ME/EECS/BioE 106B Homework 1: Dynamical Systems

Due 1/24/2022, 11:59 P.M.

Foreword

This homework is a mathematically oriented assignment designed to equip you with some of the techniques we'll need going forward to describe robotic systems. In particular, we'll focus on the analysis of *dynamical systems*, systems whose state evolves with the passage of time.
For further reading on the topics we discuss in this assignment, we recommend checking out chapters 3, 5, and 6 of *Feedback Systems: An Introduction for Scientists and Engineers* by Murray and Astrom.
You may use a symbolic toolbox such as MATLAB Symbolic or SymPy to aid in your calculations - please reference in your work where, if at all, you use these tools, and provide a screenshot of your code.

Problem 1: State Space

Ordinary differential equations, or ODEs, vital to the study of control theory. In this section, we'll discuss some important representations of ODEs that we'll refer to across our exploration of controls in this course!
Let's take a moment to develop some conventions for differential equations. In general, we may represent an arbitrary, n^{th} order nonlinear differential equation as a system of n first order differential equations, of the form:

$$\dot{x} = \frac{dx}{dt} = f(x, u) \tag{1}$$

Where $x \in \mathbb{R}^n$ is the *state vector*, $u \in \mathbb{R}^m$ is the *input vector*, t is time, and f is a smooth[1] map. Note that if f does not explicitly depend on time, the system is called *time invariant*. The representation of a nonlinear system as a system of n first order differential equations is known as the *state space* representation of the system. Let's break down the different components of this system description.
In the state space representation, x is known as the state vector. The state vector contains the smallest possible set of variables, known as state variables, that enable us to completely describe the system at any one point in time. The input vector, $u \in \mathbb{R}^n$, contains the set of variables that we have direct control over. We may change the input vector to modify the behavior of the system.
How can we convert high order, nonlinear equations into state space form? Suppose we have the following high order, nonlinear differential equation:

$$\frac{d^n x}{dt^n} = f(x) \tag{2}$$

Where $x \in \mathbb{R}$ and f is a smooth function of x. How can we convert this equation from an n^{th} order ODE to a system of n first order ODEs?

[1] For existence and uniqueness of a solution to be guaranteed, f must be *Lipschitz continuous* with respect to x and piecewise continuous with respect to t.

We may introduce a set of variables known as *phase variables*. This set of of n variables, $\{q_0, q_1, ..., q_{n-1}\}$, is defined as follows:

$$x = q_0 \tag{3}$$
$$\frac{dx}{dt} = q_1 \tag{4}$$
$$\frac{d^2x}{dt^2} = q_2 \tag{5}$$
$$\vdots \tag{6}$$
$$\frac{d^{n-1}x}{dt^{n-1}} = q_{n-1} \tag{7}$$

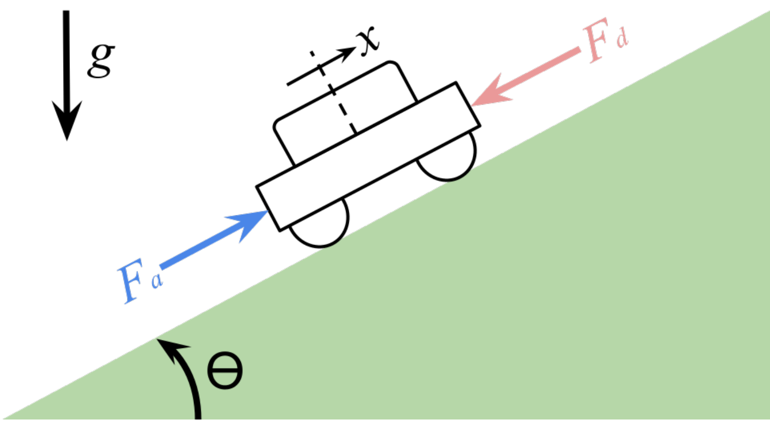
Using these phase variables, we can rewrite our original n^{th} order differential equation as a system of *coupled first order differential equations* as follows:

$$\frac{d}{dt}\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ \vdots \\ q_{n-1} \end{bmatrix} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ \vdots \\ f(q_0) \end{bmatrix} \tag{8}$$

Thus, using phase variables, we can elegantly represent higher order differential equations as systems of first order differential equations! This enables us to write all higher order nonlinear systems in standard state space form.
Let's get some practice working with state space and phase variable representations of systems.

Questions

1. Consider a car which travels uphill in the presence of air resistance.



The position and velocity of the center of mass of the car along the slope are described by $x, \dot{x}$ respectively. The car has a mass m, and is affected by the presence of gravity. F_a is the force applied to the car when the driver presses on the accelerator pedal. F_d, which is calculated according to the formula below, is the force due to drag.

$$F_d = -\frac{1}{2}\rho C_d A|\dot{x}|\dot{x} \tag{9}$$

Where C_d is the drag coefficient of the vehicle, A is the frontal area of the car, and $\dot{x}$ is the velocity of the car.

Using the methods of Newtonian or Lagrangian mechanics, show that the motion of the car is described by the following second order, nonlinear differential equation:

$$m\ddot{x} = F_a - mg\sin\theta - \frac{1}{2}\rho C_d A|\dot{x}|\dot{x} \tag{10}$$

2. Using the method of phase variables, rewrite the system dynamics as a first order system of ODEs in state space:

$$\dot{q} = f(q, u) \tag{11}$$

Where $q \in \mathbb{R}^n$ is the state vector and $u \in \mathbb{R}^m$ is the input vector to the system. Choosing a state vector $q = [x, \dot{x}]$ and an input vector $u = F_a$, rewrite the car dynamics in the form:

$$\frac{d}{dt}\begin{bmatrix} q_0 \\ \vdots \\ q_{n-1} \end{bmatrix} = \begin{bmatrix} q_1 \\ \vdots \\ h(q, u) \end{bmatrix} \tag{12}$$

Where h is a function of the state vector q and an input to the system, u. Note that $h(q, u)$ may have constant terms that don't involve the state vector variables!

3. Are there other types of nonlinear systems other than $\dot{x} = f(x, u)$? In robotics, it's often the case that the system's equations of motion may be placed in *control affine form*:

$$\dot{x} = f(x) + g(x) \cdot u, \; x \in \mathbb{R}^n, \; u \in \mathbb{R}^m \tag{13}$$

Where $f(x)$ is known as the *drift dynamics* and $g(x)$ is a matrix-valued function in $\mathbb{R}^{n \times m}$. In this form, we may explicitly separate the impact of the input, u, from the unforced behavior of the system.
Can the car's equations of motion be placed in control affine form? If so, rewrite the system in control affine form, $\dot{q} = f(q) + g(q)u$ (note that f here is *not* the same as in the previous part). If not, explain what is preventing us from writing the system in control affine form.

Problem 2: Equilibrium Points

So far, we've studied differential equations of the form:

$$\dot{x} = f(x, u) \tag{14}$$
$$\dot{x} = f(x) + g(x)u \tag{15}$$

How may we interpret these differential equations? What properties are important to study and identify in a system? One concept essential to the study of differential equations is that of an *equilibrium point*.

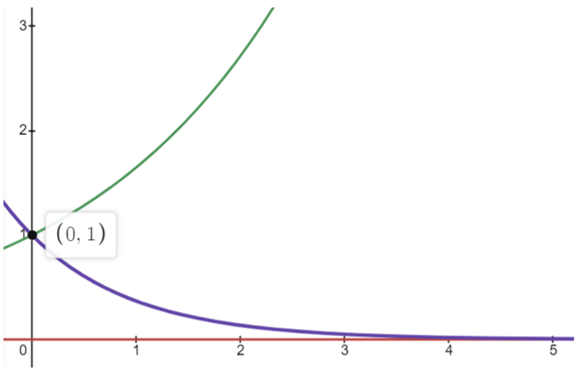
Definition 1 ***Equilibrium Point***
An equilibrium point of a dynamical system:

$$\dot{x} = f(x, u) \tag{16}$$

Is any pair of state and input vectors (x_e, u_e)*, satisfying:*

$$0 = f(x_e, u_e) \tag{17}$$

At an equilibrium point, the time derivative of the state vector is zero! This tells us that the evolution of the system appears to be "frozen" when the system is at an equilibrium point, since the state vector is *not* changing with respect to time.
Suppose we start our system at a point x_o close to the equilibrium point x_e. How can we tell if the system will *stay close* to the equilibrium point or if it will diverge from the equilibrium? This question is essential to the study of *stability*.
Conceptually, an equilibrium point is *stable* if trajectories that "start close" to the equilibrium point "stay close" to the equilibrium point for all time.[2] An equilibrium point is *unstable* if trajectories that start close to the point stray far from the equilibrium point as time passes.



Above: An unstable trajectory diverges from an equilibrium point of $y = 0$*, while a stable trajectory remains close to the equilibrium.*

Although we haven't yet developed a mathematically rigorous definition for stability, there is still a significant amount of interesting analysis we can perform! Let's get some practice with equilibrium points and stability.

[2]This language will be formalized in the definition of Lyapunov stability.

Questions

1. Consider the linear system of first order differential equations:

$$\dot{x} = Ax, \ x \in \mathbb{R}^n, A \in \mathbb{R}^{n \times n} \tag{18}$$

   Prove that $x_e \in \mathbb{R}^n$ is an equilibrium point of the linear system above if and only if it is in the null space of A or is the zero vector.

2. Consider a diagonal $n \times n$ matrix, where $\lambda_i \neq 0, \ 1 \leq i \leq n$:

$$A = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \lambda_n \end{bmatrix} \tag{19}$$

   Prove that for any initial condition $x(0) = x_0$, the solution to the differential equation:

$$\dot{x} = Ax \tag{20}$$

   Will be such that $lim_{t \to \infty} x(t) = 0$ if and only if all of the eigenvalues of A have real components less than zero ($Re(\lambda_i) < 0$). *Hint: What is the solution to the system of differential equations? Does a linear system have a unique solution for any given initial condition?*

3. In the previous problem, we discussed stability in the case where A is diagonal. Let's try to generalize our results somewhat![3]
   Suppose $A \in \mathbb{R}^{n \times n}$ is diagonalizable but not necessarily diagonal. Prove that for all initial conditions $x(0) = x_0 \in \mathbb{R}^n$, the system:

$$\dot{x} = Ax \tag{21}$$

   Has a solution $x(t)$ such that $lim_{t \to \infty} x(t) = 0$ if and only if all of the eigenvalues of A have real parts less than zero. This tells us that the stability of a linear system about its origin is characterized by its eigenvalues![4]

[3]This may be proved for a non diagonalizable matrix using the Jordan Canonical Form.
[4]In particular, this is the asymptotic stability.

Problem 3: The Jacobian Linearization

Oftentimes, we're interested in the behavior of a dynamical system close to certain values of the state vector, x. When we're close to the equilibrium points of a system, we may *approximate* a nonlinear system by a linear system. This approximation is one that often helps us simplify our analysis, as it is typically simpler to deal with linear systems compared to nonlinear ones. We may approximate a time invariant nonlinear system of n first order differential equations:

$$\dot{x} = f(x, u),\ x \in \mathbb{R}^n,\ u \in \mathbb{R}^m \tag{22}$$

About an equilibrium point (x_e, u_e) as a system of n linear differential equations:

$$\dot{z} = Az + Bv \tag{23}$$

Where z and v are defined:

$$z = x - x_e \tag{24}$$

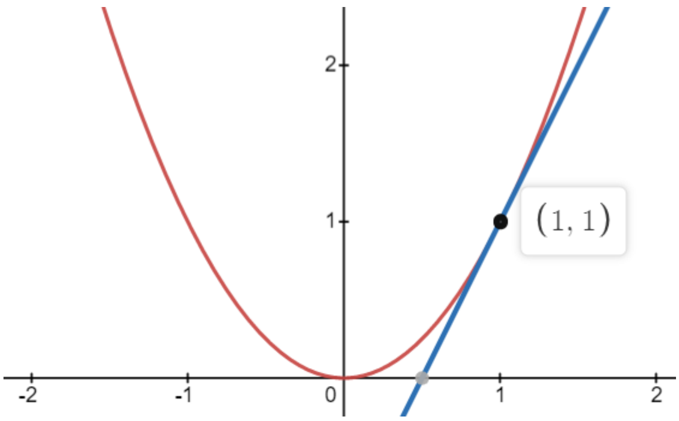
$$v = u - u_e \tag{25}$$

And A and B are matrices that are calculated by taking partial derivatives of f with respect to the state and input vectors:

$$A = \frac{\partial f}{\partial x}\bigg|_{(x,u)=(x_e,u_e)} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}_{(x,u)=(x_e,u_e)} \in \mathbb{R}^{n\times n} \tag{26}$$

$$B = \frac{\partial f}{\partial u}\bigg|_{(x,u)=(x_e,u_e)} = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \cdots & \frac{\partial f_1}{\partial u_p} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial u_1} & \cdots & \frac{\partial f_n}{\partial u_p} \end{bmatrix}_{(x,u)=(x_e,u_e)} \in \mathbb{R}^{n\times p} \tag{27}$$

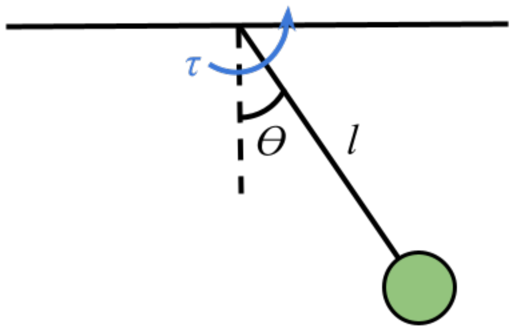
Where $|_{(x,u)=(x_e,u_e)}$ means "evaluate the derivative at $x = x_e, u = u_e$." This type of linear approximation is known as a *Jacobian linearization.*
This approximation may be compared to a Taylor series approximation of a nonlinear function at a point. In the neighborhood of a particular point, we may approximate a function by a tangent line. For example, consider the function $f(x) = x^2$, which has been plotted below.



At the point $x = 1$, we can gain a close approximation of the nonlinear function $f(x) = x^2$ by finding the derivative of f, evaluating it at the point $x = 1$, and finding the equation of the tangent line.
The Jacobian linearization takes this concept from single variable calculus, extends it to multiple variables, and applies it to the study of differential equations to approximate the behavior of a nonlinear system.

Questions

1. Consider the simple pendulum with length l and mass m, that swings under the force of gravity:



If a torque τ is applied to the pendulum about its point of rotation, show using the methods of Newtonian or Lagrangian mechanics that the pendulum dynamics are expressed:

$$ml^2\ddot{\theta} = \tau - mgl\sin\theta \tag{28}$$

Then, show that these dynamics may be written in state space as:

$$\dot{x} = f(x, u) \tag{29}$$

$$\begin{bmatrix}\dot{\theta} \\ \ddot{\theta}\end{bmatrix} = \begin{bmatrix} x_2 \\ \frac{1}{ml^2}(u - mgl\sin x_1)\end{bmatrix} \tag{30}$$

Where the state vector is $x = [x_1, x_2] = [\theta, \dot{\theta}]$ and the input vector is $u = \tau$.

2. Calculate the Jacobian linearization of the pendulum about the point:

$$x_e = \begin{bmatrix} x_{1e} \\ x_{2e}\end{bmatrix},\ u_e \tag{31}$$

Your final answer should be presented in the form:

$$\dot{z} = Az + Bv \tag{32}$$

Where $z = x - x_e$ and $v = u - u_e$. Your A and B matrices *may* depend on the equilibrium point! You may assume that (x_e, u_e) is a valid equilibrium point of the system.

3. Is the Jacobian linearization of the system at $(x, u) = (x_e, u_e)$ a good approximation for all values of x, u? When might it no longer be a good approximation of the system behavior? Provide a brief worded explanation of your reasoning.

Problem 4: Discrete Time Systems

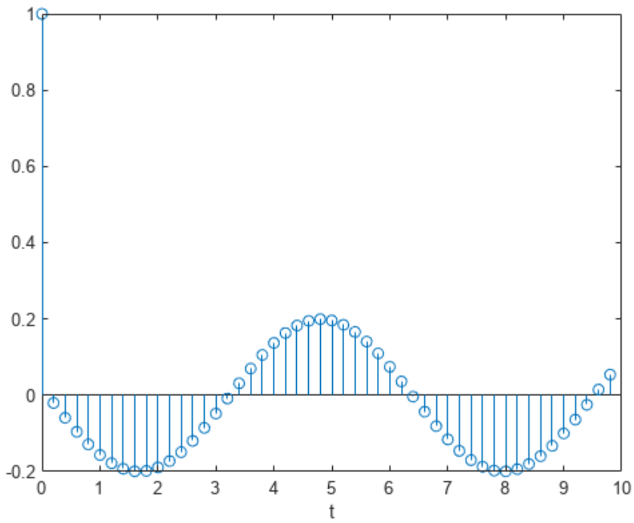
So far, we've discussed of *continuous time dynamical systems* - systems whose states change smoothly with respect to time. These systems are what we encounter when describing the motion of real-world physical objects and are thus the main class of system we'll focus on in this course.
Let's consider what happens when these systems interact with *digital* systems such as computers! Computers cannot collect data or perform computations in continuous time - they sample data from sensors and send control pulses at *discrete* intervals in time!
Because we control robotic systems with digital systems, it's often important for us to develop *discrete time approximations* of the continuous time systems we want to control. How can we describe such a discrete time system?
A nonlinear discrete time system is typically represented in the following form:

$$x(k+1) = f(x(k), u(k)) \tag{33}$$

Where k is an integer that represents the current "time step" of the system. When $k = 0$, we're examining the first time step, which happens at $t = 0$. At $k = 1$, some interval Δt of time has passed. At $k = 2$, $2\Delta t$ has passed, and so on. Assuming a constant sampling interval Δt, this gives us the following relationship between the time t at each time step and the time step k:

$$t = k\Delta t, \ k \in \mathbb{Z}_0^+ \tag{34}$$

Where $\mathbb{Z}_0^+$ is the set of positive integers including 0.



Above: A sinusoidal function sampled at discrete intervals.

How can we approximate a continuous time system of the form $\dot{x} = f(x, u)$ as a discrete time system of the form $x(k+1) = f(x(k), u(k))$? In this problem, we'll provide an answer to this question and more!

Questions

1. The process of approximating a continuous time system by a discrete time system is called *discretization*. In this problem, we'll discuss a simple type of discretization known as *Euler Discretization*, which is commonly used in optimization-based controllers.
   Euler Discretization approximates $\dot{x} = f(x, u)$ as the following discrete time system:

$$\hat{x}(k+1) = \hat{x}(k) + f(x, u)\Delta t \tag{35}$$

Where $\hat{x}(k)$ is the discrete time estimate of $x(t)$. For a constant sampling time of Δt, prove that for smooth f,[5] in the limit $\lim_{\Delta t \to 0}$, this approximation converges to the continuous time system. *Hint: For a sampling time Δt, the relation between discrete and continuous time is $t = k\Delta t$.*

2. Now that we have a discrete time system, let's investigate some of its properties! In this question, we'll begin a rigorous analysis of the stability of discrete time systems using the *contraction mapping theorem.*[6]
Note that this is by no means the only way of analyzing discrete time stability, but is rather an interesting and fun one!

Theorem 1 ***Contraction Mapping Theorem***
Let $f : \mathbb{R}^n \to \mathbb{R}^n$ be a function defined on all of $\mathbb{R}^n$ and assume that there is a constant c such that $0 \leq c < 1$ and:

$$||f(x) - f(y)|| \leq c||x - y|| \tag{36}$$

For all $x, y \in \mathbb{R}^n$. Functions that satisfy this condition are called contraction mappings. If f is a contraction mapping:

(a) f is continuous on $\mathbb{R}^n$.

(b) If y is a fixed point of f $(y = f(y))$, it is the only fixed point.

(c) If x is any arbitrary point in $\mathbb{R}^n$, the sequence $\{x, f(x), f(f(x)), f(f(f(x))), ...\}$ converges to the fixed point y.

Suppose we have a linear discrete time system of the form:

$$x(k+1) = Ax(k),\ x \in \mathbb{R}^n,\ A \in \mathbb{R}^{n\times n} \tag{37}$$

Show that if all of the eigenvalues of A have a magnitude $0 \leq |\lambda_i| < 1$, the only fixed point of the mapping $f(x) = Ax$ is the zero vector, $x = 0$. *Hint: Proceed by contradiction.*

3. Consider a diagonal matrix $A \in \mathbb{R}^{n\times n}$, defined:

$$A = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \lambda_n \end{bmatrix} \tag{38}$$

Where $0 \leq |\lambda_i| < 1$ for all i. Show that $f(x) : x \mapsto Ax$ is a contraction mapping.
Hint: remember that λ_i can be complex! Make sure to deal with this in your solution.

4. Suppose $A \in \mathbb{R}^{n\times n}$ is a diagonalizable matrix. Using the contraction mapping theorem, conclude that for any initial condition $x(0) = x_0$, the solution to the linear system:

$$x(k+1) = Ax(k),\ x(0) = x_0 \tag{39}$$

Converges to 0 as $k \to \infty$ if $0 \leq |\lambda_i| < 1$ for all i, where λ_i is an eigenvalue of A. This proves that if all of the eigenvalues of A are within the open unit disk in the complex plane (excluding $r = 1$), the origin will be a stable equilibrium point. *Note: This is also true when A is not diagonalizable, but the proof requires further mathematics.*

[5] f must be Lipschitz continuous for this to hold.
[6] The proof of the contraction mapping theorem is lots of fun, but requires some knowledge of Cauchy sequences. You're encouraged to try it! This theorem turns out to be highly important to the study of nonlinear systems.

Bonus: Phase Portraits

Thus far, we've primarily discussed analytical methods for interpreting dynamical systems. Can we take a more graphical approach to the analysis of dynamical systems to gain a visual intuition for these mathematical problems?

Phase portraits are diagrams that provide us with an interpretable picture of a dynamical system. Through analyzing these pictures, we can gain some visual perspective for properties such as stability, convergence, and oscillation!

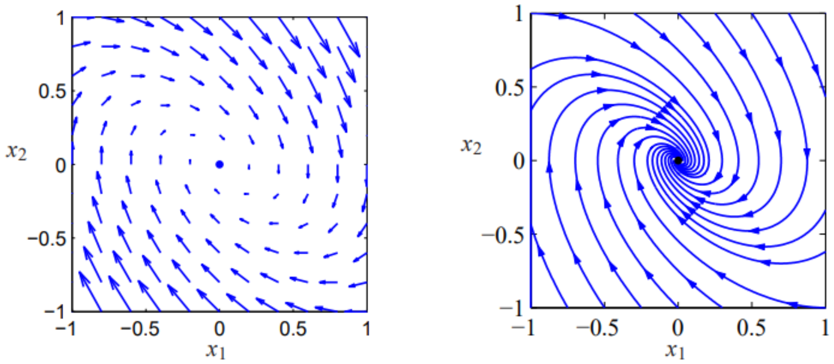
How can we draw a phase portrait? Although the concept of a phase portrait *does* generalize to n dimensions, they're only generally useful to us for two or three dimensional systems, as visualizing four and higher dimensions proves to be quite tricky in our three dimensional world. Let's consider the following two-dimensional system:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} f_1(x_1, x_2) \\ f_2(x_1, x_2) \end{bmatrix} \tag{40}$$

Where f_1 and f_2 are arbitrary smooth functions. To draw a phase portrait, our first step is to draw the *vector field* $[f_1(x_1, x_2), f_2(x_1, x_2)]$. Let's review how we can do this!

As this is a two dimensional system, our vector field will require two coordinate axes. On the first axis, we can place x_1, and on the second axis x_2. Then, at a selection of points in the (x_1, x_2) coordinate plane, we draw the vector given by $[f_1(x_1, x_2), f_2(x_1, x_2)]$.

Once we've sketched out enough of these vectors, we find that by following the direction of the vector arrows, we actually trace out *trajectories* of the system in the (x_1, x_2) plane! These trajectories, which are the solutions to the differential equation for different initial conditions, form the phase portrait of the system.



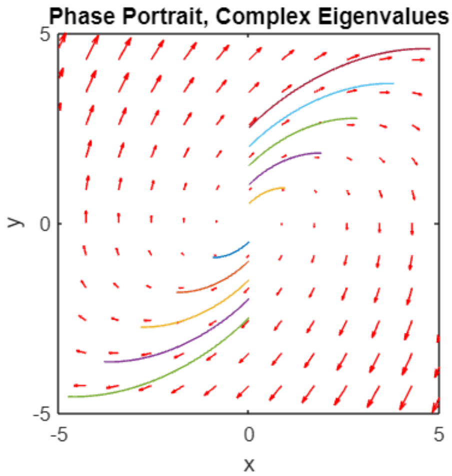
(a) Vector field (b) Phase portrait

Above: An example of a 2D phase portrait for a dynamical systems. (From Murray & Astrom, Feedback Control Systems)

Phase portraits provide us with an elegant way of visualizing the dynamical systems that arise in robotics and control. In this question, we'll see what patterns we can find in the phase portraits of various linear systems of differential equations, and see what interesting connections we can make to other fields of mathematics.

Questions

1. Let's begin our brief study of phase portraits with an observation! When a linear system of differential equations, $\dot{x} = Ax$ has complex eigenvalues, the phase portrait of the system appears to "swirl" around the origin of the system.



In vector calculus, we can measure the "swirl" of a vector field using a vector called *curl*. To calculate the curl of a vector field $f = [f_1, f_2, f_3]$, we use the formula:

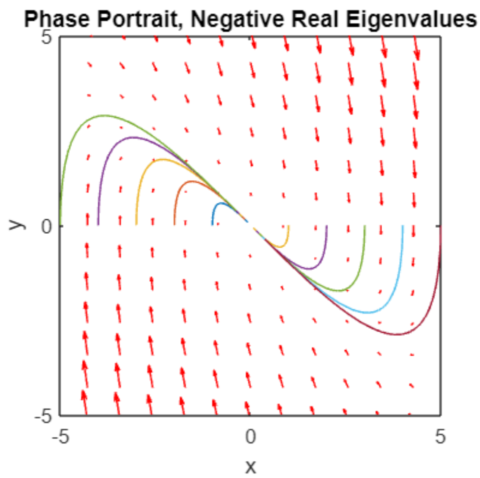
$$\text{curl } f = \nabla \times f = \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} \end{bmatrix} \times \begin{bmatrix} f_1(x,y,z) \\ f_2(x,y,z) \\ f_3(x,y,z) \end{bmatrix} \tag{41}$$

Which takes the cross product of the gradient operator with the vector field. The direction of the curl vector represents the axis of swirl, while the magnitude represents the strength. Prove that for the two-dimensional dynamical system $\dot{q} = Aq$, defined:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{42}$$

The curl of the system is nonzero if A has complex eigenvalues. *Note: Since this is a two-dimensional vector field, to compute the curl, set $f_3(x, y, z) = 0$ and use the formula provided above.*

2. Can we use a phase portrait to visually recognize if an equilibrium point is stable? If an equilibrium point is stable, all of the phase portrait trajectories passing through the equilibrium point will point *inwards* towards the point!



Let's see if we can quantify the inward and outward flow of trajectories using the language of vector calculus.

The *divergence* of a vector field is a quantity that expressed the flow coming from or moving into a particular point in a vector field. The divergence of a vector field $f(x, y, z)$ is computed as follows:

$$\text{div } f = \nabla \cdot f = \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} \end{bmatrix} \cdot \begin{bmatrix} f_1(x, y, z) \\ f_2(x, y, z) \\ f_3(x, y, z) \end{bmatrix} \tag{43}$$

Where we take the dot product of the gradient operator with the vector field.
We recall that for a linear system, $\dot{q} = Aq$, the origin is a stable equilibrium point if all of the eigenvalues of A have negative real components.
Prove *without explicitly computing the eigenvalues of* A that if the origin is a stable[7] equilibrium point, the divergence of:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{44}$$

Is less than zero. *Note: You should just use the eigenvalue conditions for stability of linear systems in this question.*

[7] Using the eigenvalue conditions we have discussed, you may assume this system is asymptotically stable.