Phase Portraits

Consider the homogeneous system

\[ \dot{x} = Ax \]

A phase portrait is a graphical depiction of the solutions to this equation, starting from a variety of initial conditions. By sketching a few such solutions ("trajectories"), the general behavior of a system can be easily understood.

Phase portraits can be constructed qualitatively, from knowledge of the eigenvalues and eigenvectors, and are often used for nonlinear system analysis as well.

Some examples:
\[ x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \]

\[ \dot{x} = Ax = \begin{bmatrix} -1 & 0 \\ 0 & -4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \]

\[ \lambda = \{-1, -4\} \]

\[ M = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \]

\{ \text{Eigen-vectors} \}

\{ \text{Show how you go from} \]

\[ x(0) \to 0 \]

Two invariant subspace you can ride \( e_1 \) or \( e_2 \) line to the origin
Two Invariant Subspaces

\[ A = \begin{bmatrix} -2 & -1 \\ -2 & -3 \end{bmatrix} \]

\[ \lambda = \{-1, -4\} \] \{ e-vals \}

\[ M = \begin{bmatrix} 1 & 1 \\ -1 & 2 \end{bmatrix} \] \{ e-vectors \}

Stable Node

x₁

x₂

e₁

e₂
A saddle point is shown in the graph with the following matrix:

\[ A = \begin{bmatrix} -2 & 0 \\ 0 & 1 \end{bmatrix} \]

The eigenvalues are given by \[ \lambda = \{-2, 1\} \]

And the eigenvectors are given by \[ M = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \]

The graph illustrates two invariant subspaces: one stable and one unstable.
A = \begin{bmatrix} -3 & -2 \\ 2 & 2 \end{bmatrix}

\lambda = \{-2, 1\}

M = \begin{bmatrix} -2 & 1 \\ 1 & -2 \end{bmatrix}

Saddle Point

stable invariant subspace

unstable invariant subspace
\[ A = \begin{bmatrix} -1 & 1 \\ -1 & -3 \end{bmatrix} \]

\[ \lambda = \{-2, -2\} \]

One eigenvector: \[ \begin{bmatrix} -1 \\ 1 \end{bmatrix} \]

One Invariant Subspace
A = \begin{bmatrix} -2 & 1 \\ 0 & -2 \end{bmatrix} \quad \text{(Jordan Form of} \begin{bmatrix} -1 & 1 \\ -1 & -3 \end{bmatrix})

\lambda = \{-2, -2\}

\text{one eigenvector:} \begin{bmatrix} 1 \\ 0 \end{bmatrix}

One Invariant Subspace Rotates Space
Stable Focus

evals
$\lambda = \{-1 + j3, -1 - j3\}$

evectors
$M = \begin{bmatrix} -1 + j1 & -1 - j1 \\ j2 & -j2 \end{bmatrix}$

geometric interpretation of invariant subspace dissolves

Inward arrows stable
Spirals denote oscillation from imaginary part
\[ A = \begin{bmatrix} -1 & 2 \\ -5 & 1 \end{bmatrix} \]

evals
\[ \lambda = \{3j, -3j\} \]

evectors
\[ M = \begin{bmatrix} 3+j & 3-j \\ j5 & -j5 \end{bmatrix} \]

oscillations for all time
**Time-varying case:** (Things get hairy)

To simplify some computations, consider the simpler homogeneous system:

\[ \dot{x}(t) = A(t)x(t) \]

(For uniqueness, we ask that the elements of \( A(t) \) be continuous functions of time). Remember that the matrix exponential is no longer an integrating factor, so we must look for a different one:

It is known that the set of solutions of an \( n \)th order linear homogeneous differential equation (or a system of \( n \) first order equations) forms an \( n \)-dimensional vector space.
A basis of $n$ such solutions can be chosen in a number of different ways, such as choosing a basis of $n$ linearly independent initial condition vectors and using the resulting solutions. To make things easy, choose:

$$x_1(t_0) = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad x_2(t_0) = \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}, \quad \ldots \quad x_n(t_0) = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}$$

When we stack the resulting solutions together side-by-side, we get the **fundamental solution matrix**:

$$X(t) = \begin{bmatrix} x_1(t) & x_2(t) & \ldots & x_n(t) \end{bmatrix}$$
Obviously, \( \dot{X}(t) = A(t)X(t) \)  \( \text{Since } \dot{x} = Ax \) \{ \text{vector system} \}

And an expansion of the solution of the state vector \( x(t) \) into this basis will be

\[
x(t) = X(t)x(t_0)
\]

So if we know the solution of the system to \( n \) linearly independent initial conditions, we know it for any by computing \( X(t) \).

Now we notice from the identity

\[
\frac{dX^{-1}(t)}{dt} = -X^{-1}(t)\frac{dX(t)}{dt}X^{-1}(t) \quad \{ \text{Matrix Identity} \}
\]
\[
\frac{dX^{-1}(t)}{dt} = -X^{-1}(t) \frac{dX(t)}{dt} X^{-1}(t) \quad \text{same Identity}
\]

\[
\frac{dX^{-1}(t)}{dt} = -X^{-1}(t) A(t) X(t) X^{-1}(t) \quad \text{substitute } \dot{X} = AX
\]

\[
= -X^{-1}(t) A(t) \quad \text{new identity}
\]

So \( X^{-1}(t) \) qualifies as a valid integrating factor for the state equations:

\[
\dot{X} = A(t) X(t) + B(t) u(t)
\]

\[
X^{-1}(t)[\dot{x}(t) - A(t)x(t) = B(t)u(t)]
\]

\[
X^{-1}(t)\dot{x}(t) - X^{-1}(t)A(t)x(t) = X^{-1}(t)B(t)u(t)
\]

\[
X^{-1}(t)\dot{x}(t) + \frac{dX^{-1}(t)}{dt} x(t) = X^{-1}(t)B(t)u(t)
\]  

substitute new identity
Solving

\[ X^{-1}(t) \dot{x}(t) + \frac{dX^{-1}(t)}{dt} x(t) = X^{-1}(t)B(t)u(t) \] \{ \text{same product rule} \}

\[ \frac{d}{dt} \left[ X^{-1}(t)x(t) \right] = X^{-1}(t)B(t)u(t) \]

\[ X^{-1}(t)x(t) - X^{-1}(t_0)x(t_0) = \int_{t_0}^{t} X^{-1}(\tau)B(\tau)u(\tau)d\tau \] \{ \text{integrate} \}

or premultiply by \( X(t) \) and simplify

\[ x(t) = X(t)X^{-1}(t_0)x(t_0) + \int_{t_0}^{t} X(t)X^{-1}(\tau)B(\tau)u(\tau)d\tau \]

This would be great if we knew \( X(t) \) all the time, but unfortunately, it is difficult to compute.

\[ X(t) \] is the solution to \( \dot{X} = A(t)X \)
State Transition Matrix: Define the State Transition Matrix as:

\[ \Phi(t, \tau) = X(t)X^{-1}(\tau) \]

This is an \( nxn \) linear transformation from the state-space into itself. For \textit{homogeneous} systems, it relates the state vectors at any two times:

\[ x(t) = \Phi(t, \tau)x(\tau) \]

\( u = 0 \)

From prev. page \( X(t) = X(t)X^{-1}(t_0)X(t_0)^{t_0=\tau} \Rightarrow x(t) = \Phi(t, \tau)X(\tau) \)

(Verify this using \( x(t) = X(t)x(t_0) \)).

By differentiating it, one can show that:
\[
\frac{d\Phi(t, \tau)}{dt} = \frac{d}{dt} X(t) X^{-1}(\tau) = \frac{dX(t)}{dt} X^{-1}(\tau)
\]

\[
= A(t) X(t) X^{-1}(\tau)
\]

\[
\frac{d\Phi(t, \tau)}{dt} = A(t) \Phi(t, \tau)
\]

(Chen uses this last line as the definition of \( \Phi(t, \tau) \) in his book.) Using our definition,

\[
\Phi(t, \tau) = X(t) X^{-1}(\tau)
\]

it should be obvious that

\[
\Phi(t_2, t_0) = \Phi(t_2, t_1) \Phi(t_1, t_0) = X(t_2) X^{-1}(t_1) (X(t_1)) X(t_0)
\]

and

\[
\Phi^{-1}(t, t_0) = \Phi(t_0, t) \Rightarrow \left[X(t) X^{-1}(t_0) \right]^{-1} = X(t_0) X^{-1}(t)
\]
If our system is time-invariant, then it is easy to verify that
\[ \Phi(t, \tau) = e^{A(t-\tau)} \]
by substitution into the definition.

When this is the case, we can compute \( \Phi(t, \tau) = e^{A(t-\tau)} \) in many ways:

1. Because
\[ X(s) = (sI - A)^{-1} BU(s) + (sI - A)^{-1} x(t_0) \]
and
\[ x(t) = e^{A(t-t_0)} x(t_0) + \int_{t_0}^{t} e^{A(t-\tau)} Bu(\tau) d\tau \]
we can compare terms and get:
\[ e^{A(t-t_0)} = L^{-1} \left\{ (sI - A)^{-1} \right\}_{t-t_0} = \Phi(t, t_0) \]

Note that \( \Phi(t, \tau) = \Phi(t - \tau, 0) \) whenever \( A \) is a constant matrix.

2. Use the Cayley-Hamilton theorem to express \( e^{At} \) as:

\[ \Phi(t, \tau) = e^{A(t-\tau)} = \alpha_0 I + \alpha_1 A + \cdots + \alpha_{n-1} A^{n-1} \]

and find the coefficients from the system of equations found by substituting eigenvalues of \( A \) in the scalar polynomial:

\[ e^{\lambda_i(t-\tau)} = \alpha_0 + \alpha_1 \lambda_i + \cdots + \alpha_{n-1} \lambda_i^{n-1} \]  

\{ \text{ith eigenvalue} \}
3. First simplify the system by putting it in diagonal form (or Jordan form). Then

\[
\Phi(t, \tau) = Me^{J(t-\tau)}M^{-1}
\]

4. "Sylvester's Expansion" (explained in Brogan)

5. Taylor series expansion:

\[
\Phi(t, \tau) = I + A(t - \tau) + \frac{1}{2!} A^2 (t - \tau)^2 + \cdots
\]
However when $A=A(t)$, **none** of the choices are good:

1. Computer simulation of $\dot{\Phi}(t, \tau) = A(t)\Phi(t, \tau)$ with $\Phi(\tau, \tau) = I$

2. Define $B(t, \tau) = \int_{\tau}^{t} A(\zeta) d\zeta$. Then if $AB = BA$,

$$\Phi(t, \tau) = e^{B(t, \tau)}$$

3. Integral expansions (Peano-Baker series):

$$\Phi(t, t_0) = I_n = \int_{t_0}^{t} A(\tau_0) d\tau_0 + \int_{t_0}^{t} A(\tau_0) \int_{t_0}^{\tau_0} A(\tau_1) d\tau_1 d\tau_0 + \cdots$$

4. Approximate with discrete-time systems.
An Introduction to Discrete-Time Systems:

Consider the continuous-time linear system

\[\dot{x}(t) = A(t)x(t) + B(t)u(t), \quad x(t_0) = x_0\]

and suppose it is sampled every \(T\) seconds to give a discrete-time system. Assume that this sampling speed is much faster than the rate at which \(u(t)\) changes, we therefore consider it to be constant over any individual sampling period \(T = t_{k+1} - t_k\), i.e., \(u(t) \approx u(t_k)\), for \(t_k \leq t \leq t_{k+1}\).

Consider time \(t_k\) to be an initial condition and use the \(t_{k+1}\) state-transition matrix to find the state vector at time \(t_{k+1}\):

\[x(t_{k+1}) = \Phi(t_{k+1}, t_k)x(t_k) + \int_{t_k}^{t_{k+1}} \Phi(t_{k+1}, \tau)B(\tau)d\tau u(t_k)\]
This can be further simplified if the system has a constant $A$-matrix, so that:

$$\Phi(t, \tau) = e^{A(t-\tau)}$$

\[
x(t_{k+1}) = e^{A(t_{k+1}-t_k)} x(t_k) + \int_{t_k}^{t_{k+1}} e^{A(t_{k+1}-\tau)} B(\tau) d\tau u(t_k)
\]

\[
x(t_{k+1}) = e^{AT} x(t_k) + \int_{t_k}^{t_{k+1}} e^{A(t_{k+1}-\tau)} B(\tau) d\tau u(t_k)
\]

This is the discrete-time approximation to the continuous-time system. If we want the state-transition matrix from a discrete-time system, we can use induction:
(We will give it a new name, $\Psi(k, j)$):

Recall the recursions we obtained in the example that introduced the concept of controllability of a discrete-time system:

$$x(k + 1) = Ax(k) + Bu(k)$$

$$x(1) = Ax(0) + Bu(0)$$

$$x(2) = A^2 x(0) + ABu(0) + Bu(1)$$

$$x(3) = A^3 x(0) + A^2 Bu(0) + ABu(1) + Bu(2)$$

$$\vdots$$

$$\vdots$$

$$x(k) = A^k x(0) + A^{k-1} Bu(0) + A^{k-2} Bu(1) + \ldots + Bu(k-1)$$
Or

\[ x(k) = A^k x(0) + \sum_{j=0}^{k-1} A^{k-j-1} B(j)u(j) \]

or

\[ x(k) = A^k x(0) + \sum_{j=1}^{k} A^{k-j} B(j-1)u(j-1) \] \hspace{1cm} \text{change the index}

Leading to:

\[ \Psi(k, j) = A^{k-j} \]

It may be apparent that in state-variables, discrete-time systems are considerably easier to analyze than continuous-time systems.
If $A = A_d(k)$, that is, a discrete-time, time-varying system, then

$$\Psi(k, j) = \prod_{i=j}^{k-1} A_d(i)$$

Computation of eigenvalues, eigenvectors, and canonical forms for discrete-time systems is exactly the same as for continuous-time systems. The interpretation of eigenvalues in the context of stability properties will be different, but modal decompositions and diagonalization procedures are exactly the same:
If $M$ is the modal matrix, we will get a diagonalized (or perhaps Jordan) form:

\[
\begin{align*}
q(k + 1) &= M^{-1}AMq(k) + M^{-1}Bu(k) \\
y(k) &= CMq(k) + Du(k)
\end{align*}
\]

and

\[
x(k) = Mq(k)
\]