Dynamical Systems TIF155/FIM770

Konstantinos Zakkas

Problem Set 3

3.2 Stability exponents for a toy model

a)

In[ ]:=
```
rDot[μ_, r_] = μ * r - r^3
thetaDot[ω_, ν_, r_] = ω + ν * r^2
```

Out[ ]= $-r^3 + r\,\mu$

Out[ ]= $r^2\,\nu + \omega$

In[ ]:=
```
sols = Solve[rDot[μ, r] == 0, r]
```

Out[ ]= $\{\{r \to 0\}, \{r \to -\sqrt{\mu}\}, \{r \to \sqrt{\mu}\}\}$

In[ ]:=
```
r0 = r /. sols[[3]]
```

Out[ ]= $\sqrt{\mu}$

In[ ]:= The angle after one period is 2π, so we can find the period by solving $\theta(T) - \theta(0) = 2\pi$

In[ ]:=
```
thetaFunction[t_] = (theta[t] /. DSolve[theta'[t] == thetaDot[ω, ν, r0], theta[t], t][[1]])
```

Out[ ]= $t\,\mu\,\nu + t\,\omega + c_1$

The period time can then be computed by solving $\theta(T) - \theta(0) = 2\pi$:

In[ ]:=
```
period = t /. Solve[thetaFunction[t] - thetaFunction[0] == 2 π, t]
```

Out[ ]= $\left\{\frac{2\pi}{\mu\,\nu + \omega}\right\}$

b)

The dynamical system (1) transformed into Cartesian coordinates will be

In[ ]:=
```
xDot[x_, y_, μ_, ω_, ν_] = μ * x - x^3 + -x * y^2 - yω - yνx^2 + -νy^3
yDot[x_, y_, μ_, ω_, ν_] = μ * y - yx^2 + y^3 + xω + νx^3 + νxy^2
```

Out[ ]= $-x^3 - x\,y^2 - y\nu x^2 - y\omega + x\,\mu - \nu y^3$

Out[ ]= $x\omega + y^3 - yx^2 + y\,\mu + \nu x^3 + \nu x y^2$

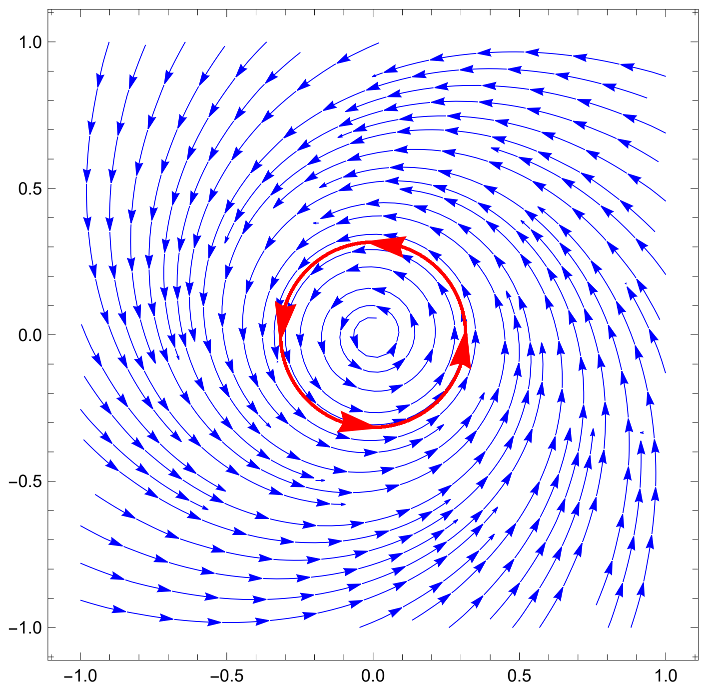
In[•]:=
```
X1Dot = 1 / 10 * X1 - X2^3 - X1 * X2^2 - X1^2 * X2 - X2 - X1^3;
X2Dot = X1 + 1 / 10 * X2 + X1 * X2^2 + X1^3 - X2^3 - X1^2 * X2;

dynSys2 =
  {X1'[t] == 1 / 10 * X1[t] - X2[t]^3 - X1[t] * X2[t]^2 - X1[t]^2 * X2[t] - X2[t] - X1[t]^3,
   X2'[t] == X1[t] + 1 / 10 * X2[t] + X1[t] * X2[t]^2 + X1[t]^3 - X2[t]^3 - X1[t]^2 * X2[t]};
```

In[•]:=
```
Tmax = 100;
sols = NDSolve[Join[dynSys2, Thread[{X1[0], X2[0]} == {1 / Sqrt[10], 0}]],
   {X1[t], X2[t]}, {t, 0, Tmax}];
```

In[•]:=
```
Show[StreamPlot[{X1Dot, X2Dot}, {X1, -1, 1},
   {X2, -1, 1}, StreamStyle → Blue, StreamColorFunction → None],
  ParametricPlot[Evaluate[{X1[t], X2[t]} /. sols], {t, 0, Tmax}, PlotStyle → Red]] /.
 Line[x_] → {Arrowheads[{{0.05, 0.1}, {0.05, 0.4}, {0.05, 0.6}, {0.05, 0.7}}], Arrow[x]}
```

Out[•]=



c) By comparing the coefficients it is clear that the two systems are identical for $\mu = 1/10$, $\omega = 1$, $\nu = 1$

d)

In[•]:=
```
μ = 1 / 10;
ω = 1;
ν = 1;
X1Dot[X1_, X2_] = xDot[X1, X2, μ, ω, ν];
X2Dot[X1_, X2_] = yDot[X1, X2, μ, ω, ν];
Tmax = 2 π / (μ * ν + ω);
```

In[∘]:=
```
mat = D[{dynSys2[[1]][[2]], dynSys2[[2]][[2]]}, {{X1[t], X2[t]}}]
```

Out[∘]=

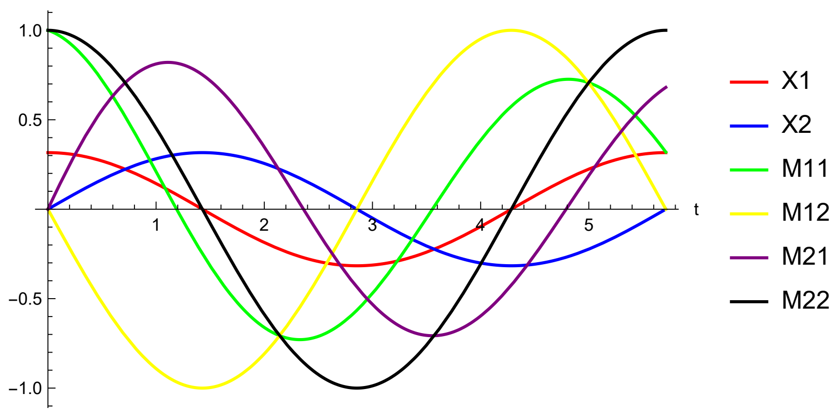
$$\left\{\left\{\frac{1}{10} - 3\,X1[t]^2 - 2\,X1[t] \times X2[t] - X2[t]^2,\ -1 - X1[t]^2 - 2\,X1[t] \times X2[t] - 3\,X2[t]^2\right\},\ \left\{1 + 3\,X1[t]^2 - 2\,X1[t] \times X2[t] + X2[t]^2,\ \frac{1}{10} - X1[t]^2 + 2\,X1[t] \times X2[t] - 3\,X2[t]^2\right\}\right\}$$

In[∘]:=
```
sols2 =
  NDSolve[Join[{dynSys2[[1]], dynSys2[[2]], M11'[t] == mat[[1]][[1]] * M11[t] + mat[[1]][[2]] * M21[t],
     M12'[t] == mat[[1]][[1]] * M12[t] + mat[[1]][[2]] * M22[t],
     M21'[t] == mat[[2]][[1]] * M11[t] + mat[[2]][[2]] * M21[t],
     M22'[t] == mat[[2]][[1]] * M12[t] + mat[[2]][[2]] * M22[t], X1[0] == Sqrt[μ],
     X2[0] == 0, M11[0] == 1, M12[0] == 0, M21[0] == 0, M22[0] == 1}],
   {X1[t], X2[t], M11[t], M12[t], M21[t], M22[t]}, {t, 0, Tmax}]
```

Out[∘]=
```
{{X1[t] → InterpolatingFunction[ Domain: {{0., 5.71}} Output: scalar ][t],
  X2[t] → InterpolatingFunction[ Domain: {{0., 5.71}} Output: scalar ][t],
  M11[t] → InterpolatingFunction[ Domain: {{0., 5.71}} Output: scalar ][t],
  M12[t] → InterpolatingFunction[ Domain: {{0., 5.71}} Output: scalar ][t],
  M21[t] → InterpolatingFunction[ Domain: {{0., 5.71}} Output: scalar ][t],
  M22[t] → InterpolatingFunction[ Domain: {{0., 5.71}} Output: scalar ][t]}}
```

In[∘]:=
```
Show[Plot[{{X1[t]} /. sols2, X2[t] /. sols2,
   M11[t] /. sols2, M12[t] /. sols2, M21[t] /. sols2, M22[t] /. sols2},
  {t, 0, Tmax}, PlotStyle → {Red, Blue, Green, Yellow, Purple, Black},
  PlotLegends → {"X1", "X2", "M11", "M12", "M21", "M22"}, AxesLabel → {"t", ""}]]
```

Out[∘]=



e)

In[∘]:=
```
x1[t_] = X1[t] /. sols2;
x1[Tmax]
x2[t_] = X2[t] /. sols2;
x2[Tmax]
m11[t_] = M11[t] /. sols2;
m11[Tmax]
m12[t_] = M12[t] /. sols2;
m12[Tmax]
m21[t_] = M21[t] /. sols2;
m21[Tmax]
m22[t_] = M22[t] /. sols2;
m22[Tmax]
```

Out[∘]= {0.316228}

Out[∘]= $\{-6.71405\times10^{-9}\}$

Out[∘]= {0.319053}

Out[∘]= $\{2.12317\times10^{-8}\}$

Out[∘]= {0.680947}

Out[∘]= {1.}