



MASTER INGÉNIERIE DE SYSTÈMES COMPLEXES PARCOURS ROBOTIQUE ET OBJETS CONNECTÉS

AUTOMATIQUE LINÉAIRE DES SYSTÈMES MULTIVARIABLES
TRAVAIL PRATIQUE

Régulation de niveau et de température dans une cuve

Étudiants :

Otmane ATTOU

Avotra RAKOTONDRAVONY

Enseignant :

Jean-François BALMAT

Table des matières

1	Introduction :	2
2	Modélisation et étude du système :	2
2.1	Modélisation du système :	2
2.2	Linéarisation du système :	3
2.3	Modèle d'état :	3
3	Analyse du système :	4
3.1	L'intérêt des bloqueurs d'ordre 0 :	4
3.2	Représentation d'état du système échantillonné :	5
3.3	La stabilité au sens de Lyapunov :	5
3.4	La commandabilité et l'observabilité du système :	5
3.5	Matrice de transfert $H(z)$ et schéma bloc :	6
3.6	Évolution de $x(k)$ pour un échelon $u(k) = [1, 1]^T$:	7
4	Commande par retour d'état :	8
4.1	Les pôles du systèmes en boucle fermée (en échantillonné) :	8
4.2	La matrice de contre-réaction L :	8
4.3	La réponse après la commande par retour d'état :	9
4.4	La matrice du gain K :	9
5	Analyse de la robustesse et de la commande d'état :	11
6	Commande modale (placement de la structure propre) :	12
7	Conclusion :	14

Table des figures

1	Le schéma simplifié du système	2
2	Matrice d'état A_{mat}	4
3	Matrice d'entrée B_{mat}	4
4	Valeurs propres de la solution de l'équation de Lyapunov	5
5	Les valeurs propres du Grammien de commandabilité	6
6	Les valeurs propres du Grammien d'observabilité	6
7	les fonctions de transfert du système	7
8	Matrice de transfert $H(z)$	7
9	Le schéma bloc simplifié du système discret	7
10	L'évolution de la hauteur de fluide pour l'entrée $u(k) = [1, 1]^T$	8
11	L'évolution de la température de fluide pour l'entrée $u(k) = [1, 1]^T$	8
12	La valeur numérique de la matrice de la contre réaction L	9
13	Les réponses à un échelon de consigne à $h_c = 50mm$ et $\theta_c = 1C$	10
14	Résultat obtenu sans K	10
15	Résultat obtenu avec K	10

16	La réponse du système autour le nouveau point de fonctionnement autour $Q_{s0} = 15l/min$ et $\theta_{s0} = 80C$	12
17	Le comportement du système après une commande modale.	13

1 Introduction :

Ce système sert à réguler le niveau et la température d'un bac qui est constitué d'un réservoir de section S et équipé d'une résistance chauffante. La figure (1) illustre son schéma simplifié.

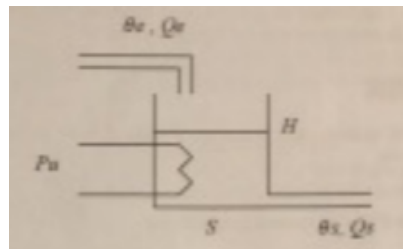


FIGURE 1 – Le schéma simplifié du système

Le but de ce TP sera de disposer un volume constant de fluide à une température désirée. Afin de simplifier le modèle, nous prenons en compte les hypothèses suivantes :

- La température d'arrivée du fluide θ_e est constante.
- Le réservoir est parfaitement calorifugé et sa capacité thermique est négligeable.
- L'échange de chaleur entre la résistance chauffante et le fluide est instantanée.

2 Modélisation et étude du système :

2.1 Modélisation du système :

Afin de modéliser notre système, nous appliquons le principe de la conservation de la masse et de la quantité de chaleur. Nous obtenons :

Bilan volumique :

$$S.H'(t) = Q_e - Q_s$$

Bilan calorimétrique :

$$\theta'_s = \frac{P_u}{\mu.c.S.H} - \frac{\theta_s - \theta_e}{S.H} \cdot Q_e$$

Avec μ correspond à la masse volumique du fluide et c à la chaleur du fluide.

Nous rappelons que $Q_s = \alpha \cdot \sqrt{H}$ qui est la source de non-linéarité de notre système.

2.2 Linéarisation du système :

Nous cherchons à linéariser le système au voisinage du point de fonctionnement $(H_0, P_{u0}, \theta_{s0}, Q_{e0})$. En effectuant un développement de Taylor au premier ordre, nous trouvons :

$$h'(t) = -\frac{Q_{e0}}{2SH_0}h + \frac{1}{S}q_e$$

$$\theta'(t) = -\frac{\theta_{s0} - \theta_e}{SH_0}q_e + \frac{Q_{e0}(\theta_{s0} - \theta_e)}{SH_0P_{u0}}p_u - \frac{Q_{e0}}{SH_0}\theta_s$$

2.3 Modèle d'état :

Nous choisissons le vecteur d'état suivant :

$$X = \begin{pmatrix} h(t) \\ \theta(t) \end{pmatrix}$$

La sortie est :

$$Y = \begin{pmatrix} h(t) \\ \theta_s(t) \end{pmatrix}$$

et l'entrée est :

$$U = \begin{pmatrix} q_e \\ p_u \end{pmatrix}$$

Comme $X' = AX + BU$ et $Y = CX + DU$, et en se basant sur les équations trouvées dans la partie 2.2, nous trouvons :

$$A = \begin{pmatrix} -Q_{e0}/(2 * S * H_0) & 0 \\ 0 & -Q_{e0}/(S * H_0) \end{pmatrix}$$

et :

$$B = \begin{pmatrix} 1/S & 0 \\ (\theta_{s0} - \theta_{e0})/(S * H_0) & (\theta_{s0} - \theta_{e0})/(S * H_0 * P_{u0}) \end{pmatrix}$$

ainsi :

$$C = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Finalement :

$$D = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$$

Afin de calculer les valeurs numériques de matrices de la représentation d'état, nous implémentons dans Matlab les différents paramètres et matrices :

```

1 %Param tres fixe de syst me
2 Qe_0 = 20 ; %en l/min
3 THETAs_0 = 50 ; %en C
4 H_0=0.6 ; %en m
5 Pu_0=20 ; %en KW
6 Qs_0=20 ; %en l/min
7 THETAe=20 ; %en C
8 S=1 ; %en m^2
9
10
11 %%
12 %%Mod lisation du syst me
13
14
15 Amat = [ -Qe_0/(2*S*H_0) 0 ;
16           0 -Qe_0/(S*H_0) ];
17
18 Bmat = [ 1/S 0 ;
19           (THETAs_0-THETAe)/(S*H_0) (THETAs_0-THETAe)/(S*H_0*Pu_0) ];
20
21 Cmat=eye(2);
22
23 Dmat=[0 0;0 0];
24
25
26 sys=ss(Amat,Bmat,Cmat,Dmat);

```

Nous trouvons :

$$\text{Amat} = \begin{bmatrix} -16.6667 & 0 \\ 0 & -33.3333 \end{bmatrix}$$

FIGURE 2 – Matrice d'état Amat

$$\text{Bmat} = \begin{bmatrix} 1.0000 & 0 \\ 50.0000 & 2.5000 \end{bmatrix}$$

FIGURE 3 – Matrice d'entrée Bmat

3 Analyse du système :

3.1 L'intérêt des bloqueurs d'ordre 0 :

Le bloqueur d'ordre zéro ne possède pas un pôle à l'origine, même si le dénominateur de sa fonction de transfert est égal à p .

En conséquence, la présence d'un bloqueur d'ordre zéro devant un système continu, que l'on échantillonne, n'entraîne pas la présence d'un pôle à un dans la fonction de transfert échantillonnée. Et pour cette raison que nous avons préféré de les placer avant les entrées du système.

3.2 Représentation d'état du système échantillonné :

Nous utilisons les commandes suivantes permettant de trouver la représentation d'état du système échantillonné :

```
1 %Représentation d'état du système échantillonné
2 sys_d = c2d(sys,T);
3 [Ad,Bd,Cd,Dd] = ssdata(sys_d);
```

3.3 La stabilité au sens de Lyapunov :

La première étape permettant d'étudier la stabilité au sens de Lyapunov consiste à résoudre l'équation de Lyapunov discrète suivante :

$$A^T X A - X + Q = 0$$

La commande suivante permet de résoudre cette équation d'inconnue X que nous noterons Xlyap, puis nous cherchons ses valeurs propres :

```
1 Xlyap = lyap(Ad,Bd,Cd); %Solve Lyapunov Equation
2 valeurspropresXlyap = eig(Xlyap);
```

Et nous trouvons :

```
valeurspropresXlyap =

-13.3333
-16.6667
```

FIGURE 4 – Valeurs propres de la solution de l'équation de Lyapunov

Ces valeurs propres sont strictement négatives, donc le système est asymptotiquement stable.

3.4 La commandabilité et l'observabilité du système :

Nous utilisant le calcul du Grammien pour analyser la commandabilité et l'observabilité du système. Nous notons le Grammien de commandabilité par Wc et le Grammien d'observabilité par Wo. Ensuite, nous calculons les valeurs propres de ces deux derniers.

```
1 Wc = gram(sys_d,'c');
2 Wo = gram(sys_d,'o');
3
4 valeurspropresWc = eig(Wc);
5 valeurspropresWo = eig(Wo);
```

```
>> valeurspropresWc

valeurspropresWc =

    0.0000
    2.2592
```

FIGURE 5 – Les valeurs propres du Grammien de commandabilité

```
valeurspropresWo =

    1
    1
```

FIGURE 6 – Les valeurs propres du Grammien d'observabilité

Nous remarquons que le grammien de commandabilité n'est pas strictement défini positif, donc nous ne pouvons pas conclure sur sa commandabilité par cette méthode. Pourtant, le grammien d'observabilité est bien défini positif, donc le système est observable.

3.5 Matrice de transfert $H(z)$ et schéma bloc :

Comme il s'agit d'un système MIMO, nous avons préféré de nommer les entrées, les sorties et les états pour avoir bien les différentes fonctions de transfert, puis nous calculons la matrice de transfert par la formule suivante :

$$H = C(sI_n - A)^{-1}B + D$$

```
1 % %Matrice de transfert
2 states = {'h' 'theta' };
3 inputs = {'qe' 'pu' };
4 outputs = {'h' 'theta' };
5
6 sys = ss(Amat,Bmat,Cmat,Dmat,'statename',states,...
7         'inputname',inputs,...
8         'outputname',outputs);
9
10 TF = tf(sys);
11 H=Cmat*inv(s*eye(2)-Amat)*Bmat+Dmat;
```

Nous obtenons :

C'est ainsi que nous construisons le schéma bloc présenté dans la figure 9.

```
TF =

From input "qe" to output...
h: -----
      1
      s + 16.67

      50
theta: -----
      s + 33.33

From input "pu" to output...
h: 0

      2.5
theta: -----
      s + 33.33

Continuous-time transfer function.
```

H =

```
[      3/(3*s + 50),      0]
[ 150/(3*s + 100), 15/(2*(3*s + 100)) ]
```

FIGURE 8 – Matrice de transfert H(z)

FIGURE 7 – les fonctions de transfert du système

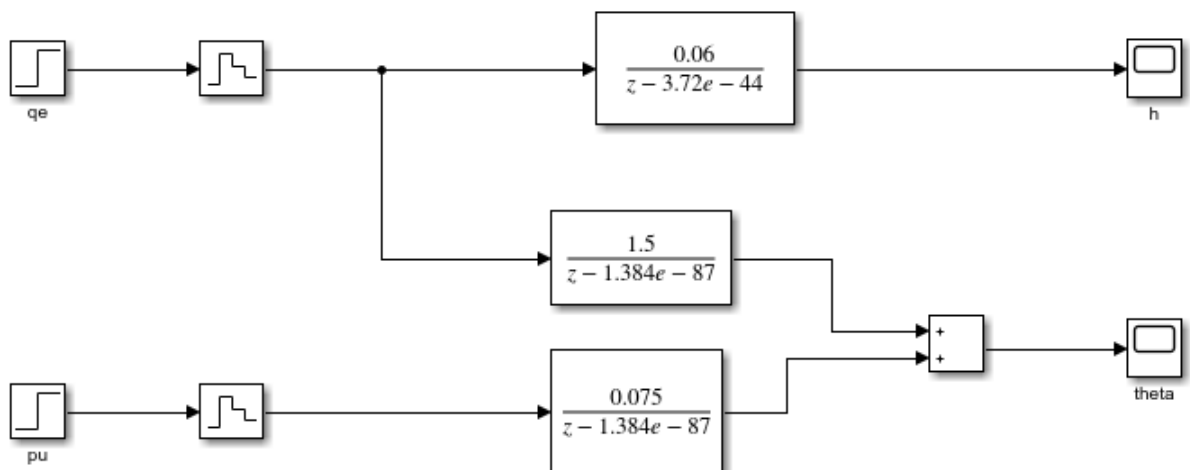


FIGURE 9 – Le schéma bloc simplifié du système discret

3.6 Évolution de $x(k)$ pour un échelon $u(k) = [1, 1]^T$:

Comme les sorties correspondent aux éléments de vecteur d'état, nous plaçons des scope à la sortie pour les visualiser. Nous obtenons :

Nous remarquons que le système est très rapide, ceci est due principalement aux constantes du temps que nous pouvons négliger.

4 Commande par retour d'état :

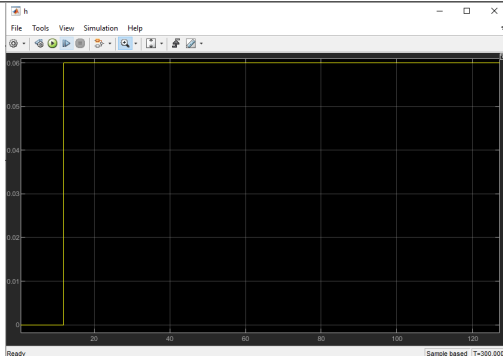


FIGURE 10 – L'évolution de la hauteur de fluide pour l'entrée $u(k) = [1, 1]^T$

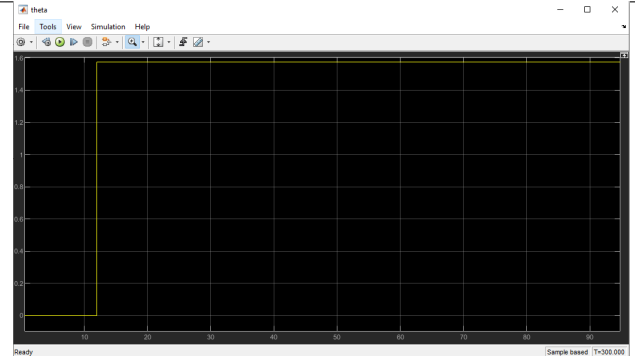


FIGURE 11 – L'évolution de la température de fluide pour l'entrée $u(k) = [1, 1]^T$

4 Commande par retour d'état :

Nous souhaitons modifier le comportement du système. Pour y faire, nous réalisons une commande par retour d'état du type :

$$u(k) = Kv(k) - Lx(k)$$

4.1 Les pôles du systèmes en boucle fermée (en échantillonné) :

La première modification concerne le comportement dynamique du système. Nous voulons que le système en boucle fermée ait comme constantes de temps : $\tau_1 = 16min$ et $\tau_2 = 15min$. Ainsi, les pôles du système en boucle fermée (en échantillonné) sont :

```
1 tau1 = 16 ; %en min
2 tau2 = 15 ; %en min
3
4 %les poles du systeme en boucles fermees sont
5 pole1 = -1/tau1 ;
6 pole2 = -1/tau2 ;
7 poles = [pole1 pole2] ;
```

4.2 La matrice de contre-réaction L :

Comme nous avons calculé la matrice d'état et la matrice d'entrée en échantillonné A_d et B_d , et nous connaissons les pôles voulues, nous pouvons avoir la matrice de la contre-réaction L par la commande suivante :

```
1 L = place (Ad,Bd,poles) ;
```

Nous obtenons :

```
L =  
  
    1.0417    0  
   -20.8333    0.8889
```

FIGURE 12 – La valeur numérique de la matrice de la contre réaction L

4.3 La réponse après la commande par retour d'état :

En prenant les C.I. nulles, et un échelon $h_c = 50mm$ et $\theta_c = 1C$ que nous implémentons dans le code correspondant :

```
1 % reponse du systeme      hc=50mm et thetaC  
2 Ac = (Ad - Bd*L);  
3 sys1 = ss(Ac,Bd,Cd,Dd);  
4 t = [0:0.1:100];  
5 u = [0.05*ones(size(t));...  
6      1*ones(size(t))];  
7 figure(1);  
8 lsim(sys1,u,t);
```

Nous obtenons les réponses suivantes :

A $t = 3\tau$, $3.16minutes = 48min$ et $3.15minutes = 45minutes$, nous avons 95% des valeurs finales.

4.4 La matrice du gain K :

Nous calculons la matrice de gain K en calculant :

$$K = \lim_{x \rightarrow 0} H_{BF} = KC'(-A + BK)^{-1}B = 1 \Rightarrow K = \frac{1}{-C(A - BK)^{-1}B}$$

Ce qui nous traduisons sur matlab par :

```
1 %% matrice de gain K  
2 K = inv(-Cd*inv(Ad-Bd*L)*Bd);  
3 v =K*u;  
4 figure(2);  
5 lsim(sys1,v,t);
```

Nous visualisons les résultats obtenus :

En conclusion, Nous avons bien un gain unitaire car valeurs finales valent les entrées.

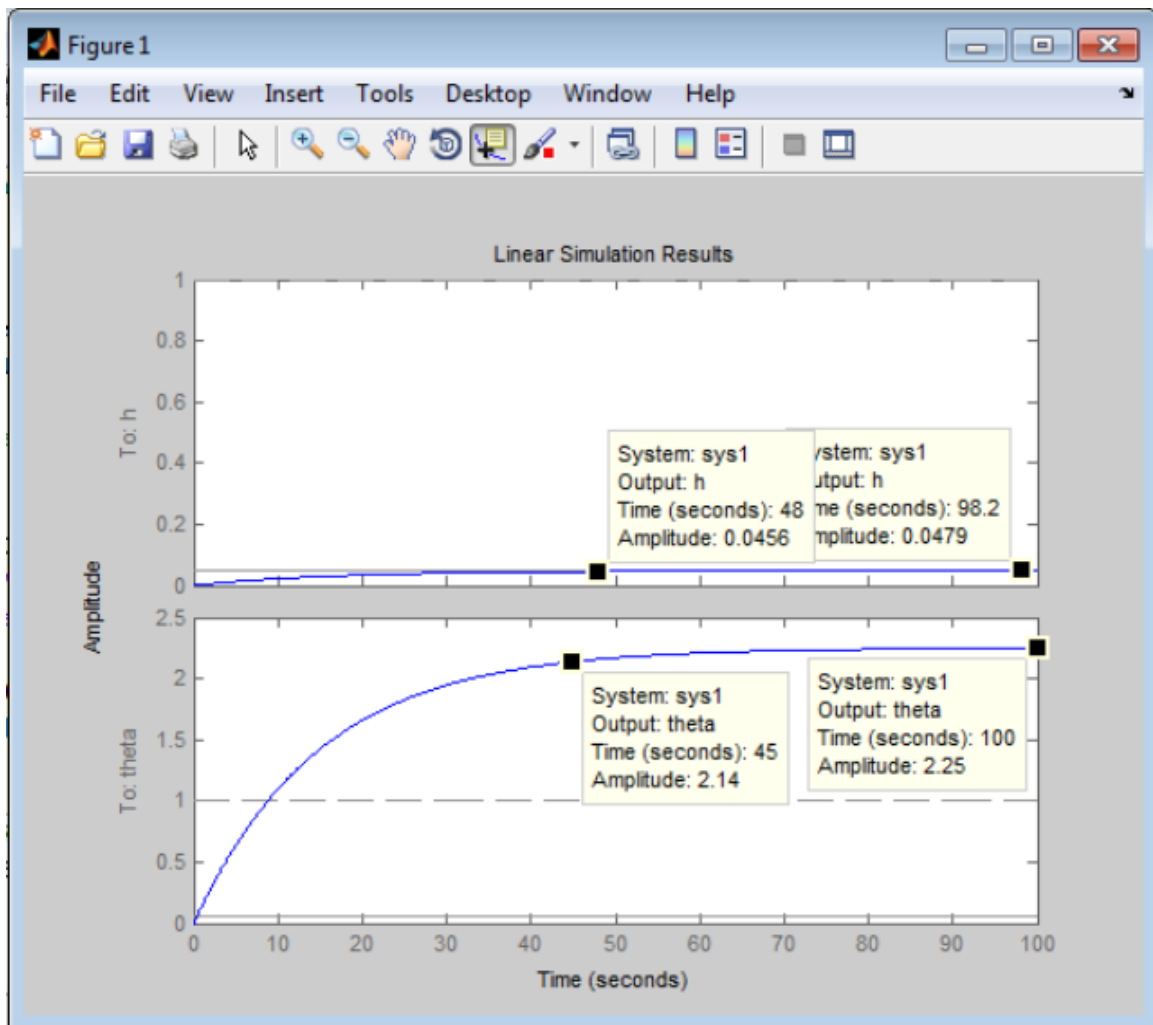


FIGURE 13 – Les réponses à un échelon de consigne à $h_c = 50mm$ et $\theta_c = 1C$

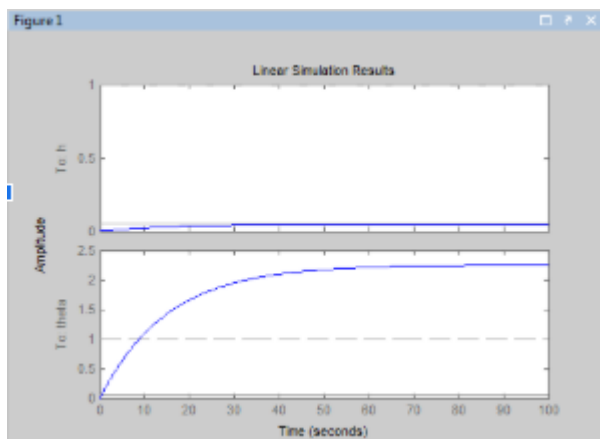


FIGURE 14 – Résultat obtenu sans K

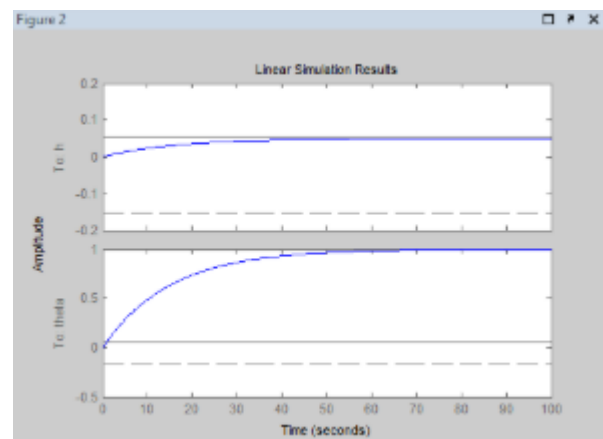


FIGURE 15 – Résultat obtenu avec K

5 Analyse de la robustesse et de la commande d'état :

Afin d'analyser la robustesse de commande d'état, nous choisissons un autre point de fonctionnement $Q_{s0} = 15l/min$ et $\theta_{s0} = 80^\circ C$ et nous comparons les résultats obtenus. Donc, nous définissons les nouveaux paramètres autour de ce point de fonctionnement, puis nous exerçons la même commande d'état.

```

1 %Param tres fixe de syst me
2 Qe_0 = 20 ; %en l/min
3 THETAs_OR = 80 ; %en C
4 H_0=0.6 ; %en m
5 Pu_0=20 ; %en KW
6 Qs_OR=15 ; %en l/min
7 THETAe=20 ; %en C
8 S=1 ; %en m^2
9 %%
10 %%Mod lisation du syst me
11 Amat_R = [ -Qe_0/(2*S*H_0) 0 ;
12            0 -Qe_0/(S*H_0) ];
13
14 Bmat_R = [ 1/S 0 ;
15            (THETAs_OR-THETAe)/(S*H_0) (THETAs_OR-THETAe)/(S*H_0*Pu_0) ];
16
17 Cmat_R=eye(2);
18
19 Dmat_R=[0 0;0 0];
20
21 sys_R=ss (Amat ,Bmat ,Cmat ,Dmat) ;
22 %Analyse du syst me
23
24 T = 6; %en min (P riode d' chantillonnage )
25
26 %Repr sentation d' tat du syst me chantillonn
27 sys_d_R = c2d(sys_R,T);
28 [Ad_R,Bd_R,Cd_R,Dd_R] = ssdata(sys_d_R);
29
30 %matrice de transfert
31 syms s
32
33 H_R=Cmat_R*inv(s*eye(2)-Amat_R)*Bmat_R;
34
35 %%commande par retour d' tat
36
37 tau1 = 16 ; %en min
38 tau2 = 15 ; %en min
39
40 L_R = place (Ad_R,Bd_R,poles);
41
42 Ac_R = (Ad_R - Bd_R*L);
43 sys1_R = ss(Ac_R,Bd_R,Cd_R,Dd_R,'statename',states,...
44             'inputname',inputs,...
45             'outputname',outputs);
46 t = [0:0.1:100];
47 u = [0.05*ones(size(t));...
48      1*ones(size(t))];
49 figure(3);
50 lsim (sys1_R,u,t);
51 %% matrice de gain K
52 K_R = inv(-Cd_R*inv(Ad_R-Bd_R*L_R)*Bd_R);
53 v =K_R*u;

```

```
44 figure(4);
45 lsim(sys1_R,v,t);
```

La figure (16) représente l'évolution de sorties de système. Nous remarquons qu'elles ont les mêmes évolutions que celles trouvées dans la partie 3. Nous en déduisons que la commande par retour d'état est robuste.

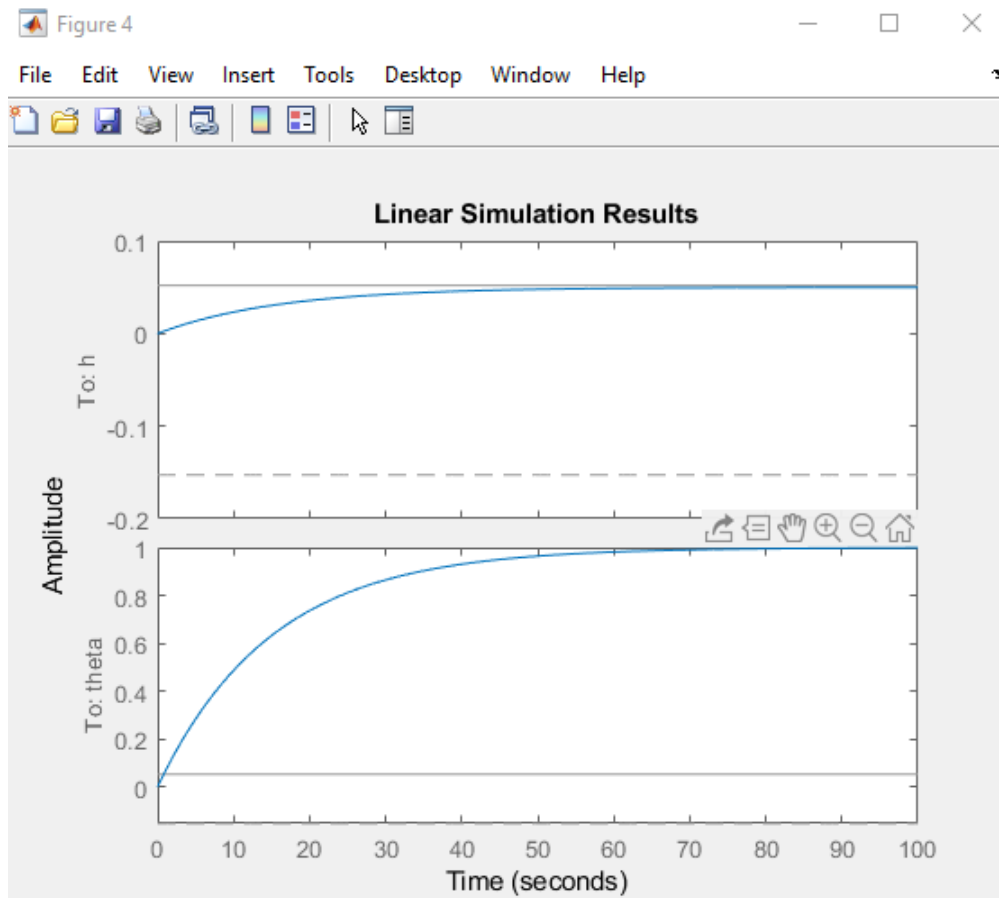


FIGURE 16 – La réponse du système autour le nouveau point de fonctionnement autour $Q_{s0} = 15l/min$ et $\theta_{s0} = 80C$

6 Commande modale (placement de la structure propre) :

Afin de trouver la matrice de contre réaction d'état, nous cherchons le modèle d'état exprimé en code par sysmod. Puis, comme nous avons fait auparavant, nous trouvons Lmod, Acmod et Kmod.

```
1 %% commande modale
2 tfqetoh = TF(1,1);
3 tfqetotheta = TF(2,1);
4 tfputoh = TF(1,2);
```

6 Commande modale (placement de la structure propre) :

```

5 tfputotheta = TF(2,2);
6 %% enlever la fonction de transfert theta = f(qe)
7 TF2 = [tfqetoh tfputoh;0 tfputotheta];
8 %% convertir en modele d'etat
9 sysmod = ss(TF2);
10 %% reprendre les calculs
11 sysmod_d = c2d(sysmod,T);
12 [Amod_d,Bmod_d,Cmod_d,Dmod_d] = ssdata(sysmod_d);
13
14 Lmod = place(Amod_d,Bmod_d,poles);
15 Acmod = (Amod_d - Bmod_d*Lmod);
16 sysmodBF = ss(Acmod,Bmod_d,Cmod_d,Dmod_d,'statename',states,...
17             'inputname',inputs,...
18             'outputname',outputs);
19 figure(5);
20 lsim(sysmodBF,u,t);
21
22 Kmod = inv(-Cmod_d*inv(Amod_d-Bmod_d*Lmod)*Bmod_d);
23
24 v= Kmod*u;
25 figure(6);
26 lsim(sysmodBF,v,t);

```

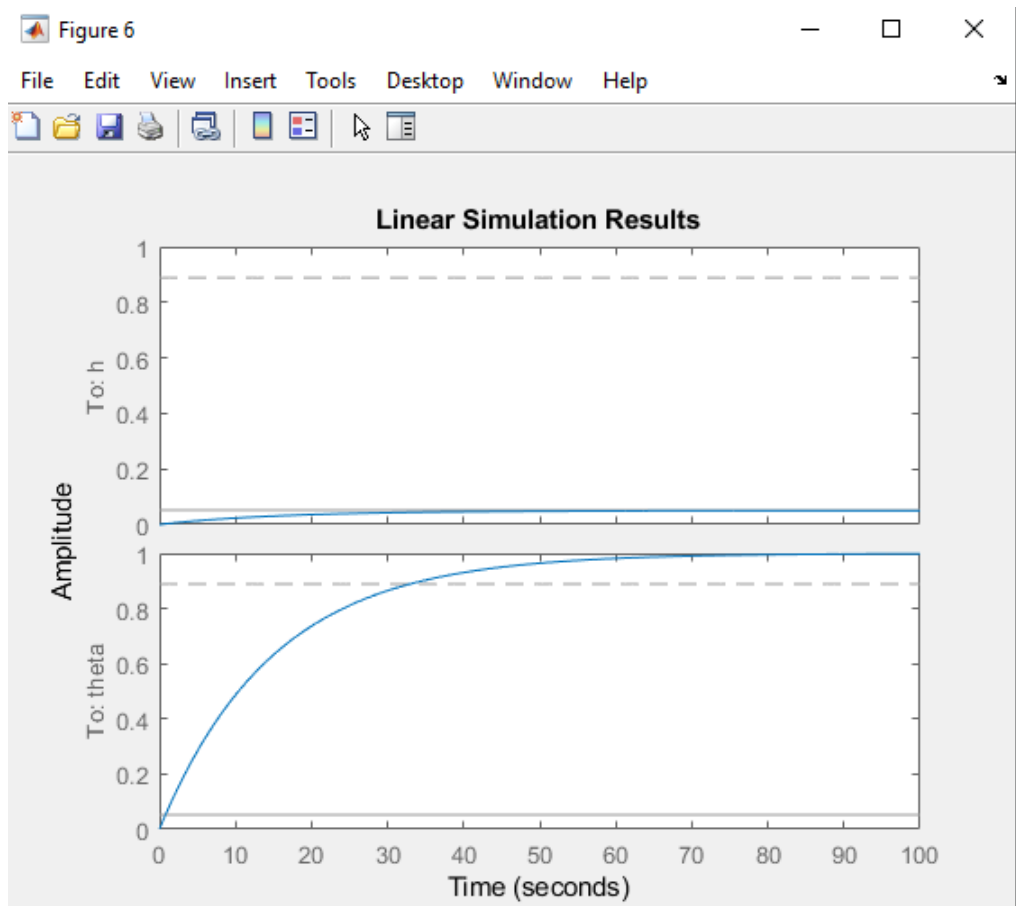


FIGURE 17 – Le comportement du système après une commande modale.

7 Conclusion :

En guise de conclusion, ce TP nous a permis à mettre en pratique les notions théoriques abordées en cours par un système que nous trouvons dans plusieurs domaines. Dans un premier temps, nous avons modéliser et étudier le système de régulation de niveau et de température dans une cuve dans l'espace d'état, en s'appuyant sur les bilans volumique et calorimétrique. Ensuite, nous avons analysé le système en boucle ouvert pour conclure sur sa stabilité, sa commandabilité et son observabilité. Ce qui nous a permet d'optimiser sa commande en utilisant la commande par retour d'état. Ainsi, nous avons conclut sur la robustesse de cette dernière. Enfin, nous avons modéliser la commande modale.