MODELS

 Sébastien Boisgérault

CONTROL ENGINEERING WITH PYTHON

- 📖 Course Materials
-
- 🏦 ITN, Mines Paris - PSL University

SYMBOLS

🐍	Code	🔍	Worked Example
📈	Graph	🧩	Exercise
🏷️	Definition	💻	Numerical Method
💎	Theorem	🧮	Analytical Method
📝	Remark	🧠	Theory
ℹ	Information	🗝️	Hint
⚠️	Warning	🔓	Solution

🐍 IMPORTS

```
from numpy import *
from numpy.linalg import *
from matplotlib.pyplot import *
```

🏷️ ORDINARY DIFFERENTIAL EQUATION (ODE)

The “simple” version:

$$\dot{x} = f(x)$$

where:

- **State:** $x \in \mathbb{R}^n$
- **State space:** $\mathbb{R}^n$
- **Vector field:** $f : \mathbb{R}^n \to \mathbb{R}^n$.

More general versions:

- Time-dependent vector-field:

$$\dot{x} = f(t, x),\ t \in I \subset \mathbb{R},$$

- $x \in X$, open subset of $\mathbb{R}^n$,
- $x \in X$, n-dimensional manifold.

🏷️ VECTOR FIELD

- Visualize $f(x)$ as an **arrow** with origin the **point** x.
- Visualize f as a field of such arrows.
- In the plane $(n=2)$, use quiver from Matplotlib.

🐍 HELPER

We define a `Q` function helper whose arguments are

- `f`: the vector field (a function)
- `xs`, `ys`: the coordinates (two `1d` arrays)

and which returns:

- the tuple of arguments expected by `quiver`.

```python
def Q(f, xs, ys):
    X, Y = meshgrid(xs, ys)
    fx = vectorize(lambda x, y: f([x, y])[0])
    fy = vectorize(lambda x, y: f([x, y])[1])
    return X, Y, fx(X, Y), fy(X, Y)
```

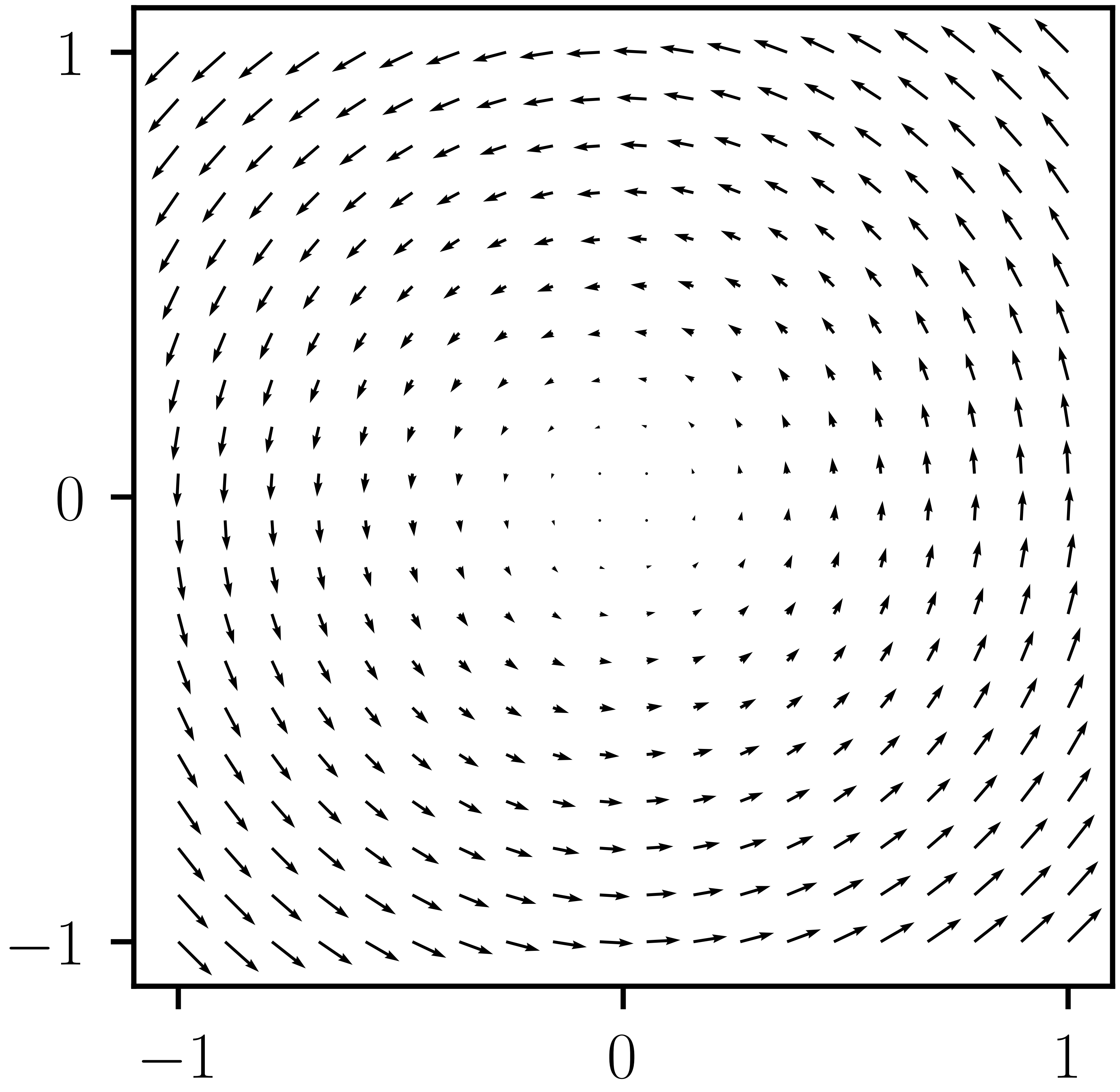
🔍 ROTATION VECTOR FIELD

Consider $f(x, y) = (-y, x)$.

```
def f(xy):
    x, y = xy
    return array([-y, x])
```

📈 VECTOR FIELD

```
figure()
x = y = linspace(-1.0, 1.0, 20)
ticks = [-1.0, 0.0, 1.0]
xticks(ticks); yticks(ticks)
gca().set_aspect(1.0)
quiver(*Q(f, x, y))
```


1
0
−1
−1
0
1

🏷️ ODE SOLUTION

A **solution** of $\dot{x} = f(x)$ is

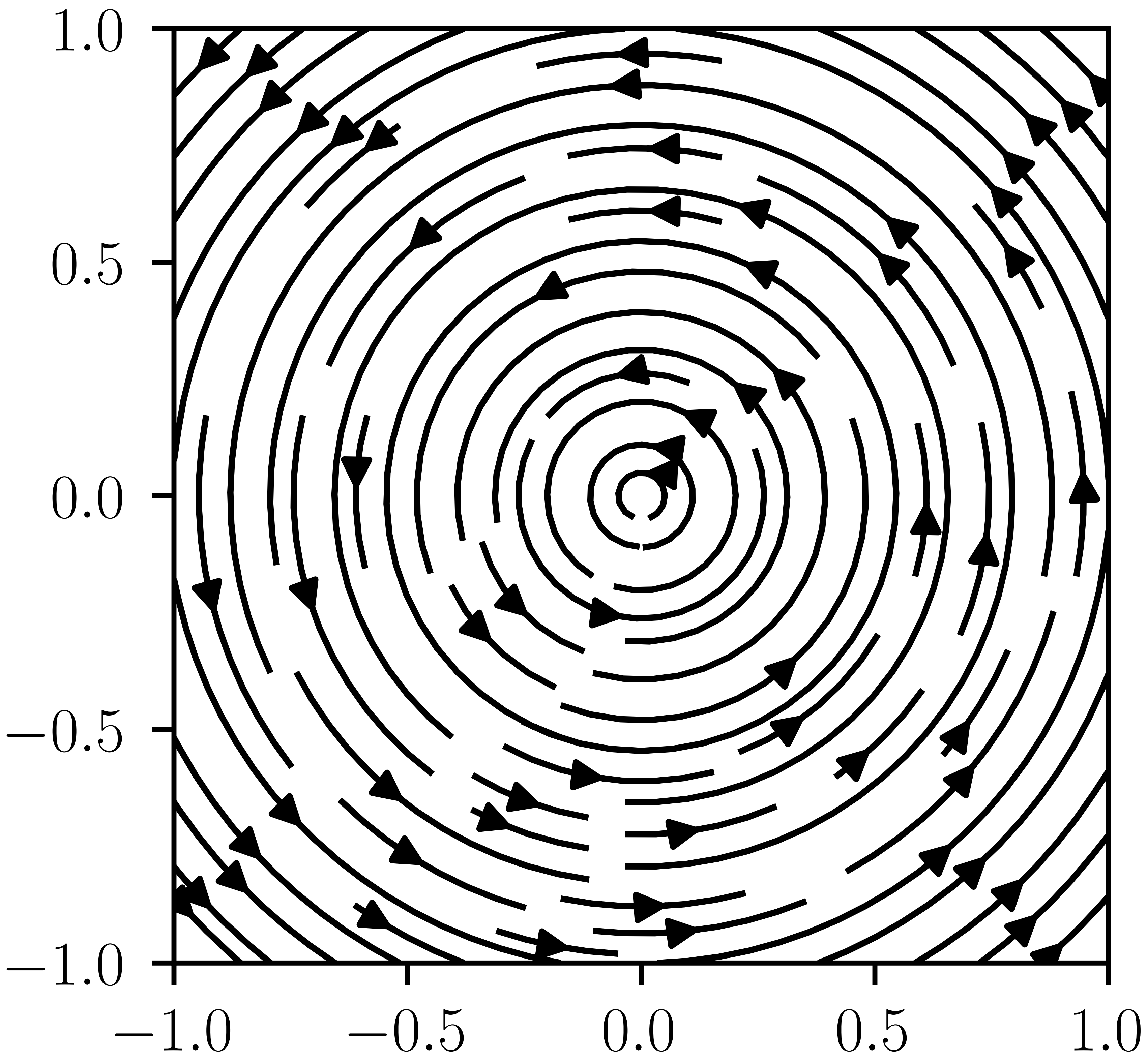
- a (continuously) differentiable function $x : I \to \mathbb{R}^n$,
- defined on a (possibly unbounded) interval I of $\mathbb{R}$,
- such that for every $t \in I$,

$$\dot{x}(t) = dx(t)/dt = f(x(t)).$$

📈 STREAM PLOT

When $n=2$, represent a diverse set of solutions in the state space with streamplot

```
figure()
x = y = linspace(-1.0, 1.0, 20)
gca().set_aspect(1.0)
streamplot(*Q(f, x, y), color="k")
```


1.0
0.5
0.0
−0.5
−1.0
−1.0
−0.5
0.0
0.5
1.0

🏷️ INITIAL VALUE PROBLEM (IVP)

Solutions $x(t)$, for $t \geq t_0$, of

$$\dot{x} = f(x)$$

such that

$$x(t_0) = x_0 \in \mathbb{R}^n.$$

The **initial condition** (t_0, x_0) is made of

- the **initial time** $t_0 \in \mathbb{R}$ and
- the **initial value** or **initial state** $x_0 \in \mathbb{R}^n$.

The point $x(t)$ is the **state at time** t.

🏷️ HIGHER-ORDER ODES

(Scalar) differential equations whose structure is

$$y^{(n)}(t) = g(y, \dot{y}, \ddot{y}, \ldots, y^{(n-1)})$$

where $n > 1$.

💎 HIGHER-ORDER ODES

The previous n-th order ODE is equivalent to the first-order ODE

$$\dot{x} = f(x),\; x \in \mathbb{R}^n$$

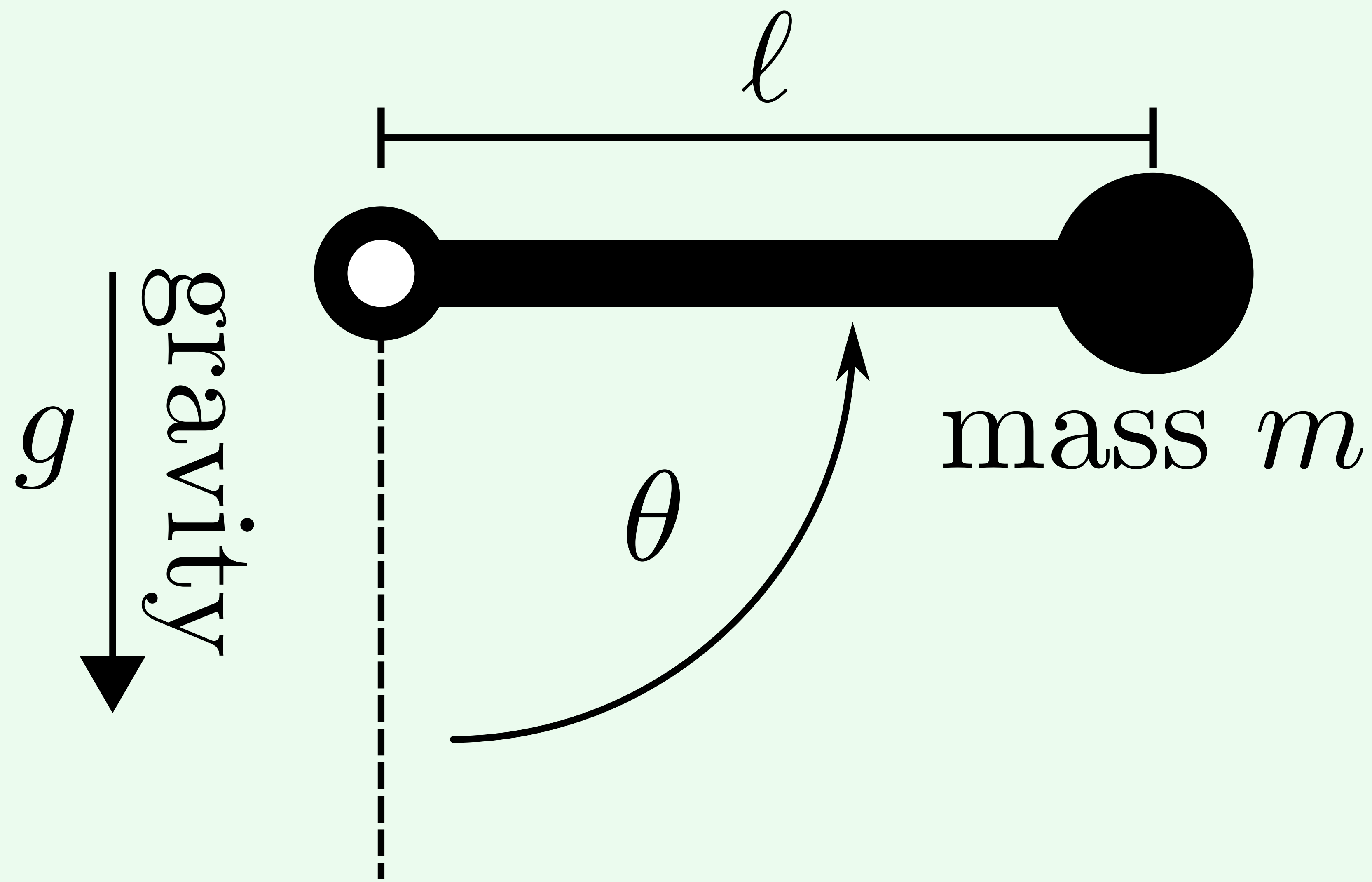
with

$$f(y_0, \dots, y_{n-2}, y_{n-1}) := (y_1, \dots, y_{n-1}, g(y_0, \dots, y_{n-1})).$$

The result is more obvious if we expand the first-order equation:

$$\begin{aligned}
\dot{y}_0 &= y_1 \\
\dot{y}_1 &= y_2 \\
&\vdots \\
\dot{y}_n &= g(y_0, y_1, \ldots, y_{n-1})
\end{aligned}$$

PENDULUM




0:00 / 0:10

1. 🧠 🧮

Establish the equations governing the pendulum dynamics.

2. 🧠 🧮

Generalize the dynamics when there is a friction torque $c = -b\dot{\theta}$ for some $b \geq 0$.

We denote ω the pendulum **angular velocity**:

$$\omega := \dot{\theta}.$$

3. 🧠 🧮

Transform the dynamics into a first-order ODE with state $x = (\theta, \omega)$.

4. 📈

Draw the system stream plot when $m = 1, \ell = 1,$ $g = 9.81$ and $b = 0$.

5. 🧠 🧮

Determine least possible angular velocity $\omega_0 > 0$ such that when $\theta(0) = 0$ and $\dot{\theta}(0) = \omega_0$, the pendulum reaches (or overshoots) $\theta(t) = \pi$ for some $t > 0$.

🔓 PENDULUM

1. 🔓

The pendulum **total mechanical energy** E is the sum of its **kinetic energy** K and its **potential energy** V:

$$E = K + V.$$

The kinetic energy depends on the mass velocity v:

$$K = \frac{1}{2}mv^2 = \frac{1}{2}m\ell^2\dot{\theta}^2$$

The potential energy mass depends on the pendulum elevation y. If we set the reference $y = 0$ when the pendulum is horizontal, we have

$$V = mgy = -mg\ell\cos\theta$$

$$\Rightarrow\ E = K + V = \frac{1}{2}m\ell^2\dot{\theta}^2 - mg\ell\cos\theta.$$

If the system evolves without any energy dissipation,

$$\begin{aligned}\dot{E} &= \frac{d}{dt}\left(\frac{1}{2}m\ell^2\dot{\theta}^2 - mg\ell\cos\theta\right)\\ &= m\ell^2\dot{\theta}\ddot{\theta} + mg\ell(\sin\theta)\dot{\theta}\\ &= 0\end{aligned}$$

$$\Rightarrow\ m\ell^2\ddot{\theta} + mg\ell\sin\theta = 0.$$

2. 🔓

When there is an additional dissipative torque $c = -b\theta$, we have instead

$$\dot{E} = c\dot{\theta} = -b\dot{\theta}^2$$

and thus

$$m\ell^2\ddot{\theta} + b\dot{\theta} + mg\ell\sin\theta = 0.$$

3. 🔓

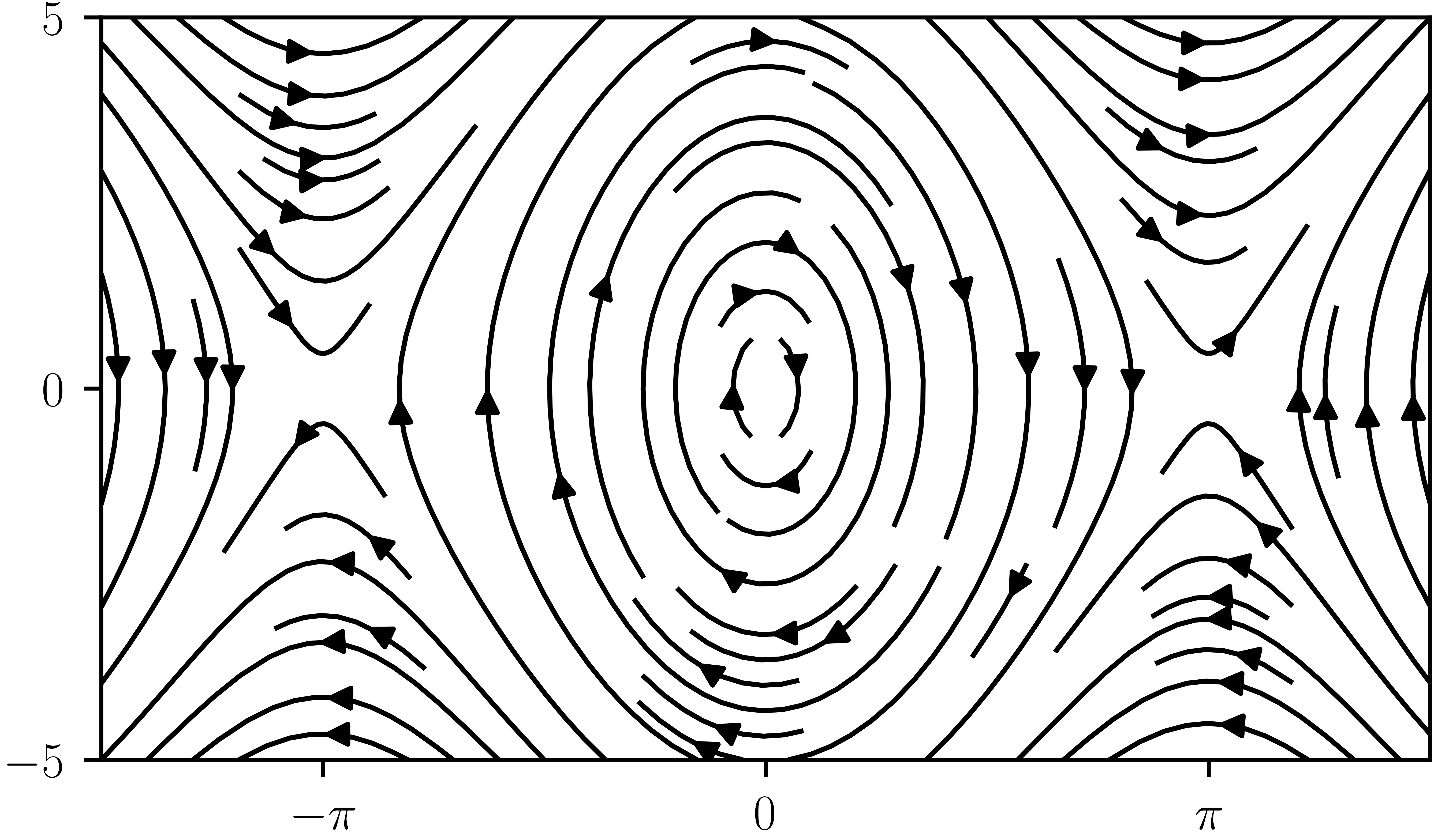
With $\omega := \dot{\theta}$, the dynamics becomes

$$\begin{aligned}\dot{\theta} &= \omega \\ \dot{\omega} &= -(b/m\ell^2)\omega - (g/\ell)\sin\theta\end{aligned}$$

4. 🔓

```python
m=1.0; b=0.0; l=1.0; g=9.81
def f(theta_d_theta):
    theta, d_theta = theta_d_theta
    J = m * l * l
    d2_theta  = - b / J * d_theta
    d2_theta += - g / l * sin(theta)
    return array([d_theta, d2_theta])
```

```
figure()
theta = linspace(-1.5 * pi, 1.5 * pi, 100)
d_theta = linspace(-5.0, 5.0, 100)
labels =  [r"$-\pi$", "$0$", r"$\pi$"]
xticks([-pi, 0, pi], labels)
yticks([-5, 0, 5])
streamplot(*Q(f, theta, d_theta), color="k")
```


5
0
−5
−π
0
π

5. 🔓

In the top vertical configuration, the total mechanical energy of the pendulum is

$$E_{\top} = \frac{1}{2}m\ell^2\dot{\theta}^2 - mg\ell\cos\pi = \frac{1}{2}m\ell^2\dot{\theta}^2 + mg\ell.$$

Hence we have at least $E_{\top} \geq mg\ell$.

On the other hand, in the bottom configuration,

$$E_\perp = \frac{1}{2}m\ell^2\dot{\theta}^2 - mg\ell\cos 0 = \frac{1}{2}m\ell^2\dot{\theta}^2 - mg\ell.$$

Hence, without any loss of energy, the initial velocity must satisfy $E_\perp \geq E_\top$ for the mass to reach the top position.

That is

$$E_{\perp} = \frac{1}{2} m\ell^2\dot{\theta}^2 - mg\ell \geq mg\ell = E_{\top}$$

which leads to:

$$|\dot{\theta}| \geq 2\sqrt{\frac{g}{\ell}}.$$