, les pôles, les zéros, et obtenir une vision claire de la stabilité du modèle.

Objectif : Étudier le comportement d'un système linéaire défini par (A, B, C, D) .

Outils disponibles :

- Pôles et zéros automatiques
- Réponses temporelles (impulsion, échelon)
- Diagramme de Bode
- Vérification de la stabilité

Étapes :

1. Entrer les matrices du système (utiliser les curseurs pour les matrices C et D).
2. Appuyer sur **Analyser le système** (bouton bleu).
3. Sélectionner les graphiques désirés : pôles, réponses temporelles, Bode, stabilité.
4. Vérifier la stabilité du système.

Exemple : système $A = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix}$, $B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, $C = [1 \ 0]$, $D = [0]$ \Rightarrow pôles : $-1, -2$ (stables).

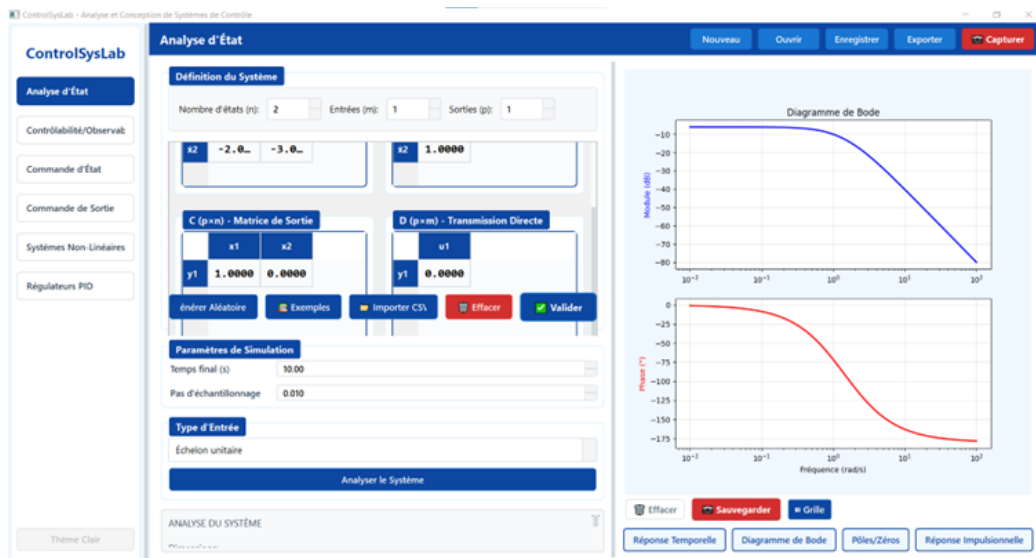


FIGURE 1.1 – fenêtre d'accueil

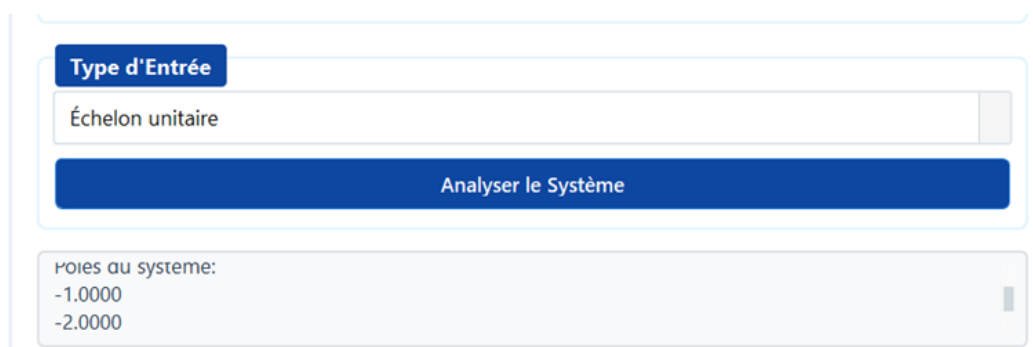


FIGURE 1.2 – pôles du système

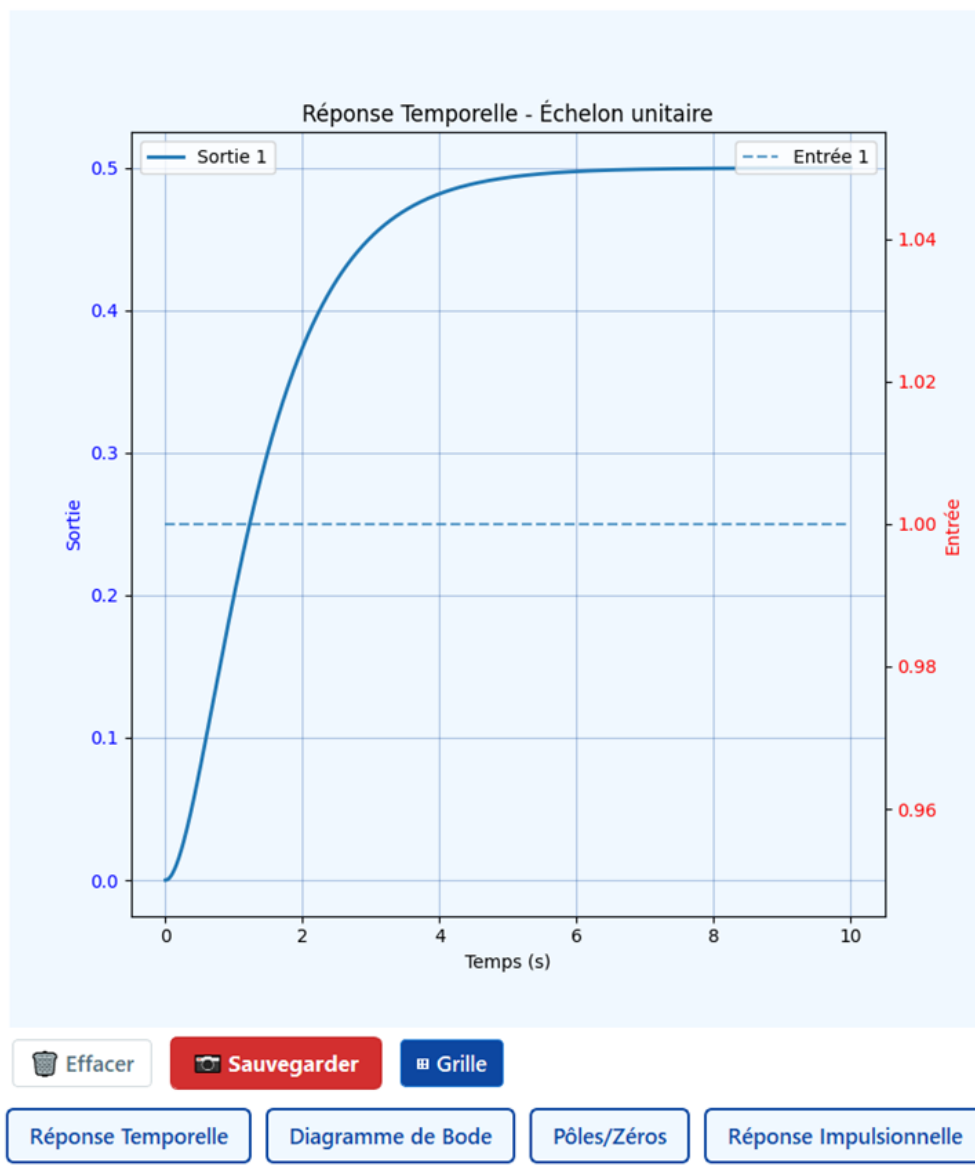


FIGURE 1.3 – Réponse temporelle- échelon unitaire

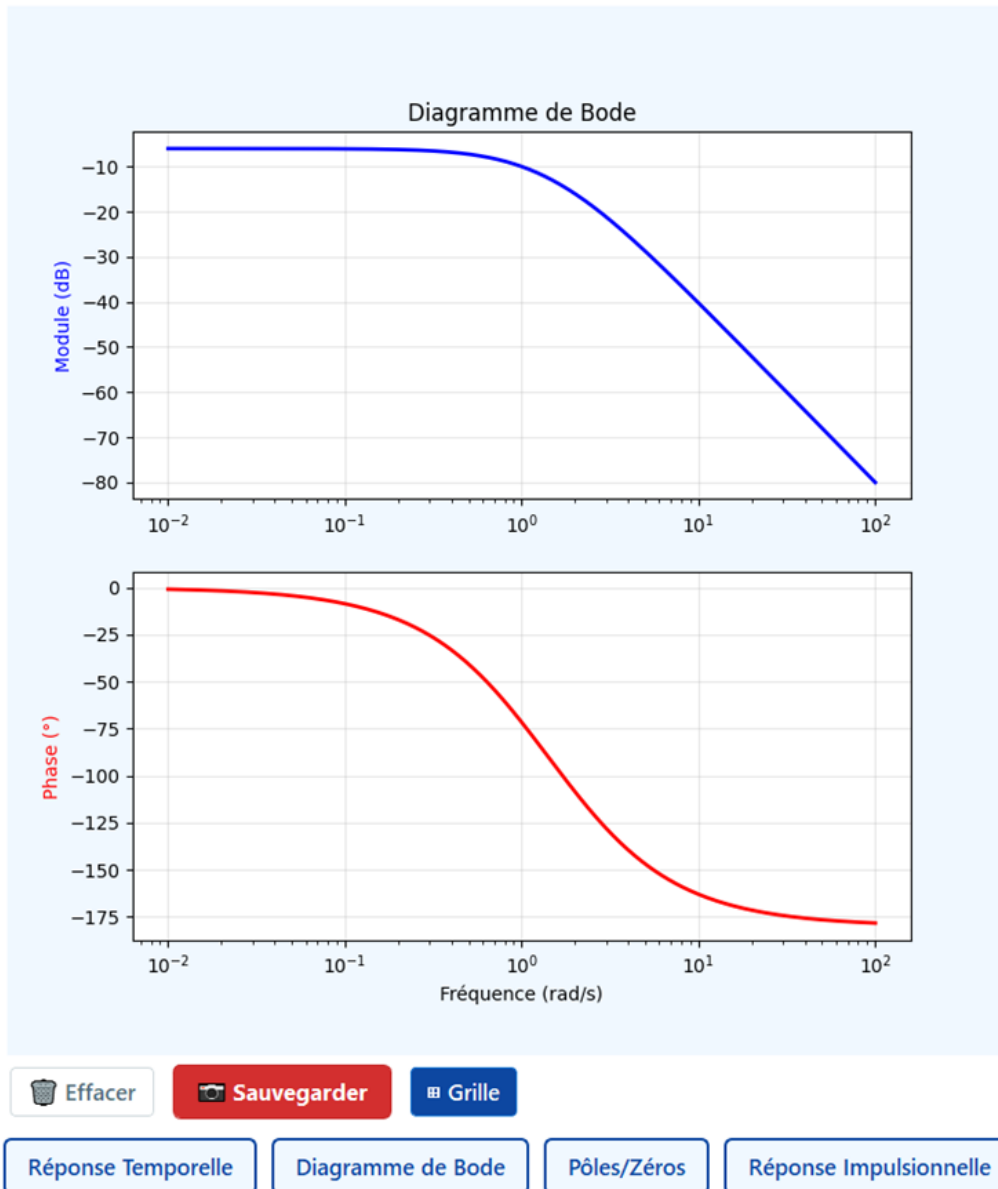


FIGURE 1.4 – Diagramme de bode

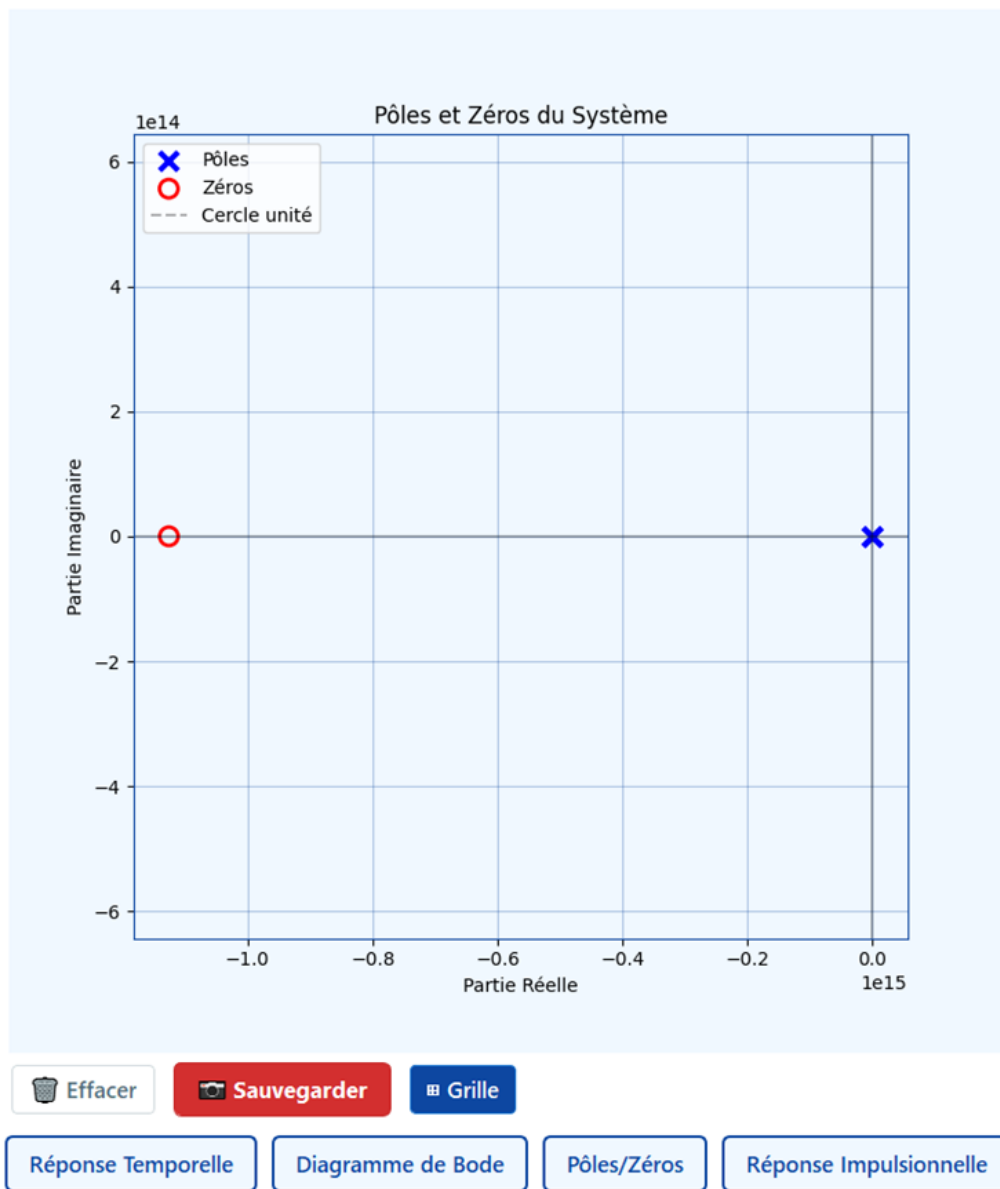


FIGURE 1.5 – Emplacement des pôles et zéros

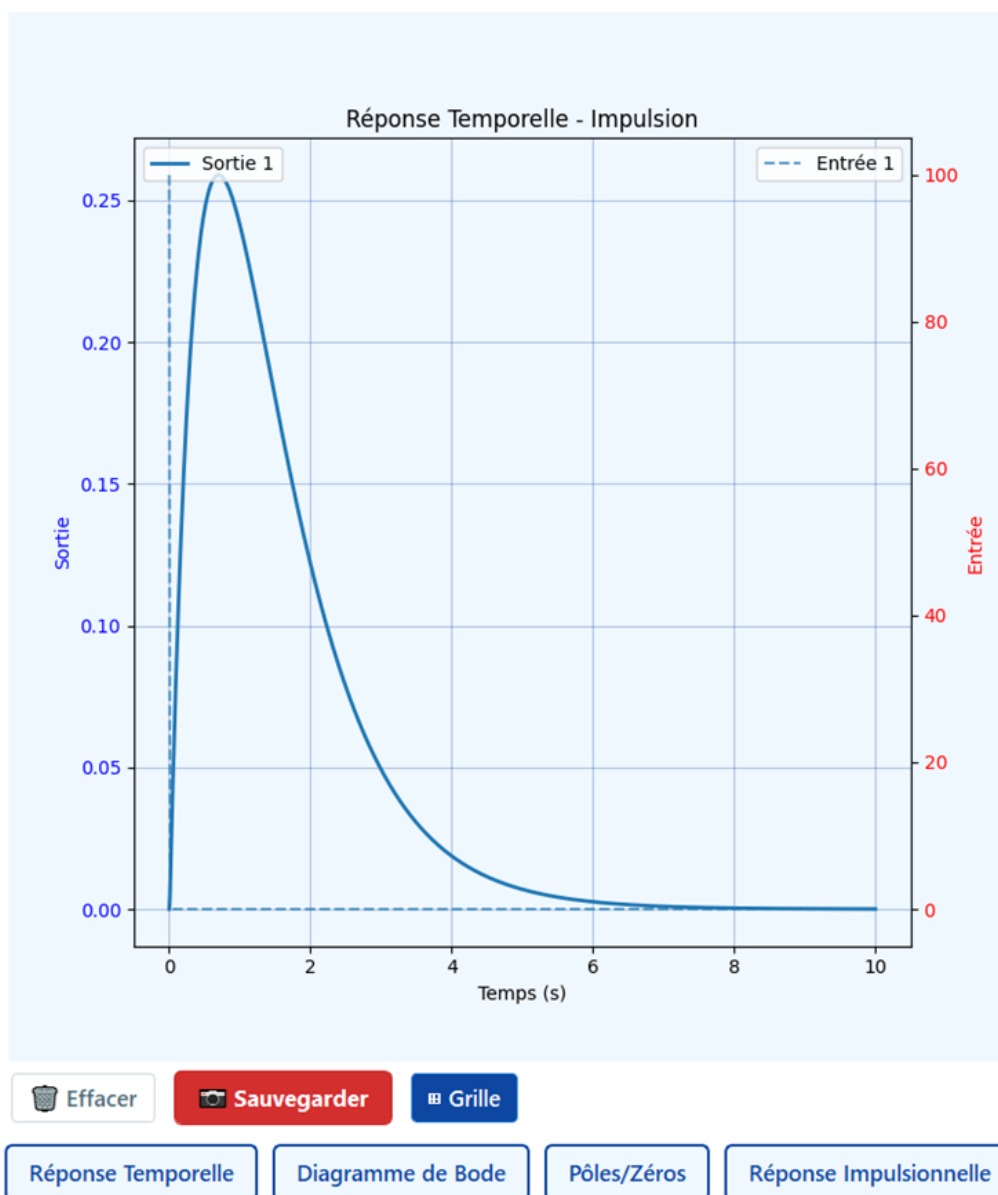


FIGURE 1.6 – Réponse temporelle- impulsion

NB : L'application permet également de :

- Générer aléatoirement des systèmes pour effectuer des tests,
- Valider les systèmes définis,
- Importer des systèmes depuis un fichier **.csv**,
- Exporter les graphiques sous forme d'images en utilisant le bouton **Capturer**.

Chapitre 2

Contrôlabilité et Observabilité

Cette section vérifie si l'on peut piloter complètement un système (contrôlabilité) et si toutes les variables internes peuvent être reconstruites à partir des mesures (observabilité).

Objectif : Vérifier si le système peut être commandé et observé.

Fonctionnalités :

- Calcul des matrices W_c et W_o ,
- Analyse des rangs,
- Détection d'états inaccessibles,
- Visualisation des valeurs singulières.

Étapes :

1. Entrer les matrices A, B, C, D ,
2. Cliquer sur **Analyser Contrôlabilité** ou **Analyser Observabilité**,
3. Lire le verdict affiché automatiquement.

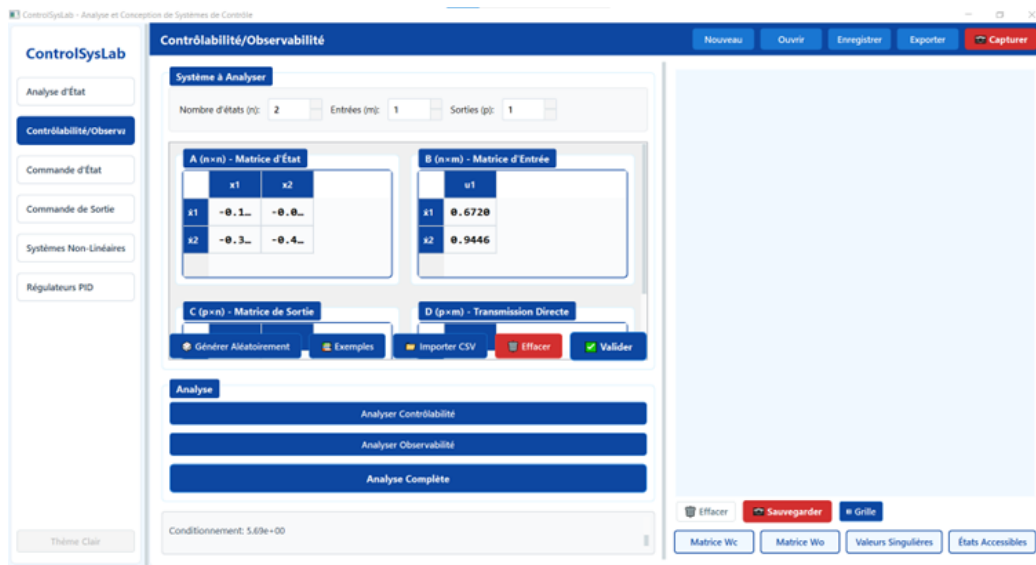


FIGURE 2.1 – Contrôlabilité et observabilité

Exemple :

$$W_c = \begin{bmatrix} 0 & 1 \\ 1 & -3 \end{bmatrix}, \quad \text{rang}(W_c) = 2 \Rightarrow \text{système contrôlable.}$$

$$W_o = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \text{rang}(W_o) = 2 \Rightarrow \text{système observable.}$$

matrice de contrôlabilité W_c :
Dimensions: 2x2
Rang: 2

FIGURE 2.2 – Rang de la matrice de commandabilité

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

FIGURE 2.3 – Verdict sur la commandabilité

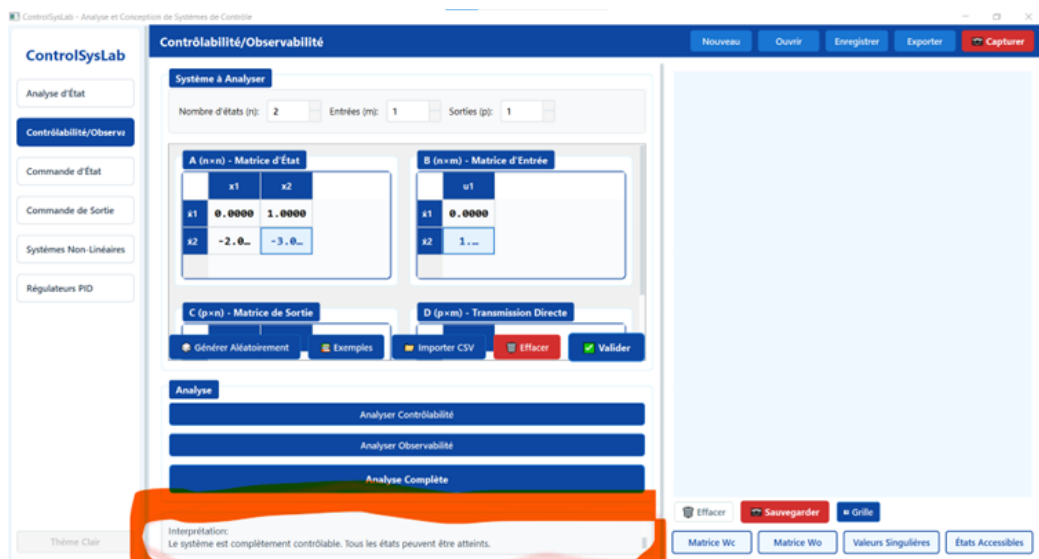


FIGURE 2.4 – résultat affiché à coté à gauche

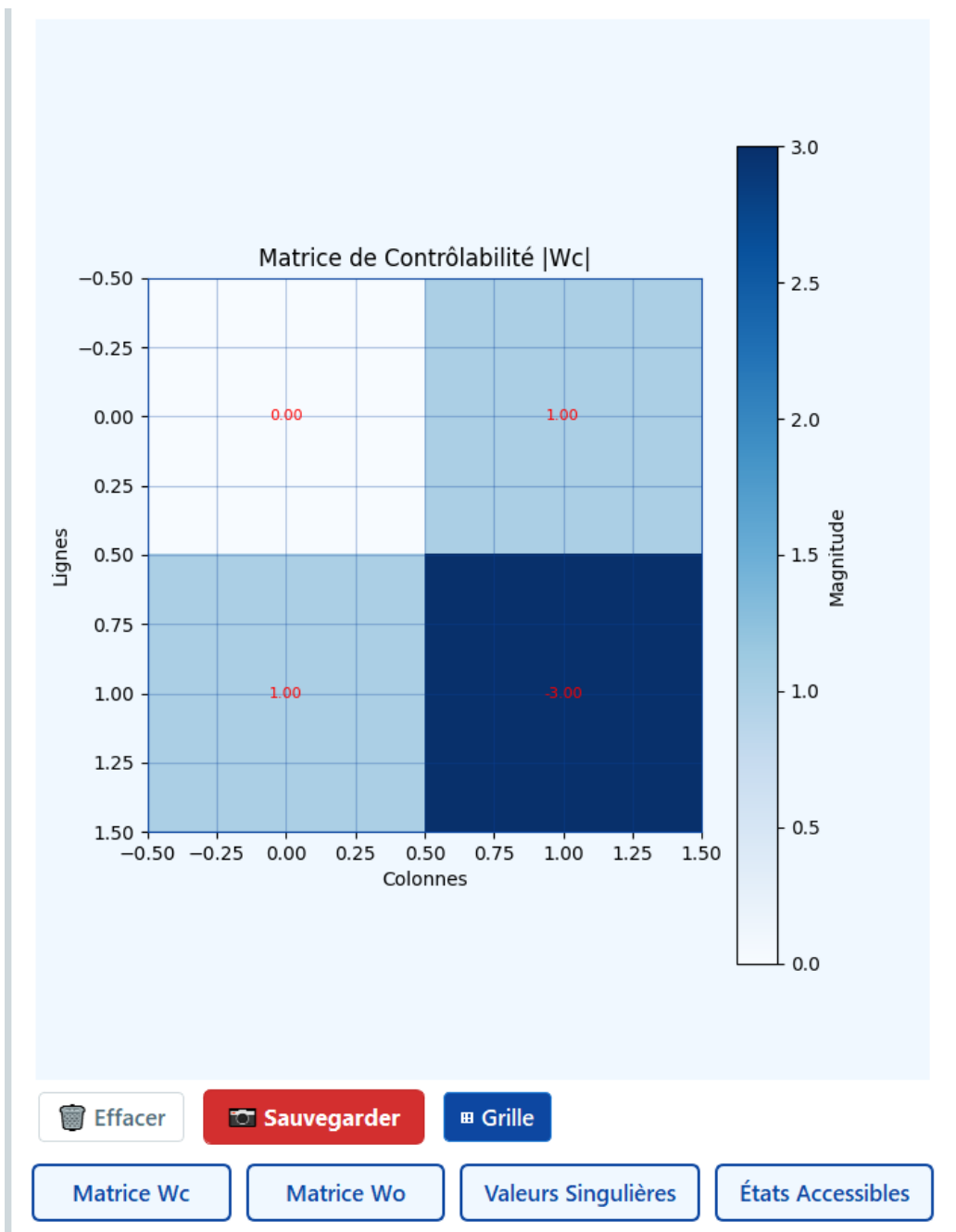


FIGURE 2.5 – matrice de controlabilité $[W_c]$

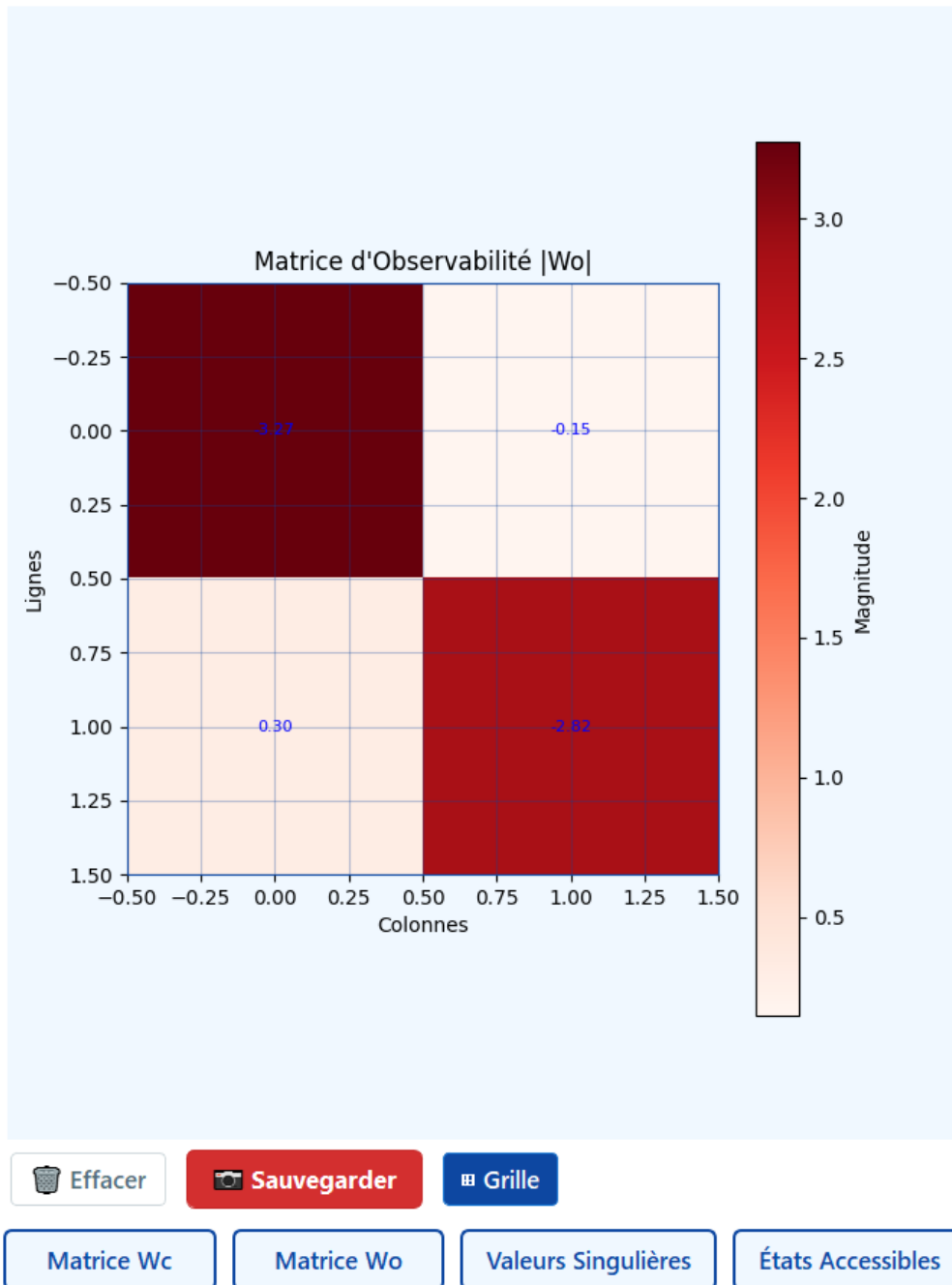


FIGURE 2.6 – matrice d'observabilité $[W_o]$

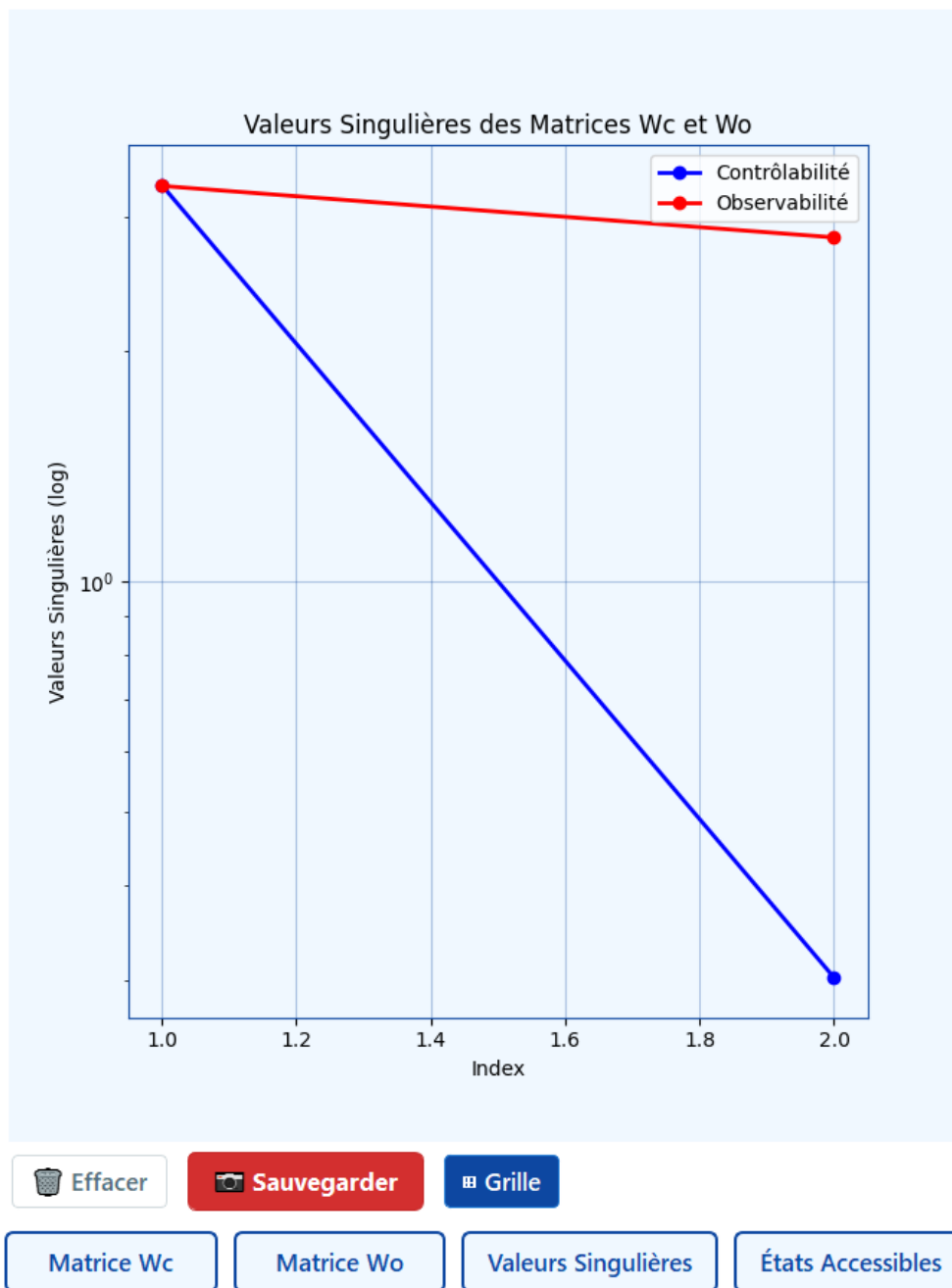


FIGURE 2.7 – Valeurs singulières de W_c et W_o

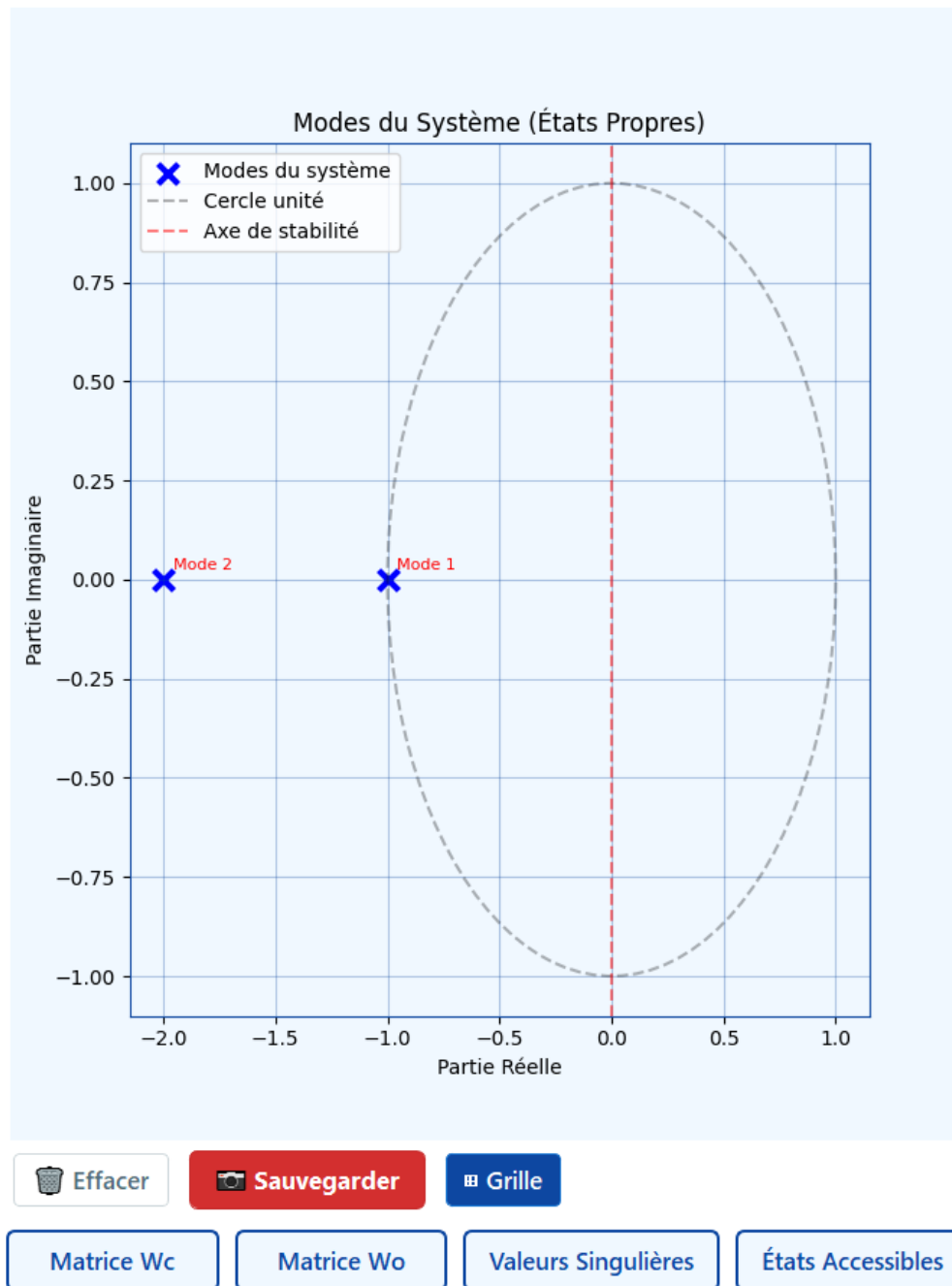


FIGURE 2.8 – États propres ou modes du systèmes

Chapitre 3

Commande d'État

Cette fonctionnalité permet de concevoir des lois de commande par retour d'état. L'utilisateur définit les pôles souhaités et l'application calcule la matrice de gains de retour d'état.

Objectif : Placer les pôles désirés par retour d'état.

Fonctionnalités :

- Placement des pôles par spécification,
- Calcul du gain K ,
- Simulation en boucle fermée.

Étapes :

1. Définir le système,
2. Indiquer les pôles souhaités,
3. Cliquer sur **Concevoir Contrôleur**,
4. L'application calcule K et simule la réponse.

Exemple :

Placement des pôles désirés : -4 et 5 .

Gain obtenu :

$$K = \begin{bmatrix} 14 & 7 \end{bmatrix}$$

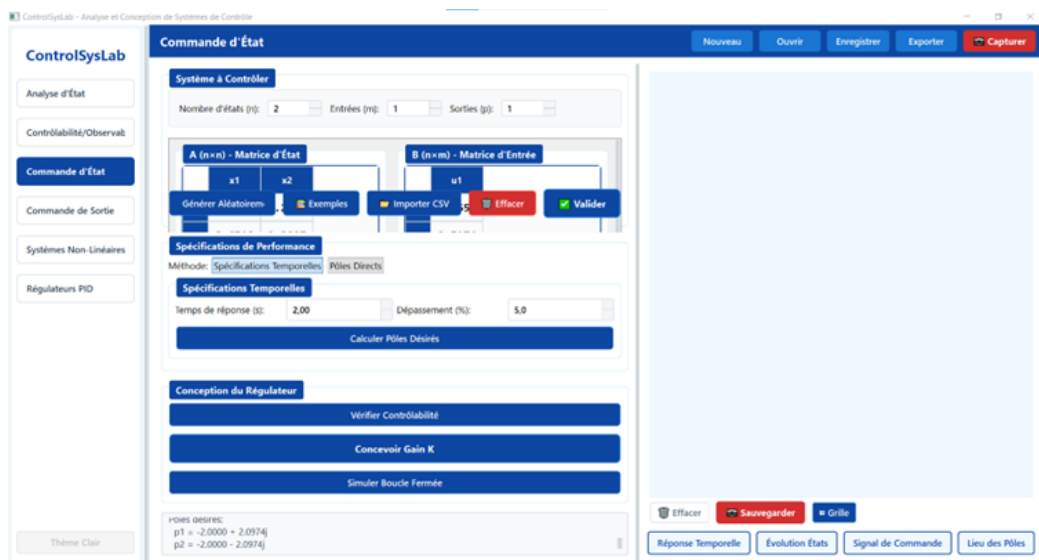


FIGURE 3.1 – Commande d'état

On peut calculer les pôles désirés

Pôles désirés:
p1 = -1.2903 + 1.5244j
p2 = -1.2903 - 1.5244j

FIGURE 3.3 – pôles désirés

- Amortissement ζ : 0.646
- Pulsation naturelle ω_n : 1.997 rad/s

FIGURE 3.4 – pôles désirés

Chapitre 4

Commande de Sortie

La commande de sortie permet de synthétiser des régulateurs en fonction des sorties mesurées. Cela est utile lorsque toutes les variables d'état ne sont pas accessibles.

Objectif : Concevoir un observateur et le combiner avec le contrôleur.

Fonctionnalités :

- Observateur de Luenberger,
- Mesure partielle ou complète,
- Analyse de l'erreur d'estimation,
- Comparaison boucle complète vs boucle partielle.

Étapes dans l'application :

1. Entrer les matrices A, B, C ,
2. Choisir les sorties mesurées,
3. Définir la vitesse de l'observateur,
4. Cliquer sur **Concevoir Observateur** (Photo 4).

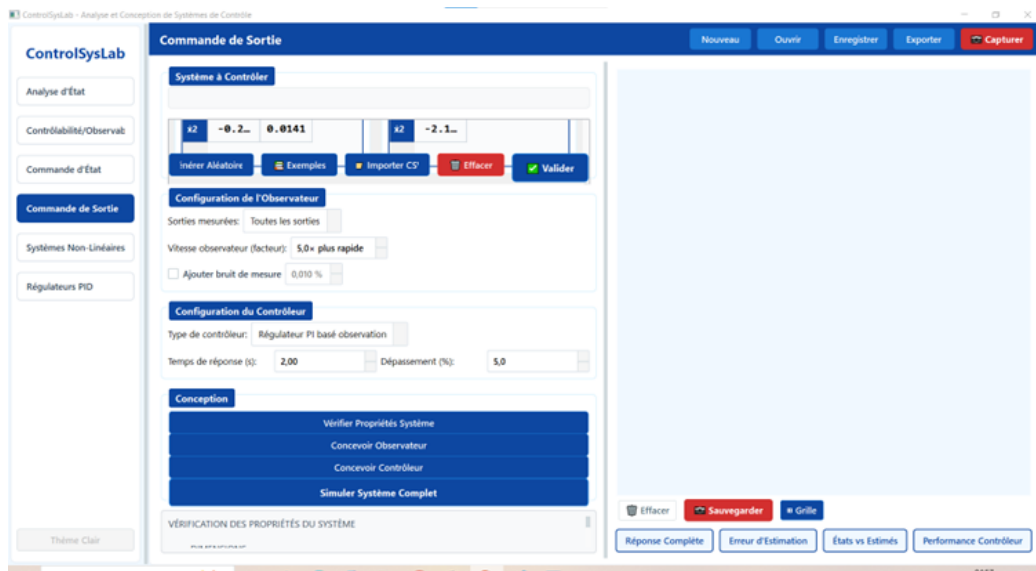


FIGURE 4.1 – commande de sortie

Exemple : Mesure partielle : $y = x_1$

Pôles de l'observateur : -6 et -7

Gain de l'observateur :

$$L = \begin{bmatrix} 11 \\ 35 \end{bmatrix}$$

Commande de Sortie

Système à Contrôler

×2

-0.2...

0.0141

×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

FIGURE 4.2 – spécifications de la commande de sortie

Chapitre 5

Systèmes Non Linéaires

Cette section permet d'analyser et de simuler des systèmes non-linéaires. Elle est dédiée aux cas où les équations d'état ne respectent pas la linéarité.

Objectif : Étudier des systèmes décrits par des équations différentielles non-linéaires.

Fonctionnalités :

- Éditeur d'équations
- Linéarisation automatique (Jacobian)
- Méthode de Lyapunov
- Portraits de phase

Utilisation dans l'application :

1. Éditer les équations dans l'éditeur (version Python / Sympy)
2. Choisir le point d'équilibre
3. Cliquer sur *Valider Système*
4. Choisir *Linéariser* ou *Analyser Lyapunov*

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

FIGURE 5.1 – Systèmes non-linéaire

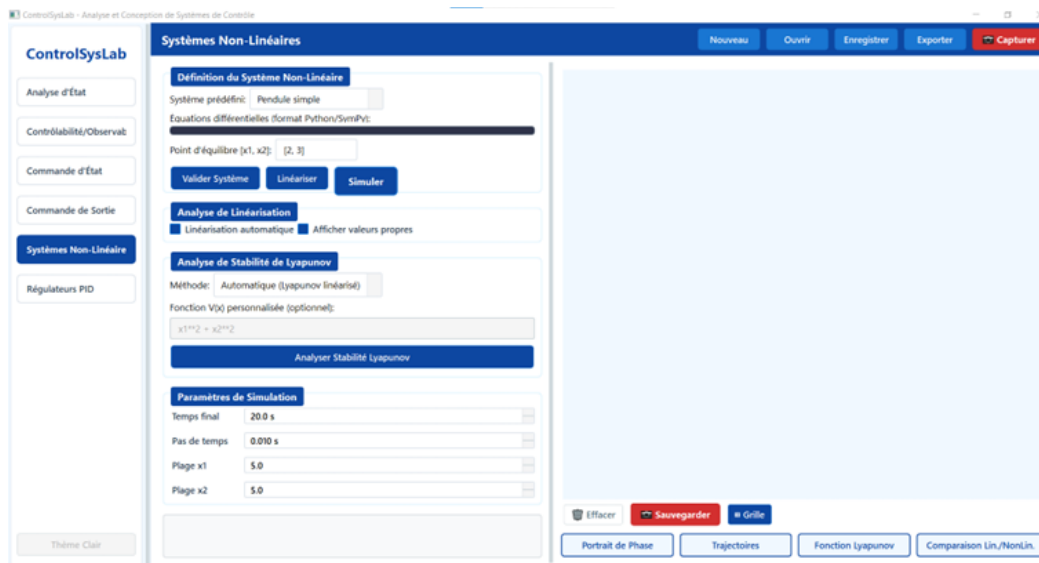


FIGURE 5.2 – Paramètres systèmes non-linéaire

NB : L'application permet également de : Il est possible de générer les portraits de phase, les trajectoires et les fonctions de Liapunov.

Chapitre 6

Régulateurs PID

La partie PID permet de concevoir des correcteurs proportionnels-intégraux-dérivatifs. Elle est utile pour obtenir un bon compromis entre rapidité, précision et robustesse.

Objectif : Régler un système par PID.

Fonctionnalités :

- Méthodes automatiques : Ziegler-Nichols, Cohen-Coon
- Réglage manuel
- Anti-windup
- Comparaison graphique

Utilisation dans l'application :

1. Définir le système (matrices A, B, C, D ou fonction de transfert)
2. Sélectionner une méthode de réglage
3. Cliquer sur *Simuler PID*
4. Comparer les performances

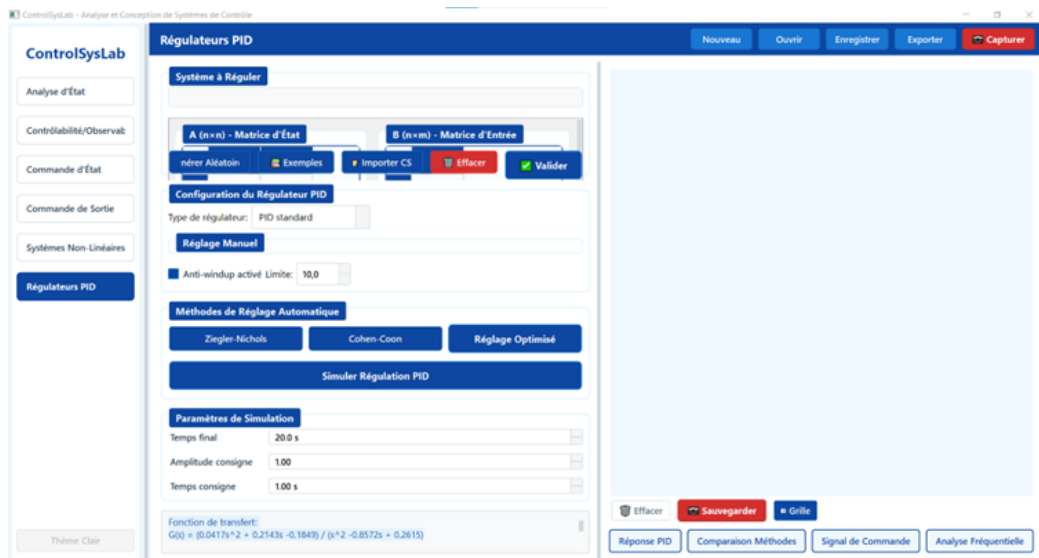


FIGURE 6.1 – Paramètres PID

Exemples

Fonction de transfert :

$$G(s) = \frac{0.0417s^2 + 0.2143s - 0.1849}{s^2 - 0.8572s + 0.2615}$$

Réglage PID : Exemple avec la méthode Ziegler-Nichols → temps de réponse plus court, erreur réduite.

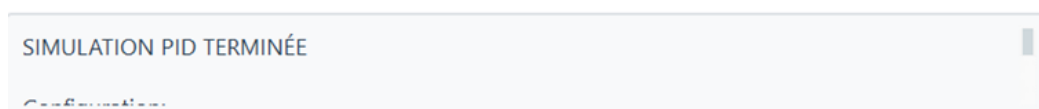


FIGURE 6.2 – Simulation PID

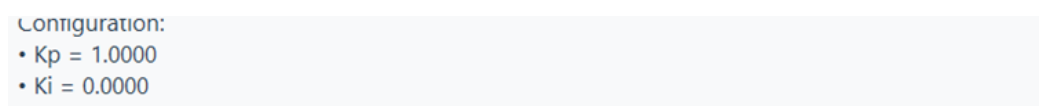


FIGURE 6.3 – gains du contrôleur PID

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

FIGURE 6.4 – gains du contrôleur PID

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

FIGURE 6.5 – performances du contrôleur PID

- temps d'établissement: 19.550 s
- Erreur statique: 84.0594
- ISE (Intégrale erreur²): 838742.336043

FIGURE 6.6 – performances du contrôleur PID

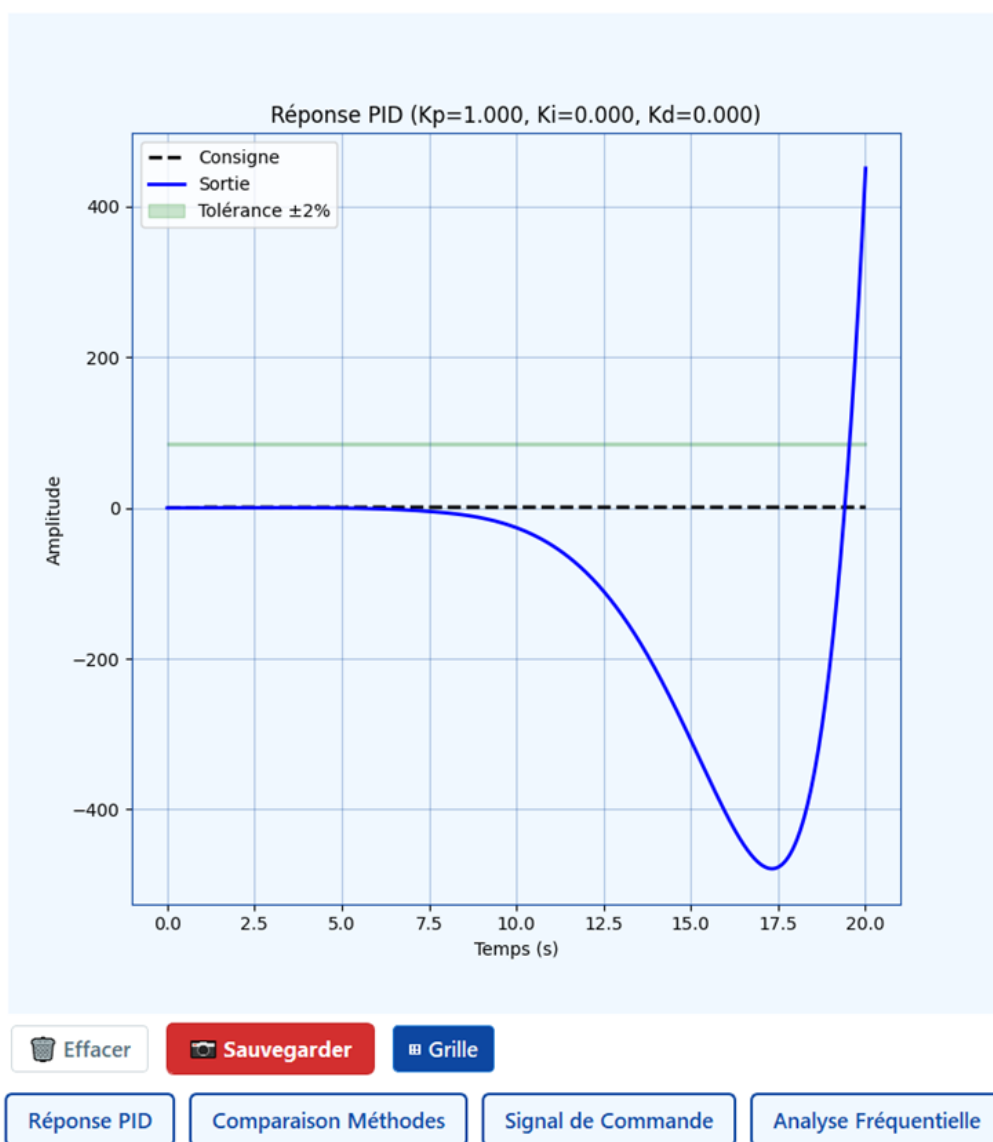


FIGURE 6.7 – réponse du contrôleur PID

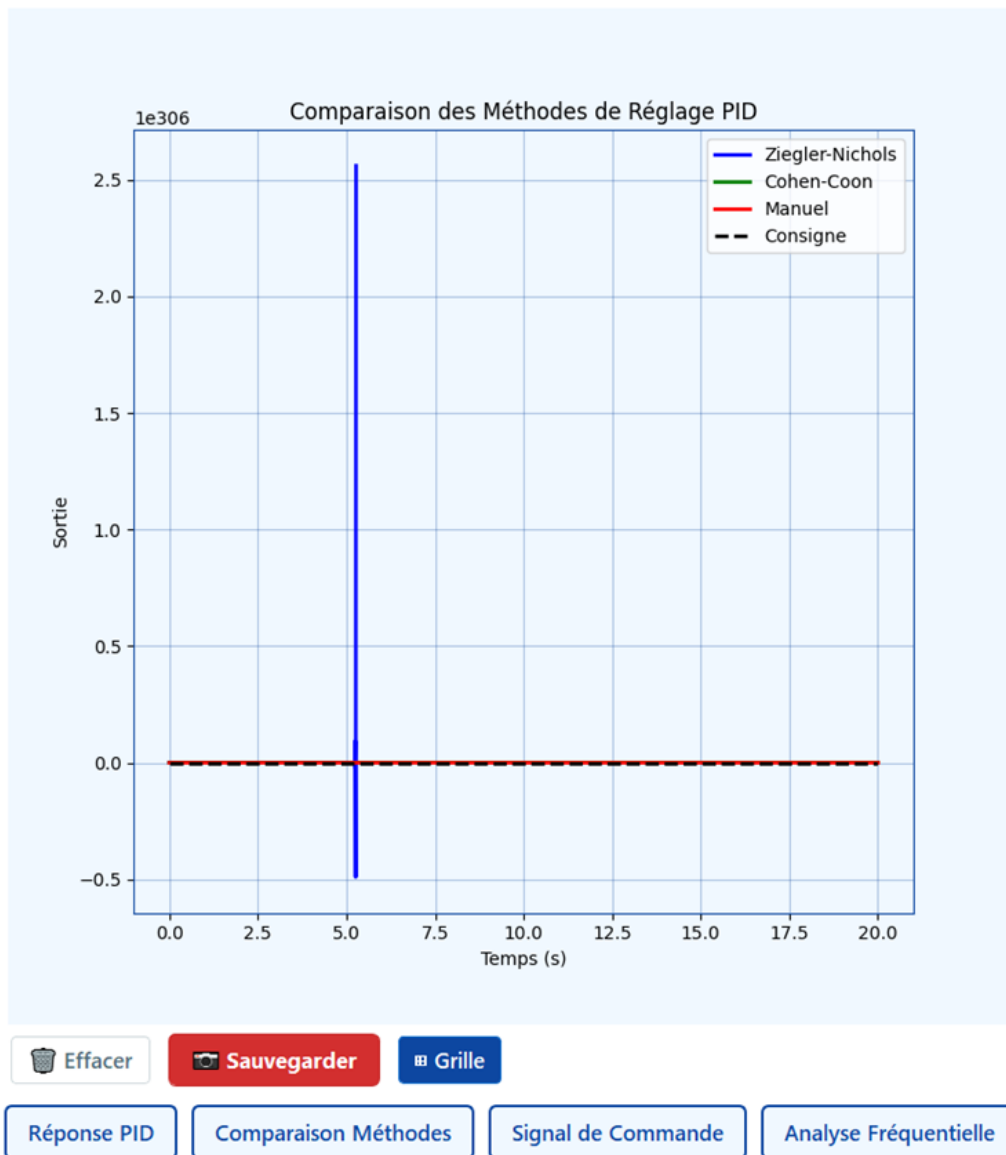


FIGURE 6.8 – comparaison des méthodes de réglage PID

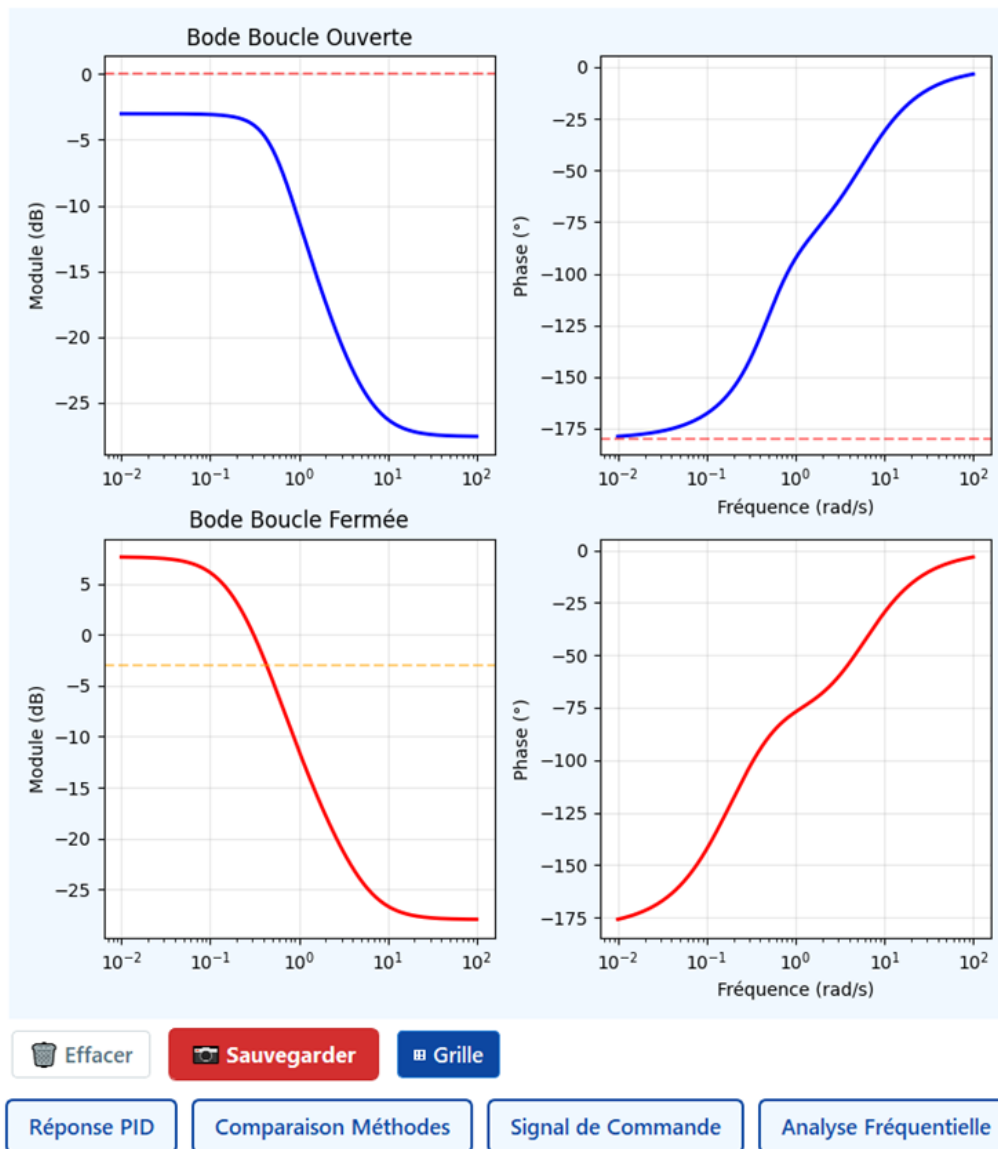


FIGURE 6.9 – Analyse fréquentielle

Conclusion

L'application **ControlSysLab** a été conçue comme un outil pédagogique et pratique pour faciliter l'analyse et la conception des systèmes de contrôle. Grâce à son interface intuitive et ses six fonctionnalités principales (Analyse d'état, Contrôlabilité/Observabilité, Commande d'état, Commande de sortie, Systèmes non-linéaires et Régulateurs PID), l'utilisateur peut explorer de manière progressive les différentes étapes de l'asservissement et de la commande des systèmes dynamiques.

Chaque module a été pensé pour permettre une mise en pratique directe, à partir de modèles théoriques jusqu'à l'obtention de résultats visuels et exploitables. Les exemples fournis et les explications associées servent de guide afin que l'utilisateur puisse appliquer ces méthodes à ses propres systèmes.

En résumé, **ControlSysLab** se veut être un pont entre la théorie et la pratique : un support d'apprentissage, d'expérimentation et d'analyse, permettant à l'étudiant comme au praticien de mieux comprendre, concevoir et valider des systèmes de contrôle modernes.