Coupled_pendulums

March 31, 2022

1 Coupled pendulums

The continous dynamical system here considered is a two-component Hamiltonian system, which governs the dynamics of two pendulums coupled by a Hooke string, which consists of two bobs of equal mass $m > 0$ and strings of equal length $l > 0$; the bobs are attached to a Hooke's string with coefficient $k > 0$.

Let be θ_j for $j = 1, 2$ the angular displacement of each bob. The kinetic and potencial energy are given by

$$T\left(\varphi_1, \varphi_2\right) = \frac{1}{2} ml^2 \left(\varphi_1^2 + \varphi_2^2\right) \,, \quad U(\theta_1, \theta_2) = mgl\left(2 - \cos\theta_1 - \cos\theta_2\right) + \frac{1}{2} kl^2 \left(\theta_1 - \theta_2\right)^2 \,,$$

where $\varphi_j = \dot{\theta}_j$ for $j = 1, 2$ and $g \approx \pi^2$ is the gravity acceleration constant. In consequence, the Hamiltonian is given by

$$H\left(\theta_1, \theta_2, \varphi_1, \varphi_2\right) = T\left(\varphi_1, \varphi_2\right) + U(\theta_1, \theta_2) \,,$$

and the Hamiltonian equations, also known as equations of motion, are

$$\text{(P1)} \quad \begin{cases} \dot{\theta}_1 = \dfrac{\partial H}{\partial \varphi_1} = ml^2 \varphi_1 \,, \\ \dot{\theta}_2 = \dfrac{\partial H}{\partial \varphi_2} = ml^2 \varphi_2 \,, \\ \dot{\varphi}_1 = -\dfrac{\partial H}{\partial \theta_1} = -mgl\sin\theta_1 - kl^2\left(\theta_1 - \theta_2\right) \,, \\ \dot{\varphi}_2 = -\dfrac{\partial H}{\partial \theta_2} = -mgl\sin\theta_2 + kl^2\left(\theta_1 - \theta_2\right) \,. \end{cases}$$

Upon considering the mapping $t \mapsto ml^2 t$, we get

$$\text{(P2)} \quad \begin{cases} \dot{\theta}_1 = \varphi_1 \,, \\ \dot{\theta}_2 = \varphi_2 \,, \\ \dot{\varphi}_1 = -\dfrac{g}{l}\sin\theta_1 - \dfrac{k}{m}\left(\theta_1 - \theta_2\right) \,, \\ \dot{\varphi}_2 = -\dfrac{g}{l}\sin\theta_2 + \dfrac{k}{m}\left(\theta_1 - \theta_2\right) \,. \end{cases}$$

Notice that, this system gives place to the nonlinear second order nonlinear ODE system

$$
\text{(P3)} \begin{cases} \ddot{\theta}_1 + \dfrac{g}{l}\sin\theta_1 + \dfrac{k}{m}(\theta_1 - \theta_2) = 0\,, \\ \ddot{\theta}_2 + \dfrac{g}{l}\sin\theta_2 - \dfrac{k}{m}(\theta_1 - \theta_2) = 0\,. \end{cases}
$$

We will now numerically solve this system as follows:

[7]:
```
# Call necessary modules for solving and plotting
using DifferentialEquations
using Plots
using LaTeXStrings
```

[8]:
```
# Define ODE system
function CoupPend(du,u,p,t)
    du[1] = u[3]
    du[2] = u[4]
    du[3] = -p[1]/p[2]*sin(u[1]) - p[3]/p[4]*(u[1] - u[2])
    du[4] = -p[1]/p[2]*sin(u[2]) + p[3]/p[4]*(u[1] - u[2])
end
```

[8]:
```
CoupPend (generic function with 1 method)
```

[9]:
```
# Set initial conditions
u0    = [pi,0,0.0,0.0]
# Set time interval
tspan = (0.0,50.0)
# Set parameter vector
p     = [pi^2,2,2,1]
# Set ODE solver
prob  = ODEProblem(CoupPend,u0,tspan,p)
```

[9]:
```
ODEProblem with uType Vector{Float64} and tType
Float64. In-place: true
timespan: (0.0, 50.0)
u0: 4-element Vector{Float64}:
 3.141592653589793
 0.0
 0.0
 0.0
```

[10]:
```
# Run solver
sol    = solve(prob)
```

[10]:
```
retcode: Success
Interpolation: specialized 4th order "free" interpolation, specialized 2nd order
"free" stiffness-aware interpolation
t: 166-element Vector{Float64}:
   0.0
```

```
  0.0001125037284706335
  0.0012375410131769684
  0.012487913860240316
  0.05463563645013597
  0.1315887931967195
  0.23466430561039348
  0.3641652333552804
  0.5209507111699843
  0.7038848940093535
  0.915651382044256
  1.153192730846068
  1.3673984685610565
  ⋮
 46.481248951307634
 46.807155819828395
 47.11710874178678
 47.5053756808112
 47.80253275352481
 48.15206381947762
 48.44422087262935
 48.79596052800267
 49.11934889790947
 49.394047886852704
 49.739385928129096
 50.0
u: 166-element Vector{Vector{Float64}}:
 [3.141592653589793, 0.0, 0.0, 0.0]
 [3.1415926138263757, 3.976341719152545e-8, -0.0007068817751235625,
0.0007068817604062292]
 [3.1415878422156935, 4.8113680390621815e-6, -0.007775701366384346,
0.00777568177762041]
 [3.1411027225694714, 0.0004898681828410099, -0.078465783629147,
0.07844565637670863]
 [3.1322126482736077, 0.009356991051969146, -0.3434460009757443,
0.341761410032254]
 [3.0871197878279193, 0.05370001383110153, -0.8290673205061871,
0.8056037714959876]
 [2.967825658695844, 0.1659967606521177, -1.4877098110609728,
1.3558772613485603]
 [2.720389505816537, 0.37680535435664103, -2.339062038838114,
1.8583448463244974]
 [2.269993514416996, 0.6920671218135565, -3.4126532779211667, 2.088079896009421]
 [1.5322176433254913, 1.0550523973326387, -4.622801642041898,
1.7781134059136103]
 [0.45283466554922636, 1.3311181639335297, -5.376022575220918,
0.7156753000932288]
 [-0.7544791750931188, 1.2906231003284119, -4.4696189596868745,
```

```
-1.1480370301059768]
 [-1.5142000996155298, 0.8390582430312106, -2.548337962797854,
-3.0532828080475154]

 [0.16515366935329825, 2.9469587822241037, 1.956732850107705, -0.60137007016338]
 [0.9567974496553865, 2.413448706317708, 2.575451696718892, -2.656770334721994]
 [1.621598116480003, 1.3057762211309782, 1.4859532346002815,
-4.3920150687683925]
 [1.6770803716922138, -0.46882786143849775, -1.438841162895385,
-4.014535815588663]
 [0.8353489688430895, -1.3064854386521199, -4.171898918844925,
-1.502413644092524]
 [-0.9093570296053286, -1.3297096749880113, -5.134339150255884,
1.1658202197278897]
 [-2.195985820294438, -0.8121943612410163, -3.4989353349740577,
2.1415451427933188]
 [-3.013205272647785, -0.14093235576557883, -1.1756423848981843,
1.323434809283536]
 [-3.0610555175922687, -0.018399190978926658, 0.8855665963078745,
-0.6176542284282038]
 [-2.5670299402565115, -0.38123831191970947, 2.7381831104265313,
-1.8639256455826736]
 [-1.2042394226237798, -1.0466124207850664, 5.048567724094413,
-1.610366101318061]
 [0.19168254785858307, -1.2849559712960033, 5.269087897436718,
-0.05599334462569483]
```

[11]:
```
# Plot all evolution states
plot(sol,vars=(0,1),xlabel=L"t",ylabel=L"\theta",label=L"\theta_1",lw=2,layout=(2,1),size␣
 ↪= (800, 600))
plot!(sol,vars=(0,2),label=L"\theta_2",lw=2)
plot!
 ↪(sol,vars=(0,3),xlabel=L"t",ylabel=L"\varphi",label=L"\varphi_1",lw=2,subplot=2)
plot!(sol,vars=(0,4),label=L"\varphi_2",lw=2,subplot=2)
# Save figure
png("argdis.png")
```

[12]:
```
# Plot orbits on a phase space projection
plot(sol,vars=(1,2),xlabel=L"\theta_1",␣
 ↪ylabel=L"\theta_2",lw=2,label="",layout=(2,2),size = (800, 600))
plot!(sol[1:1,1:1],sol[2:2,1:1],m=:star,ms=7,label="",subplot=1,c=[3])
plot!(sol[1:1,end:end],sol[2:2,end:end],m=:
 ↪circle,ms=10,label="",subplot=1,c=[2])
#
plot!(sol,vars=(3,4),lw=2,xlabel=L"\varphi_1",␣
 ↪ylabel=L"\varphi_2",label="",subplot=2)
plot!(sol[3:3,1:1],sol[4:4,1:1],m=:star,ms=7,label="",subplot=2,c=[3])
```

```
plot!(sol[3:3,end:end],sol[4:4,end:end],m=:
 ↪circle,ms=10,label="",subplot=2,c=[2])
#
plot!(sol,vars=(1,3),lw=2,xlabel=L"\theta_1",␣
 ↪ylabel=L"\varphi_1",label="",subplot=3)
plot!(sol[1:1,1:1],sol[3:3,1:1],m=:star,ms=7,label="",subplot=3,c=[3])
plot!(sol[1:1,end:end],sol[3:3,end:end],m=:
 ↪circle,ms=10,label="",subplot=3,c=[2])
#
plot!(sol,vars=(2,4),lw=2,xlabel=L"\theta_2",␣
 ↪ylabel=L"\varphi_2",label="",subplot=4)
plot!(sol[2:2,1:1],sol[4:4,1:1],m=:star,ms=7,label="",subplot=4,c=[3])
plot!(sol[2:2,end:end],sol[4:4,end:end],m=:
 ↪circle,ms=10,label="",subplot=4,c=[2])
# Save figure
png("phaseproj.png")
```