Analyse robuste d'un lanceur en utilisant RoMulOC

Corrigé de l'exercice pour Ecole des JD MACS

D. Peaucelle LAAS-CNRS peaucelle@laas.fr

Juin 2011

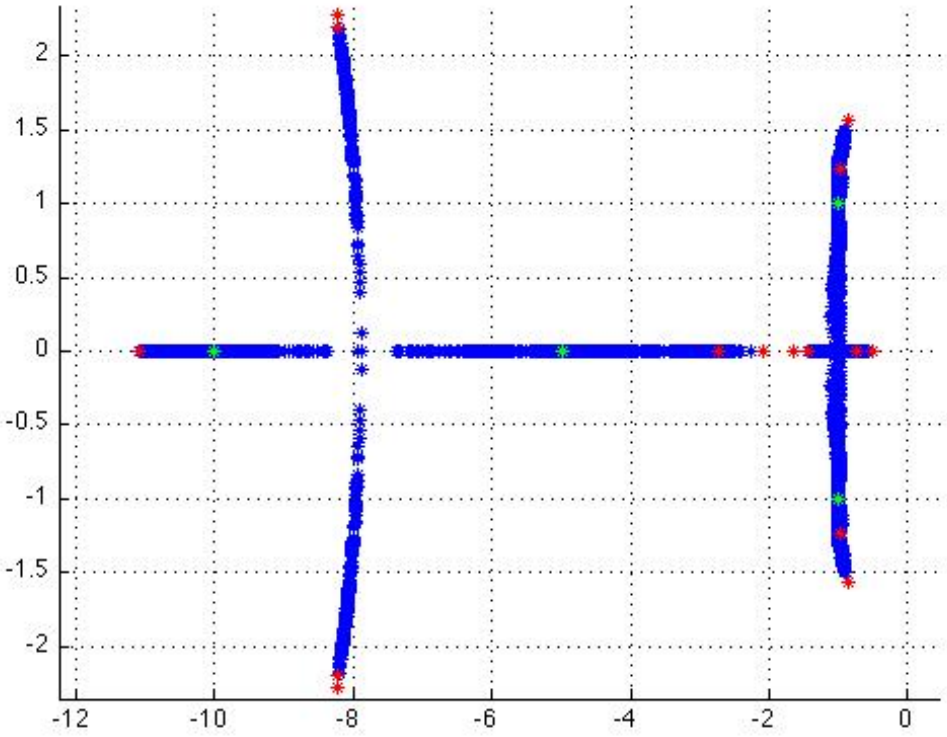
1. Modèle nominal

```
>> omega=[0.6 1.4];
>> omega_nom=mean(omega);
>> lanceur_nom=ssmodel('lanceur nominal');
>> lanceur_nom.A=[0 1;omega_nom^2 0];
>> lanceur_nom.Bu=[0;1];
>> lanceur_nom.Cy=[1 0];
>> tau=[3.5 6.5];
>> tau_nom=mean(tau);
>> moteur_nom=ssmodel('moteur nominal');
>> moteur_nom.A=-tau_nom;
>> moteur_nom.Bu=tau_nom;
>> moteur_nom.Cy=1;
>> la_mot_nom = lanceur_nom*moteur_nom;
>> correcteur=tf([466 2977 3223],[1 112 1223]);
>> bf_nom = feedback(la_mot_nom,correcteur);
>> pole(bf_nom)
ans =
 -99.9998
  -1.0009 + 1.0000i
  -1.0009 - 1.0000i
  -4.9926
 -10.0060
```

2. Modèle incertain

```
>> lanceur=ssmodel('lanceur');
>> lanceur(1).A=[0 1;omega(1)^2 0];
>> lanceur(2).A=[0 1;omega(2)^2 0];
>> lanceur.Bu=[0;1];
>> lanceur.Cy=[1 0];
>> lanceur=uinter(lanceur);
>> moteur=ssmodel('moteur');
>> moteur(1).A=-tau(1);
>> moteur(2).A=-tau(2);
>> moteur(1).Bu=tau(1);
>> moteur(2).Bu=tau(2);
>> moteur.Cy=1;
>> moteur=upoly(moteur);
>> la_mot = lanceur*moteur;
>> bf = feedback(la_mot,correcteur);
>> pole(bf)
All systems in array are stable
ans(:,:,1) =
 -99.9186
  -0.8739 + 1.5634i
  -0.8739 - 1.5634i
  -2.7386
 -11.0949
ans(:,:,2) =
  -99.9187
   -0.7531
   -1.6580
   -2.0915
  -11.0788
ans(:,:,3) =
   1.0e+02 *
  -1.0008
  -0.0099 + 0.0123i
  -0.0099 - 0.0123i
  -0.0822 + 0.0219i
  -0.0822 - 0.0219i
ans(:,:,4) =
   1.0e+02 *
  -1.0008
  -0.0051
  -0.0144
  -0.0823 + 0.0227i
  -0.0823 - 0.0227i
```

```
>> figure,hold on
>> for ii=1:1000
  plot(pole(rand(bf,5),0)+i*1e-6,'*');
end
>> for ii=1:bf.nb
  plot(pole(bf(ii),0)+i*1e-6,'r*');
end
>> plot(pole(bf_nom,0)+i*1e-6,'g*');
>> grid
```



3. Stabilité robuste

```
>> quiz= ctrpb('a','unique')+stability(bf);
>> solvesdp(quiz);
Robustly stable
```

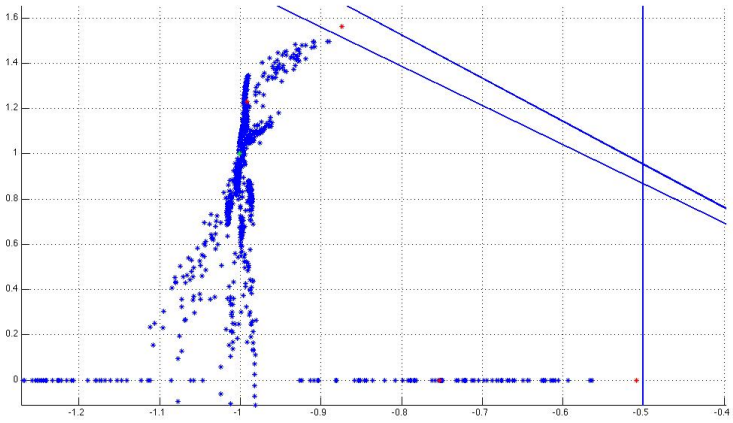
4. Amortissement et constante de temps robustes

```
>> r=region('plane',0,pi/6.5);
>> plot(r)
>> quiz= ctrpb('a','unique')+dstability(bf,r);
>> solvesdp(quiz);
Robustly D-stable for region:
Half-plane below inclinded line that crosses 0
and makes an angle of 0.48332 rad with the imag axis

>> r=region('plane',0,pi/6);
>> plot(r)
>> quiz= ctrpb('a','unique')+dstability(bf,r);
>> solvesdp(quiz);
Infeasible problem

>> r=region('plane',-0.5,0);
>> plot(r);
>> quiz= ctrpb('a','unique')+dstability(bf,r);
>> solvesdp(quiz);
Infeasible problem

>> quiz= ctrpb('a','poly')+dstability(bf,r);
>> solvesdp(quiz);
Robustly D-stable for region:
Half-plane such that: Re(z)<-0.5
```



5. Ecart entre dynamiques nominales et incertaines

```
>> bf_compare=bf+bf_nom;
>> bf_compare.Bw=[eye(bf.n);eye(bf.n)];
>> bf_compare.Cz=[eye(bf.n),-eye(bf.n)];
```

Norme H_∞

```
>> quiz= ctrpb('a','unique')+hinfty(bf_compare);
>> solvesdp(quiz)
20.3282

>> quiz= ctrpb('a','poly')+hinfty(bf_compare);
>> solvesdp(quiz)
13.4024

>> [xi,iswc,bf_compare_wc] = extractwc(quiz,bf_compare);
>> norm(bf_compare_wc,inf)
13.3672

>> quiz= ctrpb('a','unique')+hinfty(bf_compare.');
>> solvesdp(quiz)
20.3279

>> quiz= ctrpb('a','poly')+hinfty(bf_compare.');
>> solvesdp(quiz)
13.4003

>> [xi,iswc,bf_compare_wc] = extractwc(quiz,bf_compare.')
xi =
    0.0000
    1.0000
    0.0000
    0.0000
iswc =
     1
>> norm(bf_compare_wc,inf)
13.4003
```

Norme H_2

```
>> quiz= ctrpb('a','unique')+h2(bf_compare);
>> solvesdp(quiz)
114.6386

>> quiz= ctrpb('a','poly')+h2(bf_compare);
>> solvesdp(quiz)
14.0085

>> [xi,iswc,bf_wc] = extractwc(quiz,bf_compare);
>> norm(bf_wc,2)
3.0857

>> quiz= ctrpb('a','unique')+h2(bf_compare.');
>> solvesdp(quiz)
10.6773

>> quiz= ctrpb('a','poly')+h2(bf_compare.');
>> solvesdp(quiz)
6.9110

>>[xi,iswc,bf_wc] = extractwc(quiz,bf_compare.');
>> norm(bf_wc,2)
6.2307

>>for ii=1:bf.nb, norm(bf_compare(ii),2), end
ans =
    2.8767
ans =
    6.8720
ans =
    2.5109
ans =
    5.9139
```

6. Rejet de perturbations

```
>> lanceur1=lanceur;
>> lanceur1.Bw=[0;1];
>> lanceur1.Cz=[1,0];
>> la_mot1 = lanceur1*moteur;
>> bf1 = feedback(la_mot1,correcteur);
>> filter=zpk([-1 -1],[-5 -10 -20],500)
>> bf1=shape(bf1,filter);
>> figure
>> sigmaplot(bf1,{0.1 10})
>> quiz= ctrpb('a','unique')+hinfty(bf1);
>> solvesdp(quiz)
2.3753

>> quiz= ctrpb('a','poly')+hinfty(bf1);
>> solvesdp(quiz)
0.9225

>> [xi,iswc,bf1_wc] = extractwc(quiz,bf1);
>> norm(bf1_wc,inf)
0.6553

>> quiz= ctrpb('a','unique')+hinfty(bf1.');
>> solvesdp(quiz)
2.3753

>> quiz= ctrpb('a','poly')+hinfty(bf1.');
>> solvesdp(quiz)
0.8285

>> [xi,iswc,bf1_wc] = extractwc(quiz,bf1.')
>> norm(bf1_wc,inf)
0.8233

>> for ii=1:bf.nb, norm(bf1(ii),inf), end
ans =
    0.8283
ans =
    0.7404
ans =
    0.5950
ans =
    0.7404
```

7. Rejet du bruit de mesure

```
>> lanceur2=lanceur;
>> lanceur2.Dyw=0.01;
>> lanceur2.Dzu=1;
>> la_mot2 = lanceur2*moteur;
>> bf2 = feedback(la_mot2,correcteur);
>> quiz= ctrpb('a','unique')+h2(bf2);
>> solvesdp(quiz)
36.1788

>> quiz= ctrpb('a','poly')+h2(bf2);
>> solvesdp(quiz)
4.4819

>> [xi,iswc,bf2_wc] = extractwc(quiz,bf2);
>> norm(bf2_wc,2)
1.8324

>> quiz= ctrpb('a','unique')+h2(bf2.');
>> solvesdp(quiz)
3.9272

>> quiz= ctrpb('a','poly')+h2(bf2.');
>> solvesdp(quiz)
3.3871

>> [xi,iswc,bf2_wc] = extractwc(quiz,bf2.');
>> norm(bf2_wc,2)
1.4744

>> for ii=1:bf.nb, norm(bf2(ii),2), end
ans =
    1.1049
ans =
    1.1057
ans =
    2.0386
ans =
    2.0389
```

8. Pic en réponse à des conditions initiales

```
>> lanceur3=lanceur;
>> lanceur3.Bw=diag([sqrt(0.1) sqrt(10)]);
>> lanceur3.Dzu=1;
>> la_mot3 = lanceur3*moteur;
>> bf3 = feedback(la_mot3,correcteur);
>> quiz= ctrpb('a','unique')+i2p(bf3);
>> solvesdp(quiz)
12.7180

>> quiz= ctrpb('a','poly')+i2p(bf3);
>> solvesdp(quiz)
12.0537

>> [xi,iswc,bf3_wc] = extractwc(quiz,bf3);
xi =
    0.0000
    0.0000
    0.0000
    1.0000
iswc =
     1
>> figure,hold on
>> impulse(bf3,bf3_wc,'r',2)
>> Ymax=impulse(bf3_wc,1)
7.1409

>> dir=[1;0.2];dir=dir/norm(dir);
>> Ymax=impulse(shape(bf3_wc,dir),1)
7.2616
```