Compte Rendu du TP n°1 Pilote automatique d'un train

Formation : INSTRUMENTATION

Matière : Systèmes asservis

Elaboré par :

MHADHEBI Zied

AIT NOURI Rayane

2024/2025

Introduction :

Dans ce TP nous allons explorer la conception d'un pilote automatique pour un train.

L'objectif est de comprendre comment réguler la vitesse du train en réponse à diverses entrées et perturbations, nous commencerons par analyser le système en boucle ouverte pour évaluer son comportement et sa stabilité.

Ensuite, nous implémenterons un régulateur proportionnel afin d'assurer un contrôle précis de la vitesse, et enfin nous mettrons en place un pilote automatique capable de suivre une trajectoire de référence tout en respectant des contraintes spécifiques

On a la vitesse du train à l'instant t : $y(t) + \tau y^{\cdot}(t) = \beta u(t) + w(t)$.

I. Etude du système à commande (Boucle ouverte) :

1. La fonction de transfert F(p) :

On a déjà $y(t) + \tau y^{\cdot}(t) = \beta u(t) + w(t)$, et on trouve la fonction de transfert entre u et y :

$$F(p) = \frac{Y(p)}{U(p)} = \frac{\beta}{\tau p + 1}$$

Donc : $F(p) = \frac{4}{1.2p+1}$. « On a pris β=4 et τ= 1.2 min ».

On met p=0 dans la fonction de transfert pour trouver le gain statique, et on trouve :

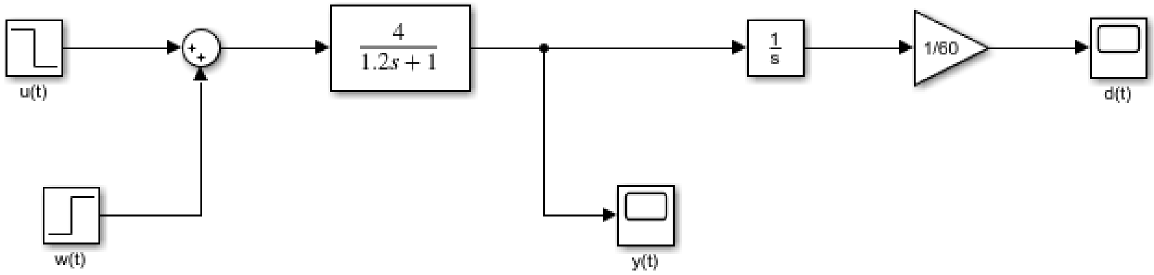
Gain statique = 4

<u>Ce système est stable</u>, car alorsque on met 1.2p+1 = 0, on trouve p=-5/6, donc p<0.

On a aussi réalisé cette fonction de transfert sur simulink.

2. Visualisation de y(t) et d(t) sur SImulink :

On a la distance $d(t) = \frac{1}{60}\int_{s=0}^{t} y(s)\, ds$, et la fonction de transfert qu'on a trouvé dans la question précédente. En utilisant les blocs, step, gain, sum, transfer function et integrator, on peut visualiser sur scope les sorties y(t) et d(t) en utilisant le schéma bloc suivant :



En cliquant sur scope on peut visualiser les sorties y(t) et d(t).

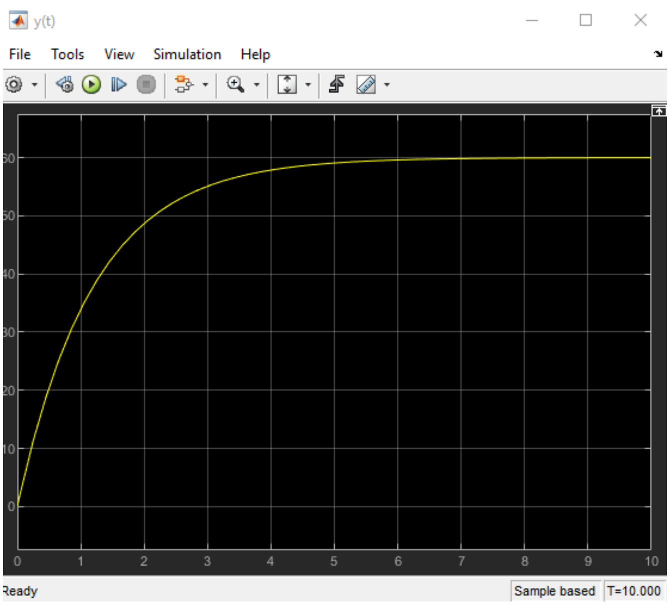


Figure 1 : y(t)

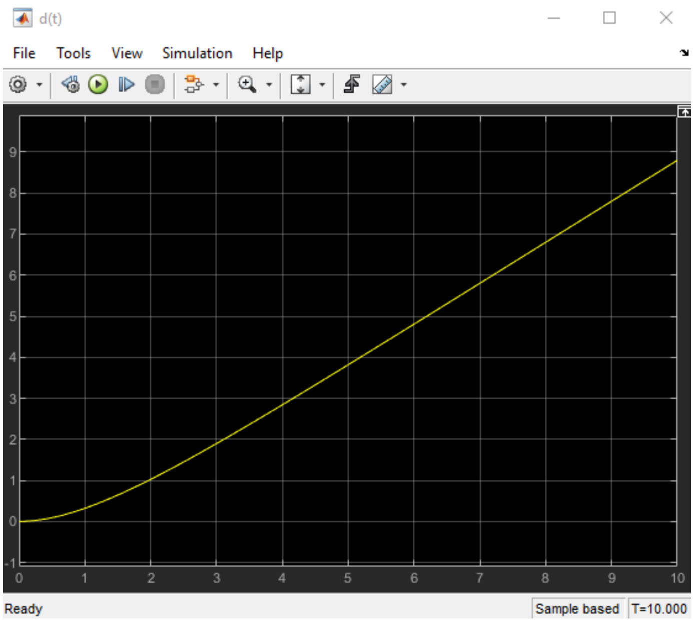


Figure 2: d(t)

-Remarque :

Dans l'énoncé on a :

U(t) = 15 si t < 10 et W(t) = 0 si t < 20

U (t) = 0 si t ≥ 10   W (t) = 10 si t ≥ 20

Il faut introduire ces valeurs sur les blocs *step* de l'entrée u et l'entrée w comme suit :

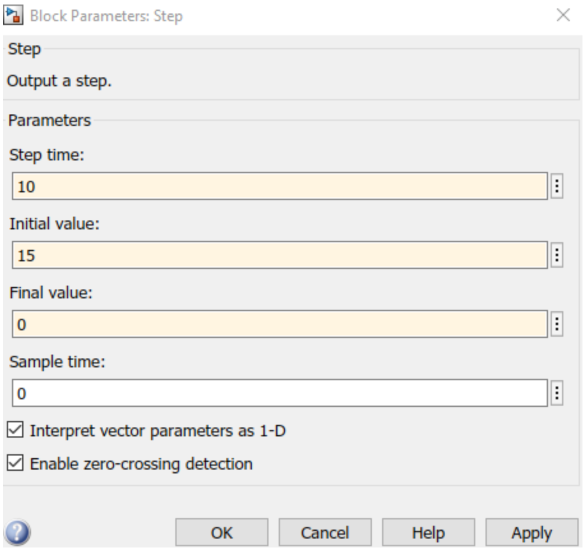


Figure 4 : Paramètres de $u(t)$

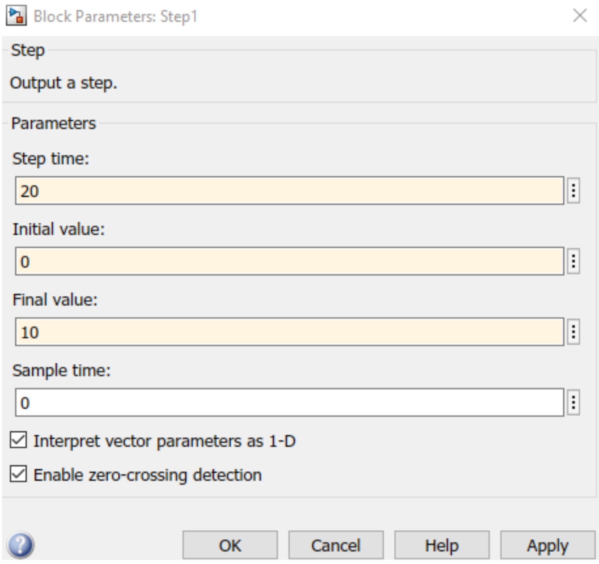


Figure 5 : Paramètres de w(t)

Les trajectoires de y et de d sont conformes aux prédictions théoriques car elles suivent les dynamiques attendues du système en réponse aux entrées échelon.

La sortie y(t) augmente rapidement au début, puis se stabilise autour d'une valeur constante 60, ce qui est typique d'un système du premier ordre soumis à une entrée échelon.

La réponse en sortie y(t)=60 $* \left(1 - e^{-\frac{t}{1.2}}\right) = \beta u(t)(1 - e^{\frac{-t}{\tau}})$

d(t) qui représente la distance parcourue, est l'intégrale de y(t), ce qui explique pourquoi elle augmente de façon continue avant de ralentir lorsque y(t) se stabilise.

⇨ Ces comportements sont cohérents avec les équations théoriques du système.

II. Régulateur proportionnel :

Dans cette partie, on veut implémenter un régulateur proportionnel de la forme :

u(t)=−k(y(t)−αyr(t))=k(αyr(t)−y(t))

1. La fonction de transfert en boucle fermée entre y_r et y :

En calculant $\frac{Y_r}{Y}$ on a trouvé la fonction de transfert :

$$\frac{Y_r}{Y} = \frac{4k\alpha}{1 + \tau p + k\alpha}$$

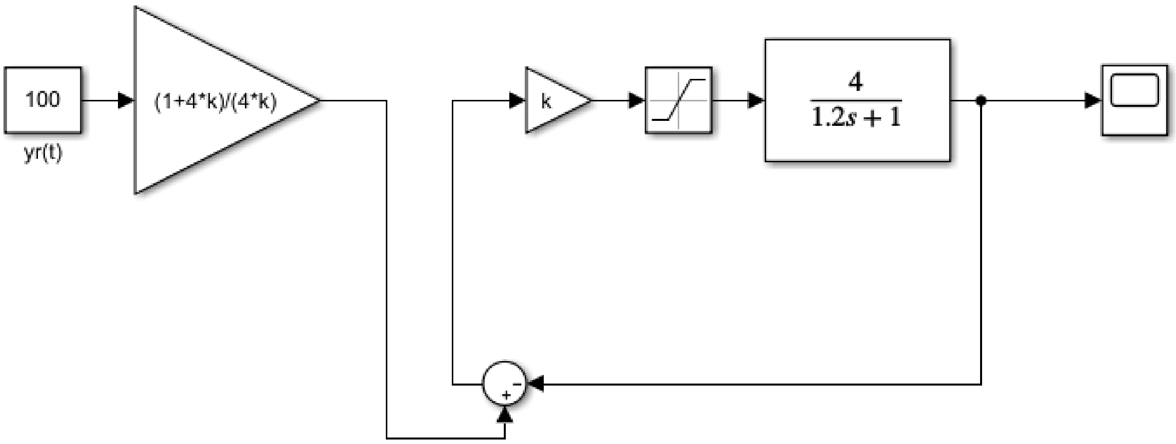
Pour obtenir un gain statique égale a 1, il faut que $\alpha_k = \frac{1+4k}{4k}$

2. Simulation :

Il y a plusieurs conditions à respecter pour faire cette simulation, « on les trouve dans l'énoncé ».

A. Sans perturbation :

Le schéma bloc utilisé :



$y_r(t) est\ une\ constante\ egale$ à $100\ km/h$. Le bloc gain contient $\alpha_k = \frac{1+4k}{4k}$

K est déjà défini dans une variable matlab, et aussi on a appliqué les valeurs

$u_{\min} = -50\ et\ u_{max} = 100$ pour la saturation.

En cliquant sur scope, on peut visualiser sur le même graphe les trajectoires de la commande calculée par le régulateur avant la saturation et de la commande appliquée après la saturation.

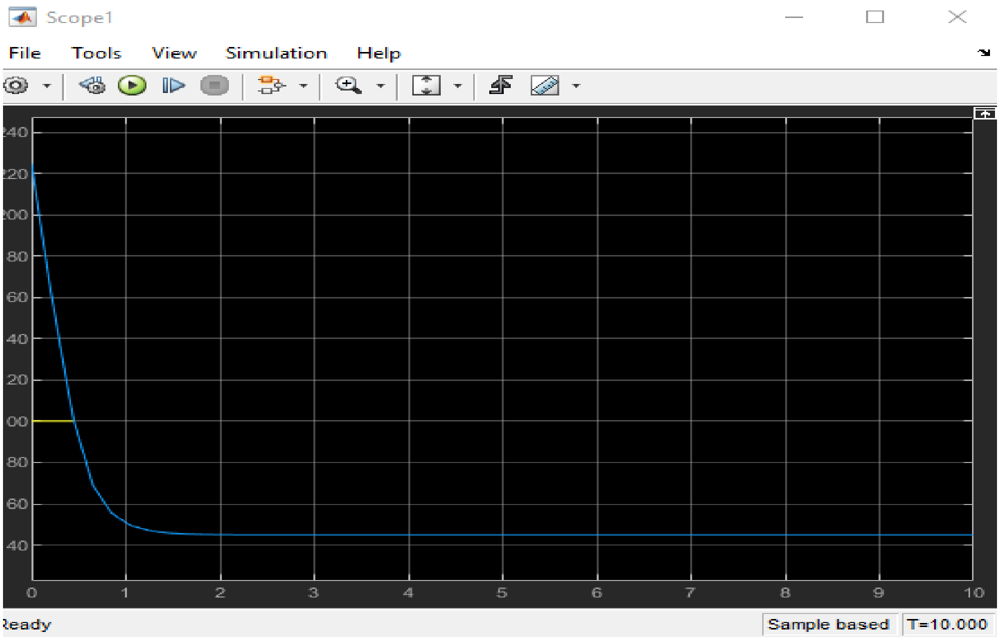


Figure 6 : Y(t) après saturation

Et on obtient aussi y(t) avec k qui permet d'avoir un temps de sortie du régime transitoire à 5% inférieur à 30 secondes. « On a pris k=2 »

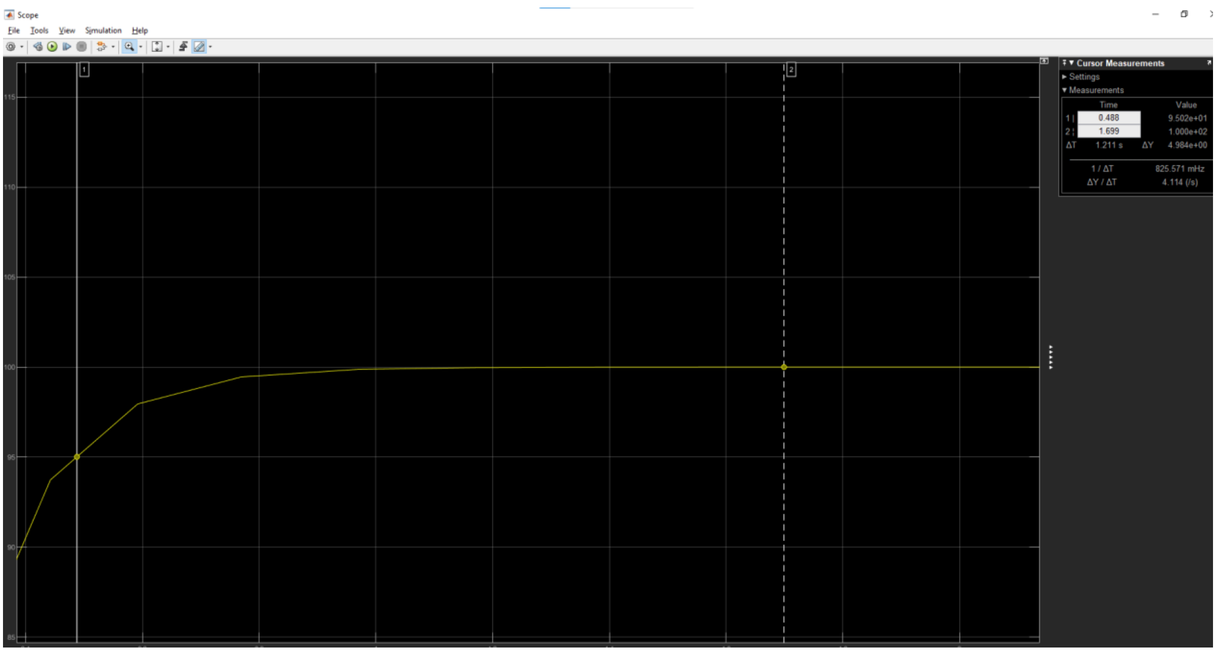
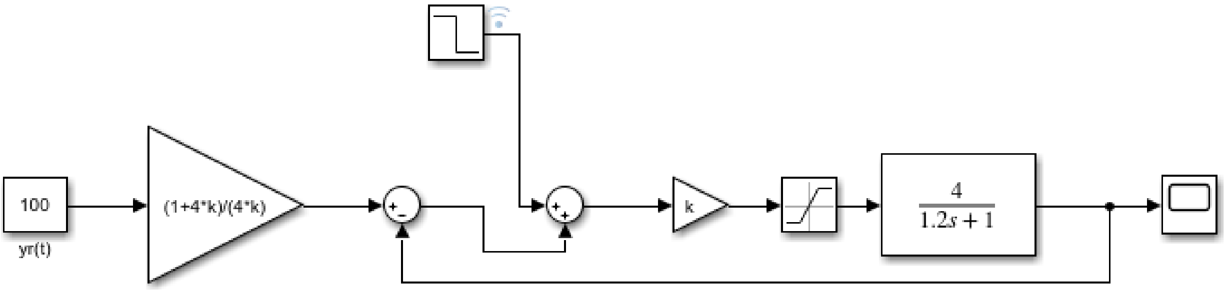


Figure 7 : y(t) sans perturbation

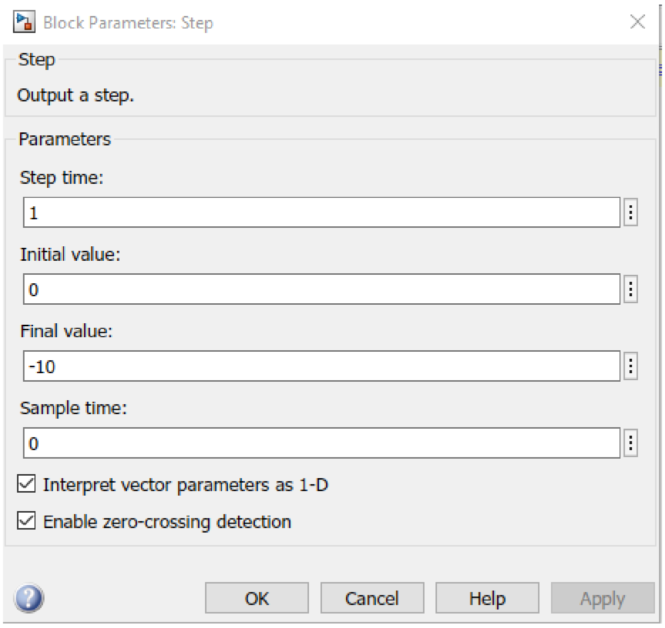
Y(t) et u(t) convergent vers les valeurs prédites.

B. Avec perturbation :

Pour cette partie, nous utilisons le même schéma que dans la question précédente, en ajoutant simplement un bloc qui représente la perturbation w(t). Et on obtient :



Et pour la perturbation, il faut introduire dans ces valeurs dans son bloc,



Pour obtenir w(t)=0 quand t est inférieur à 1 min et w(t)= -10 lorsque t est supérieur ou égal à 1 min.

Apres l'application de ces conditions, et en cliquant sur scope on observe la sortie :

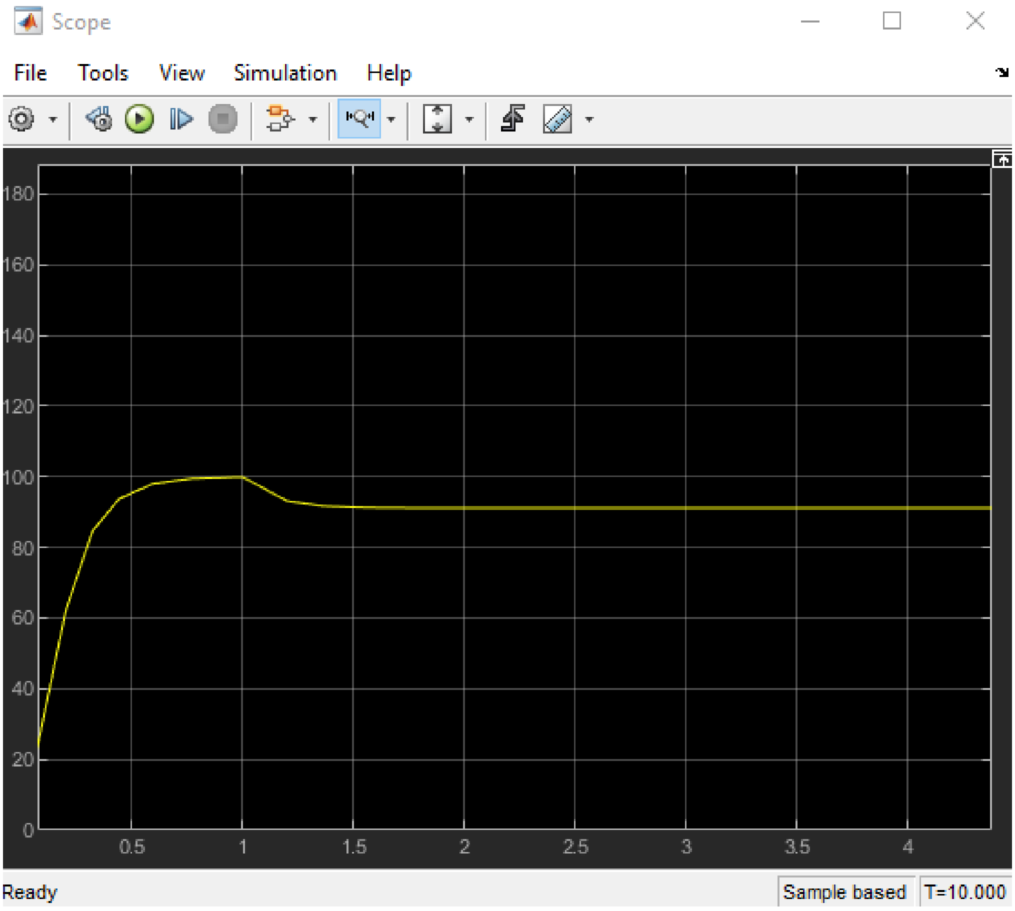


Figure 8 : Sortie après l'application des perturbations

Le comportement observé est conforme aux prédictions théoriques, car la sortie y(t) converge vers la valeur de référence de 100 km/h, ce qui indique que le système atteint et maintient la consigne comme prévu.

De plus, le système compense efficacement la perturbation w(t) introduite après 1 min, et la commande u(t) reste également dans les limites de saturation, ce qui est crucial pour éviter des comportements instables.

Enfin, le temps de réponse mesuré est inférieur à 30 secondes, respectant ainsi les spécifications souhaitées.

III. Pilote automatique :

Nous concevons un pilote automatique pour le train, basé sur le régulateur proportionnel précédemment étudié, afin de suivre une trajectoire de référence d'une durée totale de 70 minutes, définie par le tableau dans le fichier yref.mat, tout en respectant le cahier de charge qu'on trouve dans l'enoncé de tp.

1.a. Trajectoire de référence :

En utilisant la fonction plot sur matlab on peut afficher la trajectoire de référence yref comme suit :

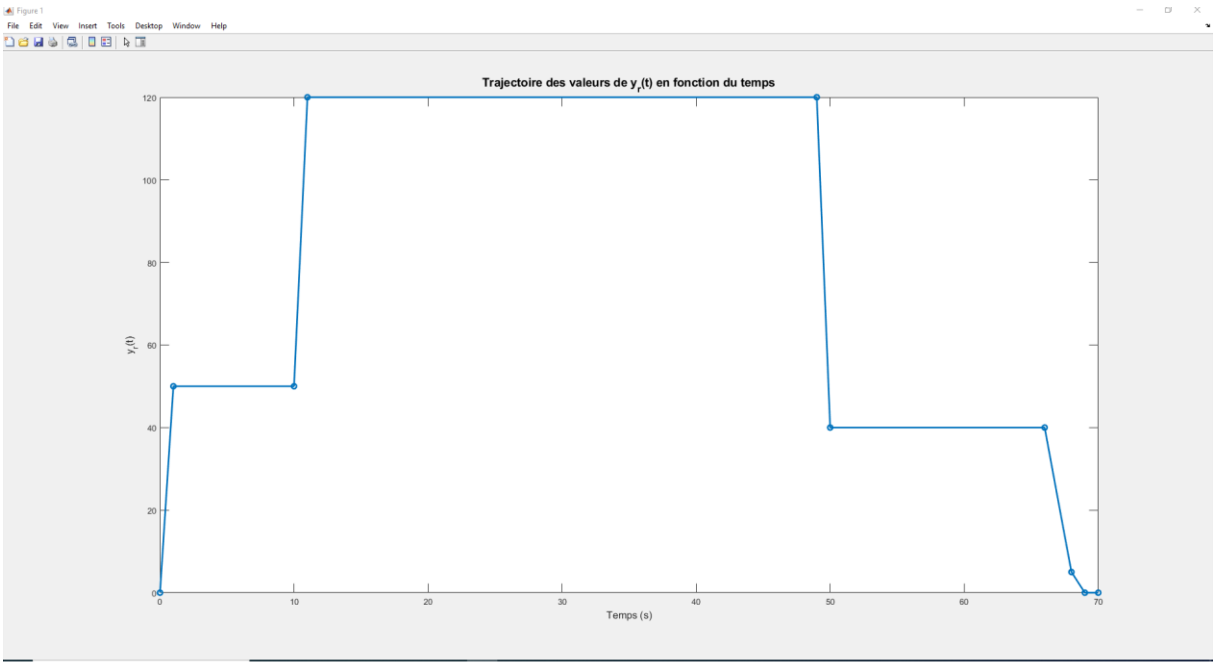
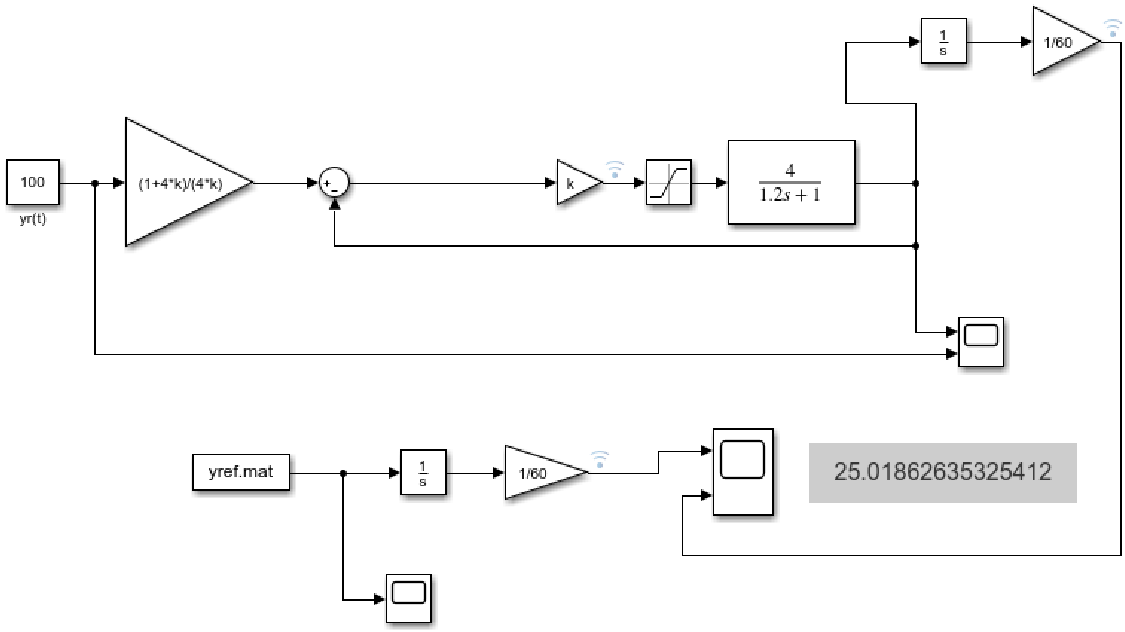


Figure 9 : Visualtion de yref(t)

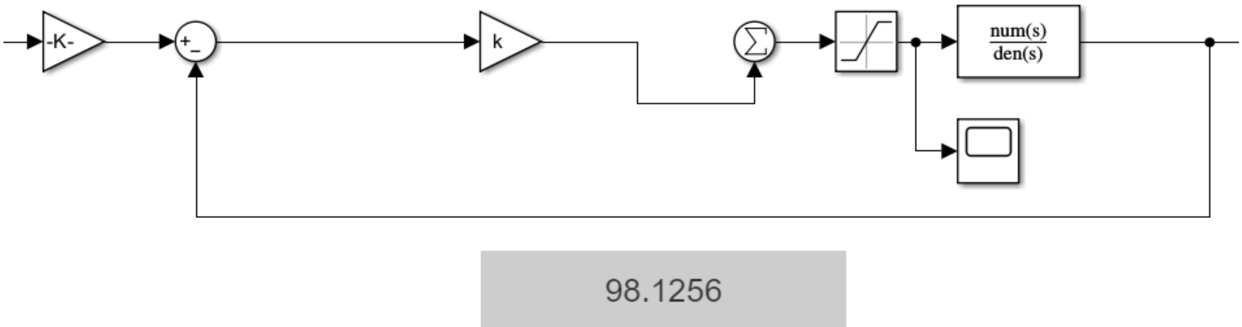
1.b. Simulateur :

Nous construisons un simulateur Simulink pour récupérer et afficher la trajectoire de yr à partir du fichier yref.mat en utilisant un bloc **From File** et pour créer et afficher la trajectoire de référence dr avec un bloc **Integrator**.

La valeur finale de dr(t) sera également affichée à l'aide d'un bloc **Display**.



On peut visualiser la trajectoire yref, ainsi que la valeur finale de dr(t) a partir de scope :



2.a.

Après la modification de schéma-bloc du système asservi de la question II pour appliquer la trajectoire de référence en entrée du régulateur proportionnel comme illustré ci-dessous :

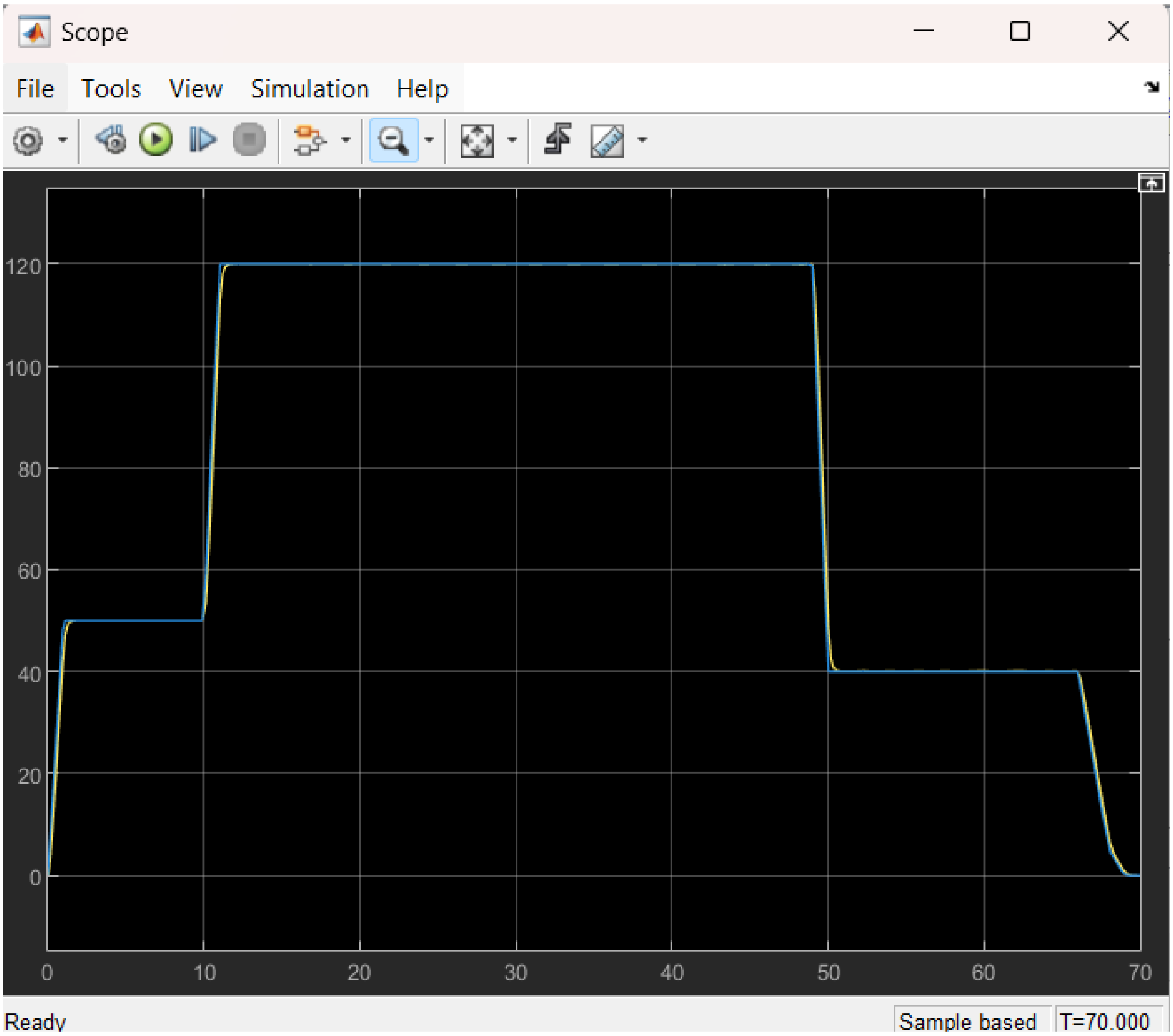


Figure 9 : Visualisation de yr(t) et y(t)

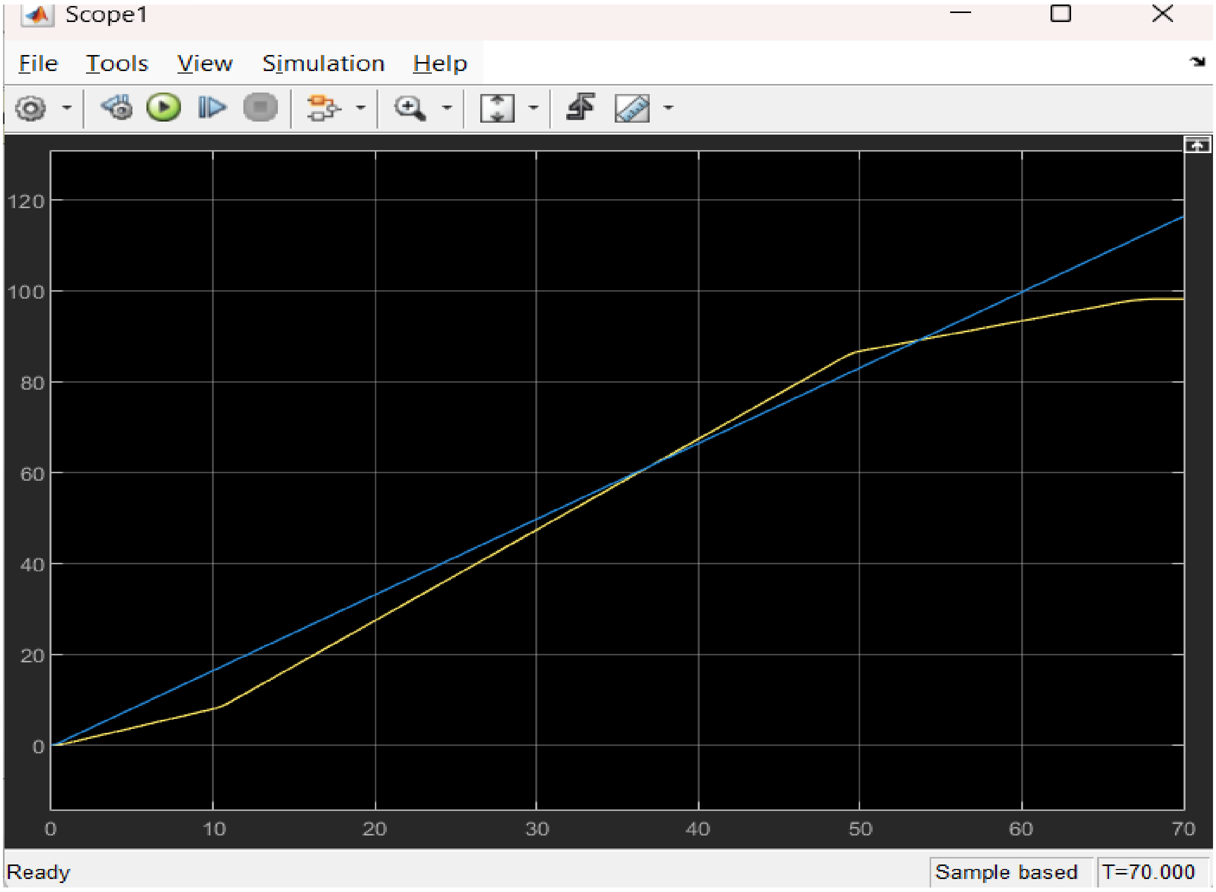
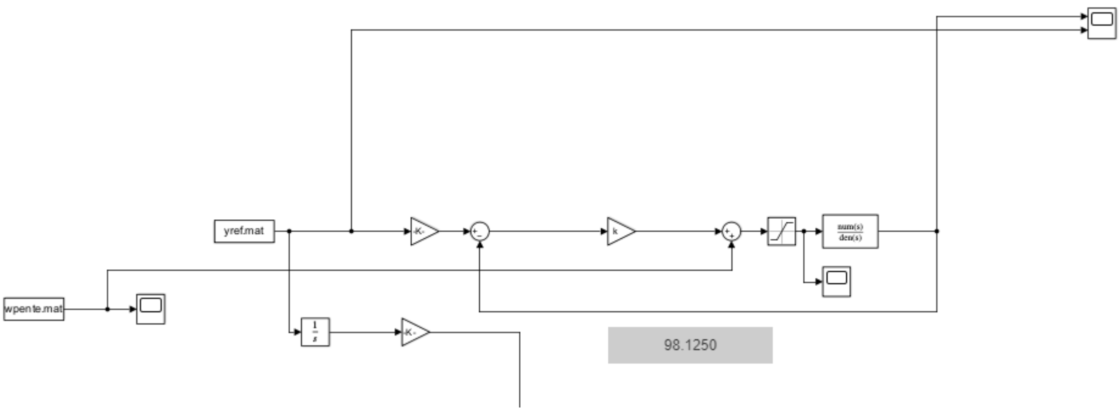


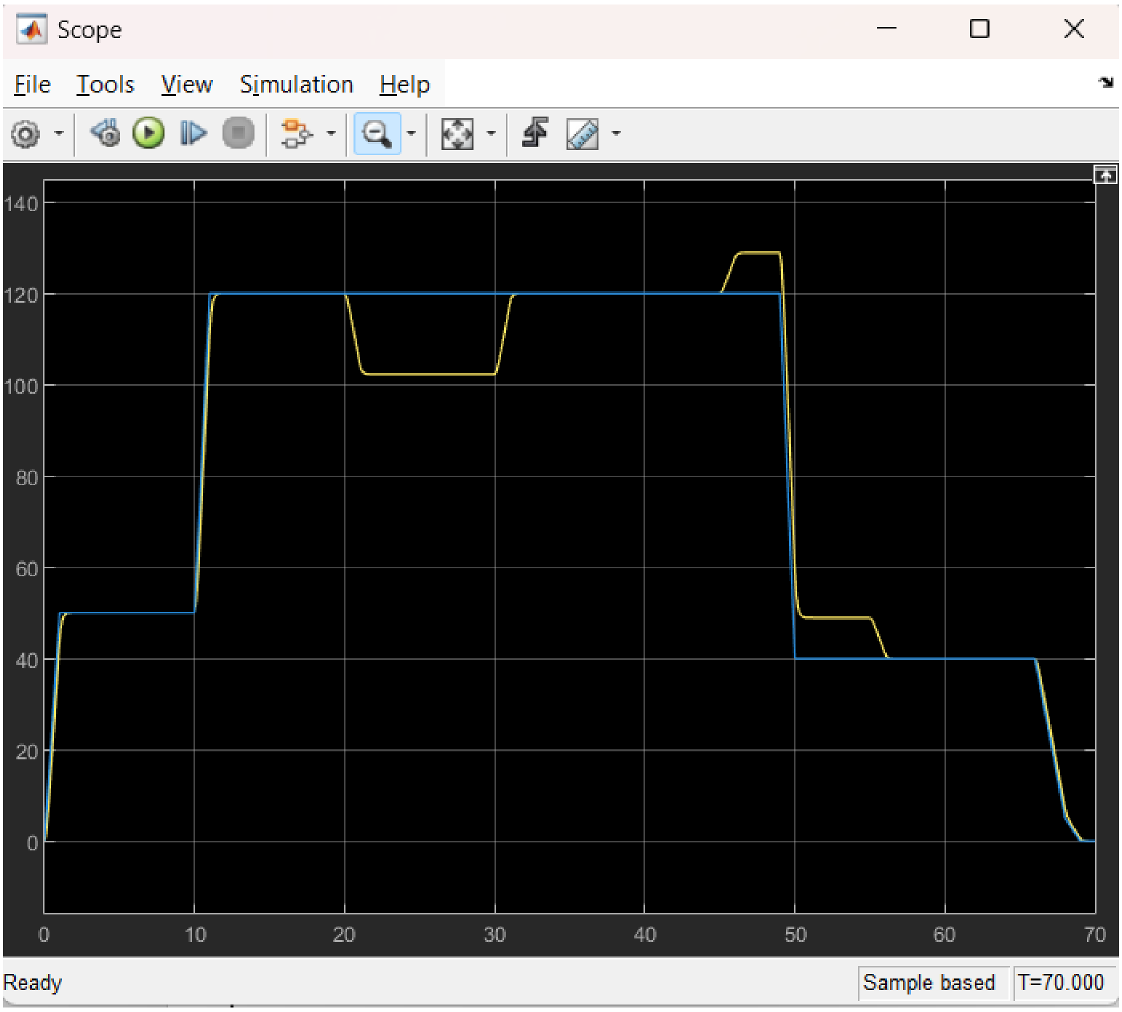
Figure 10 : Graphique de dr (t) et d(t)

2.b.

En ajoutant la perturbation Wp(t) on trouve le schéma bloc ci-dessous :



- On peut satisfaire que la vitesse ne soit jamais négative comme illustré dans la figure ci-dessous :

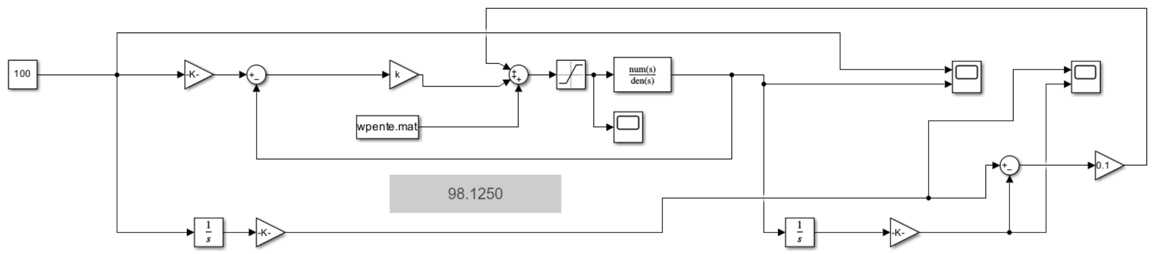


- La valeur absolue de la vitesse finale y (70) doit être inférieure à 1 km/ h.

Mais La valeur absolue de l'erreur de poursuite e (t) = y (t) − yr (t) est n'est pas toujours rester inférieure à 5 km/ h. (non satisfait)

3.a.

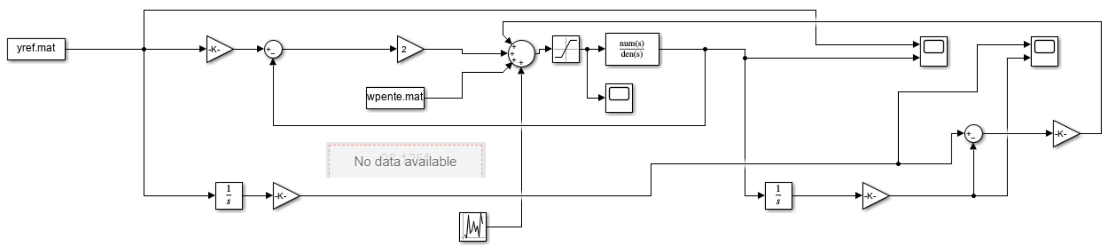
On modifie le régulateur pour ajouter une rétroaction sur l'écart de distance ed(t)=d (t)−dr(t) on obtient le schéma bloc ci-dessous :



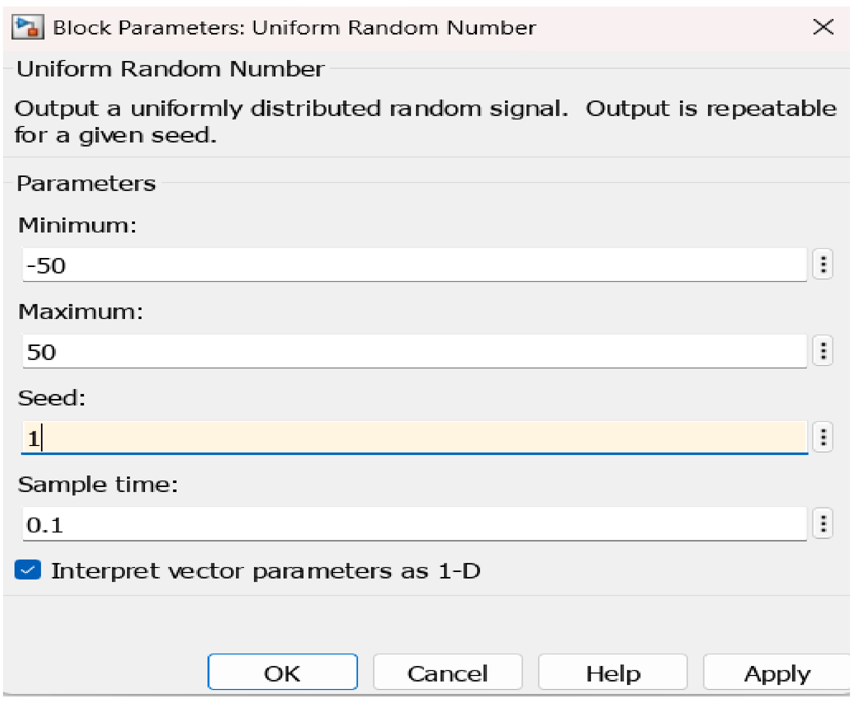
Pour toutes les valeurs de Kd le cahier est n’est pas satisfaite.

3.b.

En ajoutant une perturbation Wv supplémentaire avec le bloc Uniform Random Number on trouve le schéma bloc ci-dessous :

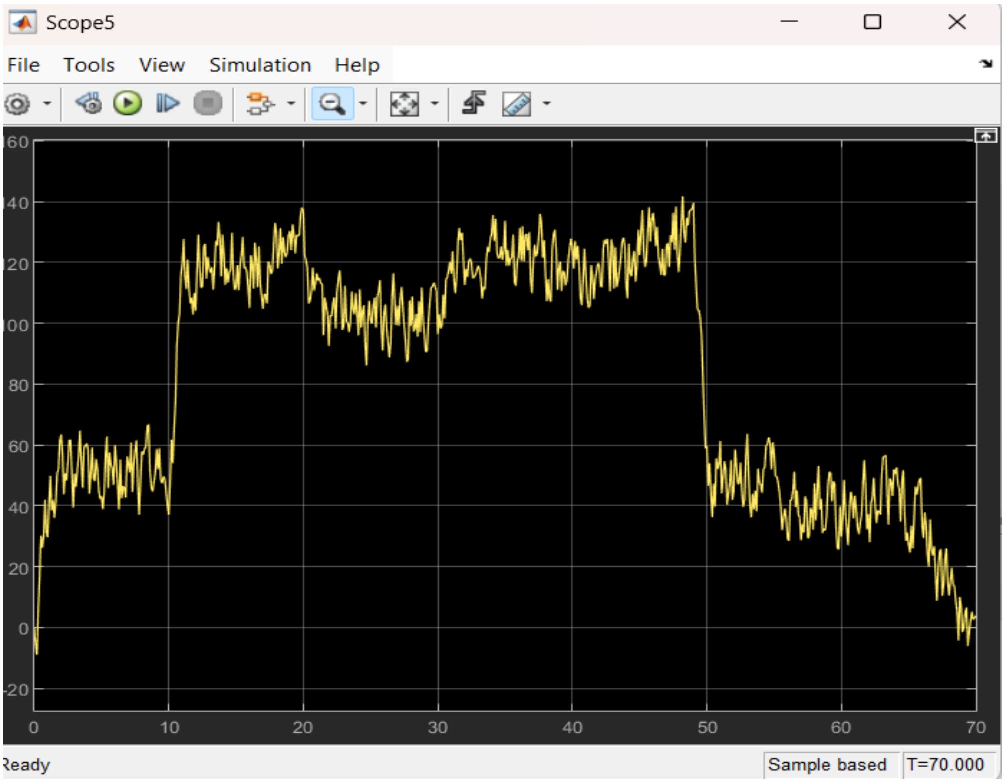


Pour le réglage de bloc Uniform Random Number :

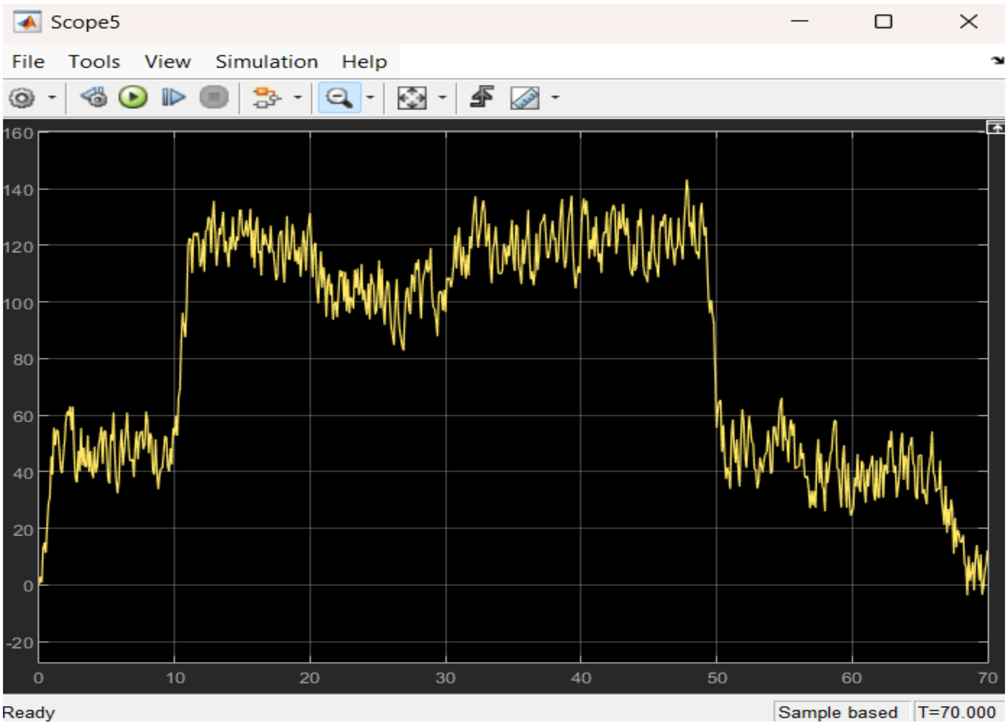


En effectuant plusieurs simulations en modifiant l'initialisation du générateur pseudo-aléatoire (paramètre Seed bloc Uniform Random Number).

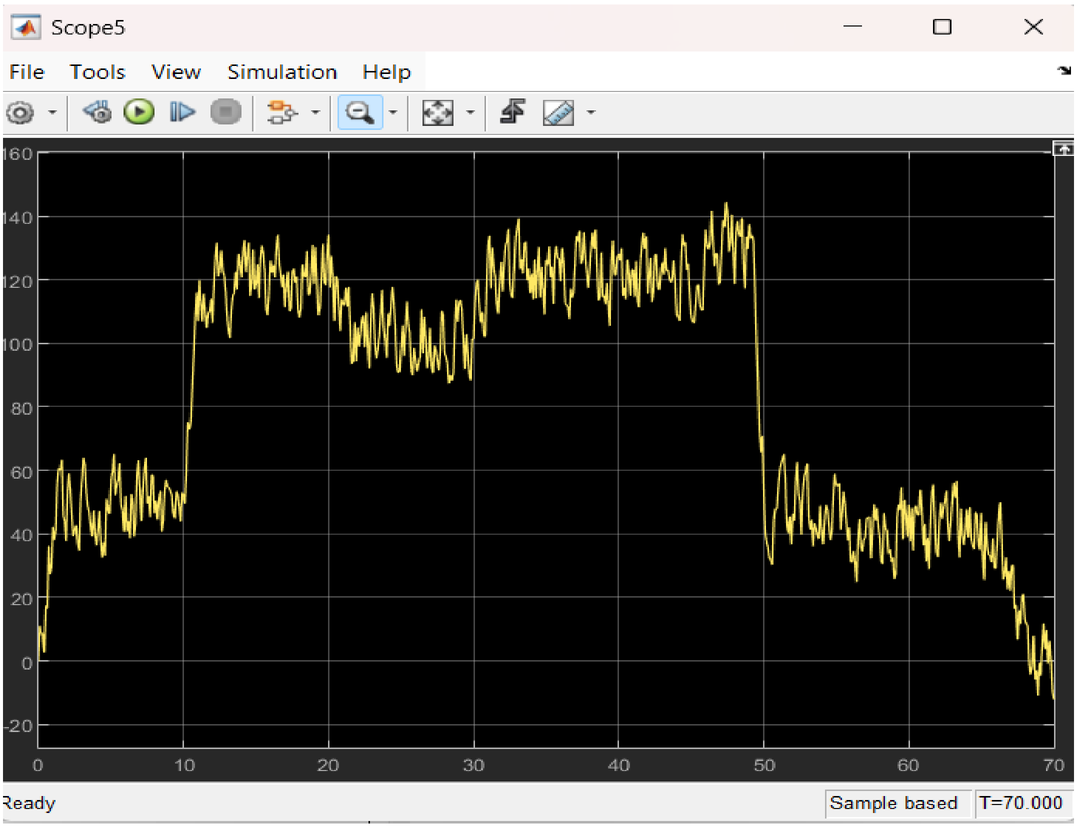
- Pour Seed = 0,1 on obtient la figure ci-dessous



- Pour Seed = 5 on obtient la figure ci-dessous

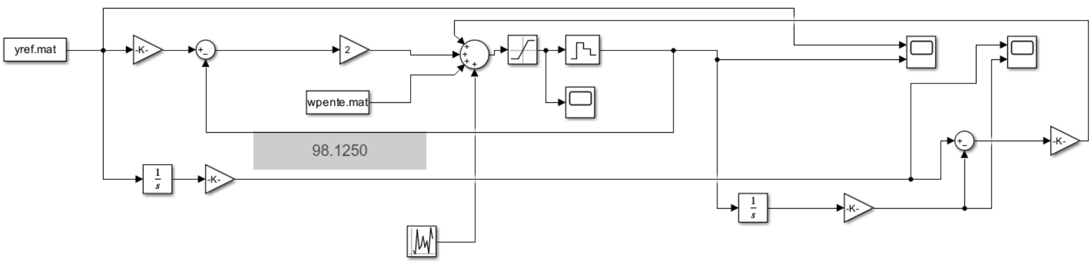


- Pour Seed = 1000 ob obtient la figure ci-dessous :



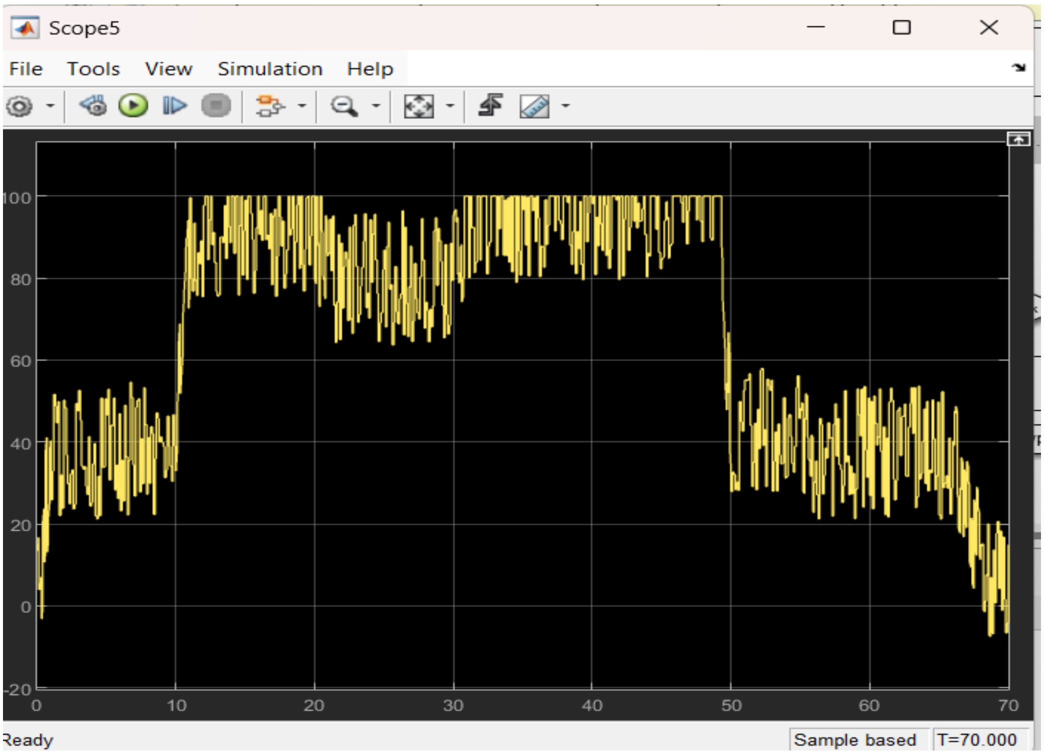
4/

- En remplaçant le régulateur analogique par un régulateur à temps discret, avec une période d'échantillonnage Te = 1/Fe on obtient le schéma bloc ci-dessous :



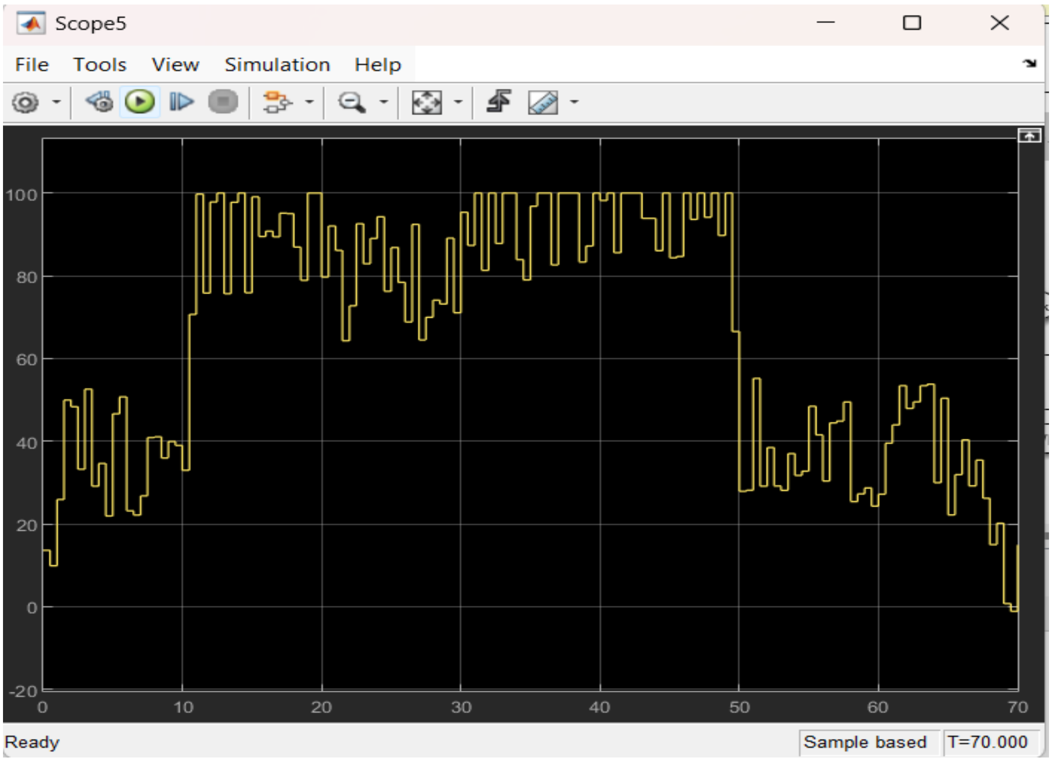
En simulant le comportement du système asservi pour déférentes valeurs de la période d'échantillonnage :

- Pour Te = 0,04

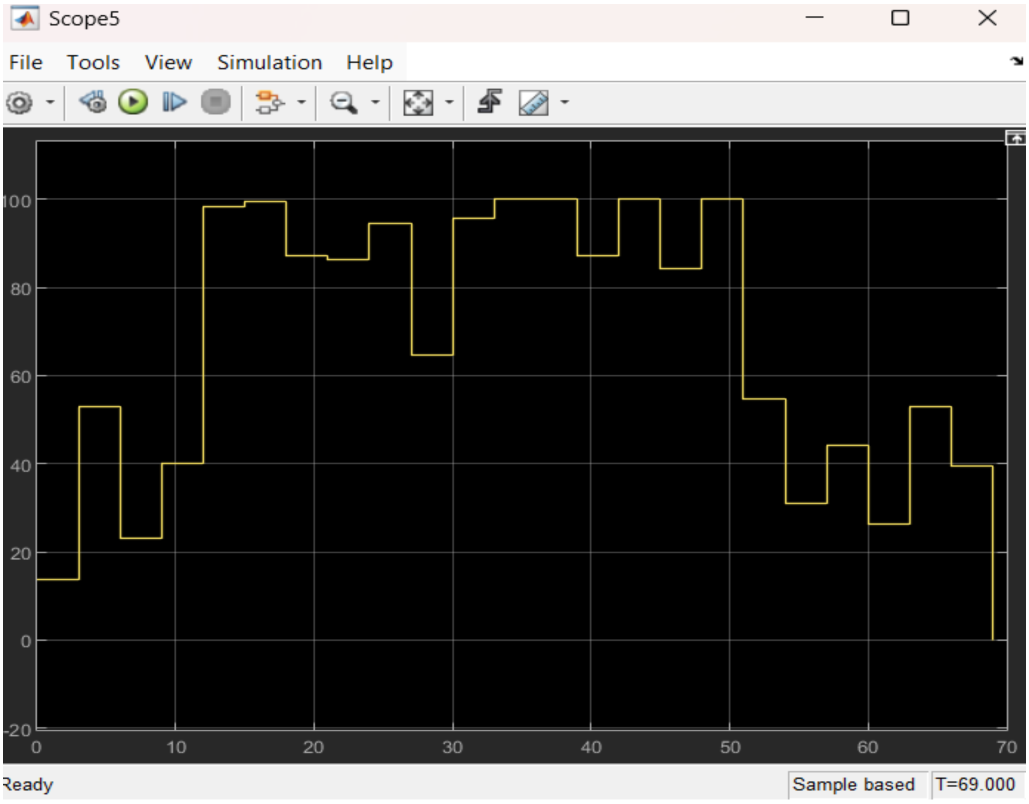


Scope5
File Tools View Simulation Help
100
80
60
40
20
0
-20
0 10 20 30 40 50 60 70
Ready
Sample based
T=70.000

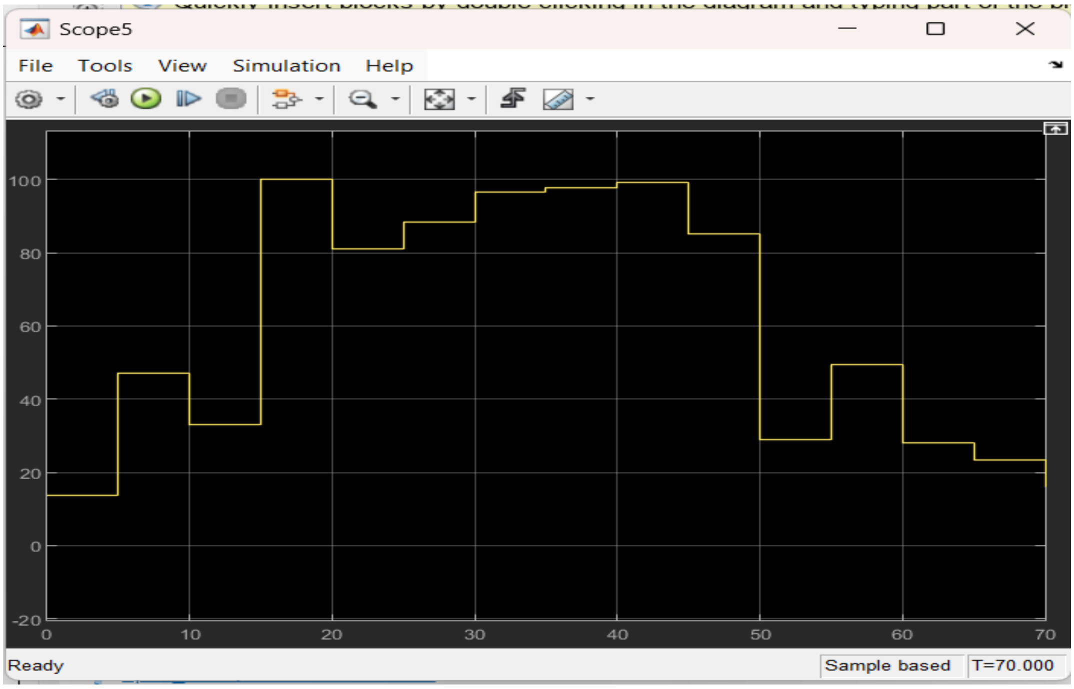
- Pour Te=0,5



- Pour Te=3



- Pour Te=5



Conclusion

En conclusion, ce travail pratique nous a permis d'explorer la conception d'un système de pilotage automatique pour un train.

On a établi la fonction de transfert du système en boucle ouverte et vérifié sa stabilité, en observant comment il réagit aux entrées échelon.

L'utilisation d'un régulateur proportionnel a été efficace pour contrôler la vitesse du train, même avec des perturbations, et grâce à Simulink, nous avons pu visualiser nos résultats théoriques à travers des expériences numériques.

Le développement d'un pilote automatique, capable de suivre une trajectoire de référence, a renforcé notre compréhension des enjeux de la régulation et de la gestion des perturbations, et les résultats obtenus sont conformes aux attentes.