

Amélioration de l'Asservissement de Visée Laser

Revue de lancement de projet 5A: 24 octobre 2024

Client : Pierre Chambert (JTL-Electronique)

Tuteur industriel : François Kersulec

Professeur Référent : Jacques Laffont



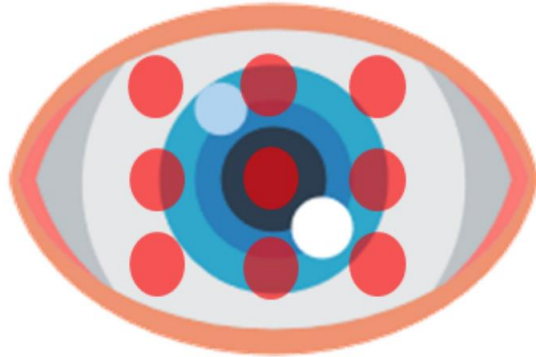
Présenté par Yoan Douarre, Marouane Hsaini et Mathis Pascal

Contexte

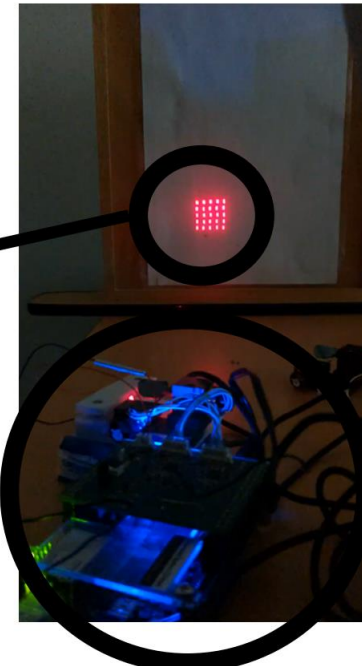
Les personnes diabétiques souffrent de lésions oculaires.

Afin d'éviter d'éventuelles complications, les chirurgiens viennent blesser l'œil à l'aide d'un laser afin de provoquer une cautérisation d'une zone souffrante de la rétine.

Patient diabétique dont les vaisseaux sanguins oculaires sont abimés



Matrice de points utilisée pour effectuer les opérations



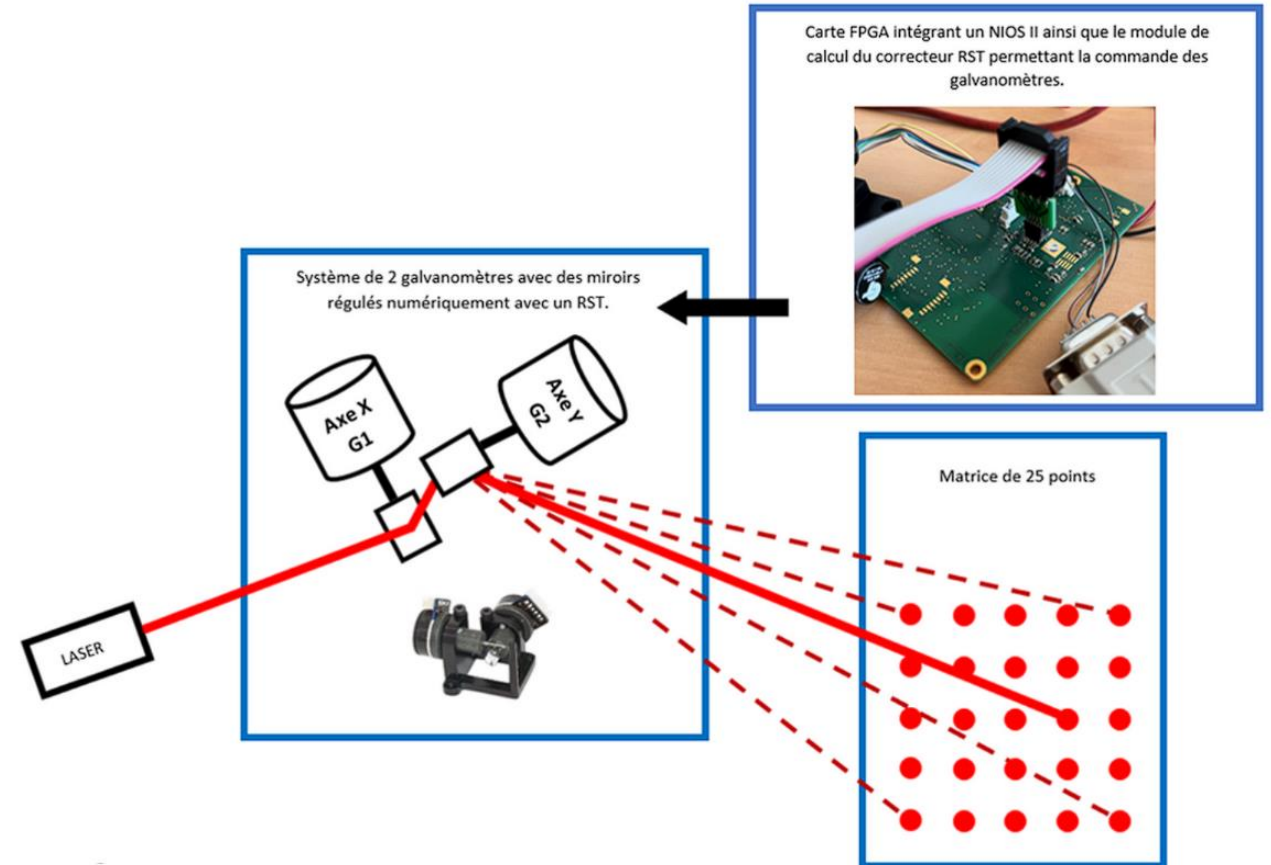
Notre système

Produits sortants

Les codes permettant au système composé de deux galvanomètres et d'un laser d'afficher une matrice de 25 points suffisamment rapidement pour ne pas percevoir d'effet de scintillements.

Les codes permettront de :

- Réaliser l'identification des galvanomètres X et Y
- Calculer les coefficients RST à appliquer
- Envoyer ces coefficients au FPGA
- Programmer le FPGA pour appliquer une correction RST
- Asservir le système pour obtenir la matrice de points.



Cahier des charges

Type	Nature	Détails
Fonction	Affichage d'une matrice de 25 points	<ul style="list-style-type: none"> Distance entre deux points consécutifs : 11mm pour une distance de 30cm entre les miroirs et la cible. Sans scintillements : affichage de la matrice complète en 20ms.
Contrainte	Temps de déplacement et de stabilisation d'un point	800µs par point : <ul style="list-style-type: none"> Temps de montée : 400µs Temps de stabilisation : 400µs
Contrainte	Contrôle de deux galvanomètres	Alimentation en -5V/+5V
Contrainte	Récupération des données sur la position des galvanomètres	Récupération de la position sur un CAN 14 bits
Contrainte	Calculs des commandes au sein de la carte	<ul style="list-style-type: none"> Utilisation du FPGA pour implémenter le correcteur RST Envoi des données sur un CNA 16 bits
Contrainte	Communication entre Scilab et la carte via le port série	Les caractéristiques du galvanomètre doivent être redéterminés : <ul style="list-style-type: none"> Identification du système calcul des coefficients du RST envoi des coefficients dans le FPGA

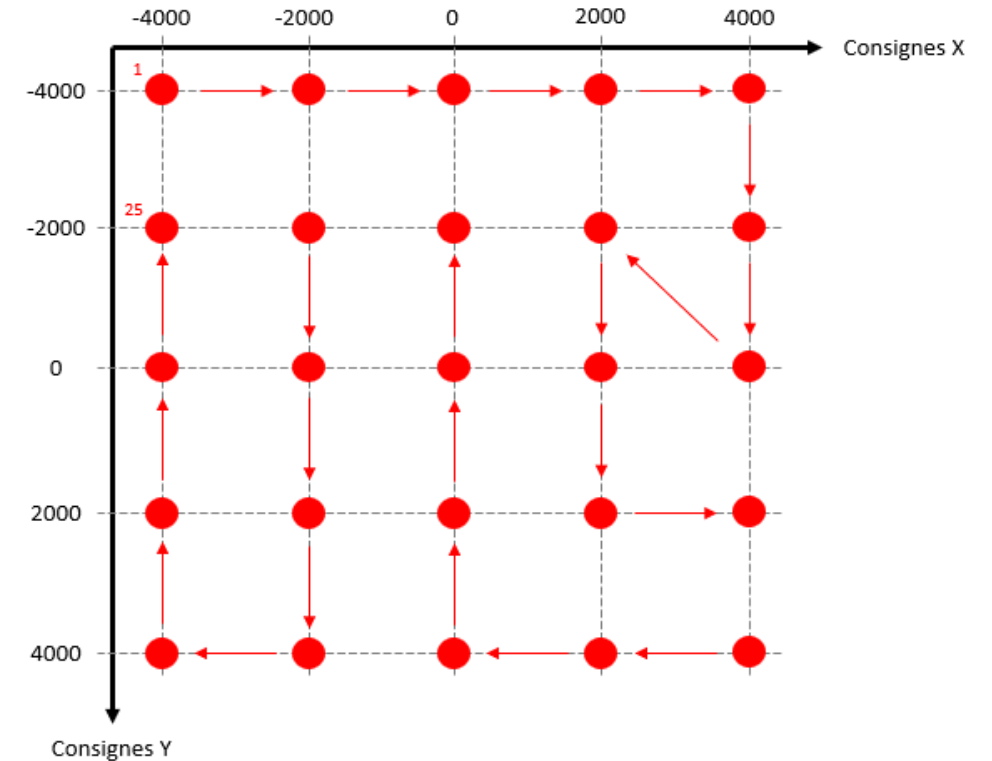
Livrables

- Notes d'application regroupées et clarifiées :
 - Fonctionnement de la carte et utilisation de signal tap
 - Fonctionnement du RST en mode debug
 - Théorie sur la méthode d'identification SBPA
 - Méthode de calcul des coefficients du correcteur RST
- L'intégralité des codes utilisés pour la réalisation finale

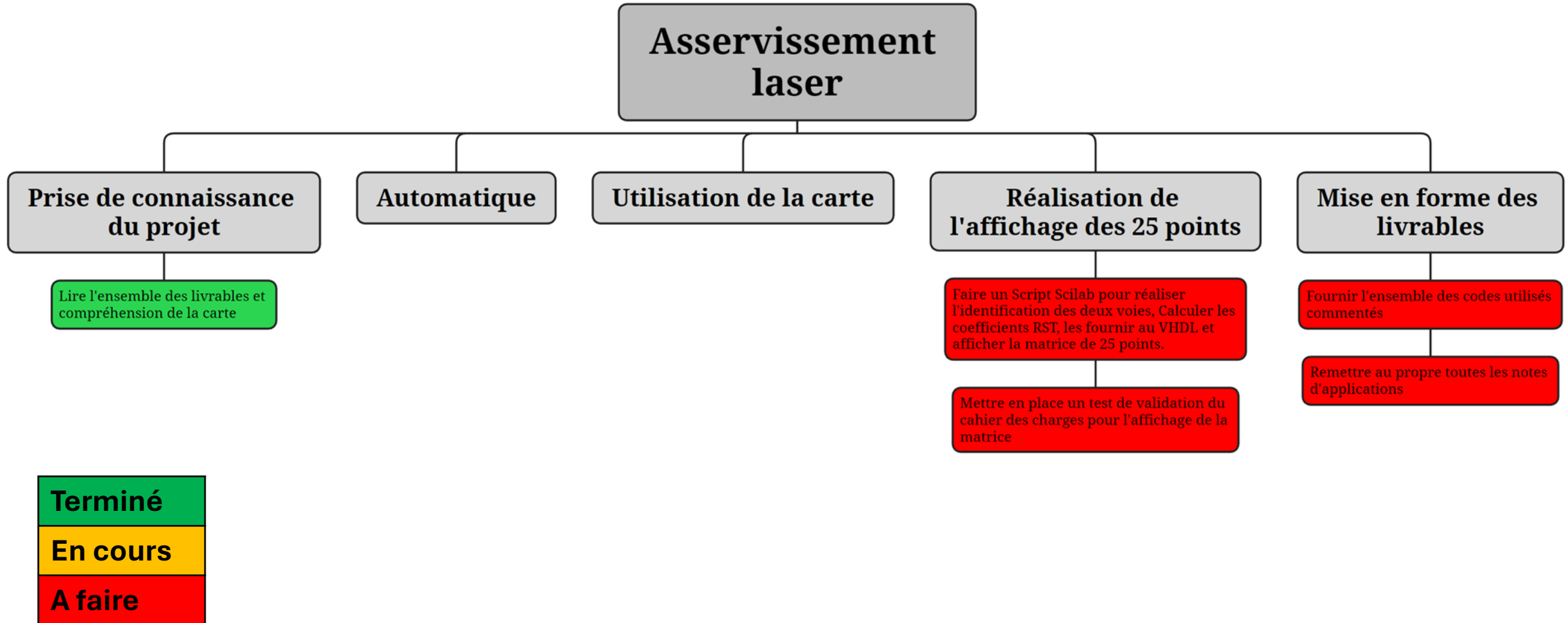
Démonstration du fonctionnement

Nous placerons le laser à une distance de 30cm du tableau sur lequel nous accrocherons une feuille avec la matrice des 25 points espacés de 11mm les uns des autres.

La démonstration sera concluante si les points se superposent à la matrice de la feuille et si nous n'observons pas d'effet de scintillement.



Etat du projet au début de l'année



Automatique

Initiation à Scilab

Créer et simuler des systèmes d'ordre 1 à 3

Mettre ces scripts sous forme de fonctions

Identification du système

Tester les Scripts d'identification SBPA de l'année précédente

Simulation d'identifications d'ordres 1 à 3 pour un système simulé connu avec et sans bruit

Simuler l'identification par la méthode de la pseudo-inverse

Simuler l'identification par la méthode des Variables Instrumentales

Simuler l'identification par la fonction ARMAX

Mettre toutes les méthodes d'identification sous forme de fonctions

Identification réelle

Trouver les paramètres d'identification optimaux

Réaliser une série d'acquisition avec les paramètres optimaux

Réaliser l'identification du système selon les 3 méthodes

Réaliser des identifications dans des conditions différentes de pression

Calcul des coefficients RST

Calculer la fonction de transfert cible qui valide le cahier des charges

Calculer les coefficients RST avec intégrateur d'une fonction d'ordre 3 simulée

Convertir les coefficients discret en continus

Calculer les coefficients RST sans intégrateur

Mettre le calcul des coefficients RST sous forme d'une fonction

Terminé

En cours

A faire



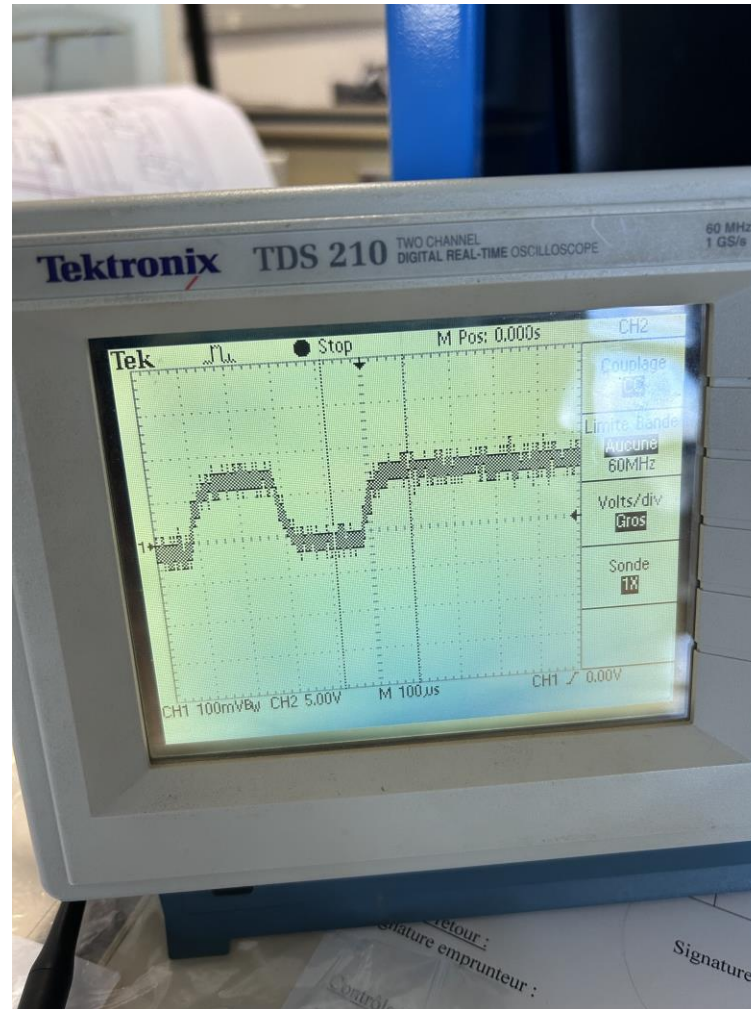
Terminé
En cours
A faire

Travail réalisé depuis le début de l'année

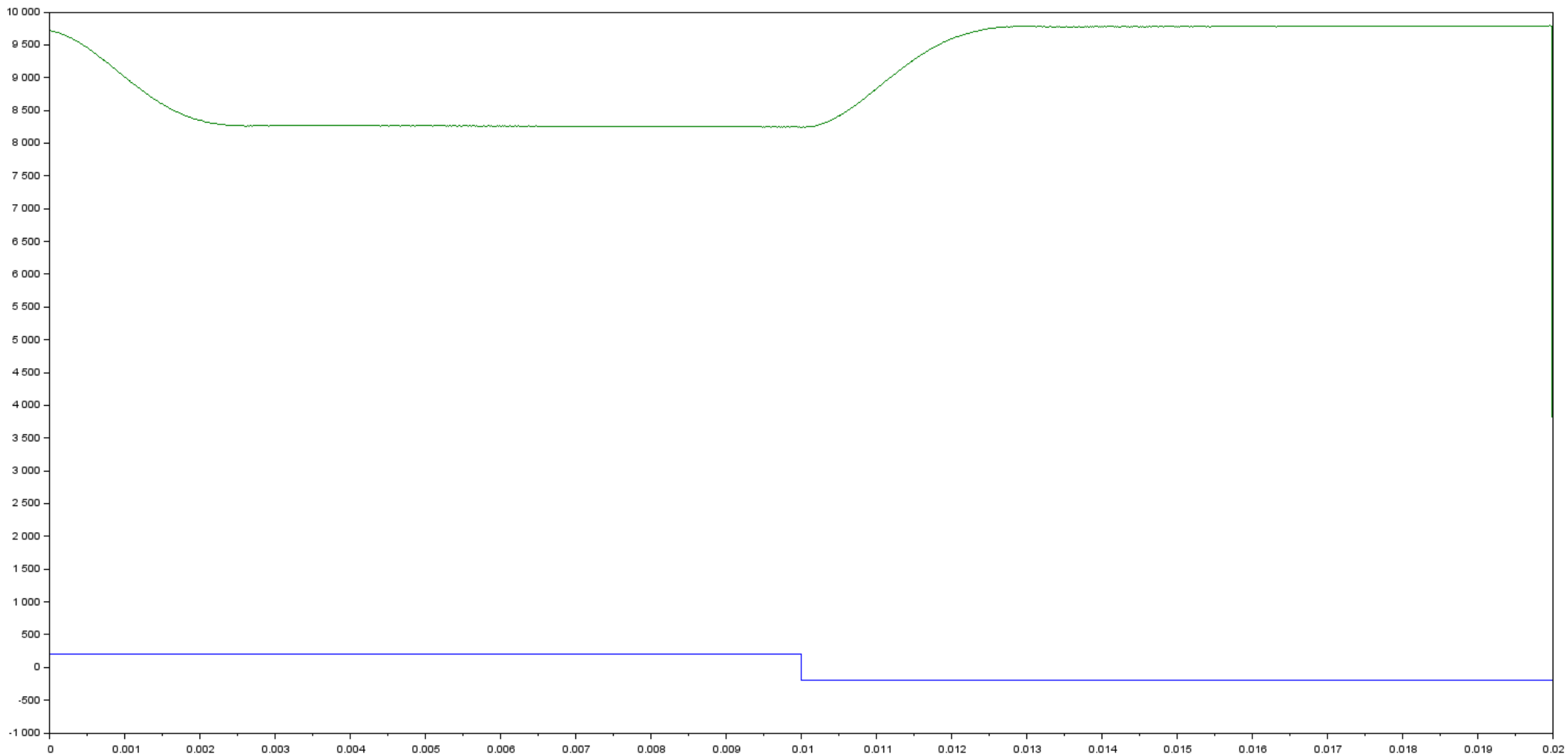
Résumé des étapes à suivre pour la partie automatique

- Connaître l'ordre du système à identifier.
- Connaître le temps de montée du système.
- Choisir une période d'échantillonnage entre 3 et 11 fois plus petite que le temps de montée pour l'identification.
- Identifier le système soumis à une SBPA en utilisant la méthode de la pseudo-inverse, la méthode des variables instrumentales ou la librairie ARMAX de Scilab.
- Convertir les coefficients du système discret identifié en paramètres continus en utilisant des algorithmes d'optimisation.
- Recalculer les coefficients du système en discret à la période d'échantillonnage utilisée pour l'asservissement.
- Calculer les coefficients discrets à la même période d'échantillonnage que précédemment des systèmes cibles du RST pour l'asservissement en régulation et en poursuite.
- Utiliser les coefficients des différents systèmes pour calculer les coefficients R, S et T à utiliser au sein de la carte.

Choix des paramètres optimaux pour l'identification

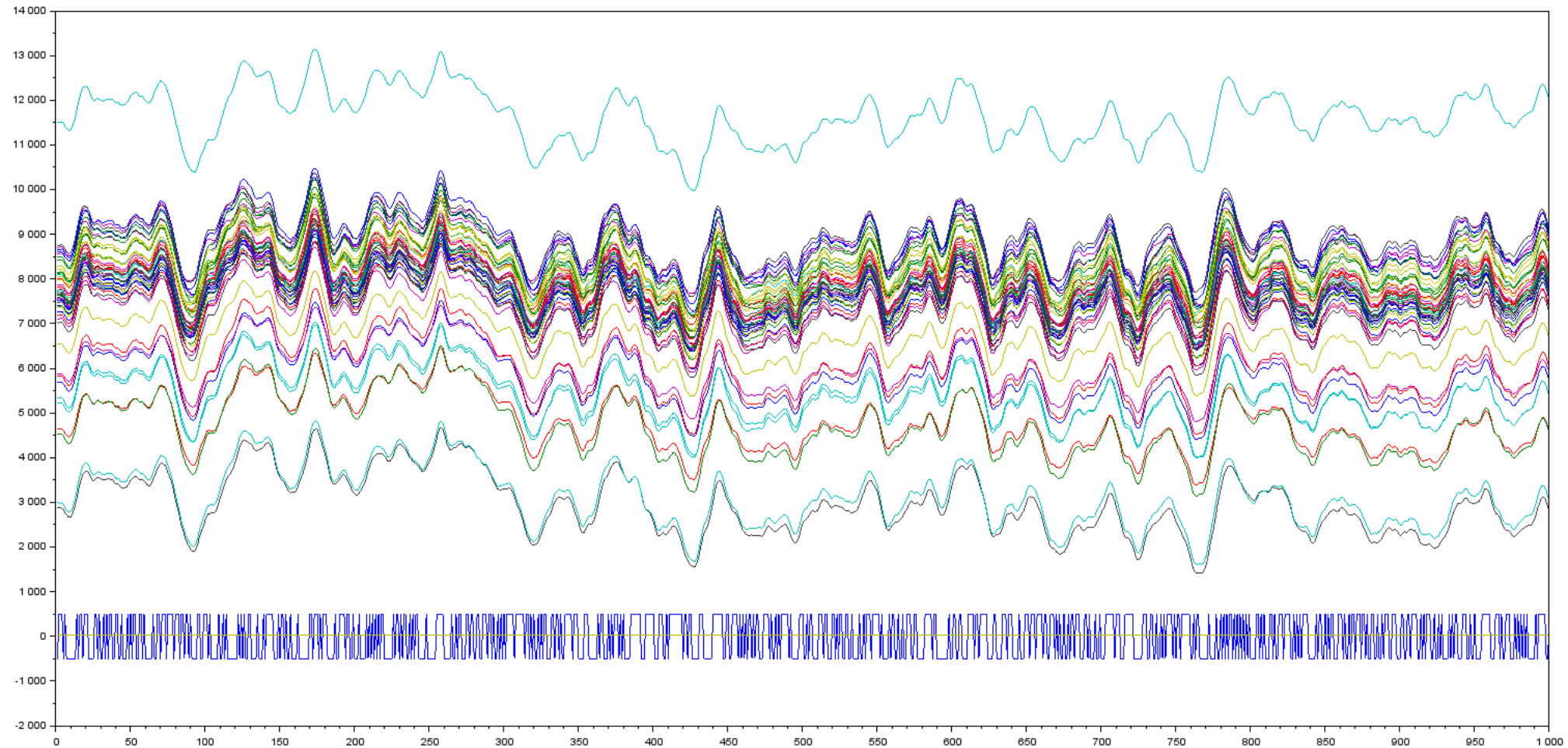


$T_e = 200 \text{ us}$

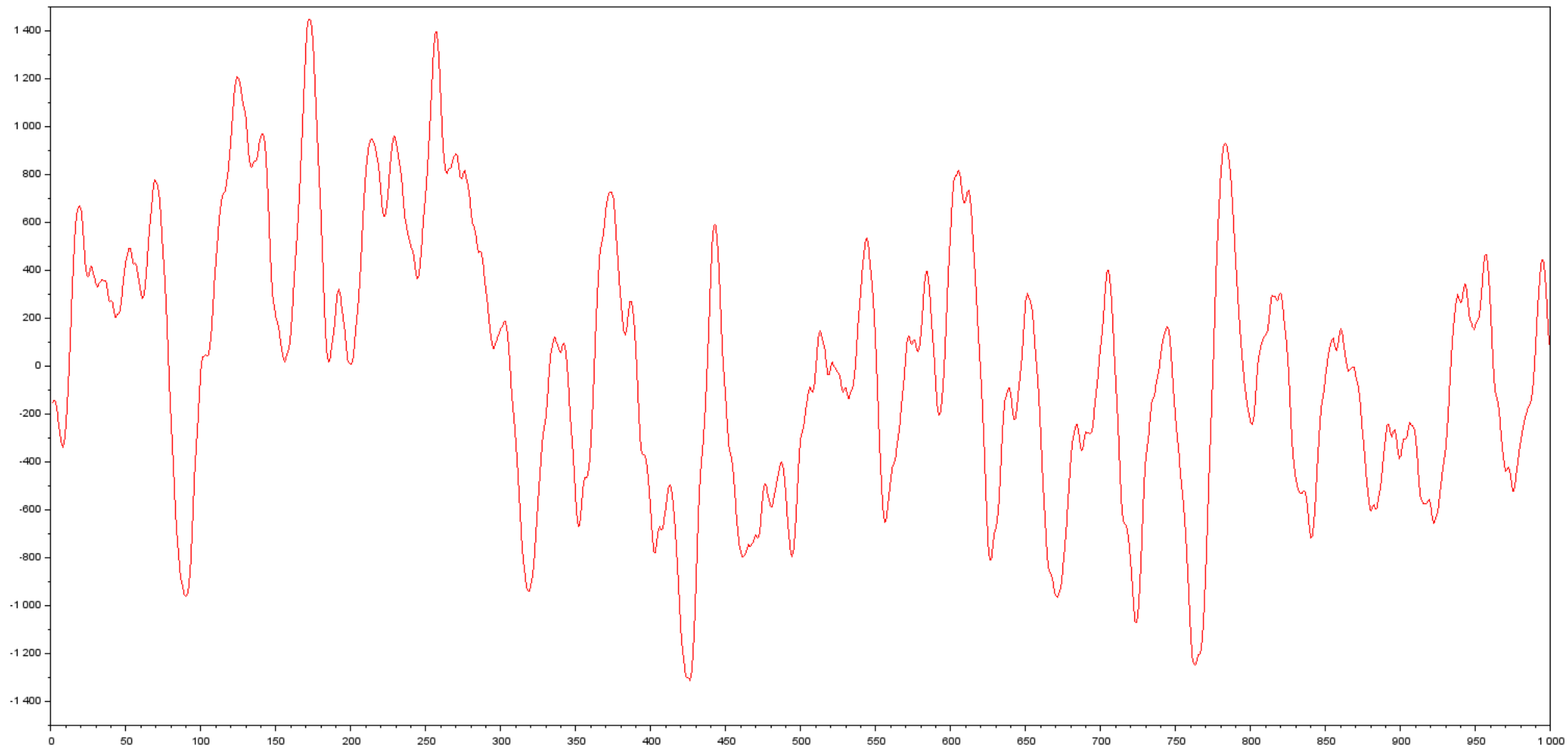


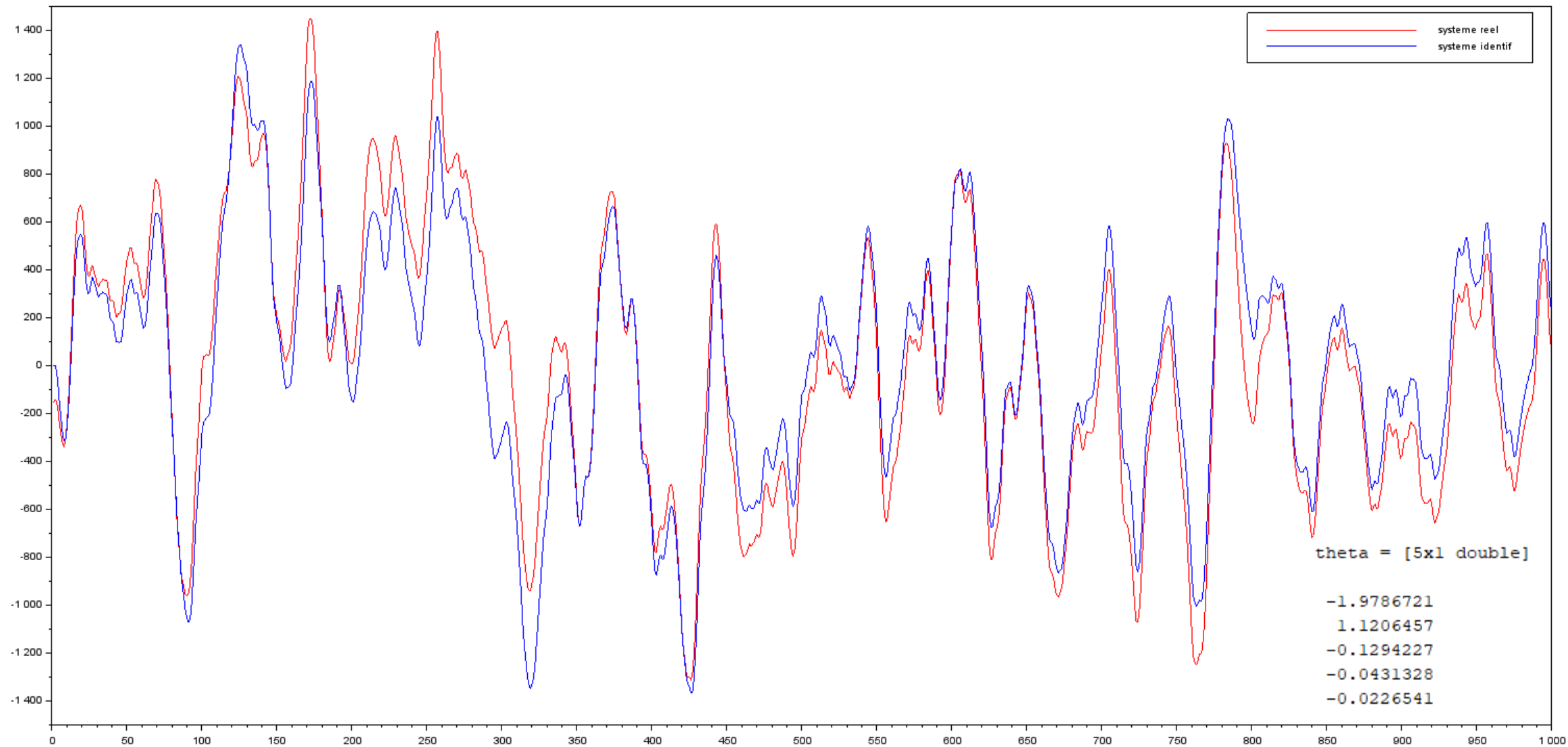
Réponse en boucle ouverte du système soumis à un échelon

Identification du système avec les paramètres optimaux



Moyenne des différentes acquisitions centrée en 0





Comparaison entre le système réel et le système identifié

Conversion des coefficients discrets identifiés en continus

Utilisation d'un algorithme d'optimisation d'Adam

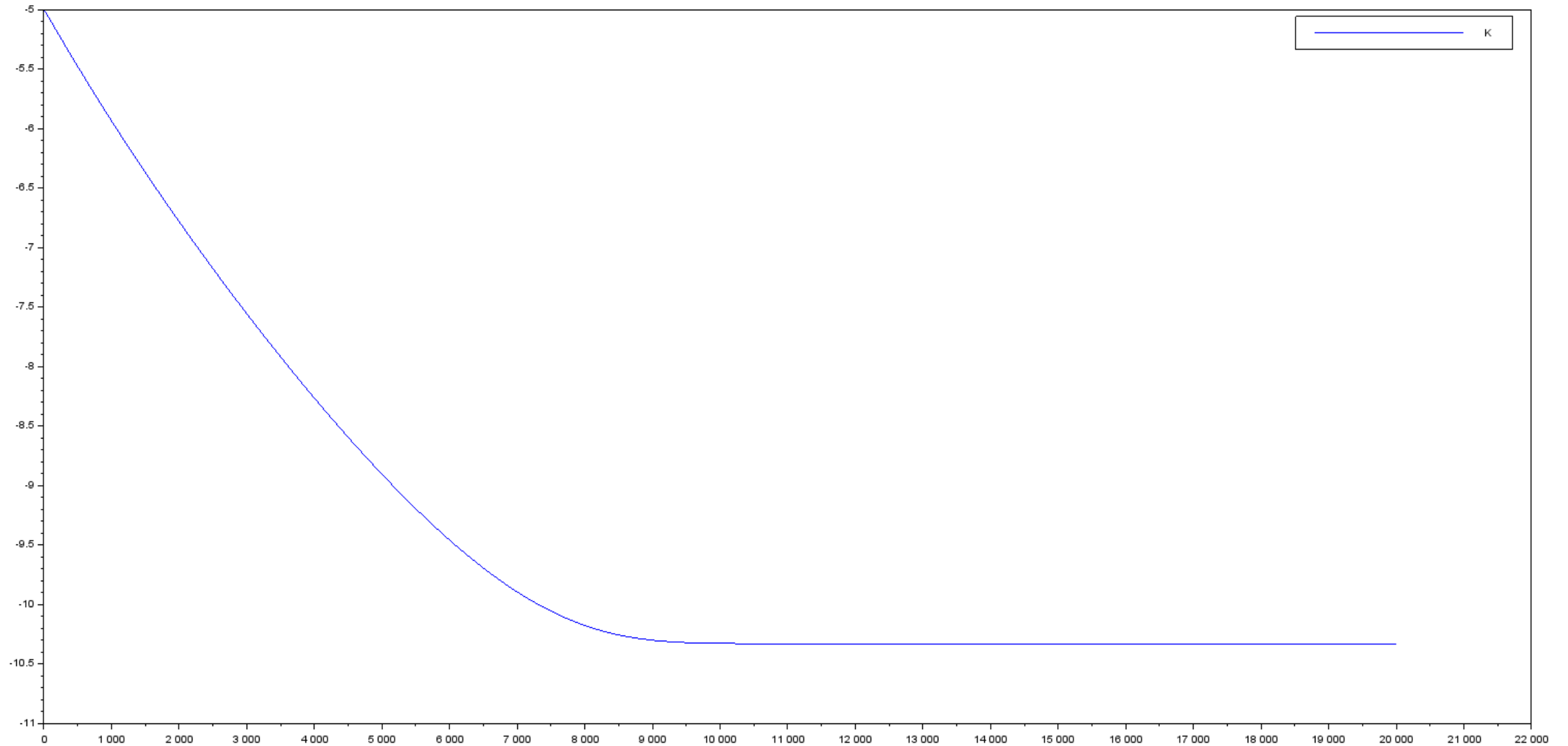
Régression de l'erreur des moindres carrés entre la sortie du système discret identifié et celle d'un système simulé à partir des coefficients continus (K , τ , ξ , t_m)

Les variables à optimiser sont donc K , τ , ξ et t_m .

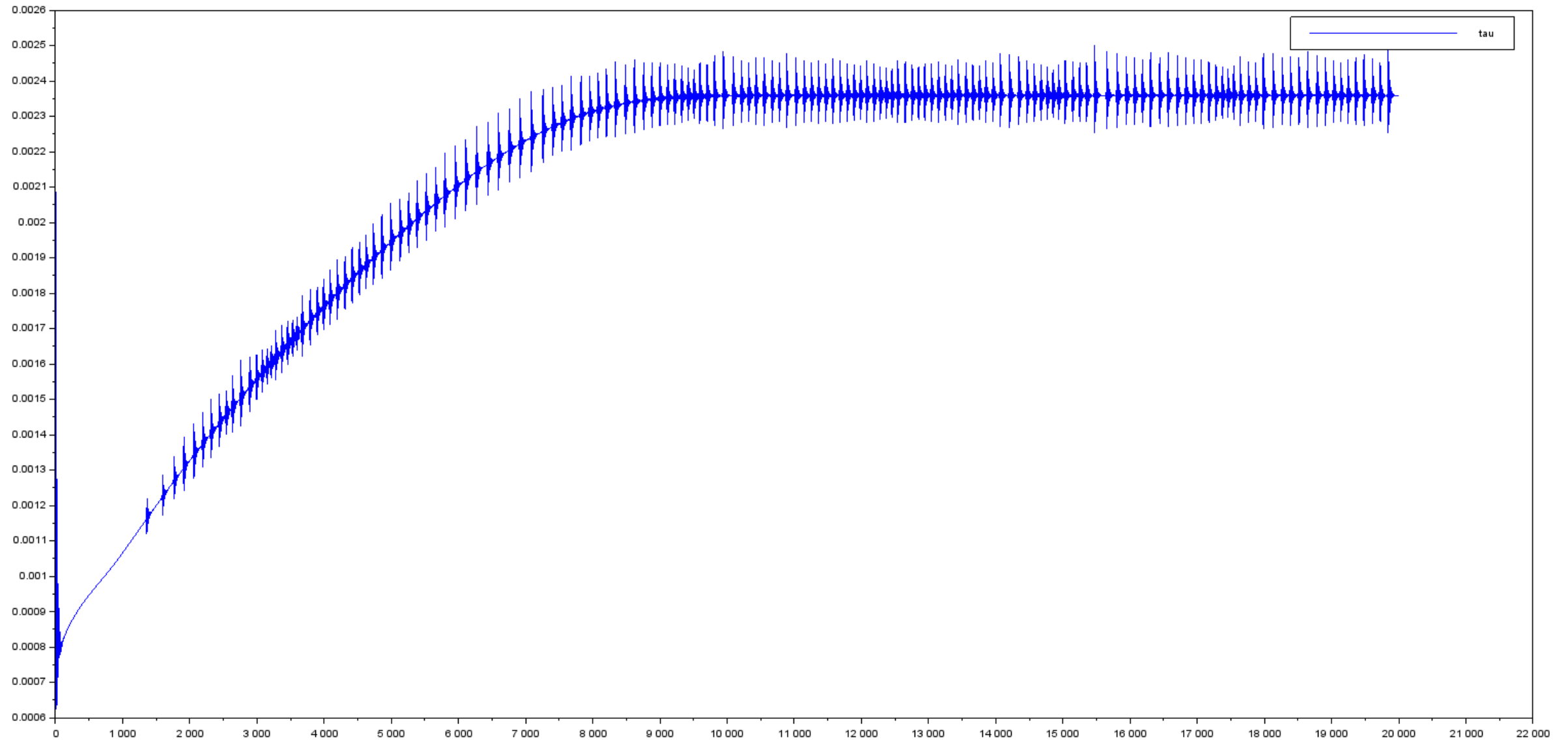
Exemple pour 20 000 itérations avec un learning rate de 0,0001



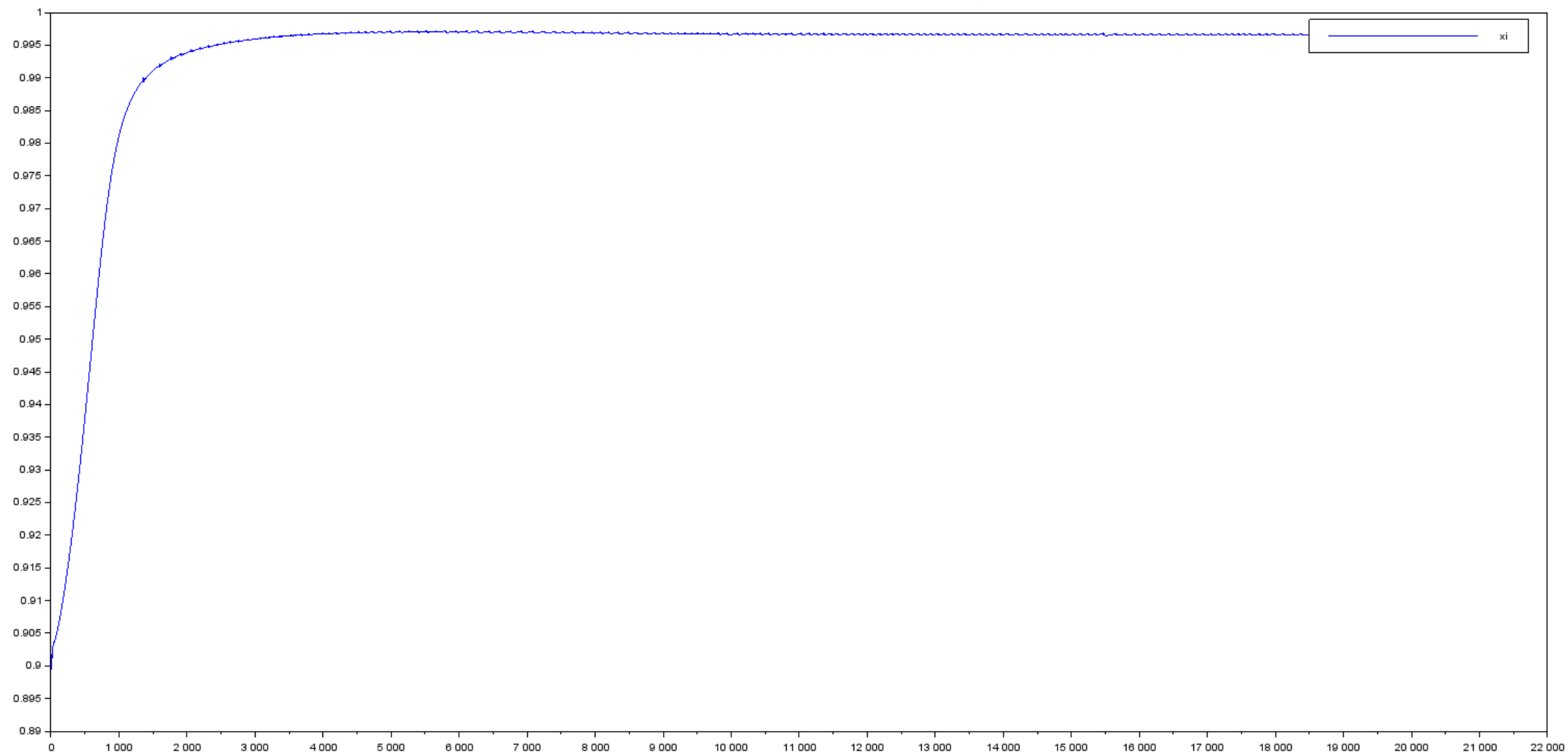
Evolution de la variable K au fur et à mesure des itérations de l'optimisation



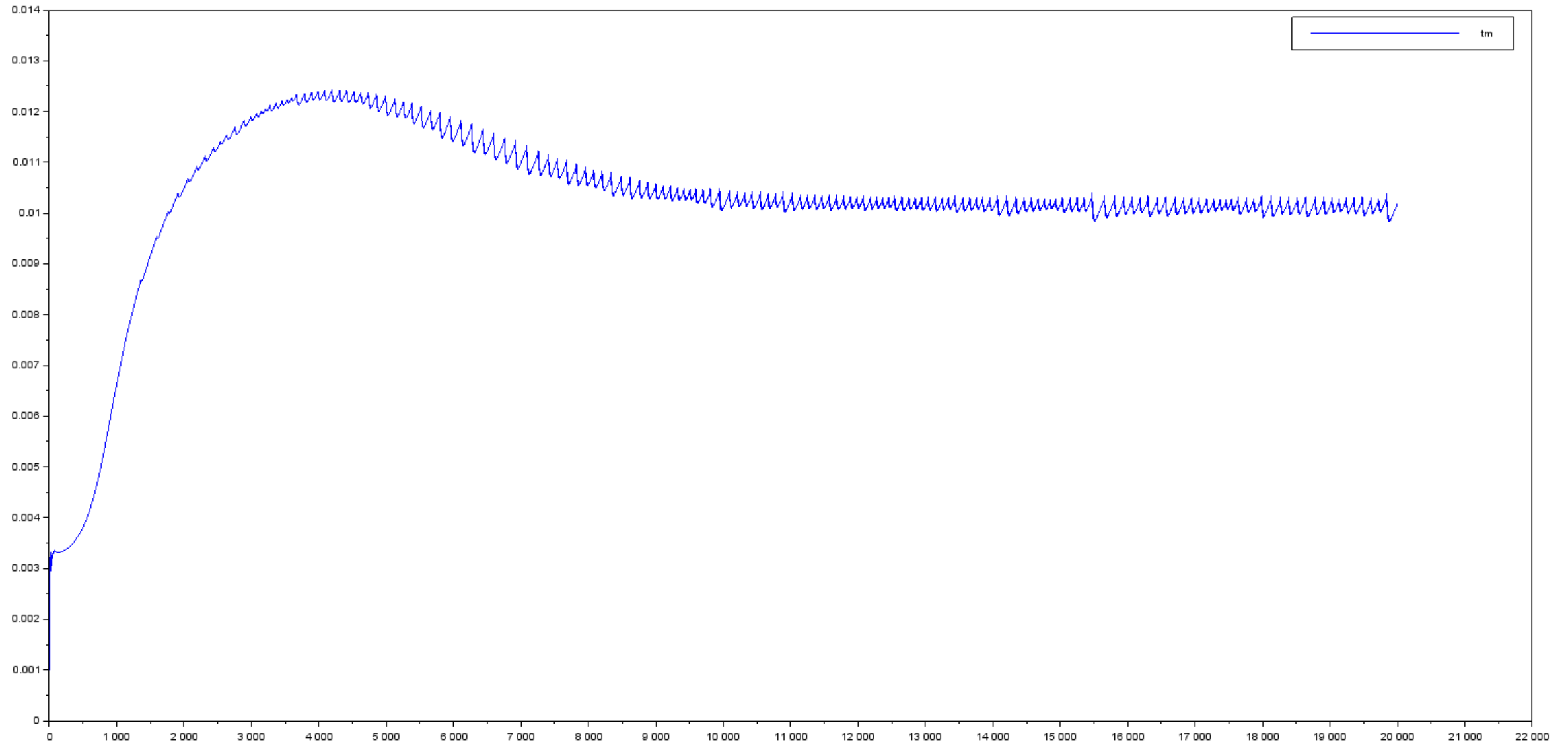
Evolution de la variable tau au fur et à mesure des itérations de l'optimisation



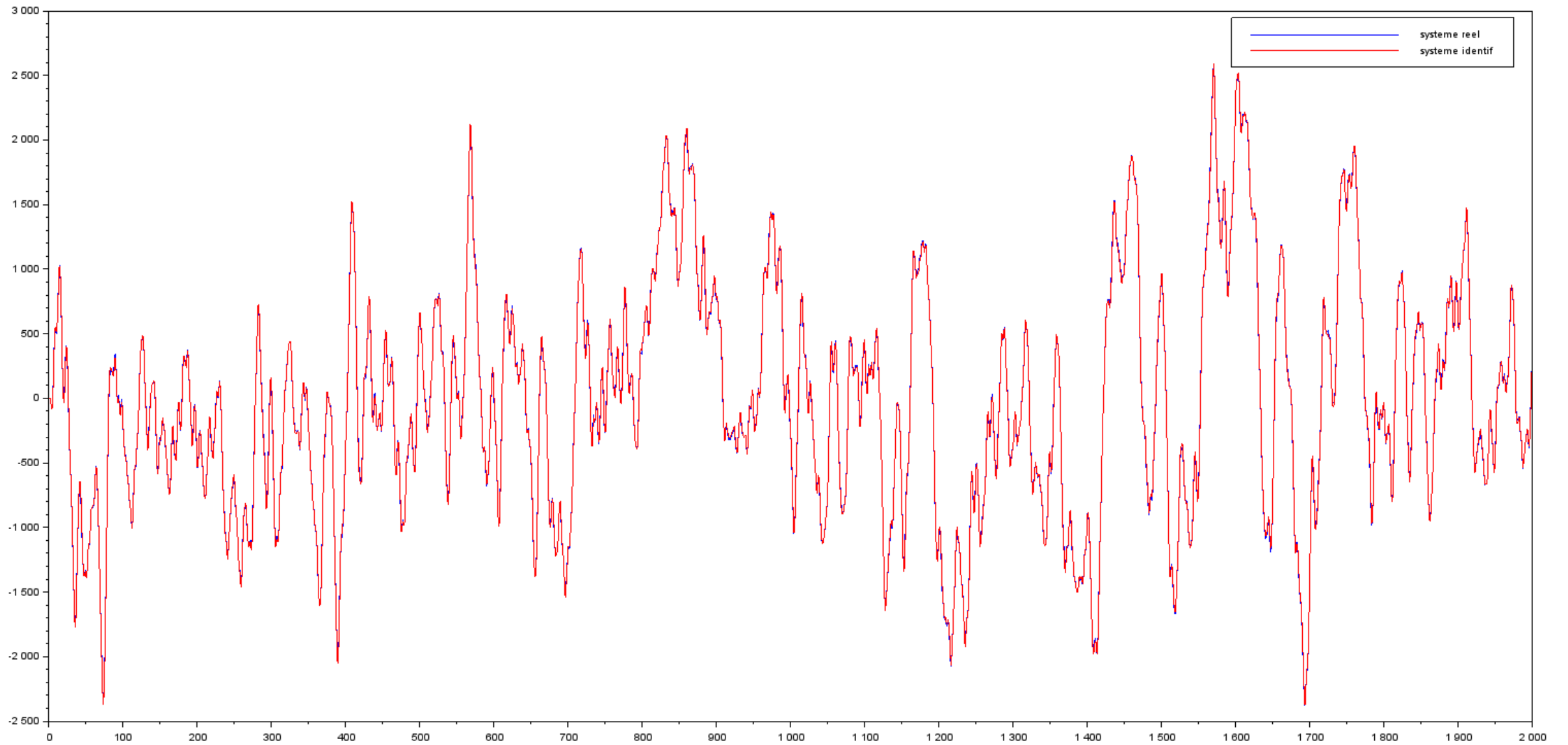
Evolution de la variable xi au fur et à mesure des itérations de l'optimisation



Evolution de la variable tm au fur et à mesure des itérations de l'optimisation



Comparaison de la sortie du système identifié discret avec le système optimisé en continu



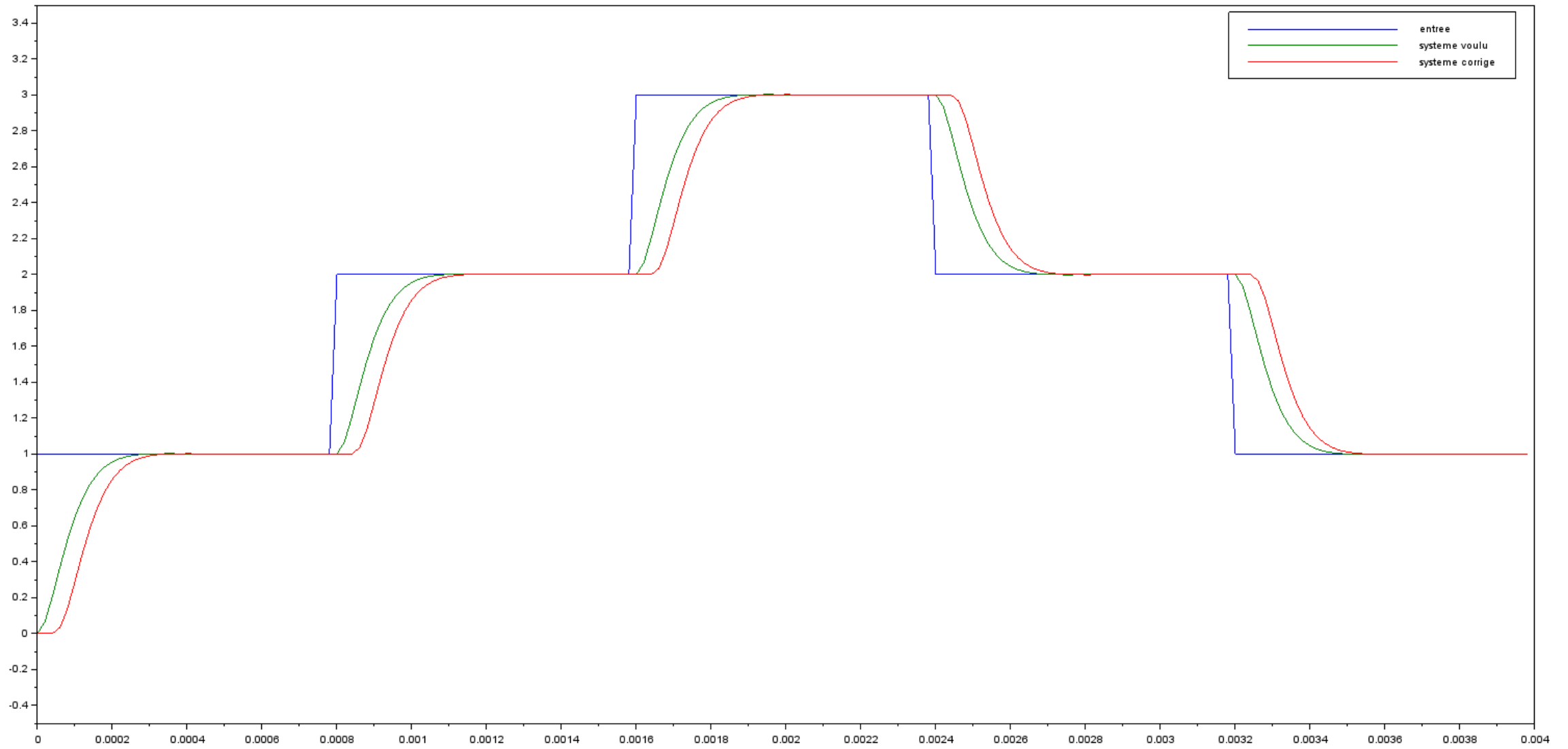
Calcul du RST (avec ou sans intégrateur)

- Conversion du système continu en discret à $T_{e_asservissement}$.
- Calcul des coefficients du système cible en régulation.
- Calcul des coefficients du système cible en poursuite.
- Calcul des coefficients des blocs R, S et T.

Exemple avec le système identifié précédemment asservi avec les paramètres :

- $T_e = 20\mu s$.
- ξ en régulation et poursuite : 0.9.
- Temps de montée en régulation : $300\mu s$.
- Temps de montée en poursuite $300\mu s$.

Reponse systeme asservi sans integrateur

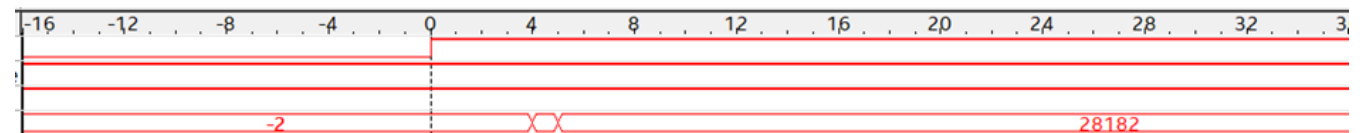


$S1=1.438688$
 $S2=0.827674$
 $R0=-5131.859130$
 $R1=8059.999342$
 $R2=-3210.945133$

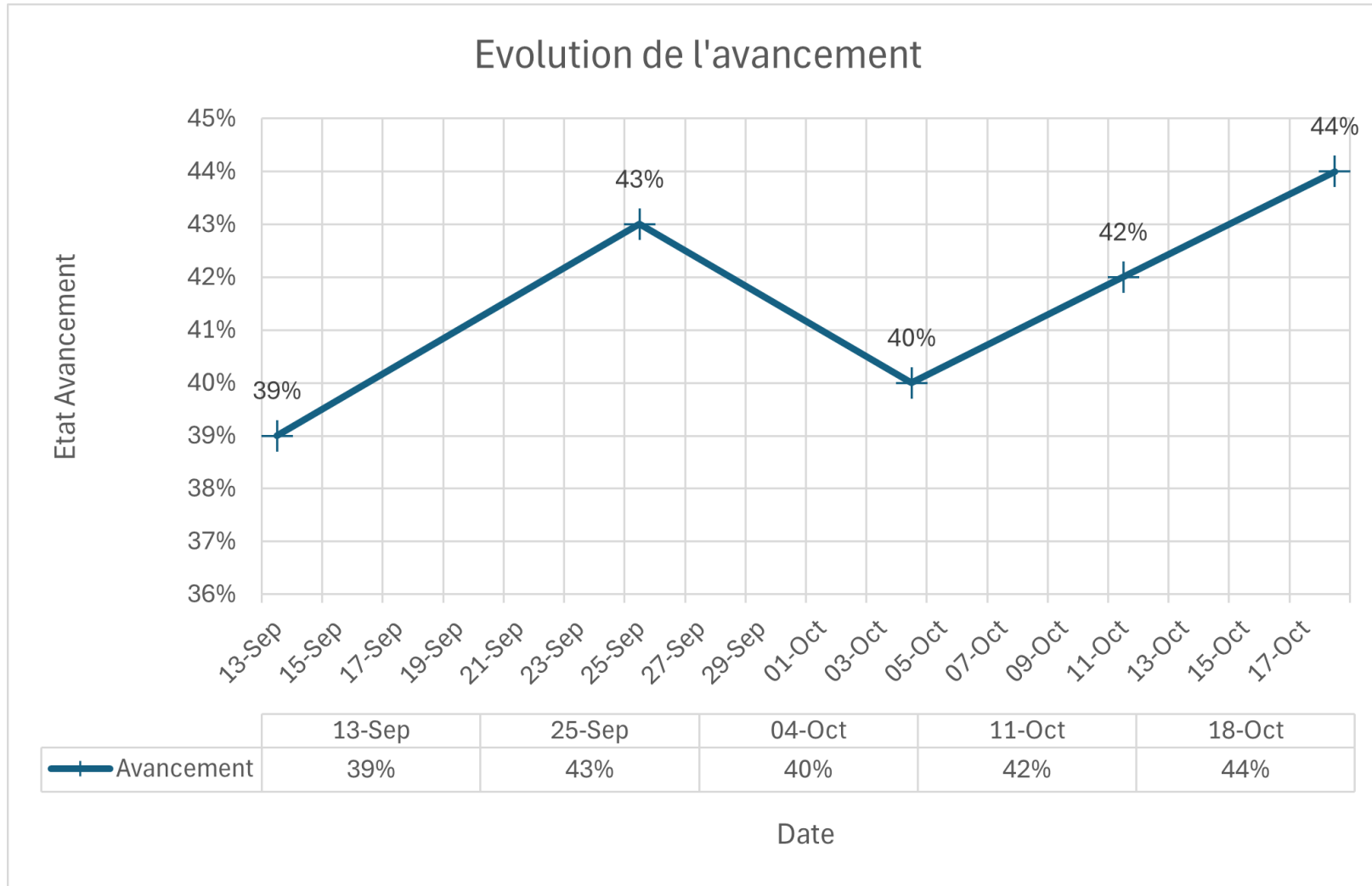
```

Inputs for RST bloc test :
1      1      1      1      1      1      1
Theoretical RST outputs :
0      0      1      -1      28182      -180701
ans  =
      0.
      0.
      1.
     -1.69463
      28182.309

```

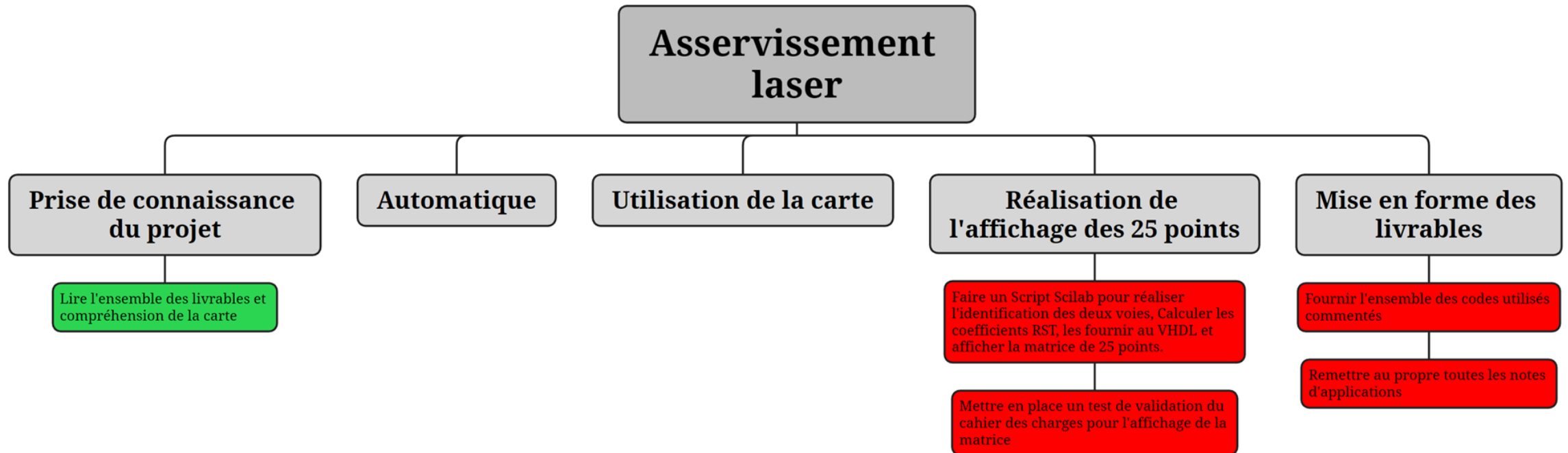


Etat d'avancement



Tâches	Nb d'hrs effectives	Criticité	Heures pondérées	Complétion	Score actuel
Lire l'ensemble des livrables et compréhension de la carte	12	4	48	1	48
Créer et simuler des systèmes d'ordre 1 à 3	8	2	16	1	16
Mettre ces scripts sous forme de fonctions	2	1	2	1	2
Tests des scripts d'identification SBPA de l'année précédente	2	2	4	1	4
Simuler l'identification par la méthode de la pseudo-inverse	8	2	16	1	16
Simuler l'identification par la méthode des Variables Instrumentales	8	2	16	1	16
Simuler l'identification par la fonction ARMAX	8	2	16	0	0
Mettre toutes les méthodes d'identification sous forme de fonctions	2	1	2	0,66	1,32
Trouver les paramètres d'identification optimaux	8	4	32	1	32
Réaliser une série d'acquisition avec les paramètres optimaux	16	4	64	1	64
Réaliser l'identification du système selon les 3 méthodes	16	4	64	0,33	19,2
Réaliser l'identification dans des conditions de pression différentes	4	2	8	0	0
Convertir les coefficients discrets en continus	12	4	48	1	48
Calculer la fonction de transfert cible qui valide le cahier des charges	2	4	8	1	8
Calculer les coefficients d'un correcteur RST avec intégrateur	8	4	32	1	32
Calculer les coefficients d'un correcteur RST sans intégrateur	8	4	32	1	32
Vérifier la stabilité des différents blocs des deux correcteurs	4	4	16	0,5	8
Mettre le calcul des coefficients RST sous forme d'une fonction	4	1	4	0	0
Installer l'ensemble des logiciels/drivers nécessaires au fonctionnement de la carte	8	4	32	1	32
Vérifier la communication via l'USB Blaster pour compiler le programme	1	4	4	1	4
Vérifier la communication via le port série	2	4	8	1	8
Utiliser les scripts de l'année précédente pour communiquer avec la carte	4	4	16	1	16
Corriger le script d'acquisition des données du galvanomètre	12	4	48	1	48
Corriger le déclenchement du galvanomètre Y lors d'une identification du galvanomètre X	12	2	24	0	0
Créer un script scilab pour convertir des nombres flottants en valeurs décimales de leur code hexadécimal	8	2	16	0	0
Créer un Script permettant d'envoyer les coefficients RST dans le VHDL	8	4	32	1	32
Vérification des codes utilisés l'an dernier pour calculer les sorties théoriques du RST	16	4	64	1	64
Refaire le script permettant de calculer les sorties théoriques des blocs RST sur scilab	16	4	64	0	0
Modifier le script de test du RST pour imposer la sortie Y	32	4	128	0	0
Tester les blocs R, S et T incluts dans le VHDL séparément	32	4	128	0	0
Tester l'intégration de l'ensemble pour des valeurs de Y imposées	4	4	16	0	0
Tester les paramètres calculés du RST en conditions réelles pour s'assurer de ne pas avoir de dépacement de valeurs	16	4	64	0	0
Faire un Script Scilab pour réaliser l'identification des deux voies, Calculer les coefficients RST, les fournir au VHDL et afficher la matrice de 25 points	32	4	128	0	0
Mettre en place un test de validation du cahier des charges pour l'affichage de la matrice.	4	4	16	0	0
Fournir l'ensemble des codes utilisés commentés	4	4	16	0	0
Remettre au propre toutes les notes d'applications	32	2	64	0	0
	375		1248		550,52

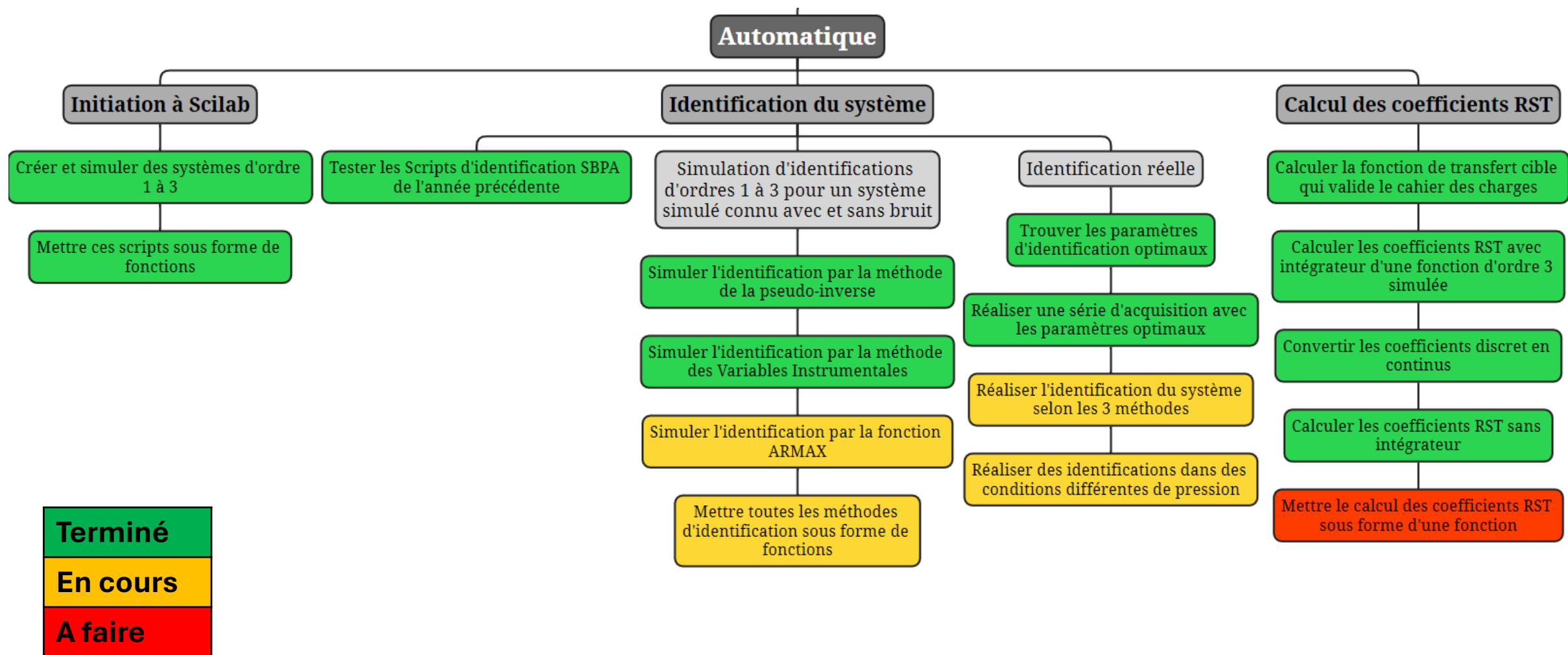
Etat d'avancement



Terminé

En cours

A faire



Utilisation de la carte

Compilation du programme dans la carte

Installer l'ensemble des logiciels/drivers nécessaires au fonctionnement de la carte

Vérifier la communication via l'USB Blaster pour compiler le programme

Récupération des données des galvanomètres

Vérifier la communication via le port série

Utiliser les scripts de l'année précédente pour communiquer avec la carte

Corriger le script d'acquisition des données du galvanomètre

Corriger le déclenchement du galvanomètre Y lors d'une identification du galvanomètre X

Vérification du bloc RST en VHDL

Créer un script Scilab permettant de convertir des nombres flottant en valeur décimales de leurs code hexadécimal

Créer un Script permettant d'envoyer les coefficients RST dans le VHDL

Vérification des codes utilisés l'an dernier afin de calculer les sorties théoriques de chaque bloc du RST

Refaire le script permettant de calculer les sorties théoriques des blocs RST sur Scilab

Modifier le test du script RST pour imposer la sortie Y

Tester les blocs R, S et T incluts dans le VHDL séparément

Tester l'intégration de l'ensemble pour des valeurs de Y imposées

Tester les paramètres calculés du RST en conditions réelles pour s'assurer de ne pas avoir de dépassement de valeurs

Terminé

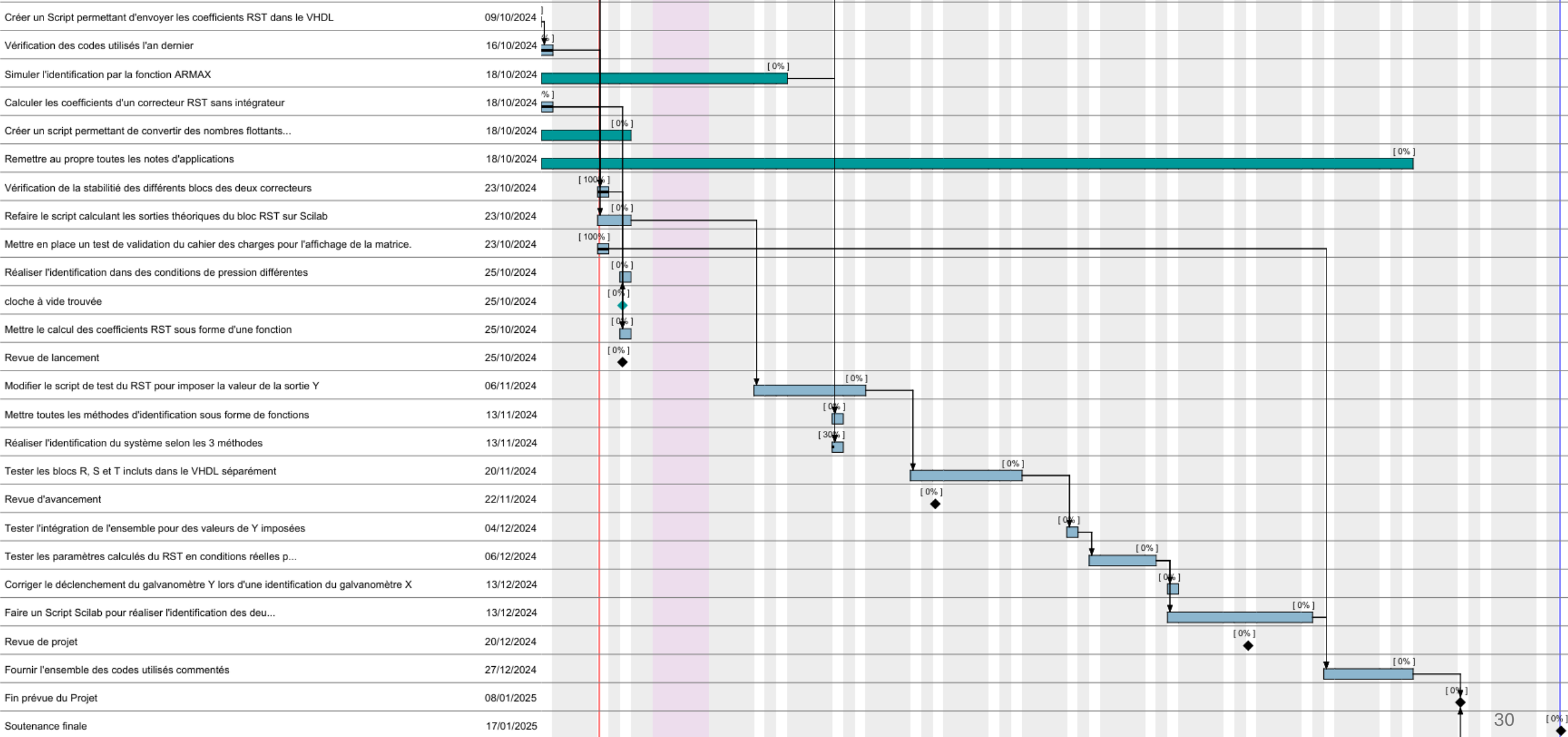
En cours

A faire

Gantt prévisionnel

Nos tâches

Tâches sous-traitées



Merci pour votre attention !