Controlling an Advection-Diffusion System

Conor Rowan

Fall 2024

Introduction

The advection-diffusion equation describes the dynamics of the concentration of a substance in a fluid medium. The substance is carried by the velocity of the fluid through advection, and also spreads out through diffusion. The underlying fluid flow is taken as a given. Representing the concentration of the substance as $c(x,t)$, the governing equation is

$$\frac{\partial c}{\partial t} + v_k(x,t)\frac{\partial c}{\partial x_k} - D\frac{\partial^2 c}{\partial x_k \partial x_k} = s(x,t)$$

where D is a material parameter called diffusivity, $v_k(x,t)$ is a prescribed velocity field satisfying the incompressibility condition $\frac{\partial v_k}{\partial x_k} = 0$ and $s(x,t)$ is a source term describing the in- or out-flow of the substance. The incompressibility assumption on the velocity is not necessary, but gives rise to a simpler form of the governing equations. The partial differential equation has a unique solution when an initial concentration field $c(x,0) = c_0(x)$ and the concentration on the boundaries are specified. For simplicity, we will take $c = 0$ along the boundaries of the domain for all time. This equation does not lead to interesting dynamics in one spatial dimension, so we will take the spatial domain Ω to be the unit square. The goal of this problem is to control the advection-diffusion system by manipulating source terms so that the concentration takes on prescribed values at specific points.

Setting up the problem

Dynamics in the advection-diffusion system will be driven by five source terms, an initial concentration field, and a prescribed fluid velocity. The sources will be point sources on a grid in the domain. This can be written as

$$s(x,t) = \sum_{k=1}^{5} u_k(t)\delta(x - x_k)$$

Each source is a scalar function of time $u_k(t)$. See Figure 1. We will investigate a range of initial concentration fields, which are specified later. The fluid velocity must be chosen such that the incompressibility condition is respected, otherwise the equation written down above is not correct. We can choose the fluid velocity as

$$\underline{v}(x,t) = v_0(t)\begin{bmatrix} -nx_2^{n-1} \\ nx_1^{n-1} \end{bmatrix} := v_0(t)\bar{\underline{v}}(x)$$

This velocity field automatically respects the incompressibility condition, and if the exponent n is taken to be large, it approximates flow confined to a box. This vector field is shown in Figure 2, with an exponent of $n = 15$. The velocity can be modulated by a function of time $v_0(t)$ to make things more interesting. This means that the flow can speed up and reverse direction, but its spatial character remains the same over time. Note that this velocity field does not satisfy Navier-Stokes, and it does not obey the no-slip boundary condition at the walls. There is some fluid flow out of the box. This is not the most physical–we must think of the domain as a leaky box!

Deriving the governing equations

First things first, we discretize the concentration as a sum of unknown time varying coefficients multiplied by known spatial shape functions. This reads

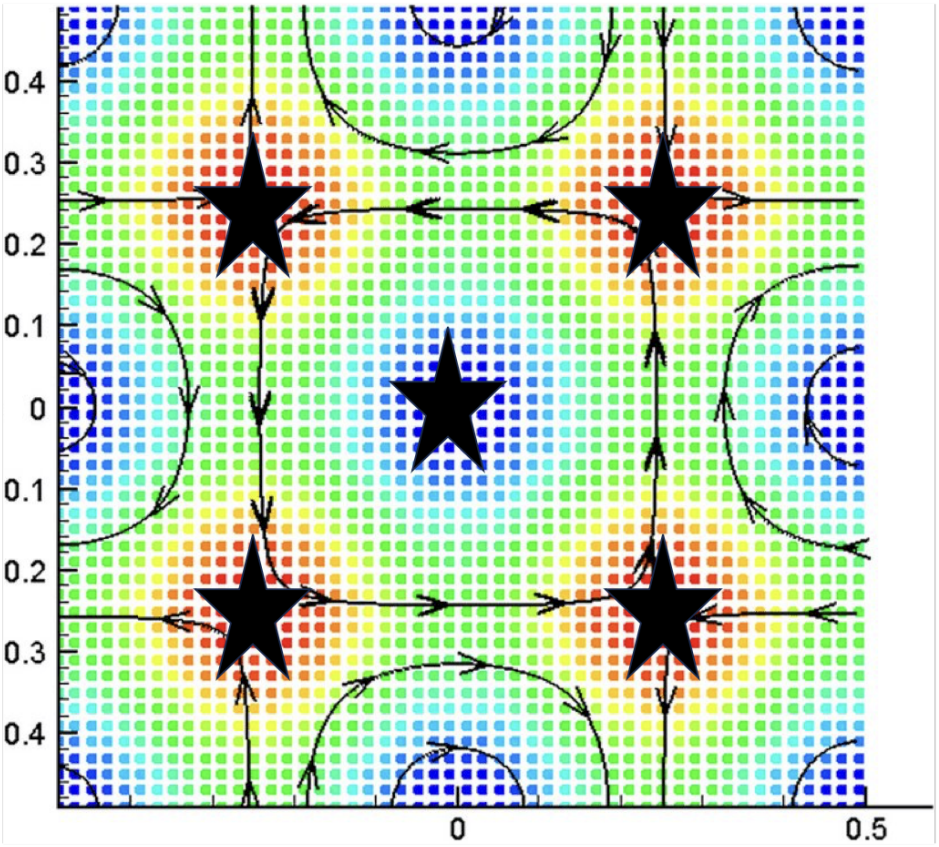

Figure 1: A schematic of an advection-diffusion system in a box. Fluid streamlines are shown in black, and the five point sources are indicated by stars. The substance spreads out through diffusion, and is pushed around with advection.

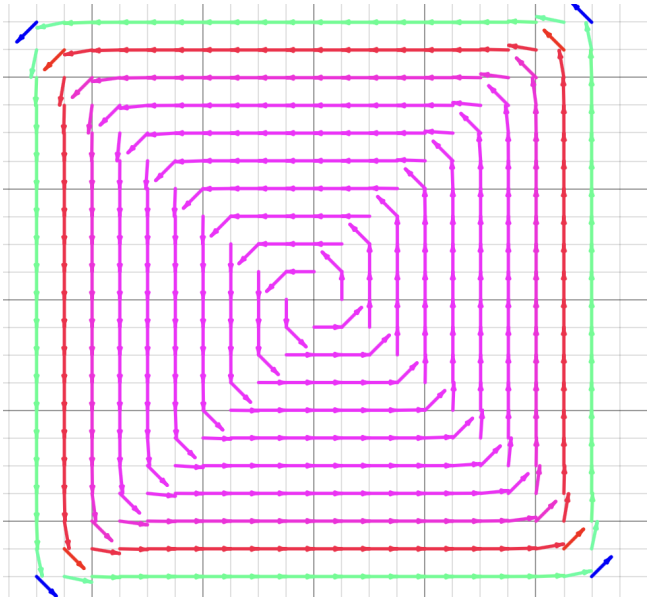

Figure 2: Vortex-like incompressible velocity field, which approximates flow in the unit square. The fluid velocity transports the chemical substance but is not affected by it.

$$c(x,t) = \sum_{i=1}^{\tilde{N}} a_i(t) f_i(x)$$

where $\tilde{N}$ controls the size of the discretization. The shape functions $f_i(x)$ will satisfy the zero concentration boundary condition by construction. To derive the governing system of ordinary differential equations, we

can weaken the governing equation by integrating against the set of "test functions" $f_j(x)$. Because the test functions are the same as the shape function used in the discretization of the solution, this is a Galerkin form. Substituting the given form of the source, this reads

$$\int_\Omega \dot{c} f_j + v_k \frac{\partial c}{\partial x_k} f_j - D \frac{\partial^2 c}{\partial x_k \partial x_k} f_j d\Omega = \int_\Omega \sum_k u_k(t) \delta(x - x_k) f_j d\Omega$$

Note that we can use the definition of delta functions to simplify the right-hand side:

$$\int_\Omega \sum_k u_k(t) \delta(x - x_k) f_j d\Omega = \sum_k u_k(t) f_j(x_k)$$

Plugging in the discretization for the concentration, the left-hand side becomes

$$\sum_i \dot{a}_i \int_\Omega f_i f_j d\Omega + \sum_i a_i \int_\Omega v_k \frac{\partial f_i}{\partial x_k} f_j d\Omega - \sum_i a_i \int_\Omega D \frac{\partial^2 f_i}{\partial x_k \partial x_k} f_j d\Omega$$

Using that the shape functions are zero along the boundary, we can integrate the third term by parts and observe that the boundary term vanishes. The governing equation is then

$$\sum_i \dot{a}_i \int_\Omega f_i f_j d\Omega + \sum_i a_i \int_\Omega v_k \frac{\partial f_i}{\partial x_k} f_j d\Omega + \sum_i a_i \int_\Omega D \frac{\partial f_i}{\partial x_k} \frac{\partial f_j}{\partial x_k} d\Omega = \sum_k u_k(t) f_j(x_k)$$

We can now define the following matrices:

$$M_{ij} := \int_\Omega f_i f_j d\Omega$$

$$K_{ij} := \int_\Omega D \frac{\partial f_i}{\partial x_k} \frac{\partial f_j}{\partial x_k} d\Omega$$

$$W_{ij} := \int_\Omega \bar{v}_k \frac{\partial f_i}{\partial x_k} f_j d\Omega$$

$$B_{jk} := f_j(x_k)$$

The system then becomes

$$\underline{\underline{M}} \dot{\underline{a}} + v_0(t) \underline{\underline{W}}^T \underline{a} + \underline{\underline{K}} \underline{a} = \underline{\underline{B}} \underline{u}(t) \implies \dot{\underline{a}} = \underline{\underline{M}}^{-1} \Big(\underline{\underline{B}} \underline{u}(t) - \underline{\underline{K}} \underline{a} - v_0(t) \underline{\underline{W}}^T \underline{a} \Big)$$

We can use backward Euler to approximate the system as

$$\frac{\underline{a}(t+1) - \underline{a}(t)}{\Delta t} = \underline{\underline{M}}^{-1} \Big(\underline{\underline{B}} \underline{u}(t+1) - \underline{\underline{K}} \underline{a}(t+1) - v_0(t+1) \underline{\underline{W}}^T \underline{a}(t+1) \Big)$$

It can be shown that this leads to the following system for the solution degrees of freedom at the next time step:

$$\underline{a}(t+1) = \Big(\underline{\underline{I}} + \Delta t \underline{\underline{M}}^{-1} \underline{\underline{K}} + \Delta t \underline{\underline{M}}^{-1} (v_0(t+1) \underline{\underline{W}}^T) \Big)^{-1} \Big(\underline{a}(t) + \Delta t \underline{\underline{M}}^{-1} \underline{\underline{B}} \underline{u}(t+1) \Big)$$

Note that we can simplify this, and make the notation agree with the standard form of the control problem, by defining

$$\tilde{\underline{\underline{A}}} := -\underline{\underline{M}}^{-1} \underline{\underline{K}} - \underline{\underline{M}}^{-1} v_0 \underline{\underline{W}}^T$$

$$\tilde{\underline{\underline{B}}} := \underline{\underline{M}}^{-1} \underline{\underline{B}}$$

where the velocity magnitude v_0 is taken as a constant. The governing equation for the plant is then

$$\dot{\underline{a}} = \tilde{\underline{\underline{A}}} \underline{a} + \tilde{\underline{\underline{B}}} \underline{u}(t)$$

These equations can be time integrated using the backward Euler method with

$$\underline{a}(t+1) = (\underline{\underline{I}} - \Delta t \tilde{\underline{\underline{A}}})^{-1} \Big(\underline{a}(t) + \Delta t \tilde{\underline{\underline{B}}} \underline{u}(t+1) \Big)$$

Defining measurements

We have formulated the governing equations for the system itself, so the next step in setting up the control problem is to define measurements of the system state. The concentration will be measured on a uniform rectangular grid at M sensor locations. The control input will not show up in measurements of the output of the system, as we only care about the concentration. Using the discretization of the solution and defining the state measurement vector as $\underline{y}$, we have that

$$y_i = c(x_i, t) = \sum_j a_j(t) f_j(x_i) \quad i = 1, 2, \ldots, M$$

$$\implies \underline{y}(t) = \underline{\underline{C}}\,\underline{a}(t) + \underline{\underline{0}}\,\underline{u}(t), \quad C_{ij} := f_j(x_i)$$

See Figure 3 for an example of a measurement grid with $M = 5$. As we will see later on, it is advantageous to choose the measurement locations as coinciding with the locations of the control inputs. This is what we will do.

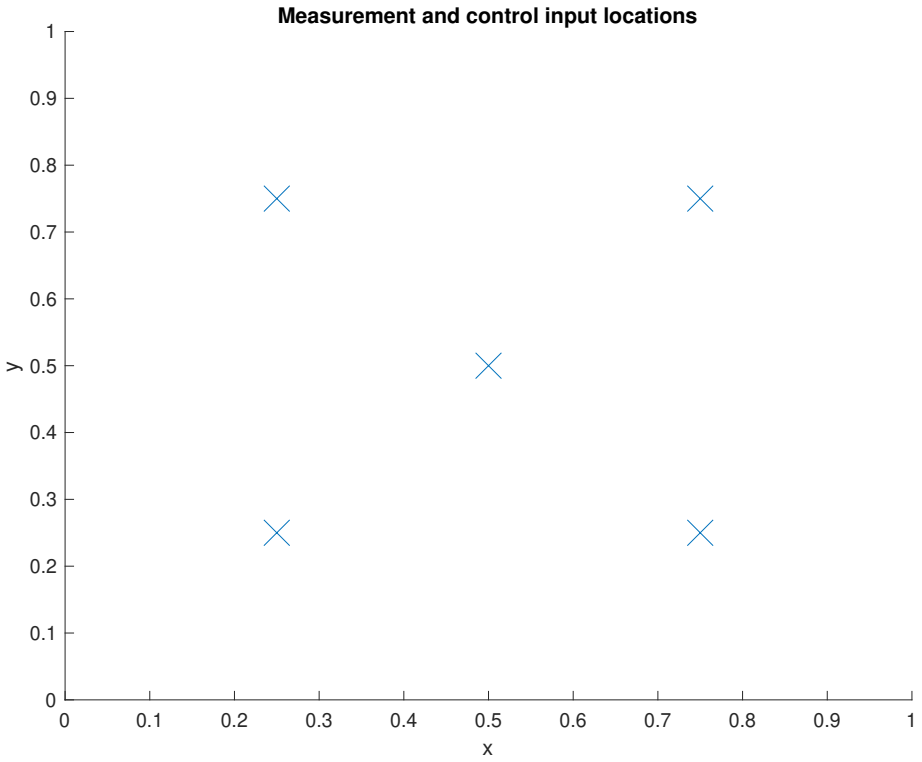

Figure 3: There are 5 coincident control and measurement locations spread throughout the domain.

System stability

The system is implemented in MATLAB using numerical integration to form all the system matrices. See the attached code. We use a Fourier basis to discretize the solution of the form

$$f_i(x_1, x_2) = \sin(n\pi x_1)\sin(m\pi x_2) \quad i = 1, \ldots, \tilde{N}$$

The zero concentration boundary conditions are automatically respected. The frequencies n and m in the x_1 and x_2 directions respectively are a function of the single index i. This is essentially a reshape operation from a matrix of shape functions up to a maximum frequency $N = \sqrt{\tilde{N}}$. The details can be seen in the code. We choose $N = 3$, meaning that the highest frequency shape function is $\sin(3\pi x)$ and there are $\tilde{N} = 9$ shape functions in the basis. This means there are 9 degrees of the freedom in the solution. The settings for other problem parameters are $D = 1E-2$, $v_0 = 1E3$, and the exponent on the velocity field is $n = 10$. There are 5 control inputs and measurements, as seen in Figure 1.

The first question we address is the stability of the system. The matrix $\tilde{A}$ governs the dynamics of the system. The eigenvalues of this matrix are used to study its stability properties. Note that we can verify in MATLAB that $\underline{\underline{\tilde{A}}}$ is full rank, meaning that there is a trivial right nullspace, and the only equilibrium point is $\underline{a}^* = 0$. This makes sense given the physics of our system–there would be no reason to expect any non-zero concentration field to be stable in the absence of source terms given the homogeneous boundary conditions. The zero boundary conditions act as sinks. See Figure 4 for eigenvalue analysis of the system's stability.

```
>> [V,lam] = eig(Atilde);
>> diag(lam)

ans =

  -0.1974 + 0.0000i
  -1.2801 + 6.1062i
  -1.2801 - 6.1062i
  -0.8883 + 4.1239i
  -0.8883 - 4.1239i
  -0.4964 + 1.3930i
  -0.4964 - 1.3930i
  -0.9870 + 0.0000i
  -1.7765 + 0.0000i
```

Figure 4: The eigenvalues of the system are seen to be distinct, which means that the matrix $\tilde{A}$ is semi-simple. The real parts of all the eigenvalues are strictly less than zero, which means that the system is both Lyapunov stable, and asymptotically stable. This means that not only does a state starting near an equilibrium point stay near that equilibrium point (Lyapunov stable), but that it converges to the equilibrium point (asymptotically stable). Informally, we can think of this as the system having damping, which means that the oscillations around/near equilibrium eventually die out.

The set of eigenvectors is computed using MATLAB. The span of the eigenvectors defines the eigenspace. It is not shown here because the matrix does not display well (too large given the complex components), but it can be seen in the code. We note that complex eigenvalues come in conjugate pairs, as do their corresponding eigenvectors. Say that for a complex eigenvalue λ, the corresponding eigenvector is $\underline{u} + i\underline{v}$. We know that $\bar{\lambda}$ is also an eigenvalue, and its corresponding eigenvector is $\underline{u} - i\underline{v}$. The modal space comes from the recognition that $\text{span}(\underline{u}+i\underline{v}, \underline{u}-i\underline{v}) = \text{span}(\underline{u}, \underline{v})$ given that the coefficients are complex. Thus, the real modal space is formed by replacing a pair of complex conjugate eigenvectors with their real and imaginary parts in the set. The vectors $\underline{u}$ and $\underline{v}$ are no longer eigenvectors, but when a solution starts in their span it stays in it. Oscillations arise from the solutions rotation through the span of these two vectors. The real modal space is formed in MATLAB by manually picking out the complex conjugate pairs of eigenvectors. The vectors spanning the real modal space are shown in the columns of the matrix $\underline{\underline{Vm}}$ in Figure 5.

We can use each of the vectors in the real modal space as an initial condition to the unforced advection-diffusion system. This is implemented in MATLAB and an animated plot of the resulting 2D concentration field is generated. We expect that initial conditions taken from the modal space with complex eigenvalues will oscillate, whereas real (and negative) eigenvalues lead to pure exponential decay of the corresponding eigenvector. We can verify this by plotting the concentration response at a point in the domain for initial conditions from the modal space corresponding to real and complex eigenvalues. See Figure 6-8. As can be seen in the animated plot, oscillation here means that the initial condition rotates around the domain, modulating the magnitude of a fixed spatial form of the concentration. This is a consequence of the vortex-like fluid velocity field advecting the concentration field. This is also a consequence of rotation in the plane spanned by the two vectors from the modal space corresponding to a complex eigenvalue.

```
Vm =

   -1.0000    0.0000    0.0000   -0.0000   -0.0000   -0.0000    0.0000   -0.0000   -0.0000
   -0.0000   -0.0045   -0.0430    0.0000   -0.0000   -0.0000   -0.7058   -0.0000   -0.0000
   -0.0000   -0.0000    0.0000   -0.0120    0.4999   -0.0000   -0.0000   -0.7071   -0.0000
   -0.0000   -0.0430    0.0045    0.0000    0.0000    0.7058         0    0.0000   -0.0000
   -0.0000   -0.0000   -0.0000   -0.7071         0    0.0000   -0.0000    0.0000    0.0000
    0.0000    0.7058         0   -0.0000    0.0000    0.0430    0.0045    0.0000    0.0000
    0.0000    0.0000   -0.0000    0.0120   -0.4999    0.0000    0.0000   -0.7071    0.0000
    0.0000   -0.0000    0.7058   -0.0000   -0.0000    0.0045   -0.0430    0.0000    0.0000
   -0.0000   -0.0000   -0.0000   -0.0000    0.0000    0.0000   -0.0000    0.0000    1.0000
```

Figure 5: The set of vectors which form the real modal space.

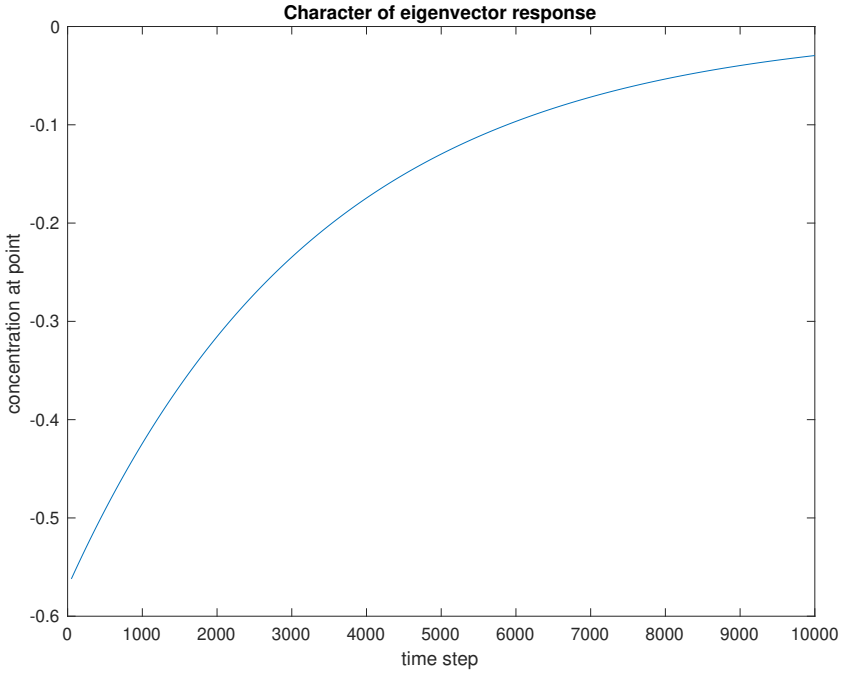

Figure 6: The first vector in the real modal space corresponds to a real eigenvalue, thus the response of the system (taken at a given spatial point) to this initial condition is pure exponential decay.

Reachability

We now want to investigate the reachability of the advection-diffusion system. This analysis will determine which states of the system can be obtained as the result of control inputs. We will assume that our system is in state $\underline{a}(0)$, and that we want to determine whether a control law exists which obtains the state $\underline{a}(t_1)$ over the time interval t_1. Using the general solution of the inhomogeneous linear system of differential equations, we can write

$$\underline{a}(t_1) = e^{\underline{\underline{\tilde{A}}}t_1}\underline{a}(0) + \int_0^{t_1} e^{\underline{\underline{\tilde{A}}}(t_1-\tau)}\underline{\underline{\tilde{B}}}\underline{u}(\tau)d\tau$$

Multiplying both sides of the equation by $e^{-\underline{\underline{\tilde{A}}}t_1}$ and rearranging, we have

$$e^{-\underline{\underline{\tilde{A}}}t_1}\underline{a}(t_1) - \underline{a}(0) = \int_0^{t_1} e^{-\underline{\underline{\tilde{A}}}\tau}\underline{\underline{\tilde{B}}}\underline{u}(\tau)d\tau$$

The left-hand side of the equation is a function of the initial and final states and will be called $\underline{z}$. If any combination of initial condition and final state should be reachable, then the control input should be capable of producing any $\underline{z}$. To investigate under what conditions this is possible, we can use the series expansion of the matrix exponential to write

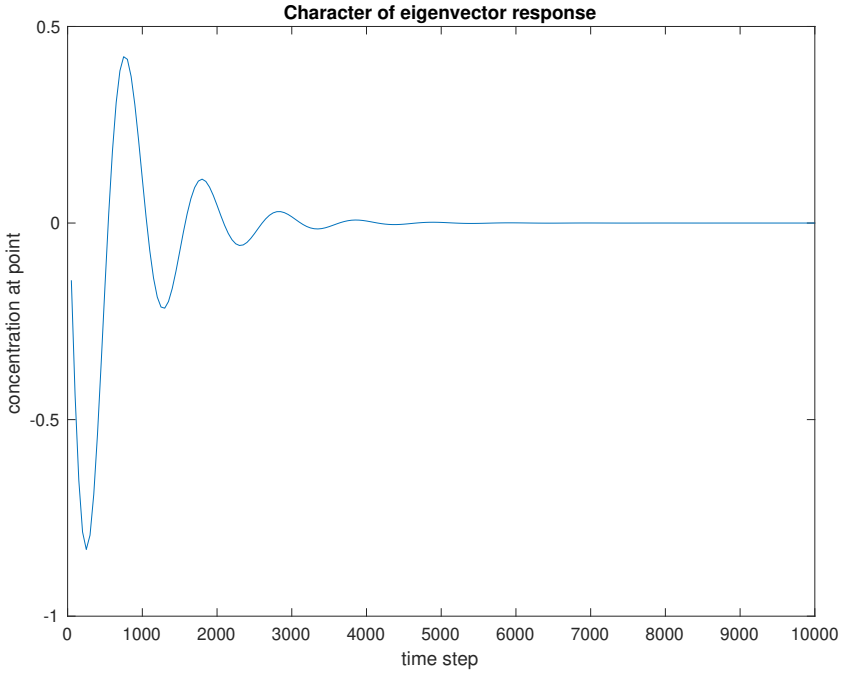

Figure 7: The second vector in the real modal space corresponds to a complex eigenvalue, this the response of the system (taken at a given spatial point) to this initial condition is oscillatory.

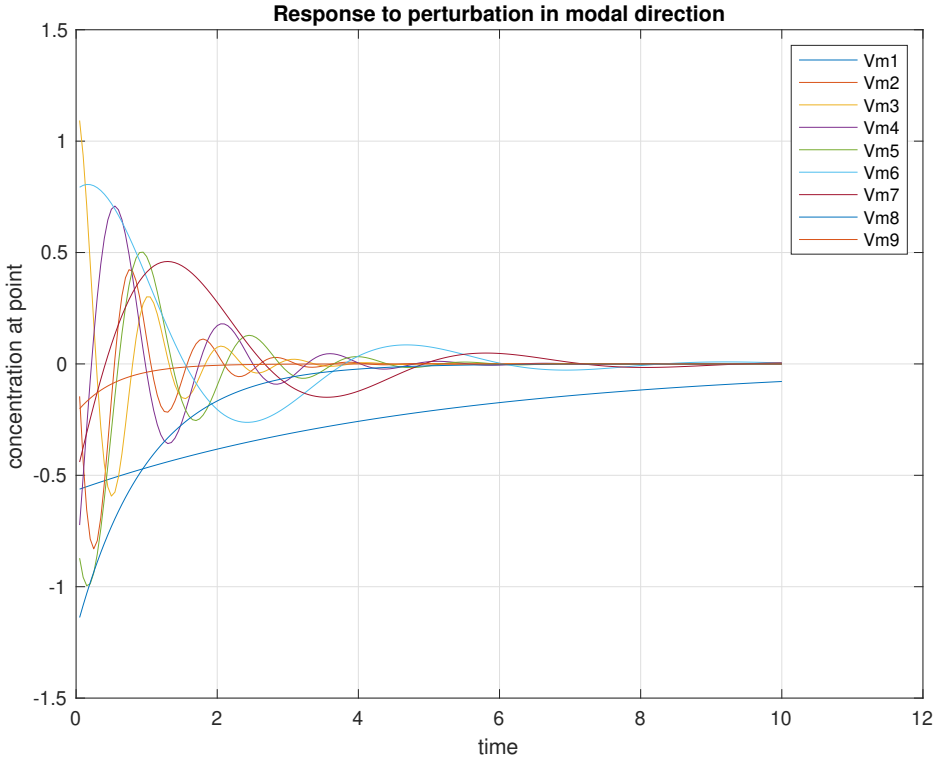

Figure 8: Decay of concentration at a point from initial conditions defined by each element of the real modal space. We cannot show the decay of the full 2D concentration field without animation. See the MATLAB code.

$$\underline{z} = \int_0^{t_1} \left(\sum_{i=1}^{\infty} \frac{\underline{\underline{\tilde{A}}}^i (-\tau)^i}{i!} \right) \underline{\underline{\tilde{B}}}\,\underline{u}(\tau) d\tau = \int_0^{t_1} \left(\sum_{j=0}^{\tilde{N}-1} f_j(\tau) \underline{\underline{\tilde{A}}}^j \right) \underline{\underline{\tilde{B}}}\,\underline{u}(\tau) d\tau = \sum_{j=1}^{\tilde{N}-1} \underline{\underline{\tilde{A}}}^j \underline{\underline{\tilde{B}}}\,\underline{\delta}_j$$

where the second equality comes from the Cayley-Hamilton theorem, which allows us to write this infinite series exactly as a finite series, and the third equality comes from the definition

$$\underline{\delta}_j := \int_0^{t_1} f_j(\tau)\underline{u}(\tau)d\tau$$

Thus, we can write

$$\underline{z} = \begin{bmatrix} \underline{\underline{\tilde{B}}} & \underline{\underline{\tilde{A}}}\,\underline{\underline{\tilde{B}}} & \underline{\underline{\tilde{A}}}^2\underline{\underline{\tilde{B}}} & \dots & \underline{\underline{\tilde{A}}}^{\tilde{N}-1}\underline{\underline{\tilde{B}}} \end{bmatrix} \begin{bmatrix} \underline{\delta}_1 \\ \underline{\delta}_2 \\ \vdots \end{bmatrix}$$

Given that dimension of $\underline{z}$ is $\tilde{N}$, it is clear that if the above matrix has rank $\tilde{N}$, then any state can be reached with appropriate choice of the $\underline{\delta}_i$. We form this matrix in MATLAB, which we will call $\underline{\underline{P}}$, and test its rank. We see that the rank is $9 = \tilde{N}$, meaning that the whole state space is reachable. Because the whole state space is reachable, we can use unit vectors aligned with the coordinate directions as an orthornormal basis. Thus, the basis is

$$\hat{\underline{b}}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \end{bmatrix}, \hat{\underline{b}}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \end{bmatrix}, \dots, \hat{\underline{b}}_{\tilde{N}} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}$$

Note that each "unit vector" corresponds to a spatial distribution of the concentration defined according to the Fourier basis. The energy of the control input over the time interval t_1 is defined to be

$$E_{[0,t_1]} = \int_0^{t_1} \underline{u}(\tau) \cdot \underline{u}(\tau)d\tau$$

It can be shown that the minimum energy control input is obtained when

$$\underline{u}(t) = \underline{\underline{\tilde{B}}}^T e^{-\underline{\underline{\tilde{A}}}^T t}\underline{v}$$

where $\underline{v}$ are unknown coefficients chosen for the particular combination of initial and final conditions. The energy associated with the minimum energy control law is

$$E_{[0,t_1]} = \underline{v} \cdot \left(\int_0^{t_1} e^{-\underline{\underline{\tilde{A}}}\tau}\underline{\underline{\tilde{B}}}\,\underline{\underline{\tilde{B}}}^T e^{-\underline{\underline{\tilde{A}}}^T \tau} d\tau \right) \underline{v} := \underline{v} \cdot \underline{\underline{G}}_{[0,t_1]}\underline{v}$$

It is easy to see than when using the minimal energy form of the control law, the reachability condition is

$$\underline{z} = \underline{\underline{G}}_{[0,t_1]}\underline{v}$$

This is another test of reachability: if this so-called "controllability Grammian" $\underline{\underline{G}}$ is full rank, any transition between initial and final states, encoded by $\underline{z}$, can be obtained in the given time frame by solving for $\underline{v}$.

We can now assess the energy required to restore the system to its zero equilibrium condition (the only equilibrium for this advection-diffusion system) in the time $t_1 = 10$. Thus, the final state is $\underline{a}(10) = \underline{0}$, and the initial states will be perturbations defined by the basis vectors. This means that the coefficients on the control law for returning the system to equilibrium are

$$\underline{v}_i = -\underline{\underline{G}}^{-1}_{[0,t_1]}\hat{\underline{b}}_i$$

This follows from the definition of $\underline{z}$, where, because the final state is $\underline{0}$, the term involving the matrix exponential drops out. The energy for the ith perturbation is simply

$$E_{[0,t_1],i} = \underline{v}_i\underline{\underline{G}}_{[0,t_1]}\underline{v}_i$$

This is implemented in MATLAB. See Figure 9 for the results.

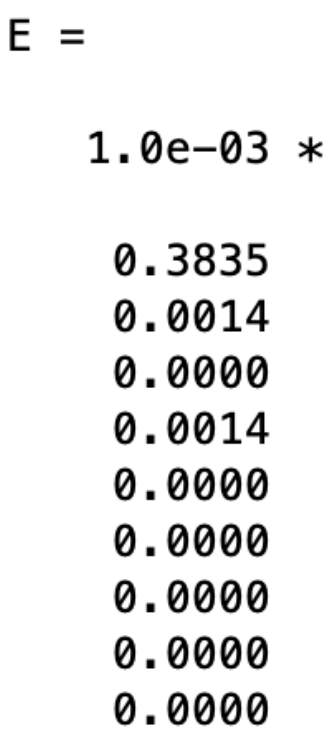

Figure 9: The energy values to restore the perturbations in the 9 coordinate directions to equilibrium in the allotted time period. The energies are very small because the system is asymptotically stable, meaning that it does not require large control inputs to reach equilibrium. The higher frequency distributions of concentration decay more quickly and thus require almost no control input.

Closed loop pole placement

We now want to explore placing the eigenvalues of the system through state variable feedback. All of the modes of the system can be influenced by closed loop control. To see this, note that the control input is taken to be of the form

$$\underline{u}(t) = -\underline{\underline{K}}\,\underline{a}(t) \implies \dot{\underline{a}}(t) = (\underline{\underline{\tilde{A}}} - \underline{\underline{\tilde{B}}}\,\underline{\underline{K}})\underline{a}(t)$$

The feedback control law gives rise to new system dynamics. The question is: can we choose the matrix $\underline{\underline{K}}$ such that this system has a specified set of eigenvalues? Note that the eigenvalue equation for this new system is

$$(\underline{\underline{\tilde{A}}} - \underline{\underline{\tilde{B}}}\,\underline{\underline{K}})\underline{\Psi} = \lambda\underline{\Psi}$$

Through some minor algebraic manipulations, this equation can be written as

$$\begin{bmatrix}\lambda_i\underline{\underline{I}} - \underline{\underline{\tilde{A}}} & \underline{\underline{\tilde{B}}}\end{bmatrix}\begin{bmatrix}\underline{\Psi}_i \\ \underline{\underline{K}}\,\underline{\Psi}_i\end{bmatrix} = 0$$

where the index i has been added to the eigenvalues and eigenvectors to be explicit about the fact that there are $\tilde{N}$ such equations. The eigenvalues λ_i are now considered to be "user-specified." This is what is meant by placing them. We note that if the eigenvalues are chosen to be distinct, the eigenvectors will be independent when the system is reachable. This comes from the fact that the closed loop feedback does not change reachability, and that assuming that the closed system is reachable can be shown to generate a contradiction with the assumption of dependent eigenvectors. We have already shown that the system is reachable, thus the eigenvectors $\underline{\Psi}_i$ are independent. The above equation says that the vector $[\underline{\Psi}, \underline{\underline{K}}\,\underline{\Psi}]$ is an element of the nullspace of the matrix it multiplies. This matrix has more columns than rows, so the nullspace is always nontrivial. Once the eigenvalue λ_i is selected, we can compute a vector in its nullspace. The first $\tilde{N}$ entries of this vector are the eigenvector $\underline{\Psi}_i$. The remaining entries we will call $\underline{\alpha}_i$. In order to be able to place all of the eigenvalues, it must be the case that

$$\underline{\alpha}_i^{[5\times 1]} = \underline{\underline{K}}^{[9\times 9]}\underline{\Psi}_i^{[9\times 1]} \quad i = 1, 2, \ldots, \tilde{N}$$

where the $\underline{\alpha}_i$ and $\underline{\Psi}_i$ are numerically computed from an element of the matrix's nullspace. The dimensions of each of these quantities is shown as a superscript. Thus, the only unknown in this set of equations is the matrix $\underline{\underline{K}}$. In our case, $\underline{\underline{K}}$ has $5 * 9 = 45$ components. For each index i, there are 5 equations in the system, and there are 9 such systems. Because there are 45 equations and 45 unknowns, the "system of systems" above has a solution, and we can place the eigenvalues however we want, so long as they are distinct.

One potentially desirable property of a control law is that the system does not oscillate. We can choose the eigenvalues so that the solution has no oscillations by zeroing the imaginary part. Thus, the user-specified eigenvalues λ_i will be purely real. Similarly, we can control the speed of the system's response through the eigenvalues. More negative eigenvalues lead to a faster rate of decay. Because this is a macroscopic mechanical system, we expect responses on the order of seconds or longer. When the eigenvalues are real and negative, the solution will scale with

$$\underline{a}(t) \propto e^{\lambda_i t} = e^{-t/\tau_i}$$

where τ_i is a time constant. The time constant controls the rate of decay, where large time constants mean the system responds slowly. We will choose

$$\lambda_i = -\frac{1}{\tau_i} = -\frac{1}{i} \quad i = 1, 2, \ldots, \tilde{N}$$

This ensures that the system response is on the order of seconds, which is what we intuitively expect for a mechanical system of this sort. Intuitively, diffusion of a chemical substance over the unit square on the order of minutes seems quite slow, and fractions of a second seems unrealistically fast. There is some deliberate ambiguity about units here (unwanted additional considerations), but as we have already seen, the natural decay rates as determined by the eigenvalues are already on the order of seconds. Thus, we do not push the system too hard by making this choice.

Solving the pole placement system

We are now in a position to determine the matrix $\underline{\underline{K}}$ such that the eigenvalues are placed as outlined in the past section. A linear system governs the entries of this matrix, but it needs to be rearranged to be in matrix vector form (we have a matrix of unknowns as it stands). Call the rows of this unknown gain matrix $\underline{K}_i$ for $i = 1, 2, \ldots, 5$. We can write the governing equations as

$$\begin{bmatrix} — & \underline{K}_1^T & — \\ — & \underline{K}_2^T & — \\ & \vdots & \end{bmatrix} \underline{\Psi}_1 = \underline{\alpha}_1, \begin{bmatrix} — & \underline{K}_1^T & — \\ — & \underline{K}_2^T & — \\ & \vdots & \end{bmatrix} \underline{\Psi}_2 = \underline{\alpha}_2, \ldots$$

Some serious rearranging/reshaping operations allows us to write

$$\begin{bmatrix} — & \underline{\Psi}_1^T & — & & & & \\ & — & \underline{\Psi}_1^T & — & & & \\ & & — & \underline{\Psi}_1^T & — & & \\ & & & — & \underline{\Psi}_1^T & — & \\ — & \underline{\Psi}_2^T & — & & & & \\ & — & \underline{\Psi}_2^T & — & & & \\ & & — & \underline{\Psi}_2^T & — & & \\ & & & \vdots & & & \end{bmatrix}_{[45\times 45]} \begin{bmatrix} \underline{K}_1 \\ \underline{K}_2 \\ \vdots \end{bmatrix}_{[45\times 1]} = \begin{bmatrix} \underline{\alpha}_1 \\ \underline{\alpha}_2 \\ \vdots \end{bmatrix}_{[45\times 1]}$$

All entries outside the eigenvectors are zero. This is a linear system for the rows of $\underline{\underline{K}}$. This can be solved, and then the matrix form can be reconstructed with a reshape operation. We can verify that this approach has worked by checking the eigenvalues of the state variable feedback system agree with the ones we specified. This is confirmed in Figure 10.

Because the eigenvalues are all real by construction, the eigenvectors will also be real. This means there is no difference between the modal space and the set of eigenvectors. See Figure 11 for eigenvectors (spatial modes) of the new system.

```
>> [v,d] = eig(Atilde-Btilde*K);
>> diag(d)

ans =

   -1.0000
   -0.5000
   -0.3333
   -0.2500
   -0.2000
   -0.1667
   -0.1111
   -0.1429
   -0.1250
```

Figure 10: The order is shuffled around a bit, but we see agreement between the specified eigenvalues, of the form $-1/i$, and eigenvalues of the feedback system with the computed entries of $\underline{\underline{K}}$.

```
v =

   -0.2738   -0.3629    0.3688    0.3706   -0.3676    0.3684    0.3676    0.3685    0.3695
    0.1542    0.1270   -0.1317   -0.0796    0.1209    0.1443    0.1838    0.1556    0.1415
    0.2110    0.1926   -0.1617   -0.1279    0.1646    0.1749    0.1921    0.1729    0.1649
   -0.8348   -0.8290    0.8057    0.7929   -0.8070   -0.7998   -0.7986   -0.7995   -0.7994
   -0.0010    0.0432   -0.0620   -0.0717    0.0749    0.0796    0.0839    0.0811    0.0827
    0.1071    0.1152   -0.1122   -0.1071    0.1109    0.1113    0.1138    0.1119    0.1107
   -0.2483   -0.3245    0.3916    0.4396   -0.3907   -0.3911   -0.3666   -0.3873   -0.3948
   -0.0814   -0.0046   -0.0178   -0.0354    0.0379    0.0390    0.0417    0.0408    0.0451
    0.2830    0.0860   -0.0347   -0.0128    0.0006    0.0070    0.0190    0.0123    0.0162
```

Figure 11: The columns are eigenvectors of the system altered by the state variable feedback control with the $\underline{\underline{K}}$ matrix designed to obtain given eigenvalues.

We can check that the system is doing as we expect by first simulating the response to eigenvector initial conditions. We expect pure decay of the concentration for the feedback system with eigenvector initial conditions. We pick an arbitrary point in the spatial domain, and observe the response. This is shown in Figure 12. An animated plot can also be found in the MATLAB code to visualize the full-field response of the system.

Having verified that the system is doing what we expect, we can visualize the response to unit perturbations in the coordinate directions, which is the orthonormal basis that we have chosen. We pick the same spatial point, and plot the concentration vs. time. See Figure 13. Note that we are only picking a point because we cannot visualize the trajectory of the 2D concentration field in time without animation. See the MATLAB code.

We can compare the control input $\underline{u}(t)$ obtained from state variable feedback control with the minimum energy control input. There are 5 control inputs for each of the control methods for each of 9 unit perturbations. Instead of showing 90 plots, the MATLAB code is set up to easily switch which unit perturbation is shown. We include the control inputs for the first three unit perturbations here. See Figures 14-16. In general, we note that the minimum energy and the state variable feedback methods lead to very different control inputs.

Input gain matrix

The last task that we have is to ensure that the system accurately tracks reference inputs. So far, we have designed control laws to bring the system back to its zero equilibrium state, and have shown that it is possible

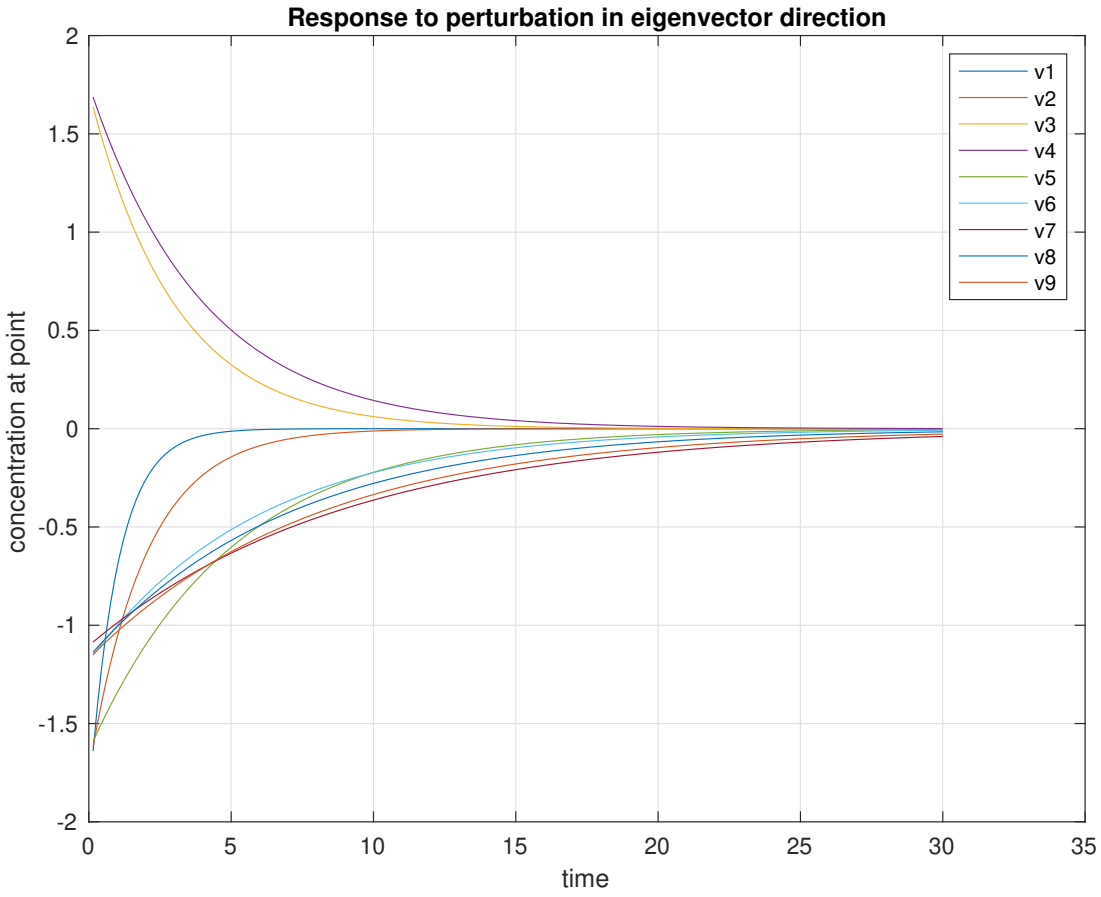

Figure 12: The concentration at the chosen spatial decays with time for each of the 9 eigenvector initial conditions. This is what we expect, given that the eigenvalues were chosen to be purely real and negative.

to place the eigenvalues which govern this response. Now we want the system to follow a specified trajectory. The control input is now of the form

$$\underline{u}(t) = \underline{\underline{K}}\,\underline{a}(t) + \underline{\underline{F}}\,\underline{r}(t)$$

where $\underline{r}(t)$ is a reference input which has the same dimensions as the control input, meaning that $\underline{\underline{F}}$ is a square "gain matrix." Given an input $\underline{r}(t)$, we want to choose the gain such that the system closely follows it. To see how this is accomplished, note that the governing equation for the system is

$$\dot{\underline{a}} = (\underline{\underline{\tilde{A}}} - \underline{\underline{\tilde{B}}}\,\underline{\underline{K}})\underline{a} + \underline{\underline{F}}\,\underline{r}(t)$$

Taking the component-wise Laplace transform of these dynamics and rearranging, we have that

$$\underline{a}(s) = \left(s\underline{\underline{I}} - \underline{\underline{\tilde{A}}} + \underline{\underline{\tilde{B}}}\,\underline{\underline{K}}\right)^{-1} \underline{\underline{\tilde{B}}}\,\underline{\underline{F}}\,\underline{r}(s)$$

Noting that governing equation for the output of the system is $\underline{y} = \underline{\underline{C}}\,\underline{a}$, we can take the Laplace transform of this and substitute for the solution degrees of freedom to obtain

$$\underline{y}(s) = \underline{\underline{C}}\left(s\underline{\underline{I}} - \underline{\underline{\tilde{A}}} + \underline{\underline{\tilde{B}}}\,\underline{\underline{K}}\right)^{-1} \underline{\underline{\tilde{B}}}\,\underline{\underline{F}}\,\underline{r}(s) := \underline{\underline{T}}(s)\underline{r}(s)$$

Now say that we want the output $\underline{y}$ of the system to track the input $\underline{r}$. One way this could be accomplished is to ensure that each component of the output tracks a corresponding step input. In other words, we want that for a unit step function applied to the i-th control input, the correspond measurement $y_i(t)$ converges to 1 in the limit. We can use the final value theorem, familiar from a study of Laplace transforms, to write

$$1 = \lim_{t\to\infty} y_i(t) = \lim_{s\to 0} s y_i(s) = \lim_{s\to 0} \sum_j s T_{ij}(s) r_j(s)$$

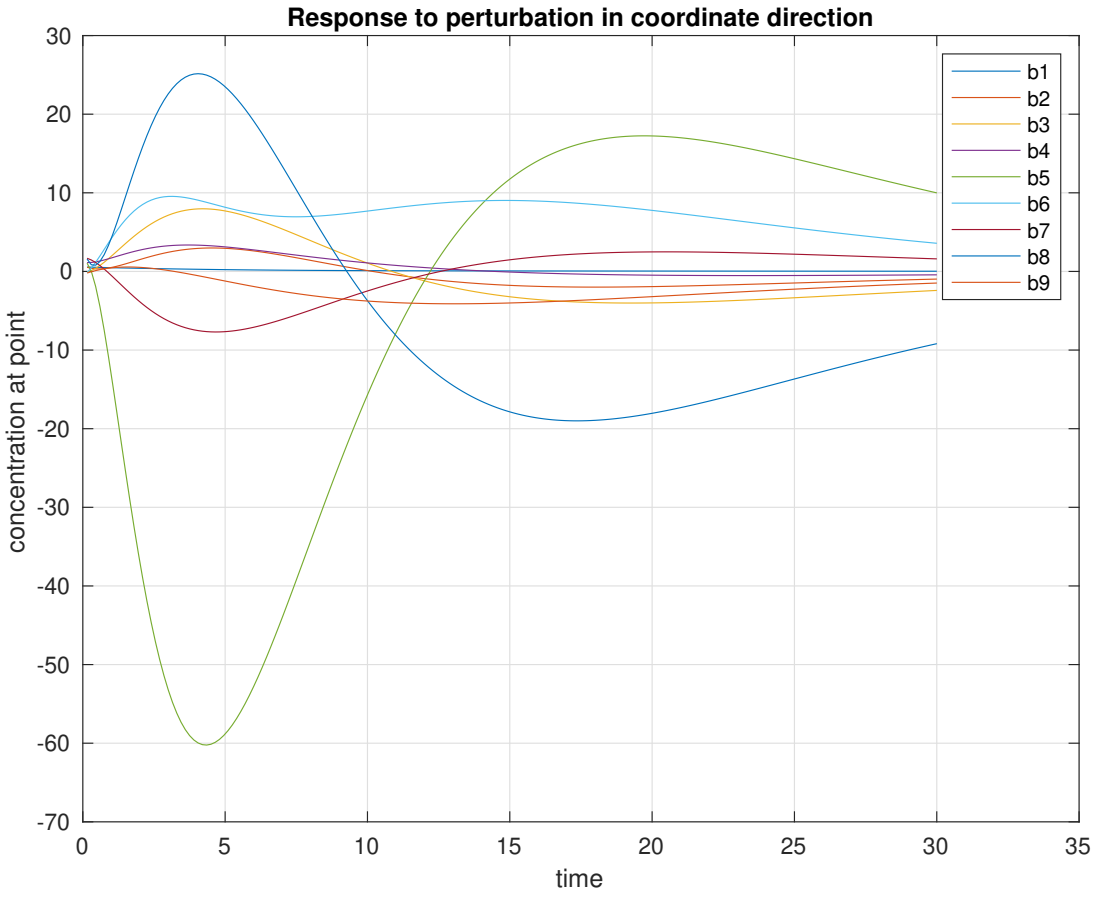

Figure 13: When the initial condition is in a coordinate direction, we no longer see pure decay of the concentration. This may seem counterintuitive, but it is a consequence of the Fourier basis we have used to discretize the concentration in space. Just because the coefficients on the basis functions decay in time does not mean that their weighted sum decays in time, as there can be constructive/destructive interference. That being said, we see that all solutions are settling to the zero equilibrium state, which is to be expected from the eigenvalues being real and negative. Playing with the sliders in this plot can be used to validate the claim that decaying coefficients does not lead to global decay in the solution. In other words, even though the coefficients on the Fourier basis may monotonically decrease, the concentration need not decrease monotonically.

Noting that, by construction, $\underline{r}(s)$ is the Laplace transform of a step input for component i, and that the Laplace transform of a step function is $1/s$, this becomes

$$1 = \lim_{s \to 0} T_{ii}(s) = T_{ii}(0)$$

Thus, we see that the diagonal entries of the transfer function evaluated at $s = 0$ must all be unity. Returning to its definition, this means that

$$\left[\underline{\underline{C}}\left(-\underline{\underline{\tilde{A}}} + \underline{\underline{\tilde{B}}}\,\underline{\underline{K}}\right)^{-1}\underline{\underline{\tilde{B}}}\,\underline{\underline{F}}\right]_{ii} = 1$$

Remember that $\underline{\underline{F}}$ is unknown here, and we are trying to design it such that this condition is met. One particular matrix that has 1's along the diagonal is the identity. This means that we can satisfy this condition with

$$\underline{\underline{F}} = \left[\underline{\underline{C}}\left(-\underline{\underline{\tilde{A}}} + \underline{\underline{\tilde{B}}}\,\underline{\underline{K}}\right)^{-1}\underline{\underline{\tilde{B}}}\right]^{-1}$$

It turns out that the requirements on the diagonal are necessary conditions, whereas this formula is a necessary and sufficient condition to obtain convergence. Note that this requires that number of measurements taken, given by the number of rows in $\underline{\underline{C}}$, must match the number of control inputs otherwise the matrix whose inverse we take is not square. See Figure 17 for the numerical values of this matrix. As a final note, if there is

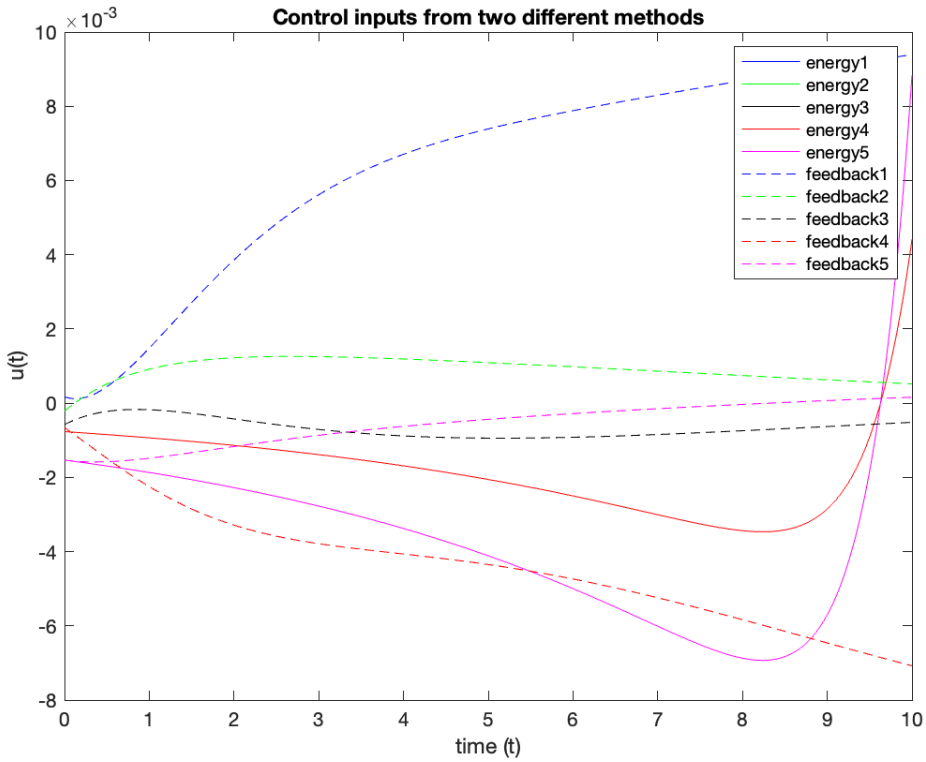

Figure 14: The magnitude of the control inputs from the two methods are comparable in the case of the first unit perturbation. Remember that these perturbations are in coordinate directions. The first unit coordinate direction corresponds to the spatial mode $\sin(\pi x_1)\sin(\pi x_2)$, which decays much more slowly than higher frequency modes. Thus, even the minimum energy control law requires some nontrivial input to restore it to equilibrium in the given time frame. Due to the symmetry of the spatial mode and the symmetry of the control input's spatial distribution, the minimum energy method leads to four of the five control inputs being equivalent. On the other hand, the state variable feedback method, which was designed to obtain specified system eigenvalues, breaks the symmetry of the problem and each control input is different. This can be seen in animated plot of the decay of the first spatial mode (unit coordinate perturbation) with state variable feedback, which is no longer symmetric. Unlike the minimum energy method, the state variable feedback method does not force the concentration to zero. As the solution becomes smaller, so does the control input.

concern that a step function is a not a "low-frequency" forcing, given that it is discontinuous at the origin, we can see that the same requirements are put on the transfer function matrix for a reference input of

$$r_i(t) = 1 - e^{-t/c}$$

which behaves like a smoothed step function for very small c, and acts as a low frequency input for large c. This can be shown by following the same derivation as above.

The first thing that we can do is demonstrate the importance of designing the gain matrix for the output to track the input. In Figure 18, we take $\underline{\underline{F}}$ to be the identity and observe the concentration at each of the control/sensor locations. The system converges to steady state values, but these values differ wildly from the limiting value of the step function ($u(t) \to 1$). On the other hand, when the gain matrix is chosen appropriately, we see that the system tracks the input $\underline{r}(t)$. This can be seen in Figure 18. Note that each curve in these two figures corresponds to measurements of the concentration at the location of the step function input. Thus, these figures correspond to five different simulations of the system–one for a step input at each of the five control input locations.

The transient behavior seen in Figure 19 is what we expect given the eigenvalues of our closed loop system. The system oscillates in spite of the real and negative eigenvalues because of the representation of the concentration with a Fourier basis. A linear combination of sine functions can grow in magnitude at a point even when some or all of the coefficients are decreasing, as a consequence of interference. The oscillations are damped out on the time scale of seconds, given that the eigenvalues were placed for this to be the case. We can think of these

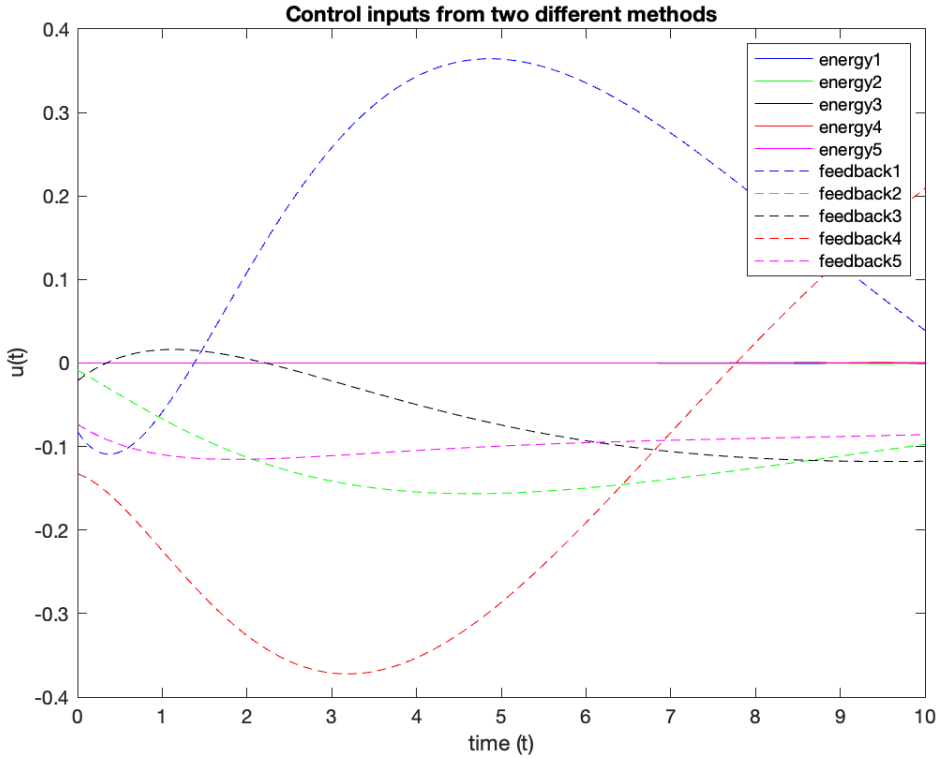

Figure 15: The unit perturbation in the second coordinate direction corresponds to a higher frequency spatial mode, and thus the minimum energy method requires essentially no actuation to restore it to equilibrium. On the other hand, as noted previously, decay of concentration fields that are not eigenvectors can exhibit complex oscillatory behavior in the case of state variable feedback. This is why the actuation is larger for the state variable feedback method.

oscillations as analogous to the oscillations seen in the unforced decay from perturbations in the coordinate directions. We could force the system to converge to its limiting value faster by designing $\underline{K}$ to make the eigenvalues more negative.

Observability

We are interested in investigating the observability of the system. A system with state space dynamics given by $\underline{\underline{\tilde{A}}}$ and measurements by $\underline{y} = \underline{\underline{C}}\underline{a}$ is said to be observable if any initial state $\underline{a}(0)$ can be distinguished from $\underline{0}$ through the measurement $\underline{y}$ in a finite time interval $[0, t_1]$. We can assess the conditions under which this is true using a similar approach to the reachability analysis we did previously. When the system is unforced, the state is given by $\underline{a}(t) = \exp(\underline{\underline{\tilde{A}}}t)\underline{a}(0)$, and the measurement is

$$\underline{y} = \underline{\underline{C}}e^{\underline{\underline{\tilde{A}}}t}\underline{a}(0) = \underline{\underline{C}}\left(\sum_{k=0}^{\tilde{N}-1}\alpha_k(t)\underline{\underline{\tilde{A}}}^k\right)\underline{a}(0)$$

where the second equality follows from the series definition of the matrix exponential and the Cayley-Hamilton theorem, which is used to truncate the sum. The functions $\alpha_k(t)$ can be known in theory, but are difficult to determine in practice. Moving the measurement matrix inside the sum and writing this expression as a matrix product, we have

$$= \begin{bmatrix}\alpha_0(t)\underline{\underline{I}} & \alpha_1(t)\underline{\underline{I}} & \dots & \alpha_{\tilde{N}-1}(t)\underline{\underline{I}}\end{bmatrix}\begin{bmatrix}\underline{\underline{C}} \\ \underline{\underline{C}}\underline{\underline{\tilde{A}}} \\ \vdots \\ \underline{\underline{C}}\underline{\underline{\tilde{A}}}^{\tilde{N}-1}\end{bmatrix}\underline{a}(0)$$

This shows that if $\underline{a}(0)$ is in the right nullspace of the matrix it multiplies, the measurement is zero for

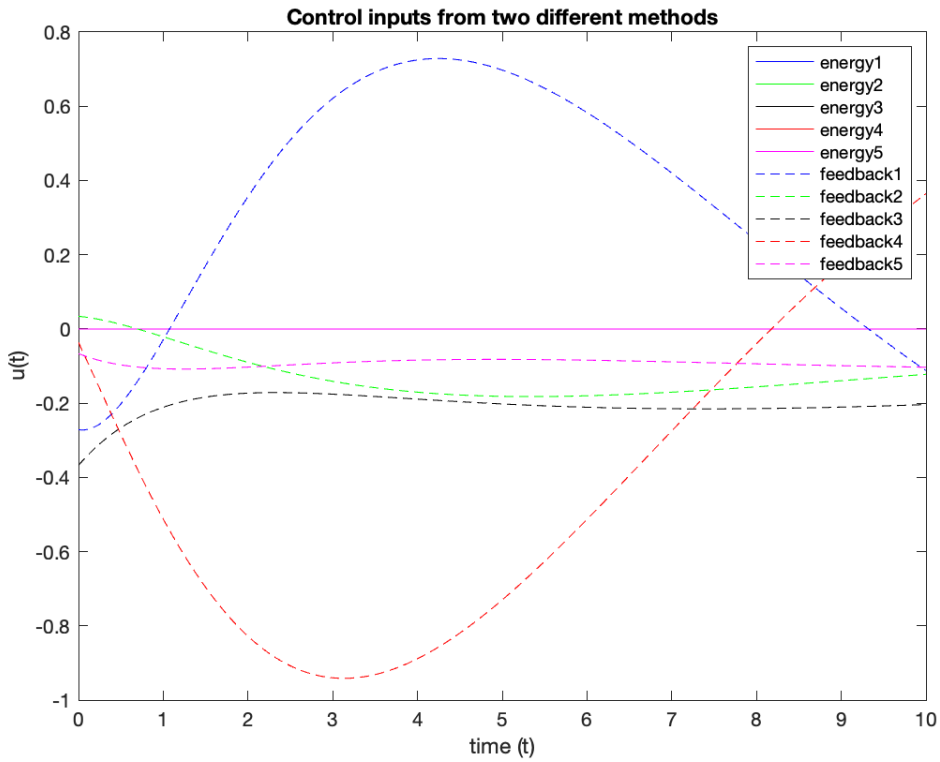

Figure 16: All further perturbations in coordinate directions exhibit the behavior outlined here. The minimum energy control input is very small because the higher frequency concentration fields decay quickly. The decay of non-eigenvector concentration fields leads to some oscillations before settling, which accounts for the larger state variable feedback control inputs.

```
F =

    0.0167   -0.0175    0.0148   -0.0113   -0.0013
    0.0507   -0.0553    0.0412   -0.0346   -0.0007
    0.0889   -0.0967    0.0694   -0.0608    0.0000
    0.0994   -0.1072    0.0767   -0.0672    0.0001
    0.1290   -0.1381    0.0822   -0.0916    0.0105
```

Figure 17: The entries of the gain matrix are all less than 1, meaning that it scales down the step input in order to obtain the desired steady state value.

all time. Thus, if this matrix has a trivial right nullspace, there are no initial conditions $\underline{a}(0)$ which are not registered by the measurement $\underline{y}$. Thus, we can test the rank of the "observability" matrix

$$\underline{\underline{O}} = \begin{bmatrix} \underline{\underline{C}} \\ \underline{\underline{C}}\underline{\underline{\tilde{A}}} \\ \vdots \\ \underline{\underline{C}}\underline{\underline{\tilde{A}}}^{\tilde{N}-1} \end{bmatrix}$$

in order to determine whether the system is observable. We form the observability matrix in MATLAB and show that it has rank $9 = \tilde{N}$, which means that the system is fully observable. There are no non-zero states which are invisible to the output of the system.

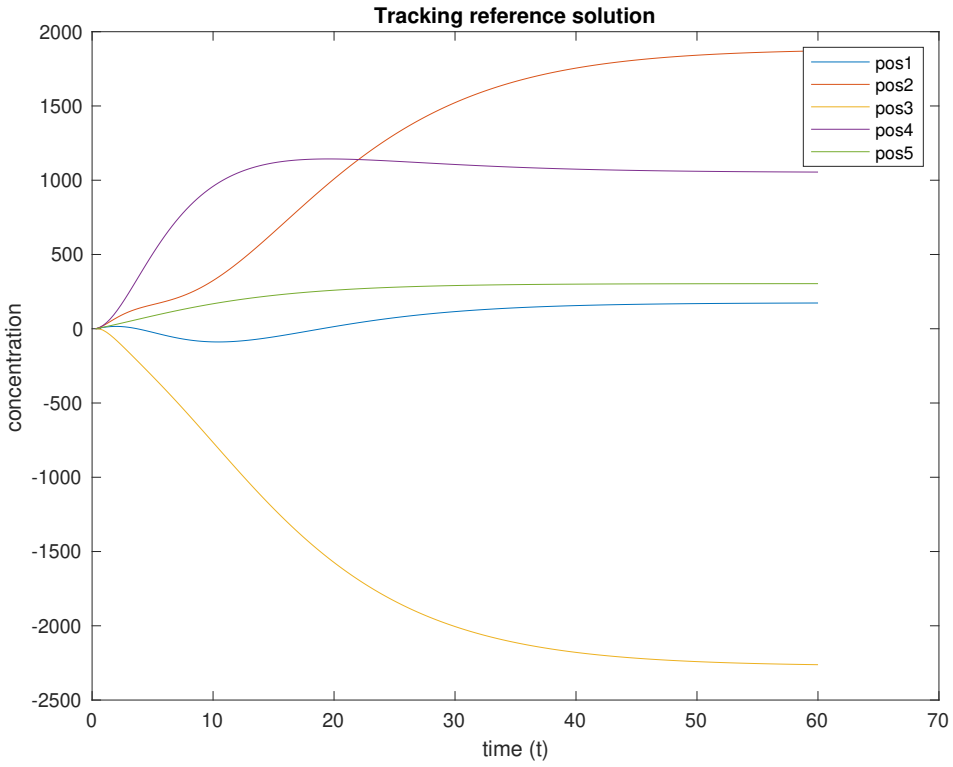

Figure 18: Without a properly designed gain matrix, the concentration at the measurement locations does not track the step input.

The energy of the output of the system over a given time interval is the time integral of the squared magnitude of the output. This reads

$$E_{[0,t_1]} = \int_0^{t_1} \underline{y}^T \underline{y} dt = \underline{a}(0)^T \left(\int_0^{t_1} e^{\underline{\underline{\tilde{A}}}^T \tau} \underline{\underline{C}}^T \underline{\underline{C}} e^{\underline{\underline{\tilde{A}}} \tau} d\tau \right) \underline{a}(0)$$

We can define the "observability Grammian," defined over a given time interval $[0, t_1]$, as

$$\underline{\underline{G}}_{[0,t_1]} := \int_0^{t_1} e^{\underline{\underline{\tilde{A}}}^T \tau} \underline{\underline{C}}^T \underline{\underline{C}} e^{\underline{\underline{\tilde{A}}} \tau} d\tau$$

This implies that the energy of the output over the given time interval can be written simply as

$$E_{[0,t_1]} = \underline{a}(0)^T \underline{\underline{G}}_{[0,t_1]} \underline{a}(0)$$

We can use this expression to compute the energy over a given time interval $t_1 = 10$ for the decay of unit initial conditions taken from the real modal space. We use the same time interval as in the earlier reachability analysis. See Figure 20 for results of this calculation for the 9 unit vectors forming the basis of the real modal space. The larger the energy associated with the output, the "more" observable the corresponding initial state is. This is best illustrated by looking at the eigenvalues and eigenvectors of the observability Grammian. The observability Grammian arises from the square of a quantity, which means that it always has non-negative eigenvalues. The size of the eigenvalues determines the degree of observability of the corresponding eigenvector. If the eigenvalues are small, this means that the energy associated with the corresponding eigenvector is small, and there is a "weak" signal $\underline{y}$. Though an initial state might be large in magnitude, its associated energy is small. See Figure 21 for the eigenvalues of our observability Grammian. There are some eigenvalues less than 1, but there are no eigenvalues which give rise to energies that are orders of magnitude smaller than the magnitude of the initial state. Thus, all states should have a good degree of observability.

We now want to determine if we can define an observer on the plant whose estimate of the system state converges to the true state. Not only do we want to enforce convergence of the estimate though, but we want to be able to control the rate of convergence through the eigenvalues of the observer dynamics. The Luenberger observer gain matrix $\underline{\underline{L}}$ is defined with

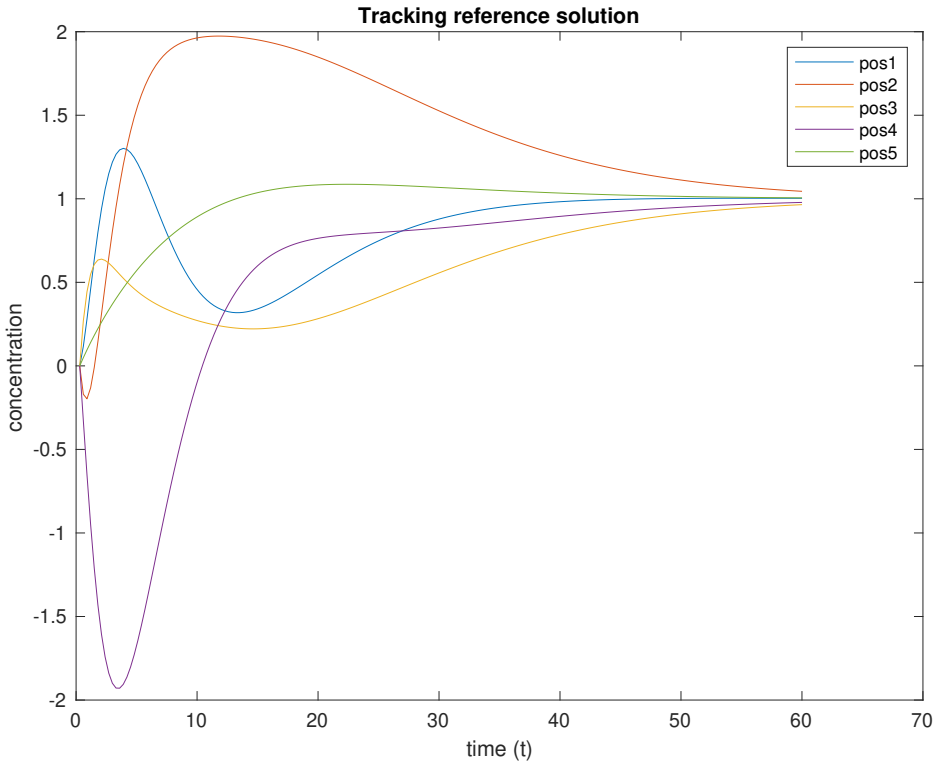

Figure 19: When the gain matrix is designed according to the procedure outlined above, the concentration at the measurement location tracks the input. The concentration at each measurement location converges to the appropriate limiting value after some transient behavior.

```
E =

    4.9683
    0.6864
    0.6864
    1.1750
    1.0764
    2.2586
    2.2586
    2.0264
    0.5629
```

Figure 20: The energy associated with the decay of initial conditions taken from the real modal space. The i-th entry corresponds to the i-th basis vector in the real modal space. In our case, we can think of the measurement $\underline{y}$ as a coarse snapshot of the full-field state, as it is a measurement of the concentration at each of the 5 control locations. Thus, the energy associated with the first element of the real modal space is largest because it has the slowest decay time, meaning $\underline{y}$ is larger for a longer time interval. Note that we verify these energy values against $\int y^T y dt$ in the code to ensure that observability Grammian is computed correctly.

$$\dot{\underline{\hat{a}}} = \underline{\underline{\tilde{A}}}\underline{\hat{a}} + \underline{\underline{\tilde{B}}}\underline{u} + \underline{\underline{L}}(\underline{y} - \underline{\hat{y}})$$

```
d =

    5.1902
    0.3410
    2.0264
    1.3954
    0.8684
    2.2913
    2.2913
    0.6477
    0.6477
```

Figure 21: Eigenvalues of the observability Grammian. There are no eigenvalues much less than 1, so the energies associated with each of the eigenvector initial states are comparable to the magnitude of the eigenvector initial states. This means that all eigenvectors of $\underline{\underline{G}}_{[0,t_1]}$ have a good degree of observability, and because the eigenvectors are independent, they form a basis for the whole state space. This means that all states have a good degree of observability.

where $\hat{a}$ is the estimated state and the observer gain is unknown. We have that the estimated output is $\hat{y} = \underline{\underline{C}}\hat{a}$. Using these definitions, it can be shown that the error between the true and estimated state obeys the following differential equation:

$$\dot{\underline{e}} = \dot{\underline{a}} - \dot{\hat{\underline{a}}} = (\underline{\underline{\tilde{A}}} - \underline{\underline{L}}\underline{\underline{C}})(\underline{a} - \hat{\underline{a}}) = (\underline{\underline{\tilde{A}}} - \underline{\underline{L}}\underline{\underline{C}})\underline{e}$$

We want to design the observer gain such that the error decays to zero, which means $(\underline{\underline{\tilde{A}}} - \underline{\underline{L}}\underline{\underline{C}})$ has purely real, negative eigenvalues. This is a pole placement problem analogous to the state feedback pole placement problem we have already explored. If we take the transpose of this matrix defining the evolution of the state estimation error, the unknown observer gain post-multiplies a known matrix, thus the problem has the same form as state feedback control. We will choose the desired eigenvalues for the system defining the error, and design $\underline{\underline{L}}$ such that $\underline{\underline{\tilde{A}}}^T - \underline{\underline{C}}^T\underline{\underline{L}}^T$ has the desired eigenvalues. This relies on the fact that transposing a matrix does not change its eigenvalues. As we have already shown, it is possible to place $\tilde{N}$ distinct eigenvalues arbitrarily for a problem of this form when the pair $(\underline{\underline{\tilde{A}}}^T, \underline{\underline{C}}^T)$ is reachable. If this is the case, all observer modes can be chosen to take on any value. As it turns out, the test of reachability for this pair is the same as the observability test we already performed on the system. Thus, all the observer modes can be placed anywhere we want.

Luenberger observer

Having established that the observer modes can be placed anywhere, we can now design the observer to accomplish this. We use MATLAB's place command (instead of hardcoding it, this time) with $\underline{\underline{\tilde{A}}}^T$, $\underline{\underline{C}}^T$ and a given set of eigenvalues. For state feedback control, we chose state feedback eigenvalues of the form

$$\lambda_i = -1/i \quad i = 1, 2, \ldots, \tilde{N}$$

Using the place command, we now determine the observer $\underline{\underline{L}}$ such that the observer eigenvalues have the form

$$\lambda_i = -c/i \quad i = -1, 2, \ldots, \tilde{N}$$

for $c = 1/5$, $c = 1$, and $c = 5$. See Figures 22-24 for the three observer gain matrices. We verify in the code that the eigenvalues of the matrix $\underline{\underline{\tilde{A}}} - \underline{\underline{L}}\underline{\underline{C}}$ agree with the input to the place command.

```
L1 =

    0.0015    0.0026    0.0015    0.0029    0.0069
    0.1760   -0.1997   -0.1173    0.2581   -0.0574
    0.5884   -0.8053    0.5866   -0.8018    0.2400
   -0.2340   -0.1583    0.2221    0.1385    0.0161
   -0.3199    0.4550   -0.3306    0.4763    0.0056
    1.3895    2.4629   -1.2181   -2.3341   -0.1432
   -0.8263    0.5672   -0.8246    0.5637    0.2364
   -2.2329    1.4995    2.5892   -1.1137   -0.3656
   -0.4466   -0.4584   -0.4468   -0.4616   -0.9558
```

Figure 22: Luenberger observer gain to obtain eigenvalues with $c = 1/5$.

```
L2 =

   -0.0017   -0.0030   -0.0032   -0.0016   -0.0047
    0.1573   -0.2684   -0.2196    0.1992   -0.0337
    0.6226   -0.8030    0.6079   -0.7876    0.1911
   -0.2157   -0.2193    0.1809    0.1573   -0.0094
   -0.2090    0.3258   -0.3242    0.4454   -0.0508
    1.4204    1.8450   -1.8385   -2.4548   -0.1523
   -0.8225    0.6059   -0.8051    0.5885    0.2079
   -2.4175    1.3015    2.1317   -1.5361   -0.1140
   -0.3696   -0.3528   -0.3273   -0.3682   -0.6186
```

Figure 23: Luenberger observer gain to obtain eigenvalues with $c = 1$.

```
L3 =

    0.0544    0.1044    0.1241    0.0542    0.1511
    0.1386   -0.0655   -0.2617    0.0182    0.0121
    0.7372   -0.7938    0.7337   -0.7437    0.0136
   -0.0652   -0.4468   -0.0250    0.3917   -0.0719
    0.1612   -0.2936    0.4033   -0.1863   -0.2834
    2.7402    0.4500   -2.9956   -0.5203   -0.4506
   -0.7228    0.7274   -0.7532    0.7607    0.0169
   -2.1859    2.1159    1.3155   -2.4011   -0.4276
   -0.1113   -0.0525   -0.0372   -0.1048   -0.1754
```

Figure 24: Luenberger observer gain to obtain eigenvalues with $c = 5$.

State feedback with observer

In the presence of the Luenberger observer, the state space system is

$$\dot{\underline{a}} = \underline{\underline{\tilde{A}}}\,\underline{a} + \underline{\underline{\tilde{B}}}\,\underline{u}$$

$$\underline{\underline{y}} = \underline{\underline{C}}\underline{a}$$

$$\dot{\underline{\hat{a}}} = \underline{\underline{\tilde{A}}}\underline{\hat{a}} + \underline{\underline{\tilde{B}}}\underline{u} + \underline{\underline{L}}(\underline{y} - \underline{\hat{y}})$$

$$\underline{\hat{y}} = \underline{\underline{C}}\underline{\hat{a}}$$

$$\underline{u} = -\underline{\underline{K}}\underline{\hat{a}} + \underline{\underline{F}}\underline{r}$$

State feedback control is now done on the estimated state $\underline{\hat{a}}$, as opposed to the true state. The Luenberger observer is designed so that the estimated state converges to the true state, as shown in the previous section. The dynamics of the true state and error can be written in a single system as

$$\begin{bmatrix} \dot{\underline{a}} \\ \dot{\underline{e}} \end{bmatrix} = \begin{bmatrix} \underline{\underline{\tilde{A}}} - \underline{\underline{\tilde{B}}}\underline{\underline{K}} & \underline{\underline{\tilde{B}}}\underline{\underline{K}} \\ \underline{\underline{0}} & \underline{\underline{\tilde{A}}} - \underline{\underline{L}}\underline{\underline{C}} \end{bmatrix} \begin{bmatrix} \underline{a}(t) \\ \underline{e}(t) \end{bmatrix} + \begin{bmatrix} \underline{\underline{\tilde{B}}}\underline{\underline{F}} \\ \underline{\underline{0}} \end{bmatrix} \underline{r}(t)$$

The separation principle ensures that, even though $\underline{\underline{K}}$ and $\underline{\underline{L}}$ were designed independent of each other, the closed loop system with Luenberger state estimation will have the eigenvalues coming from each of these independent problems. In other words, there is no downside to carrying out these two design problems in isolation.

We can now experiment with the three different Luenberger observer matrices we have designed. We will input a unit step function at each of the 5 control locations. Both the estimated state and true state will have zero initial conditions. This means that the initial condition on the error is $\underline{e}(0) = \underline{0}$. Notice from the above system that the error is unforced, meaning that its dynamics are entirely driven by the initial condition. This means that with zero initial condition, we expect the error to be zero for all time. This is even when the state dynamics are nontrivial, as there is no explicit coupling between the state $\underline{a}(t)$ and the error. We can confirm this by time integrating the dynamics for each of the three observers, and for step inputs at each of the control locations. As we expect, all error dynamics are trivial, as shown in Figure 25. To be clear, we are using the same $\underline{\underline{K}}$ and $\underline{\underline{F}}$ gain matrices we designed for the closed system with no observer. The input gain matrix $\underline{\underline{F}}$ is designed to track a unit step input. The step inputs should still be accurately tracked in the Luenberger system. We can verify that this is the case. See Figure 26 for the step input response for the Luenberger observer with $c = 5$. Finally, we can plot the state dynamics of the system for each of the step inputs. Previously, we displayed the concentration vs. time at a point in the 2D spatial domain of the advection-diffusion system. For the sake of diversity, we will now show the dynamics of the state $\underline{a}(t)$, which defines the Fourier coefficients used to construct the spatial distribution of concentration. See Figure 27.

Simulating the system

We now simulate the response of the system with the Luenberger observer to nonzero initial conditions but zero forcing. For given initial conditions on the error and state, the governing equation is

$$\begin{bmatrix} \dot{\underline{a}} \\ \dot{\underline{e}} \end{bmatrix} = \begin{bmatrix} \underline{\underline{\tilde{A}}} - \underline{\underline{\tilde{B}}}\underline{\underline{K}} & \underline{\underline{\tilde{B}}}\underline{\underline{K}} \\ \underline{\underline{0}} & \underline{\underline{\tilde{A}}} - \underline{\underline{L}}\underline{\underline{C}} \end{bmatrix} \begin{bmatrix} \underline{a}(t) \\ \underline{e}(t) \end{bmatrix}$$

We will assume zero initial conditions on the estimated state $\underline{\hat{a}}$, and take initial conditions on the true state to be the orthonormal set of basis vectors $\underline{\hat{b}}_i$ for $i = 1, 2, \ldots, \tilde{N}$. For each of the three Luenberger observers we designed, we will simulate the response of system for each of these initial conditions. These basis vectors were discussed earlier in the report. Thus, the initial condition on the error is

$$\underline{e}_i(0) = \underline{\hat{b}}_i - \underline{\hat{a}}(0) = \underline{\hat{b}}_i$$

The unforced closed loop system with the each of three observers and the given set of initial conditions is time integrated in MATLAB. See Figures 29-31 for results. The slow observer, where $\underline{\underline{L}}$ is computed to place the observer eigenvalues at $\lambda_i = -1/5i$, does not converge on an accurate estimate of the true state in the given simulation time. The simulation time $T = 50$ is rather a long for an advection-diffusion system with the given diffusivity. This can be justified by noting that even the original system matrix $\underline{\underline{\tilde{A}}}$ has a slowest eigenvalue of $\lambda = -0.2$. Within a time 20, an eigenvector initial condition corresponding to this eigenvalue will decay to less than 2% of its initial value. The closed loop system has a much faster response time than this. Thus, for the observer to not obtain an accurate estimate of the state in this simulation time is problematic. Because the control law is defined on the estimated state rather than the true state, this means we will not be able to accurately control the system. The next fastest observer, with eigenvalues computed using $c = 1$, converges to an accurate estimate of the state within the simulation time. However, for some components of the error, much of the duration of the simulation is spent with an inaccurate estimate of the state. Again, because the control

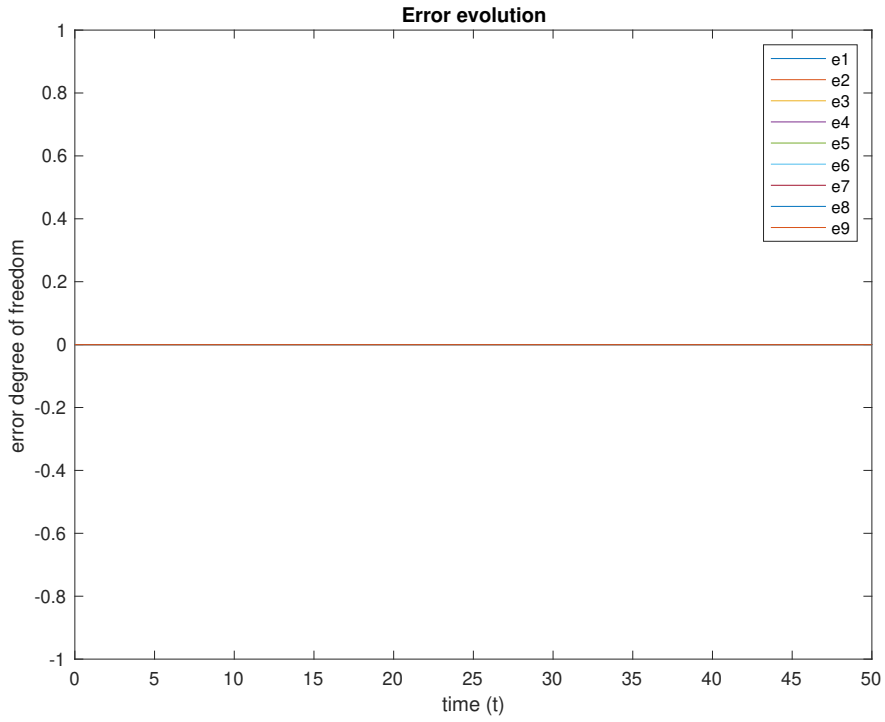

Figure 25: Regardless of the Luenberger observer or the location of the step input, all degrees of freedom which define the spatial discretization of the error are zero for all time. This is expected given that the dynamics of the error are driven entirely by the initial condition, and the initial error is zero.

input is defined using the estimated state, this means the state feedback rule on the system will be suboptimal (in the sense of not performing as it is design to). Finally, the observer with $c = 5$ converges on an accurate estimate of the system state in a fraction of the simulation time, which means that the state feedback quickly performs as designed. The first two "slow" observers would be objectionable in the presence of step inputs given that the state feedback law is designed to stabilize the system. This stabilization is what allows the step input to be accurately and rapidly tracked. If the estimate of the state is inaccurate, the state feedback law will not perform as originally designed, and the system may not track the step input in the given simulation time.

Linear quadratic regulator

We now want to use the linear quadratic regular to design a new state feedback gain $\underline{\underline{K}}_{lqr}$. We will use the built-in MATLAB LQR commands to find a minimum of the following objective:

$$J = \int_0^\infty \underline{a}^T \underline{a} + r\underline{u}^T \underline{u} dt$$

where $\underline{a}$ and $\underline{u}$ are such that the governing equation for the system dynamics are satisfied and r is a parameter that controls how much the control energy is penalized. Large values of r correspond to "expensive" control, meaning the LQR solution will favor lower energy control laws. In all following analyses, we will use the Luenberger observer corresponding to $c = 5$, as this leads to the most accurate control laws given the quick convergence of the state estimate.

We want to choose the trade-off parameter r such that the eigenvalues of the state feedback system with the LQR gain are as fast or faster than the pole placement method we used previously. Of course, the eigenvalues must be negative in order for the system to be stable. The slowest eigenvalue from the pole placement method is $\lambda_9 = -1/9$. From some numerical experimentation, choosing $r = 1$ gives rise to stable eigenvalues faster than the closed loop system. See Figure 32 for the closed loop eigenvalues computed with the LQR. See the Figure 33 for the entries of the new feedback gain given by the matrix $\underline{\underline{K}}_{lqr}$, and Figure 35 for verification that initial states in the real modal space decay at the rate predicted by the eigenvalues.

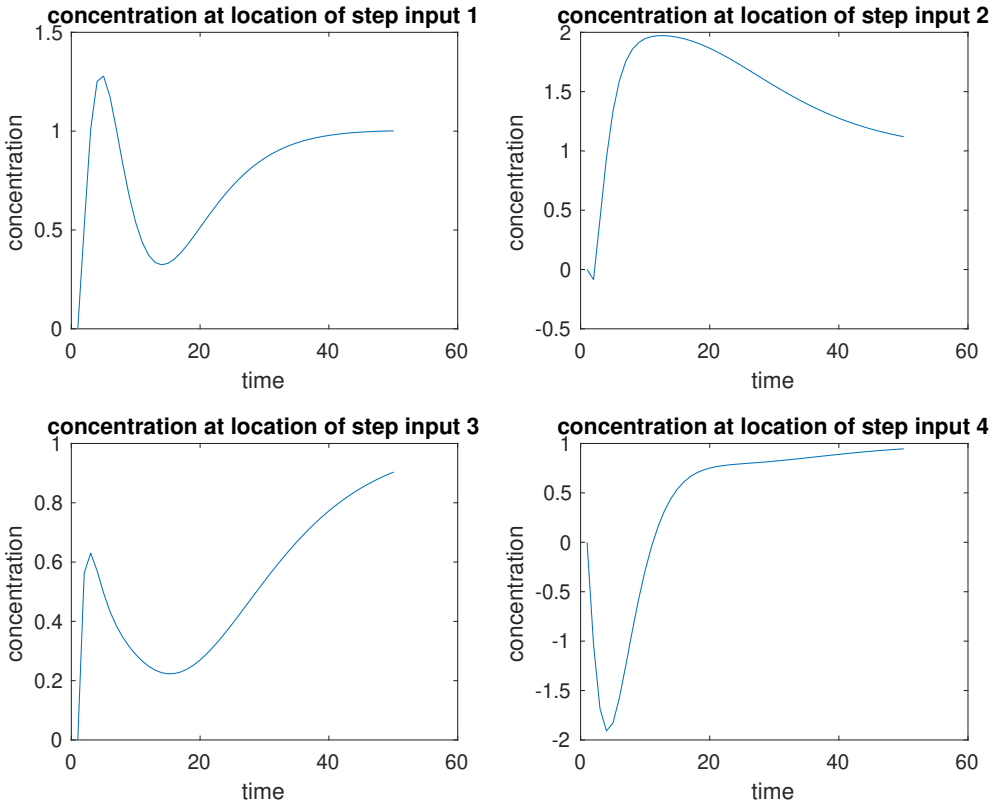

Figure 26: We can verify that the input gain matrix $\underline{\underline{F}}$ still forces the system to track a unit step input, even when the system state is estimated. It is clear from these plots that the system tends toward a concentration of 1. Four of the five control input locations are shown for convenience of plotting.

Because we have changed the state feedback gain, we need to recompute the input gain matrix in order for the system to accurately track step inputs. This is accomplished by using the same formula as before:

$$\underline{\underline{F}}_{lqr} = \left[\underline{\underline{C}}\left(-\underline{\underline{\tilde{A}}} + \underline{\underline{\tilde{B}}}\,\underline{\underline{K}}_{lqr}\right)^{-1}\underline{\underline{\tilde{B}}}\right]^{-1}$$

We can verify that this input gain leads to accurate tracking of step inputs. We test this by time integrating the system with the Luenberger observer for zero initial conditions (on the error and state) with step inputs applied at each of the five control locations. We see that the system quickly settles to a concentration of 1 at the location of the unit step input, as desired. See Figure 36.

Note that time constants for the system are the reciprocal of the real part of the eigenvalues. In the pole placement system, the smallest eigenvalue had a magnitude $1/9$, which corresponds to a time constant of $\tau = 9$. We will compute the energy associated with the control input for both the pole placement state feedback gain and the LQR state feedback gain over a time period $T = 3\tau$. These two quantities are computed as

$$E^1_{[0,27]}(i) = \int_0^{27} \left(-\underline{\underline{K}}\,\underline{\hat{a}} + \underline{\underline{F}}\,\underline{r}\right)^T \left(-\underline{\underline{K}}\,\underline{\hat{a}} + \underline{\underline{F}}\,\underline{r}_i\right) dt$$

$$E^2_{[0,27]}(i) = \int_0^{27} \left(-\underline{\underline{K}}_{lqr}\,\underline{\hat{a}} + \underline{\underline{F}}_{lqr}\,\underline{r}\right)^T \left(-\underline{\underline{K}}_{lqr}\,\underline{\hat{a}} + \underline{\underline{F}}_{lqr}\,\underline{r}_i\right) dt$$

where the i index corresponds to step inputs at each of the 5 control locations and $\underline{\hat{a}}$ is the estimated state. Note that we must compute the estimated state with

$$\underline{\hat{a}} = \underline{a} - \underline{e}$$

When we time integrate the system we only have access to the true state $\underline{a}$ and the error $\underline{e}$, but the control law is defined in terms of the estimated state. In our case, we expect the error to be zero because it begins with zero initial condition. E^1 is the energy associated with the control law from the pole placement method,

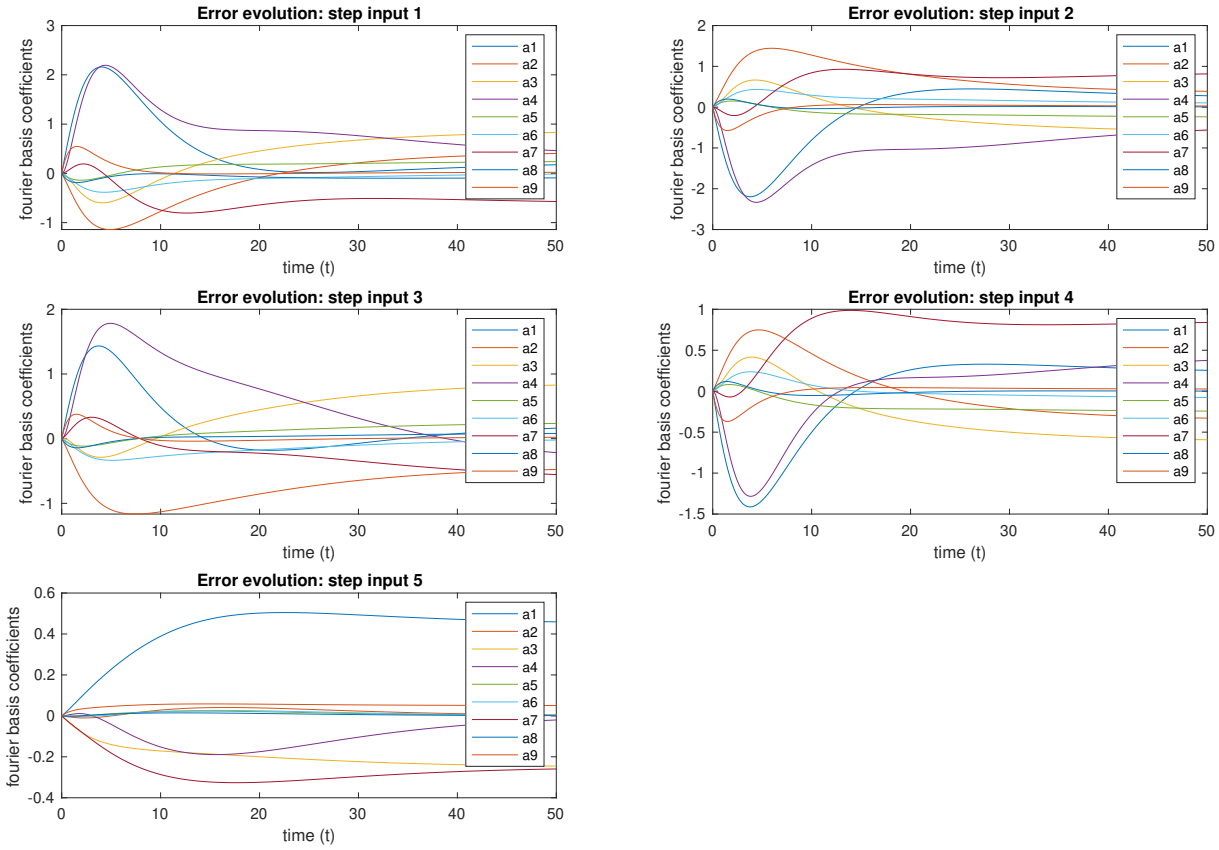

Figure 27: There are five control inputs in the advection-diffusion system, and we show the time evolution of the coefficients on the Fourier basis for concentration under the action of a step input at each of the control locations. The system starts with zero initial conditions, and evolves toward some steady state where the concentration is 1 at the location of the unit step control input.

and E^2 is the energy from LQR. When we time integrate the system, we simply compute the estimated state at each time step, use this to form the squared magnitude of the control input, then numerically integrate this quantity after the simulation finishes. See Figures 37 and 38 for results.

Lastly, we can compare the control inputs coming from the pole placement and LQR methods. To do this, we plot each of the 5 components of the control input for each of the 5 step inputs. Doing this for both methods will allow us to make a thorough comparison between the control inputs. We note that the two methods give very different results. The most obvious difference is that LQR gives rise to much faster convergence. We saw that this was due to the fact the magnitude of the real part of the closed loop eigenvalues is larger. The system quickly settles on a steady state, which means that the control input becomes constant. The system with placed poles takes longer to settle on a steady state. Similar to the case of the energies, we see that the first four control inputs coming from LQR are equivalent, due to the symmetry of the problem. See Figures 39 and 40.

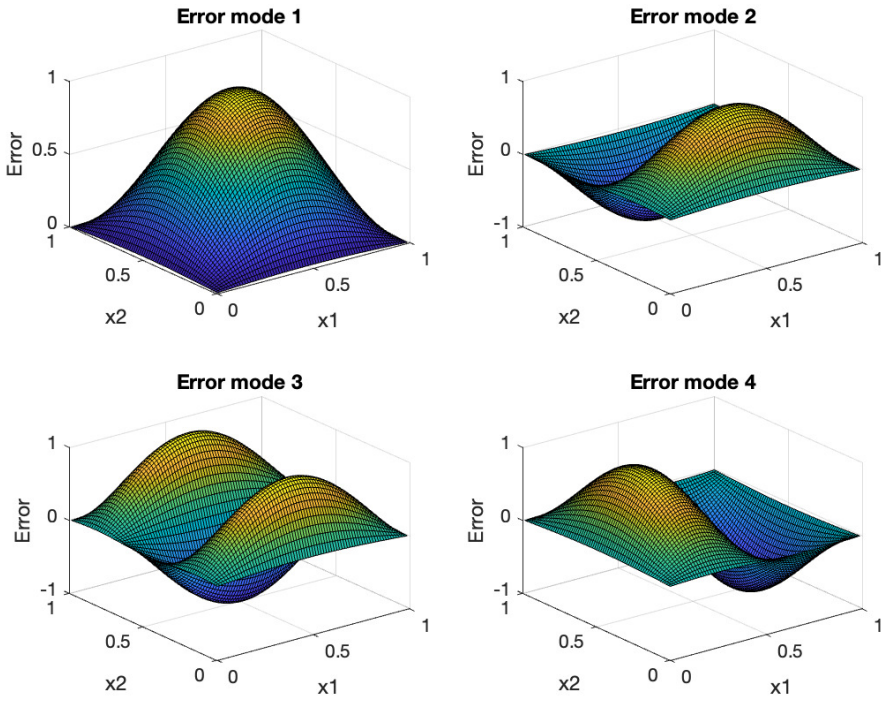

Figure 28: Showing what the Fourier coefficients on the error (and concentration) correspond to in terms of spatial distribution of concentration. Four of the nine basis functions (modes) are shown.

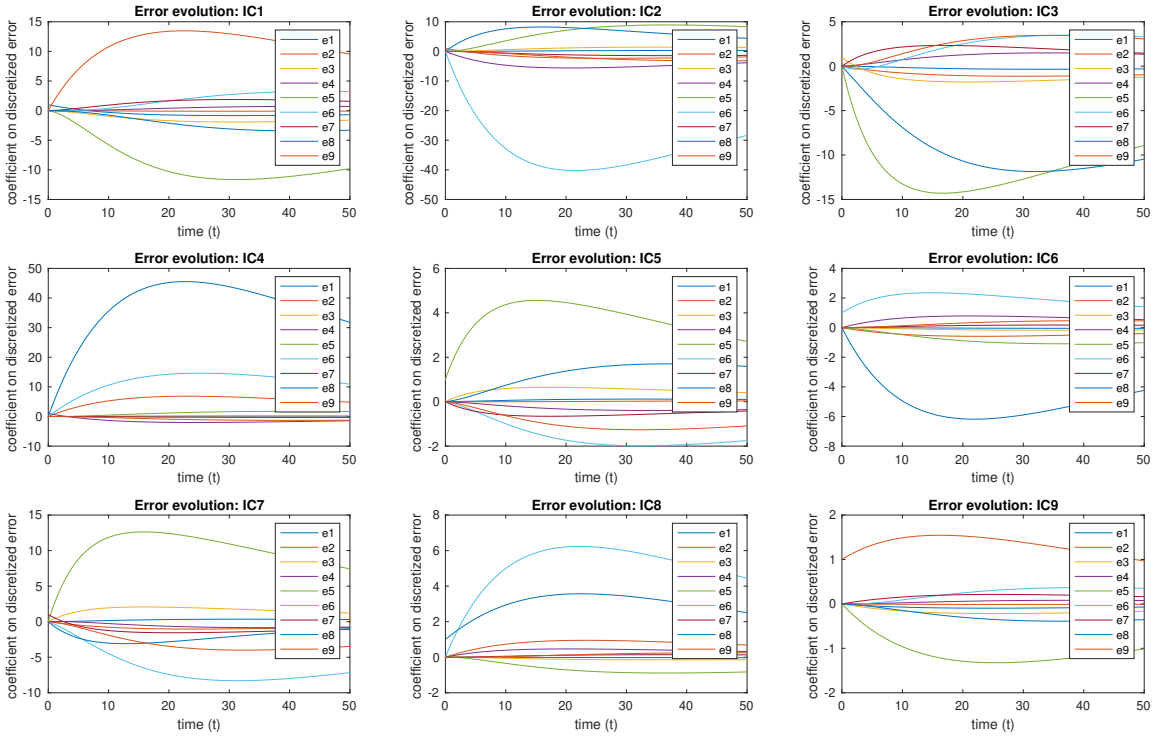

Figure 29: Dynamics of the error Fourier coefficients for each of the unit vector initial conditions for the Luenberger observer with $c = 1/5$. Relative to the simulation time of $T = 50$, the error decays to zero very slowly. This makes sense, given that we chose the observer modes to be slow relative to the closed loop system's eigenvalues. That the system takes so long to obtain an accurate estimate of the state is a problem if we want the state feedback control to behave as it is designed to in the given simulation time. Note that we cannot compare the time constants of this decay in error to the observer eigenvalues because the initial conditions are not eigenvectors.

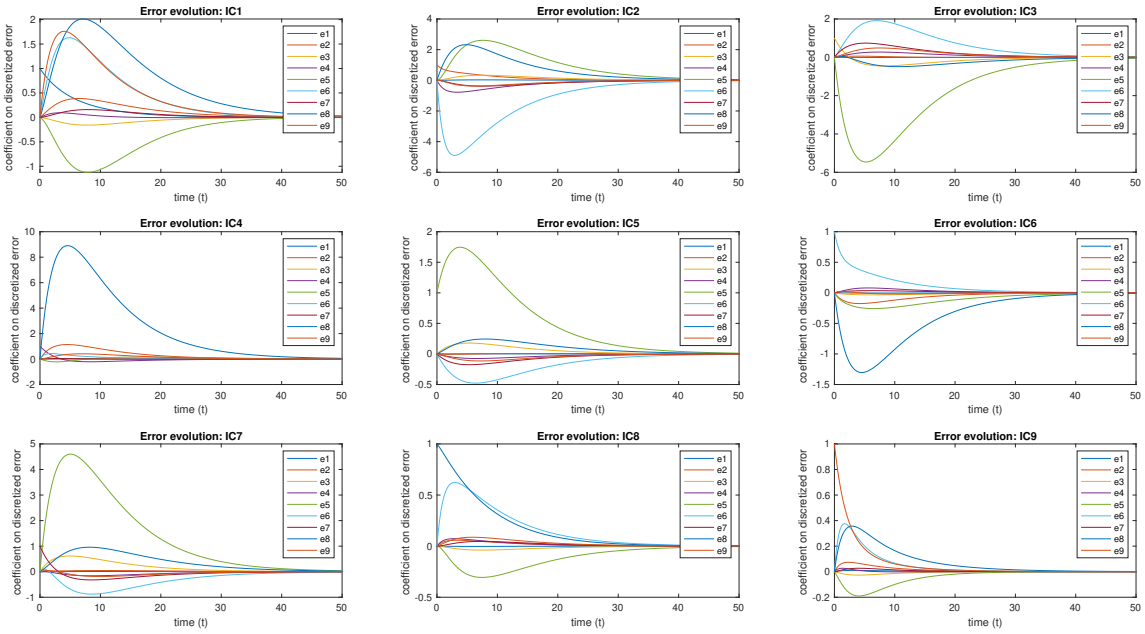

Figure 30: Dynamics of the error Fourier coefficients for each of the unit vector initial conditions for the Luenberger observer with $c = 1$. Relative to the simulation time of $T = 50$, the error decays to zero at a moderate pace. This makes sense, given that we chose the observer modes to be the same as the closed loop system's eigenvalues.

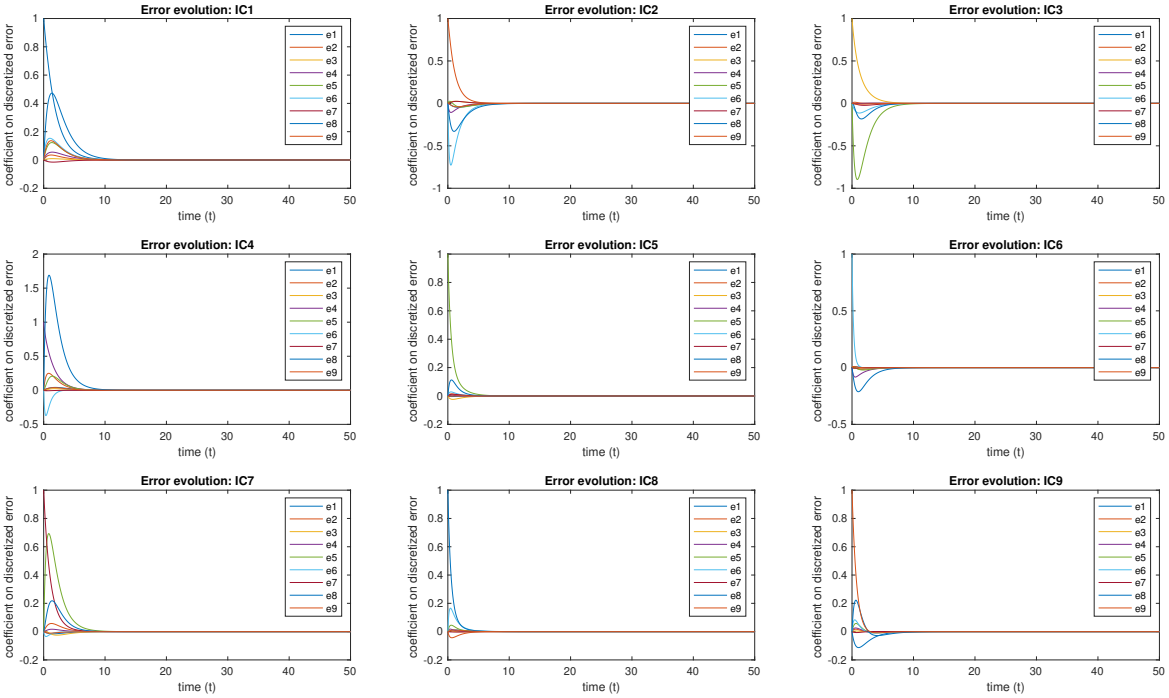

Figure 31: Dynamics of the error Fourier coefficients for each of the unit vector initial conditions for the Luenberger observer with $c = 5$. Relative to the simulation time of $T = 50$, the error decays to zero very quickly. This makes sense, given that we chose the observer modes to be fast compared to the closed loop system's eigenvalues. Because an accurate estimate of the state is quickly obtained, the state feedback law should stabilize the system as it is designed to.

```
P =

  -1.2488 + 0.0000i
  -4.7415 + 5.1681i
  -4.7415 - 5.1681i
  -5.2009 + 3.3566i
  -5.2009 - 3.3566i
  -5.5762 + 0.4549i
  -5.5762 - 0.4549i
  -8.0607 + 0.0000i
  -8.1016 + 0.0000i
```

Figure 32: Eigenvalues computed with the LQR state feedback gain using $r = 1$. These eigenvalues are no longer purely real, meaning the system oscillates, but the damping is so large that the system quickly damps out any oscillations.

```
Klqr =

    0.3289    0.5484    0.3497    0.2685    0.5391    0.4865    0.0924    0.2066    0.1321
    0.3289    0.2685    0.0924   -0.5484   -0.5391   -0.2066    0.3497    0.4865    0.1321
    0.3289   -0.5484    0.3497   -0.2685    0.5391   -0.4865    0.0924   -0.2066    0.1321
    0.3289   -0.2685    0.0924    0.5484   -0.5391    0.2066    0.3497   -0.4865    0.1321
    0.6579    0.0000   -0.4421    0.0000   -0.0000   -0.0000   -0.4421   -0.0000    0.2642
```

Figure 33: State feedback gain matrix computed using the LQR objective and trade off parameter $r = 1$.

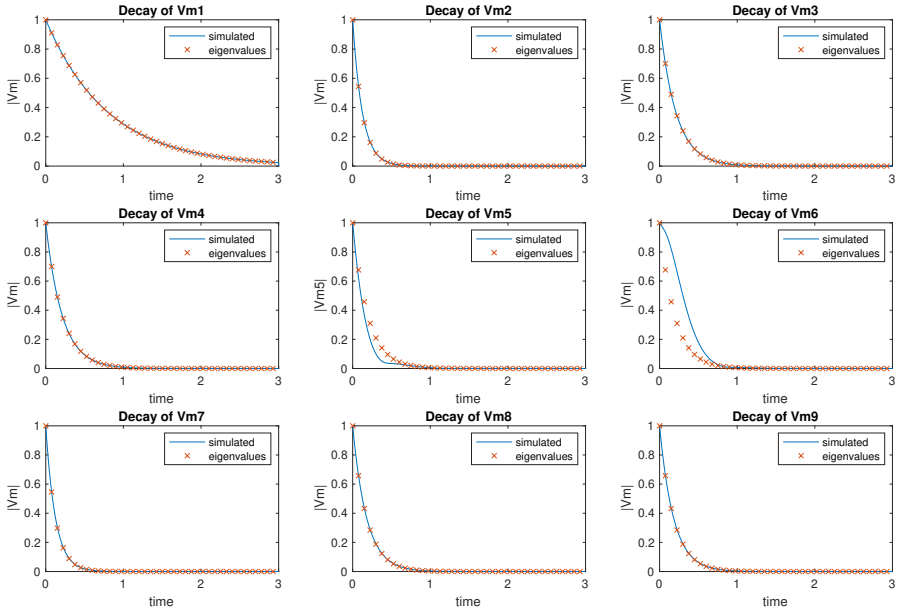

Figure 34: Verifying that initializing the time integration of the system with unit vectors from the real modal space leads to decay rates predicted by the eigenvalues for the LQR system. Note that eigenvalues are not purely real, but the decay rate is so fast the oscillations are invisible in all cases except for two. The real part of the eigenvalue dominates the response of the LQR system.

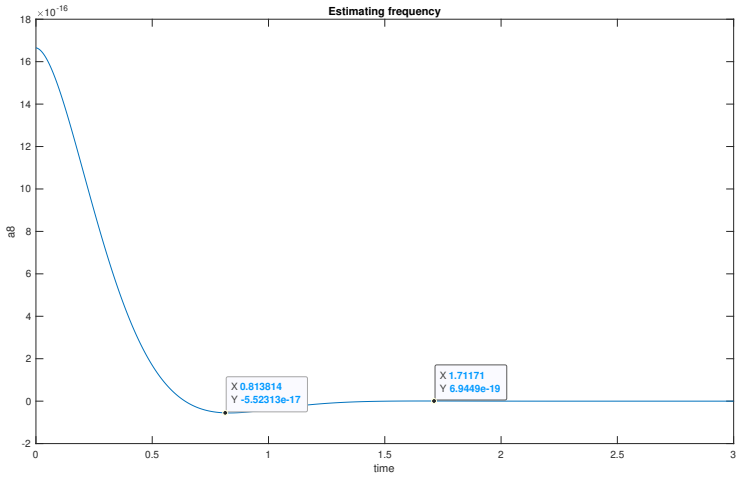

Figure 35: We can estimate the frequency of oscillation from the decay of an initial state taken from the real modal space corresponding to a complex eigenvalue. We use the response of one of the state coefficients to verify that the rate of oscillation agrees with the complex part of the eigenvalue. The spacing between a minimum and maximum in the signal is estimated to be $1.711 - 0.814 = 0.897$. This would be the half-period of a sine wave, so the predicted period is $T = 1.794$. The estimate of the complex part of the eigenvalue is then $2\pi/1.794 = 3.50$, whereas the true complex part of the eigenvalue is 3.36. This is just a sanity check–the discrepancy is due to error in picking out the minimum and maximum of the signal!

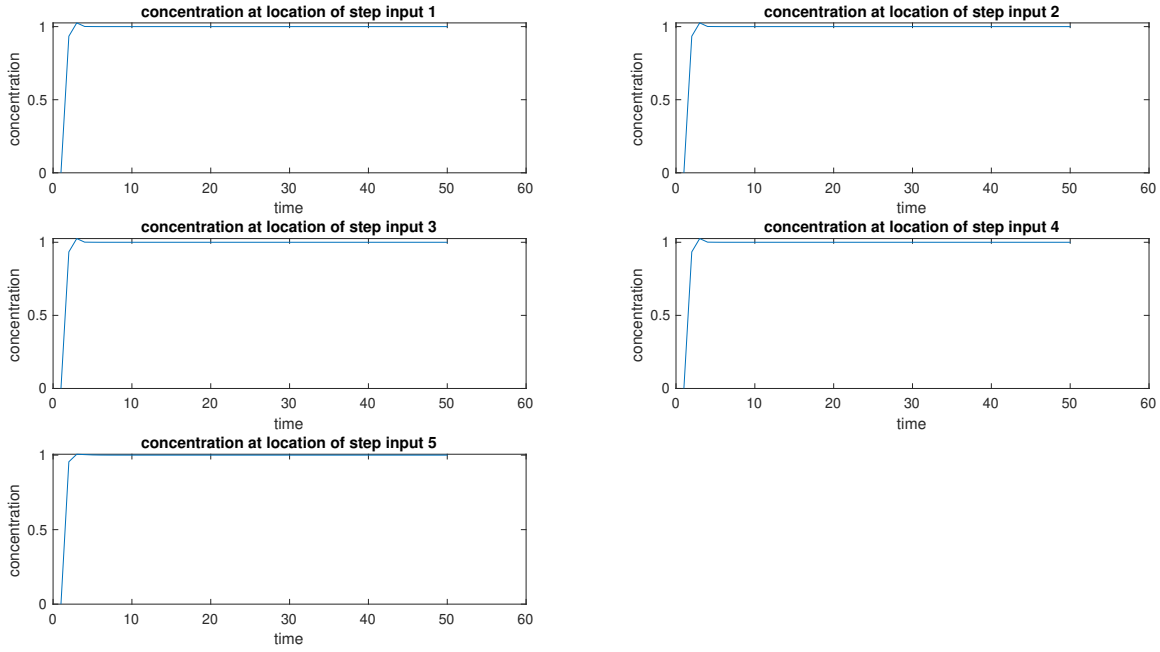

Figure 36: The concentration of the system at the location of each unit step input quickly rises from zero initial conditions, and after a brief overshoot, converges to a concentration $c = 1$. The convergence is rapid because the eigenvalues coming from the closed loop system with LQR are larger in magnitude than the pole placement method. This corresponds to faster settling to steady state.

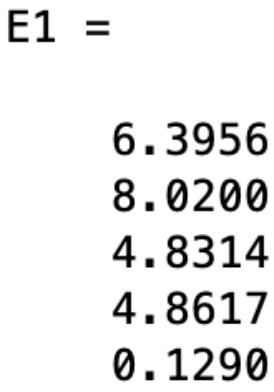

Figure 37: Energy associated with step inputs at each of the five control locations for the pole placement method. Note that though the original problem has rotational symmetry (there is nothing to distinguish the first four control inputs, which are at the corners of the square domain), each energy value is different. As discussed earlier in the project, in order to dial in the chosen distribution of eigenvalues, the pole placement method generates eigenvectors (corresponding to spatial modes of the concentration) that break the symmetry of the problem. Thus, we expect there to be different energies associated with actuation at the different control locations for the pole placement method.

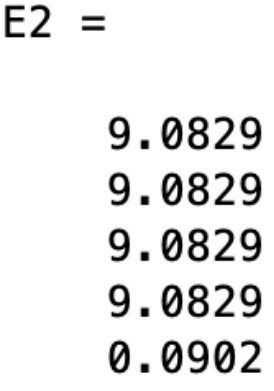

Figure 38: Unlike the pole placement method, the first four energies, corresponding to the control locations at the corners of the square domain, all have the same energy associated with them. The LQR method is based on an optimality condition, and it makes sense that there would no advantage from the standpoint of minimizing the control energy to breaking the symmetry of the problem. We note that the energy values are not the same between the two methods, but are comparable in size. The energies coming from LQR are larger because the system has a faster response, which requires more control effort to obtain.

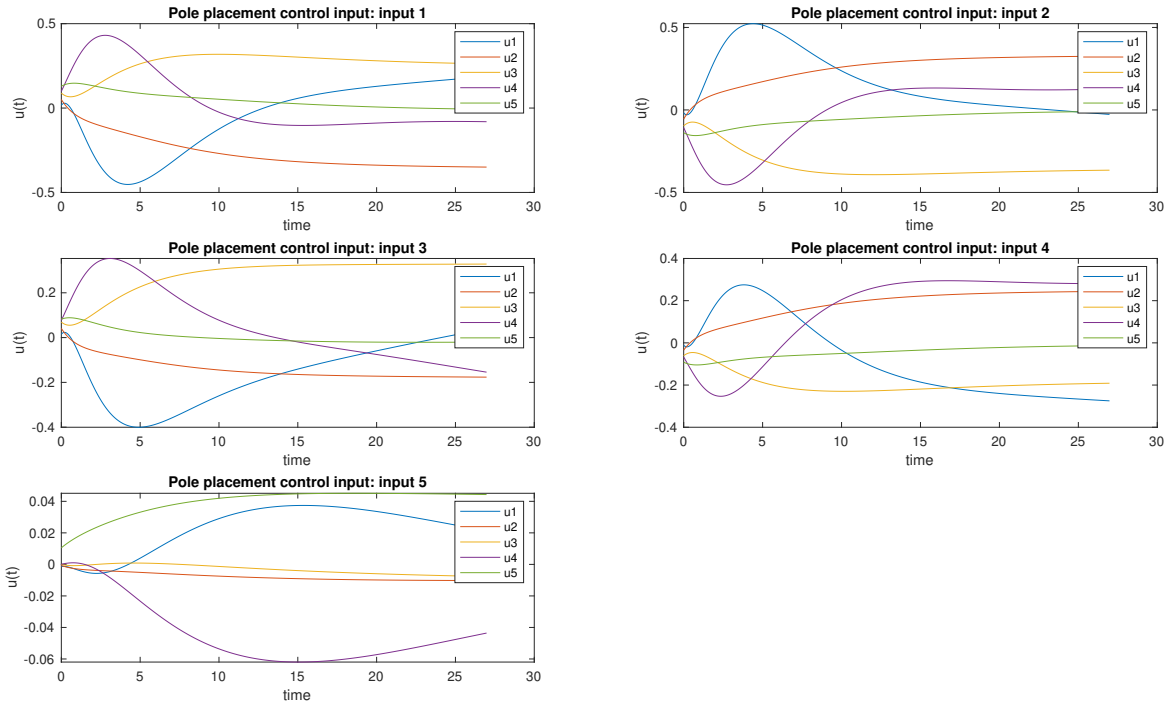

Figure 39: Control input from the pole placement method for step inputs at each of the 5 control locations. The control inputs do not fully settle to constant values in the allotted time frame. This is a consequence of choosing the simulation time of 3 time constants for this system. For a simple decaying exponential $e^{-t/\tau}$, a simulation time $t = 3\tau$ corresponds to decay to 5% of the initial state. This means that we do not expect the system to be fully settled on the steady state.

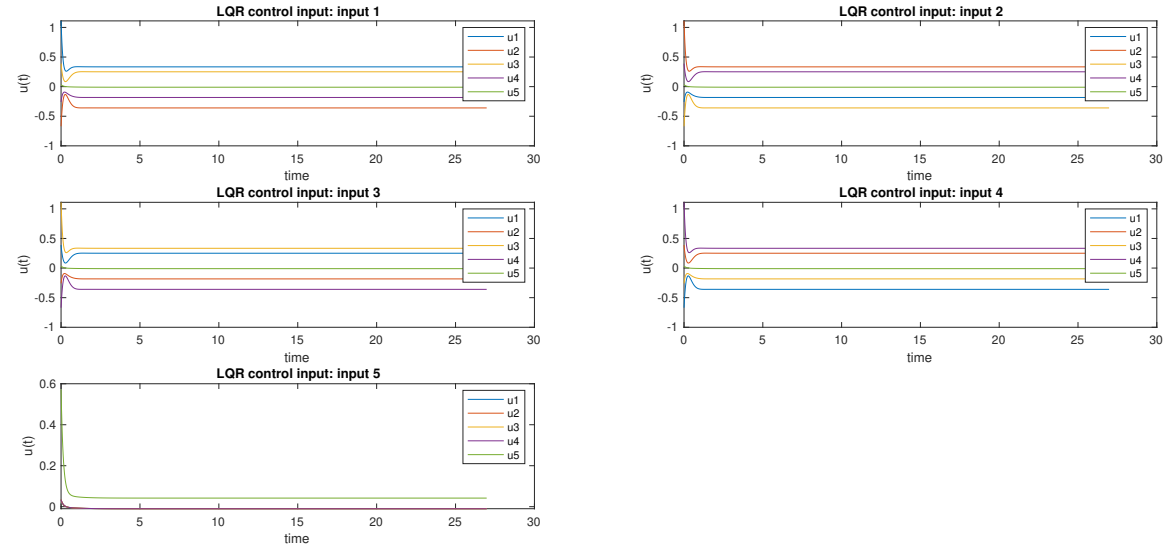

Figure 40: Control inputs from LQR for step inputs at each of the 5 control locations. As we have discussed, the system settles to steady state very quickly, at which point the control inputs become constant. Due to LQR not breaking the symmetry of the problem, the first four control inputs have the same form.