4 state aircraft problem with nominal dynamics

Contents

deg(V) = 2 analysis

Form the vector field

```
pvar x1 x2 x3 x4
x = [x1;x2;x3;x4];

f0 = [-0.24366*x2^3+0.082272*x1*x2+0.30492*x2^2+0.015426*x2*x3-0.082272*x2
      -0.054444*x2^2+0.10889*x2*x3-0.054444*x3^2+0.91136*x1-0.64516*x2-0.0
      x1;
      -0.864*x1-0.3211*x3];

fdX = [ 0.30765*x2^3+0.099232*x2^2+0.12404*x1+0.90912*x2+0.023258*x3-0.124
        0.00045754*x2;
        0;
        0];

fdm = [0;
       -0.054444*x2^2+0.10889*x2*x3-0.054444*x3^2-0.6445*x2-0.016621*x3+0.
       0;
       0];

fQ = [-0.0068307*x2^2-0.001428*x2;
      0;
      0;
      0];

pvar dX dm

f = f0+fdX*dX+fdm*dm+fQ*dm*dX;

dXrange = [0.99 2.05];
dmrange = [-0.1 0.1];
```

Form the nominal vector field

```
fnom = subs(f,{'dX','dm'},{sum(dXrange)/2,sum(dmrange)/2});
```

Iteration options and basis vector for V

Quadratic V

```
zV = monomials(x,2:2);
```

Use iterations with no bisection

```
Bis.flag = 0;
Bis.r1deg = 1;
```

Generate all options

```
[roaconstr,opt,sys] = GetRoaOpts(fnom, x, zV, [], Bis);
```

Modify the options

Options for coordinate-wise affine iterations (stopping tolerance max number of iterations)

```
opt.coordoptim.IterStopTol = 1e-2;
opt.coordoptim.MaxIters = 15;
```

Display the results of the simulations and iterations

```
opt.display.roaest = 1;
```

Use pt.sim.NumConvTraj convergent trajectories in forming the LP and display the simulations data after every pt.sim.NumConvTraj convergent trajectories are foud

```
opt.sim.NumConvTraj = 100;
opt.sim.dispEveryNth = 20;
```

opt.sim.flag = 1 for simulations + linearization for initial V

opt.sim.flag = 0 only linearization for initial V

```
opt.sim.flag = 1;
```

Start collecting simulation data from p <= opt.sim.BetaInit

```
opt.sim.BetaMax = 8;
opt.sim.BetaInit = 8;
```

Solve the problem and extract the solution

```
outputs = wrapper(sys,[],roaconstr,opt);
[V,beta,gamma,p,multip,betaUpper] = extractSol(outputs);
```

```
------------------Beginning simulations
System 1: Num Stable = 20  Num Unstable = 0  Beta for Sims = 8.000  Beta U
System 1: Num Stable = 40  Num Unstable = 0  Beta for Sims = 8.000  Beta U
System 1: Num Stable = 60  Num Unstable = 0  Beta for Sims = 8.000  Beta U
System 1: Num Stable = 80  Num Unstable = 0  Beta for Sims = 8.000  Beta U
System 1: Num Stable = 100  Num Unstable = 0  Beta for Sims = 8.000  Beta
------------------End of simulations
------------------Begin search for feasible V
Try = 1   Beta for Vfeas = 8.000
------------------Found feasible V
Initial V (from the cvx outer bnd) gives Beta = 6.295
-------------------Iteration = 1
Beta = 7.929 (Gamma = 0.885)
-------------------Iteration = 2
Beta = 8.553 (Gamma = 0.925)
-------------------Iteration = 3
Beta = 8.855 (Gamma = 0.943)
-------------------Iteration = 4
Beta = 8.966 (Gamma = 0.950)
-------------------Iteration = 5
Beta = 9.018 (Gamma = 0.953)
-------------------Iteration = 6
Beta = 9.047 (Gamma = 0.955)
-------------------Iteration = 7
Beta = 9.067 (Gamma = 0.956)
-------------------Iteration = 8
Beta = 9.081 (Gamma = 0.957)
-------------------Iteration = 9
Beta = 9.093 (Gamma = 0.958)
-------------------Iteration = 10
Beta = 9.103 (Gamma = 0.958)
-------------------Iteration = 11
Beta = 9.112 (Gamma = 0.959)
```

Certified beta

```
beta
```

```
beta =
```

```
    9.1122
```

Upper bound

```
if betaUpper >=  opt.getbeta.maxbeta
    display('No upper bound has been established');
else
    betaUpper
end
```

```
No upper bound has been established
```

deg(V) = 4 analysis

Iteration options and basis vector for V

Change the degree of V to 4

```
zV = monomials(x,2:4);
```

Generate all options

```
[roaconstr,opt,sys] = GetRoaOpts(fnom, x, zV, [], Bis);
```

Modify the options (see comments above)

```
opt.coordoptim.IterStopTol = 1e-2;
opt.coordoptim.MaxIter = 20;
opt.display.roaest = 1;
opt.sim.NumConvTraj = 200;
opt.sim.dispEveryNth = 20;
opt.sim.flag = 1;
opt.sim.BetaMax = 20;
opt.sim.BetaInit = 20;
```

Solve the problem and extract the solution

```
outputs = wrapper(sys,[],roaconstr,opt);
[V,beta,gamma,p,multip,betaUpper] = extractSol(outputs);
```

```
------------------Beginning simulations
```

```
System 1: Num Stable = 20  Num Unstable = 0  Beta for Sims = 20.000  Beta f
System 1: Num Stable = 40  Num Unstable = 0  Beta for Sims = 20.000  Beta f
System 1: Num Stable = 57  Num Unstable = 1  Beta for Sims = 19.000  Beta f
System 1: Num Stable = 60  Num Unstable = 1  Beta for Sims = 19.000  Beta f
System 1: Num Stable = 80  Num Unstable = 1  Beta for Sims = 19.000  Beta f
System 1: Num Stable = 100  Num Unstable = 1  Beta for Sims = 19.000  Beta
System 1: Num Stable = 120  Num Unstable = 1  Beta for Sims = 19.000  Beta
System 1: Num Stable = 130  Num Unstable = 2  Beta for Sims = 18.050  Beta
System 1: Num Stable = 140  Num Unstable = 2  Beta for Sims = 18.050  Beta
System 1: Num Stable = 160  Num Unstable = 2  Beta for Sims = 18.050  Beta
System 1: Num Stable = 180  Num Unstable = 2  Beta for Sims = 18.050  Beta
System 1: Num Stable = 181  Num Unstable = 3  Beta for Sims = 17.148  Beta
System 1: Num Stable = 200  Num Unstable = 3  Beta for Sims = 17.148  Beta
------------------End of simulations
------------------Begin search for feasible V
Try = 1   Beta for Vfeas = 17.148
------------------Found feasible V
Initial V (from the cvx outer bnd) gives Beta = 2.771
-------------------Iteration = 1
Beta = 7.002 (Gamma = 0.730)
-------------------Iteration = 2
Beta = 9.503 (Gamma = 1.043)
-------------------Iteration = 3
Beta = 10.779 (Gamma = 1.235)
-------------------Iteration = 4
Beta = 11.707 (Gamma = 1.357)
-------------------Iteration = 5
Beta = 12.346 (Gamma = 1.434)
-------------------Iteration = 6
Beta = 12.929 (Gamma = 1.519)
-------------------Iteration = 7
Beta = 13.449 (Gamma = 1.598)
-------------------Iteration = 8
Beta = 13.892 (Gamma = 1.665)
-------------------Iteration = 9
Beta = 14.256 (Gamma = 1.719)
-------------------Iteration = 10
Beta = 14.566 (Gamma = 1.764)
-------------------Iteration = 11
Beta = 14.828 (Gamma = 1.802)
-------------------Iteration = 12
Beta = 15.060 (Gamma = 1.836)
-------------------Iteration = 13
Beta = 15.249 (Gamma = 1.862)
-------------------Iteration = 14
Beta = 15.409 (Gamma = 1.884)
-------------------Iteration = 15
```

```
Beta = 15.537 (Gamma = 1.901)
------------------Iteration = 16
Beta = 15.645 (Gamma = 1.917)
------------------Iteration = 17
Beta = 15.725 (Gamma = 1.928)
------------------Iteration = 18
Beta = 15.782 (Gamma = 1.937)
------------------Iteration = 19
Beta = 15.828 (Gamma = 1.943)
------------------Iteration = 20
Beta = 15.862 (Gamma = 1.949)
```

Certified beta

```
beta
```

```
beta =

   15.8618
```

Upper bound

```
if betaUpper >=  opt.getbeta.maxbeta
    display('No upper bound has been established');
else
    betaUpper
end
```

```
betaUpper =

   17.9737
```

Published with MATLAB® 7.6