Massimiliano de Sa
Haeyoon Han

Lectures on Linear Systems Theory

Draft: January 2025

Caltech Control & Dynamical Systems

Preface

These lecture notes on linear systems theory are designed for a roughly 10 week course on the subject. The aim is to provide a balanced overview of both the state space and I/O perspectives on linear systems, with an eye towards precise mathematical formulations of problems in control. The approach we follow is inspired by the Fall 2023 offering of CDS 131, Linear Systems Theory, by John Doyle.

Texts that have notably influenced our presentation of the material include *Linear System Theory* by Callier and Desoer, *Mathematical Control Theory* by Sontag, and *Feedback Control Theory* by Doyle, Francis, and Tannenbaum. The lecture notes on linear systems theory by John Lygeros and those by Richard Murray have also proven to be invaluable resources, as has as Stephen Boyd's course EE 363, Linear Dynamical Systems.

Each section of the text corresponds to a roughly 1.5 hour lecture. Subsections that can be skipped without loss of continuity are marked with a ★. To demarcate the difficulty of problems, we also use a star system; ★ means challenging and ★★ means very challenging, compared to the average unstarred problem.

Although the only formal prerequisites for this course are a strong background in linear algebra, prior exposure to control systems and real analysis is certainly useful. Typically, PhD students in control taking this course have either taken or are concurrently taking courses in convex optimization and linear functional analysis, and have had a first (undergraduate) course in control. Prior knowledge of these subjects is not, however, required in order to succeed in learning the course material. We provide a cursory review of the essentials in Chapter 1.

Special thanks go to Minwoo Kim for close reading!

Contents

Chapter 1
Mathematical Preliminaries

As its name suggests, control *theory* is a fundamentally mathematical subject. At the start of a course in linear systems theory, one is often expected to have mastered topics from linear algebra, real analysis, and functional analysis. Given the modern day engineering curriculum, it's not altogether realistic to expect students to know all of these right from the get-go.

In this brief chapter, we introduce the fundamental mathematical concepts needed to study mathematical systems theory. We stress - you *don't* need to master everything in this chapter upon the first read! If it's your first time seeing this material, try to get a basic feel for the definitions in your first read - you can always come back and refresh your knowledge as the course progresses.

Math is a contact sport, and learning to deal with mathematical abstraction is something that only comes with practice. As such, we leave many of the easier results of this chapter as exercises - you're encouraged to do them as you go along to build your understanding. With this said, let's begin!

1.1 Vector Spaces

The fundamental setting for linear system theory is the *vector space*. In this section, we'll get to grips with the basic definitions and properties of vector spaces. Notably, we'll construct very general definitions of vector spaces which don't assume finite-dimensionality.

First, let's construct a formal definition of a vector space. From your first course in linear algebra, you might recall working in vector spaces such as $\mathbb{R}^n$ or $\mathbb{C}^n$, in which *vectors* are tuples of real or complex numbers. You might have defined *vector addition* as the operation which takes two tuples of numbers and adds each pair of entries to form a new vector, or *scalar multiplication*, which multiplies each element of the vector by a scalar to form a new vector. The following definition of a vector space takes the fundamental properties of vectors, addition, and scalar multiplication that are familiar to us from $\mathbb{R}^n$ and $\mathbb{C}^n$, and abstracts away the "tuples of numbers" into an abstract vector.

Definition 1.1 (Vector Space) Let $\mathbb{K} = \mathbb{R}$ or $\mathbb{C}$. A vector space V over $\mathbb{K}$ is a set, V, together with two operations, $+ : V \times V \to V$ and $(\cdot) : \mathbb{K} \times V \to V$, satisfying:

1. Closure under operations: For all $u, v \in V$ and all $\alpha, \beta \in \mathbb{K}$, $\alpha u + \beta v \in V$.

2. Associativity: For all $u, v, w \in V$, $u + (v + w) = (u + v) + w$.
3. Commutativity: For all $u, v \in V$, $u + v = v + u$.
4. Additive Identity: There exists an element $0 \in V$ for which $0 + v = v$, for all $v \in V$.
5. Additive Inverse: For all $v \in V$, there exists an element $-v \in V$ for which $v + (-v) = 0$.
6. Compatibility: For all $\alpha, \beta \in \mathbb{K}$ and $v \in V$, $\alpha \cdot (\beta \cdot v) = (\alpha\beta) \cdot v$.
7. Multiplicative Identity: For all $v \in V$, $1 \cdot v = v$.
8. Distributivity in V: For all $\alpha \in \mathbb{K}$ and $u, v \in V$, $\alpha \cdot (u + v) = \alpha u + \alpha v$.
9. Distributivity in $\mathbb{K}$: For all $\alpha, \beta \in \mathbb{K}$ and $v \in V$, $(\alpha + \beta) \cdot v = \alpha v + \beta v$.

V is called the *set of vectors* and $\mathbb{K}$ is called the *field*. Elements of the set V are called *vectors*, while elements of the set $\mathbb{K}$ are called *scalars*.

Remark 1.1 Above, we used $(\cdot)$ to denote the multiplication of a vector by a scalar. One typically suppresses the $(\cdot)$, and writes $c \cdot v = cv$, for $c \in \mathbb{K}$ and $v \in V$.

Remark 1.2 Formally, we denote a vector space V over $\mathbb{K}$ by the pair $(V, \mathbb{K})$. However, if $\mathbb{K}$ is clear from context, one refers to a vector space simply by the set V.

Let's take a moment to appreciate what's going on underneath all of the abstraction. First, let's take a moment to outline the structure of an abstract definition, for those who might be unfamiliar with definition-theorem-proof mathematics. Typically, when writing an abstract definition, one starts by specifying out a couple of objects - here, we specify a set, V, a set $\mathbb{K}$, and the operations $+$ and $(\cdot)$. Then, we specify some *axioms* - properties that the objects must have. By laying out each definition in this manner, we can ensure there are no ambiguities in the foundations of our theory.

With this in mind, let's think about what Definition 1.1 is actually saying. All that Definition 1.1 *really* says is that a vector space is any set V along with a set of scalars $\mathbb{K}$, equipped with two operations $+$ and $(\cdot)$ that act like $+$ and $(\cdot)$ on $\mathbb{R}^n$ with scalars in $\mathbb{R}$. Each condition of the definition simply specifies a property that we have between tuples of numbers in $\mathbb{R}^n$, so that our *abstract* vector space behaves just like $\mathbb{R}^n$ might. The definition is summarized as follows:

1. V is a set of *vectors*, analogous to tuples of real numbers in $\mathbb{R}^n$.
2. $\mathbb{K}$ is a set of *scalars* called the field, analogous to scalar real numbers in $\mathbb{R}$.
3. The operations $+$ and $(\cdot)$ are defined to behave just like $+$ and $(\cdot)$ between real number scalars and tuples of real numbers in $\mathbb{R}^n$.

Let's consider a few examples to make things more concrete.

Example 1.1 Consider the vector space $(\mathbb{R}^n, \mathbb{R})$ (read as $\mathbb{R}^n$ over $\mathbb{R}$). Here, $V = \mathbb{R}^n$, the set of tuples of n real numbers, the set of scalars is $\mathbb{K} = \mathbb{R}$, and the operations $+$ and $(\cdot)$ are defined,

$$(x_1, ..., x_n) + (y_1, ..., y_n) = (x_1 + y_1, ..., x_n + y_n) \tag{1.1}$$
$$c \cdot (x_1, ..., x_n) = (c \cdot x_1, ..., c \cdot x_n), \tag{1.2}$$

where $c \in \mathbb{R}$ is any scalar in $\mathbb{R}$ and $(x_1, ..., x_n)$, $(y_1, ..., y_n)$ are tuples of n real numbers.

Example 1.2 Consider the vector space $(\mathbb{C}^{n\times n}, \mathbb{C})$ (read $\mathbb{C}^{n\times n}$ over $\mathbb{C}$), in which the set of vectors is $V = \mathbb{C}^{n\times n}$, the set of $n \times n$ matrices with complex entries, the set of scalars is $\mathbb{K} = \mathbb{C}$, and the operations $+$ and $(\cdot)$ are defined,

$$\begin{bmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \dots & a_{nn} \end{bmatrix} + \begin{bmatrix} b_{11} & \dots & b_{1n} \\ \vdots & \ddots & \vdots \\ b_{n1} & \dots & b_{nn} \end{bmatrix} = \begin{bmatrix} a_{11}+b_{11} & \dots & a_{1n}+b_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1}+b_{n1} & \dots & a_{nn}+b_{nn} \end{bmatrix} \tag{1.3}$$

$$c \cdot \begin{bmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \dots & a_{nn} \end{bmatrix} = \begin{bmatrix} c \cdot a_{11} & \dots & c \cdot a_{1n} \\ \vdots & \ddots & \vdots \\ c \cdot a_{n1} & \dots & c \cdot a_{nn} \end{bmatrix}. \tag{1.4}$$

Thus, we observe that the set of $n \times n$ complex matrices forms a vector space over $\mathbb{C}$.

So far, the two examples we've considered have been fairly routine. For the next example, we consider something a little bit more abstract.

Example 1.3 Let V be a set and W a vector space over $\mathbb{R}$. Consider the vector space $(\mathcal{F}_{V,W}, \mathbb{R})$, where the vectors are functions $f : V \to W$, and the operations $+, (\cdot)$ are defined by function addition and function scalar multiplication,

$$(f+g)(v) = f(v) + g(v), \ v \in V \tag{1.5}$$

$$(c \cdot f)(v) = c \cdot f(v), \qquad v \in V, \ c \in \mathbb{R}, \tag{1.6}$$

where in the right hand side of the expressions above, the operations $+, (\cdot)$ are those from the vector space W. Thus, we can generate *new* vector spaces from existing vector spaces! Note that we *don't require* V to be a vector space for this construction to work, only W.

This example is certainly more abstract than those we've considered so far, but follows the *same* underlying principles of a vector space. All we have is a set of vectors, a set of scalars, an addition rule, and a scalar multiplication rule. This example highlights a few important principles:

1. Vectors are more than tuples of numbers: Vectors in a vector space are *not* just tuples of numbers - they can take on far more abstract forms such as matrices and functions between vector spaces.
2. Abstraction is your friend: When thinking of vector spaces such as the space of functions above, thinking about the vectors as functions between spaces can get confusing! Instead of *looking inside* the set of vectors, use abstraction to your advantage! When performing any standard algebraic operations, you can *forget* about the complex internal structure, and abstract everything away behind the definition of a vector space. Similarly to computer science, where we hide away implementation details behind classes and methods, we hide away the "implementation" of individual vector spaces (e.g. vectors are *functions*) behind the nice abstraction of Definition 1.1. Using abstraction to your advantage is critical to managing complexity in mathematics.

Exercise 1.1 Verify that the examples presented above are indeed vector spaces by showing they satisfy all of the axioms of Definition 1.1.

Let's build upon the basic structure of a vector space we outlined above. One of the axioms of a vector space, *closure under operations*, states that for all $u, v \in V$ and $\alpha, \beta \in \mathbb{K}$, one must have $\alpha u + \beta v \in V$. It stands to reason that we might like to add or scale more than two vectors at a time! The following definition puts a name to a scaled sum of an arbitrary, finite collection of vectors.

Definition 1.2 (Linear Combination) Consider a vector space V over a field $\mathbb{K}$, and a finite collection $\{v_1, ..., v_k\} \subseteq V$ of vectors in V. A linear combination of the collection $\{v_1, ..., v_k\}$ is any vector of the form,

$$c_1 v_1 + ... + c_k v_k, \tag{1.7}$$

where $c_1, ..., c_k \in \mathbb{K}$ are any scalars in $\mathbb{K}$.

Remark 1.3 It's important to note that a linear combination of vectors is a scaled sum of a *finite* collection of vectors. We can take as many vectors as we want in a linear combination, as long as the number is not infinite.

There is a subtlety present in the definition of a linear combination that we've glossed over at first pass. The definition of a vector space states that a linear combination of *two* vectors will always belong to the vector space - how do we know that the linear combination of *any* finite linear combination of vectors will be in the vector space? Although this seems like a trivial detail, it's vital that we dot all of our i's and cross all of our t's before we begin using these concepts in earnest. Therefore, every time we make a new definition, we must ensure that the definition *well-posed* - that it is logically sound and doesn't lead to any contradictions in our theory. The following proposition confirms that the definition of a linear combination *is* in fact well-posed.

Proposition 1.1 (Vector Spaces are Closed Under Linear Combinations) *Consider a vector space V and a collection $\{v_1, ..., v_k\} \subseteq V$ of vectors in V. Any linear combination of $v_1, ..., v_k$ also belongs to V.*

Exercise 1.2 Prove Proposition 1.1. (Hint: use induction on k).

Oftentimes, we'll be interested in closely examining a special subset of a given vector space. If the subset of a given vector space *still* has the structure of a vector space, we call it a *subspace*. We make this idea explicit with the following definition.

Definition 1.3 (Subspace) Consider a vector space V over $\mathbb{K}$ with operations $+$ and $(\cdot)$. A subset $W \subseteq V$ is said to be a subspace of V if, under the operations $+$ and $(\cdot)$ of V, it is also a vector space over $\mathbb{K}$.

Taking a look at the definition of a subspace, it seems like it would be an awful lot of work to verify that a given subset of a vector space is in fact a subspace. Fortunately, there is a *much* easier way to verify a given set is a subspace than checking that all 10^n axioms of a vector space hold. Since a subspace is already a subset of a vector space, subspaces automatically inherit a number of the vector space properties. What remains to be checked is covered by the following proposition.

Proposition 1.2 *Consider a vector space V. A subset $W \subseteq V$ is a subspace of V if and only if, for all $\alpha, \beta \in \mathbb{K}$ and $u, v \in W$, $\alpha u + \beta v \in W$.*

In other words, a subset of a vector space is a subspace if and only if it is closed under linear combinations, with respect to the operations of the original vector space. All other properties of a vector space are inherited from the fact that a subspace is a subset of the vector space.

Exercise 1.3 Prove Proposition 1.2.

An important example of a subspace is the *span* of a collection of vectors, which we now define.

Definition 1.4 (Span) Consider a collection $\{v_1, ..., v_k\} \subseteq V$ of vectors. The span of $\{v_1, ..., v_k\}$, denoted $\text{span}\{v_1, ..., v_k\}$, is the set of all linear combinations of $v_1, ..., v_k$,

$$\text{span}\{v_1, ..., v_k\} = \{c_1 v_1 + ... + c_k v_k : c_i \in \mathbb{K}\} \subseteq V. \tag{1.8}$$

Exercise 1.4 Show that the span of a finite collection of vectors of a vector space V is a subspace of V.

Let's study some more basic properties of vector spaces and linear combinations. As you may recall, an important property that a collection of vectors can have is *linear independence*. Fundamentally, a collection of vectors is linearly independent if each vector "points in a new direction" compared to the other vectors in the collection. We formalize this intuitive notion of linear independence in the following definition.

Definition 1.5 (Linear Independence) Let V be a vector space over $\mathbb{K}$. Consider a collection $\{v_1, ..., v_k\} \subseteq V$ of vectors in V. The collection is said to be *linearly independent* if, for all $c_1, ..., c_k \in \mathbb{K}$ with $c_1, ..., c_k$ not all zero, $c_1 v_1 + ... + c_k v_k \neq 0$. If the collection is *not* linearly independent, it is said to be *linearly dependent*.

Let's verify that the formal definition of linear independence matches up with our intuitive notion of vectors "pointing in new directions." Suppose we're given a collection of linearly *dependent* vectors, $\{v_1, ..., v_k\}$. Then, by Definition 1.5, there exist constants $c_1, ..., c_k \in \mathbb{K}$, not all zero, for which

$$c_1 v_1 + c_2 v_2 + ... + c_k v_k = 0. \tag{1.9}$$

Without loss of generality, suppose that $c_1 \neq 0$ (the choice to focus on c_1 is arbitrary, as no c_i, v_i pair has any property making it more "special" than the others - we can therefore focus on c_1 *without loss of generality*). Then, by the above, we can write,

$$v_1 = -\frac{c_2}{c_1} v_2 - ... - \frac{c_k}{c_1} v_k. \tag{1.10}$$

Therefore, we conclude that v_1 "points in a direction" that is *already* covered by the remaining $k - 1$ vectors. Thus, it is *not* true that in a linearly dependent set, every vector "points in a new direction." We conclude that the formal definition of linear independence is consistent with the intuitive notion we outlined above.

Armed with the definition of linear independence, we now have the ability to study a variety of more sophisticated constructions. First, we introduce the definition of a basis for a vector space.

Definition 1.6 (Basis) Consider a vector space V over $\mathbb{K}$. A basis $\mathcal{B}$ for V is a linearly independent collection, $\mathcal{B} = \{v_1, ..., v_k\} \subseteq V$, such that for all $v \in V$, there exist scalars $c_1, ..., c_k \in \mathbb{K}$ for which

$$v = c_1 v_1 + ... + c_k v_k. \tag{1.11}$$

Any vector $v_i \in \mathcal{B}$ belonging to the basis is called a basis vector.

Exercise 1.5 Let $\mathcal{B} = \{v_1, ..., v_k\}$ be a basis for a vector space V. Show that for any $v \in V$, the constants $c_1, ..., c_k \in \mathbb{K}$ for which $v = c_1 v_1 + ... + c_k v_k$ are uniquely determined by v.

Thus, a collection of vectors is a *basis* for a vector space if the collection is linearly independent and we can write any element of the vector space as a (unique) linear combination of the basis vectors. Eagle-eyed readers will note that here, we've defined a basis to be a *finite* collection of vectors! We'll address our reasoning for this shortly.

In order to understand why we've defined a basis in terms of a finite collection of vectors, we first need to define the *dimension* of a vector space. Recall that, given a set S with a finite number of elements, $|S|$ denotes the number of elements in S. With this in mind, we make the following definition.

Definition 1.7 (Dimension) Consider a vector space V over $\mathbb{K}$. Suppose $\mathcal{B} \subseteq V$ is a basis for V. The dimension of V, $\dim(V)$, is the number of elements in the basis, $\dim(V) = |\mathcal{B}|$.

In order for this definition to be well-defined, it's important that we check that *all bases* for a given vector space have the same dimension! The following result confirms that this is in fact the case.

Proposition 1.3 (Dimension is Well-Defined) *Consider a vector space V over $\mathbb{K}$. If $\mathcal{B}_1$ and $\mathcal{B}_2$ are two bases for V, then $|\mathcal{B}_1| = |\mathcal{B}_2|$.*

Let's discuss why Proposition 1.3 implies dimension is a well-posed quantity. Given any two bases for V, Proposition 1.3 states that the bases must contain the same number of elements. Since dimension is defined as the number of elements in a basis, Proposition 1.3 confirms that dimension is a property of a vector space, rather than a property of a basis. Thus, it makes sense to write $\dim(V)$, rather than $\dim(\mathcal{B})$.

Exercise 1.6 Prove Proposition 1.3.

Before we move on to the study of linear transformations, we have one more point to discuss regarding bases! In your first course in linear algebra, you likely worked exclusively with *finite-dimensional* vector spaces - vector spaces in which there exists a finite basis. What, then, does it mean for a vector space to be infinite-dimensional?

Definition 1.8 (Finite/Infinite-dimensional Vector Space) A vector space V is said to be finite-dimensional if it has a basis with a finite number of elements. If no such basis exists, V is said to be infinite-dimensional.

Defining bases for infinite-dimensional spaces is a somewhat more subtle problem; one that is beyond the scope of our brief review of linear algebra. We refer the interested reader to the references at the end of the chapter for a treatment of infinite-dimensional bases.

Now that we've studied vector spaces in reasonable detail, we can discuss linear transformations, which are special maps between vector spaces. From your first course in linear algebra, you might immediately think of a matrix when you think of a linear transformation. If the vector space you're working in is $\mathbb{R}^n$, this is certainly justifiable! However, in abstract vector spaces, matrices are *not* immediately associated with linear transformations, analogous to how vectors are not immediately associated with tuples of numbers. Consider the following, abstract definition.

Definition 1.9 (Linear Transformation) Consider two vector spaces V, W over $\mathbb{K}$. A linear transformation between V and W is a map $A : V \to W$ such that for all $\alpha, \beta \in \mathbb{K}$ and $u, v \in V$, $A(\alpha u + \beta v) = \alpha A(u) + \beta A(v)$.

Remark 1.4 Note that, instead of writing $A(u)$ for the action of a linear transformation $A : V \to W$ on a vector $u \in V$, it is convention to write Au. This convention *does not* extend to nonlinear maps.

Thus, an "abstract" linear transformation is simply a map between vector spaces that *respects* the linear structure of the vector spaces. The study of linear transformations between vector spaces is one that is surprisingly deep. We'll return to linear transformations after a brief digression into *analysis* on the real line and in vector spaces.

1.2 Crumbs of Real Analysis

In systems and control theory, bounds are used to estimate complex quantities in terms of simple ones. For instance, in Chapter 2, when we study stability, we'll look for *exponential bounds* on the trajectories of our system. To effectively study mathematical control theory, it's therefore important that we have the tools we need to bound sets of real numbers, and reason about when these bounds are *sharp*.

This is where real analysis comes in. Fundamentally, elementary real analysis is the centered around the study of *convergence* of sequences and the *shapes* of sets of real numbers. In this section, we'll focus on this second point, and develop the necessary concepts required to find sharp bounds on sets of real numbers.

First, we'll introduce some basic language and facts about subsets of the real line, $\mathbb{R}$, and then proceed to develop sharp bounds - called suprema and infima - on these sets. Finally, we'll briefly look at how these sharp bounds on sets interact with functions. As a first step towards achieving these goals, we define what it means for a set to be *bounded*.

Definition 1.10 (Bounded Above/Below) Consider a subset $A \subseteq \mathbb{R}$. A is said to be:

1. Bounded above: if there exists an $U \in \mathbb{R}$ such that $x \leq M$, for all $x \in A$. In this case, such a U is said to be an upper bound on A.
2. Bounded below: if there exists an $L \in \mathbb{R}$ such that $L \leq x$, for all $x \in A$. In this case, such an L is said to be a lower bound on A.
3. Bounded: if there exists an $M \in \mathbb{R}$ such that $|x| \leq M$, for all $x \in A$.

The definitions presented above simply tell us whether or not a set is bounded. Notably, we do *not* specify how close or far our upper or lower bounds are from the set. For instance, if one takes the interval $[0, 1) \subseteq \mathbb{R}$, both 1 and 100 are equally valid upper bounds. However, the upper bound of 1 *clearly* provides us with more information about the set than the upper bound of 100. Now, we seek a bound on a set that provides us with the *most possible information* about a set - i.e. a bound that is as tight as possible. Consider the following definitions.

Definition 1.11 (Supremum) Consider a set $A \subseteq \mathbb{R}$. The supremum of A, denoted $\sup A$, is the least upper bound of A. That is, if $U \in \mathbb{R}$ is any upper bound on A, $\sup A \leq U$.

Just as we define the least upper bound, we also define the greatest lower bound.

Definition 1.12 (Infimum) Consider a set $A \subseteq \mathbb{R}$. The infimum of A, denoted $\inf A$, is the greatest lower bound of A. That is, if L is any lower bound on A, then $L \leq \inf A$.

The supremum and infimum are defined to be the *tightest possible* upper and lower bounds on a given subset of $\mathbb{R}$. Let's look at a couple of quick examples.

Example 1.4 Consider the set $A = [a, b) \subseteq \mathbb{R}$, where $a < b$. Here, $\sup A = b$, as b is the smallest possible upper bound on A. Likewise, $\inf A = a$, since a is the greatest possible lower bound on A.

This example highlights an important feature of suprema and infima - it is *not* necessarily the case that $\sup A \in A$ or $\inf A \in A$. In the example of $[a, b)$, the infimum belonged to the set, while the supremum did not. Let's take a second look at the definitions of the supremum and infimum to see why this is. When looking at the definitions of suprema and infima, one makes a natural comparison to the maximum and minimum of a set. Let's make a formal definition of a maximum and minimum to clear up the difference between a maximum/supremum and minimum/infimum.

Definition 1.13 (Maximum/Minimum) Consider a subset $A \subseteq \mathbb{R}$. A point $a \in A$ is said to be the maximum element of A if $x \leq a$ for all $x \in A$. Likewise, a point $b \in A$ is said to be the minimum element of A if $b \leq x$ for all $x \in A$.

Thus, we observe that unlike the supremum and infimum, the maximum or minimum of a set *must belong* to the set. As a consequence of this, given an arbitrary subset $A \subseteq \mathbb{R}$, one is *not* guaranteed to have a maximum or minimum value, even if it is bounded above and below! Consider, for example, the set $(a, b) \subseteq \mathbb{R}$. Here, the set is bounded above and below *but* has no maximum or minimum value, since a and b are not included. However, both the supremum and infimum exist; $\sup(a, b) = b$ and $\inf(a, b) = a$.

In the event where they exist, how do the maximum and minimum of a set relate to the supremum and infimum? The following result provides an answer.

Proposition 1.4 *Consider a set $A \subseteq \mathbb{R}$. If A has a maximum element, then* $\max A = \sup A$. *Likewise, if A has a minimum element, then* $\min A = \inf A$.

Thus, as our intuition might confirm, the maximum and minimum coincide with the supremum and infimum *when they exist*.

Exercise 1.7 Prove Proposition 1.4.

The question of existence of maxima and minima naturally begs the question - do the supremum and infimum of a set always exist? The following fact answers this question.

Fact (Axiom of the Supremum) A nonempty, bounded above set $A \subseteq \mathbb{R}$ has a finite supremum, $\sup A < \infty$. □

Notably, we state this result as a *fact*, not as a proposition! Why is this? As it happens, the axiom of the supremum is baked into the formal definition of the real numbers, $\mathbb{R}$.[1] As such, it is not something that we formally need to prove. In the case where A is *not* bounded above, we take $\sup A = \infty$ by convention. In the case where $A = \emptyset$, we take $\sup A = -\infty$ by convention.

This answers the question of existence of the supremum. What about existence of the infimum? In order to answer this question, we state a handy proposition which lets us translate results about the supremum to results about the infimum.

[1] This is a component of one of a number of equivalent, formal definitions for $\mathbb{R}$. The interested reader is encouraged to consult the references provided at the end of the chapter to learn more about the construction of the reals.

Proposition 1.5 *For a set $A \subseteq \mathbb{R}$, $\inf A = -\sup(-A)$, where $-A := \{-x : x \in A\}$.*

This result enables us to directly translate results about the supremum to results about the infimum, simply by flipping the sign of elements in the set. Generally, we'll prove results for the supremum and translate them to the infimum using Proposition 1.5.

Exercise 1.8 Prove Proposition 1.5.

By Proposition 1.5 and the Axiom of the Supremum, the following result is immediate.

Proposition 1.6 *Any nonempty, bounded below set $A \subseteq \mathbb{R}$ has a finite infimum.*

Using our conventions for the supremum along with Proposition 1.5, we have that $\inf A = -\infty$ when A is not bounded below and that $\inf A = \infty$ when $A = \emptyset$.

Thus far, we've only discussed the suprema and infima of generic subsets of $\mathbb{R}$. Now, we'll consider how suprema and infima interact with real-valued functions. Although this might initially seem like a step up from working with suprema and infima of sets, all we need to do to treat suprema and infima of functions is introduce a little bit of notation. Consider an arbitrary set A (not even necessarily a subset of $\mathbb{R}$), and a function $f : A \to \mathbb{R}$. We define,

$$\sup_{x \in A} f(x) := \sup\{f(x) : x \in A\} = \sup f(A) \tag{1.12}$$

$$\inf_{x \in A} f(x) := \inf\{f(x) : x \in A\} = \inf f(A). \tag{1.13}$$

Thus, taking the supremum and infimum of real-valued functions is really the same thing as taking the supremum and infimum of sets - we simply take the supremum and infimum of the *images* of sets, which are nothing more than standard subsets of $\mathbb{R}$. Now that we've discussed the foundational aspects of suprema and infima, we state a number of their basic properties.

Proposition 1.7 (Properties of Suprema) *Consider sets $A, B \subseteq \mathbb{R}$. The suprema of A and B satisfy the following properties.*

1. *Subset Inequality: If $B \subseteq A$, then $\sup B \leq \sup A$.*
2. *Sum of Sets Equality: If A, B are nonempty, then $\sup(A + B) = \sup A + \sup B$, where $A + B$ is defined, $A + B = \{a + b : a \in A, b \in B\}$.*
3. *Sum of Functions Inequality: For D an arbitrary set and $f, g : D \to \mathbb{R}$ functions,*

$$\sup_{x \in D}(f(x) + g(x)) \leq \sup_{x \in D} f(x) + \sup_{x \in D} g(x). \tag{1.14}$$

It's extremely important that we distinguish between properties (2) and (3) listed above. Although it might initially seem that the same rule would apply for functions and sets, this is not the case! The intuitive reasoning for this is as follows. For functions, taking the sum $\sup_{x \in D} f(x) + g(x)$ means taking the supremum of the sum where f and g are evaluated at the *same* point x. However, taking the sum $\sup_{x \in D} f(x) + \sup_{x \in D} g(x)$ means that f and g can take in different values! This yields an inequality, as both f and g are free to individually take on their suprema. Note that this reasoning takes a little bit of sharpening up to yield a formal proof - in particular, one can formally prove (3) by applying (1) and (2). We leave the details of this task as an exercise.

Exercise 1.9 Apply Proposition 1.5 to reformulate Proposition 1.7 in terms of infima.

Exercise 1.10 Supply a formal proof of item (3) of Proposition 1.7 using items (1) and (2).

1.3 Normed Vector Spaces

Now that we've sketched out some basic real analysis, we return to the domain of linear algebra and study basic analysis in vector spaces. First, we define a norm on a vector space.

Definition 1.14 (Norm) Consider a vector space V over $\mathbb{K}$. A norm is a map $\|\cdot\| : V \to \mathbb{R}_{\geq 0}$ satisfying the following conditions:

1. Positive Definite: $\|u\| \geq 0$ for all $u \in V$, and $\|u\| = 0$ if and only if $u = 0$.
2. Positive Homogeneity: For all $u \in V$ and $c \in \mathbb{K}$, $\|cu\| = |c| \, \|u\|$.
3. Triangle Inequality: For all $u, v \in V$, $\|u + v\| \leq \|u\| + \|v\|$.

Remark 1.5 Here, we use $|\cdot|$ to denote the magnitude of a scalar. For $\mathbb{K} = \mathbb{R}$, this is equal to the absolute value, and for $\mathbb{K} = \mathbb{C}$, this is equal to the complex magnitude.

Since $\|\cdot\| : V \to \mathbb{R}_{\geq 0}$ maps to the positive reals (which *do not* include ∞), it is a requirement that the norm of any given vector is finite! If $\|v\|$ is not finite for some $v \in V$, then $\|\cdot\|$ is *not* a valid norm on V. Let's consider a few common examples of norms on $\mathbb{R}^n$. In each of the following examples, convince yourself that each norm is finite for all vectors in the vector space on which they are defined.

Example 1.5 (ℓ^2 Norm) The ℓ^2 norm on $\mathbb{R}^n$, alternatively called the 2-norm or Euclidean norm, is defined

$$\|x\|_2 = \sqrt{x_1^2 + ... + x_n^2} = \sqrt{\sum_{i=1}^{n} x_i^2}. \tag{1.15}$$

Example 1.6 (ℓ^1 Norm) The ℓ^1 norm on $\mathbb{R}^n$, alternatively called the 1-norm, is defined

$$\|x\|_1 = |x_1| + ... + |x_n| = \sum_{i=1}^{n} |x_i|. \tag{1.16}$$

Example 1.7 (ℓ^∞ Norm) The ℓ^∞ norm on $\mathbb{R}^n$, alternatively called the ∞-norm, is defined

$$\|x\|_\infty = \max_{i \in 1,...,n} |x_i|. \tag{1.17}$$

Exercise 1.11 Show that the $\ell^1, \ell^2, \ell^\infty$ norms proposed above are indeed norms on $\mathbb{R}^n$.

With the definition of a norm in hand, we're ready to define a *normed vector space*. Scary as this might sound, the definition of a normed vector space is actually quite innocent.

Definition 1.15 (Normed Vector Space) A normed vector space is a pair $(V, \|\cdot\|)$ of a vector space V and a norm $\|\cdot\|$ on V.

That was easy! Let's consider a couple of basic examples of normed vector spaces.

Example 1.8 $(\mathbb{R}, |\cdot|)$, the real line equipped with the absolute value, is perhaps the simplest example of a normed vector space. Similarly, $(\mathbb{C}, |\cdot|)$, the complex numbers equipped with the complex magnitude, also forms a normed vector space. Note that if we refer to $\mathbb{R}$ or $\mathbb{C}$ as normed vector spaces we will *always* assume the norms in question are the absolute value and complex magnitude, respectively, unless directed otherwise.

Example 1.9 $(\mathbb{R}^n, \|\cdot\|_1)$, $(\mathbb{R}^n, \|\cdot\|_2)$, and $(\mathbb{R}^n, \|\cdot\|_\infty)$ are all normed vector spaces.

Example 1.10 $(\mathbb{R}^{n\times n}, \|\cdot\|_2)$, the vector space of $n \times n$ matrices with real values, along with the matrix 2-norm, $\|A\|_2 = \sigma_{\max}(A)$ is a normed vector space.

Example 1.11 $(\mathbb{R}^{n\times n}, \|\cdot\|_F)$, the vector space of $n \times n$ matrices with real values, along with the Frobenius norm, $\|A\|_F = \sqrt{\text{tr}(A^\top A)}$, is a normed vector space.

These examples are all fairly simple in nature - we take a vector space like $\mathbb{R}^n$ or $\mathbb{R}^{n\times n}$, which we understand well, and place a norm on it that is easy to compute in terms of the entries of the vector. Shortly, we'll discuss some more complex, infinite-dimensional examples. Before we can do this, however, we first need to learn some more analysis.

When we first took a crack at real analysis, we studied different subsets of the real line. We'll begin our study of analysis in normed vector spaces in the same spirit. We start by introducing a special subset of a normed vector space, called an *ϵ-ball*.

Definition 1.16 (Epsilon Ball) Consider a normed vector space $(V, \|\cdot\|)$. Let $\epsilon > 0$ be a fixed real number and $v \in V$ a vector. The ϵ-ball centered at v is the set,

$$B_\epsilon(v) := \{u \in V : \|u - v\| < \epsilon\}. \tag{1.18}$$

Remark 1.6 An ϵ-ball is also commonly referred to as an *ϵ-neighborhood*.

Let's make a few comments on this definition. First, we note that an epsilon ball *doesn't* include points that exactly a distance ϵ away from v; rather it only includes points that are strictly *less* than ϵ away from v. Thus, an ϵ ball has a "fuzzy" boundary. Secondly, we note that the actual shape of an ϵ-ball depends on the choice of norm! For instance, an ϵ-ball in $(\mathbb{R}^2, \|\cdot\|_2)$ will look like an actual ball, while an ϵ ball in $(\mathbb{R}^2, \|\cdot\|_\infty)$ will look like a square.

Exercise 1.12 Sketch the epsilon ball $B_1(0)$ on a pair of coordinate axes in the spaces $(\mathbb{R}^2, \|\cdot\|_1)$, $(\mathbb{R}^2, \|\cdot\|_2)$, and $(\mathbb{R}^2, \|\cdot\|_\infty)$. *Hint: the boundaries of these sets should be "fuzzy."*

Above, we mentioned that an ϵ ball has a "fuzzy" boundary - i.e. $B_\epsilon(v)$ does not have a sharp boundary where the set ends. We know that ϵ-balls are not the only sets with "fuzzy" boundaries we can draw - any set without a sharp boundary fits into this category. How, then, can we specify a general class of sets in a normed vector space with a "fuzzy" boundary? Consider the following definition.

Definition 1.17 (Open Set) Let $(V, \|\cdot\|)$ be a normed vector space. A subset $A \subseteq V$ is said to be an open set if, for all $v \in V$, there exists an $\epsilon > 0$ such that $B_\epsilon(v) \subseteq A$.

Remark 1.7 Note that a few expressions are commonly used to declare a set $A \subseteq V$ is open. The expressions "A is an open set," "A is open" (if the space V is clear), or "A is open in V" (if one wishes to emphasize the vector space), are all equivalently used to declare that a set A is open.

Thus, we declare a set $A \subseteq V$ to be *open* if, around each point in A, we can squeeze in an ϵ-ball that is still contained in the set. Open sets are the natural way of making precise the idea of a set with a "fuzzy" boundary. Let's run through a couple of examples of open sets.

Example 1.12 (Examples of Open Sets)

1. Any open interval, $(a, b),\ a < b$, is an open set in $(\mathbb{R}, |\cdot|)$.

2. Any union of open intervals, $(a,b) \cup (c,d),\ a < b,\ c < d$, is an open set in $(\mathbb{R}, |\cdot|)$.
3. In a normed vector space $(V, \|\cdot\|)$ any open ball $B_\epsilon(v)$ is open.

Exercise 1.13 Confirm that the examples above are indeed open sets. Sketch out each example and confirm that the sets have "fuzzy boundaries."

Now that we've seen some basic examples of open sets, let's outline some basic properties of open sets.

Proposition 1.8 (Properties of Open Sets) *Let $(V, \|\cdot\|)$ be a normed vector space. The open subsets of V satisfy the following properties:*

1. *Nothing & everything: $\emptyset$ and V are open sets.*
2. *Stability under unions: For $\{U_\alpha\}_{\alpha\in\Lambda}$ an arbitrary collection of open sets, the union $\cup_{\alpha\in\Lambda} U_\alpha$ is an open set.*
3. *Stability under finite intersections: For $\{U_i\}_{i=1}^k$ a finite collection of open sets, the intersection $\cap_{i=1}^k U_i$ is an open set.*

Remark 1.8 Just like we write $\{U_i\}_{i=1}^k$ to refer to the collection $\{U_1, ..., U_k\}$ of k sets, we use the notation $\{U_\alpha\}_{\alpha\in\Lambda}$ to refer to an arbitrary collection of sets, indexed by an arbitrary set Λ. Here, α is the index of an individual set in the collection (analogous to i), and Λ is the set of all indices of the collection (analogous to $\{1, ..., k\}$). For instance, if one has a collection of sets corresponding to real numbers, one might write $\{U_\alpha\}_{\alpha\in\mathbb{R}}$. When the index set is clear or unimportant, we will simply write $\{U_\alpha\}$ as shorthand to refer to an arbitrary collection of sets.

Let's review what each condition of the proposition says. The first condition, *nothing & everything*[2], states that the empty set (nothing) and the entire vector space V (everything) are both open sets. The empty set trivially satisfies Definition 1.17, since it has no points to check for, and V satisfies Definition 1.17 since any ϵ-ball is automatically contained in V.

The next condition, stability under unions, states that an *arbitrary* (potentially uncountably infinite) collection of open sets has a union that is also open. Why is this? If we pick a point $v \in \cup_\alpha U_\alpha$, the definition of a union tells us there exists an α for which $v \in U_\alpha$. Since U_α is open, there exists a ball $B_\epsilon(v) \subseteq U_\alpha \subseteq \cup_\alpha U_\alpha$. Thus, the union is open.

The final condition, stability under finite intersections, states that if we have a *finite* collection of open sets, their intersection must be open. The formal argument for this case takes a little more thought than stability under unions - we leave its proof as an exercise (with a hint) below. Why can't we take infinite intersections? Consider the following counterexample. Define a collection $\{B_{1/n}(0)\}_{n\in\mathbb{N}}$, of ϵ-balls centered at the origin with shrinking radius $1/n$. With a little work, one may show that,

$$\bigcap_{n=1}^{\infty} B_{1/n}(0) = \{0\}, \tag{1.19}$$

since all of the sets in the collection shrink down towards zero as $n \to \infty$. Since $\{0\}$ isn't an open set (we can't fit an epsilon ball of positive radius around 0 into the set $\{0\}$), this yields an example of an infinite collection of open sets whose intersection is *not* open. This results in condition (3) holding only for finite intersections.

[2] The nice terminology "nothing and everything" is due to Joel Tropp.

Exercise 1.14 Prove item (2) of Proposition 1.8. Hint: to pick the radius of an ϵ ball that fits in the intersection, try experimenting with the *minimum* of a set of epsilons.

Now that we've defined an open set, a natural question to ask is - what, if anything, is a *closed* set? If an open set is a set with a fuzzy boundary, perhaps a closed set should be a set with a sharp boundary. Although this intuitive definition covers a variety of closed sets, it isn't quite expressive enough to capture everything we need (for instance, are $\emptyset$ and V closed sets?). Consider the following, abstract definition.

Definition 1.18 (Closed Set) Let $(V, \|\cdot\|)$ be a normed vector space. A subset $A \subseteq V$ is said to be a closed set if its complement, $A^c = V \setminus A$, is an open set.

Let's think for a moment about why this definition might match up with our intuition regarding closed sets having "sharp" boundaries. If a set has a fuzzy boundary, then it stands to reason that its complement should have a sharp boundary. Likewise, if a set has a sharp boundary, then its complement should have a fuzzy boundary. Thus, a set with a sharp boundary should be closed.

Exercise 1.15 Draw some pictures to reconcile the "sharp boundary" intuition behind closed sets with the formal definition.

Let's explore some basic properties of closed sets.

Proposition 1.9 (Examples & Properties of Closed Sets) *Let $(V, \|\cdot\|)$ be a normed vector space. The closed subsets of V satisfy the following:*

1. *Nothing & Everything: $\emptyset$ and V are closed sets.*
2. *Stability under intersections: For $\{C_\alpha\}_{\alpha\in\Lambda}$ an arbitrary collection of closed sets, the intersection $\cap_{\alpha\in\Lambda} C_\alpha$ is a closed set.*
3. *Stability under finite unions: For $\{C_i\}_{i=1}^k$ a finite collection of closed sets, the union $\cup_{i=1}^k C_i$ is a closed set.*

Thus, we observe that closed sets seem to satisfy the exact *opposite* properties of open sets! The reason for this is precisely that closed sets are defined as complements of open sets - the complement *flips* the properties of open sets to properties of closed sets.

Exercise 1.16 Use DeMorgan's laws to prove Proposition 1.9 directly from Proposition 1.8.

Exercise 1.17 Produce an infinite collection of closed sets whose union is not closed.

Closed sets have a number of convenient properties that makes them easy to work with when considering matters such as continuity and convergence. As such, given an arbitrary subset A of a normed vector space V, we often wants to find the "smallest" closed set containing A. This way, we can preserve the basic structure of A while gaining the extra properties of a closed set. Consider the following definition, which defines the "smallest" closed set containing any given set.

Definition 1.19 (Closure) Let $(V, \|\cdot\|)$ be a normed vector space and $A \subseteq V$ an arbitrary subset. The closure of A, denoted $\overline{A}$, is the smallest closed set containing A,

$$\overline{A} = \bigcap_{\alpha\in\Lambda} C_\alpha, \text{ where } \{C_\alpha\}_{\alpha\in\Lambda} = \{C_\alpha \subseteq V : A \subseteq C_\alpha \text{ and } C_\alpha \text{ is closed}\}. \tag{1.20}$$

As the closure of A is defined as the *intersection* of all closed sets containing A, one can think of the closure of A as "shrink wrapping" the set A with closed sets. The following two exercises provide some quick sanity checks regarding the closure.

Exercise 1.18 Verify that the closure of any given set is in fact closed.

Exercise 1.19 Show that a set $A \subseteq V$ is closed if and only if $A = \overline{A}$.

The following is a nice consequence of these two exercises.

Example 1.13 Consider a normed vector space $(V, \|\cdot\|)$. The closure of any open ball $B_\epsilon(x)$ in V is the closed ball,

$$\overline{B}_\epsilon(x) = \{y \in V : \|x - y\| \leq \epsilon\}. \tag{1.21}$$

Thus, the closure makes the ordinarily "fuzzy" boundary of an open ball a "sharp" boundary. Try sketching out the intersections of some closed sets containing the ball $B_\epsilon(x)$ to convince yourself that this also follows from the definition of the closure.

Just as we can define the smallest closed set containing a given set, we can define the *largest open set* contained within a given set.

Definition 1.20 (Interior) Let $(V, \|\cdot\|)$ be a normed vector space and $A \subseteq V$ an arbitrary subset. The interior of A, denoted A°, is the largest open set contained in A,

$$A^\circ = \bigcup_{\alpha \in \Lambda} O_\alpha, \text{ where } \{O_\alpha\}_{\alpha \in \Lambda} = \{O_\alpha \subseteq V : O_\alpha \subseteq A \text{ and } O_\alpha \text{ is open}\}. \tag{1.22}$$

Instead of "shrink wrapping" a set with a collection of closed sets, as we did in the case of the closure, we can think of the interior as inflating a collection of open sets inside the given set, until we fill up the entire inside space with open sets. What remains after this operation is the largest open set contained inside the given set.

Exercise 1.20 Verify that the interior of any given set is in fact open.

Exercise 1.21 Show that a set $A \subseteq V$ is open if and only if $A = A^\circ$.

Example 1.14 Consider a normed vector space $(V, \|\cdot\|)$. The interior of any closed ball $\overline{B}_\epsilon(x)$ in V is the open ball, $B_\epsilon(x)$. Thus, the interior makes the ordinarily "sharp" boundary of a closed ball a "fuzzy" boundary.

So far, we've only examined normed vector spaces $(V, \|\cdot\|)$ with a fixed choice of norm on V. Now, we examine what happens when we change the norm on a given vector space. What effect does the choice of norm $\|\cdot\|$ on V have on the properties of the normed vector space $(V, \|\cdot\|)$? As we saw in our definition of an open set above, the choice of a norm on a vector space *entirely* determines which sets are open and closed. Thus, to determine the effect of a norm on the vector space, it's logical to ask - when do two norms on a single vector space determine the same open sets?

As a first step towards answering this question, we define what it means for two norms on the same vector space to be *equivalent*.

Definition 1.21 (Equivalent Norms) Consider a vector space V. Two norms, $\|\cdot\|_a$ and $\|\cdot\|_b$, on V are said to be equivalent if there exist constants $k_1, k_2 > 0$ for which

$$k_1 \|v\|_a \leq \|v\|_b \leq k_2 \|v\|_a, \ \forall v \in V. \tag{1.23}$$

Thus, two norms on a vector space are said to be equivalent if one norm can be "sandwiched" between some positive multiples of the other. We now show that the equivalence of two norms *entirely* determines whether the open sets of the normed vector spaces determined by the norms will be the same.

Proposition 1.10 (Equivalent Norms Determine the Same Open Sets) *Consider a vector space V and norms $\|\cdot\|_a$ and $\|\cdot\|_b$ on V. The open sets of $(V, \|\cdot\|_a)$ and $(V, \|\cdot\|_b)$ are the same if and only if $\|\cdot\|_a$ and $\|\cdot\|_b$ are equivalent norms.*

Proof First, suppose $\|\cdot\|_a$ and $\|\cdot\|_b$ are equivalent. Consider an arbitrary open ball $B^a_{\epsilon_0}(x)$, with respect to the norm $\|\cdot\|_a$. By the definition of norm equivalence, it follows that there exists an open ball $B^b_{\epsilon_1}(x)$ with respect to the norm $\|\cdot\|_b$, satisfying $B^b_{\epsilon_1}(x) \subseteq B^a_{\epsilon_0}(x)$.

Now, let U_a be an arbitrary open set of $(V, \|\cdot\|_a)$. We aim to show that U_a is also open in $(V, \|\cdot\|_b)$. Let $x \in U_a$ be arbitrary. Since U_a is open in x, there exists a ball $B^a_\epsilon(x) \subseteq U_a$. But, by the reasoning above, there exists a ball $B^b_\epsilon(x) \subseteq B^a_\epsilon(x) \subseteq U_a$. So, around every point in U_a, we can squeeze in a ball defined with respect to $\|\cdot\|_b$. We conclude that U_a is open in $(V, \|\cdot\|_b)$. To show that any open set U_b in $(V, \|\cdot\|_b)$ is also open in $(V, \|\cdot\|_a)$, one follows the same reasoning.

Now, we prove the opposite direction. For this direction, all we have to work with are open sets. We'll have to be a little bit clever, and pick our open sets such that they directly give us the constants we need for norm equivalence. Suppose the open sets of $(V, \|\cdot\|_a)$ and $(V, \|\cdot\|_b)$ are the same - i.e. a set $U \subseteq V$ is open in $(V, \|\cdot\|_a)$ if and only if it is open in $(V, \|\cdot\|_b)$. Consider the open ball of radius 1 around the origin in $\|\cdot\|_b$, denoted $B^b_1(0)$. Since $B^b_1(0)$ is open in $\|\cdot\|_b$, it must also be open in $\|\cdot\|_a$. Therefore, there exists a ball $B^a_{\epsilon_1}(0)$ in $\|\cdot\|_a$ satisfying $B^a_{\epsilon_1}(0) \subseteq B^b_1(0)$. There also exists a ball $B^a_{\epsilon_2}(0)$ in $\|\cdot\|_a$ satisfying $B^b_1(0) \subseteq B^a_{\epsilon_2}(0)$. The inclusion of balls,

$$B^a_{\epsilon_1}(0) \subseteq B^b_1(0) \subseteq B^a_{\epsilon_2}(0), \tag{1.24}$$

implies that for any vector $v \in V$ satisfying $\|v\|_b = 1$,

$$\epsilon_1 \|v\|_a \leq 1 \leq \epsilon_2 \|v\|_a . \tag{1.25}$$

Now, consider an arbitrary, nonzero vector $v \in V$. By the positive homogeneity property of norms, it follows that,

$$\epsilon_1 \left\| \frac{v}{\|v\|_b} \right\|_a \leq \frac{\|v\|_b}{\|v\|_b} \leq \epsilon_2 \left\| \frac{v}{\|v\|_b} \right\|_a \tag{1.26}$$

$$\frac{\epsilon_1}{\|v\|_b} \|v\|_a \leq \frac{\|v\|_b}{\|v\|_b} \leq \frac{\epsilon_2}{\|v\|_b} \|v\|_a \tag{1.27}$$

$$\epsilon_1 \|v\|_a \leq \|v\|_b \leq \epsilon_2 \|v\|_a . \tag{1.28}$$

For the remaining case of $v = 0$, the final inequality above holds trivially. We conclude that $\|\cdot\|_a$ and $\|\cdot\|_b$ are in fact equivalent. □

The next result - the proof of which we leave to the problems at the end of the section - is one of the most fundamental results in analysis in normed vector spaces. It tells us that, in any finite-dimensional vector space, *all* norms are equivalent. Because of this fact, the choice of norm in a finite-dimensional vector space often isn't critical - any two norms on

a finite-dimensional vector space will produce normed vector spaces with similar analytical properties.

Theorem 1.1 (Norm Equivalence in Finite Dimensions) *Consider a finite-dimensional vector space V. Any two norms $\|\cdot\|_a$ and $\|\cdot\|_b$ on V are equivalent.*

Proof See [4] for the details. □

Now that we've discussed the structure of some important subsets of normed vector spaces, we turn our attention to the *continuity* of maps between normed vector spaces. Consider the following definition.

Definition 1.22 (Continuity) Consider two normed vector spaces, $(V, \|\cdot\|_V)$ and $(W, \|\cdot\|_W)$. A mapping $f : V \to W$ is continuous at a point $x \in V$ if, for all $\epsilon > 0$, there exists a $\delta > 0$ (possibly a function of ϵ and x) for which

$$\|x - y\|_V < \delta \implies \|f(x) - f(y)\|_W < \epsilon. \tag{1.29}$$

If $f : V \to W$ is continuous at all x in a subset $U \subseteq V$, f is said to be continuous on U. Likewise, if f is continuous at all $x \in V$, it is simply said to be continuous.

Let's briefly check our understanding of this definition. Definition 1.22 states that, if a mapping $f : V \to W$ is continuous, $f(x)$ can change by a small amount if x changes by a small amount. It's important to note - the norm on $x - y$ is the norm $\|\cdot\|_V$ (since $x, y \in V$), while the norm on $f(x) - f(y)$ is the norm $\|\cdot\|_W$ (since $f(x), f(y) \in W$). One must take special care to ensure the correct norms are being used on $x - y$ and $f(x) - f(y)$ - these norms will *not* in general be the same!

Proposition 1.11 (Properties of Continuous Functions) *Let $(U, \|\cdot\|_U)$, $(V, \|\cdot\|_V)$, and $(W, \|\cdot\|_W)$ be normed vector spaces over $\mathbb{K}$ and $f : V \to W$, $g : V \to W$, and $h : W \to U$ be continuous mappings.*

1. *Algebraic Combinations: For any continuous functions $\alpha, \beta : V \to \mathbb{K}$, the function $p : V \to W$, defined $p(x) = \alpha(x) \cdot f(x) + \beta(x) \cdot g(x)$, is also continuous.*
2. *Composition: The composition $h \circ f : V \to U$ is continuous.*

Remark 1.9 In item (1) of the proposition above, we use continuous functions $\alpha, \beta : V \to \mathbb{K}$. Here, we treat $\mathbb{K}$ as the normed vector space $(\mathbb{K}, |\,\|\cdot\|\,|)$, where $|\cdot|$ represents either absolute value (for $\mathbb{K} = \mathbb{R}$) or complex magnitude (for $\mathbb{K} = \mathbb{C}$). It's necessary to treat $\mathbb{K}$ as a normed vector space in order to apply Definition 1.22!

When studying a function $f : V \to \mathbb{R}$ from a normed vector space V to the real line $\mathbb{R}$, it's useful to have some criteria to determine whether or not the function attains a maximum or minimum value on a given set. A special class of sets - termed *compact sets* - enable exactly this ability, among many others. In order to provide a sufficiently abstract definition of a compact set, we first require the definition of an open cover.

Definition 1.23 (Cover/Open Cover) Consider a normed vector space $(V, \|\cdot\|)$ and a subset $A \subseteq V$. A collection $\{U_\alpha\}_{\alpha \in \Lambda}$ of subsets of V is said to be a *cover* of A if

$$A \subseteq \bigcup_{\alpha \in \Lambda} U_\alpha. \tag{1.30}$$

If each U_α is open, the collection $\{U_\alpha\}_{\alpha \in \Lambda}$ is said to be an *open cover* of A. A cover is said to be *finite* if it contains a finite number of sets.

Remark 1.10 Sometimes, one will encounter the phrase, "$\{U_\alpha\}_{\alpha\in\Lambda}$ *covers* A." This is simply another way of saying that $\{U_\alpha\}_{\alpha\in\Lambda}$ is a cover of A.

This definition tells us that a *cover* of a given set is a collection of sets that, when pasted together, contain the given set.

Definition 1.24 (Subcover) Consider a normed vector space $(V, \|\cdot\|)$, a subset $A \subseteq V$, and a cover $\{U_\alpha\}_{\alpha\in\Lambda}$ of A. A subcollection $\{V_\beta\}_{\beta\in B} \subseteq \{U_\alpha\}_{\alpha\in\Lambda}$ is said to be a subcover of $\{U_\alpha\}_{\alpha\in A}$ if $\{V_\beta\}_{\beta\in B}$ is still a cover of A. A subcover is said to be *finite* if it contains a finite number of sets.

Remark 1.11 It's extremely important to note that $\{V_\beta\}_{\beta\in B} \subseteq \{U_\alpha\}_{\alpha\in\Lambda}$ *does not* mean that the sets V_β are subsets of the sets U_α. Here, the subset relation is a relation on the *collections* $\{V_\beta\}$ and $\{U_\alpha\}$ - we pick out a few of the elements of the collection $\{U_\alpha\}$ to form $\{V_\beta\}$. If the indices of the sets are not changed when one picks these elements out, one will have $B \subseteq \Lambda$.

Remark 1.12 Notice how the use of the prefix *sub* mirrors that of a subspace. Generally a sub-"object" is a subset of an "object" that retains the object's key properties. This is true of a subspace, wherein we have a subset of vector space that remains a vector space, and of a subcover, wherein we have a subset of a cover that remains a cover.

Thus, a subcover of a cover of a given set takes picks out a few sets from the cover that, when pasted together, still cover the given set. Armed with these definitions, we're ready to state an abstract definition of a compact set.

Definition 1.25 (Compact Set) Let $(V, \|\cdot\|)$ be a normed vector space. A subset $K \subseteq V$ is said to be a compact set if every open cover of K has a finite subcover.

Remark 1.13 To say that a set $K \subseteq V$ is a compact set, one will often say "K is compact," or "K is compact in V." This is just like how, to declare a set O is an open set, we said "O is open," or "O is open in V."

At first glance, this definition seems entirely mystifying. In order to pull back the curtain on compactness, we'll study a few basic results of compact sets. The following result tells us a few properties that compact sets must satisfy in a normed vector space. Note that the proofs of the next few compactness results are generally out of scope of our treatment of the material - one may consult the references at the end of the chapter for their proofs.

Proposition 1.12 (Compact Sets are Closed & Bounded) *Let $(V, \|\cdot\|)$ be a normed vector space and $K \subseteq V$ be a compact subset of V. Then, K must satisfy the following two properties:*

1. *Closed: K is closed in V.*
2. *Bounded: K is bounded in V - there exists an $M \geq 0$ such that $\|x\| \leq M$ for all $x \in M$.*

It's natural to wonder whether the converse of the result above is true - is every closed & bounded set compact? In *finite-dimensional* normed vector spaces, this turns out to be true, but in infinite-dimensional spaces, this is generally false. We postpone a more complete characterization of the infinite-dimensional case to our later discussion of Banach spaces. For now, we content ourselves with the (very nice) finite-dimensional result.

Proposition 1.13 (Compact Sets in Finite-Dimensional Spaces) *Consider a normed vector space $(V, \|\cdot\|)$. If V is finite-dimensional, then a subset $K \subseteq V$ is compact if and only if it is closed & bounded.*

This proposition produces a quick and easy way to verify a given set in a finite-dimensional vector space is compact. In the following two examples, we illustrate how easy it is to determine some simple subsets of a finite-dimensional space are compact.

Example 1.15 In $\mathbb{R}$, any closed interval $[a, b]$, $a \leq b$ is compact, since any closed interval is closed and bounded.

Example 1.16 In $\mathbb{R}^n$, any closed box $[a, b]^n = \{x \in \mathbb{R}^n : a \leq x_i \leq b\}$, $a \leq b$, is compact, since closed boxes are closed and bounded.

Example 1.17 In a finite-dimensional normed vector space $(V, \|\cdot\|)$, any closed ball $\overline{B}_\epsilon(x) = \{y \in V : \|x - y\| \leq \epsilon\}$ is compact, since closed balls are closed and bounded.

As alluded to earlier, one of the most appealing features of compact sets is that they interact well with continuous functions. In particular, a continuous function *always* achieves a maximum and minimum value on a compact set. In the following proposition, we summarize some of the most useful properties.

Proposition 1.14 (Compactness & Continuity) *Consider two normed vector spaces, $(V, \|\cdot\|_V)$ and $(W, \|\cdot\|_W)$.*

1. *Let $f : V \to W$ be a continuous map. For any compact set $K \subseteq V$, the image of K under f, $f(K) \subseteq W$, is also compact.*
2. *Let $f : V \to \mathbb{R}$ be a continuous map. For any compact set $K \subseteq V$, the set $f(K)$ has a maximum and a minimum value. Consequently,*

$$\sup_{x \in K} f(x) = \max_{x \in K} f(x) \text{ and } \inf_{x \in K} f(x) = \min_{x \in K} f(x). \tag{1.31}$$

Property (2) is particularly useful when bounding the values of a function over a given set. Oftentimes, one can find a compact set containing the given set, and use the maxima and minima on the compact set to bound the function values on the given set. The fact that a continuous function achieves a maximum and a minimum value over a compact set means that there is *no risk* of the function "blowing up" to $+\infty$ or $-\infty$ on the set.

The following exercise presents a nice consequence of this proposition.

Exercise 1.22 Let $(V, \|\cdot\|_V)$ and $(W, \|\cdot\|_W)$ be normed vector spaces and $f : V \to W$ be a continuous map. Show that on any compact set $K \subseteq V$, the function $g : V \to \mathbb{R}$, defined $g(x) = \|f(x)\|_W$, attains a maximum and a minimum value. *Hint: are norms continuous functions?*

Definition 1.22 offers one definition of continuity of a map between normed vector spaces. What else might we be interested in? Oftentimes, the form of continuity presented in Definition 1.22 is not strong enough! Next, we consider a stronger form of continuity, called Lipschitz continuity.

Definition 1.26 (Lipschitz Continuity) Let $(V, \|\cdot\|_V)$ and $(W, \|\cdot\|_W)$ be two normed vector spaces. A mapping $f : V \to W$ is said to be Lipschitz continuous on V if there exists a constant $L \geq 0$, called a Lipschitz constant, for which

$$\|f(x) - f(y)\|_W \leq L \|x - y\|_V , \ \forall x, y \in V. \tag{1.32}$$

Remark 1.14 Unlike continuity, we specify Lipschitz continuity as a global property of a function. Lipschitz continuity is defined over the entire space, rather than at a single point.

By nature of the name Lipschitz *continuity*, it's natural to expect a Lipschitz continuous function to be continuous in the sense of Definition 1.22. In the next proposition, we confirm that our expectations are met.

Proposition 1.15 *Consider a function $f : V \to W$ between normed vector spaces $(V, \|\cdot\|_V)$ and $(W, \|\cdot\|_W)$. If f is Lipschitz continuous on V, then it is continuous on V.*

Proof Suppose $f : V \to W$ is Lipschitz continuous. We must show that f is continuous at every $x \in V$. First, fix a point $x \in V$ and a number $\epsilon > 0$. To show that f is continuous at x, we must find a number $\delta > 0$ - possibly dependent on ϵ and x, such that $\|x - y\|_V < \delta$ implies $\|f(x) - f(y)\|_W < \epsilon$. In order to identify such a δ, we appeal to the definition of Lipschitz continuity. By Definition 1.26, there exists a constant $L \geq 0$ such that for all $x, y \in V$, $\|f(x) - f(y)\|_W \leq L \|x - y\|_V$. Looking at the inequalities,

$$\|f(x) - f(y)\|_W \leq L \|x - y\|_V \qquad \text{(what we have)} \tag{1.33}$$

$$\|f(x) - f(y)\|_W < \epsilon \qquad \text{(what we want)}, \tag{1.34}$$

it seems reasonable that choosing $\delta = \epsilon / L$ will meet our needs. Let's check that this value works. For any y satisfying $\|x - y\|_W < \delta = \epsilon / L$, we have

$$\|f(x) - f(y)\|_W \leq L \|x - y\|_V < L \cdot \frac{\epsilon}{L} = \epsilon. \tag{1.35}$$

Thus, we conclude that f is continuous at x. Since x was chosen arbitrarily, we conclude that f is continuous on V. □

Above, we showed that all Lipschitz continuous functions are continuous. What about the other direction? Are all continuous functions Lipschitz? As the following exercise illustrates, Lipschitz continuity is a *strictly stronger* condition than continuity.

Exercise 1.23 The function $f : \mathbb{R} \to \mathbb{R}$, defined $f(x) = x^2$ is well-known to be continuous. Show that it is *not* Lipschitz continuous. *Hint: proceed by contradiction.*

An important example of a Lipschitz function on any normed vector space $(V, \|\cdot\|)$ is the norm $\|\cdot\|$ itself, as a function from $V \to \mathbb{R}$.

Proposition 1.16 *For $(V, \|\cdot\|)$ a normed vector space, $\|\cdot\| : V \to \mathbb{R}$ is Lipschitz continuous.*

This result implies that the norm of a normed vector space is a continuous function.

Exercise 1.24 Prove Proposition 1.23.

So far in our discussion of continuity, we've dealt only with *arbitrary* mappings between normed vector spaces. What can we say about the continuity of *linear* transformations? Consider, for example, the simple linear transformation from $\mathbb{R} \to \mathbb{R}$, defined $f(x) = ax$, $a \in \mathbb{R}$. It's clear that this function is continuous; in fact, it is Lipschitz continuous with Lipschitz constant $|a|$. Therefore, as an initial guess, it seems not too far-fetched to suggest that linear transformations between normed vector spaces are Lipschitz continuous.

Unfortunately, this is *not* true for general linear transformations between normed vector spaces! The class of linear transformations for which this *is* true is called the class of *bounded linear operators*.

Definition 1.27 (Bounded Linear Operator/Transformation) Consider two normed vector spaces, $(V, \|\cdot\|_V)$ and $(W, \|\cdot\|_W)$, and a linear transformation $A : V \to W$. A is said to be a bounded linear operator/transformation if there exists a $K \geq 0$ for which

$$\|Ax\|_W \leq K \|x\|_V , \ \forall x \in V. \tag{1.36}$$

Remark 1.15 Here, we've undergone a little bit of a terminology change from "transformation" to "operator." The "operator" terminology is more commonly used in functional analysis, while "transformation" is more often seen in linear algebra. These two terms are entirely interchangeable - for instance one may say "bounded linear operator" or "bounded linear transformation" in reference to the definition above.

Thus, a linear transformation is bounded if it doesn't "scale" any vector to be arbitrarily large! Taking a closer look at the definition of a bounded linear operator, one immediately notices a similarity to the definition of Lipschitz continuity. This observation leads to the following result.

Proposition 1.17 *Bounded linear operators are Lipschitz continuous.*

Proof Consider a bounded linear operator $A : V \to W$. Such a map satisfies $\|Ax\|_W \leq K \|x\|_V , \ \forall x \in V$, where $K \geq 0$ is some fixed constant. By the axioms of a vector space, for all $x, y \in V$, $x - y \in V$. Therefore, one has that for all $x, y \in V$,

$$\|Ax - Ay\|_W = \|A(x - y)\|_W \leq K \|x - y\|_V . \tag{1.37}$$

We conclude that A is Lipschitz continuous with Lipschitz constant K. □

One of the most interesting and useful facts about the set of bounded linear operators between two vector spaces is that they themselves form a vector space! In the following theorem, we define the vector space of bounded linear operators.

Theorem 1.2 (Vector Space of Bounded Linear Operators) *Consider two normed vector spaces $(V, \|\cdot\|_V)$ and $(W, \|\cdot\|_W)$ over $\mathbb{K}$. The set of all bounded linear operators between V and W, denoted $\mathcal{L}(V, W)$, is itself a vector space under the operations $+$ and $(\cdot)$, defined*

$$(A + B)(v) = A(v) + B(v) \tag{1.38}$$
$$(c \cdot A)(v) = c(Av), \tag{1.39}$$

for all $v \in V$ and $c \in \mathbb{K}$.

Exercise 1.25 Prove Theorem 1.2.

One may also verify that the *composition* of any two bounded linear operators is a bounded linear operator.

Proposition 1.18 (Composition of Bounded Linear Operators) *Consider two bounded linear operators, $A \in \mathcal{L}(V, W)$ and $B \in \mathcal{L}(U, V)$. The composition of A and B, denoted AB, is defined $ABv = A(Bv)$. AB is a bounded linear operator from U to W.*

Remark 1.16 The convention of writing the composition $A \circ B$ as AB derives from matrix multiplication, where one writes the product of two matrices A and B as AB.

Now that we've defined a vector space of bounded linear operators, it's natural to seek out a norm that turns this vector space into a normed vector space. A natural choice would be to define a norm on $\mathcal{L}(V,W)$ in terms of the norms on V and W.

Definition 1.28 (Induced Operator Norm) Let $(V,\|\cdot\|_V)$ and $(W,\|\cdot\|_W)$ be normed vector spaces and $A \in \mathcal{L}(V,W)$ a bounded linear operator from V to W. The induced operator norm of A is defined,

$$\|A\|_{V,W} = \sup_{x\in V\setminus\{0\}} \frac{\|Ax\|_W}{\|x\|_V}. \tag{1.40}$$

Remark 1.17 The "induced" in the name "induced operator norm" refers to the fact that the norm $\|\cdot\|_{V,W}$ is induced by the norms on V and W. For convenience, one often refers to an induced operator norm simply as an "operator norm." If no choice of norm on an operator is specified, it is typically assumed to be the operator norm induced by the normed vector spaces it maps between.

Remark 1.18 The $\sup_{x\in V\setminus\{0\}}$ simply ensures that we don't divide by zero in the quotient defining the operator norm. Shortly, we'll state a few equivalent ways of calculating the operator norm that avoid the use of a quotient.

An important special case of the operator norm is the following.

Definition 1.29 (Left Multiplication Operator/Induced Matrix Norm) Consider a matrix $A \in \mathbb{K}^{m\times n}$, and two normed vector spaces $(\mathbb{K}^m,\|\cdot\|_b)$ and $(\mathbb{K}^n,\|\cdot\|_a)$.

1. Left Multiplication Operator: the linear operator of left multiplication with A, is the operator $L_A : \mathbb{K}^n \to \mathbb{K}^n$, defined $L_A v := Av$ for all $v \in \mathbb{K}^n$.
2. Induced Matrix Norm: the induced matrix norm of A is defined $\|A\|_{a,b} = \|L_A\|_{V,W}$.

Remark 1.19 If the spaces $\mathbb{K}^m$ and $\mathbb{K}^n$ have the same type of norm, for instance the 2-norm, one will simply denote the induced matrix norm as $\|A\|_2$, rather than $\|A\|_{2,2}$. In this case, one refers to the induced 2-norm of A simply as the 2-norm of A.

Example 1.18 (Matrix 2-Norm) The induced 2-norm of $A \in \mathbb{K}^{m\times n}$ (where $\mathbb{K} = \mathbb{R}$ or $\mathbb{K} = \mathbb{C}$), is calculated $\|A\|_2 = \sigma_{\max}(A)$.

Proposition 1.19 (Operator Norms Define a Normed Vector Space) *Let $(V,\|\cdot\|_V)$ and $(W,\|\cdot\|_W)$ be normed vector spaces. The space $\mathcal{L}(V,W)$, equipped with the operator norm $\|\cdot\|_{V,W}$, is a normed vector space.*

Exercise 1.26 Prove Proposition 1.19 by showing $\|\cdot\|_{V,W}$ is a norm on $\mathcal{L}(V,W)$.

It's important to note that the operator norm $\|\cdot\|_{V,W}$ is generally not the only norm giving $\mathcal{L}(V,W)$ the structure of a normed vector space. Rather, operator norms are a *natural* choice of norm that derive from the normed vector spaces they map between.

Example 1.19 The Frobenius norm of a matrix $A \in \mathbb{K}^{m\times n}$, $\|A\|_F = \sqrt{\operatorname{tr}(A^*A)}$, is not induced by any ℓ^p-norms on $\mathbb{K}^n$ and $\mathbb{K}^m$.

In addition to endowing the vector space $\mathcal{L}(V,W)$ the structure of a normed vector space, the induced operator norm $\|\cdot\|_{V,W}$ enjoys a number of other useful properties.

Proposition 1.20 (Properties of the Operator Norm) *Consider three normed vector spaces, $(U, \|\cdot\|_U)$, $(V, \|\cdot\|_V)$, and $(W, \|\cdot\|_W)$. The following properties are satisfied:*

1. *Submultiplicative: For all $A \in \mathcal{L}(V, W)$ and $B \in \mathcal{L}(U, V)$, $\|AB\|_{U,W} = \|A\|_{V,W} \|B\|_{V,W}$.*
2. *Vector Inequality: For all $A \in \mathcal{L}(V, W)$ and $x \in V$, $\|Ax\|_W \leq \|A\|_{V,W} \|x\|_V$.*
3. *Equivalent Definitions: The operator norm is equivalently computed by the formulas,*

$$\|A\|_{V,W} = \sup_{\|x\|_V = 1} \|Ax\|_W \text{ and } \|A\|_{V,W} = \inf\{K \geq 0 : \|Ax\| \leq K \|x\| \ \forall x \in V\}. \quad (1.41)$$

Let's run through the different components of this proposition. The first item of the proposition tells us that operator norms are submultiplicative - if we take two bounded linear operators and *compose* them, the operator norm of their composition $AB : U \to W$ is bounded above by the product of their operator norms. The second item tells us that the operator norm provides us with an upper bound on how much a linear operator scales the norm of a vector. The final item tells us two equivalent ways of calculating the operator norm. Due to the scaling property of norms, one has

$$\sup_{\|x\|_V = 1} \|Ax\|_W = \sup_{x \in V \setminus \{0\}} \frac{\|Ax\|_W}{\|x\|_V}. \quad (1.42)$$

This enables us to calculate the operator norm without the use of a quotient. The second formula for the operator norm - which is not as practical for computation - yields an interpretation of the operator norm as the *tightest possible* Lipschitz constant of the operator A. To complete our study of the space of linear operators, we define some special bounded linear operators.

Definition 1.30 (Identity Operator) Let V be a normed vector space. The identity operator on V, $\mathrm{Id}_V : V \to V$ is defined $\mathrm{Id}_V\, v = v$, for all $v \in V$.

Exercise 1.27 Assuming both the domain and codomain instances of V are equipped with the same norm, confirm that the identity operator is a bounded linear operator.

Note that if $V = \mathbb{R}^n$, the identity linear operator $\mathrm{Id}_{\mathbb{R}^n}$ coincides with the linear operator L_{I_n} of left multiplication by the $n \times n$ identity matrix, I_n.

Definition 1.31 (Invertible Linear Operator) Let V and W be normed vector spaces. A linear operator $A \in \mathcal{L}(V, W)$ is said to be invertible if there exists a $B \in \mathcal{L}(W, V)$ for which $AB = \mathrm{Id}_W$ and $BA = \mathrm{Id}_V$. In this case, B is said to be the *inverse* of A, denoted A^{-1}.

Remark 1.20 Note that we have used the language *the* inverse. One may show that the inverse of a linear operator is always unique - thus, it makes sense to talk about *the* inverse rather than *an* inverse.

Using invertible linear operators, one defines what it means for two vector spaces to be *isomorphic*.

Definition 1.32 (Isomorphism) Two normed vector spaces $(V, \|\cdot\|_V)$ and $(W, \|\cdot\|_W)$ are said to be isomorphic if there exists an invertible linear mapping $A \in \mathcal{L}(V, W)$. Such an A is said to be an isomorphism of the vector spaces V and W.

Roughly speaking, if two vector spaces are isomorphic, they share the same underlying "algebraic structure." It is a fundamental theorem of linear algebra that every finite-dimensional vector space over $\mathbb{R}$ is isomorphic to $\mathbb{R}^n$ for some n. We conclude that any finite-dimensional vector space over $\mathbb{R}$ has the same *algebraic structure* as $\mathbb{R}^n$, for some n. Consequently, it's often sufficient to prove results for general finite-dimensional vector spaces in $\mathbb{R}^n$.

1.4 Banach Spaces

Thus far in our study of normed vector spaces, we've focused primarily on questions regarding the structure of sets, norms, and mappings of vector spaces. In this section, we study *convergence*, another fundamental subject of basic analysis. In order to fully understand convergence, we'll need a little bit more structure than the basic normed vector space setting. In this section, we develop the basic theory of *Banach spaces*, a special class of normed vector space which enjoys additional convergence properties.

In order to define Banach spaces, we first need to understand sequences in normed vector spaces. Let's start by discussing sequences in $\mathbb{R}$. In $\mathbb{R}$, we think of sequences as ordered lists of real numbers, for example,

$$(a_1, a_2, a_3, ...), \text{ where each } a_i \in \mathbb{R}. \tag{1.43}$$

What such a sequence *really* is is a mapping from $\mathbb{N} \to \mathbb{R}$, taking an index of the sequence in $\mathbb{N}$ and mapping it to a value in $\mathbb{R}$. Above, we map $1 \mapsto a_1$, $2 \mapsto a_2$, and so on. This definition is generalized to the setting of vector spaces.

Definition 1.33 (Sequence) A sequence in a vector space V is a mapping $a : \mathbb{N} \to V$. Individual elements of the sequence are denoted a_n, while the entire sequence object is denoted $\{a_n\} \subseteq V$.

Remark 1.21 The definition of a sequence of a mapping $a : \mathbb{N} \to V$ suggests that we might write elements of a sequence as $a(n)$ rather than a_n. However, as we often like to distinguish sequences from "regular" mappings, we favor the notation a_n over $a(n)$. Likewise, instead of referring to a sequence via the actual mapping $a : \mathbb{N} \to V$, one writes $\{a_n\} \subseteq V$ to define a sequence $a : \mathbb{N} \to V$.

Using the language of normed vector spaces, we may formulate a precise definition for the *convergence* of a sequence.

Definition 1.34 (Sequential Convergence) Consider a normed vector space $(V, \|\cdot\|)$. A sequence $\{a_n\} \subseteq V$ is said to converge if there exists a vector $a \in V$ such that, for all $\epsilon > 0$, there exists an $N \in \mathbb{N}$ (possibly dependent on ϵ, for which $n \geq N \implies \|a_n - a\| < \epsilon$. In this case, one says that a is the limit of $\{a_n\}$, and writes $\lim_{n\to\infty} a_n = a$.

What is this definition saying? Essentially, a sequence $\{a_n\}$ converges to a limit a if a_n eventually comes and remains arbitrarily close to a. Here, ϵ encodes a specification of how close we want the sequence to come to a, and N tells us how far into the sequence we need to look for $\{a_n\}$ to come and remain within a distance ϵ of a.

Since we're working in normed vector spaces, sequences can take on far more interesting forms than simple sequences of real numbers. Sequences of *functions* will be of particular

interest to us. In the following example, we focus on sequences in a particularly important function space.

Example 1.20 (Uniform Convergence) Consider the normed vector space $(V, \|\cdot\|_\infty)$, where V is the set of functions $f : \mathbb{R} \to \mathbb{R}$ with finite supremum norm,

$$\|f\|_\infty = \sup_{t\in\mathbb{R}} |f(t)|. \tag{1.44}$$

Sequences in this normed vector space are sequences of *functions*, $\{f_n\} \subseteq V$. If a sequence of functions $\{f_n\}$ converges to a function f with respect to the supremum norm $\|\cdot\|_\infty$, one says that f_n *converges uniformly* to f.

Uniform convergence has a particularly useful relationship with differentiation.

Theorem 1.3 (Uniform Convergence & Differentiation) *Let $I \subseteq \mathbb{R}$ be a compact interval. Suppose $\{f_n\}$, $f_n : I \to \mathbb{R}$, is a sequence of functions for which $\{f_n(t_0)\}$ converges for some $t_0 \in I$. If $\{f_n'\}$ converges uniformly on I, then $\{f_n\}$ converges uniformly on I to a differentiable function f, with $f'(t) = \lim_{n\to\infty} f_n'(t)$ for all $t \in I$.*

Proof See [30] for the details. □

Fortunately, sequences interact well with the algebraic operations of a vector space, as well as with continuous functions between normed vector spaces.

Proposition 1.21 (Sequential Limit Properties) *Let $(V, \|\cdot\|_V)$ and $(W, \|\cdot\|_W)$ be normed vector spaces over $\mathbb{K}$ and $\{a_n\}, \{b_n\} \subseteq V$ convergent sequences with limits a and $b \in V$.*

1. *Algebraic Combinations: for $\alpha, \beta \in \mathbb{K}$, $\lim_{n\to\infty}(\alpha a_n + \beta b_n) = \alpha a + \beta b$.*
2. *Function Composition: for any continuous mapping $f : V \to W$, $\lim_{n\to\infty} f(a_n) = f(a)$.*

So far, we've discussed a definition of convergence which requires a candidate limit in order to certify convergence. Although in simple cases, it's not too hard to come up with a candidate for a limit (e.g. we can guess $1/n \to 0$ without too much trouble), for more complex sequences this becomes challenging.

What we'd like is a way to certify a sequence converges *without* knowing ahead of time what the sequence converges to. Let's reason about how we might do this in $\mathbb{R}$. In $\mathbb{R}$, our intuition tells us that a sequence $\{a_n\}$ will converge if its terms get closer and closer together as n grows large. We generalize this idea to normed vector spaces with the following definition.

Definition 1.35 (Cauchy Sequence) Let $(V, \|\cdot\|)$ be a normed vector space. A sequence $\{a_n\} \subseteq V$ is said to be a Cauchy sequence if, for all $\epsilon > 0$, there exists an $N > 0$ (possibly dependent on ϵ), such that $n, m \geq N \implies \|a_n - a_m\| < \epsilon$.

Remark 1.22 Frequently, instead of saying that a sequence $\{a_n\} \subseteq V$ is a Cauchy sequence, we will simply say "$\{a_n\}$ is Cauchy," and drop the extra label of "sequence."

Based on the definition, we see that a sequence is Cauchy if, given any $\epsilon > 0$, the terms of the sequence eventually come and remain within a distance ϵ of each other. In other words, for large n, the terms of the sequence begin to "cluster" together. Notably, since this definition only relies on the terms of the sequence, one does not require a candidate limit to prove that a given sequence is Cauchy.

Let's determine if Cauchy sequences meet our requirement. Is is true that a Cauchy sequence in any normed vector space converges? In finite dimensions, the answer is *yes*, but in infinite dimensions, the answer is (of course) much more subtle. Since it is not a given that a Cauchy sequence converges in any given normed vector space, we define a special class of normed vector spaces in which Cauchy sequences *always* converge. Although this might initially seem like a restrictive condition, this class of normed vector spaces turns out to be quite vast, containing a wide variety of interesting spaces.

Definition 1.36 (Banach Space) A Banach space is a normed vector space $(V, \|\cdot\|)$ in which every Cauchy sequence converges to a limit in V.

Remark 1.23 Spaces in which Cauchy sequences always converge to a limit in the space are also referred to as *complete spaces* - this terminology extends to more general spaces beyond normed vector spaces. Using this language, one refers to a Banach space is a *complete* normed vector space.

Remark 1.24 It's critical to note that for a space V to be a Banach space, the limits of all Cauchy sequences must belong to V. It's not enough to have a Cauchy sequence converge to a limit outside of the space V - the limit must be contained in V itself.

Since every Cauchy sequence in a Banach space converges, Banach spaces afford us the ability to study the convergence of sequences without actually knowing what the limits of the sequences might be. This is an incredibly powerful ability that has far reaching implications. In fact, one can use the consequences of completeness to prove the existence and uniqueness of solutions to certain differential equations.

Now, we consider some important examples of Banach spaces. As we alluded to above, it is true that *every* finite-dimensional normed vector space is a Banach space.

Theorem 1.4 *Any finite-dimensional normed vector space is a Banach space.*

Proof See Problem 1.3. □

This result alone encompasses an enormous class of interesting spaces. Because of Theorem 1.4, we must turn to infinite-dimensional spaces to find other examples of Banach spaces. A particularly rich class of Banach spaces is supplied by the ℓ^p and L^p spaces, which we now define.

Definition 1.37 (ℓ^p Space) Fix a number $p \in [1, \infty)$. The normed vector space $(\ell^p, \|\cdot\|_{\ell^p})$ has as vectors the set of all sequences $u : \mathbb{N} \to \mathbb{R}$ with finite ℓ^p norm,

$$\|u\|_{\ell^p} = \left(\sum_{n=1}^{\infty} |u_n|^p \right)^{\frac{1}{p}} < \infty. \tag{1.45}$$

This space is equipped with the operations of addition and scalar multiplication of sequences. For $p = \infty$, the normed vector space $(\ell^\infty, \|\cdot\|_{\ell^\infty})$ consists of all sequences $u : \mathbb{N} \to \mathbb{R}$ with finite ℓ^∞ norm,

$$\|u\|_{\ell^\infty} = \sup_{n \in \mathbb{N}} |u_n| < \infty. \tag{1.46}$$

Remark 1.25 The definition of an ℓ^p space can be easily extended from sequences $u : \mathbb{N} \to \mathbb{R}$ to sequences with other domains and codomains, for instance $u : \mathbb{Z}_{\geq 0} \to \mathbb{R}$ or $u : \mathbb{N} \to \mathbb{R}^m$.

In these two examples, one would adjust the indices of the sum or trade the absolute value for a norm on $\mathbb{R}^m$ to accommodate the different domain or codomain of u. We'll return to the general case in Chapter 3.

As one can observe from the definition, the vectors of an ℓ^p space are *infinite sequences* of real numbers with bounded ℓ^p norm.

Definition 1.38 (L^p Space) Fix a number $p \in [1, \infty)$. The normed vector space $(L^p, \|\cdot\|_{L^p})$ consists of all functions $f : \mathbb{R} \to \mathbb{R}$ with finite L^p norm,

$$\|f\|_{L^p} = \left(\int_{\mathbb{R}} |f(x)|^p dx \right)^{\frac{1}{p}} < \infty \tag{1.47}$$

This space is equipped with the operations of function addition and scalar multiplication. For $p = \infty$, the normed vector space $(L^\infty, \|\cdot\|_{L^\infty})$ consists of all functions $f : \mathbb{R} \to \mathbb{R}$ with finite L^∞ norm,

$$\|f\|_{L^\infty} = \sup_{x \in \mathbb{R}} |f(x)| < \infty. \tag{1.48}$$

As opposed to ℓ^p spaces, where elements are real-valued sequences, in an L^p space, the elements are real-valued *functions* with bounded L^p norm. Despite the change from sequence to function, the ℓ^p and L^p norms have clear parallels in their definitions.

Remark 1.26 As with ℓ^p spaces, the definition of L^p spaces provided above is easily extended to more general domains and codomains (for example to functions taking values in $\mathbb{R}^n$). As with ℓ^p, we'll return to the general case in Chapter 3.

Remark 1.27 In defining L^p spaces as spaces of functions with finite integrals for $p \in [1, \infty)$, we've glossed over a number of concerns regarding Riemann/Lebesgue integrability of the functions f. Since properly treating these concerns requires a (significant & not immediately revealing) detour into a measure theory, we simply assume that elements of L^p spaces are "well-behaved" enough to have well-defined integrals. We also ignore the concerns that follow from functions differing on sets of measure zero having the same norm. We direct the interested reader to the additional reading specified at the end of the chapter for a rigorous treatment of these spaces.

In the previous section, we promised that we would return to finish the story of compactness once we defined Banach spaces. Now, we make good on this promise and provide a characterization of compactness in Banach space through convergent sequences. In order to state this characterization, we first require the concept of a *subsequence*. Given a sequence,

$$a_1, a_2, ..., a_n, ... \tag{1.49}$$

in a vector space V, a subsequence is a subset of the sequence in which elements are picked out in an order that respects that of the original sequence. For instance, a valid subsequence of the sequence above would be,

$$a_1, a_3, a_5, ... \tag{1.50}$$

since the terms of the subsequence appear in the same order as the original sequence. An *invalid* subsequence would then be,

$$a_2, a_1, a_3, a_2, ..., \tag{1.51}$$

which does *not* form a valid subsequence since the terms appear out of order compared to the original sequence. We abstractly define this "preservation of order" requirement in the following definition of a subsequence.

Definition 1.39 (Subsequence) Let V be a vector space and $\{a_n\} \subseteq V$ a sequence. A subsequence of $\{a_n\}$ is a subset $\{a_{n_k}\} \subseteq \{a_n\}$, where $n_k : \mathbb{N} \to \mathbb{N}$ is a strictly increasing sequence of indices,

$$n_1 < n_2 < n_3 < ..., \tag{1.52}$$

which specify the indices of the terms drawn from $\{a_n\}$.

For instance, for the subsequence $a_1, a_3, a_5, ...$, one would define n_k by $n_k = 2k - 1$. This produces the subsequence,

$$a_{n_1} = a_1,\ a_{n_2} = a_3,\ a_{n_3} = a_5, ..., \tag{1.53}$$

which is exactly the desired subsequence. Clearly, the indices of the subsequence are strictly increasing. Using the language of subsequences, one may formulate an equivalent definition of compactness in Banach spaces. Consider the following theorem, the proof of which is beyond our scope.

Theorem 1.5 (Compactness in Banach Space) *Let $(V, \|\cdot\|)$ be a Banach space and $K \subseteq V$. The following provide two equivalent characterizations of the compactness of K:*

1. *K is compact iff every sequence $\{a_n\} \subseteq K$ has a subsequence with a limit in K.*
2. *K is compact iff every sequence $\{a_n\} \subseteq K$ has a subsequence that is Cauchy.*

Remark 1.28 The abbreviation *"iff"* is commonly used as shorthand for "if and only if."

Thus, in Banach spaces, one can detect the compactness of a set K knowing only information about the sequences contained in K.

Exercise 1.28 Using Theorem 1.5, prove that a single-element subset $\{v\}$ of a Banach space (called a *singleton set*) is compact.

1.5 A Refresher on ODEs

Ordinary differential equations (ODEs) are the lingua franca of control in continuous time. In this section, we review some basic properties of scalar, linear differential equations. We postpone a formal discussion of the existence and uniqueness of solutions to such equations until Chapter 2, and only touch upon the most essential conceptual aspects here. As such, our treatment is significantly more informal than the previous sections of this chapter. For the reader concerned by the appalling lack of theorems - don't worry, plenty are coming down the pipeline - sit tight for a few more pages!

We begin by discussing initial value problems. Here, we'll keep our discussion to the case of ordinary differential equations in $\mathbb{R}$. Let $f : \mathbb{R} \to \mathbb{R}$ be a scalar function. The ordinary differential equation (ODE) governed by f is the equation,

$$\frac{d}{dt}x(t) = f(x(t)). \tag{1.54}$$

To find the *general solution* to the ordinary differential equation, one must identify all functions of the form $x : I \to \mathbb{R}$, where $I \subseteq \mathbb{R}$ is a nonempty open interval, satisfying $\frac{d}{dt}x(t) = f(x(t))$, for all $t \in I$.

Since it's a bit cumbersome to repeatedly write $\frac{d}{dt}$ one writes $\frac{d}{dt}x(t)$ in shorthand as $\dot{x}(t)$. Additionally, it's common to suppress the argument of $x(t)$. We therefore write the ordinary differential equation in shorthand as,

$$\dot{x} = f(x). \tag{1.55}$$

Now, we consider an *initial value problem* associated to an ordinary differential equation. Given a constant $x_0 \in \mathbb{R}$, the goal of the initial value problem is to solve the problem,

$$\frac{d}{dt}x(t) = f(x(t)),\ x(0) = x_0 \in \mathbb{R}. \tag{1.56}$$

That is, one wishes to find a curve $x : I \to \mathbb{R}$ (for I a nonempty, open interval), where $0 \in I$, satisfying $\frac{d}{dt}x(t) = f(x(t))$, for all $t \in I$ and $x(0) = x_0$. In our shorthand introduced above, one would abbreviate this problem as

$$\dot{x} = f(x),\ x(0) = x_0. \tag{1.57}$$

For a general nonlinear function f, initial value problems are challenging, and at worst impossible, to solve. A special case which has a well-defined solution is the scalar, linear initial value problem.

Theorem 1.6 (Scalar, Linear, First Order IVP) *Let $a \in \mathbb{R}$ be a fixed scalar. Consider the initial value problem,*

$$\dot{x} = ax,\ x(0) = x_0, \tag{1.58}$$

where $x_0 \in \mathbb{R}$ is a fixed initial condition. The unique solution $x : \mathbb{R} \to \mathbb{R}$ to this initial value problem is $x(t) = e^{at}x_0$.

Exercise 1.29 Verify that the solution proposed in Theorem 1.6 solves the initial value problem. You do not need to prove uniqueness - we'll return to this in Chapter 2.

Interestingly, this result states that, not only is $x(t) = e^{at}x_0$ a solution to the proposed initial value problem, but it is the *only* solution! That is, there is *no other function* satisfying the initial value problem. Further, the solution is defined on *all* of $\mathbb{R}$, rather than on some bounded, open interval. These properties are *not* shared by every initial value problem.

Example 1.21 Consider the initial value problem, $\dot{x} = 2\sqrt{|x|}$, $x(0) = 0$. For all $a \geq 0$, the function

$$x(t) = \begin{cases} (t-a)^2 & t \geq a \\ 0 & t < a, \end{cases} \tag{1.59}$$

is a solution of the initial value problem. Thus, the initial value problem has an *infinite* number of solutions.

Thus far, we've only considered *first order*, scalar ordinary differential equations. We can easily extend the definition of a differential equation to one that involves higher derivatives. In the following setup, we denote by $x^{(n)}(t)$ the n'th derivative $\frac{d^n x(t)}{dt^n}$. Let $f : \mathbb{R}^n \to \mathbb{R}$, and consider the ordinary differential equation,

$$x^{(n)}(t) = f(x(t), x^{(1)}(t), ..., x^{(n-1)}(t)). \tag{1.60}$$

Here, a solution of the differential equation is a function $x : I \to \mathbb{R}$ whose n'th derivative equals the function f of its first $n-1$ derivatives (by convention, the 0'th derivative of $x(t)$ is taken to be $x(t)$ itself).

Do we need to construct an entirely new theory for higher order differential equations? Fortunately, by lifting our problem from a *scalar* differential equation to a *vector* differential equation, we can transform any n'th order scalar differential equation into a system of n first order differential equations. Consider the following change of variables for the n'th order initial value problem specified above. Define,

$$q_0(t) = x(t) \tag{1.61}$$
$$q_1(t) = \dot{x}(t) \tag{1.62}$$
$$q_2(t) = \ddot{x}(t) \tag{1.63}$$
$$\vdots \tag{1.64}$$
$$q_{n-1}(t) = x^{(n-1)}(t). \tag{1.65}$$

Differentiating each of the q variables, one has

$$\dot{q}_0(t) = \dot{x}(t) = q_1(t) \tag{1.66}$$
$$\dot{q}_1(t) = \ddot{x}(t) = q_2(t) \tag{1.67}$$
$$\dot{q}_2(t) = x^{(3)}(t) = q_3(t) \tag{1.68}$$
$$\vdots \tag{1.69}$$
$$\dot{q}_{n-1}(t) = x^{(n)}(t) = f(x(t), ...x^{(n-1)}(t)). \tag{1.70}$$

We recognize that we can rewrite $f(x(t), ..., x^{(n-1)}(t))$ as $f(q_0(t), ..., q_{n-1}(t))$. Thus, the differential equation $x^{(n)}(t) = f(x(t), ..., x^{(n)}(t))$ can be rewritten as a *vector* differential equation in $\mathbb{R}^n$,

$$\frac{d}{dt}\begin{bmatrix} q_0(t) \\ \vdots \\ q_{n-1}(t) \end{bmatrix} = \begin{bmatrix} q_1(t) \\ \vdots \\ f(q_0(t), ..., q_{n-1}(t)) \end{bmatrix}. \tag{1.71}$$

Defining a vector $q = (q_0, ..., q_{n-1}) \in \mathbb{R}^n$ and a function $F : \mathbb{R}^n \to \mathbb{R}^n$ as $F(q) = (q_1, ..., f(q_0, ..., q_{n-1}))$, we compactly rewrite this differential equation in vector form as,

$$\dot{q} = F(q). \tag{1.72}$$

To solve the differential equation, we must identify a *vector* function $q : I \to \mathbb{R}^n$ satisfying $\dot{q}(t) = F(q(t))$ for all $t \in I$. To recover the solution to our original, scalar ODE, we the

extract the component function $q_0(t)$ of $q(t)$, which by definition equals the solution $x(t)$ to the original ODE.

Initial value problems are also defined similarly to the scalar case. In order to define an initial value problem, one specifies a *vector* initial condition, $q_0 \in \mathbb{R}^n$, to get the problem,

$$\dot{q} = F(q),\ q(0) = q_0. \tag{1.73}$$

Just as in the scalar case, the solution to the vector initial value problem is a function $q : I \to \mathbb{R}^n$ satisfying $\dot{q}(t) = F(q(t))$ for all $t \in I$ and $q(0) = q_0$. The transformation from an n'th order scalar ODE into a first order vector ODE tells us that it's sufficient just to develop a theory for first order, vector ODEs to study general ODEs. We resume this story in the next chapter!

1.6 Further Reading

For an abstract treatment of linear algebra, we refer the reader to the texts *Linear Algebra* by Friedberg, Insel, & Spence [13], and *Linear Algebra Done Right* by Sheldon Axler [5]. For a user-friendly introduction to real analysis in $\mathbb{R}$, we recommend *Understanding Analysis* by Stephen Abbott [1]. For a similarly user-friendly treatment of analysis in normed vector spaces & Banach spaces, we refer the reader to *Measure, Integration, & Real Analysis* by Sheldon Axler [4]. Here, the reader can find proofs of a number of the concepts treated in this section, as well as a rigorous treatment of L^p spaces, enabled by measure theory.

1.7 Problems

Problem 1.1 (Consequences of Norm Equivalence) In this problem, we'll consider some simple consequences of norm equivalence. Consider a vector space V with two equivalent norms, $\|\cdot\|_a$ and $\|\cdot\|_b$.

1. Show that a sequence $\{v_n\} \subseteq V$ converges with respect to $\|\cdot\|_a$ if and only if it converges with respect to $\|\cdot\|_b$.
2. Show that a mapping $f : V \to V$ is continuous with respect to norm $\|\cdot\|_a$ if and only if it is continuous with respect to $\|\cdot\|_b$. Does the same property hold for Lipschitz continuity?

Problem 1.2 (Unbounded Linear Operators) We know that every linear transformation between finite-dimensional normed vector spaces is bounded. In infinite dimensions, we're not quite so lucky! Product an example of an unbounded linear transformation from $\ell^2 \to \ell^2$. *Hint: The ℓ^2 norm is defined as an infinite series - think about some series that converge, and how they can be linear modified to no longer converge. It helps to work with the square of the ℓ^2 norm here.*

Problem 1.3 ($(\mathbb{R}^n, \|\cdot\|)$ is a Banach Space) One may show that in $\mathbb{R}$, a sequence converges (with respect to the absolute value norm) if and only if it is Cauchy. That is, $(\mathbb{R}, |\cdot|)$ is a Banach space. In this problem, we'll show that for any norm $\|\cdot\|$ on $\mathbb{R}^n$, $(\mathbb{R}^n, \|\cdot\|)$ is a Banach space - this is a special case of the result that *every* finite-dimensional normed vector space is Banach.

1. Consider a sequence $\{v_k\} \subseteq \mathbb{R}^n$. Let $\{v_k^i\} \subseteq \mathbb{R}$ represent the sequence formed from the i'th components of each $v_k \in \mathbb{R}^n$ (i.e. $v_k = (v_k^1, v_k^2, ..., v_k^n)$). Show that the sequence $\{v_k\}$ converges to a vector $v \in \mathbb{R}^n$ with respect to the ℓ^∞ norm on $\mathbb{R}^n$ if and only if each component sequence $\{v_k^i\}$ converges to v^i in $\mathbb{R}$.
2. Show that $\{v_k\} \subseteq \mathbb{R}^n$ is a Cauchy sequence with respect to the ℓ^∞ norm on $\mathbb{R}^n$ if and only if each component sequence $\{v_k^i\}$ is Cauchy in $\mathbb{R}$.
3. Using norm equivalence on $\mathbb{R}^n$, show that for any norm on $\mathbb{R}^n$, a sequence is Cauchy if and only if it is convergent.

Problem 1.4 (The Space of Polynomials) Consider the set $\mathcal{P}$ consisting of all polynomials (of all finite degrees) $p : I \to \mathbb{R}$ on a compact interval $I \subseteq \mathbb{R}$.

1. Show that $(\mathcal{P}, \|\cdot\|_\infty)$, where $\|\cdot\|_\infty$ is the sup norm, $\|p\|_\infty = \sup_{t\in I} |p(t)|$, is a normed vector space.
2. Is $(\mathcal{P}, \|\cdot\|_\infty)$ a Banach space? Explain why or why not. *Hint: think about Taylor series.*

Problem 1.5 (Systems of First Order Equations)

1. Show that an n'th order linear ODE,

$$\frac{d^n x}{dt^n} + a_{n-1}\frac{d^{n-1}x}{dt^n} + ... + a_1\frac{dx}{dt} + a_0 x = 0,\ a_i \in \mathbb{R}, \tag{1.74}$$

   can be rewritten as a system of n, first order differential equations of the form,

$$\dot{z} = Az, \tag{1.75}$$

   where $z \in \mathbb{R}^n$ and $A \in \mathbb{R}^{n\times n}$. This tells us that it's sufficient to examine *linear systems of first order ODEs* in order to reach conclusions about linear n'th order ODEs.
2. Show that an n'th order recurrence,

$$x[k+n] + a_{n-1}x[k+n-1] + ... + a_1 x[k+1] + a_0 x[k] = 0, \tag{1.76}$$

   can be rewritten as a system of n, first order recurrences of the form,

$$z[k+1] = Az[k], \tag{1.77}$$

   where $z \in \mathbb{R}^n$ and $A \in \mathbb{R}^{n\times n}$. This tells us that it's sufficient to examine *linear systems of first order recurrences* in order to reach conclusions about linear n'th order recurrences.

Problem 1.6 (The Structured Singular Value ★★) The (complex) structured singular value is a function from the set of $n \times n$ complex matrices to the reals that helps us understand the "gain" of matrices with structured uncertainty. The first step towards defining the structured singular value is to define a set of matrices $\underline{\Delta} \subseteq \mathbb{C}^{n\times n}$. Let $r_1, ..., r_S$ and $m_1, ..., m_F$ be positive integers for which $\sum_{i=1}^S r_i + \sum_{j=1}^F m_j = n$. Then, define a set $\underline{\Delta} \subseteq \mathbb{C}^{n\times n}$ as

$$\underline{\Delta} := \{\text{blkdiag}(\delta_1 I_{r_1}, ..., \delta_S I_{r_S}, \Delta_{S+1}, ..., \Delta_{S+F}) : \delta_i \in \mathbb{C},\ \Delta_{s+j} \in \mathbb{C}^{m_j\times m_j}\}, \tag{1.78}$$

where I_k represents the $k\times k$ identity matrix. In short, $\underline{\Delta}$ is the set of block diagonal matrices with *repeated scalar blocks* of dimensions $r_i \times r_i$ (these are the blocks $\delta_i I_{r_i}$) and and *full blocks* of dimensions $m_j \times m_j$ (these are the blocks Δ_{S+j}). Given a matrix $M \in \mathbb{C}^{n\times n}$ and

a set $\underline{\Delta} \subseteq \mathbb{C}^{n\times n}$ of the form above, one defines the structured singular value of M, $\mu_{\underline{\Delta}}(M)$, as follows.

Definition 1.40 (Structured Singular Value) For $M \in \mathbb{C}^{n\times n}$, $\mu_{\underline{\Delta}}(M)$ is defined,

$$\mu_{\underline{\Delta}}(M) := \frac{1}{\inf\{\overline{\sigma}(\Delta) : \Delta \in \underline{\Delta} \text{ and } \det(I - M\Delta) = 0\}}, \tag{1.79}$$

unless no $\Delta \in \underline{\Delta}$ makes $I - M\Delta$ singular, in which case $\mu_{\underline{\Delta}}(M) := 0$.

Note that here, we use $\overline{\sigma}(M)$ to denote the maximum singular value of M. Based on this definition, $\mu_{\underline{\Delta}}(M)$ depends both on M and on the set $\underline{\Delta}$. Now, let's get started on our analysis of $\mu_{\underline{\Delta}}$! *Note: In the following problems, you can assume for simplicity that one does not encounter the case where no Δ makes $I - M\Delta$ singular.*

1. Compute $\mu_{\underline{\Delta}}(M)$ in the case where $\underline{\Delta}$ is *unstructured*, i.e. $\underline{\Delta} = \mathbb{C}^{n\times n}$.
2. Recall that the spectral radius of a matrix $M \in \mathbb{C}^{n\times n}$ is defined,

$$\rho(M) := \max_i |\lambda_i(M)|. \tag{1.80}$$

   Define the set $B_{\underline{\Delta}} = \{\Delta \in \underline{\Delta} : \overline{\sigma}(\Delta) \leq 1\}$. Prove that the structured singular value can be calculated as the following function of spectral radius:

$$\mu_{\underline{\Delta}}(M) = \sup_{\Delta \in B_{\underline{\Delta}}} \rho(\Delta M). \tag{1.81}$$

   Now, consider the special case where $\underline{\Delta} = \{\delta I_n : \delta \in \mathbb{C}\}$. In this case, show that $\mu_{\underline{\Delta}}(M) = \rho(M)$.
3. Let's consider some additional methods of computing μ. Define the following subset of $\mathbb{C}^{n\times n}$:

$$\underline{D} = \{\text{blkdiag}(D_1, ..., D_S, d_{S+1}I_{m_1}, ..., d_{S+F}I_{m_F}) : D_i \in \mathbb{C}^{r_i\times r_i}, D_i \succ 0, d_{S+j} \in \mathbb{R}_{>0}\}. \tag{1.82}$$

   Prove that, for all $D \in \underline{D}$,

$$\mu_{\underline{\Delta}}(M) = \mu_{\underline{\Delta}}(D^{\frac{1}{2}}MD^{-\frac{1}{2}}). \tag{1.83}$$

   Then, show that,

$$\mu_{\underline{\Delta}}(M) \leq \inf_{D\in\underline{D}} \overline{\sigma}(D^{\frac{1}{2}}MD^{-\frac{1}{2}}). \tag{1.84}$$

4. Fix a matrix $M \in \mathbb{C}^{n\times n}$. For the set $\underline{D}$ introduced in part (3), show that the following set is convex for each fixed $\beta \in \mathbb{R}$:

$$\{D \in \underline{D} : \overline{\sigma}(D^{\frac{1}{2}}MD^{-\frac{1}{2}}) \leq \beta\}. \tag{1.85}$$

   Hint: Rewrite as a linear matrix inequality. Such inequalities are amenable to implementation in convex optimization solvers!

Chapter 2
Linear Dynamical Systems

In this chapter, we begin in earnest our study of systems and control theory. First, we introduce the main players in linear systems theory: linear dynamical systems and their state space representations. Following this, we study solutions to state space representations of linear systems, and see how the state space representations generate formal linear dynamical systems. Then, we'll move on to study linear systems from the I/O perspective, developing the theory of Laplace and $\mathcal{Z}$-transforms along the way. Let's begin!

2.1 Dynamical Systems & State Space Models

In order to develop a precise, mathematical theory of systems and control, it's vital that we understand what control systems actually are. Let's begin with a simple, motivating example. Suppose I have a shower with a handle I can use to control the water temperature. If I stick my hand into the water, I can gain some information as to whether the temperature is too high or too low, and I can adjust the position of the handle accordingly. I can continue to repeat this process - move handle, touch water, correct handle position - until the shower is at a temperature I want. If I continually take measurements and adjust the handle position, I can also adapt to unanticipated changes in the environment, such as my roommates washing dishes and reducing the supply of hot water. In other words, by adapting my actions according to measurements, I can become *robust* to changes in the shower environment.

Fundamentally, this is an example of a *feedback control system*, a system with an input (the position of the handle), a measurement (the temperature of the water that I estimate with my hand), a state (the *true* temperature of the water), internal dynamics (how the handle position affects the water temperature), and a description of time (the number of seconds that have passed). By incorporating feedback from the environment into my actions, I can *control* the system to reach a desired temperature.

This is the fundamental idea of feedback control: by measuring the environment, we can make informed decisions that enable us to control the state of the environment. Additionally, by taking repeated measurements, we can make decisions that adapt to unexpected events that might occur. Thus, in addition to affording us the ability of control over our environment, feedback yields the potential to be *robust* to uncertainty and disturbances in the environment.

2.1.1 Causal Input/Output Dynamical Systems

In order to develop the mathematical foundations of systems and control theory, we first need a precise definition for a *system*. Let's distill the most essential components of the shower system to gain some insight into the problem of determining a precise definition for an abstract, mathematical system. First, let's focus on the different objects making up the shower system. For simplicity, we'll assume that we have an infinite supply of hot water, and that the temperature of the shower is entirely determined by its current temperature, the time that has passed, and the history of shower handle positions. The key objects of this simple shower system are the following.

1. Time: we know the time of day at which we entered the shower, and the current time of day. We can represent both the entry time and the current time with a real number, $t \in \mathbb{R}$, corresponding to the number of seconds (or any other appropriate unit of time) that have passed since the beginning of the day.
2. Inputs: the shower handle was an *input* to the shower system. Input signals to the system composed of different positions of the shower handle over time. We can measure the values of inputs by the angle, $\theta(t) \in \mathbb{R}$, of the shower handle at time t. An input signal would therefore be a function of time, $\theta(\cdot) : \mathbb{R} \to \mathbb{R}$, assigning to each time a shower handle position.
3. Outputs: the outputs (measurements) we took of the shower system were tests of the water temperature with our hand. Since we only took measurements with our hand, and not a thermometer, the measurements of our system might take on values in a set,

$$\{\text{icy, cold, mild, hot, ouch!}\} \tag{2.1}$$

   Measurement *signals* would then be mappings from time, $t \in \mathbb{R}$, to this set of measurement values.
4. State: we know that our set of measurements, {icy, cold, mild, hot, ouch!} doesn't quite cover the actual temperature of the system! We define the *state* of the shower to be the *actual* water temperature, $T \in \mathbb{R}$, measured in degrees Celsius (or any other appropriate unit of temperature). This describes the entire state of the shower at a given time t. Using knowledge of the state, T, time, t, and input, we should be able to *completely determine* what the shower will do next (within the scope of our very simple shower model).

How are all of these basic objects tied together? Underneath the shower system, we know there exist some *shower dynamics* that determine how the temperature of the shower changes according to the passage of time and the history of shower handle inputs. Additionally, we know there is some underlying map that determines which measurement value out of the set {icy, cold, mild, hot, ouch!} we will feel given any true temperature and time. These concepts are encoded in the following two maps.

1. State Transition Map: the state transition map of the shower determines how the state (true temperature) of the shower is influenced by the start time, current time, starting temperature of the water, and history of shower handle positions. This gives us a *complete description* of how the true temperature of the shower changes over time. Notably, the current state of the system only depends on the previous and current shower handle positions! The state does not depend on the future inputs to the system (our shower is unfortunately not fancy enough to predict the future).

2. Readout Map: given any time, temperature, and input value, we should know what measurement value our hand is feeling. The readout map maps from a pair of time, temperature, and shower handle position to this measurement value. Notice that the readout map is *memoryless* - it does not require a history of temperatures or a full input signal, only the current temperature and the current input value!

Finally, we know there are a couple of simple properties all of these objects should obey.

1. Time: we can add and subtract time in the shower system without any confusion.
2. Restriction: suppose we have two input signals to the system which match from times t_0 to t_1. Over the time period t_0 to t_1, these two input signals should produce the same behavior, regardless of if they differ after time t_1.
3. Composition: suppose over the time period t_0 to t_1, an input signal takes us from temperature T_0 to temperature T_1, and over the time period t_1 to t_2, an input signal takes us from temperature T_1 to temperature T_2. Then, applying the input signals from t_0 to t_2 should take us from T_1 to T_2.
4. Identity: if no time passes, the temperature of the shower should stay the same.

When stated in context of the shower example, these three conditions are all fairly "obvious." Although it may seem like these points are too trivial to mention, they'll help us make a well-posed, abstract definition of a system that behaves in the way we expect.

This exploratory example provides us with a template definition for a formal *causal input/output dynamical system*. In order to state the formal definition, all we need to do is abstract away the details of the shower into the language of mathematics. As you're reading the definition, relate the formal mathematical expressions to the analogous components of the shower system we outlined above.

Note that, in our formal definition, everything is named similarly to the shower example with the one exception: the "composition" property has been given the shiny new name of the "semigroup axiom" - we'll discuss the rationale behind this after stating the definition.

Definition 2.1 (Causal Input/Output Dynamical System) Let $\mathcal{T} \subseteq \mathbb{R}$ be a nonempty set. A causal, input/output dynamical system on $\mathcal{T}$ is a tuple $\mathcal{D} = (\mathcal{U}, \mathcal{Y}, \Sigma, \varphi, r)$. Each term is defined as follows:

1. Time set: $\mathcal{T}$ is the time set, a subset of $\mathbb{R}$ describing the possible times in the system.
2. Input space: $\mathcal{U}$ is the input space, a set of mappings from $\mathcal{T}$ to a fixed set U,

$$\mathcal{U} \subseteq \{u : \mathcal{T} \to U\}. \tag{2.2}$$

   U is referred to as the input value space. Elements of $\mathcal{U}$ represent the possible input signals to the system.
3. Output space: $\mathcal{Y}$ is the output space, a set of mappings from $\mathcal{T}$ to a fixed set Y,

$$\mathcal{Y} \subseteq \{y : \mathcal{T} \to Y\}. \tag{2.3}$$

   Y is referred to as the output value space. Elements of $\mathcal{Y}$ represent the possible output (measurement) signals of the system.
4. State space: Σ is the state space, a set representing the possible states of the system.
5. State transition map: φ is the state transition map, a map,

$$\varphi : \mathbf{T} \times \Sigma \times \mathcal{U} \to \Sigma, \tag{2.4}$$

where $\mathbf{T} = \{(t_1, t_0) \in \mathcal{T} \times \mathcal{T} : t_1 \geq t_0\}$, which describes how the state of the system evolves. In particular, for $t_0, t_1 \in \mathcal{T}$ with $t_0 \leq t_1$, $x_0 \in \Sigma$, and $u(\cdot) \in \mathcal{U}$, $\varphi(t_1, t_0, x_0, u(\cdot))$ returns the state of the system at time t_1 after starting from x_0 at time t_0 and applying input signal $u(\cdot)$.

6. Readout map: r is the readout map, a map,

$$r : \mathcal{T} \times \Sigma \times U \to Y, \tag{2.5}$$

which returns the measured output at time t given the system has a current state of $x(t) \in \Sigma$ and a current input value of $u(t) \in U$.

A dynamical system $\mathcal{D}$ must additionally satisfy the following four axioms:

1. Time axiom: for all $t_1, t_2 \in \mathcal{T}$, $t_1 + t_2 \in \mathcal{T}$ and $t_1 - t_2 \in \mathcal{T}$.
2. Restriction axiom: for all $t_0, t_1 \in \mathcal{T}$ with $t_0 \leq t_1$, $x_0 \in \Sigma$, and $u_1(\cdot), u_2(\cdot) \in \mathcal{U}$, one has that

$$u_1(t) = u_2(t)\ \forall t \in [t_0, t_1] \cap \mathcal{T} \Longrightarrow \varphi(t_1, t_0, x_0, u_1(\cdot)) = \varphi(t_1, t_0, x_0, \tilde{u}_2(\cdot)). \tag{2.6}$$

3. Semigroup axiom: for all $t_0, t_1, t_2 \in \mathcal{T}$ with $t_0 \leq t_1 \leq t_2$, $x_0 \in \Sigma$, and $u(\cdot) \in \mathcal{U}$,

$$\varphi(t_2, t_1, \varphi(t_1, t_0, x_0, u(\cdot)), u(\cdot)) = \varphi(t_2, t_0, x_0, u(\cdot)). \tag{2.7}$$

4. Identity axiom: for all $t \in \mathcal{T}$, $x \in \Sigma$, and $u(\cdot) \in \mathcal{U}$,

$$\varphi(t, t, x, u(\cdot)) = x. \tag{2.8}$$

Phew, what a mouthful! Let's highlight some subtle yet important consequences of the definition.

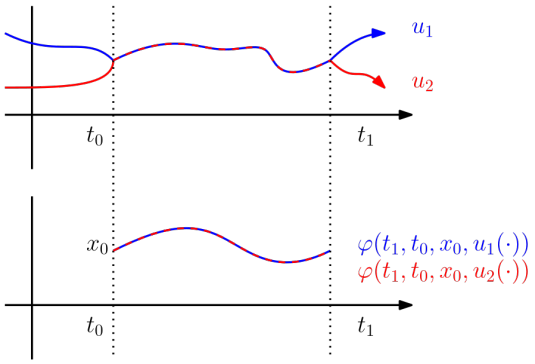


Fig. 2.1 The restriction axiom states that if two input signals are equal on a time interval, they will produce the same state behavior on that time interval.

Remark 2.1 (Signal versus Value) In Definition 2.1, we define the input space $\mathcal{U}$ and the output space $\mathcal{Y}$ to be spaces of *signals*, not spaces of values! Associated to the space of input signals $\mathcal{U}$, we have the space of input values, U. Likewise, associated to the space of output signals $\mathcal{Y}$, we have the space of output values, Y. To make this concrete, the space of input

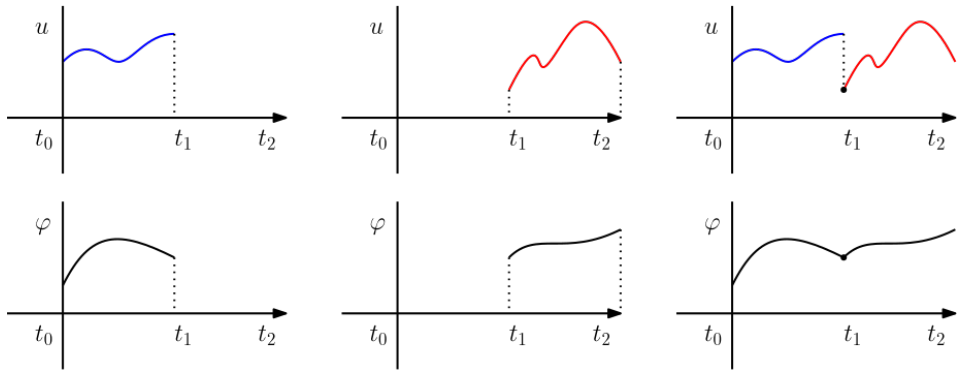


Fig. 2.2 The semigroup axiom states that the state transition map is well-behaved under *composition*. The semigroup axiom implies that, if we stitch together two inputs, the resulting behavior is the same as that which results from applying the first input over its domain and the second input over its domain.

values might be $U = \mathbb{R}^m$, but the space of input signals might be the set of continuous maps from $\mathbb{R} \to \mathbb{R}^m$. To distinguish between the two, we will write $u(\cdot)$ to represent a signal and $u(t)$ to represent its value at time t.

Remark 2.2 (Causality) A system is said to be *causal* if its state and output behavior depends only on its previous and current inputs, *not* on its future inputs! As we can see from the definition of the state transition map, causality is *baked into* the definition of a causal dynamical system. The restriction axiom tells us that the state at time t_1 *only* depends on the state x_0 at time t_0 and the input signal $u(\cdot)$ applied from t_0 to t_1. Thus, any values of the input signal after time t_1 are entirely irrelevant to the behavior of the system. We conclude that any dynamical system satisfying Definition 2.1 must be causal.

Remark 2.3 (Time Axiom) The time axiom can also be stated in terms of *algebraic* language. One may equivalently require $\mathcal{T}$ to be a *subgroup* of $(\mathbb{R}, +)$, the group of real numbers with the addition operation. If you're unfamiliar familiar with algebraic language, this connection isn't something you need to worry about - there are no practical differences between this and what we stated in Definition 2.1.

Remark 2.4 (Semigroup Axiom) Why rename the "composition" axiom as the semigroup axiom? The name *semigroup* alludes to another connection between causal I/O dynamical systems and abstract algebra. If you're interested, read the definition of a semigroup and a semigroup action and see if you can draw a connection between a algebraic semigroups/semigroup actions and the semigroup axiom of Definition 2.1.

Remark 2.5 (Generality) This is but one of many different definitions of an I/O dynamical system, and is by no means the most general definition possible. For instance, the systems proposed above are causal, deterministic (not random), and have fixed input and readout spaces that do not change with state or time. Further, the state transition map is defined on the entire time set, rather than on subsets thereof. Although these assumptions are sufficiently general for our purposes in this course, one should keep these limitations in mind! We direct the reader to the references at the end of the chapter for the more general definitions.

To get some practice with identifying the components of a dynamical system, try the following three exercises. Make sure to state your assumptions where necessary; in each case, you'll need to lay out some simplifying assumptions in order to come up with a manageable dynamical system.

Exercise 2.1 Come up with a dynamical system representing a falling rock. Specify each component of the tuple $\mathcal{D} = (\mathcal{U}, \mathcal{Y}, \Sigma, \varphi, r)$, as well as the time set $\mathcal{T}$. Explain why the time, restriction, semigroup, and identity axioms hold for this system.

Exercise 2.2 Come up with a dynamical system representing an airplane. Specify each component of the tuple $\mathcal{D} = (\mathcal{U}, \mathcal{Y}, \Sigma, \varphi, r)$, as well as the time set $\mathcal{T}$. Explain why the time, restriction, semigroup, and identity axioms hold for this system.

Exercise 2.3 Come up with a dynamical system representing a computer. Specify each component of the tuple $\mathcal{D} = (\mathcal{U}, \mathcal{Y}, \Sigma, \varphi, r)$, as well as the time set $\mathcal{T}$. Explain why the time, restriction, semigroup, and identity axioms hold for this system.

These three examples - of a falling rock, an airplane, and a computer - illustrate the flexibility of the dynamical system definition we posed above. Each system is vastly different, yet fits into the same framework under minimal assumptions.

However, this flexibility comes at a price. Whenever one makes a highly abstract, general definition such as Definition 2.1, there is a fundamental tradeoff that one makes. Almost always, generality comes at the expense of *practicality* - the more general the definition, the less practical it is, and the harder it is to come to interesting conclusions. In order to state more interesting results about the behavior of dynamical systems, we'll need to consider dynamical systems with more structure than the basic scaffolding offered by Definition 2.1.

First, we outline two simple classes of dynamical system, based on the examples of an airplane and a computer. In order to properly describe the behavior of an airplane, one needs to use a time set $\mathcal{T} = \mathbb{R}$. On the other hand, for a computer—which makes decisions at discrete instants—one might use the time set $\mathcal{T} = \mathbb{Z}$. We distinguish between systems on these two important time sets as follows.

Definition 2.2 (Continuous-Time System) A dynamical system $\mathcal{D}$ is said to be a continuous-time system if its time set is $\mathcal{T} = \mathbb{R}$.

Definition 2.3 (Discrete-Time System) A dynamical system $\mathcal{D}$ is said to be a discrete-time system if its time set is $\mathcal{T} = \mathbb{Z}$.

As a general rule of thumb, if the state of a system varies as time passes in seconds (with no fixed jumps or increments in time), the system will be continuous-time. On the other hand, if the state of a system jumps at fixed, discrete increments, the system will be discrete-time[1].

These two classes of system provide some nice, additional structure to Definition 2.1. What other structure might be interesting to add? Let's think back to the example of a falling rock, and see if anything else jumps out at us that isn't explicitly covered by Definition 2.1.

Suppose the state of the rock is its position and velocity in space and that its output is its position. As an input, let's take a force acting on the rock. This enables us to treat the action of "dropping" the rock as an input signal. To track the trajectory of the rock after

[1] Note that the choice of $\mathcal{T} = \mathbb{Z}$ for discrete-time is somewhat arbitrary - one could reasonably replace $\mathbb{Z}$ with another countable subset of $\mathbb{R}$ that satisfies the time axiom.

we drop it, we could check the readout map. This tells us that, if we drop the rock at time t_0 from initial state x_0, the rock position at time t is,

$$\text{rock position at time } t = r(t, \varphi(t, t_0, x_0, u(\cdot)), u(t)). \tag{2.9}$$

What if, instead of dropping the rock at time t_0, we sat around for ten minutes and dropped the rock at time $t_0' = t_0 + 10$? Would we expect the rock to fall in the same way? If our rock is friends with Isaac Newton, the answer is—of course! To determine the trajectory of the rock, we shouldn't need to know the number of seconds since the beginning of time at which we drop it—what matters is how much time has *passed* since we dropped it. This behavior—of time *passed* being the relevant quantity of time—is prevalent in a number of systems. A dynamical system possessing this property is said to be *time-invariant*.

Before we can properly define time invariance, we must make a few auxiliary definitions. First, we make a definition that will reduce our notational overhead. We note that in the example of the falling rock, we composed the readout map with the state transition map in order to get the output at time t, given an initial time, initial state, and input signal. Since it's quite cumbersome to rewrite this composition every time we're interested in these objects, we define the following map.

Definition 2.4 (Input/Output Map) Given a dynamical system $\mathcal{D}$, the input/output map (I/O map) is the map $\rho : \mathbf{T} \times \Sigma \times \mathcal{U} \to Y$ (for $\mathbf{T} = \{(t_1, t_0) \in \mathcal{T} \times \mathcal{T} : t_1 \geq t_0\}$) defined as the composition,

$$\rho(t_1, t_0, x_0, u(\cdot)) = r(t_1, \varphi(t_1, t_0, x_0, u(\cdot)), u(t_1)), \tag{2.10}$$

of the readout and the state transition maps.

Remark 2.6 An I/O map takes in a pair of times, an initial state, and an input signal and returns an output *value*, not an output signal! This is because we want the I/O map to convey information about the output at a particular time, rather than at all times.

With this definition made, we turn our attention back to the problem of defining time-invariance. First, we focus on how the components of a dynamical system change over time. We define a *delay-invariant* set of signals.

Definition 2.5 (Delay-Invariant Set) Consider a set of signals, $\mathcal{U} \subseteq \{u : \mathcal{T} \to U\}$, where $\mathcal{T} \subseteq \mathbb{R}$ is a time set and U is an arbitrary set. If, for all $\tau \in \mathcal{T}$ and all $u(\cdot) \in \mathcal{U}$, the signal

$$\hat{u} : \mathcal{T} \to U,\ \hat{u}(t) = u(t - \tau), \tag{2.11}$$

also belongs to $\mathcal{U}$, then $\mathcal{U}$ is said to be a *delay-invariant set* with respect to $\mathcal{T}$.

Remark 2.7 When the time set $\mathcal{T}$ is clear from context, one refers to a delay-invariant set with respect to $\mathcal{T}$ simply as a "delay-invariant set." The "with respect to $\mathcal{T}$" can be dropped.

Remark 2.8 Delay-invariant sets are so-called since they are defined by delaying signals by a time τ. It's important to note that if $\tau < 0$, a "delay" will actually shift a signal *forward* in time rather than shifting it backward. In this context, the name "delay" is therefore not entirely consistent with our intuitive understanding of the word.

Remark 2.9 In this definition, we implicitly make use of the time axiom. Without the guarantee that $t_1 - t_2 \in \mathcal{T}\ \forall t_1, t_2 \in \mathcal{T}$, we would not know $\hat{u}(t) = u(t - \tau)$ was a valid signal.

Thus, a set is delay-invariant if any signal in $\mathcal{U}$ can be delayed by any time τ and remain in $\mathcal{U}$. Equipped with this definition, we define a *delay map* on a delay-invariant set of signals.

Definition 2.6 (Delay Map) Consider a delay-invariant set of signals, $\mathcal{U}$. For $\tau \in \mathcal{T}$, the map $T_\tau : \mathcal{U} \to \mathcal{U}$, defined $(T_\tau(u))(t) = u(t-\tau)\ \forall t \in \mathcal{T}$, is called the delay map of time τ.

Based on this definition, a shift map T_τ simply *delays* any input signal by a fixed time τ. Notice how the definition of a delay-invariant set ensures the delay map is well-defined—since we don't have to worry about a delayed signal leaving the set $\mathcal{U}$, we define the delay map to be a map from $\mathcal{U} \to \mathcal{U}$.

Definition 2.7 (Time-Invariant System) A causal I/O dynamical system $\mathcal{D}$ is said to be time invariant if, for all $\tau \in \mathcal{T}$, the following conditions are satisfied:

1. Delay-invariant input space: $\mathcal{U}$ is a delay-invariant set.
2. Delay-invariant output space: $\mathcal{Y}$ is a delay-invariant set.
3. Delay-invariant transition map: For all $t_0, t_1, \tau \in \mathcal{T}$ with $t_0 \leq t_1$ and all $x_0 \in \Sigma$, $u(\cdot) \in \mathcal{U}$,

$$\rho(t_1, t_0, x_0, u(\cdot)) = \rho(t_1 + \tau, t_0 + \tau, x_0, T_\tau(u(\cdot))), \tag{2.12}$$

where $T_\tau : \mathcal{U} \to \mathcal{U}$ is the delay map of time τ on $\mathcal{U}$ and ρ is the I/O map of $\mathcal{D}$.

Item (3) of this definition—delay-invariant transition map—is by far the most important component of the definition. It states that the output of the system depends on the amount of time that has *passed*, rather than on the explicit start and end times. In particular, if we delay the inputs to the system by time τ, we will get the same output at time $t + \tau$ as the undelayed system at time t.

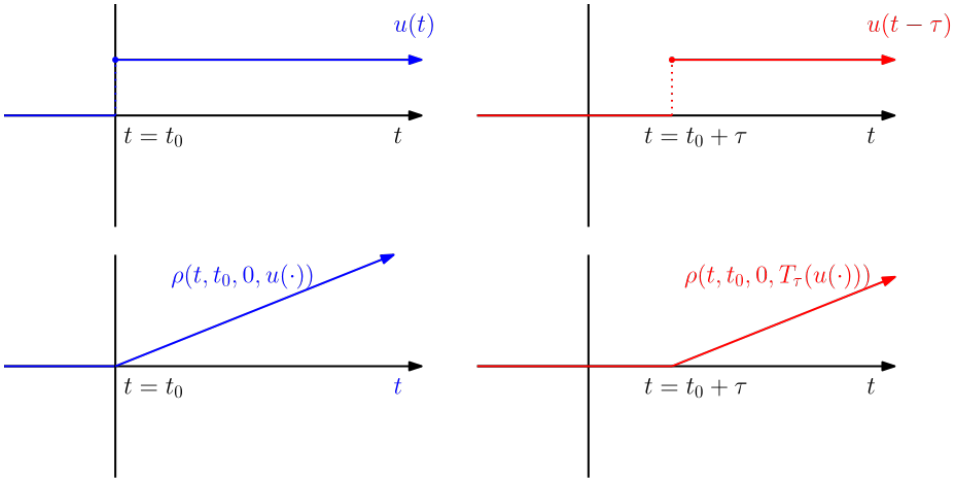


Fig. 2.3 An example of a time-invariant response. On the left-hand side, we apply an input which jumps up at time $t = t_0$. The system responds by increasing along a ramp at time $t = t_0$. If we *delay* the input by τ, the appearance of the ramp also delays by τ. Thus, we observe that, for this input, the system respects the equality $\rho(t + \tau, t_0 + \tau, T_\tau(u(\cdot))) = \rho(t, t, t_0, u(\cdot))$.

What other structure can we add to a causal I/O dynamical system? Thus far, we've only placed structure on the *time* component of the I/O map, and haven't considered any algebraic or analytic conditions.

Although algebraically and analytically unstructured maps lend themselves well to generality, one cannot say the same for practicality. Without placing further algebraic or analytic constraints on the I/O map, we'll find it hard to perform any meaningful system analysis. For the class of *linear* I/O systems, however, a wide array of concepts become mathematically and computationally tractable. This is the class of systems we will focus on in this course.

Definition 2.8 (Linear I/O System) Consider an I/O system $\mathcal{D} = (\mathcal{U}, \mathcal{Y}, \Sigma, \varphi, r)$ with I/O map ρ. $\mathcal{D}$ is said to be a linear I/O system if the following conditions are satisfied:

1. Linear Spaces: $\mathcal{U}, \mathcal{Y}$, and Σ are vector spaces over a common field, $\mathbb{K} \in \{\mathbb{R}, \mathbb{C}\}$.
2. Linear I/O Map: For each fixed pair $t_0, t_1 \in \mathcal{T}$ with $t_0 \leq t_1$, the I/O map is linear in $\Sigma \times \mathcal{U}$. That is, for all $x_0, \hat{x}_0 \in \Sigma$, $u(\cdot), \hat{u}(\cdot) \in \mathcal{U}$, and $\alpha, \beta \in \mathbb{K}$,

$$\rho(t_1, t_0, \alpha x_0 + \beta \hat{x}_0, \alpha u(\cdot) + \beta \hat{u}(\cdot)) = \alpha \rho(t_1, t_0, x_0, u(\cdot)) + \beta \rho(t_1, t_0, \hat{x}_0, \hat{u}(\cdot)). \tag{2.13}$$

Remark 2.10 If the spaces $\mathcal{U}, \mathcal{Y}, \Sigma$ of $\mathcal{D}$ are all over a field $\mathbb{K}$, one says that $\mathcal{D}$ itself is a system over a field $\mathbb{K}$.

Remark 2.11 We'll refer to a linear I/O system simply as a "linear system" or a"linear dynamical system" where context allows.

Let's discuss the different components of the definition. The first condition, *linear spaces*, states that the input and output *signal* spaces are vector spaces, as is the state space Σ. This means that any linear combination of input signals, $\alpha u(\cdot) + \beta \hat{u}(\cdot)$, remains in the input space. Likewise, linear combinations of output signals and states remain in the output and state spaces, respectively.

The second condition, *linear I/O map*, states that the output of a linear I/O system must be linear in its initial condition and input. That is, if we scale the initial condition and output by the same value, the output should scale by that value as well. Additionally, if we add two sets of initial conditions and inputs, the corresponding output should be the sum of the individual outputs.

Let's get a basic feel for what this linear structure enables. In the following proposition, we state a few simple consequences of Definition 2.8.

Proposition 2.1 (Output Response of Linear I/O Systems) *Any linear I/O system $\mathcal{D}$ over a field $\mathbb{K}$ satisfies the following:*

1. *Zero Input Response: For all $x_0, \hat{x}_0 \in \Sigma$, $t_0, t_1 \in \mathcal{T}$ with $t_0 \leq t_1$, and $\alpha, \beta \in \mathbb{K}$,*

$$\rho(t_1, t_0, \alpha x_0 + \beta \hat{x}_0, 0) = \alpha \rho(t_1, t_0, x_0, 0) + \beta \rho(t_1, t_0, \hat{x}_0, 0). \tag{2.14}$$

2. *Zero State Response: For all $u(\cdot), \hat{u}(\cdot) \in \mathcal{U}$, $t_0, t_1 \in \mathcal{T}$ with $t_0 \leq t_1$, and $\alpha, \beta \in \mathbb{K}$,*

$$\rho(t_1, t_0, 0, \alpha u(\cdot) + \beta \hat{u}(\cdot)) = \alpha \rho(t_1, t_0, 0, u(\cdot)) + \beta \rho(t_1, t_0, 0, \hat{u}(\cdot)). \tag{2.15}$$

3. *Zero Input/Zero State Decomposition: For all $x_0 \in \Sigma$, $u(\cdot) \in \mathcal{U}$, and $t_0, t_1 \in \mathcal{T}$ with $t_0 \leq t_1$,*

$$\rho(t_1, t_0, x_0, u(\cdot)) = \rho(t_1, t_0, x_0, 0) + \rho(t_1, t_0, 0, u(\cdot)). \tag{2.16}$$

Here, $\rho(t_1, t_0, x_0, 0)$ is called the zero-input response and $\rho(t_1, t_0, 0, u(\cdot))$ the zero-state response.

The last item of Proposition 2.1 tells us that, in order to understand the response of a linear I/O system to *any* input, all we need is the zero-input response and the zero-state response—there is an exact decomposition of the total response into the zero-input and zero-state components.

Exercise 2.4 Prove Proposition 2.1.

Using our earlier definition of time-invariance, one may determine a further classification of linear dynamical systems.

Definition 2.9 (Linear Time-Invariant/Varying System) An I/O system $\mathcal{D}$ is said to be linear, time-invariant (LTI) if it is a linear I/O system and it is time-invariant. If a linear I/O system is not necessarily linear, time-invariant, it is said to be linear, time-varying (LTV).

2.1.2 State Space Representations of Linear Systems

Thus far, we've only dealt with *abstract* dynamical systems, in which the evolution of the system is described by an arbitrary state transition map. Is this the most convenient way of describing a dynamical system? In practice, dynamical systems are typically not specified via their state transition map. Instead, one often specifies a set of equations (such as a differential equation or a recurrence relation) from which a state transition map can be determined. If we wish to describe a Newtonian physical system, for instance, we might start by writing down Newton's second law, $F = m\ddot{x}$, and deriving *differential equations* of motion. How do systems with dynamics of this form correspond to the abstract dynamical systems we discussed above?

In order to establish this connection, we make the important distinction between a *representation* of a dynamical system and the dynamical system itself. When we write down a differential equation such as $F = m\ddot{x}$, we define a differential equation *representation* of an abstract dynamical system. More generally, we say that a *system representation* is a description of a dynamical system using some mathematical framework (such as an ordinary differential equation, partial differential equation, recurrence relation, etc.) that fully determines the dynamical system.

Definition 2.10 ((Informal) I/O System Representation) An I/O system representation is a collection of mathematical data that uniquely determines a causal I/O dynamical system.

Let's return to the example of a physical system described by $F = m\ddot{x}$ to illustrate what we mean by this. Let's take F to be the input to the dynamical system, $(x, \dot{x})$ to be the state, and x to be the output. The pair,

$$\ddot{x} = \frac{1}{m}F, \quad r(x, \dot{x}, F) = x, \tag{2.17}$$

of an ordinary differential equation $\ddot{x} = \frac{1}{m}F$ and an readout function $r(x, \dot{x}, F) = x$, together with sets of admissible inputs and outputs, constitute a representation of the abstract dynamical system. The solutions of the differential equation uniquely determine the state transition map of the dynamical system while the equation $r(x, \dot{x}, F) = x$ determines the

readout map. Thus, the physical system is *represented* by a differential equation, a readout map, and input and output spaces.

This simple example leads us to ask a few important questions regarding representations of dynamical systems. What are common representations of dynamical systems? Linear dynamical systems? Linear, time-invariant dynamical systems?

We'll first answer these questions for the continuous-time case, in which $\mathcal{T} = \mathbb{R}$. In order to present a well-posed definition for system representations of continuous-time linear systems, we first need to define a special class of signals: *piecewise continuous signals*. In the next section, we'll find that piecewise continuous signals are the "right" class of input signal for continuous-time linear system representations.

Definition 2.11 (Piecewise Continuity) Let V be a normed vector space and $I \subseteq \mathbb{R}$ a (possibly infinite) interval. A mapping $u : I \to V$ is said to be piecewise continuous on I if there exists a set $D \subseteq I$, called the discontinuity set, for which the following hold:

1. Continuity outside D: u is continuous on $I \setminus D$.
2. Left and right limits: for all $\tau \in D$, the left and right limits $\lim_{t \to \tau^-} u(t)$ and $\lim_{t \to \tau^+} u(t)$ exist and are finite.
3. Finite intersections: for all $t_0, t_1 \in \mathbb{R}$ with $t_0 \leq t_1$, the set $D \cap [t_0, t_1]$ contains a finite number of points.

The set of all piecewise continuous mappings from I into V is denoted $PC(I, V)$.

Remark 2.12 The above is sometimes referred to as "piecewise continuity with one-sided limits." Here, we incorporate the one-sided limits into the definition of piecewise continuity.

Example 2.1 (Unit Step Function) The unit step function, $\mathbb{1} : \mathbb{R} \to \mathbb{R}$, defined

$$\mathbb{1}(t) = \begin{cases} 0 & t < 0 \\ 1 & t \geq 0, \end{cases} \tag{2.18}$$

is a piecewise continuous function in $PC(\mathbb{R}, \mathbb{R})$, with discontinuity set $D = \{0\}$.

Exercise 2.5 Verify that any continuous mapping $f : I \to V$ is piecewise continuous.

Importantly, the set of piecewise continuous functions has a natural vector space structure.

Proposition 2.2 ($PC(I, V)$ is a Vector Space) *Let $I \subseteq \mathbb{R}$ a nonempty interval and V a normed vector space over $\mathbb{K}$. When equipped with the operations $+$ of function addition and $(\cdot)$ of scalar multiplication of functions, $PC(I, V)$ forms a vector space over $\mathbb{K}$.*

Proof See Problem 2.2. □

With these definitions in our toolbelt, we formulate a precise, well-posed definition for a continuous-time LTV system representation.

Definition 2.12 (Continuous-Time LTV System Representation) A continuous-time LTV system representation consists of the following data:

1. Input, output, and state spaces: an input space $\mathcal{U} = PC(\mathbb{R}, \mathbb{R}^m)$, output space $\mathcal{Y} = PC(\mathbb{R}, \mathbb{R}^p)$, and state space $\mathcal{Z} = \mathbb{R}^n$.

2. Matrix functions: matrix functions $A(\cdot)$, $B(\cdot)$, $C(\cdot)$, and $D(\cdot)$ satisfying,

$$A(\cdot) \in PC(\mathbb{R}, \mathbb{R}^{n\times n}),\ B(\cdot) \in PC(\mathbb{R}, \mathbb{R}^{n\times m}), \tag{2.19}$$
$$C(\cdot) \in PC(\mathbb{R}, \mathbb{R}^{p\times n}),\ D(\cdot) \in PC(\mathbb{R}, \mathbb{R}^{p\times m}). \tag{2.20}$$

3. State & output equations: a differential equation and an algebraic equation,

$$\dot{x}(t) = A(t)x(t) + B(t)u(t) \text{ (state equation)} \tag{2.21}$$
$$y(t) = C(t)x(t) + D(t)u(t) \text{ (output equation)}, \tag{2.22}$$

where $x(t) \in \mathbb{R}^n$, $u(\cdot) \in \mathcal{U} = PC(\mathbb{R}, \mathbb{R}^m)$, and $y(\cdot) \in \mathcal{Y} = PC(\mathbb{R}, \mathbb{R}^p)$.

We refer to the system representation by the tuple $(A(\cdot), B(\cdot), C(\cdot), D(\cdot))$. Such a representation is said to be a continuous-time, *state-space* system representation. The vector $x(t)$ is referred to as the *state vector*, $u(t)$ as the *input vector*, and $y(t)$ as the *output vector*.

Remark 2.13 Since the input space of a continuous-time LTV system representation is $\mathcal{U} = PC(\mathbb{R}, \mathbb{R}^n)$, the input-value space is $U = \mathbb{R}^m$. Likewise, the output-value space is $Y = \mathbb{R}^p$.

This definition has a lot of moving parts, so let's take a moment to summarize the key points. A continuous time, linear time-varying system is defined by a tuple $(A(\cdot), B(\cdot), C(\cdot), D(\cdot))$ of *matrix-valued* functions, each of which is piecewise continuous. The time set of the system is $\mathbb{R}$, which makes it a continuous-time system. The state space of such a system is $\mathbb{R}^n$, while the input and output spaces are $PC(\mathbb{R}, \mathbb{R}^m)$ and $PC(\mathbb{R}, \mathbb{R}^p)$. Since the system is described by a pair of a state equation (the differential equation $\dot{x} = A(t)x + B(t)u$) and an output equation (the algebraic equation $y = C(t)x + D(t)u$), such a representation is referred to as a *state-space* representation.

Why the emphasis on piecewise continuity? As we'll see in the next section, the piecewise continuity assumption is *essential* for the system representation to determine a unique dynamical system. Without this assumption, we aren't guaranteed to have unique solutions to the differential equation $\dot{x} = A(t)x + B(t)u$, which would cause us trouble in defining a state transition map.

Now that we've defined a linear, time-varying system representation, we have an enormous open question on our hands:

Does Definition 2.12 determine a formal linear I/O dynamical system in the sense of Definition 2.8?

In order to answer this question, one must perform a nontrivial study of the existence and uniqueness of solutions to differential equations. Nevertheless, we will find in the next section that, after performing this study, Definition 2.12 does indeed yield a valid system representation. More on this later!

Let's write down a few more important classes of system representations. Now that we've defined a continuous time, linear time-*varying* system representation, it's only natural to define a continuous time, linear time-*invariant* system representation.

Definition 2.13 (Continuous-Time LTI System Representation) A continuous-time LTI system representation consists of the following data:

1. Input, output, and state spaces: an input space $\mathcal{U} = PC(\mathbb{R}, \mathbb{R}^m)$, output space $\mathcal{Y} = PC(\mathbb{R}, \mathbb{R}^p)$, and state space $\Sigma = \mathbb{R}^n$.

2. Matrices: fixed matrices $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$, and $D \in \mathbb{R}^{p\times m}$.
3. State & output equations: a differential equation and an algebraic equation,

$$\dot{x}(t) = Ax(t) + Bu(t) \text{ (state equation)} \tag{2.23}$$
$$y(t) = Cx(t) + Du(t) \text{ (output equation)}, \tag{2.24}$$

where $x(t) \in \mathbb{R}^n$, $u(\cdot) \in \mathcal{U} = PC(\mathbb{R}, \mathbb{R}^m)$, and $y(\cdot) \in \mathcal{Y} = PC(\mathbb{R}, \mathbb{R}^p)$.

We refer to the system representation by the tuple (A, B, C, D).

Thus, in order to define a continuous time, linear time *invariant* system, we simply take the definition of a continuous time, linear time varying system and remove all dependence on time from the matrix functions $(A(\cdot), B(\cdot), C(\cdot), D(\cdot))$. Once again, we face a question regarding how this representation relates to the LTI systems we defined above.

Does Definition 2.13 determine a formal LTI I/O dynamical system in the sense of Definition 2.9?

Fortunately, we'll find that the answer is yes! As with the above, we'll wait until the next section to prove this.

Now, we define discrete-time analogues of Definitions 2.12 and 2.13. Since the time set of a discrete-time system is $\mathbb{Z}$, we can drop all of the piecewise continuity assumptions on matrix functions and signals when defining discrete-time system representations.

Definition 2.14 (Discrete-Time LTV System Representation) A discrete-time LTV system representation consists of the following data:

1. Input, output and state spaces: an input space $\mathcal{U} = \{u : \mathbb{Z} \to \mathbb{R}^m\}$ and output space $\mathcal{Y} = \{y : \mathbb{Z} \to \mathbb{R}^p\}$ of all functions from $\mathbb{Z}$ to $\mathbb{R}^m$ and $\mathbb{R}^p$, and state space $\Sigma = \mathbb{R}^n$.
2. Matrix functions: matrix-valued functions $A[\cdot], B[\cdot], C[\cdot], D[\cdot]$,

$$A[\cdot] : \mathbb{Z} \to \mathbb{R}^{n\times n},\ B[\cdot] : \mathbb{Z} \to \mathbb{R}^{n\times m} \tag{2.25}$$
$$C[\cdot] : \mathbb{Z} \to \mathbb{R}^{p\times n},\ D[\cdot] : \mathbb{Z} \to \mathbb{R}^{p\times m}. \tag{2.26}$$

3. State & output equations: a recurrence relation and an algebraic equation,

$$x[k+1] = A[k]x[k] + B[k]u[k] \text{ (state equation)} \tag{2.27}$$
$$y[k] = C[k]x[k] + D[k]u[k] \text{ (output equation)}, \tag{2.28}$$

where $x[k] \in \mathbb{R}^n$, $u[\cdot] \in \mathcal{U} = \{u : \mathbb{Z} \to \mathbb{R}^m\}$, and $y[\cdot] \in \mathcal{Y} = \{y : \mathbb{Z} \to \mathbb{R}^p\}$.

We refer to the system representation by the tuple $(A[\cdot], B[\cdot], C[\cdot], D[\cdot])$. Such a system is said to be a discrete time, *state-space* system representation. The vector $x[k]$ is referred to as the *state vector*, $u[k]$ as the *input vector*, and $y[k]$ as the *output vector*.

Remark 2.14 Note that there are several popular ways of writing the state and output equations of a discrete-time system. Above, we've used square brackets to describe the time, $k \in \mathbb{Z}$. Other popular notation includes,

$$x_{k+1} = A_k x_k + B_k u_k \qquad x(k+1) = A(k)x(k) + B(k)u(k) \tag{2.29}$$
$$y_k = C_k x_k + D_k u_k \qquad y(k) = C(k)x(k) + D(k)u(k). \tag{2.30}$$

Remark 2.15 In Definition 2.14, we've defined inputs, outputs, and matrix-valued functions to be functions from $\mathbb{Z}$ into each of their respective spaces. This means that we can view the inputs, outputs, and matrix functions as *sequences*.

Definition 2.15 (Discrete-Time LTI System Representation) A discrete-time LTI system representation consists of the following data:

1. Input, output, and state spaces: an input space $\mathcal{U} = \{u : \mathbb{Z} \to \mathbb{R}^m\}$, output space $\mathcal{Y} = \{y : \mathbb{Z} \to \mathbb{R}^p\}$, and state space $\Sigma = \mathbb{R}^n$.
2. Matrices: fixed matrices $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$, and $D \in \mathbb{R}^{p\times m}$.
3. State & output equations: a recurrence relation and an algebraic equation,

$$x[k+1] = Ax[k] + Bu[k] \text{ (state equation)} \tag{2.31}$$
$$y[k] = Cx[k] + Du[k] \text{ (output equation)}, \tag{2.32}$$

where $x[k] \in \mathbb{R}^n$, $u[\cdot] \in \mathcal{U} = \{u : \mathbb{Z} \to \mathbb{R}^m\}$, and $y[\cdot] \in \mathcal{Y} = \{y : \mathbb{Z} \to \mathbb{R}^p\}$.

We refer to the system representation by the tuple (A, B, C, D).

The fact that we don't have to worry about regularity conditions on our signals (as we did with piecewise-continuity in continuous-time) hints that the analysis of discrete-time systems might be easier than the analogous analysis of continuous-time systems. This is in fact the case in a number of scenarios.

To wrap this section up, we define SISO and MIMO systems. Frequently, we'll distinguish between systems that only have a single input and output (which are generally easier to analyze) and systems that have multiple inputs and outputs.

Definition 2.16 (SISO/MIMO System Representations) Consider a continuous or discrete-time system representation in which

$$\mathcal{U} = PC(\mathbb{R}, \mathbb{R}^m) \text{ or } \mathcal{U} = \{u : \mathbb{Z} \to \mathbb{R}^m\} \tag{2.33}$$
$$\mathcal{Y} = PC(\mathbb{R}, \mathbb{R}^p) \text{ or } \mathcal{Y} = \{y : \mathbb{Z} \to \mathbb{R}^p\}. \tag{2.34}$$

If $m = p = 1$, the system is said to be single-input, single-output (SISO). If $m, p \geq 1$, the system is said to be multi-input, multi-output (MIMO).

Remark 2.16 Note that we don't take the definition of MIMO to be strictly greater than 1—this way, we can consider MIMO to be a straightforward generalization of SISO.

Based on this definition, SISO systems appear to be scalar from the input-output perspective: we put in a scalar as an input and get another scalar as an output. It's important to note that the SISO/MIMO distinction has *nothing* to do with the dimension of the state space! One can have a SISO system with an arbitrarily high-dimension state space, provided the input and output-value spaces are both one-dimensional.

2.1.3 Further Reading

This section was mainly influenced by [33], [24], and [6]. Our development of abstract dynamical systems most closely follows that of Chapter 5 of [6]. For a more in-depth look at abstract dynamical systems, we refer the reader to Chapter 2 of [33]. The example of a shower control system, used all the way at the start of the section, is from [27].

2.1.4 Problems

Problem 2.1 (Causal & Noncausal Maps [2]) As we mentioned in the section above, one can represent the input/output relationship of a system for a *fixed* initial time and state directly with a function $H : \mathcal{T} \times \mathcal{U} \to \mathcal{Y}$. That is, one has $y(t) = H(t, u(\cdot))$ for any time t and admissible input $u(\cdot)$. In this problem, we'll determine definitions for causality, linearity, and time-invariance of an arbitrary map $H : \mathcal{T} \times \mathcal{U} \to \mathcal{Y}$.

1. Given an arbitrary map $H : \mathcal{T} \times \mathcal{U} \to \mathcal{Y}$, formulate a definition of *time-invariance* for H. Formulate a definition of *causality*. Formulate a definition of *linearity*. *Hint: for causality, think about the restriction of a signal to a certain time interval.*
2. Let's put our definitions to the test. In each of the following cases, determine whether the system is causal/time-invariant/linear. Use your best judgment to identify the input and output spaces in each case.
   a. Consider a discrete-time system with I/O description $y[k] = c_1 u[k+1] + c_2$, where $c_1, c_2 \in \mathbb{R}$. Is this system causal? Is it time-invariant? Is it linear?
   b. Consider a continuous-time system with I/O description $y(t) = u(t - \tau)$, where $\tau \in \mathbb{R}$ is fixed and positive. Is this system causal? Is it time invariant? Is it linear?
   c. Consider a continuous-time system with I/O description,

$$y(t) = \begin{cases} u(t) & t \leq \tau \\ 0 & t > \tau. \end{cases} \tag{2.35}$$

   Is this system causal? Is it time-invariant? Is it linear?
   d. Consider a continuous-time system with I/O description,

$$y(t) = \min\{u_1(t), u_2(t)\}, \tag{2.36}$$

   where $u(t) = [u_1(t); u_2(t)]^\top$ is the system input. Is this system causal? Is it time-invariant? Is it linear?

Problem 2.2 (Properties of Piecewise-Continuous Functions) In the section above, we introduced the class of piecewise-continuous functions. In this problem, we'll prove some basic properties of this function class.

1. Show that $PC(\mathbb{R}, \mathbb{R}^n)$ forms a vector space over $\mathbb{R}$ under the operations of function addition and scalar multiplication.
2. Let $I, K \subseteq \mathbb{R}$ be compact intervals. Show that any $f \in PC(I, \mathbb{R})$ must be bounded above on $I \cap K$,

$$\sup_{t \in I \cap K} f(t) < \infty. \tag{2.37}$$

3. Let $I \subseteq \mathbb{R}$ be a compact interval and $\|\cdot\|$ be an arbitrary norm on $\mathbb{R}^n$. Show that the supremum norm,

$$\|f\|_\infty = \sup_{t \in I} \|f(t)\|, \tag{2.38}$$

   is finite for all $f \in PC(I, \mathbb{R}^n)$. Then, prove that $\|\cdot\|_\infty$ makes $PC(I, \mathbb{R}^n)$ into a normed vector space.

4. Is $PC(I, \mathbb{R}^n)$ a Banach space with respect to the supremum norm $\|\cdot\|_\infty$, $\|f\|_\infty = \sup_{t\in\mathbb{R}} \|f(t)\|$? Provide a proof or a counterexample.

2.2 Solutions of Linear, Time-Varying Systems

Now that we've introduced a set of state space representations of linear systems, we must show that these representations *are* in fact representations in the formal sense. Recall that in the previous section, we posed two questions:

Does an LTV system representation determine a linear I/O system?
Does an LTI system representation determine an LTI I/O system?

Further, we promised that in this section, we would provide a *precise* answer to both of these queries. Now that we're here, we need to make good on this promise! In this section, we establish answers to these questions by studying the existence, uniqueness, and structure of solutions to linear ordinary differential equations and recurrence relations. We'll then apply the results of this study to answer the two questions above.

In order to answer these questions, we'll split up into the continuous and discrete-time cases. In order to study continuous-time linear systems, we must study linear ordinary differential equations, while to study discrete-time linear systems, we must study linear recurrence relations. Along the way, we'll draw connections between the techniques used to study the two. Let's begin!

2.2.1 Solutions of Continuous-Time Linear Systems

We'll begin by laying out a brief plan of attack for our study of continuous-time linear system representations. Recall from the previous section that a continuous-time LTV system representation is specified by a tuple $(A(\cdot), B(\cdot), C(\cdot), D(\cdot))$ of piecewise continuous matrix-valued functions. These functions define two equations,

$$\dot{x}(t) = A(t)x(t) + B(t)u(t) \text{ (state equation)} \tag{2.39}$$
$$y(t) = C(t)x(t) + D(t)u(t) \text{ (output equation)}, \tag{2.40}$$

which govern how the state $x(t)$ and the output $y(t)$ change over time as input signals $u : \mathbb{R} \to \mathbb{R}^m$ are applied to the system.

In order to determine if the LTV system representation yields a valid linear input/output dynamical system in the sense of Definition 2.8, we must tick a couple of boxes. To verify that the LTV system representation yields a linear I/O system, we must compute the I/O map, ρ, associated to the representation, and verify that it satisfies the linearity conditions proposed in the previous section. In order to compute ρ, however, we require the state transition map, φ, of the representation.

Thus, we begin by studying the state transition map. The state transition map associated to the LTV system representation maps from times $t_0, t_1 \in \mathbb{R}$ with $t_0 \leq t_1$, initial state $x_0 \in \mathbb{R}^n$, and input signal $u \in PC(\mathbb{R}, \mathbb{R}^m)$ to the solution of the differential equation,

$$\dot{x}(t) = A(t)x(t) + B(t)u(t), \tag{2.41}$$

at time t_1, with initial condition $x(t_0) = x_0$. Therefore, in order to have a state transition map $\varphi(t_1, t_0, x_0, u(\cdot))$ that is well-defined on the input space $\mathcal{U} = PC(\mathbb{R}, \mathbb{R}^m)$, there are a couple of things we need:

1. Existence: for all inputs $u \in PC(\mathbb{R}, \mathbb{R}^m)$, initial conditions $x_0 \in \mathbb{R}^n$, and times $t_0 \in \mathbb{R}$, a solution to the initial value problem,

$$\dot{x}(t) = A(t)x(t) + B(t)u(t),\ x(t_0) = x_0, \tag{2.42}$$

must exist. This ensures that we will always be able to compute φ on the time set, state space, and input space of the representation.
2. Uniqueness: for all inputs $u \in PC(\mathbb{R}, \mathbb{R}^m)$, initial conditions $x_0 \in \mathbb{R}^n$, and times $t_0 \in \mathbb{R}$, the solution to the initial value problem,

$$\dot{x}(t) = A(t)x(t) + B(t)u(t),\ x(t_0) = x_0, \tag{2.43}$$

must be *unique*. If we want the state transition map to be well-defined, we can't have two or more solutions to the initial value problem!

Thus, in order to solve the problem of verifying the LTV system representation yields a linear I/O system, we must first establish the existence and uniqueness of solutions to the initial value problem $\dot{x} = A(t)x(t) + B(t)u(t),\ x(t_0) = x_0$. We'll break this down into the following multi-step process:

1. Define Solutions: first, we'll formulate a precise definition for a solution to a time-varying initial value problem with piecewise continuous data.
2. Matrix IVP: next, we'll argue that it will be insightful to study solutions to a simpler yet more fundamental initial value problem, the *matrix* initial value problem

$$\dot{X}(t) = A(t)X(t),\ X(t_0) = I. \tag{2.44}$$

We will prove that this problem has a unique solution and will study its basic structure.
3. State Transition Matrix: after showing that a unique solution to the IVP $\dot{X}(t) = A(t)X(t),\ X(t_0) = I$ exists, we'll *abstract away* details of the solution into a special operator called the state transition matrix. Then, we'll establish a few important properties of the state transition matrix.
4. LTV-IVP: finally, we'll show that we can use the state transition matrix to study solutions to the general linear time-varying, initial value problem. Here, we'll complete the problem of proving existence and uniqueness of solutions.

2.2.1.1 Defining Solutions to IVPs

Let's tackle the first step of the process we outlined above. What does it mean to solve a time-varying initial value problem with piecewise continuous data? Although it might *seem* like all we need is to find a differentiable function which satisfies $\dot{x}(t) = A(t)x(t) + B(t)u(t)$ for all $t \in \mathbb{R}$ and $x(t_0) = x_0$, the reality is somewhat more complex! To illustrate what goes wrong with this "naive" definition of a solution, consider the case of a simple scalar, time-varying initial value problem,

$$\dot{x}(t) = b(t),\ x(0) = 0. \tag{2.45}$$

Suppose $b \in PC(\mathbb{R}, \mathbb{R})$ is the *step function*, the function which is identically zero before $t = 0$ and identically one at and after $t = 0$,

$$b(t) = \begin{cases} 1, & t \geq 0 \\ 0, & t < 0. \end{cases} \tag{2.46}$$

Using our basic knowledge of ODEs, we know that the solution to this initial value problem *should* be the ramp function,

$$x(t) = \begin{cases} t & t \geq 0 \\ 0 & t < 0. \end{cases} \tag{2.47}$$

However, this function is clearly *not* differentiable at the point $t = 0$—right where b makes

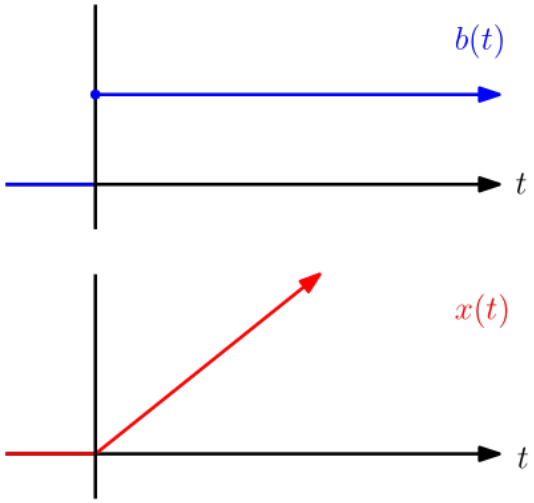


Fig. 2.4 The solution to the initial value problem $\dot{x}(t) = b(t)$, with b the step function, *should* be the ramp function. However, the ramp function is *not* differentiable at $t = 0$! Thus, our definition for a solution an IVP must account for points of non-differentiability.

its jump, we find that the proposed solution of the initial value problem has a sharp "corner." As a result of this, we find that requiring a solution to this IVP to be a *differentiable* function $x : \mathbb{R} \to \mathbb{R}$ satisfying $\dot{x}(t) = b(t)\ \forall t$, $x(0) = 0$, is *too strong*.

This example is a particular instance of a general fact from analysis: if a function has a jump discontinuity, it *cannot* be the derivative of another function (see Problem 2.4 for the formal details of this argument). We conclude that, in order to make a *well-posed* definition for a solution to an initial value problem with piecewise continuous data, we must explicitly account for the points of discontinuity. This leads us to the following, formal definition of a solution.

Definition 2.17 (Solution to LTV-IVP) Consider the piecewise-continuous maps $A(\cdot) \in PC(\mathbb{R}, \mathbb{R}^{n\times n})$, $B(\cdot) \in PC(\mathbb{R}, \mathbb{R}^{n\times m})$, and $u(\cdot) \in PC(\mathbb{R}, \mathbb{R}^m)$, which have a shared discontinuity set $D \subseteq \mathbb{R}$. For $x_0 \in \mathbb{R}^n$ and $t_0 \in \mathbb{R}$, a solution to the initial value problem,

$$\dot{x}(t) = A(t)x(t) + B(t)u(t),\ x(t_0) = x_0, \tag{2.48}$$

is a continuous map $x : \mathbb{R} \to \mathbb{R}^n$, satisfying the conditions:

1. Initial condition: $x(t_0) = x_0$.
2. Derivative: For all $t \in \mathbb{R} \setminus D$, $\frac{d}{dt}x(t) = A(t)x(t) + B(t)u(t)$.

Remark 2.17 Here, we take solutions to be defined for *all* $t \in \mathbb{R}$. We'll later see that this is justified for the case of LTV systems with piecewise continuous data. However, this requirement should be relaxed for *nonlinear* initial value problems.

Remark 2.18 Here, we assume that each map has the same discontinuity set - this assumption is made without loss of generality, since one can always take the union of the discontinuity sets of $A(\cdot), B(\cdot), u(\cdot)$ if they do not initially coincide.

This formal definition of a solution to an initial value problem *relaxes* our "intuitive" definition of a solution. It tells us that we only need to check the derivative condition at times when all data defining the initial value problem is continuous. Since carrying around the discontinuity set D can get a little cumbersome, we state an *equivalent* definition of a solution to an initial value problem which doesn't require the use of D. We state this definition in the form of a proposition.

Proposition 2.3 (Integral Solution of IVPs) *Consider the piecewise continuous maps $A(\cdot) \in PC(\mathbb{R}, \mathbb{R}^{n\times n})$, $B(\cdot) \in PC(\mathbb{R}, \mathbb{R}^{n\times m})$, and $u \in PC(\mathbb{R}, \mathbb{R}^m)$. A continuous function $x : \mathbb{R} \to \mathbb{R}^n$ is a solution to the initial value problem,*

$$\dot{x}(t) = A(t)x(t) + B(t)u(t),\ x(t_0) = x_0, \tag{2.49}$$

if and only if, for all $t \in \mathbb{R}$, it satisfies

$$x(t) = x_0 + \int_{t_0}^{t} A(\tau)x(\tau) + B(\tau)u(\tau)d\tau. \tag{2.50}$$

The proof of this proposition is essentially a straightforward application of the fundamental theorem of calculus, which we now recall.

Theorem 2.1 (Fundamental Theorem of Calculus) *. Let $f : \mathbb{R} \to \mathbb{R}^n$ be a Riemann-integrable function.[2] Then, the following results hold:*

1. *Fix a number $t_0 \in \mathbb{R}$ and define $F(t) = \int_{t_0}^{t} f(\tau)d\tau$. Then, F is continuous. Further, if f is continuous at t, then F is differentiable at t, with $F'(t) = f(t)$.*
2. *If $F : \mathbb{R} \to \mathbb{R}^n$ is a Riemann-integrable function satisfying $F'(t) = f(t)$ for all but a finite number of points in an interval $[t_0, t_1] \subseteq \mathbb{R}$, then $\int_{t_0}^{t_1} f(\tau)d\tau = F(t_1) - F(t_0)$.*

Proof See [1] for details. □

Now, we return to the proof of Proposition 2.3.

Proof (Of Proposition 2.3) Suppose $x : \mathbb{R} \to \mathbb{R}^n$ is a solution to the initial value problem in the sense of Definition 2.17. Then, for D the shared discontinuity set of $A(\cdot), B(\cdot), u(\cdot)$, one has that $\dot{x}(t) = A(t)x(t) + B(t)u(t)$, for all $t \in \mathbb{R} \setminus D$. We aim to show that x satisfies the integral condition proposed above.

Fix a time $t \in \mathbb{R}$, assuming $t \geq t_0$ (the proof is identical for $t < t_0$). By definition of a piecewise continuous function, it follows that $[t_0, t] \cap D$ only contains a finite number of

[2] We say that a function $f : \mathbb{R} \to \mathbb{R}^n$ is Riemann-integrable if its Riemann integral $\int_a^b f(t)dt$ is defined for all (finite) $a, b \in \mathbb{R}$. One may show that all piecewise continuous functions are Riemann-integrable.

points. Thus, between t_0 and t_1, $\dot{x}(t) = A(t)x(t) + B(t)u(t)$ at all but a finite number of points. By the fundamental theorem of calculus, it then follows that

$$x(t) = x(t_0) + \int_{t_0}^{t} A(\tau)x(\tau) + B(\tau)u(\tau)d\tau = x_0 + \int_{t_0}^{t} A(\tau)x(\tau) + B(\tau)u(\tau)d\tau. \quad (2.51)$$

This completes the first direction of the proof. Now, we proceed in the other direction. Suppose $x(t) = x_0 + \int_{t_0}^{t} A(\tau)x(\tau) + B(\tau)u(\tau)d\tau$ for all $t \geq t_0$. Taking $t = t_0$, one gets that $x(t_0) = x_0$, yielding the initial condition constraint. Now, we focus on differentiability. We know that $A(t)x(t) + B(t)u(t)$ must be continuous at all $t \in \mathbb{R} \setminus D$, since x is continuous by the fundamental theorem and $A(\cdot), B(\cdot), u(\cdot)$ are continuous on $\mathbb{R} \setminus D$. By the fundamental theorem, x is differentiable on $\mathbb{R} \setminus D$ and satisfies $\dot{x}(t) = A(t)x(t) + B(t)u(t)$ for all $t \in \mathbb{R} \setminus D$. We conclude that x is a solution to the initial value problem. □

2.2.1.2 A Matrix Initial Value Problem & The Peano-Baker Series

In the previous section, we wrote a formal definition for a solution to an initial value problem and formulated an equivalent definition of a solution using integration. In this subsection, we will examine a special, simple initial value problem that will provide almost *complete* insight into the existence and uniqueness problem we're aiming to solve.

Since the initial value problem $\dot{x}(t) = A(t)x(t) + B(t)u(t)$, $x(t_0) = x_0$ seems quite complex to analyze, having a number of moving parts, it might be beneficial to start with a simpler problem. Let's drop the input term, and study solutions to the initial value problem,

$$\dot{x}(t) = A(t)x(t),\ x(t_0) = x_0. \quad (2.52)$$

Can we simplify this problem even further? Using the integral definition of a solution to an initial value problem, we make the following observation.

Lemma 2.1 (Matrix IVP/Vector IVP) *Consider the initial value problem,*

$$\dot{x}(t) = A(t)x(t),\ x(t_0) = x_0. \quad (2.53)$$

If $X : \mathbb{R} \to \mathbb{R}^{n \times n}$ solves the matrix initial value problem,

$$\dot{X}(t) = A(t)X(t), X(t_0) = I, \quad (2.54)$$

then $x(t) = X(t)x_0$ solves the vector initial value problem $\dot{x}(t) = A(t)x(t)$, $x(t_0) = x_0$.

Remark 2.19 In this lemma, we use a solution to a *matrix* initial value problem. Solutions to such initial value problems are defined identically to vector initial value problems. In fact, we can write down a vector initial value problem corresponding to any given matrix initial value problem by stretching the matrix out into a vector. Because of this equivalence, the integral definition of a solution to an IVP still holds in the matrix case. Try writing down a formal definition of a solution to a matrix IVP to check your understanding!

Proof Suppose $X(t)$ solves the matrix IVP defined in the statement of the lemma. Then,

$$X(t) = I + \int_{t_0}^{t} A(\tau)X(\tau)d\tau. \quad (2.55)$$

Multiplying by x_0, one has,

$$X(t)x_0 = x_0 + \int_{t_0}^{t} A(\tau)X(\tau)x_0 d\tau, \tag{2.56}$$

which implies $x(t) = X(t)x_0$ is a solution to the IVP $\dot{x}(t) = A(t)x(t)$, $x(t_0) = x_0$. □

This lemma yields a great deal of insight into into the structure of solutions to $\dot{x}(t) = A(t)x(t),\ x(t_0) = x_0$. In particular, it tells us that there exist solutions to the initial value problem that are *linear* in the initial condition! Further, these solutions are *entirely* determined by solutions to the *matrix* IVP $\dot{X}(t) = A(t)X(t),\ X(t_0) = I$. Thus, in order to understand solutions to the vector initial value problem $\dot{x}(t) = A(t)x(t),\ x(t_0) = x_0$, we will study solutions to the associated *matrix* initial value problem, $\dot{X}(t) = A(t)X(t),\ X(t_0) = I$.

What do we know about solutions to this matrix IVP? Do solutions exist? If so, what form do they take? A solution to the matrix IVP must satisfy the integral equation,

$$X(t) = I + \int_{t_0}^{t} A(\tau)X(\tau)d\tau. \tag{2.57}$$

We notice that $X(\cdot)$ appears both on the left and right hand sides of the expression. Let's try re-plugging in the integral form of the solution into the $X(\tau)$ on the right hand side. This gives,

$$X(t) = I + \int_{t_0}^{t} A(\tau)\left[I + \int_{t_0}^{\tau} A(\tau')X(\tau')d\tau'\right]d\tau \tag{2.58}$$

$$= I + \left[\int_{t_0}^{t} A(\tau)d\tau\right] + \int_{t_0}^{t} A(\tau)\left[\int_{t_0}^{\tau} A(\tau')X(\tau')d\tau'\right]d\tau. \tag{2.59}$$

Interestingly, what we get inside the larger integral is the *same* expression that we originally substituted into. Thus, if we substitute again—this time for $X(\tau')$—we would find the same pattern! Indefinitely performing this substitution leads to the following definition.

Definition 2.18 (Peano-Baker Series) Let $A(\cdot) \in PC(\mathbb{R}, \mathbb{R}^{n\times n})$. The Peano-Baker series with respect to $A(\cdot)$ is the infinite matrix series,

$$\Phi(t, t_0) = \sum_{k=0}^{\infty} S_k(t, t_0), \tag{2.60}$$

whose summands are defined by the recurrence,

$$S_0 = I,\ S_{k+1}(t, t_0) = \int_{t_0}^{t} A(\tau)S_k(\tau, t_0)d\tau. \tag{2.61}$$

In order to confirm that this definition is well-posed, we must, we must confirm that the Peano-Baker series actually converges. A useful tool for certifying the uniform convergence[3] of a series of functions is the *Weierstrass M-Test.*

[3] Recall from Chapter 1 that a sequence of functions $f_n : I \subseteq \mathbb{R} \to V$ (for $(V, \|\cdot\|)$ a normed vector space) converges *uniformly* if it converges with respect to the sup norm, $\|f\|_\infty = \sup_{t\in I} \|f(t)\|$.

Theorem 2.2 (Weierstrass M-Test) *Let $(V, \|\cdot\|)$ be a finite dimensional, normed vector space. Let $\{f_n\}$ be a collection of mappings $f_n : A \to V$ on a set A. Let $\{M_n\} \subseteq \mathbb{R}$ be a sequence for which $\sup_{t\in A} \|f_n(t)\| \leq M_n$. If $\sum_{n=1}^{\infty} M_n$ converges, then $\sum_{n=1}^{\infty} f_n(t)$ converges uniformly on A.*

Proof See [1] for the details. □

In order to apply the Weierstrass M-test to certify the convergence of the Peano-Baker series, we require the following lemma.

Lemma 2.2 (Suprema of Piecewise Continuous Functions) *Consider a piecewise continuous function $f \in PC(I, \mathbb{R})$, where $I \subseteq \mathbb{R}$. For any $K \subseteq I$ which is compact in $\mathbb{R}$,*

$$\sup_{t\in K\cap I} f(t) < \infty \tag{2.62}$$

Proof See Problem 2.2. □

With these results in mind, we study the convergence of the Peano-Baker series.

Proposition 2.4 (Convergence of Peano-Baker Series) *Let $A(\cdot) \in PC(\mathbb{R}, \mathbb{R}^{n\times n})$. For any $t' \in \mathbb{R}_{\geq 0}$, the Peano-Baker series $\Phi(\cdot,\cdot)$ defined by $A(\cdot)$ converges uniformly on $[-t', t']^2$.*

Remark 2.20 The notation $[-t, t']^2$ refers to the Cartesian product $[-t', t'] \times [-t', t']$.

Proof Our proof follows the Weierstrass M-test. Fix an interval $[-t', t'] \subseteq \mathbb{R}$ and an initial time $t_0 \in [-t', t']$. Recall that the Peano-Baker series at time $t \in [-t', t']$ is defined,

$$\Phi(t, t_0) = \sum_{k=0}^{\infty} S_k(t, t_0),\ S_0 = I,\ S_k(t, t_0) = \int_{t_0}^{t} A(\tau) S_{k-1}(\tau, t_0) d\tau, \tag{2.63}$$

In order to prove that this series converges uniformly using the Weierstrass M-test, we'll exhibit a uniform bound on each $S_k(\cdot,\cdot)$. First, we'll show that the bound,

$$\|S_k(t, t_0)\| \leq \frac{1}{k!}\Big(\sup_{t\in[-t',t']} \|A(t)\|\Big)^k |t - t_0|^k,\ \forall t \in [-t', t'], \tag{2.64}$$

must hold. Notice that $\sup_{t\in[-t',t']} \|A(t)\|$ is finite by Lemma 2.2. Note that—although it seems like we're pulling this bound out of thin air—this is actually something one can discover by playing around with the first few terms of the series. You're encouraged to try this if you're not convinced!

Let's prove that the bound holds by induction on k. The base case, $k = 0$, is trivial. We get $\|S_0\| = \|I\| = 1$, which matches the proposed bound exactly. Let $k \geq 1$, and assume for induction that the proposed bound holds. Now, we bound S_{k+1} for arbitrary t, t_0—keep in mind, we may have $t \geq t_0$ or $t < t_0$. We have,

$$\|S_{k+1}(t, t_0)\| = \left\| \int_{t_0}^{t} A(\tau) S_k(\tau, t_0) d\tau \right\| \tag{2.65}$$

$$\leq \left| \int_{t_0}^{t} \|A(\tau)\| \, \|S_k(\tau, t_0)\| \, d\tau \right| \tag{2.66}$$

$$\leq \left| \int_{t_0}^{t} \sup_{\tau\in[-t',t']} \|A(\tau)\| \cdot \frac{1}{k!}\Big(\sup_{\tau\in[-t',t']} \|A(\tau)\|\Big)^k |\tau - t_0|^k d\tau \right|. \tag{2.67}$$

Now, we split into two cases. First, assume $t \geq t_0$. In this case, we have that the above is bounded,

$$\|S_{k+1}(t,t_0)\| \leq \frac{1}{k!}\Big(\sup_{t\in[-t',t']}\|A(t)\|\Big)^{k+1}\left|\int_{t_0}^{t}|\tau-t_0|^k d\tau\right| \tag{2.68}$$

$$= \frac{1}{k!}\Big(\sup_{t\in[-t',t']}\|A(t)\|\Big)^{k+1}\frac{1}{k+1}|t-t_0|^{k+1} \tag{2.69}$$

$$= \frac{1}{(k+1)!}\Big(\sup_{t\in[-t',t']}\|A(t)\|\Big)^{k+1}|t-t_0|^{k+1},\ \forall t\in[t_0,t_1]. \tag{2.70}$$

Thus, the proposed bound holds for $t \geq t_0$. For $t < t_0$, the same procedure is followed—simply flip t_0 and t in the integral and perform the same bounds. So, by induction on k, we conclude that the proposed bound holds for all $k \in \mathbb{N}$. We can then bound $\|S_k(t,t_0)\|$ uniformly in t, t_0 on $[-t', t']$ by,

$$\sup_{t,t_0\in[-t',t']}\|S_k(t,t_0)\| \leq \frac{1}{k!}\Big(\sup_{t\in[-t,t]}\|A(t)\|\Big)^{k}(2t')^k. \tag{2.71}$$

Now, we're ready to apply the Weierstrass M-test. Define M_k as the right hand side of the inequality above. Does $\sum_{k=0}^{\infty} M_k$ converge? We recognize the sum as the Taylor series definition of the *exponential*! Thus, we have,

$$\sum_{k=0}^{\infty}\frac{1}{k!}\Big(\sup_{t\in[-t',t']}\|A(t)\|\Big)^{k}(2t')^k = \exp\Big(\sup_{t\in[-t',t']}\|A(t)\|\,(2t')\Big) < \infty. \tag{2.72}$$

By the Weierstrass M-test, we conclude that the Peano-Baker series $\Phi(\cdot,\cdot)$ converges uniformly on any compact interval $[-t', t']^2$. □

Let's summarize what we've done so far. In analyzing solutions to the matrix IVP, $\dot{X}(t) = A(t)X(t)$, $X(t_0) = I$, we discovered a recurrent pattern that led us to the Peano-Baker series. Then, we proved that the Peano-Baker series converges uniformly on any compact interval. Now, we ask the question—does it converge to the solution of the matrix IVP? The following theorem provides an answer.

Proposition 2.5 (Peano-Baker Series Solves the Matrix IVP) *Let $A \in PC(\mathbb{R}, \mathbb{R}^{n\times n})$. Consider the matrix initial value problem $\dot{X}(t) = A(t)X(t)$, $X(t_0) = I$. The Peano-Baker series $\Phi(t, t_0)$ defined by $A(\cdot)$ solves the matrix initial value problem.*

Proof We can prove this either via direct differentiation or the integral method. Let's proceed via differentiation. Let $D \subseteq \mathbb{R}$ be the discontinuity set of A. By definition, one has that for all $t \in \mathbb{R} \setminus D$, each term of the Peano-Baker series is differentiable in its first argument. Thus, for any time $t \in \mathbb{R}$,

$$\frac{d}{dt}\sum_{k=0}^{n} S_k(t,t_0) = \sum_{k=0}^{n}\frac{d}{dt}S_k(t,t_0) \tag{2.73}$$

$$= \sum_{k=1}^{n}\frac{d}{dt}\int_{t_0}^{t} A(\tau)S_{k-1}(\tau,t_0)d\tau \tag{2.74}$$

$$= A(t)\sum_{k=1}^{n} S_{k-1}(t,t_0) \tag{2.75}$$

$$= A(t)\sum_{k=0}^{n-1} S_k(t,t_0). \tag{2.76}$$

Now, fix an interval $[-t_1,t_1] \subseteq \mathbb{R}$, for which $t_0 \in [-t_1,t_1]$. We know that the Peano-Baker series converges uniformly in t on this interval. Additionally, the chain of equalities above implies that $\sum_{k=0}^{n}\frac{d}{dt}S_k(t,t_0)$ converges uniformly in t on $[-t_1,t_1]$. Thus, by Theorem 1.3 on uniform convergence and differentiation, it follows that $\frac{d}{dt}\lim_{n\to\infty}\sum_{k=0}^{n} S_k(t,t_0) = \lim_{n\to\infty}\sum_{k=0}^{n}\frac{d}{dt}S_k(t,t_0)$, for all $t \in (-t_1,t_1)\cap(\mathbb{R}\setminus D)$. This means,

$$\frac{d}{dt}\sum_{k=0}^{\infty} S_k(t,t_0) = A(t)\lim_{n\to\infty}\sum_{k=0}^{n-1} S_k(t,t_0) = A(t)\sum_{k=0}^{\infty} S_k(t,t_0), \tag{2.77}$$

for all $t \in (-t_1,t_1)\cap(\mathbb{R}\setminus D)$. Thus, we have that,

$$\frac{d}{dt}\Phi(t,t_0) = A(t)\Phi(t,t_0),\ \forall t \in (-t_1,t_1)\cap(\mathbb{R}\setminus D). \tag{2.78}$$

Since the interval $[-t_1,t_1]$ can be made arbitrarily large, we conclude that the Peano-Baker series satisfies the derivative property of the initial value problem for all $t \in \mathbb{R}\setminus D$. Further, we have that $\Phi(t_0,t_0) = I$ by definition. We conclude that the Peano-Baker series solves the matrix initial value problem. □

We've now established that the Peano-Baker series is *a* solution to the matrix initial value problem—is it the *only* solution? The following inequality helps us answer this question.

Lemma 2.3 (Gronwall Inequality) *Let $y,k \in PC(\mathbb{R},\mathbb{R}_{\geq 0})$ and $c \in \mathbb{R}_{\geq 0}$, and $t_0 \in \mathbb{R}$. If for all $t \in \mathbb{R}$, y satisfies,*

$$y(t) \leq c + \left|\int_{t_0}^{t} k(\tau)y(\tau)d\tau\right|, \tag{2.79}$$

then for all $t \in \mathbb{R}$,

$$y(t) \leq c\exp\left|\int_{t_0}^{t} k(\tau)d\tau\right|. \tag{2.80}$$

Proof See Problem 2.5. □

Using the Gronwall inequality, we prove that solutions to the matrix IVP are *unique*.

Theorem 2.3 (Existence & Uniqueness of Solutions to Matrix IVP) *Let $A(\cdot) \in PC(\mathbb{R}, \mathbb{R}^{n\times n})$. The Peano-Baker series, $\Phi(t, t_0)$, is the unique solution to the matrix initial value problem $\dot{X}(t) = A(t)X(t),\ X(t_0) = I$.*

Proof By Proposition 2.5, we already know $\Phi(t, t_0)$ is a solution the IVP. Now, we show it is the unique solution. Suppose $X : \mathbb{R} \to \mathbb{R}^{n\times n}$ is another solution. Then, for any t, t_0, both X and Φ must satisfy,

$$X(t) = I + \int_{t_0}^{t} A(\tau)X(\tau)d\tau \tag{2.81}$$

$$\Phi(t, t_0) = I + \int_{t_0}^{t} A(\tau)\Phi(\tau, t_0)d\tau. \tag{2.82}$$

Subtracting and taking the norm, we get,

$$\|\Phi(t, t_0) - X(t)\| = \left\| \int_{t_0}^{t} A(\tau)(\Phi(\tau, t_0) - X(\tau))d\tau \right\| \tag{2.83}$$

$$\leq \left| \int_{t_0}^{t} \|A(\tau)\| \, \|\Phi(\tau, t_0) - X(\tau)\| \, d\tau \right|. \tag{2.84}$$

Applying the Gronwall lemma, we find that

$$\|\Phi(t, t_0) - X(t)\| = 0,\ \forall t \in \mathbb{R}. \tag{2.85}$$

We conclude that $\Phi(t, t_0) = X(t)$ for all $t \in \mathbb{R}$, and that solutions to the IVP are unique. □

2.2.1.3 The State Transition Matrix

In the previous subsection, we developed the theory of the *Peano-Baker series* to prove the existence of a unique solution to the matrix initial value problem,

$$\dot{X}(t) = A(t)X(t),\ X(t_0) = I. \tag{2.86}$$

Since the integral formula for the Peano-Baker series is rather impractical to work with, we'll find it convenient to *abstract away* the computation of the Peano-Baker series and focus on $\Phi(t, t_0)$ as the solution to the matrix initial value problem. In this spirit, we make the following definition.

Definition 2.19 (State Transition Matrix) Let $A(\cdot) \in PC(\mathbb{R}, \mathbb{R}^{n\times n})$. The state transition matrix with respect to $A(\cdot)$ is a map $\Phi : \mathbb{R} \times \mathbb{R} \to \mathbb{R}^{n\times n}$, such that $\Phi(\cdot, t_0) : \mathbb{R} \to \mathbb{R}^{n\times n}$ is the unique solution to the initial value problem,

$$\dot{X}(t) = A(t)X(t),\ X(t_0) = I. \tag{2.87}$$

Remark 2.21 Despite its name, the state transition *matrix* is not a fixed matrix but rather a *map* into the set of matrices.

We emphasize—the state transition matrix $\Phi(t, t_0)$ is *exactly* calculated by the Peano-Baker series. Here, we simply hide the Peano-Baker series behind a layer of abstraction—the *state transition matrix*—to emphasize that we don't want to use the series as an analysis tool.

By focusing on the "abstracted" definition of $\Phi(t, t_0)$ as the unique solution of a differential equation, as opposed to the definition of $\Phi(t, t_0)$ as an infinite series, we'll find that we can write much more elegant proofs.

Proposition 2.6 (Properties of the State Transition Matrix) *Let $A(\cdot) \in \mathbb{P}C(\mathbb{R}, \mathbb{R}^{n\times n})$. The state transition matrix Φ with respect to $A(\cdot)$ satisfies the following properties:*

1. *Composition: For all $t_0, t_1, t_2 \in \mathbb{R}$, $\Phi(t_2, t_0) = \Phi(t_2, t_1)\Phi(t_1, t_0)$.*
2. *Inverse: For all $t_0, t_1 \in \mathbb{R}$, $\Phi(t_1, t_0)$ is invertible with $[\Phi(t_1, t_0)]^{-1} = \Phi(t_0, t_1)$.*

Proof First, we show the composition property. To prove this, we will use the uniqueness property of solutions to the initial value problem, $\dot{X}(t) = A(t)X(t)$, $X(t_0) = X_0$, which follows from the Gronwall Lemma. In particular, we will show that, for all $t_0, t_1 \in \mathbb{R}$, both

$$\Phi(\cdot, t_0) : \mathbb{R} \to \mathbb{R}^{n\times n} \tag{2.88}$$

$$\Phi(\cdot, t_1)\Phi(t_1, t_0) : \mathbb{R} \to \mathbb{R}^{n\times n}, \tag{2.89}$$

are solutions to the matrix IVP $\dot{X}(t) = A(t)X(t)$, $X(t_1) = \Phi(t_1, t_0)$. Then, we'll use uniqueness to conclude that they are equal. First, we know that $\Phi(\cdot, t_0)$ is a solution to the matrix IVP by definition of the state transition matrix. Thus, $\Phi(\cdot, t_0)$ satisfies,

$$\frac{d}{dt}\Phi(t, t_0) = A(t)\Phi(t, t_0) \ \forall t \in \mathbb{R} \setminus D. \tag{2.90}$$

where D is the discontinuity set of A. This implies that $\Phi(t, t_0)$ is also the solution the matrix IVP, $\dot{X}(t) = A(t)X(t)$, $X(t_1) = \Phi(t_1, t_0)$. Now, we check the same for the second. We have that, for D the discontinuity set of $A(\cdot)$,

$$\frac{d}{dt}[\Phi(t, t_1)\Phi(t_1, t_0)] = A(t)\Phi(t, t_1)\Phi(t_1, t_0) = A(t)[\Phi(t, t_1)\Phi(t_1, t_0)], \ \forall t \in \mathbb{R} \setminus D. \tag{2.91}$$

Further, we have that $\Phi(t_1, t_1)\Phi(t_1, t_0) = I\Phi(t_1, t_0) = \Phi(t_1, t_0)$. Therefore, $\Phi(\cdot, t_1)\Phi(t_1, t_0)$ *also* solves the initial value problem! By the Gronwall Lemma, it follows that solutions to the IVP are unique, which implies,

$$\Phi(t_2, t_1)\Phi(t_1, t_0) = \Phi(t_2, t_0), \ \forall t_0, t_1, t_2 \in \mathbb{R}. \tag{2.92}$$

This completes the proof of the first item. The second item follows by direct application of the first. Fix times $t_0, t_1 \in \mathbb{R}$. Then, it follows from the composition rule that

$$\Phi(t_0, t_1)\Phi(t_1, t_0) = \Phi(t_0, t_0) = I \tag{2.93}$$

$$\Phi(t_1, t_0)\Phi(t_0, t_1) = \Phi(t_1, t_1) = I. \tag{2.94}$$

So, we conclude by the uniqueness of the matrix inverse that $\Phi(t_0, t_1) = [\Phi(t_1, t_0)]^{-1}$. $\square$

2.2.1.4 The Continuous-Time, LTV Initial Value Problem

We're finally ready to tackle our original problem: proving the existence & uniqueness of solutions to the initial value problem,

$$\dot{x}(t) = A(t)x(t) + B(t)u(t),\ x(t_0) = x_0. \tag{2.95}$$

Amazingly, *all we need* to construct solutions to this problem is the state transition matrix. To determine a formula for the solutions to such initial value problems, we'll need the differentiation under the integral rule, which we now recall.

Theorem 2.4 (Leibniz Rule for Differentiation Under the Integral) *Let $f \in C^1(\mathbb{R} \times \mathbb{R}, \mathbb{R}^n)$ and $a(\cdot), b(\cdot) \in C^1(\mathbb{R}, \mathbb{R})$ be continuously differentiable functions. For all $t \in \mathbb{R}$,*

$$\frac{d}{dt}\Big[\int_{a(t)}^{b(t)} f(t,\tau)d\tau\Big] = \int_{a(t)}^{b(t)} \frac{\partial}{\partial t} f(t,\tau)d\tau + f(t,b(t))\frac{d}{dt}b(t) - f(t,a(t))\frac{d}{dt}a(t). \tag{2.96}$$

Proof See Problem 2.6 for details. □

With this in mind, we state a theorem on the existence and uniqueness of solutions to the LTV initial value problem.

Theorem 2.5 (Existence & Uniqueness of Solutions to LTV-IVP) *Let $A(\cdot) \in PC(\mathbb{R}, \mathbb{R}^{n\times n})$, $B(\cdot) \in PC(\mathbb{R}, \mathbb{R}^{n\times m})$, and $u(\cdot) \in PC(\mathbb{R}, \mathbb{R}^m)$. The unique solution to the initial value problem,*

$$\dot{x}(t) = A(t)x(t) + B(t)u(t),\ x(t_0) = x_0, \tag{2.97}$$

is given by the map $x : \mathbb{R} \to \mathbb{R}^n$, defined,

$$x(t) = \Phi(t,t_0)x_0 + \int_{t_0}^{t} \Phi(t,\tau)B(\tau)u(\tau)d\tau, \tag{2.98}$$

where Φ is the state transition matrix with respect to $A(\cdot)$.

Proof Our proof follows by direct differentiation. Let D be the shared discontinuity set of $A(\cdot)$, $B(\cdot)$, and $u(\cdot)$. It follows from the uniform convergence property that $\Phi(\cdot,\cdot)$ is continuously differentiable outside of the discontinuity set D. Let $t \in \mathbb{R} \setminus D$. By definition of the state transition matrix, it follows that

$$\frac{d}{dt}x(t) = \frac{d}{dt}\Phi(t,t_0)x_0 + \frac{d}{dt}\int_{t_0}^{t} \Phi(t,\tau)B(\tau)u(\tau)d\tau \tag{2.99}$$

$$= A(t)\Phi(t,t_0)x_0 + \Phi(t,t)B(t)u(t) + \int_{t_0}^{t} \frac{\partial}{\partial t}\Phi(t,\tau)B(\tau)u(\tau)d\tau \tag{2.100}$$

$$= A(t)\Phi(t,t_0)x_0 + B(t)u(t) + \int_{t_0}^{t} A(t)\Phi(t,\tau)B(\tau)u(\tau)d\tau \tag{2.101}$$

$$= A(t)\Big[\Phi(t,t_0)x_0 + \int_{t_0}^{t} \Phi(t,\tau)B(\tau)u(\tau)d\tau\Big] + B(t)u(t) \tag{2.102}$$

$$= A(t)x(t) + B(t)u(t). \tag{2.103}$$

Note that here, since $[t_0, t_1] \cap D$ contains a finite number of points (by definition of piecewise continuity), application of the Leibniz rule is justified. Thus, the solution satisfies the differentiation property. Also by definition of the state transition matrix,

$$x(t_0) = \Phi(t_0, t_0)x_0 + \int_{t_0}^{t_0} \Phi(t_0, \tau)B(\tau)u(\tau)d\tau = Ix_0 + 0 = x_0. \tag{2.104}$$

Therefore, the initial condition is also satisfied. This establishes the *existence* of solutions to the initial value problem. Now, we verify uniqueness using the Gronwall inequality. Suppose $\hat{x} : \mathbb{R} \to \mathbb{R}^n$ is another solution to the initial value problem. Then, both x and $\hat{x}$ must satisfy,

$$x(t) = x_0 + \int_{t_0}^{t} A(\tau)x(\tau) + B(\tau)u(\tau)d\tau \tag{2.105}$$

$$\hat{x}(t) = x_0 + \int_{t_0}^{t} A(\tau)\hat{x}(\tau) + B(\tau)u(\tau)d\tau, \ \forall t \in \mathbb{R}. \tag{2.106}$$

Subtracting and taking the norm, one gets,

$$\|x(t) - \hat{x}(t)\| = \left\| \int_{t_0}^{t} A(\tau)(x(\tau) - \hat{x}(\tau)d\tau \right\| \tag{2.107}$$

$$\leq \left| \int_{t_0}^{t} \|A(\tau)\| \, \|x(\tau) - \hat{x}(\tau)\| \, d\tau \right|. \tag{2.108}$$

Applying the Gronwall inequality, it follows that $\|x(t) - \hat{x}(t)\| = 0$, for all $t \in \mathbb{R}$. We conclude that $x = \hat{x}$, and that solutions to the IVP are unique. □

This result yields the final piece in the puzzle in establishing that a continuous-time, linear time-varying system representation yields a continuous-time, linear I/O dynamical system. Since the existence and uniqueness theorem above does the bulk of the work, we leave the details of the following theorem to the reader.

Theorem 2.6 (LTV System Representations Determine Linear I/O Systems) *Consider a continuous-time, LTV system representation $(A(\cdot), B(\cdot), C(\cdot), D(\cdot))$. This representation determines a linear I/O system $\mathcal{D}$, comprised of the following data:*

1. *Time Set: $\mathcal{T} = \mathbb{R}$.*
2. *Spaces: $\mathcal{U} = PC(\mathbb{R}, \mathbb{R}^m)$, $\mathcal{Y} = PC(\mathbb{R}, \mathbb{R}^p)$, and $\Sigma = \mathbb{R}^n$.*
3. *State Transition Map: the state transition map is computed,*

$$\varphi(t_1, t_0, x_0, u(\cdot)) = \Phi(t_1, t_0)x_0 + \int_{t_0}^{t_1} \Phi(t_1, \tau)B(\tau)u(\tau)d\tau. \tag{2.109}$$

4. *Readout Map: the readout map is computed,*

$$r(t, x, u) = C(t)x + D(t)u. \tag{2.110}$$

5. *I/O Map: the I/O map is computed,*

$$\rho(t_1, t_0, x_0, u(\cdot)) = C(t_1)\Phi(t_1, t_0)x_0 + C(t_1)\int_{t_0}^{t_1} \Phi(t_1, \tau)B(\tau)u(\tau)d\tau + D(t_1)u(t_1). \tag{2.111}$$

Proof See Problem 2.7. □

To conclude, we make the following important observation. The evolution of the output of any continuous-time, LTV system representation can be *decomposed* into the sum of the zero-input and zero-state components:

$$\rho(t_1, t_0, x_0, u(\cdot)) = \underbrace{C(t_1)\Phi(t_1, t_0)x_0}_{\text{Zero-Input Response}} + \underbrace{C(t_1)\int_{t_0}^{t_1} \Phi(t_1, \tau)B(\tau)u(\tau)d\tau + D(t_1)u(t_1)}_{\text{Zero-State Response}}. \tag{2.112}$$

Note that the zero-input and zero-state responses are sometimes referred to as the *free* and *forced* responses, respectively. This tells us that the response of any linear, time-varying system to an input signal has a component due to the initial condition and a separate component due to the input.

2.2.2 Solutions of Discrete-Time Linear Systems

After taking on a monumental challenge in the desert of continuous-time systems, it's now time to relax in the oasis of discrete-time systems. Kick back, grab your favorite normed vector space, and prepare to be relieved by a *substantially* easier theory.

In this section, we'll work through the process of proving that discrete-time, linear time-varying systems define discrete-time linear I/O systems. Why is the theory in the discrete-time case so much easier than in the continuous-time case? Let's take a quick look at the state equation for the discrete-time, linear time-varying system and see what we find. We have,

$$x[k+1] = A[k]x[k] + B[k]u[k]. \tag{2.113}$$

That is, given $x[k]$ and $u[k]$, we can immediately calculate $x[k+1] = A[k]x[k] + B[k]u[k]$. This means that, for any initial condition $x[k_0] = x_0$ and input signal $u[\cdot] : \mathbb{Z} \to \mathbb{R}^m$, we can recursively solve for $x[k]$, $k \geq k_0$. This makes the problem of existence and uniqueness of solutions trivial in discrete-time.

Despite this great simplification of the discrete-time initial value problem, we'll still find it fruitful to examine closely the structure of solutions to a discrete-time system—just because we can write down a solution directly from the recurrence doesn't mean there isn't more at play! Interestingly, we'll find that a state transition matrix similar to that of the continuous-time case also appears in the discrete-time case.

In the remainder of this section, we'll follow the same general procedure as in the continuous-time case. Here, we'll leave many of the results as exercises or problems, due to their simpler analytical nature.

2.2.2.1 Defining Solutions to Discrete-Time Systems

As with the continuous-time case, we begin by specifying a formal definition of a solution to a discrete-time initial value problem. In this case, since the time set is discrete and the state equation is a recurrence relation, we won't need to worry about regularity conditions such as piecewise continuity. This is reflected in the simpler form of the definition.

Definition 2.20 (Solution to Discrete-Time Recurrence) Let $A[\cdot] : \mathbb{Z} \to \mathbb{R}^{n\times n}$, $B[\cdot] : \mathbb{Z} \to \mathbb{R}^{n\times m}$, and $u[\cdot] : \mathbb{Z} \to \mathbb{R}^m$ be sequences. For $x_0 \in \mathbb{R}^n$ and $k_0 \in \mathbb{Z}$, a solution to the discrete-time recurrence,

$$x[k+1] = A[k]x[k] + B[k]u[k],\ x[k_0] = x_0, \tag{2.114}$$

is a sequence $x[\cdot] : \mathbb{Z}_{\geq k_0} \to \mathbb{R}^n$, satisfying:

1. Initial condition: $x[k_0] = x_0$.
2. Recurrence: For all $k \geq k_0$, $x[k+1] = A[k]x[k] + B[k]u[k]$.

We observe that the solution to a discrete-time initial value problem has exactly the structure we expect! It's important to note that—instead of the solution being defined on all of $\mathbb{Z}$, solutions are defined as sequences starting at a time k_0.

The reasoning behind this is essentially as follows: for $k < k_0$, it's possible that the trajectory leading to $x[k_0] = x_0$ is *not* uniquely defined (can you think of the reason why? We'll provide an answer below). For these reasons, we restrict the definition of a solution to be for $k \geq k_0$. Shortly, we'll describe some conditions that let us extend the domain of the definition of a solution to all of $\mathbb{Z}$.

2.2.2.2 A Matrix Initial Value Problem & The State Transition Matrix

In order to uncover the structure underlying the discrete-time initial value problem, we again take the approach of studying a *matrix* initial value problem. Fortunately, as mentioned above, the existence and uniqueness of solutions are no longer a concern! As such, we can directly jump to the definition of a state transition matrix.

Definition 2.21 (Discrete-Time State Transition Matrix) Consider a sequence $A[\cdot] : \mathbb{Z} \to \mathbb{R}^{n\times n}$. The (discrete-time) state transition matrix with respect to $A[\cdot]$ is a map

$$\Phi[\cdot,\cdot] : \mathbf{T} \to \mathbb{R}^{n\times n},\ \mathbf{T} := \{(k, k_0) \in \mathbb{Z} \times \mathbb{Z} : k \geq k_0\}, \tag{2.115}$$

such that for all $k_0 \in \mathbb{Z}$, $\Phi[\cdot, k_0]$ is the solution to the discrete-time, matrix initial value problem

$$X[k+1] = A[k]X[k],\ X[k_0] = I. \tag{2.116}$$

Thus, we define the state transition matrix Φ in *exactly* the same way as for the continuous-time case—as a solution to a matrix initial value problem defined by $A[\cdot]$, with an initial condition given by the identity matrix. Here, however, in order to get a well-defined solution, we must define the domain of Φ such that $k \geq k_0$. We'll see shortly how this assumption on the domain can be relaxed when the sequence $\{A[k]\}_{k\in\mathbb{Z}}$ has all nonsingular elements.

Proposition 2.7 (Structure of the Discrete-Time State Transition Matrix) *Consider a sequence $A[\cdot] : \mathbb{Z} \to \mathbb{R}^{n\times n}$. For any $k_0 \in \mathbb{Z}$ and $k \geq k_0$, the state transition matrix $\Phi[k, k_0]$ with respect to $A[\cdot]$ is computed*

$$\Phi[k_0, k_0] = I \tag{2.117}$$
$$\Phi[k+1, k_0] = A[k]\Phi[k, k_0]. \tag{2.118}$$

Exercise 2.6 Prove Proposition 2.7.

This result gives a method of calculating the state transition matrix for a given k_0 and all $k \geq k_0$. Why can't we calculate the state transition matrix for all $k \in \mathbb{Z}$? In the event where the matrix $A[k]$ is not invertible, we lose uniqueness in the definition of Φ—thus, for $k < k_0$, Φ might be ill-defined. In the event where $A[k]$ is nonsingular (i.e. invertible), however, we can make the following conclusion.

Proposition 2.8 (State Transition Matrix for Invertible A) *Consider a sequence $A[\cdot] : \mathbb{Z} \to \mathbb{R}^{n\times n}$, in which each $A[k]$ is nonsingular. For such a sequence, the state transition matrix Φ can be uniquely defined on all of $\mathbb{Z} \times \mathbb{Z}$.*

Exercise 2.7 Prove Proposition 2.8. What can go wrong if $A[k]$ is singular? Provide an example.

Finally, we show that the discrete-time state transition matrix satisfies a composability property. Here, due to the risk of singular $A[k]$, we *cannot* prove a composability property for all k_0, k_1, k_2—we are restricted to $k_0 \leq k_1 \leq k_2$. For this same reason, we are not guaranteed that the discrete-time state transition matrix is invertible for all $k_0, k_1 \in \mathbb{Z}$.

Proposition 2.9 (Composability of the Discrete-Time State Transition Matrix) *Consider a sequence $A[\cdot] : \mathbb{Z} \to \mathbb{R}^{n\times n}$. The state transition matrix Φ with respect to $A[\cdot]$ then satisfies $\Phi[k_2, k_0] = \Phi[k_2, k_1]\Phi[k_1, k_0]$, for all $k_0 \leq k_1 \leq k_2 \in \mathbb{Z}$.*

2.2.2.3 Solutions to the Discrete-Time, LTV Recurrence

Using the state transition matrix, we can find the unique solution to the discrete-time recurrence defined by the state equation of the discrete-time, LTV representation.

Theorem 2.7 (Solutions to Discrete-Time, LTV Recurrence) *Let $A[\cdot] : \mathbb{Z} \to \mathbb{R}^{n\times n}$, $B[\cdot] : \mathbb{Z} \to \mathbb{R}^{n\times m}$, and $u[\cdot] : \mathbb{Z} \to \mathbb{R}^m$ be sequences. For $x_0 \in \mathbb{R}^n$ and $k_0 \in \mathbb{Z}$, the unique solution to the discrete-time recurrence,*

$$x[k+1] = A[k]x[k] + B[k]u[k],\ x[k_0] = x_0, \tag{2.119}$$

is given by the sequence sequence $x[\cdot] : \mathbb{Z}_{\geq k_0} \to \mathbb{R}^n$, defined,

$$x[k] = \Phi[k, k_0]x_0 + \sum_{j=k_0}^{k-1} \Phi[k, j+1]B[j]u[j]. \tag{2.120}$$

Exercise 2.8 Prove Theorem 2.7 by induction on k. Why is the expression $\Phi[k, j+1]$ well-defined for $j \in [k_0, k-1] \cap \mathbb{Z}$?

Finally, we confirm that discrete-time, linear time-varying representations determine discrete-time linear I/O systems.

Theorem 2.8 (DT-LTV System Representations Determine DT Linear I/O Systems) *Consider a discrete-time, LTV system representation $(A[\cdot], B[\cdot], C[\cdot], D[\cdot])$. This representation determines a discrete-time linear I/O system $\mathcal{D}$, comprised of the data:*

1. *Time Set: $\mathcal{T} = \mathbb{Z}$.*
2. *Spaces: $\mathcal{U} = \{u : \mathbb{Z} \to \mathbb{R}^m\}$, $\mathcal{Y} = \{y : \mathbb{Z} \to \mathbb{R}^p\}$, and $\Sigma = \mathbb{R}^n$.*
3. *State Transition Map: the state-transition map is computed,*

$$\varphi(k_1, k_0, x_0, u[\cdot]) = \Phi[k_1, k_0]x_0 + \sum_{j=k_0}^{k_1-1} \Phi[k_1, j+1]B[j]u[j] \tag{2.121}$$

4. *Readout Map: the readout map is computed,*

$$r(k, x, u) = C[k]x + D[k]u \tag{2.122}$$

5. *I/O Map: The I/O map is computed,*

$$\rho(k_1, k_0, x_0, u[\cdot]) = C[k_1]\Phi[k_1, k_0]x_0 + C[k_1] \sum_{j=k_0}^{k_1-1} \Phi[k_1, j+1]B[j]u[j] + D[k_1]u[k_1]. \tag{2.123}$$

Proof See Problem 2.7. □

As with the case of a continuous-time linear system, we note that the state response of any discrete-time linear, time-varying system representation can be decomposed as the sum,

$$\rho(k_1, k_0, x_0, u[\cdot]) = \underbrace{C[k_1]\Phi[k_1, k_0]x_0}_{\text{Zero-Input Response}} + \underbrace{C[k_1] \sum_{j=k_0}^{k_1-1} \Phi[k_1, j+1]B[j]u[j] + D[k_1]u[k_1]}_{\text{Zero-State Response}}, \tag{2.124}$$

of a zero-input and a zero-state response. As with the continuous-time case, these components are also referred to as the *free* and *forced* responses, respectively.

2.2.3 Further Reading

This section was mainly influenced by [6], [22], and [24]. For an approach to the existence & uniqueness problem that uses a more general existence & uniqueness theorem for differential equations, the interested reader is encouraged to consult [17]. For a treatment of existence & uniqueness of solutions to differential equations with *measurable* data (more general than piecewise continuous), a measure-theoretic treatment of ordinary differential equations is found in Appendix C of [33].

2.2.4 Problems

Problem 2.3 (Transition Matrix Under Change of Variable) Consider a continuous-time linear, time-varying system representation $(A(\cdot), B(\cdot), C(\cdot), D(\cdot))$,

$$\dot{x}(t) = A(t)x(t) + B(t)u(t) \tag{2.125}$$
$$y(t) = C(t)x(t) + D(t)u(t). \tag{2.126}$$

1. Consider an invertible linear transformation $T \in \mathbb{R}^{n\times n}$ and a corresponding change of variables, $z = Tx$. Identify the system representation $(\hat{A}(\cdot), \hat{B}(\cdot), \hat{C}(\cdot), \hat{D}(\cdot))$ for which solutions to,

$$\dot{z}(t) = \hat{A}(t)\hat{z}(t) + \hat{B}(t)u(t) \tag{2.127}$$
$$\hat{y}(t) = \hat{C}(t)\hat{z}(t) + \hat{D}(t)u(t) \tag{2.128}$$

   satisfy $z(t) = Tx(t)$ and $\hat{y}(t) = y(t)$ for all initial conditions x_0 and Tx_0 and piecewise continuous input signals $u(\cdot)$. Conclude that the input to output behavior of the system *does not* depend on changes of state coordinates.
2. Write the state transition matrix $\hat{\Phi}(t, t_0)$ of the transformed system in terms of the state transition matrix $\Phi(t, t_0)$ of the original system and the transformation T.
3. Does the relation you derived in part (2) also hold for a discrete-time system representation? Explain why or why not.

Problem 2.4 (The Intermediate Value Property of the Derivative) In this problem, we formalize the "intermediate value property" of the derivative, which states that the derivative of a function cannot have any jump discontinuities. Recall that the derivative of a scalar function $f : \mathbb{R} \to \mathbb{R}$ is defined,

$$f'(t) = \lim_{\tau \to t} \frac{f(t) - f(\tau)}{t - \tau}. \tag{2.129}$$

It is *not* necessarily the case that the derivative of a differentiable function is continuous! However, we *can* exclude certain types of discontinuities.

1. First, suppose $f : \mathbb{R} \to \mathbb{R}$ is differentiable on an open interval (a, b). Show that if f attains a maximum or minimum value at a point $c \in (a, b)$, then $f'(c) = 0$.
2. Now, suppose $f : \mathbb{R} \to \mathbb{R}$ is differentiable on an open set $A \subseteq \mathbb{R}$ containing an interval $[a, b]$. Suppose $\alpha \in \mathbb{R}$ satisfies $f'(a) < \alpha < f'(b)$. Show there exists a point $c \in (a, b)$ for which $f'(c) = \alpha$. *If you get stuck, consult Chapter 5.2 of [1] for some hints.*
3. Apply the conclusion of part (2) to study the jump discontinuities of the derivative of a function $f : \mathbb{R} \to \mathbb{R}^n$.

Problem 2.5 (Gronwall Inequality) In this problem, we'll walk through a proof of the Gronwall inequality (Lemma 2.3). Recall that the Gronwall inequality is formulated as follows. Let $y, k \in PC(\mathbb{R}, \mathbb{R}_{\geq 0})$ and $\mu \in PC(R, \mathbb{R}_{\geq 0})$, $c \in \mathbb{R}_{\geq 0}$, and $t_0 \in \mathbb{R}$. If for all $t \in \mathbb{R}$, y satisfies,

$$y(t) \leq c + \left| \int_{t_0}^{t} k(\tau)y(\tau)d\tau \right|, \tag{2.130}$$

then for all $t \in \mathbb{R}$,

$$y(t) \leq c \exp \left| \int_{t_0}^{t} k(\tau) d\tau \right|. \tag{2.131}$$

Let's get to work on assembling a proof of this result.

1. Fix times $t, t_0 \in \mathbb{R}$ with $t > t_0$. Define a function,

$$Y(t) = c + \int_{t_0}^{t} k(\tau) y(\tau) d\tau. \tag{2.132}$$

   Argue that $y(t) \leq Y(t)$ for all $t \geq t_0$, and that $Y(t)$ satisfies $\frac{d}{dt} Y(t) = k(t) y(t)$.
2. Prove that,

$$y(t) \leq Y(t) k(t) \exp(- \int_{t_0}^{t} k(\tau) d\tau), \tag{2.133}$$

   and that

$$\frac{d}{dt} [Y(t) \exp(- \int_{t_0}^{t} k(\tau) d\tau)] \leq 0. \tag{2.134}$$

3. Conclude that $y(t) \leq Y(t) \leq c e^{\int_{t_0}^{t} k(\tau) d\tau}$.

Problem 2.6 (Differentiation Under the Integral Sign) Using the limit definition of the derivative, prove Theorem 2.4, the Leibniz rule for differentiation under the integral sign.

Problem 2.7 (LTV System Representations Determine Linear I/O Systems) Above, we stated two Theorems - 2.6 and 2.8 - which claimed that linear time-varying system representations generate linear I/O dynamical systems. Supply proofs of Theorems 2.6 and 2.8.

Problem 2.8 (An Inverse Initial Value Problem) We know that $\Phi(t, t_0)$ is the solution to the initial value problem $\dot{X}(t) = A(t) X(t),\ X(t_0) = I$. In this problem, we'll find out what $\Phi(t_0, t)$ corresponds to.

1. Consider a continuously differentiable, matrix-valued function $M(\cdot) : \mathbb{R} \to \mathbb{R}^{n \times n}$. Suppose for all $t \in \mathbb{R}$, $M(t)$ is nonsingular. Determine an expression for $\frac{d}{dt}[M^{-1}(t)]$ in terms of $\dot{M}(t)$ and $M^{-1}(t)$.
2. Now, consider a matrix $A(\cdot) \in PC(\mathbb{R}, \mathbb{R}^{n \times n})$. Find an expression for the derivative $\frac{\partial}{\partial \tau} \Phi(t, \tau)$ of the state transition matrix Φ with respect to $A(\cdot)$, in terms of $\Phi(t, t_0)$ and $A(t)$. *You may assume the derivative is being taken at a point where $A(\cdot)$ is continuous.*
3. Prove that $\Phi(t_0, t)$ is the unique solution of the matrix initial value problem,

$$\dot{X}(t) = -X(t) A(t),\ X(t_0) = I. \tag{2.135}$$

Problem 2.9 (The Jacobi-Liouville Formula ★ [8]) Above, we showed that the continuous-time state transition matrix is always invertible. Here, we'll provide another proof of this by means of the *Jacobi-Liouville formula*, which explicitly provides a formula

for the determinant of the state transition matrix. In particular, the Jacobi-Liouville formula is,

$$\det \Phi(t, t_0) = \exp\left(\int_{t_0}^{t} \operatorname{tr}(A(\tau)) d\tau \right). \tag{2.136}$$

1. Prove that, for $M \in \mathbb{R}^{n \times n}$ and $\epsilon \in \mathbb{R}$, there exists a continuous function $R : \mathbb{R} \to \mathbb{R}$ for which

$$\det(I + \epsilon M) = 1 + \epsilon \operatorname{tr}(M) + R(\epsilon) \text{ and } \lim_{\epsilon \to 0} \frac{R(\epsilon)}{\epsilon} = 0. \tag{2.137}$$

   Hint: consider working with eigenvalues.
2. Using the determinant formula from (1), show that

$$\frac{d}{dt} \det[\Phi(t, t_0)] = \operatorname{tr}(A(t)) \det[\Phi(t, t_0)]. \tag{2.138}$$

   Hint: Work with the limit definition of the derivative. If you use a Taylor approximation, be rigorous about your use of the remainder term.
3. Conclude the Jacobi-Liouville formula. Using the Jacobi-Liouville formula, provide a proof that $\Phi(t, t_0)$ is invertible for all $(t, t_0) \in \mathbb{R} \times \mathbb{R}$.

Problem 2.10 (Solution of a Matrix Differential Equation [7]) Let $A_1(\cdot), A_2(\cdot)$, and $F(\cdot)$ be elements of $PC(\mathbb{R}, \mathbb{R}^{n \times n})$. Let Φ_i be the state transition matrix of $\dot{x}(t) = A_i(t)x(t)$ for $i = 1, 2$. Show that the solution of the matrix differential equation:

$$\dot{X}(t) = A_1(t)X(t) + X(t)A_2^{\top}(t) + F(t),\ X(t_0) = X_0, \tag{2.139}$$

is given by,

$$X(t) = \Phi_1(t, t_0)X_0\Phi_2^{\top}(t, t_0) + \int_{t_0}^{t} \Phi_1(t, \tau)F(\tau)\Phi_2^{\top}(t, \tau)d\tau. \tag{2.140}$$

Is this the unique solution of the matrix differential equation? Back up your answer with a proof or disproof.

Problem 2.11 (A Special State Transition Matrix) Consider a piecewise continuous matrix $A \in PC(\mathbb{R}, \mathbb{R}^{n \times n})$, and let Φ denote the state transition matrix of $\dot{x}(t) = A(t)x(t)$. If for every $(\tau, t) \in \mathbb{R} \times \mathbb{R}$, one has,

$$A(t)\left(\int_{\tau}^{t} A(\eta)d\eta \right) = \left(\int_{\tau}^{t} A(\eta)d\eta \right) A(t), \tag{2.141}$$

prove using the Peano-Baker series that,

$$\Phi(t, \tau) = \exp\left(\int_{\tau}^{t} A(\eta)d\eta \right) = \sum_{k=0}^{\infty} \frac{1}{k!} \left(\int_{\tau}^{t} A(\eta)d\eta \right)^{k}. \tag{2.142}$$

Using this result, calculate the state transition matrix associated to the matrix,

$$A(t) = \begin{bmatrix} 0 & 0 \\ t & 0 \end{bmatrix}. \tag{2.143}$$

2.3 Solutions of Linear, Time-Invariant Systems

In the previous section, we studied the structure of solutions to linear, time-varying systems. Now, we'll specialize to the linear, time-*invariant* case. Since the set of linear, time-invariant systems are a *strict subset* of the set of linear, time-varying systems, we can directly apply all of the results we developed in the previous section to the LTI case. Thus, we immediately have access to all of our results concerning the state transition matrix, existence & uniqueness of solutions, the structure of solutions, and so on.

What, then, do we aim to accomplish by focusing on the linear, time-invariant case? Recall that in the previous section, we established that the state transition matrix of a linear, time-varying continuous-time system can be computed via the *Peano-Baker series*,

$$\Phi(t, t_0) = I + \int_{t_0}^{t} A(\tau)d\tau + \int_{t_0}^{t} A(\tau) \int_{t_0}^{\tau} A(\tau')d\tau' d\tau + ..., \tag{2.144}$$

a rather unwieldy infinite series with no immediately apparent simplifications. Likewise, we showed that the state transition matrix of a linear, time-varying discrete-time system is computed via the similarly unrevealing product,

$$\Phi[k, k_0] = A[k-1]A[k-2] \cdot ... \cdot A[k_0+1]A[k_0]. \tag{2.145}$$

Let's study how these expressions simplify in the time-invariant case. We recall that in the time-invariant case, the matrices $A(\cdot)$ and $A[\cdot]$ cease to be functions of time, and are specified by a constant matrix $A \in \mathbb{R}^{n \times n}$. Let's substitute such a matrix into the Peano-Baker series. A little bit of computation yields,

$$\Phi(t, t_0) = I + A(t - t_0) + \frac{A^2(t - t_0)^2}{2} + \frac{A^3(t - t_0)^3}{6} + ... \tag{2.146}$$

Interestingly, such an expression seems to mirror the power series,

$$\sum_{k=1}^{\infty} \frac{A^k(t - t_0)^k}{k!}. \tag{2.147}$$

If A were a scalar, $A = a \in \mathbb{R}$, this would mean that the Peano-Baker series would *exactly* equal the exponential, $\exp(a(t - t_0))$—amazing! Now, we examine the discrete-time case. Here, the product $A[k-1]A[k-2] \cdot ... \cdot A[k_0]$ simply reduces to the matrix power A^{k-k_0}.

This simple analysis leads us to the following conclusion: compared to the time-varying case, the time-invariant case has a *significant* amount of structure that we can exploit. In particular, by examining properties of power series of matrices (in the continuous-time case) and of exponents of matrices (in the discrete-time case), we can gain significant insight into the behavior of linear, time-invariant systems. Let's begin!

2.3.1 Continuous-Time LTI Systems

We begin our study of linear, time-invariant systems with the continuous-time case. In our brief, expository analysis above, we discovered that the Peano-Baker series formula for the

state transition matrix *seems* to follow the pattern,

$$\Phi(t,t_0) = I + A(t-t_0) + \frac{A^2(t-t_0)^2}{2} + \frac{A^3(t-t_0)^3}{6} + ... \tag{2.148}$$

That is, it appeared as if the state transition matrix for a continuous-time, linear time-invariant system was computable from the power series for the exponential. Is this truly the case? Let's find out! As a first step, we formally define the exponential of a matrix. For the sake of generality, we make this definition for a *complex* matrix.

Definition 2.22 (Matrix Exponential) The matrix exponential on $\mathbb{C}^{n\times n}$ is the mapping $\exp : \mathbb{C}^{n\times n} \to \mathbb{C}^{n\times n}$, mapping a matrix $A \in \mathbb{C}^{n\times n}$ to $\exp(A)$,

$$\exp(A) := I + A + \frac{A^2}{2!} + \frac{A^3}{3!} + ... = \sum_{k=0}^{\infty} \frac{A^k}{k!}. \tag{2.149}$$

Remark 2.22 Here, we'll denote the matrix exponential of $A \in \mathbb{C}^{n\times n}$ either by $\exp(A)$ or e^A. The choice between the two is really a matter of personal preference.

In order to ensure the matrix exponential is *well-defined*, we must ensure that it converges. For the case of $A \in \mathbb{R}^{n\times n}$, this is something we've already proven! Recall that in the previous section, we proved that the Peano-Baker series converges uniformly on any interval $[-t', t']$, where $t' \in \mathbb{R}_{\geq 0}$.

Since the exponential of a real matrix is a special case of the Peano-Baker series, for which $A(\cdot) = A \in \mathbb{R}^{n\times n}$, the exponential of a real matrix enjoys similar convergence properties to the Peano-Baker series. Luckily, none of the convergence properties are lost when generalizing from $A \in \mathbb{R}^{n\times n}$ to $A \in \mathbb{C}^{n\times n}$. We state this fact in the following proposition.

Proposition 2.10 (Convergence of the Matrix Exponential) *Let $A \in \mathbb{C}^{n\times n}$. On any compact interval $[-t', t'] \subseteq \mathbb{R}$, $t' \in \mathbb{R}_{\geq 0}$, the power series,*

$$\exp(At) = \sum_{k=0}^{\infty} \frac{A^k t^k}{k!}, \tag{2.150}$$

converges uniformly.

Remark 2.23 By taking $t' \geq 1$ and $t = 1$, one may confirm by application of Proposition 2.10 that the exponential $\exp(A)$ converges for any $A \in \mathbb{C}^{n\times n}$.

Exercise 2.9 Prove Proposition 2.10 directly without use of the Peano-Baker series. Note that the Weierstrass M-test *still holds* on complex-valued, finite-dimensional vector spaces!

Now that we've established its convergence, we know that the matrix exponential is a well-defined quantity. Thus, we can move on to study other properties of the exponential. In the following proposition, we summarize a few important, algebraic properties of the matrix exponential.

Proposition 2.11 (Basic Properties of the Matrix Exponential) *Consider a matrix $A \in \mathbb{C}^{n\times n}$. The exponential of A satisfies the following algebraic properties:*

1. *Eigenvalue-eigenvector pairs: for an eigenvalue-eigenvector pair (λ, v) of A, (e^λ, v) is an eigenvalue-eigenvector pair of $\exp(A)$.*

2. Determinant: The determinant of the exponential is computed $\det(\exp A) = e^{\operatorname{tr} A}$.
3. Invertibility: The matrix exponential is invertible, with $(\exp(A))^{-1} = \exp(-A)$.

Proof See Problem 2.14. □

Now that we've defined the matrix exponential and some of its basic, algebraic properties, we consider its applications in the study of linear, time-invariant systems. Recall that—when motivating the matrix exponential—we derived the formula for the exponential by substituting a constant matrix into the formula for the Peano-Baker series. Since the Peano-Baker series is used to compute the state transition matrix of a linear, time-varying system, it stands to reason that the matrix exponential can be used to compute the state transition matrix of a linear, time-invariant system.

Proposition 2.12 (Continuous-Time, LTI State Transition Matrix) *Consider a fixed matrix* $A \in \mathbb{R}^{n\times n}$. *The continuous-time state transition matrix with respect to* A *is computed,*

$$\Phi(t, t_0) = \exp(A(t - t_0)). \tag{2.151}$$

That is, $\Phi(t, t_0) = \exp(A(t - t_0))$ *is the unique solution to the matrix initial value problem,*

$$\dot{X}(t) = AX(t),\ X(t_0) = I. \tag{2.152}$$

Remark 2.24 In this initial value problem, $A(\cdot) = A$ is constant. Since constant functions are *always continuous*, the discontinuity set of $A(\cdot) = A$ is empty. Thus, we must verify that the derivative condition $\dot{X}(t) = AX(t)$ holds for all $t \in \mathbb{R}$.

Proof There are a few ways we can prove this result. First, we could verify that the exponential satisfies the initial value problem by showing the exponential $\exp(A(t-t_0))$ equals the Peano-Baker series. Secondly, we can proceed by direct differentiation. Here, we'll take the direct differentiation approach in order to get a feel for the definition of the exponential—we leave the (more straightforward) Peano-Baker series approach as an exercise below.

Let's show via direct differentiation that $\exp(A(t-t_0))$ is the solution to the given initial value problem. First, let's examine the derivatives of the partial sums of the exponential. Fix a pair of times $t, t_0 \in \mathbb{R}$. We have, for $p \in \mathbb{Z}_{\geq 0}$,

$$\frac{d}{dt}\sum_{k=0}^{p}\frac{A^k(t-t_0)^k}{k!} = \sum_{k=0}^{p}\frac{d}{dt}\frac{A^k(t-t_0)^k}{k!} = \sum_{k=1}^{p}\frac{A^k(t-t_0)^{k-1}}{(k-1)!} = A\sum_{k=0}^{p-1}\frac{A^k(t-t_0)^k}{k!}. \tag{2.153}$$

Let's use uniform convergence to pass to the limit. Fix a time $t' \geq 0$ for which $t, t_0 \in (-t', t')$. We know that $\exp(A(t-t_0))$ converges uniformly on $[-t', t']$, and that the sequence of derivatives above converges uniformly on $[-t', t']$. This implies,

$$\frac{d}{dt}\lim_{p\to\infty}\sum_{k=0}^{p}\frac{A^k(t-t_0)^k}{k!} = \lim_{p\to\infty}\sum_{k=0}^{p}\frac{d}{dt}\frac{A^k(t-t_0)^k}{k!} = A\sum_{k=0}^{\infty}\frac{A^k(t-t_0)^k}{k!}. \tag{2.154}$$

We conclude that $\frac{d}{dt}\exp(A(t-t_0)) = A\exp(A(t-t_0))$. Additionally, by definition of the power series, we have $\exp(A(t_0 - t_0)) = \exp(0) = I$. Thus, $\exp(A(t-t_0))$ is a solution to the initial value problem,

$$\dot{X}(t) = AX(t),\ X(t_0) = I. \tag{2.155}$$

By uniqueness of solutions, it follows that $\Phi(t, t_0) = \exp(A(t - t_0))$. □

Exercise 2.10 Provide an alternate proof of Proposition 2.12 by showing the matrix exponential $\exp(A(t - t_0))$ equals the Peano-Baker series defined by A.

Now that we've confirmed that the matrix exponential enables computation of the state transition matrix in the linear, time-invariant case, all of the properties we proved about state transition matrices in the previous section immediately apply to the exponential. With this knowledge, we confirm that a continuous-time, LTI system representation determines a *formal* continuous-time, LTI dynamical system.

Theorem 2.9 (CT-LTI Representation Determines a CT-LTI System) *Consider a continuous-time, LTI system representation (A, B, C, D). This representation determines a continuous-time, linear time-invariant I/O system $\mathcal{D}$, specified by the following data:*

1. *Time set: $\mathcal{T} = \mathbb{R}$.*
2. *Spaces: $\mathcal{U} = PC(\mathbb{R}, \mathbb{R}^m)$, $\mathcal{Y} = PC(\mathbb{R}, \mathbb{R}^p)$, and $\Sigma = \mathbb{R}^n$.*
3. *State transition map: the state transition map is computed,*

$$\varphi(t_1, t_0, x_0, u(\cdot)) = e^{A(t_1 - t_0)} x_0 + \int_{t_0}^{t_1} e^{A(t_1 - \tau)} B u(\tau) d\tau. \tag{2.156}$$

4. *Readout map: the readout map is computed,*

$$r(t, x, u) = Cx + Du \tag{2.157}$$

5. *I/O map: The I/O map is computed,*

$$\rho(t_1, t_0, x_0, u(\cdot)) = Ce^{A(t_1 - t_0)} x_0 + C \int_{t_0}^{t_1} e^{A(t_1 - \tau)} B u(\tau) d\tau + Du(t_1). \tag{2.158}$$

Proof The proof of this result follows directly from application of Theorem 2.6, Proposition 2.12, and verification of the time-invariance property. □

Exercise 2.11 Provide the details of the proof of Theorem 2.9.

2.3.2 Discrete-Time LTI Systems

Now, we undertake a similar procedure for the discrete-time case. As with the continuous-time case, we begin by computing the state transition matrix for a discrete-time, linear time-invariant system.

Proposition 2.13 (Discrete-Time, LTI State Transition Matrix) *Consider a fixed matrix $A \in \mathbb{R}^{n \times n}$. The discrete-time state transition matrix with respect to A is computed,*

$$\Phi[k, k_0] = A^{k - k_0}, \ \forall k \geq k_0. \tag{2.159}$$

This is the unique solution to the matrix recurrence, $X[k + 1] = AX[k]$, $X[k_0] = I$.

Exercise 2.12 Prove Proposition 2.13 by using the formula for the discrete-time, linear time-varying state transition matrix derived in the previous section.

Using this result, we can prove that any discrete-time, LTI system representation induces a *formal* discrete-time, LTI I/O dynamical system. The following result is a direct discrete-time analogue of Theorem 2.9.

Theorem 2.10 (DT-LTI Representation Determines a DT-LTI System) *Consider a discrete-time, LTI system representation (A, B, C, D). This representation determines a discrete-time, linear time-invariant I/O system $\mathcal{D}$, specified by the following data:*

1. *Time Set: $\mathcal{T} = \mathbb{Z}$.*
2. *Spaces: $\mathcal{U} = \{u : \mathbb{Z} \to \mathbb{R}^m\}$, $\mathcal{Y} = \{y : \mathbb{Z} \to \mathbb{R}^p\}$, and $\Sigma = \mathbb{R}^n$.*
3. *State Transition Map: the state transition map is computed,*

$$\varphi(k_1, k_0, x_0, u[\cdot]) = A^{k_1 - k_0} x_0 + \sum_{j=k_0}^{k_1 - 1} A^{k_1 - j - 1} B u[j]. \tag{2.160}$$

4. *Readout Map: the readout map is computed,*

$$r(k, x, u) = Cx + Du \tag{2.161}$$

5. *I/O Map: The I/O map is computed,*

$$\rho(k_1, k_0, x_0, u[\cdot]) = CA^{k_1 - k_0} x_0 + C\left[\sum_{j=k_0}^{k_1 - 1} A^{k_1 - j - 1} B u[j]\right] + Du[k_1]. \tag{2.162}$$

Proof Follows directly from Proposition 2.13, Theorem 2.8, and verification of the time-invariance property. □

Exercise 2.13 Provide the details of the proof of Theorem 2.10.

2.3.3 The Jordan Canonical Form

Let's take stock of where we're at in the study of LTI systems. Above, we showed that the state transition matrix of an LTI system is computed,

$$\Phi(t, t_0) = \exp(A(t - t_0)) \qquad \text{(continuous-time)} \tag{2.163}$$

$$\Phi[k, k_0] = A^{k - k_0} \qquad \text{(discrete-time)}. \tag{2.164}$$

Although both formulas represent considerable simplifications over the time-varying case, there's still work do be done. In order to understand how linear, time-invariant systems evolve, we must actually be able to compute the matrix exponential and compute arbitrary powers of A. This ability will be essential when studying the stability of linear, time-invariant system, where we'll require explicit bounds on the size of the matrix exponential and the size of the matrix power.

Let's begin by studying the discrete-time state transition matrix. We know that, for an arbitrary matrix $A \in \mathbb{C}^{n\times n}$, there's no easy way to directly compute a power A^k for any given k. Instead of directly computing a power of A, what we might like to do is to compute a power of a *transformed* version of A, where the transformed version is easier to work with. The following lemma suggests that matrix transformations behave well under exponents.

Lemma 2.4 (Similarity Transform and Matrix Power) *Let $A \in \mathbb{C}^{n\times n}$. For any invertible matrix $T \in \mathbb{C}^{n\times n}$ and any $k \in \mathbb{Z}_{\geq 0}$, $(T^{-1}AT)^k = T^{-1}A^kT$.*

Remark 2.25 A transformation of the form $T^{-1}AT$ is called a *similarity transform* of A. If there exists a T for which $T^{-1}AT = B$, then A and B are said to be *similar matrices*. One may show that similar matrices share the same characteristic polynomial and eigenvalues.

Proof We'll prove this by induction on k. For the base case, $k = 0$, one has, $(T^{-1}AT)^0 = I$ and $T^{-1}A^0T = T^{-1}T = I$. Thus, the base case holds. Now, assume for induction that the result is true for $k > 0$. One has,

$$(T^{-1}AT)^{k+1} = (T^{-1}AT)^kT^{-1}AT = T^{-1}A^kTT^{-1}AT = T^{-1}A^{k+1}T. \tag{2.165}$$

Thus, by induction on k, the proposed result holds. □

Lemma 2.4 tells us that, if we can transform a matrix A into a form whose exponents are easy to compute, we can easily recover the exponents of A. We illustrate this with the case of a diagonal matrix.

Proposition 2.14 (Power of a Diagonalizable Matrix) *Suppose $A \in \mathbb{C}^{n\times n}$ is diagonalizable. That is, there exists an invertible matrix $T \in \mathbb{C}^{n\times n}$ for which $T^{-1}AT = D = \operatorname{diag}(\lambda_1, ..., \lambda_n)$. Then, for any $k \in \mathbb{Z}_{\geq 0}$, A^k is computed,*

$$A^k = T\operatorname{diag}(\lambda_1^k, ..., \lambda_n^k)T^{-1}. \tag{2.166}$$

Proof First, one may verify through an induction argument that, for a diagonal matrix $D = \operatorname{diag}(\lambda_1, ..., \lambda_k)$, $D^k = \operatorname{diag}(\lambda_1^k, ..., \lambda_n^k)$ for any $k \in \mathbb{Z}_{\geq 0}$. If there exists an invertible matrix T for which $T^{-1}AT = D$, one has that $A = TDT^{-1}$. Applying Lemma 2.4 and the formula for the exponential of a diagonal matrix, it follows that, for $k \in \mathbb{Z}_{\geq 0}$,

$$A^k = TD^kT^{-1} = T\operatorname{diag}(\lambda_1^k, ..., \lambda_n^k)T^{-1}, \tag{2.167}$$

which is the desired result. □

We conclude that any nonnegative exponent of a diagonalizable matrix is easy to compute. To find any (nonnegative) exponent of a diagonalizable matrix A, all we need is the matrix T which diagonalizes A, the inverse of T, and the power of the diagonalization of A (which is easy to compute). This sketches out a basic technique—*transform and compute*—for calculating the state transition matrix in the discrete-time case. Now, we study the continuous-time case.

How might we calculate $\exp(At)$ for a given matrix A? An initial guess is that $[\exp(At)]_{ij} = \exp([A]_{ij}t)$, i.e. that taking the exponential of a matrix is the *same thing* as taking a term-by term exponential of the entries of the matrix. However, a simple example shows that this is certainly *not* the case.

Exercise 2.14 (Matrix Exponential is not Element-Wise) Show that for the 2×2 identity matrix, I_2,

$$\exp\left(\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix} t\right) \neq \begin{bmatrix}e^t & e^0\\ e^0 & e^t\end{bmatrix}. \tag{2.168}$$

Now that we've ruled out the only "obvious" guess for calculating the matrix exponential, we must come up with some more clever techniques for computing the exponential. Let's follow a similar process to the discrete-time case—*transform and compute.* The first step in doing this is to verify that the matrix exponential "behaves well" under transformations.

Proposition 2.15 (Similarity Transform and Matrix Exponential) *Consider a matrix $A \in \mathbb{C}^{n\times n}$. For any invertible matrix $T \in \mathbb{C}^{n\times n}$,*

$$\exp(T^{-1}AT) = T^{-1}\exp(A)T. \tag{2.169}$$

Proof Recall from Lemma 2.4 that, for all $k \in \mathbb{Z}_{\geq 0}$, $(T^{-1}AT)^k = T^{-1}A^kT$. Applying this result to the partial sums of the exponential, we have,

$$\sum_{k=0}^{p}\frac{(T^{-1}AT)^k}{k!} = \sum_{k=0}^{p}\frac{T^{-1}A^kT}{k!} = T^{-1}\Big(\sum_{k=0}^{p}\frac{A^k}{k!}\Big)T. \tag{2.170}$$

Since matrix multiplication is a continuous operation, we can pass to the limit as $p \to \infty$. This yields,

$$\exp(T^{-1}AT) = \sum_{k=0}^{\infty}\frac{(T^{-1}AT)^k}{k!} = T^{-1}\Big(\sum_{k=0}^{\infty}\frac{A^k}{k!}\Big)T = T^{-1}\exp(A)T, \tag{2.171}$$

which completes the proof. □

Now that we've confirmed that the matrix exponential "behaves well" under transformations, we can find a transformation under which the exponential is easy to compute. As with the case of a matrix power, we'll find that the exponential of a diagonalizable matrix is easy to compute.

Proposition 2.16 (Exponential of a Diagonalizable Matrix) *Consider a diagonalizable matrix $A \in \mathbb{C}^{n\times n}$ and an invertible transformation $T \in \mathbb{C}^{n\times n}$ for which $T^{-1}AT = \operatorname{diag}(\lambda_1, ..., \lambda_n)$. The exponential of such a matrix may be computed,*

$$\exp(At) = T\operatorname{diag}(e^{\lambda_1 t}, ..., e^{\lambda_n t})T^{-1},\ \forall t \in \mathbb{R}. \tag{2.172}$$

Proof First, we compute the exponential of a diagonal matrix, $D = \operatorname{diag}(\lambda_1, ..., \lambda_n)$. Since $D^k = \operatorname{diag}(\lambda_1^k, ..., \lambda_n^k)$, one has that,

$$\exp(Dt) = \sum_{k=0}^{\infty}\frac{(Dt)^k}{k!} = \operatorname{diag}\Big(\sum_{k=0}^{\infty}\frac{(\lambda_1 t)^k}{k!}, ..., \sum_{k=0}^{\infty}\frac{(\lambda_n t)^k}{k!}\Big) = \operatorname{diag}(e^{\lambda_1 t}, ..., e^{\lambda_n t}). \tag{2.173}$$

Now, suppose that $T^{-1}AT = D = \operatorname{diag}(\lambda_1, ..., \lambda_n)$. Then, applying Proposition 2.15,

$$\exp(At) = \exp(TDT^{-1}t) = T\exp(Dt)T^{-1} = T\operatorname{diag}(e^{\lambda_1 t}, ..., e^{\lambda_n t})T^{-1}. \tag{2.174}$$

This completes the proof. □

These results tell us that, as long as the A matrix is diagonalizable, we can compute the state transition matrix in closed form. But, the general problem persists—not every matrix is diagonalizable. For instance, one may show that the matrix,

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, \tag{2.175}$$

is non-diagonalizable. In order to determine a method of calculating the matrix exponential of *any* complex matrix $A \in \mathbb{C}^{n \times n}$, we'll study a (quite fragile) tool called the *Jordan canonical form* (or Jordan form, for short). This is a is a powerful theoretical tool that enables us to find decompositions of non-diagonalizable matrices that are amenable to computing exponents and exponentials.

Before we jump into the theory of the Jordan form, we must make a disclaimer: the Jordan form is an extremely useful tool for *theory* but is extremely impractical for *computation*. Later, we'll illustrate that the Jordan form of a matrix is extremely fragile to numerical perturbations, something that makes it generally unsuitable for computational use.

Let's motivate the Jordan form by examining what worked well about diagonalization in context of the state transition matrix. The key property of diagonalization for computing the state transition matrix is that a diagonal matrix has a *predictable structure* under exponents. That is, a diagonal matrix enjoys the property,

$$\begin{bmatrix} \lambda_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \lambda_n \end{bmatrix}^k = \begin{bmatrix} \lambda_1^k & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \lambda_n^k \end{bmatrix}, \; k \in \mathbb{Z}_{\geq 0}. \tag{2.176}$$

The main problem with diagonalization is that diagonalizability is *too strong* a condition—it's not always possible to transform a matrix A into a diagonal matrix via a similarity transform $D = T^{-1}AT$. What we'd like is a *relaxation* of diagonalizability that applies to any matrix and retains a predictable structure under exponentiation.

Consider the following idea: instead of enforcing a strictly diagonal structure, what if we enforce a *block-diagonal* structure? We recall the definition of a block diagonal matrix.

Definition 2.23 (Block Diagonal Matrix) A matrix $D \in \mathbb{C}^{n \times n}$ is said to be block diagonal if there exist matrices $\Lambda_i \in \mathbb{C}^{n_i \times n_i}$, $i = 1, ..., p$, with $\sum_{i=1}^{p} n_i = n$ and

$$D = \begin{bmatrix} \Lambda_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \Lambda_p \end{bmatrix}, \tag{2.177}$$

where each 0 is an appropriately sized zero matrix. We write $D = \text{blkdiag}(\Lambda_1, ..., \Lambda_p)$.

Since a block diagonal matrix has a similar structure to a diagonal matrix, it enjoys a similar behavior under exponentiation.

Proposition 2.17 (Exponents of Block Diagonal Matrices) *Let $D = \text{blkdiag}(\Lambda_1, ..., \Lambda_p)$ be a block diagonal matrix. For any $k \in \mathbb{Z}_{\geq 0}$, $D^k = \text{blkdiag}(\Lambda_1^k, ..., \Lambda_p^k)$,*

$$\begin{bmatrix} \Lambda_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \Lambda_p \end{bmatrix}^k = \begin{bmatrix} \Lambda_1^k & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \Lambda_p^k \end{bmatrix}. \tag{2.178}$$

Exercise 2.15 Verify the claim of Proposition 2.17.

Thus, we observe that—just like a diagonal matrix—a block diagonal matrix has a predictable structure under exponentiation. But, examining the exponent of a block diagonal matrix, we see that it's not "good enough" to just have the block diagonal structure! In order to compute an exponent of a block diagonal matrix, we still need to compute the exponents of the individual blocks contained within the matrix. So, in addition to asking for a block-diagonal structure, we need the blocks themselves to have a "nice" structure for exponentiation. By relaxing a diagonal structure to an "almost-diagonal" structure, the following definition introduces exactly the type of block we will need.

Definition 2.24 (Jordan Block) Let $\lambda \in \mathbb{C}$. The Jordan block of size n corresponding to λ is the matrix $J = \lambda I + N_0$, where I is the $n \times n$ identity matrix and N_0 is an $n \times n$ matrix of ones in the superdiagonal and zeros elsewhere. That is, J is the matrix in which all diagonal entries equal λ and entries just above the diagonal equal 1,

$$J = \begin{bmatrix} \lambda & 1 & 0 & \dots & 0 \\ 0 & \lambda & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \\ 0 & 0 & \dots & \lambda & 1 \\ 0 & 0 & \dots & 0 & \lambda \end{bmatrix}, \; N_0 = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \\ 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & \dots & 0 & 0 \end{bmatrix} \in \mathbb{C}^{n \times n} \tag{2.179}$$

The structure of a Jordan block as a matrix $J = \lambda I + N_0$ makes its exponents particularly easy to compute. We know how to compute any exponent of λI, since λI is diagonal. Now, we show how to compute any exponent of N_0.

Lemma 2.5 (Exponents of N_0) *Consider the $n \times n$ Jordan block with respect to 0, $N_0 \in \mathbb{C}^{n \times n}$. The exponents of N_0 respect the sequence,*

$$N_0 = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \\ 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & \dots & 0 & 0 \end{bmatrix}, \; N_0^2 = \begin{bmatrix} 0 & 0 & 1 & \dots & 0 \\ & \ddots & \ddots & \ddots & \vdots \\ & & 0 & 0 & 1 \\ & & & 0 & 0 \\ 0 & & & & 0 \end{bmatrix}, \dots, \; N_0^n = 0_{n \times n}. \tag{2.180}$$

That is, for each successive exponent, the superdiagonal of ones shifts one step to the right.

Remark 2.26 A square matrix N for which $N^k \neq 0$ for $0 \leq k < n$ and $N^k = 0$ for $k \geq n$ is said to be a *nilpotent matrix* of order n. N_0, defined above, is an example of such a matrix.

Exercise 2.16 Prove Lemma 2.5.

For $N_0 \in \mathbb{C}^{n \times n}$, we can use the "shift to the right" rule to compute N_0^k when $k < n$, and $N_0^k = 0$ for $k \geq n$. Since for any Jordan block J, $J = \lambda I + N_0$, this fact lets us compute an arbitrary exponent of *any* Jordan block in terms of the (known) exponents of N_0.

Proposition 2.18 (Exponents of a Jordan Block) *Consider a Jordan block $J \in \mathbb{C}^{n\times n}$ with respect to λ, $J = \lambda I + N_0$. Then, for all $k \in \mathbb{Z}_{\geq 0}$,*

$$J^k = \sum_{j=0}^{k} \binom{k}{j} \lambda^{k-j} N_0^j. \tag{2.181}$$

Proof This proof follows from simple application of the binomial theorem, which states, for $k \in \mathbb{Z}_{\geq 0}$ and $A, B \in \mathbb{C}^{n\times n}$,

$$(A+B)^k = \sum_{j=0}^{k} \binom{k}{j} A^{k-j} B^j. \tag{2.182}$$

Let's apply this formula for $A = \lambda I$ and $B = N_0$. We have that,

$$(\lambda I + N_0)^k = \sum_{j=0}^{k} \binom{k}{j} (\lambda I)^{k-j} N_0^j = \sum_{j=0}^{k} \binom{k}{j} \lambda^{k-j} N_0^j. \tag{2.183}$$

This is the desired formula. □

Exercise 2.17 Expand the binomial formula to show that for a Jordan block $J = \lambda I + N_0$,

$$J^k = \begin{bmatrix} \lambda^k & \binom{k}{1}\lambda^{k-1} & \binom{k}{2}\lambda^{k-2} & \dots & \\ 0 & \lambda^k & \binom{k}{1}\lambda^{k-1} & \dots & \\ \vdots & \vdots & \ddots & \ddots & \\ 0 & 0 & \dots & \lambda^k & \binom{k}{1}\lambda^{k-1} \\ 0 & 0 & \dots & 0 & \lambda^k \end{bmatrix}. \tag{2.184}$$

Let's summarize what we've found so far. We established that requiring *diagonalizability* is too strong an ask, and that *block diagonalizability* might be a suitable relaxation for computing exponents. Then, we showed that when each block has the structure of a Jordan block, we can compute its exponent. This means that for any block-diagonal matrix,

$$\begin{bmatrix} J_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & J_p \end{bmatrix}, \tag{2.185}$$

in which J_i is a Jordan block with respect to some constant λ_i, we can compute *any* power of the matrix. Thus, we've constructed a non-diagonal matrix form that we can easily exponentiate. This matrix form has the following special name.

Definition 2.25 (Jordan Canonical Form) Consider a matrix $J \in \mathbb{C}^{n\times n}$. J is said to be in Jordan canonical form (JCF) if there exist Jordan blocks $J_i \in \mathbb{C}^{n_i \times n_i}, i = 1, ..., p$ such that $\sum_{i=1}^{p} n_i = n$ and $J = \text{blkdiag}(J_1, ..., J_p)$:

$$J = \begin{bmatrix} J_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & J_p \end{bmatrix} \in \mathbb{C}^{n\times n},\ J_i = \begin{bmatrix} \lambda_i & 1 & 0 & \dots & 0 \\ 0 & \lambda_i & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \\ 0 & 0 & \dots & \lambda_i & 1 \\ 0 & 0 & \dots & 0 & \lambda_i \end{bmatrix} \in \mathbb{C}^{n_i\times n_i}. \tag{2.186}$$

Now that we've established a definition for the Jordan canonical form, we ask the essential question—can we transform any matrix into Jordan canonical form? That is, for any given matrix $A \in \mathbb{C}^{n\times n}$, does there exist an invertible matrix $T \in \mathbb{C}^{n\times n}$ for which $J = T^{-1}AT$ is in Jordan canonical form? Amazingly, the answer to this question is *yes*.

How might we construct the transformation into Jordan form? To start off, let's review how a transformation of a diagonalizable matrix $A \in \mathbb{C}^{n\times n}$ into a diagonal matrix $D \in \mathbb{C}^{n\times n}$ is constructed. In order to diagonalize a matrix A, one finds a linearly independent *basis of eigenvectors* of A—a linearly independent collection $\{v_1, ..., v_n\} \subseteq \mathbb{C}^{n\times n}$ for which $Av_i = \lambda_i v_i$, for some $\lambda_i \in \mathbb{C}$. Then, for such a collection, one has

$$\begin{bmatrix} | & & | \\ v_1 & \dots & v_n \\ | & & | \end{bmatrix}^{-1} A \begin{bmatrix} | & & | \\ v_1 & \dots & v_n \\ | & & | \end{bmatrix} = \begin{bmatrix} \lambda_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \lambda_n \end{bmatrix}. \tag{2.187}$$

Thus, in the event that A is diagonalizable, one may use the eigenvectors of A to construct a matrix T for which $T^{-1}AT$ is diagonal. We conclude that, in the event where A is *not* diagonalizable, we cannot form a basis $\{v_1, ..., v_n\}$ for $\mathbb{C}^n$ of eigenvectors of A. Therefore, we have "too few" linearly independent eigenvectors in the non-diagonalizable case.

In order to construct a transformation of any matrix into a form *close* to a diagonal matrix—the Jordan canonical form—it seems reasonable that we might seek a *generalization* of the concept of an eigenvector. In order to define a *generalized eigenvector*, we start with the definition of an eigenvector. For $A \in \mathbb{C}^{n\times n}$ and $\lambda \in \mathbb{C}$, we recall that a vector $v \in \mathbb{C}^n$ is an eigenvector of A with eigenvalue λ if,

$$Av = \lambda v. \tag{2.188}$$

Rearranging this expression, we find that $Av = \lambda v$ is equivalent to,

$$(A - \lambda I)v = 0. \tag{2.189}$$

An easy way to *generalize* the condition $(A - \lambda I)v = 0$ is to require $(A - \lambda I)^k v = 0$, for some $k \in \mathbb{N}$. This leads to the following definition.

Definition 2.26 (Generalized Eigenspace/Eigenvector) Consider a matrix $A \in \mathbb{C}^{n\times n}$ with eigenvalue $\lambda \in \mathbb{C}$. The *generalized eigenspace* of A corresponding to λ is the space,

$$K_\lambda(A) := \{v \in \mathbb{C}^n : (A - \lambda I)^m v = 0 \text{ for some } m \in \mathbb{N}\}. \tag{2.190}$$

Any nonzero vector $v \in K_\lambda(A)$ is called a *generalized eigenvector* of A corresponding to the eigenvalue λ.

We quickly summarize a number of important properties of generalized eigenspaces.

Proposition 2.19 (Properties of Generalized Eigenspaces) *Let $A \in \mathbb{C}^{n\times n}$ be a matrix with eigenvalue $\lambda \in \mathbb{C}$. The generalized eigenspace $K_\lambda(A)$ satisfies:*

1. *Eigenvector: $K_\lambda(A)$ contains at least one eigenvector of A.*
2. *Subspace: $K_\lambda(A)$ is a subspace.*
3. *Invariance: $v \in K_\lambda(A)$ implies $Av \in K_\lambda(A)$.*
4. *Dimension: if the algebraic multiplicity[4] of λ is m, then $\dim(K_\lambda(A)) = m$.*
5. *Alternate Definition: $K_\lambda(A) = \{v \in V : (T - \lambda I)^{\dim V} = 0\}$.*
6. *Decomposition: For $\lambda_1, ..., \lambda_p \in \mathbb{C}$ the distinct eigenvalues of A, $\mathbb{C}^n$ is decomposed,*

$$\mathbb{C}^n = K_{\lambda_1}(A) \oplus ... \oplus K_{\lambda_n}(A), \tag{2.191}$$

where $\oplus$ represents the direct sum[5] of subspaces.

Proof See Problem 2.20 and the starred, optional section below. □

The *generalized eigenvectors* of a matrix A form a candidate pool of vectors from which we will construct a transformation into Jordan form. Which generalized eigenvectors should we actually pick? To answer this question, we'll perform a cursory analysis of a transformation T which transforms A into a Jordan canonical form. Through this analysis, we'll determine necessary conditions on the generalized eigenvectors which enable a transformation into Jordan form. Suppose $J = T^{-1}AT$ is a Jordan canonical form of A. Then, we must have,

$$TJ = AT \tag{2.192}$$

$$\begin{bmatrix} | & & | \\ V_{J_1} & \dots & V_{J_p} \\ | & & | \end{bmatrix} \begin{bmatrix} J_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & J_p \end{bmatrix} = A \begin{bmatrix} | & & | \\ V_{J_1} & \dots & V_{J_p} \\ | & & | \end{bmatrix}, \tag{2.193}$$

where each $V_{J_i} = [v_{J_i}^{(1)}, ..., v_{J_i}^{(n_i)}] \subseteq \mathbb{C}^{n \times n_i}$ is a block of vectors corresponding to the i'th Jordan block, J_i. Let's focus on a single Jordan block, and see what necessary conditions this structure enforces. Examining the block J_i, we have

$$V_{J_i} J_i = A V_{J_i} \tag{2.194}$$

$$\begin{bmatrix} | & & | \\ v_{J_i}^{(1)} & \dots & v_{J_i}^{(n_i)} \\ | & & | \end{bmatrix} \begin{bmatrix} \lambda_i & 1 & 0 & \dots & 0 \\ 0 & \lambda_i & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \\ 0 & 0 & \dots & \lambda_i & 1 \\ 0 & 0 & \dots & 0 & \lambda_i \end{bmatrix} = \begin{bmatrix} | & & | \\ Av_{J_i}^{(1)} & \dots & Av_{J_i}^{(n_i)} \\ | & & | \end{bmatrix}. \tag{2.195}$$

Now, let's zoom in on an individual column. For $j = 1$, we have that

$$Av_{J_i}^{(1)} = \lambda_i v_{J_i}^{(1)}, \tag{2.196}$$

implying that $v_{J_i}^{(1)}$ must be an *eigenvector* of A with eigenvalue λ_i. For $j > 1$, we have,

---

[4] Recall that the algebraic multiplicity of an eigenvalue λ of A is the number of times it appears as a root of the characteristic polynomial of A. For instance, if a matrix A has the characteristic polynomial $\chi_A(s) = (s - \lambda_1)^{m_1} \cdot ... \cdot (s - \lambda_p)^{m_p}$, the algebraic multiplicity of λ_1 would be m_1.

[5] Let $V_1, ..., V_n$ be subspaces of a vector space V. One says that V is the *direct sum* of $V_1, ..., V_n$, written $V = V_1 \oplus ... \oplus V_n$ if, for any $v \in V$, there exist unique $v_i \in V_i$ for which $v = v_1 + ... + v_n$.

$$\lambda_i v_{J_i}^{(j)} + v_{J_i}^{(j-1)} = A v_{J_i}^{(j)} \tag{2.197}$$

$$v_{J_i}^{(j-1)} = A v_{J_i}^{j} - \lambda_i v_{J_i}^{(j)} \tag{2.198}$$

$$v_{J_i}^{(j-1)} = (A - \lambda_i I) v_{J_i}^{(j)}. \tag{2.199}$$

This suggests that—if we zoom in on each Jordan block—the vectors associated with that block must satisfy a *recurrence* relation in which the first element of the recurrence is an eigenvector of A with eigenvalue λ_i. We define a sequence of vectors satisfying this recurrence as a *chain of generalized eigenvectors*.

Definition 2.27 (Chain of Generalized Eigenvectors) Let $A \in \mathbb{C}^{n\times n}$ and $\lambda \in \mathbb{C}$ be an eigenvalue of A. A sequence of vectors $\{v^i\}_{i=1}^m \subseteq \mathbb{C}^n$ satisfying the properties,

1. Linear independence: the set $\{v^i\}_{i=1}^m$ is linearly independent,
2. Recurrence: for $(A - \lambda I)v^1 = 0$ and $(A - \lambda I)v^i = v^{i-1}$ for $i = 2, ..., m$,

is called a *chain of generalized eigenvectors* of length m, corresponding to λ. A chain is said to be *maximal* if it cannot be extended while respecting (1) and (2).

Definition 2.27 suggests that a sequence of vectors satisfying properties (1) and (2) should be *generalized eigenvectors*. Let's confirm that this is the case before moving on.

Lemma 2.6 (Chains are Composed of Generalized Eigenvectors) *Let $A \in \mathbb{C}^{n\times n}$ and $\lambda \in \mathbb{C}$ be an eigenvalue of A. Any chain of generalized eigenvectors corresponding to λ is composed of generalized eigenvectors belonging to $K_\lambda(A)$.*

Proof Let $\{v^i\}_{i=1}^m$ be a chain of generalized eigenvectors of A, corresponding to λ. We will prove that $\{v^i\}_{i=1}^m \subseteq K_\lambda(A)$ by induction. We know that v^1 is an eigenvector of A with eigenvalue λ, which implies $v^1 \in K_\lambda(A)$.

Suppose for induction that, for some $i < m$, v^i is an element of $K_\lambda(A)$. We will show v^{i+1}, which satisfies $(A - \lambda I)v^{i+1} = v^i$, also belongs to $K_\lambda(A)$. Since v^i is a generalized eigenvector of A, there exists a $k > 0$ for which $(A - \lambda I)^k v^i = 0$. Then, one has,

$$(A - \lambda I)v^{i+1} = v^i \tag{2.200}$$

$$(A - \lambda I)^{k+1} v^{i+1} = (A - \lambda I)^k v^i = 0. \tag{2.201}$$

We conclude that $v^{i+1} \in K_\lambda(A)$. Thus, each element of the chain belongs to $K_\lambda(A)$. □

We've now completed all of the setup required to construct a transformation into the Jordan form. We introduced *generalized eigenvectors*, which encompassed a wider class of vectors than (normal) eigenvectors, and identified special sets of generalized eigenvectors, termed *chains*, which were deemed necessary to construct a transformation into Jordan form.

In order to complete the Jordan form story, we must confirm that we can construct a basis for $\mathbb{C}^n$ consisting of chains of generalized eigenvectors, and must prove that such a basis does indeed produce the desired transformation into Jordan canonical form.

Theorem 2.11 (Transformation into Jordan Canonical Form) *Let $A \in \mathbb{C}^{n\times n}$. Suppose A has k linearly independent eigenvectors, $v_1, ..., v_k \in \mathbb{C}^n$. Let $\{v_j^i\}_{i=1}^{m_j}$ be the maximal Jordan chain with initial vector $v_j^1 = v_j$, $j = 1, ..., k$. Then, the set,*

$$\beta = \{v_1^i\}_{i=1}^{m_1} \cup ... \cup \{v_k^i\}_{i=1}^{m_k}, \tag{2.202}$$

forms a basis for $\mathbb{C}^n$*. For* $T = [v_1^1, ..., v_k^{m_k}] \in \mathbb{C}^{n\times n}$*, the matrix* $J = T^{-1}AT$ *is in Jordan canonical form.*

As the proof of this result is rather involved, we defer it to the starred, optional subsection below—the proof can be skipped without loss of continuity.

2.3.3.1 Computing the Jordan Canonical Form

Theorem 2.11 tells us that, for any matrix $A \in \mathbb{C}^{n\times n}$, we can compute a transformation T for which $T^{-1}AT$ is in Jordan canonical form. What's more, the theorem statement provides a method of *constructing* this transformation. We summarize the proposed method of constructing the transformation into Jordan form with the following algorithm.

Corollary 2.1 (Algorithm for Computing the Jordan Form) *Let* $A \in \mathbb{C}^{n\times n}$*. The following procedure outlines a technique for computing a Jordan form of* A*.*

1. *Eigenvalues: identify all distinct eigenvalues,* $\lambda_1, ..., \lambda_p \in \mathbb{C}$ *of* A*.*
2. *Eigenvectors: for each eigenvalue* λ_i*, find a maximal collection of linearly independent eigenvectors,* $v_{i,1}, ..., v_{i,k_i}$ *associated to* λ_i*.*
3. *Jordan chains: for each eigenvector* $v_{i,j}$ *from step (2), compute the maximal Jordan chain starting at* $v_{i,j}$*. Repeat for all eigenvalues and eigenvectors.*
4. *Basis: construct an ordered basis for* $\mathbb{C}^n$ *by taking the unions of all Jordan chains from step (3), preserving the order of each Jordan chain.*
5. *Matrix: construct a transformation matrix* T*, whose columns are the basis vectors from step (4). Be careful to preserve the order of the basis vectors!*
6. *Transformation: compute* $T^{-1}AT$ *to find the Jordan form of* A*.*

Proof Immediate from Theorem 2.11. □

We illustrate the use of this algorithm with a few examples.

Example 2.2 Consider the matrix,

$$A = \begin{bmatrix} 3 & 0 & 0 \\ 1 & 3 & 0 \\ 0 & 0 & 3 \end{bmatrix}. \tag{2.203}$$

Let's work out the steps of the procedure.

1. Eigenvalues: it's apparent from the lower triangular structure of this matrix that the only eigenvalue of A is $\lambda = 3$.
2. Eigenvectors: next, we compute all linearly independent eigenvectors associated with the eigenvalue $\lambda = 3$. By inspection of the matrix, we find,

$$v_{3,1} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \; v_{3,2} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}. \tag{2.204}$$

   We cannot find any more linearly independent eigenvectors for $\lambda = 3$. Thus, we move on to step (3).

3. Jordan chains: now, we compute the maximal Jordan chains starting at $v_{3,1}$ and $v_{3,2}$. Let's start with $v_{3,1}$. We wish to solve for $v_{3,1}^2$ for which

$$(A - 3I)v_{3,1}^2 = v_{3,1} \tag{2.205}$$

$$\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} v_{3,1}^2 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \tag{2.206}$$

From this equality, it's clear that no such $v_{3,1}^2$ exists. Thus, the Jordan chain starting at $v_{3,1}$ is already maximal. Let's instead try the Jordan chain starting at $v_{3,2}$. We require,

$$(A - 3I)v_{3,2}^2 = v_{3,2} \tag{2.207}$$

$$\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} v_{3,2}^2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}. \tag{2.208}$$

Selecting the vector,

$$v_{3,2}^2 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \tag{2.209}$$

accomplishes this.

4. Basis: now, we define a basis by taking the union of Jordan chains, preserving the order of each chain. We define,

$$\beta = \left\{ \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \right\}. \tag{2.210}$$

5. Matrix: now, we assemble the basis vectors into a transformation matrix, preserving the order of the basis. Define,

$$T = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}. \tag{2.211}$$

6. Transformation: finally, we compute $T^{-1}AT$. This yields,

$$T^{-1}AT = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 3 & 1 \\ 0 & 0 & 3 \end{bmatrix}, \tag{2.212}$$

which is in Jordan canonical form with two Jordan blocks,

$$J_1 = [3], J_2 = \begin{bmatrix} 3 & 1 \\ 0 & 3 \end{bmatrix}. \tag{2.213}$$

This example illustrates the following point—the Jordan form of a matrix is generally *not* unique—one can permute the order of the Jordan blocks and get a different Jordan form.

However, one may show that—*up to permutation of the block order*—the Jordan form of a matrix is unique. The interested reader is referred to [13] for proof of this fact.

This example also highlights a critical problem with the Jordan form, *numerical stability*. Suppose we're given the Jordan form matrix,

$$\begin{bmatrix} 3 & 0 & 0 \\ 0 & 3 & 1 \\ 0 & 0 & 3 \end{bmatrix}, \tag{2.214}$$

from the example above. Now, let's perturb one of the terms by arbitrarily small constant, $\epsilon > 0$, to get the matrix

$$\begin{bmatrix} 3 & 0 & 0 \\ 0 & 3 & 1 \\ 0 & 0 & 3+\epsilon \end{bmatrix}. \tag{2.215}$$

Is this matrix still in Jordan form? No! Now, the lower block of the matrix is no longer a Jordan block, since $3 + \epsilon \neq 3$ for $\epsilon > 0$. Thus, we find that for *any* arbitrarily small perturbation, the Jordan form of a matrix is at risk of changing structure. For instance, one may show that the Jordan form of the perturbed matrix above would be,

$$\begin{bmatrix} 3 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 3+\epsilon \end{bmatrix}. \tag{2.216}$$

Thus, we observe that the *structure* of the Jordan form is fragile to numerical perturbations. This illustrates the following general principle: the Jordan form is very useful for theoretical computation, but is impractical for numerical computation, where computer roundoff errors might arise. In summary:

The JCF is generally good for theory and bad for application.

2.3.3.2 Computing $\exp(A)$ and A^k

We finally have all the tools we need to meet the original goal we outlined at the beginning of the section: computing $\exp(A)$ and A^k in closed form. Above, when motivating the Jordan form, we showed how to compute any exponent of a Jordan block. Now, we'll focus on computing the matrix exponential of a Jordan block.

There are a couple of ways of computing the matrix exponential of a Jordan block—we'll use a particularly slick method that takes advantage of *nilpotent matrices*, but more brute-force methods are possible. First, consider the following intermediate result.

Lemma 2.7 (Commutativity & the Exponential) *Let $A, B \in \mathbb{C}^{n\times n}$.*

$$AB = BA \implies \exp(A+B) = \exp(A)\exp(B). \tag{2.217}$$

Proof See Problem 2.13. □

This result mirrors the sum-product rule for the scalar exponential: $e^{a+b} = e^a e^b$. Since scalars always commute ($ab = ba \ \forall a, b \in \mathbb{C}$) the scalar exponential rule actually follows from the lemma above! With this fact in mind, we state the following result.

Proposition 2.20 (Exponential of a Jordan Block) *Consider a Jordan block $J = \lambda I + N_0 \in \mathbb{C}^{n\times n}$. The exponential $\exp(Jt)$ is computed,*

$$\exp(Jt) = e^{\lambda t} \begin{bmatrix} 1 & t & \dots & \frac{t^{n-1}}{(n-1)!} \\ 0 & 1 & \ddots & \vdots \\ \vdots & & \ddots & t \\ 0 & \dots & \dots & 1 \end{bmatrix} \in \mathbb{C}^{n\times n}. \tag{2.218}$$

Proof Consider a Jordan block $J = \lambda I + N$. We wish to compute $\exp(Jt) = \exp((\lambda I + N)t)$. We observe that the matrices λI and N_0 commute, since $\lambda I N_0 = N_0 \lambda I$. Applying Lemma 2.7, it follows that,

$$\exp(Jt) = \exp(\lambda I t + N_0 t) = \exp(\lambda I t)\exp(N_0 t). \tag{2.219}$$

Thus, in order to compute the exponential of J, we just need to compute the exponential of λI and the exponential of N_0. Since λI is diagonal, we know that $\exp(\lambda I t) = e^{\lambda t} I$. Now, we compute $\exp(N_0 t)$. We recall that N_0 has the useful property that for every successive power of n, the diagonal of ones "shifts one to the right" and that $N_0^n = 0$. This means,

$$\exp(N_0 t) = I + N_0 t + \frac{1}{2!} N_0^2 t^2 + \frac{1}{3!} N_0^3 t^3 + ... + \frac{1}{(n-1)!} N_0^{n-1} t^{n-1}, \tag{2.220}$$

since all exponents of degree n and higher vanish. All that remains is to compute the product $e^{\lambda t} I \cdot \exp(N_0 t)$. This gives,

$$\exp(Jt) = e^{\lambda t} \begin{bmatrix} 1 & t & \dots & \frac{t^{n-1}}{(n-1)!} \\ 0 & 1 & \ddots & \vdots \\ \vdots & & \ddots & t \\ 0 & \dots & \dots & 1 \end{bmatrix}, \tag{2.221}$$

which is the desired formula. □

Now that we've computed both J^k and $\exp(Jt)$ for a Jordan block J, we state the general result concerning state transition matrices. As the following theorem simply collects results that we've already proven and puts them under the same umbrella, we provide no further proof.

Theorem 2.12 (Jordan Forms for the LTI State Transition Matrices) *Consider a matrix $A \in \mathbb{C}^{n\times n}$. Suppose $T \in \mathbb{C}^{n\times n}$ is an invertible matrix for which $J = T^{-1}AT$ is in the Jordan canonical form $J = \text{blkdiag}(J_1, ..., J_p)$.*

1. The continuous-time state transition matrix $\Phi(t, t_0)$ with respect to A is computed,

$$\Phi(t,t_0) = T\begin{bmatrix} e^{J_1(t-t_0)} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & e^{J_p(t-t_0)} \end{bmatrix} T^{-1}, \tag{2.222}$$

$$e^{J_i(t-t_0)} = e^{\lambda_i(t-t_0)}\begin{bmatrix} 1 & t & \dots & \frac{t^{n-1}}{(n-1)!} \\ 0 & 1 & \ddots & \vdots \\ \vdots & & \ddots & t \\ 0 & \dots & \dots & 1 \end{bmatrix}. \tag{2.223}$$

2. The discrete-time state transition matrix $\Phi[k,k_0]$ with respect to A is computed,

$$\Phi[k,k_0] = T\begin{bmatrix} J_1^{k-k_0} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & J_p^{k-k_0} \end{bmatrix} T^{-1}, \tag{2.224}$$

$$J_i^{k-k_0} = \sum_{j=0}^{k-k_0} \binom{k-k_0}{j} \lambda_i^{k-k_0-j} N_0^j. \tag{2.225}$$

2.3.3.3 Constructing the Jordan Canonical Form ★

This subsection is optional, and can be skipped without loss of continuity.

In this subsection, we complete the proof of Theorem 2.11, and show that we can *always* construct a transformation into Jordan form.[6] Our proof will closely follow that of [5]. In order to keep our treatment sufficiently brief, we'll abstract away a few details—we'll write (check this) in places where we skip a few steps. Let's begin!

The first insight into the problem of proving Theorem 2.11 is that, instead of focusing on a general Jordan block, we should focus on a *nilpotent* block, a block N for which $N^k = 0$ for some $k > 0$. We recall the following important result from linear algebra.

Lemma 2.8 (Basis from Nilpotent Matrices) *Let V be a finite dimensional vector space over $\mathbb{C}$. Let $T : V \to V$ be a nilpotent linear transformation of order m - that is, $T^m = 0$ and $T^k \neq 0$ for $k < m$. There exists a basis of $\mathbb{C}^n$ of the form,*

$$\{u_1, Tu_1, ..., T^{a_1-1}u_1, u_2, ..., T^{a_{k-1}-1}u_{k-1}, u_k, Tu_k, ..., T^{a_k-1}u_k\}, \tag{2.226}$$

where each $a_i > 0$ and $T^{a_i}u_i = 0$ for $1 \leq i \leq k$.

Proof We'll give proof of a special case of this result, and refer the reader to [5], Proposition 8.45 for the details of the general case. Let T be a nilpotent of order m. Since $T^{m-1} \neq 0$, there exists a vector $u \in V$ for which $T^{m-1}u \neq 0$. Now, consider the collection,

[6] This "always" comes with a couple of technical caveats. Here, by taking $A \in \mathbb{C}^{n\times n}$, we assume that the underlying field of the vector space is *complex*, and that complex transformations $T \in \mathbb{C}^{n\times n}$ are allowed. It is *not* the case that for $A \in \mathbb{R}^{n\times n}$, there always exists a real transformation $T \in \mathbb{R}^{n\times n}$ taking A into Jordan form (e.g. consider a real, diagonalizable matrix with complex eigenvalues). In the next chapter, we'll show how to construct a strictly real analogue of the Jordan form.

$$\{u, Tu, ..., T^{m-1}u\}. \tag{2.227}$$

We will show that this is a linearly independent collection. Suppose there exist complex numbers $c_0, ..., c_{m-1}$ for which

$$c_0u + c_1Tu + ... + c_{m-1}T^{m-1}u = 0. \tag{2.228}$$

Now, apply T^{m-1} to both sides. This yields,

$$c_0T^{m-1}u = 0 \Rightarrow c_0 = 0. \tag{2.229}$$

Next, apply T^{m-2} to both sides. This yields $c_1 = 0$. Repeating this process, we find that $c_0 = ... = c_{m-1} = 0$, which implies the collection is linearly independent. In the event where $\text{span}\{u, Tu, ..., T^{m-1}u\} = V$, this completes the proof. For the general case, we refer the reader to Proposition 8.45 of [5]. □

By examining the structure of the basis proposed in Lemma 2.8, one notices a resemblance to the *chains* of generalized eigenvectors we defined above. Now, we show how to use Lemma 2.8 to generate a basis for $\mathbb{C}^n$ composed of chains of generalized eigenvectors from each generalized eigenspace.

Note that, in the proof of the next result and in the results that follow, we will make use of an abstract linear transformation $L_A : \mathbb{C}^n \to \mathbb{C}^n$, defined $L_A(v) = Av$ for all $v \in \mathbb{C}^n$. That is, L_A is the abstract linear transformation associated to left multiplication by A. Although this transformation may seem useless at a first glance, we'll find it useful when discussing the restriction of L_A to certain subspaces of $\mathbb{C}^n$. We'll see exactly how this shakes out in the following result.

Lemma 2.9 (Basis for a Generalized Eigenspace) *Let $A \in \mathbb{C}^{n\times n}$ and $K_\lambda(A)$ be a generalized eigenspace of A. Then, the following properties are satisfied:*

1. *Invariance: $K_\lambda(A)$ is invariant under L_A.*
2. *Basis: $K_\lambda(A)$ has a basis of maximal chains of generalized eigenvectors of A.*

Proof First, we will prove that $K_\lambda(A)$ is invariant under L_A. That is, we will show $v \in K_\lambda(A)$ implies $Av \in K_\lambda(A)$. Suppose $v \in K_\lambda(A)$. Then, v satisfies $(A - \lambda I)^m v = 0$ for some $m \in \mathbb{N}$. This implies that,

$$(A - \lambda I)^{m+1}v = 0 \tag{2.230}$$
$$(A - \lambda I)^m(A - \lambda I)v = 0 \tag{2.231}$$
$$(A - \lambda I)^m(Av - \lambda v) = 0 \tag{2.232}$$
$$(A - \lambda I)^m Av = (A - \lambda I)^m \lambda v \tag{2.233}$$
$$(A - \lambda I)^m Av = 0. \tag{2.234}$$

We conclude that $L_A(v) = Av \in K_\lambda(A)$ and that $K_\lambda(A)$ is invariant under L_A. Using this fact, we can prove (2). If $K_\lambda(A)$ is invariant under A, we can define a linear transformation $T : K_\lambda(A) \to K_\lambda(A)$ by $T = L_A|_{K_\lambda(A)}$—the restriction of the linear transformation L_A to the subspace $K_\lambda(A)$. Since $K_\lambda(A)$ is invariant under L_A, it makes sense that T is a transformation from $K_\lambda(A) \to K_\lambda(A)$.

Since T is the restriction of multiplication by A to $K_\lambda(A)$, it follows that the only eigenvalue of T is λ. Therefore, the transformation $(T - \lambda I)$ has all zero eigenvalues and is nilpotent

(see the exercise below for a proof of this implication). By Lemma 2.8, we conclude there exists a basis of $K_\lambda(A)$ consisting of vectors,

$$\{u_1, (T-\lambda I)u_1, ..., (T-\lambda I)^{a_1-1}u_1, ..., u_k, (T-\lambda I)u_k, ..., (T-\lambda I)^{a_k-1}u_k\}. \tag{2.235}$$

Each sequence of vectors, $u_i, (T-\lambda I)u_i, ..., (T-\lambda I)^{a_i-1}u_i$ contained in this set is actually a maximal chain of generalized eigenvectors, written in reverse! We see this by applying the definition of T as left multiplication by A. We conclude that $K_\lambda(A)$ has a basis consisting of maximal chains of generalized eigenvectors of A. □

Exercise 2.18 Prove that $T \in \mathcal{L}(V,V)$ is nilpotent if and only if its only eigenvalue is zero. *Hint: one direction has a simple proof using the Cayley-Hamilton theorem, which we'll discuss later in the course.*

Great! We've now shown that we can find a basis for each $K_\lambda(A)$ consisting of a maximal chain of generalized eigenvectors. Next, we'll show that our *entire space* has a basis of generalized eigenvectors.

Lemma 2.10 ($\mathbb{C}^n$ has a Basis of Generalized Eigenvectors) *Let $A \in \mathbb{C}^{n\times n}$. Then, $\mathbb{C}^n$ has a basis consisting of generalized eigenvectors of A.*

Proof We will prove this by induction on n, the dimension of $\mathbb{C}^n$. In the base case, $n = 1$, this is obvious, since every nonzero vector is an eigenvector of A. Suppose for induction that, for some $n \geq 1$, $\mathbb{C}^k$ has a basis consisting of generalized eigenvectors of A for all $1 \leq k < n$. Consider an arbitrary eigenvalue λ of A. One can show with a little algebra that,

$$\mathbb{C}^n = \ker(A-\lambda I)^n \oplus \operatorname{range}(A-\lambda I)^n, \tag{2.236}$$

(check this if you don't believe me). Now, we have two cases to consider. First, suppose $\ker(A-\lambda I)^n = \mathbb{C}^n$. Then, every element of $\mathbb{C}^n$ must be a generalized eigenvector of A - the result trivially follows in this case.

Let's examine the other case. Suppose $\ker(A-\lambda I)^n \neq \mathbb{C}^n$, which implies $\operatorname{range}(A-\lambda I)^m \neq \{0\}$. Since λ is known to be an eigenvalue of A, we also have that $\ker(A-\lambda I)^m \neq \{0\}$. This leads us to conclude,

$$0 < \dim \operatorname{range}(A-\lambda I)^n < n. \tag{2.237}$$

By the same reasoning as in Lemma 2.9, $\operatorname{range}(A-\lambda I)^n$ is invariant under L_A. Now, let S be the linear transformation $S : \operatorname{range}(A-\lambda I)^n \to \operatorname{range}(A-\lambda I)^n$ defined $S = L_A|_{\operatorname{range}(A-\lambda I)^n}$—as the restriction of left multiplication by A to $\operatorname{range}(A-\lambda I)^n$. By induction, there exists a basis of $\operatorname{range}(A-\lambda I)^n$ consisting of generalized eigenvectors of S (which are also generalized eigenvectors of A by definition of S). Take this basis and append a basis of $\ker(A-\lambda I)^n$. This yields a basis of $\mathbb{C}^n$ consisting of generalized eigenvectors of A. This completes the inductive step, and the lemma follows. □

Next, we'll show that we can decompose our space into a direct sum of generalized eigenspaces. First, we recall that $V = V_1 \oplus ... \oplus V_m$ (where the V_i are subspaces of V) if all $v \in V$ admit a unique decomposition $v = v_1 + ... + v_m$, $v_i \in V_i$.

Lemma 2.11 (Generalized Eigenspace Decomposition) *Let $A \in \mathbb{C}^{n\times n}$ and let $\lambda_1, ..., \lambda_p$ be the distinct eigenvalues of A. Then, $\mathbb{C}^n$ decomposes as the direct sum,*

$$\mathbb{C}^n = K_{\lambda_1}(A) \oplus ... \oplus K_{\lambda_p}(A). \tag{2.238}$$

Proof First, we show that each generalized eigenvector corresponds to only one generalized eigenvalue of A. Suppose v corresponds to both λ and μ, and that m is the smallest integer for which $(A-\mu I)^k v = 0$. Then, since $m \leq n$,

$$0 = (A-\lambda I)^n v \tag{2.239}$$
$$= ((A-\mu I) + (\mu-\lambda)I)^n v \tag{2.240}$$
$$= \sum_{k=0}^{n} \binom{n}{k} (\mu-\lambda)^{n-k} (A-\mu I)^k v. \tag{2.241}$$

Applying $(A-\mu I)^{m-1}$ to both sides yields $0 = (\mu-\lambda)^n (A-\mu I)^{m-1} v$. Since $(A-\mu I)^{m-1} v \neq 0$, we conclude $\mu = \lambda$ and that v must correspond to *one* eigenvalue. Next, we will show that any collection of generalized eigenvectors corresponding to different generalized eigenspaces,

$$\{v_1, ..., v_p\},\ v_i \in K_{\lambda_i}(A),\ \lambda_i \neq \lambda_j, \tag{2.242}$$

is linearly independent. Suppose for contradiction that there exists a linearly dependent collection, $\{v_1, ..., v_p\}$. Then, there exist constants $a_1, ..., a_p$ - not all zero - for which $a_1 v_1 + ... + a_p v_p = 0$. Apply $(A-\lambda_p I)^n$ to both sides to get,

$$a_1 (A-\lambda_p I)^n v_1 + ... + a_p (A-\lambda_p I)^n v_p = 0 \tag{2.243}$$
$$a_1 (A-\lambda_p I)^n v_1 + ... + a_{p-1} (A-\lambda_p I)^n v_{p-1} = 0, \tag{2.244}$$

since v_p is a generalized eigenvector of λ_p and therefore satisfies $(A-\lambda_p I)^n v = 0$. Each remaining term *must* be nonzero, otherwise each remaining v_i would be a generalized eigenvector corresponding to two eigenvalues, which we proved above cannot happen. Yet, we know that, for $k \neq p$.

$$(A-\lambda_k I)^n (A-\lambda_p I)^n v_k = (A-\lambda_p I)^n (A-\lambda_k I)^n v_k = 0. \tag{2.245}$$

So, $(A-\lambda_k I)^n v_k$ is a generalized eigenvector corresponding to λ_p. This contradicts what we proved above! We conclude the collection is linearly independent.

Finally, we show the direct sum property. By Lemma 2.10, it follows that $\mathbb{C}^n = K_{\lambda_1}(A) + ... + K_{\lambda_p}(A)$. Now, we need to show that this sum is a *direct sum*. It is a nice fact from linear algebra that $V = V_1 \oplus ... \oplus V_p$ if $V = V_1 + ... + V_p$ and $v_1 + ... + v_p = 0$, $v_i \in V_i$ implies $v_i = 0$ for all i. Suppose $v_1 + ... + v_p = 0$, for $v_i \in K_{\lambda_i}(A)$. Since generalized eigenvectors corresponding to distinct eigenvalues are linearly independent, each v_i must be zero. We conclude that $\mathbb{C}^n = K_{\lambda_1}(A) \oplus ... \oplus K_{\lambda_p}(A)$, which completes the proof. □

We're almost there! We've shown that $\mathbb{C}^n$ admits a decomposition into a direct sum of generalized eigenspaces *and* that each generalized eigenspace has a basis of chains of generalized eigenvectors of A. All that remains is to put these facts together to conclude Theorem 2.11.

Proof (Of Theorem 2.11 on Existence of a Transformation into Jordan Form) Let $A \in \mathbb{C}^{n \times n}$ and let $\lambda_1, ..., \lambda_p$ be the distinct eigenvalues of A. Then, by Lemma 2.11, $\mathbb{C}^n = K_{\lambda_1}(A) \oplus ... \oplus K_{\lambda_p}(A)$. As such, for any bases $\beta_1, ..., \beta_p$ of each of the generalized eigenspaces, the transformation $[A]_{\beta_1 \cup ... \beta_p} = T^{-1} A T$, $T = [\beta_1, ..., \beta_p]$ should be in block diagonal form. Pick each basis β_i to be a basis for $K_{\lambda_i}(A)$, consisting of maximal chains of generalized eigenvectors (each starting at a true eigenvector of λ_i), as in Lemma 2.9. Each resulting block is a Jordan block, and the resulting matrix is in Jordan canonical form. □

Phew, what a proof! We note that this is *not* the only proof of the existence of the Jordan form - rather, this is one that generalizes well to arbitrary linear operators on finite dimensional vector spaces beyond $\mathbb{C}^n$. Other, shorter proofs that are more computation-heavy are possible. The interested reader is directed to [15] for the details of one such proof.

2.3.4 Further Reading

The treatment of the matrix exponential and state transition matrix in this section was primarily influenced by those of [22], [6], [11], and [24]. The proof of the construction of the Jordan form closely follows that of [5], and the algorithm for computing the Jordan form is from [16]. As mentioned above, for a more computational proof of the Jordan form theorem, we recommend the treatment of [15].

2.3.5 Problems

Problem 2.12 (Zero-Order Hold Discretization) In this problem, we'll show how a continuous-time linear can be *exactly* discretized into a discrete-time linear system. Consider the continuous-time system representation $(A(\cdot), B(\cdot), C(\cdot), D(\cdot))$,

$$\dot{x}(t) = A(t)x(t) + B(t)u(t) \tag{2.246}$$
$$y(t) = C(t)x(t) + D(t)u(t), \tag{2.247}$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$, and $y(t) \in \mathbb{R}^p$. Consider a strictly increasing sequence of sampling times, $\{t_k\}_{k\in\mathbb{Z}} \subseteq \mathbb{R}$, satisfying $t_k < t_{k+1}$ for all $k \in \mathbb{Z}$. Suppose the input signals $u(\cdot)$ to the continuous-time system are constant on the sampling intervals. That is, for every input signal $u(\cdot)$ to the continuous-time system, there exists a sequence $u[\cdot] : \mathbb{Z} \to \mathbb{R}^m$ for which $u(t) = u[k] \in \mathbb{R}$ for all $t \in [t_k, t_{k+1})$.

1. Show there exists a discrete-time system representation $(\hat{A}[\cdot], \hat{B}[\cdot], \hat{C}[\cdot], \hat{D}[\cdot])$ such that for all initial conditions $x_0 = x(t_0) = x[0] \in \mathbb{R}^n$, solutions to the system,

$$x[k+1] = \hat{A}[k]x[k] + \hat{B}[k]u[k] \tag{2.248}$$
$$y[k] = \hat{C}[k]x[k] + \hat{D}[k]u[k], \tag{2.249}$$

   satisfy $x[k] = x(t_k)$ and $y[k] = y(t_k)$ for all $k \in \mathbb{Z}$, where $x(t_k)$ and $y(t_k)$ are the state and output of the continuous-time system at time t_k. This tells us that we can *exactly* discretize the continuous-time LTV system.
2. Now, suppose each matrix in the continuous-time system representation is constant,

$$(A(\cdot), B(\cdot), C(\cdot), D(\cdot)) = (A, B, C, D). \tag{2.250}$$

   Further, assume that for each $k \in \mathbb{Z}$, $t_{k+1} - t_k = \Delta$. Using your answer to (1), show that there exists a discrete-time LTI system representation $(\hat{A}, \hat{B}, \hat{C}, \hat{D})$ which exactly discretizes the continuous-time LTI system.

Discretization in which one holds an input signal constant across a sampling period is referred to as *zero-order hold (ZOH)* discretization.

Problem 2.13 (Commutativity & The Exponential ★) Let $A, B \in \mathbb{R}^{n\times n}$. In this problem, we'll prove that commutativity of A and B, $AB = BA$, implies $\exp(A+B) = \exp(A)\exp(B)$. Notably, we'll give a proof that uses an *existence and uniqueness* argument, rather than a direct algebraic argument.

1. Give examples of matrices A, B for which $\exp(A+B) \neq \exp(A)\exp(B)$. *Feel free to use a computational tool to experiment with different matrices.*
2. Using an existence and uniqueness argument, prove that $AB = BA$ implies $\exp(A+B) = \exp(A)\exp(B)$. *Hint: set up an initial value problem involving A and B.*

Problem 2.14 (Algebraic Properties of the Matrix Exponential) Let $A \in \mathbb{R}^{n\times n}$. Prove the following properties of the matrix exponential:

1. For every $t_1, t_0 \in \mathbb{R}$, $\exp(A(t_1+t_0)) = \exp(At_1)\exp(At_0) = \exp(At_0)\exp(At_1)$.
2. For an eigenvalue-vector pair (λ, v) of A, (e^{λ}, v) is an eigenvalue-vector pair of $\exp(A)$.
3. $\det(\exp A) = e^{\operatorname{tr} A}$.
4. $(\exp(A))^{-1} = \exp(-A)$.

Hint: read the result of Problem 2.13 before attempting these problems.

Problem 2.15 (Some State Transition Matrices [8]) Sometimes, we can use the matrix exponential to gain insight into the structure of linear, time-varying initial value problems. Consider the following two problems.

1. Suppose $A, B \in \mathbb{R}^{n\times n}$. Show that the unique solution to the initial value problem,
$$\dot{x}(t) = e^{-At}Be^{At}x(t),\ x(t_0) = x_0, \tag{2.251}$$
is given by $x(t) = e^{-At}e^{(A+B)(t-t_0)}e^{At_0}x_0$.
2. Let $f : \mathbb{R} \to \mathbb{R}$ be a continuous function and $A \in \mathbb{R}^{n\times n}$. Show that the state transition matrix of the system,
$$\dot{x}(t) = f(t)Ax(t), \tag{2.252}$$
is computed $\Phi(t, t_0) = \exp[\int_{t_0}^{t} f(\tau)d\tau)A]$.

Problem 2.16 (The Complex Jordan Block) Consider a matrix $A \in \mathbb{R}^{2\times 2}$. If A has complex eigenvalues, there are sometimes more convenient transformations than the (complex) Jordan form! Suppose A has complex eigenvalues $\sigma \pm j\omega$.

1. Prove there exists a matrix $T \in \mathbb{C}^{n\times n}$ for which,
$$T^{-1}AT = \begin{bmatrix} \sigma & \omega \\ -\omega & \sigma \end{bmatrix}. \tag{2.253}$$
2. Show that the exponential of $T^{-1}AT$ is computed,
$$\exp(T^{-1}AT) = \begin{bmatrix} e^{\sigma t}\cos(\omega t) & e^{\sigma t}\sin(\omega t) \\ -e^{\sigma t}\sin(\omega t) & e^{\sigma t}\cos(\omega t) \end{bmatrix}. \tag{2.254}$$

Problem 2.17 (The Floquet Decomposition ★) Consider the system $\dot{x}(t) = A(t)x(t)$, in which $A(\cdot)$ is periodic with period $T > 0$, $A(t+T) = A(t)$ for all $t \in \mathbb{R}$. The basic idea of the *Floquet decomposition* is that, by constructing a time-varying transformation that "syncs up" with the periodicity of $A(\cdot)$, we can use time-invariant tools to study a time-varying system.

1. Let $\Phi(t, t_0)$ denote the state transition matrix of $\dot{x}(t) = A(t)x(t)$. Show that $\Phi(t+T, 0) = \Phi(t, 0)\Phi(T, 0)$.
2. Prove that for every nonsingular matrix $B \in \mathbb{C}^{n\times n}$, there exists a matrix $A \in \mathbb{C}^{n\times n}$ for which $\exp(A) = B$. *Hint: the complex (scalar) logarithm is defined on the nonzero complex numbers.*
3. Prove there exists an $R \in \mathbb{C}^{n\times n}$ for which $\Phi(T, 0) = \exp(TR)$.
4. Consider a time-varying transformation $P : \mathbb{R} \to \mathbb{C}^{n\times n}$, for which

$$P(t)^{-1} = \Phi(t, 0)e^{-tR}. \tag{2.255}$$

   Show that $P(t)$ is in fact invertible for all t. Then, prove that for any $t, t_0 \in \mathbb{R}$,

$$\Phi(t, t_0) = P(t)^{-1}e^{R(t-t_0)}P(t_0). \tag{2.256}$$

   Comment on the significance of this result. *The eigenvalues of the R matrix, called Floquet multipliers, help in determining the stability of periodic systems.*

Problem 2.18 (Fun with Jordan Forms) In the following problem, we'll get some practice with Jordan forms.

1. Find a Jordan canonical form of each of the following matrices,

$$A_1 = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}, \; A_2 = \begin{bmatrix} 2 & 1 & 0 \\ 0 & 2 & 0 \\ 0 & -1 & 2 \end{bmatrix}. \tag{2.257}$$

2. Calculate the matrix exponentials $\exp(A_i t), \; i = 1, 2$.
3. For each of A_i, illustrate how an arbitrarily small numerical perturbation can change the structure of the Jordan form. How do you think this affects numerical computation of the Jordan form?

Problem 2.19 (Schur Triangulation) The Jordan normal form—though convenient for theoretical calculations—suffers from a number of numerical problems. Here, we consider an alternative technique for computing the matrix exponential, based on the *Schur triangulation* of a matrix.

1. A matrix $T \in \mathbb{C}^{n\times n}$ is said to be *unitary* if $TT^* = T^*T = I$. Prove the *Schur triangulation theorem*, which states that, for any $A \in \mathbb{C}^{n\times n}$, there exists a unitary matrix T for which $U := T^*AT$ is upper triangular. *Hints: Proceed by induction.*
2. Determine a method for computing the matrix exponential of an upper triangular matrix. *Your method does not have to be computationally efficient, it just needs to work.*
3. Comment on the benefits and drawbacks of using your Schur triangulation method versus the Jordan normal form method of calculating the matrix exponential.

If you're interested in reading about more ways of computing the matrix exponential, check out the paper *Nineteen Dubious Ways to Compute the Exponential of a Matrix, Twenty-Five Years Later*.

Problem 2.20 (Some Invariant Subspace Theory ★) Fundamentally, the construction of the Jordan canonical form relies on the fact that the generalized eigenspaces are *invariant subspaces* - subspaces $V \subseteq \mathbb{C}^n$ satisfying $AV \subseteq V$. In this problem, we'll study some basic properties of invariant subspaces and see how they relate to the Jordan form.

1. Let $T : V \to V$ be a linear transformation on an n-dimensional vector space V over $\mathbb{K}$. A subspace $M \subseteq V$ is said to be T-invariant if $Tx \in M$ for all $x \in M$. Suppose that V is the direct sum of two subspaces $M_1, M_2 \subseteq V$, $V = M_1 \oplus M_2$. If both M_1 and M_2 are T-invariant, prove there exists a basis β for V in which T has the matrix representation,

$$[T]_\beta = A = \begin{bmatrix} A_{11} & 0 \\ 0 & A_{12} \end{bmatrix} \in \mathbb{K}^{n \times n}, \tag{2.258}$$

   where $\dim(M_1)$ and $\dim(A_2)$ equal the sizes of A_{11} and A_{12}. Argue that the restrictions $T|_{M_1} : M_1 \to M_1$ and $T|_{M_2} : M_2 \to M_2$ are well-defined maps.
2. We define a generalized eigenspace of the linear transformation T to be a space,

$$K_\lambda(T) = \{v \in V : (T - \lambda I)^m = 0 \text{ for some } m \in \mathbb{N}\} \subseteq V, \tag{2.259}$$

   where λ is an eigenvalue of T. Prove the following:

   a. $K_\lambda(T)$ contains at least one eigenvector of T.
   b. $K_\lambda(T)$ is a subspace.
   c. $K_\lambda(T)$ is T-invariant.
   d. $K_\lambda(T) = \{v \in V : (T - \lambda I)^{\dim V} = 0\}$.

3. Suppose T has two distinct eigenvalues, λ_1 and λ_2, and $V = K_{\lambda_1}(T) \oplus K_{\lambda_2}(T)$. Suppose for each $i = 1, 2$, the sets

$$\beta_i = \{v_i, (T - \lambda_i I)v_i, ..., (T - \lambda_i I)^{m_i - 1} v_i\}, \tag{2.260}$$

   form bases for K_{λ_1} and K_{λ_2}. Construct a basis β for V in which T is in Jordan canonical form.

2.4 Impulse Response & Transfer Functions

Thus far, our study of linear systems has focused primarily on the *state equations*,

$$\dot{x}(t) = A(t)x(t) + B(t)u(t) \tag{2.261}$$
$$x[k+1] = A[k]x[k] + B[k]u[k], \tag{2.262}$$

of state space representations. We spent some time studying the existence and uniqueness of solutions to these equations, as well as establishing the structure of their solutions via the state transition matrix. At large, we haven't taken a closer look at the output equations.

In this section, this all changes! Instead of looking at the state equation of the system to gain insight into its behavior, we'll zoom out and examine how the input to a system directly influences its output. The border between these two approaches hints at two perspectives on control theory.

On one side, we have the *internal* (state space) approach, where we look into the internal dynamics of the system to make conclusions about its behavior. On the other side, we have the *input/output (I/O) approach*, where we focus directly on the relationship between the input and output. We'll continue to discuss the interplay between these approaches, as well as their various merits, as the course progresses.

This section is outlined as follows. First, we'll study discrete-time systems from the I/O perspective, and will focus on an object called the *impulse response*. Following this, we'll take on the more challenging task of defining the impulse response of a continuous-time system, taking a short detour to discuss the Dirac delta distribution along the way. We'll conclude with a discussion of two fundamental transforms—the Laplace and $\mathcal{Z}$-transforms—which take advantage of the unique structure of the impulse response to transform real, LTI systems into complex, algebraic functions. Let's begin!

2.4.1 Impulse Response of Discrete-Time Systems

We'll begin by studying discrete-time systems from the I/O perspective. Although we usually start with continuous-time and translate to discrete-time, here, we'll find taking the opposite approach to be helpful. In particular, we'll find that the discrete-time theory is considerably easier and provides substantial insight into the continuous-time case.

Let's recall what we know about the I/O behavior of a discrete-time system. Recall that, in Section 2, we stated that the input/output map ρ of a discrete-time linear, time-varying system representation is calculated,

$$\rho(k_1, k_0, x_0, u[\cdot]) = \underbrace{C[k_1]\Phi[k_1, k_0]x_0}_{\text{Zero-Input Response}} + \underbrace{C[k_1]\sum_{j=k_0}^{k_1-1}\Phi[k_1, j+1]B[j]u[j] + D[k_1]u[k_1]}_{\text{Zero-State Response}}, \tag{2.263}$$

where we can split up the I/O map into the zero-input and zero-state response terms. From this formula, we observe that the zero-state response *entirely* characterizes how inputs interact with the response of the system. Thus, if we'd like to develop the input/output perspective, we should focus on the behavior of the zero-state response.

Let's examine the formula for the zero-state response in greater detail. We know that the zero-state response is calculated,

$$\rho(k_1, k_0, 0, u[\cdot]) = C[k_1] \sum_{j=k_0}^{k_1-1} \Phi[k_1, j+1] B[j] u[j] + D[k_1] u[k_1], \tag{2.264}$$

We observe that the zero-state response at any given moment *only* depends on the history of inputs to the system—there is *no* presence of state in the zero-state response whatsoever! This hints at an interesting idea in the input/output approach to systems: assuming the initial state of the system is fixed (for example at $x_0 = 0$), we can come up with a map from a pair of times and an input signal that maps *directly* to the value of an output signal. By studying the structure underlying such a map, perhaps we could come up with more elegant methods of studying the input/output behavior of a system than directly using the I/O map ρ. Amazingly, for linear systems, all we need to characterize this direct input-to-output function is the response of the system to a single input: the *unit impulse*.

Definition 2.28 (Discrete-Time Unit Impulse) The discrete-time unit impulse function is the function $\delta[\cdot] : \mathbb{Z} \to \mathbb{R}$, defined,

$$\delta[k] = \begin{cases} 1 & k = 0, \\ 0 & k \neq 0. \end{cases} \tag{2.265}$$

The shifted unit impulse, $\delta^{k_0}[\cdot] : \mathbb{Z} \to \mathbb{R}$, is defined $\delta^{k_0}[k] = \delta[k - k_0]$, for a fixed $k_0 \in \mathbb{Z}$.

Thus, the discrete-time unit impulse function (also called the *unit pulse* or *unit sample* function) is a scalar-valued discrete-time signal that jumps up to 1 at time $k = 0$ and is zero elsewhere. We can *shift* the unit impulse to jump up at time k_0 by defining $\delta^{k_0}[k] = \delta[k-k_0]$. Using the unit impulse function, we define the impulse response of a discrete-time linear system. We begin with the SISO case, and then generalize to the MIMO case.

Definition 2.29 (Impulse Response of a DT-SISO System) Consider a SISO, discrete-time linear I/O system with I/O map ρ. The impulse response map of the system is the map $h[\cdot, \cdot] : \mathbb{Z} \times \mathbb{Z} \to \mathbb{R}$, defined,

$$h[k, k_0] = \begin{cases} \rho(k, k_0, 0, \delta^{k_0}[\cdot]) & k \geq k_0 \\ 0 & k < k_0, \end{cases} \tag{2.266}$$

where $\delta^{k_0}[k] = \delta[k - k_0]$, the unit impulse applied at time k_0.

Remark 2.27 As opposed to being defined on $\mathbf{T} = \{(k_1, k_0) \in \mathbb{Z} \times \mathbb{Z} : k_1 \geq k_0\}$ (like the I/O map, which requires $k_1 \geq k_0$), the impulse response map is defined on all of $\mathbb{Z} \times \mathbb{Z}$. This proves convenient in calculations.

Thus, we define the impulse response $h[k, k_0]$ of a discrete-time linear system to be the zero-state response of a system to a unit impulse applied at time k_0. For all $k < k_0$, we define $h[k, k_0]$ to be zero, as the I/O map is not defined on this domain. Defining the impulse response to be zero for $k < k_0$ is justifiable, since we wouldn't expect a *causal* linear system to stray from zero if it has zero initial condition and zero input applied—causality implies that, for $k < k_0$, the system behavior is not affected by the impulse at time k_0. Now, we generalize this definition to a MIMO system.

Definition 2.30 (Impulse Response of a MIMO System) Consider a MIMO discrete-time, linear I/O system with I/O map ρ, input-value space $\mathbb{R}^m$, and output-value space $\mathbb{R}^p$. The impulse response map of the system is the map $H[\cdot,\cdot] : \mathbb{Z} \times \mathbb{Z} \to \mathbb{R}^{p\times m}$, defined

$$H[k,k_0] = \begin{bmatrix} h_{11}[k,k_0] & \dots & h_{1m}[k,k_0] \\ \vdots & \ddots & \vdots \\ h_{p1}[k,k_0] & \dots & h_{pm}[k,k_0] \end{bmatrix} \in \mathbb{R}^{p\times m} \tag{2.267}$$

$$h_{ij}[k,k_0] = \begin{cases} \rho(k,k_0,0,\delta^{k_0}[\cdot]e_j)_i & k \geq k_0 \\ 0 & k < k_0, \end{cases} \tag{2.268}$$

where $\rho(k,k_0,0,\delta^{k_0}[\cdot]e_j)_i \in \mathbb{R}$ is the i'th component of the zero-state response to an input $\delta^{k_0}[k]e_j = (0,...,\delta[k-k_0],...,0) \in \mathbb{R}^m$ containing a unit impulse function in its j'th component.

In summary, the impulse response of a MIMO system is a matrix-valued function, in which entry ij is the response at time k of output coordinate i to an impulse applied in input coordinate j at time k_0. In order to ensure that the MIMO impulse response map is conceptually well-defined, it pays to check that each element h_{ij} corresponds to the impulse response map of a SISO system. The following exercise asks you to verify this.

Exercise 2.19 (The MIMO Impulse Response is Well-Defined) Consider a discrete-time, linear I/O system with input-value space $\mathbb{R}^m$, output-value space $\mathbb{R}^p$, and I/O map ρ. Prove there exists a SISO system with I/O map ρ_i^j satisfying $\rho_i^j(k,k_0,0,u_j[\cdot]) = \rho(k,k_0,0,u_j[\cdot]e_j)_i$, for all $u_j[\cdot] : \mathbb{Z} \to \mathbb{R}$.

Now that we've defined the impulse response of a linear I/O system, let's calculate the impulse response corresponding to a discrete-time, linear system representation.

Proposition 2.21 (Impulse Response Map of a DT System Representation) *Consider a discrete-time, linear system representation $(A[\cdot],B[\cdot],C[\cdot],D[\cdot])$. The impulse response map of the system is computed,*

$$H[k,k_0] = \begin{cases} C[k]\Phi[k,k_0+1]B[k_0] & k > k_0 \\ D[k] & k = k_0 \\ 0 & k < k_0. \end{cases} \tag{2.269}$$

Proof Let's compute element ij of $H[k,k_0]$—the response of output coordinate i to a unit impulse applied at time $k = k_0$ in input coordinate j. We'll compute this for each of the three cases above. We first focus on the case of $k > k_0$. With $u[\cdot] = \delta^{k_0}[\cdot]e_j$ the input signal having δ^{k_0} in coordinate j and zeros elsewhere, we have

$$\rho(k,k_0,0,\delta^{k_0}[\cdot]e_j)_i = \Big[C[k]\sum_{j=k_0}^{k-1}\Phi[k,j+1]B[j]e_j\delta^{k_0}[j] + D[k]e_j\delta^{k_0}[k]\Big]_i \tag{2.270}$$

$$= \Big[C[k]\Phi[k,k_0+1]B[k_0]e_j\Big]_i, \tag{2.271}$$

since for all indices $j \neq k_0$ in the sum, $\delta^{k_0}[j]$ equals zero, and for $j = k_0$, $\delta^{k_0}[j]$ equals one. Now, we recognize this term as,

$$= \Big[C[k]\Phi[k, k_0+1]B[k_0]\Big]_{ij}, \tag{2.272}$$

since multiplication by e_j selects column j of $C[k]\Phi[k, k_0+1]B[k_0]$. This confirms the formula for $k > k_0$. Now, we verify the formula for $k = k_0$. We have, for the same $u[\cdot]$ as above,

$$\rho(k_0, k_0, \delta^{k_0}[\cdot]e_j)_i = \Big[C[k_0]\sum_{j=k_0}^{k_0-1}\Phi[k_0, j+1]B[j]e_j\delta^{k_0}[j] + D[k_0]e_j\delta^{k_0}[k_0]\Big]_i \tag{2.273}$$

$$= \Big[D[k_0]e_j\Big]_i \tag{2.274}$$

$$= \Big[D[k_0]\Big]_{ij}, \tag{2.275}$$

where the sum is empty (and therefore zero) since $k = k_0$, $u[k_0]$ picks out column j of $D[k_0]$, and $[\cdot]_i$ picks out row i. This again matches the given formula. The final component, $k < k_0 \Rightarrow H[k, k_0] = 0$, simply follows from the definition of the impulse response. □

Now that we've established a few basic facts concerning the impulse response, we can return to the goal we outlined above—characterizing the zero-state response of a linear system to *any* signal via a simple, direct input-output relationship. A key step in accomplishing this goal is in recognizing the following fact.

Lemma 2.12 (Signal Representation via Impulses) *Consider a signal $u[\cdot] : \mathbb{Z}_{\geq k_0} \to \mathbb{R}^m$. For any $k \geq k_0$, $u[k]$ can be written as a sum of impulses,*

$$u[k] = \sum_{j=k_0}^{k}\delta^j[k]u[j] = \sum_{j=k_0}^{k}\delta[k-j]u[j]. \tag{2.276}$$

Remark 2.28 The notation $\mathbb{Z}_{\geq k_0}$ refers to the set of integers that are greater than or equal to a fixed integer $k_0 \in \mathbb{Z}$.

Exercise 2.20 Verify the claim of Lemma 2.12.

Now that we know we can represent *any* discrete-time signal using impulses, we can use impulse response to characterize the zero-state response of a system to any input.

Theorem 2.13 (Zero-State Response via Impulse Response) *Consider a discrete-time linear system with I/O map ρ and impulse response map $H[\cdot,\cdot] : \mathbb{Z} \times \mathbb{Z} \to \mathbb{R}^{p\times m}$. For any input signal $u : \mathbb{Z}_{\geq k_0} \to \mathbb{R}^m$, the zero-state response of the system is computed,*

$$\rho(k, k_0, 0, u[\cdot]) = \sum_{j=k_0}^{k}H[k, j]u[j]. \tag{2.277}$$

Remark 2.29 We should be a little careful when defining $u : \mathbb{Z}_{\geq k_0} \to \mathbb{R}^m$! Formally, we defined the input signal space of a discrete-time system to be $\mathcal{U} = \{u : \mathbb{Z} \to \mathbb{R}^m\}$, so we shouldn't accept signals defined only on $\mathbb{Z}_{\geq k_0}$. As such, when we define a signal $u[\cdot] : \mathbb{K}_{\geq k_0} \to \mathbb{R}^m$, we will identify it with the "extended" input signal,

$$u_e[k] = \begin{cases} u[k] & k \geq k_0 \\ 0 & k < k_0, \end{cases} \tag{2.278}$$

which is defined on all of $\mathbb{Z}$. Here, by "identify," we mean that we will assume u is equal to u_e without explicit mention. Since the future behavior of a causal I/O system only depends on the current state and inputs (as a consequence of the restriction axiom), this is a harmless identification to make. Under the identification with u_e, the input u formally belongs to $\mathcal{U}$, and can be passed into the I/O map ρ without trouble.

Now, we return to the proof of Theorem 2.13.

Proof The proof of this result follows from an application of linearity of the I/O map. Here, to keep the notation simple, we'll prove the result in the case where $y \in \mathbb{R}$ and $u \in \mathbb{R}^m$—this is sufficient to show the general case, which consists of assembling the full output vector $y \in \mathbb{R}^p$ by stacking scalar, y_i terms. With that said, let's get started on the proof. In the case where $y \in \mathbb{R}^m$ and $u \in \mathbb{R}$, Lemma 2.12 tells us that, for $u[\cdot] : \mathbb{Z}_{\geq k_0} \to \mathbb{R}^m$, we can write,

$$u[k] = \sum_{j=k_0}^{k} \delta^j[k]u[j] = \sum_{j=k_0}^{k} \sum_{i=1}^{m} \delta^j[k]u_i[j]e_i, \tag{2.279}$$

where each e_i is the i'th standard basis vector of $\mathbb{R}^m$, with a 1 in index i and zeros elsewhere. Now, we apply the linearity of the I/O map. We have,

$$\rho(k, k_0, 0, u[\cdot]) = \sum_{j=k_0}^{k} \sum_{i=1}^{m} \rho(k, k_0, 0, \delta^j[k]e_i)u_i[j] \tag{2.280}$$

$$= \sum_{j=k_0}^{k} \sum_{i=1}^{m} \rho(k, j, 0, \delta^j[k]e_i)u_i[j] \tag{2.281}$$

$$= \sum_{j=k_0}^{k} \sum_{i=1}^{m} h_{1i}(k, j)u_i[j] \tag{2.282}$$

$$= \sum_{j=k_0}^{k} \begin{bmatrix} h_{11}[k, j] & \dots & h_{1m}[k, j] \end{bmatrix} u[j] \tag{2.283}$$

$$= \sum_{j=k_0}^{k} H[k, j]u[j], \tag{2.284}$$

where we make the final equality under the assumption that $p = 1$. The general case follows by stacking the formula above, p times. □

This tells us that we can *entirely* characterize the zero-state response of a discrete-time linear system to *any* input from knowledge of its impulse response! Further, this impulse response formula is significantly simpler than the complex, I/O map formula we stated above.

Now, let's specialize what we've learned from the time-varying case to the time-invariant case. Due to time-invariance, the impulse response map $H[k, k_0]$ of an LTI system representation will depend only on the amount of time that has passed, $k - k_0$, rather than the objective times k and k_0. Thus, we expect the impulse response map of a linear, time-invariant system to satisfy,

$$H[k, k_0] = H[k - k_0, 0] \; \forall k, k_0 \in \mathbb{Z} \text{ (LTI case)}, \tag{2.285}$$

since the differences between the first and second arguments are identical in the left and right-hand sides. Consequently, in order to characterize the impulse response of an LTI system, it's sufficient to know $H[k,0]$ for all $k \in \mathbb{Z}$, since $H[k,k_0]$ will equal $H[k-k_0,0]$. We introduce the following special "definition," (which is really just special notation) to highlight this fact.

Definition 2.31 (DT-LTI Impulse Response Map) Consider a discrete-time, LTI system with impulse response map $H[\cdot,\cdot] : \mathbb{Z} \times \mathbb{Z} \to \mathbb{R}^{p\times m}$. The LTI impulse response map of the system is the map $H[\cdot] : \mathbb{Z} \to \mathbb{R}^{p\times m}$, defined

$$H[k] := H[k,0], \ \forall k \in \mathbb{Z}. \tag{2.286}$$

Remark 2.30 In the definition above, we use the same letter for $H[\cdot,\cdot]$ and $H[\cdot]$. Although this *is* admittedly an overload of notation, one should always be able to determine from context which impulse response map is being used. If only one argument appears, assume the LTI impulse response map is being used, while if two arguments appear, assume the general (LTV) impulse response map is being used. Just like we use the same letter to refer to both maps, we'll also freely refer to both the (LTV) impulse response map and the LTI impulse response map as *the* impulse response map of the system. Again, context should make clear which map we're referring to.

Now, we compute the impulse response map of a discrete-time, LTI system.

Proposition 2.22 (Impulse Response Map of a DT-LTI System Representation) *Consider a discrete-time, LTI system representation (A,B,C,D). The LTI impulse response map $H[\cdot] : \mathbb{Z} \to \mathbb{R}^{p\times m}$ of the system is computed,*

$$H[k] = \begin{cases} CA^{k-1}B & k > 0 \\ D & k = 0 \\ 0 & k < 0. \end{cases} \tag{2.287}$$

Exercise 2.21 Using the formula for the discrete-time, LTI state transition matrix, verify the formula proposed in Proposition 2.22.

Using the LTI impulse response map $H[\cdot] : \mathbb{Z} \to \mathbb{R}^{p\times m}$ in conjunction with Theorem 2.13, we observe that we can write the zero-state response $y[k] = \rho(k,k_0,0,u[\cdot])$ of a discrete-time LTI system to an input $u : \mathbb{Z}_{\geq k_0} \to \mathbb{R}^m$ as the sum,

$$y[k] = \sum_{j=k_0}^{k} H[k,j]u[j] = \sum_{j=k_0}^{k} H[k-j]u[j] = \sum_{j=-\infty}^{\infty} H[k-j]u[j], \tag{2.288}$$

where we extend the lower bound to $-\infty$ using our convention that $u[k] = 0$ for $k < k_0$ and extend the upper bound to $+\infty$ using $H[k-j] = 0$ for $k - j < 0$. Sums of this form are of *fundamental* importance in control theory, signal processing, and beyond.

Definition 2.32 (Convolution Sum) Consider two discrete-time signals, $u[\cdot]$ and $H[\cdot]$, of compatible dimension, for which $H[k] \cdot u[k] \in \mathbb{R}^p$. The *convolution* of H and u is the signal $(H * u)[\cdot] : \mathbb{Z} \to \mathbb{R}^p$, defined as the sum,

$$(H * u)[k] = \sum_{j=-\infty}^{\infty} H[k-j]u[j], \ \forall k \in \mathbb{Z}. \tag{2.289}$$

Such a sum is called a *convolution sum*.

Remark 2.31 It's critical to note - the convolution of two signals is *another signal*, not a fixed value! That is, $H * u$ is a map from $\mathbb{Z} \to \mathbb{R}^p$, not a single vector in $\mathbb{R}^p$.

Remark 2.32 Since the convolution sum is an *infinite sum*, special care should be taken to ensure that the sum converges before using it.

Remark 2.33 If one takes the convolution of signals H and u, one says that H and u are being *convolved* (not *convoluted*).

Remark 2.34 We briefly elaborate on what we mean by "compatible dimension." Frequently, we'll convolve a matrix-valued signal $H[\cdot] : \mathbb{Z} \to \mathbb{R}^{p\times m}$ with a vector-valued signal, $u[\cdot] : \mathbb{R} \to \mathbb{R}^m$. Since the product $H[k] \cdot u[k]$ is well-defined for $H[k] \in \mathbb{R}^{p\times m}$ and $u[k] \in \mathbb{R}^m$, we say that the signals are of compatible dimension. Sometimes, we'd also like to convolve a scalar signal $f[\cdot] : \mathbb{Z} \to \mathbb{R}$ with a vector signal $u[\cdot] : \mathbb{Z} \to \mathbb{R}^m$. Despite $f[k]$ not having m rows, the product $f[k] \cdot u[k]$ is still well-defined, so we still say that the dimensions are compatible. It's for this reason that we don't require $H[k] \in \mathbb{R}^{p\times m}$ and $u[k] \in \mathbb{R}^m$—asking for *compatible dimension* yields greater flexibility.

By applying Lemma 2.12 and the definition of a convolution sum, we can rephrase the signal representation property of the discrete-time impulse in the language of convolution.

Lemma 2.13 (Impulse is the Identity of Convolution) *For any signal $u[\cdot] : \mathbb{Z} \to \mathbb{R}^p$ and the discrete-time impulse $\delta[\cdot] : \mathbb{Z} \to \mathbb{R}^p$, it follows that $u[\cdot] = (\delta * u)[\cdot]$.*

Proof Immediate from Lemma 2.12 and Definition 2.32. □

Applying this lemma together with Theorem 2.13, we arrive at the following result.

Corollary 2.2 (Convolution of Input & Impulse Response Equals Output) *Consider a discrete-time, LTI system with impulse response map $H[\cdot] : \mathbb{Z} \to \mathbb{R}^{p\times m}$. For any input signal $u[\cdot] : \mathbb{Z}_{\geq 0} \to \mathbb{R}^m$, the zero-state response of the system is equal to the convolution of the impulse response and input,*

$$\rho(k, 0, 0, u[\cdot]) = (H * u)[k]. \tag{2.290}$$

Proof This result follows from a straightforward application of Theorem 2.13 and Definition 2.32. Since for $k < 0$, one has $H[k] = 0$ (by definition of impulse response), it follows that

$$(H * u)[k] = \sum_{j=-\infty}^{\infty} H[k-j]u[j] = \sum_{j=-\infty}^{k} H[k-j]u[j] = \sum_{j=0}^{k} H[k-j]u[j], \tag{2.291}$$

where we use that $H[k-j]$ is zero for $k-j < 0$ and that $u[j]$ is zero for $j < 0$ (under the convention we established for signals on $\mathbb{Z}_{\geq 0}$). Since $H[k-j] = H[k, j]$, we observe that the final expression matches the formula for $\rho(k, 0, 0, u[\cdot])$ from Theorem 2.13. □

Remark 2.35 Recall that, in Section 1 of this chapter, we defined all of our systems to be *causal*. Due to causality, $H[k-j] = 0$ for $k - j < 0$, since causal systems don't require information about the future (time j, for $j > k$) to determine information about the present (time k). As we observed in the proof above, this property ensured the convolution sum did not truly extend to $+\infty$. For *noncausal* systems, however, this is not necessarily the case—one must be more careful to ensure that the sum converges.

Exercise 2.22 So far, we've focused on systems with zero initial condition. Confirm that, for a discrete-time, LTI system representation (A, B, C, D), the output of the system from initial state $x[0] = x_0$ and input signal $u[\cdot] : \mathbb{Z}_{\geq 0} \to \mathbb{R}^m$ is computed $y[k] = C\Phi[k, 0]x_0 + (H * u)[k]$.

2.4.2 Impulse Response of Continuous-Time Systems

Note: Since a rigorous construction of the continuous-time impulse response involves a digression into distribution theory, we'll be a little bit more hand-wavy than usual in this subsection. We direct the reader to the references at the end of the section for a more detailed mathematical approach to the content covered here.

Now that we've developed a theory for the discrete-time impulse response, we aim to construct an analogous theory for continuous-time systems. Following the same path we took in the discrete-time case, our first step is to answer the question: what is the correct notion of impulse response for a continuous-time system?

In order to answer this question, we must first define a continuous-time impulse signal. To motivate the definition of this signal, let's recall the important properties of the discrete-time impulse. Perhaps the most important property of the discrete-time impulse $\delta[\cdot]$ was that any signal $u[\cdot] : \mathbb{Z} \to \mathbb{R}^m$ could be written as a *convolution* with the impulse,

$$u[k] = (\delta * u)[k] = \sum_{j=-\infty}^{\infty} \delta[k-j]u[j]. \tag{2.292}$$

If we seek a continuous-time analogue of the discrete-time unit impulse, perhaps we should define a continuous-time analogue of a convolution sum and a continuous-time signal δ for which $u = \delta * u$.

First, let's define an appropriate notion of continuous-time convolution. Thus far, when translating between discrete and continuous-time, we've swapped sums for integrals wherever they appear. In order to develop a continuous-time convolution, it's therefore natural to replace a convolution sum with a convolution integral.

Definition 2.33 (Convolution Integral) Consider two continuous-time signals $u(\cdot)$ and $H(\cdot)$ of compatible dimension, for which $H(t) \cdot u(t) \in \mathbb{R}^p$. The convolution of H and u is the signal $(H * u)(\cdot) : \mathbb{R} \to \mathbb{R}^p$, defined

$$(H * u)(t) = \int_{\mathbb{R}} H(t-\tau)u(\tau)d\tau. \tag{2.293}$$

Remark 2.36 Above, we make use of the notation $\int_{\mathbb{R}}$ to indicate that the integral is being taken from $-\infty$ to $+\infty$. As in the case of the convolution sum, one must take special care to

ensure the convolution integral converges before performing any operations, as the integral is being taken over all of $\mathbb{R}$.

Remark 2.37 By "compatible dimension," we mean the same thing as in the discrete-time case—the dimensions of H and u must be such that the product $H(t) \cdot u(t)$ is defined.

Now that we've defined a continuous-time notion of convolution, we can define a continuous-time unit impulse function, $\delta(\cdot) : \mathbb{R} \to \mathbb{R}$. If we translate directly from discrete-time to continuous-time, we find that the continuous-time unit impulse function should satisfy,

$$u(t) = (\delta * u)(t) = \int_{\mathbb{R}} \delta(t-\tau)u(\tau)d\tau, \ \forall t \in \mathbb{R}, \tag{2.294}$$

for all piecewise continuous signals $u : \mathbb{R} \to \mathbb{R}^m$. What *is* this continuous-time function δ? The following proposition gives quite a worrying answer to this question: δ *cannot* exist.

Proposition 2.23 (Nonexistence of the δ Function) *There is no integrable*[7] *function $\delta : \mathbb{R} \to \mathbb{R}$ such that for all $u \in PC(\mathbb{R}, \mathbb{R})$, $\int_{\mathbb{R}} \delta(t-\tau)u(\tau)d\tau = u(t)$, $\forall t \in \mathbb{R}$.*

Remark 2.38 Notice that we state this proposition for a signal $u : \mathbb{R} \to \mathbb{R}$, rather than a signal $u : \mathbb{R} \to \mathbb{R}^m$. Since integrals of vector-valued functions are evaluated element-wise, the scalar case is sufficient to conclude the vector case.

Proof (Sketch) Suppose for contradiction that there exists a function $\delta : \mathbb{R} \to \mathbb{R}$ satisfying $u(t) = \int_{\mathbb{R}} \delta(t-\tau)u(\tau)d\tau$, for all $u \in PC(\mathbb{R}, \mathbb{R})$. By necessity, such a function must be zero almost everywhere[8]—if this were not the case, one would not have $u(t) = \int_{\mathbb{R}} \delta(t-\tau)u(\tau)d\tau$ for all u (draw some pictures to convince yourself of this). Now, consider the constant signal, $u(t) \equiv 1$. By assumption, for all $t \in \mathbb{R}$,

$$u(t) = 1 = \int_{\mathbb{R}} \delta(t-\tau)u(\tau)d\tau = \int_{\mathbb{R}} \delta(t-\tau)d\tau, \tag{2.295}$$

which implies $\int_{\mathbb{R}} \delta(\tau)d\tau = 1$. But, since δ is zero almost everywhere, it must be that $\int_{\mathbb{R}} \delta(\tau)d\tau = 0$—contradiction! We conclude a δ function cannot exist. □

This presents us with quite the problem! In order to define the impulse response of a continuous-time system, we need a continuous-time impulse signal with the correct convolution property. Yet, we've just shown that no such signal exists!

Consequently, in order to provide a proper definition of the continuous-time impulse, we must look beyond simple signals and turn to the theory of *distributions*, or "generalized signals," as they're sometimes called. We now provide an informal definition of a special distribution, called the *Dirac delta distribution*, which has the property we're looking for.

Definition 2.34 ((Informal) Dirac Delta Distribution) Let $\mathcal{E} \subseteq \{f : \mathbb{R} \to \mathbb{C}\}$ be a vector space of complex-valued functions, called *test functions*. The Dirac delta distribution on $\mathcal{E}$ is a linear map $\delta : \mathcal{E} \to \mathbb{C}$, equipped with an operation $*$, called convolution, for which δ and $*$ satisfy the following:

[7] Here, by *integrable*, we mean a function whose magnitude has a well-defined Lebesgue integral over $\mathbb{R}$. If you're unfamiliar with the Lebesgue integral, feel free to treat it as a fancy-sounding Riemann integral—we won't focus on the technical details of integration theory in this course.

[8] A function is zero *almost everywhere* if it is zero everywhere but on a set of Lebesgue measure zero. Lebesgue measure zero sets are small sets (e.g. $\{0\}$, $\mathbb{N}$, etc.) that are "ignored" by Lebesgue integrals.

1. Origin: $\delta(f) = f(0)$ for all $f \in \mathcal{E}$.
2. Impulse: $(\delta * f)(t) = f(t)$, for all $t \in \mathbb{R}$ and $f \in \mathcal{E}$.
3. Linearity: $(\delta * (\alpha f + \beta g))(t) = \alpha f(t) + \beta g(t)$, for all $t \in \mathbb{R}, \alpha, \beta \in \mathbb{C}$, and $f, g \in \mathcal{E}$.
4. Associativity: $\delta * (f * g) = (\delta * f) * g$ for all $f, g \in \mathcal{E}$, where $f * g$ represents the (standard) integral convolution of f and g.

We define the *shifted* Dirac delta distribution to be the map $\delta^{t_0} : \mathcal{E} \to \mathbb{C}$, satisfying $\delta^{t_0}(f) = f(t_0)$ and $(\delta^{t_0} * f)(t) = f(t - t_0)$, where $t_0 \in \mathbb{R}$ is a fixed time.

Remark 2.39 It's *critical* to note—the Dirac delta distribution is *not* a signal! Instead of taking in *time* (like a signal might), it directly takes in *signals* (from the space of test functions). It is this aspect of the definition that lets us work around the problem posed by Proposition 2.23. We have simply defined the Dirac delta distribution to be a map on a space of signals that gives us exactly the properties we want.

Remark 2.40 There are a few things that require sharpening in order to make this definition formal. First, the space $\mathcal{E}$ of test functions is usually chosen to have some "nice" structure. For instance, $\mathcal{E}$ is often taken to be a set of C^∞ functions with compact support. Secondly, convolution should be defined for *all* distributions, not just for the Dirac δ, and should be confirmed to interact well with the standard integral convolution. There are a couple of other technical points—regarding convergence and continuity properties—that must be satisfied to form a proper definition. We direct the interested reader to the references at the end of the chapter for these details.

Properly motivating and studying distribution theory takes quite a bit of work—as such, in the remainder of this section, we'll give a semi-formal treatment of the continuous-time impulse response using the informal definition of the Dirac delta that we posed above.

In many engineering texts, one will find the Dirac delta written as a *function of time*, $\delta(t)$. Although this is not correct in the formal mathematical sense, in situations where rigor is *not* a concern, manipulating δ as if it were a function of time can help in gaining some basic insight into problems. As such, we quickly lay out a few *informal* ground rules for working with a Dirac delta as if it were a function of time. We stress—these rules *should not* be used in a rigorous mathematical context—the Dirac delta distribution is a map on a space of functions, *not* a regular function of time! In the remainder of this section, we'll be precise in pointing out where we use these "function of time" formulas.

Proposition 2.24 ((Informal) Dirac Delta as a Function of Time) *Let $u \in PC(\mathbb{R}, \mathbb{R})$. The following informal "rules" are used to manipulate the Dirac delta as a function of time. For all t (or $t - t_0$ for the second formula) at which $u(\cdot)$ is continuous,*

$$u(t) = (\delta * u)(t) \text{ “} = \text{” } \int_{\mathbb{R}} \delta(t - \tau) u(\tau) d\tau \tag{2.296}$$

$$u(t - t_0) = (\delta^{t_0} * u)(t) \text{ “} = \text{” } \int_{\mathbb{R}} \delta^{t_0}(t - \tau) u(\tau) d\tau. \tag{2.297}$$

If $u(\cdot)$ is continuous at 0 (or at t_0 for the second formula), then

$$u(0) = \delta(u) \text{ “} = \text{” } \int_{\mathbb{R}} \delta(\tau) u(\tau) d\tau \tag{2.298}$$

$$u(t_0) = \delta^{t_0}(u) \text{ “} = \text{” } \int_{\mathbb{R}} \delta^{t_0}(\tau) u(\tau) d\tau. \tag{2.299}$$

Remark 2.41 It's essential to remember: the integral "rules," written with equality in quotations, are *not* mathematically rigorous! It's best to think of each "rule" as *notation*, rather than as mathematical fact. In the optional, starred subsection following this, we will use a technique called *approximations to the identity* to more rigorously justify the use of these informal integral formulas.

Remark 2.42 Here, we've asked that each property hold at a time where $u(\cdot)$ is continuous. This request is a consequence of how integrals deal with discontinuity. If u had a jump discontinuity at a single point, the integral would simply *ignore* the jump discontinuity, since integrals are agnostic to changes on single-point sets. Thus, if we ask that $u(t) = \int_{\mathbb{R}} \delta(t-\tau)u(\tau)$, we should at best expect the equality to hold at points of continuity, since the integral "filters out" single point discontinuities. The idea of integrals "filtering out" function values on certain, small sets is the starting point of a rigorous theory of integration via measure theory.

Proof (Informal) These "rules" are motivated as follows. The first two formulas follow from the definition of convolution with the Dirac delta,

$$u(t) = (\delta * u)(t),\ u(t-t_0) = (\delta^{t_0} * u)(t). \tag{2.300}$$

Here, we just swap out the abstract convolution operation we defined for distributions for the integral definition of convolution. The second two formulas follow from the first two convolution integrals, and are analogous to the properties $u(0) = \delta(u)$ and $u(t_0) = \delta^{t_0}(u)$, which we stated in the definition above. □

Now that we've sketched out a basic, informal definition for the continuous-time impulse "signal," we can return to the problem of defining impulse response for a continuous-time system. Recall that, for a discrete-time linear, time-varying system, the impulse response map was defined as a map $H : \mathbb{Z} \times \mathbb{Z} \to \mathbb{R}^{p\times m}$, in which $[H[k,k_0]]_{ij}$ was the zero-state response of output coordinate i to an impulse applied in input coordinate j at time k_0.

Using the discrete-time definition as a guide, let's motivate a definition for continuous-time impulse response. Recall that the zero-state response of a continuous-time linear, time-varying system is computed,

$$\rho(t,t_0,0,u(\cdot)) = C(t)\int_{t_0}^{t} \Phi(t,\tau)B(\tau)u(\tau)d\tau + D(t)u(t). \tag{2.301}$$

Let's calculate the output of the system for the input $\delta^{t_0}e_j$, a time-shifted Dirac delta applied in input coordinate j. Using the informal integral rules we established above, we have,

$$\rho(t,t_0,0,\delta^{t_0}e_j) = C(t)\int_{t_0}^{t} \Phi(t,\tau)B(\tau)e_j\delta^{t_0}(\tau)d\tau + D(t)e_j\delta^{t_0} \tag{2.302}$$

$$= C(t)\Phi(t,t_0)B(t_0)e_j + D(t)e_j\delta^{t_0}. \tag{2.303}$$

Taking the i'th coordinate, we find that the zero-state response of coordinate i is,

$$\rho(t,t_0,0,\delta^{t_0}e_j)_i = [C(t)\Phi(t,t_0)B(t_0) + D(t)\delta^{t_0}]_{ij}. \tag{2.304}$$

This motivates the following definition for the continuous-time linear, time-varying impulse response map.

Definition 2.35 (CT-LTV Impulse Response Map) Consider a continuous-time linear, time-varying system representation $(A(\cdot), B(\cdot), C(\cdot), D(\cdot))$. The impulse response map $H(\cdot,\cdot)$ of the system representation is defined on $\mathbb{R} \times \mathbb{R}$ as,

$$H(t, t_0) = \begin{cases} C(t)\Phi(t, t_0)B(t_0) + D(t)\delta^{t_0} & t \geq t_0 \\ 0 & t < t_0. \end{cases} \tag{2.305}$$

Entry $[H(t, t_0)]_{ij} = h_{ij}(t, t_0)$ is interpreted as the i'th component of the zero-state response to an input $\delta^{t_0} e_j$ containing a shifted Dirac delta at t_0 in its j'th component.

Remark 2.43 Due to the presence of the Dirac delta when $D(\cdot)$ is nonzero, H is *not* a map that can be evaluated like a "normal" function into $\mathbb{R}^{p\times m}$, since δ^{t_0} is not something we can evaluate as a function of time. Rather, H is a tool that we will use in conjunction with integration and convolution—operations in which we understand how δ^{t_0} behaves.

Now, let's show that we can use the impulse response map to compute the zero-state response of the system to any (admissible) input. What should we expect the formula for zero-state response to look like? Recall that, for a discrete-time linear, time-varying system, the zero-state response of the system to an input $u[\cdot]$ was computed $\rho(k, k_0, 0, u[\cdot]) = \sum_{j=k_0}^{k} H[k, j]u[j]$. Translating this *sum* formula into an *integral* formula, we expect the continuous-time case to satisfy,

$$\rho(t, t_0, 0, u(\cdot)) = \int_{t_0}^{t} H(t, \tau)u(\tau)d\tau. \tag{2.306}$$

The following theorem confirms that this formula holds at every t at which the problem data is continuous.

Theorem 2.14 (Impulse Response of a Linear, Time-Varying System) *Consider an LTV system representation $(A(\cdot), B(\cdot), C(\cdot), D(\cdot))$ with impulse response map $H(\cdot,\cdot)$. For any $u \in PC(\mathbb{R}_{\geq t_0}, \mathbb{R}^m)$ and $t \in \mathbb{R}$ at which $A(\cdot), B(\cdot), C(\cdot), D(\cdot), u(\cdot)$ are continuous,*

$$\rho(t, t_0, 0, u(\cdot)) = \int_{t_0}^{t} H(t, \tau)u(\tau)d\tau. \tag{2.307}$$

Remark 2.44 Just like in the discrete-time case, when we define a signal $u \in PC(\mathbb{R}_{\geq t_0}, \mathbb{R}^m)$, we will automatically identify it with the extended signal $u_e \in PC(\mathbb{R}, \mathbb{R}^m)$, which is zero for $t < t_0$ and equal to u for $t \geq t_0$. This way, an input signal $u \in PC(\mathbb{R}_{\geq t_0}, \mathbb{R}^m)$ can be passed into the I/O map ρ without trouble.

Proof (Informal) We'll provide an informal proof of this result using the informal "rules" for manipulating δ as if it were a function of time. In particular, we will use the "rule" that, for every t at which $u(\cdot)$ is continuous,

$$u(t) = \int_{\mathbb{R}} \delta(t - \tau)u(\tau)d\tau. \tag{2.308}$$

Since we are given $u : \mathbb{R}_{\geq t_0} \to \mathbb{R}^m$, we will perform our standard identification of u with a signal $u_e : \mathbb{R} \to \mathbb{R}^m$, defined on all of $\mathbb{R}$, by setting $u(t) = 0$ for $t < t_0$. Plugging into the formula for the zero-state response of a CT-LTV system yields,

$$\rho(t, t_0, 0, u(\cdot)) = C(t) \int_{t_0}^{t} \Phi(t, \tau) B(\tau) u(\tau) d\tau + D(t) u(t) \tag{2.309}$$

$$= C(t) \int_{-\infty}^{t} \Phi(t, \tau) B(\tau) u(\tau) d\tau + D(t) u(t), \tag{2.310}$$

since the input signal is taken to be zero for $t < t_0$. Letting $\mathbb{1}_{\tau \leq t}(\tau)$ denote the indicator function which is equal to 1 when $\tau \leq t$ and zero elsewhere, it follows that,

$$= \int_{\mathbb{R}} \mathbb{1}_{\tau \leq t}(\tau) C(t) \Phi(t, \tau) B(\tau) u(\tau) d\tau + D(t) \int_{\mathbb{R}} \delta(t - \tau) u(\tau) d\tau \tag{2.311}$$

$$= \int_{\mathbb{R}} \mathbb{1}_{\tau \leq t}(\tau) C(t) \Phi(t, \tau) B(\tau) u(\tau) d\tau + \int_{\mathbb{R}} \mathbb{1}_{\tau \leq t}(\tau) D(t) \delta(t - \tau) u(\tau) d\tau \tag{2.312}$$

$$= \int_{\mathbb{R}} \mathbb{1}_{\tau \leq t}(\tau) [C(t) \Phi(t, \tau) B(\tau) + D(t) \delta(t - \tau)] u(\tau) d\tau, \tag{2.313}$$

where we use our informal integral rules to add the indicator to the $D(t)$ integral. Now, we recognize the term in brackets as $H(t, \tau)$. We therefore have,

$$\rho(t, t_0, 0, u(\cdot)) = \int_{\mathbb{R}} \mathbb{1}_{\tau \leq t}(\tau) H(t, \tau) u(\tau) d\tau \tag{2.314}$$

$$= \int_{\tau \leq t} H(t, \tau) u(\tau) d\tau \tag{2.315}$$

$$= \int_{t_0}^{t} H(t, \tau) u(\tau) d\tau, \tag{2.316}$$

where in the last step, we use that $u(\tau)$ is zero for $\tau \leq t_0$. □

Next, we specialize to the LTI case. As with the discrete-time case, we recognize that time passed, rather than objective start and end times, is the relevant quantity.

Definition 2.36 (CT-LTI Impulse Response Map) Consider a continuous-time, LTI system with impulse response map $H(\cdot, \cdot)$ on $\mathbb{R} \times \mathbb{R}$. The LTI impulse response map of the system is the map $H(\cdot)$ on $\mathbb{R}$, defined

$$H(t) = H(t, 0), \ \forall t \in \mathbb{R}. \tag{2.317}$$

As with the discrete-time case, this map satisfies $H(t, t_0) = H(t - t_0)$ due to time-invariance. We now compute the impulse response map of a continuous-time, LTI system, and prove that the zero-state response of such a system can be computed via a convolution.

Corollary 2.3 (Impulse Response of a Linear, Time-Varying System) *Consider a continuous-time, LTI system representation (A, B, C, D). The LTI impulse response map of the system is computed,*

$$H(t) = \begin{cases} C \exp(At) B + D\delta & t \geq 0 \\ 0 & t < 0. \end{cases} \tag{2.318}$$

For any $u \in PC(\mathbb{R}_{\geq 0}, \mathbb{R}^m)$ and any $t \in \mathbb{R}$ at which $u(\cdot)$ is continuous,

$$\rho(t, 0, 0, u(\cdot)) = (H * u)(t). \tag{2.319}$$

Proof The formula for H follows directly from Definition 2.35 and application of the formula $\Phi(t, t_0) = \exp(A(t - t_0))$ for a continuous-time, LTI system. The convolution formula for the output response follows directly from Theorem 2.14 and the definition of continuous-time convolution. We have, for $u(\cdot) \in PC(\mathbb{R}_{\geq 0}, \mathbb{R}^m)$ and any $t \geq 0$ at which $u(\cdot)$ is continuous,

$$\rho(t, 0, 0, u(\cdot)) = \int_0^t H(t - \tau)u(\tau)d\tau = \int_{\mathbb{R}} H(t - \tau)u(\tau)d\tau = (H * u)(t). \tag{2.320}$$

Thus, the desired formula holds. □

2.4.3 Approximations to the Identity ★

Above, we gave an informal introduction to the continuous-time impulse response using an informal definition of the Dirac delta. In doing this, we introduced a number of "rules" for manipulating the Dirac delta as if it were a function of time. In this optional subsection, we provide a firmer theoretical justification for these "rules," and see that we can *approximate* the behavior of the Dirac delta distribution with a sequence of well-behaved functions. Note that this section can be skipped without much loss of continuity—however, skimming the basic ideas and looking at the figures could be helpful for your understanding. Let's get started!

Above, we proved that there is *no* continuous-time signal which has the convolution property of the Dirac delta. In order to get around this problem, one may take two approaches. First, one may take the approach we hinted at above, and learn some (formidable) distribution theory. Second, instead of studying the Dirac delta via distribution theory, one can try to *approximate* the Dirac delta using a sequence of continuous-time signals that—in the limit—give us the same convolution behavior as the *distribution* δ. Here, we'll provide a brief overview of the second approach. Notably, by approximating δ with a family of well-behaved signals, we'll be able to draw connections between the informal integration "rules" we outlined above and formal mathematical facts.

How do we approximate the Dirac delta *distribution* (which is a map on a space of functions) with continuous-time signals? Let's turn to the informal integral "rules" for manipulating the Dirac delta as if it were a signal to get some ideas. Ideally, we want a family of approximations, $\delta_\epsilon : \mathbb{R} \to \mathbb{R}$, parameterized by $\epsilon > 0$, such that for every admissible signal $u(\cdot)$ and sufficiently small ϵ,

$$(\delta_\epsilon * u)(t) = \int_{\mathbb{R}} \delta_\epsilon(t - \tau)u(\tau)d\tau \approx u(t). \tag{2.321}$$

Further, we'd like this approximation to become exact as $\epsilon \to 0$. In order for this to happen, it seems like $\delta_\epsilon(t - \tau)$ should have a high value near t, and small values elsewhere. That is, $\delta_\epsilon(t - \tau)$ should have a *sharp peak* near t and should drop off to zero elsewhere—this way, the integral will "pick out" the value of $u(\cdot)$ at time t. With this in mind, we define a class of "well-behaved" approximations of δ, called *approximations to the identity*.

Definition 2.37 (Approximation to the Identity) An approximation to the identity is a collection $\{\delta_\epsilon\}_{\epsilon>0}$ of integrable functions $\delta_\epsilon : \mathbb{R} \to \mathbb{R}$, for which the following are satisfied:

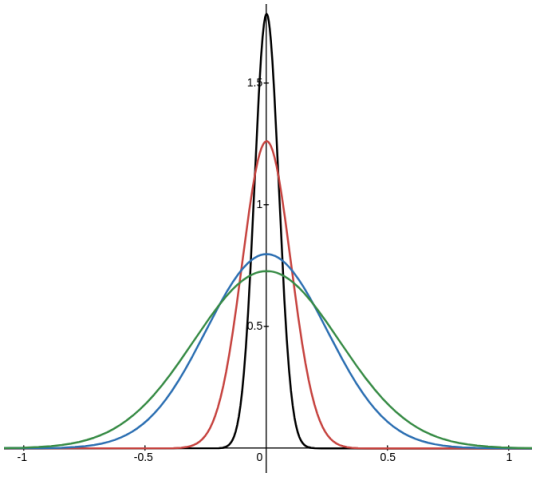


Fig. 2.5 A series of approximations of the δ "function," formed by Gaussian distributions of successively smaller variances, which place more and more mass at the origin as $\epsilon \to 0$. Once ϵ "equals" zero, all of the mass is placed at the origin, and the approximation appears to be a "spike" that jumps to ∞ at the origin.

1. Unit mass: for all $\epsilon > 0$, $\int_{\mathbb{R}} \delta_\epsilon(t)dt = 1$.
2. Uniform bound: there exists an $A > 0$ such that for all $\epsilon > 0$, $\int_{\mathbb{R}} |\delta_\epsilon(t)|dt \leq A$.
3. Limiting behavior: for every $\eta > 0$, $\int_{|t|\geq\eta} |\delta_\epsilon(t)|dt \to 0$ as $\epsilon \to 0$.

For $t_0 \in \mathbb{R}$, an approximation to the identity at t_0 is a family $\{\delta_\epsilon^{t_0}\}_{\epsilon>0}$ for which $\{\delta_\epsilon\}_{\epsilon>0}$, $\delta_\epsilon(t) := \delta_\epsilon^{t_0}(t + t_0)$, is an approximation to the identity.

Remark 2.45 As we'll find out below, this definition is sufficient to enable pointwise convergence theorems for convolutions with piecewise continuous functions. If one wants stronger convergence modes or convergence results on more general classes of signals, stronger versions of (2) and (3) should be imposed—see Chapter 3.2 of [34] for further details.[9]

Remark 2.46 The name *approximation to the identity* follows from the fact that δ is the "identity" of the convolution, $\delta * u = u$. Since approximations to the identity yield approximations of δ, it follows that they approximate the "identity" of the convolution operation.

As demonstrated in Figure 2.5, approximations to the identity shift all of their mass to the origin as $\epsilon \to 0$. Since approximations to the identity must *also* satisfy $\int_{\mathbb{R}} \delta_\epsilon(t)dt = 1$, this shifting property implies that an approximation must "spike" towards infinity at 0 as $\epsilon \to 0$. This is where the *heuristic* definition of a continuous-time impulse function as the spike,

$$\delta(t) \text{ “} = \text{”} \begin{cases} \infty & t = 0 \\ 0 & t \neq 0, \end{cases} \tag{2.322}$$

originates from. Although is a nice picture to have in mind when reasoning about the continuous-time impulse at a non-rigorous level, one should always remember—this is *not*

[9] The definition above is referred to as a *good kernel* in [34] and as an *approximation to the identity* in [36]. Here, our definition is consistent with [36].

the definition of δ—Proposition 2.23 tells us that a δ signal cannot exist. The Dirac delta is truly a map on a function space, *not* a continuous-time signal. Let's consider a simple example of an approximation to the identity.

Example 2.3 Consider the family of functions, $\delta_\epsilon : \mathbb{R} \to \mathbb{R}$, defined

$$\delta_\epsilon(t) = \begin{cases} 0 & t < 0 \\ 1/\epsilon & t \in [0, \epsilon] \\ 0 & t > \epsilon. \end{cases} \tag{2.323}$$

We now verify this satisfies the properties of an approximation to the identity. We have that $\int_{\mathbb{R}} \delta_\epsilon(t)dt = 1$ for all $\epsilon > 0$ and that $\int_{\mathbb{R}} |\delta_\epsilon(t)|dt = 1$, from which we conclude the first two conditions. Now, fix $\eta > 0$. For $\eta > \epsilon$, $\int_{|t|\geq\eta} |\delta_\epsilon(t)|dt = 0$, which implies that for every $\eta > 0$, $\int_{|t|\geq\eta} |\delta_\epsilon(t)|dt \to 0$ as $\epsilon \to 0$. Thus, the family $\{\delta_\epsilon\}_{\epsilon>0}$ is an approximation to the identity.

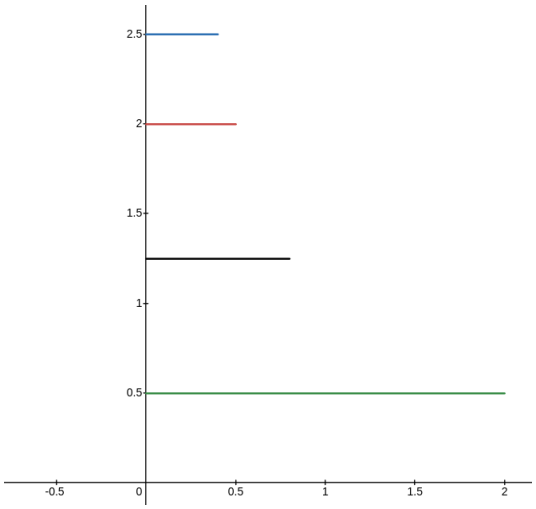


Fig. 2.6 A second approximation to the identity, formed from $\delta_\epsilon(t) = 1/\epsilon$, for $t \in [0, \epsilon]$.

We now apply these approximations to the identity to study the convolution integral.

Theorem 2.15 (Approximations to the Identity Converge to δ) *Let $u \in PC(\mathbb{R}, \mathbb{R}^m)$. For $\{\delta_\epsilon\}_{\epsilon>0}$ an approximation to the identity, at every $t \in \mathbb{R}$ at which u is continuous,*

$$\lim_{\epsilon\to 0} \int_{\mathbb{R}} \delta_\epsilon(t-\tau)u(\tau)dt = \lim_{\epsilon\to 0} (\delta_\epsilon * u)(t) = u(t). \tag{2.324}$$

Remark 2.47 This property rigorously justifies our integral "rules" for treating the Dirac delta function as a continuous-time signal. Recall that earlier, we stated that if one manipulates δ as if it were a function of time, one has the "rule,"

$$\int_{\mathbb{R}} \delta(t-\tau)u(\tau)d\tau \text{ " = " } u(t), \tag{2.325}$$

at every t at which $u(\cdot)$ is continuous. Now, we observe that this "rule" is really just special notation for the limiting case of Theorem 2.15. Theorem 2.15 therefore provides theoretical grounding for the informal notation we introduced when manipulating δ as if it were a function of time.

Proof Consult Chapter 2.4 of [36]. □

This result tells us that at every point of continuity, an approximation to the identity gives us exactly what we want—a way to precisely approximate the convolution behavior of the Dirac delta distribution using *signals* that are functions of time. Now, we study how approximations to the identity interact with impulse response. For simplicity, we will focus on the time-invariant case.

Definition 2.38 (Approximate LTI Impulse Response Map) Consider a continuous-time LTI system representation (A, B, C, D) and an approximation to the identity $\{\delta_\epsilon\}_{\epsilon>0}$ consisting of piecewise continuous functions. The approximate LTI impulse response map with respect to $\{\delta_\epsilon\}_{\epsilon>0}$ is the map $H_\epsilon(\cdot) : \mathbb{R} \to \mathbb{R}^{p\times m}$, parameterized by $\epsilon > 0$ and defined,

$$H_\epsilon(t) = \begin{cases} C\exp(At)B + D\delta_\epsilon(t) & t \geq 0 \\ 0 & t < 0. \end{cases} \tag{2.326}$$

Remark 2.48 Unlike in the previous subsection, where the continuous-time impulse response map was *not* a map we could evaluate as a function of time (due to the presence of the Dirac delta distribution), $H_\epsilon(\cdot)$ *is* a map we can evaluate as a function of time. This is because $\delta_\epsilon(\cdot)$ is a piecewise continuous function of time, and is therefore a true signal.

Using the approximate LTI impulse response map, we state an approximate version of Corollary 2.3 that relies on approximations to the identity.

Proposition 2.25 (LTI Response via Approximation) *Consider a continuous-time, LTI system representation (A, B, C, D) with I/O map ρ. Let $\{\delta_\epsilon\}_{\epsilon>0}$ be an approximation to the identity consisting of $\delta_\epsilon \in PC(\mathbb{R}_{\geq 0}, \mathbb{R})$, and $H_\epsilon(\cdot)$ the approximate LTI impulse response map with respect to $\{\delta_\epsilon\}_{\epsilon>0}$. For any $u(\cdot) \in PC(\mathbb{R}_{\geq 0}, \mathbb{R}^m)$, and any $t \in \mathbb{R}$ at which $u(\cdot)$ is continuous,*

$$\rho(t, 0, 0, u(\cdot)) = \lim_{\epsilon\to 0} \int_{\mathbb{R}} H_\epsilon(t-\tau)u(\tau)d\tau = \lim_{\epsilon\to 0}(H_\epsilon * u)(t). \tag{2.327}$$

Remark 2.49 As a consequence of how we defined the approximate LTI impulse response map, we restrict ourselves to approximations defined on $\mathbb{R}_{\geq 0}$ in this proposition.

Proof Consider a piecewise continuous input signal $u(\cdot) \in PC(\mathbb{R}_{\geq 0}, \mathbb{R}^m)$. If $\mathbb{1}_{t\geq\tau}(\tau)$ represents the indicator function of $t \geq \tau$, the convolution $H_\epsilon * u$ is computed at time t as,

$$(H_\epsilon * u)(t) = \int_{\mathbb{R}} H_\epsilon(t-\tau)u(\tau)d\tau \tag{2.328}$$

$$= \int_{\mathbb{R}} \mathbb{1}_{t\geq\tau}(\tau)[C\exp(A(t-\tau))B + D\delta_\epsilon(t-\tau)]u(\tau)d\tau \tag{2.329}$$

$$= \int_{\mathbb{R}} \mathbb{1}_{t\geq\tau}(\tau)C\exp(A(t-\tau))Bu(\tau)d\tau + D\int_{\mathbb{R}} \delta_\epsilon(t-\tau)u(\tau)d\tau, \tag{2.330}$$

where we drop the indicator on the D integral under the assumption that each function $\delta_\epsilon(\cdot) \in PC(\mathbb{R}_{\geq 0}, \mathbb{R})$. Now, suppose t is a time at which $u(\cdot)$ is continuous. Applying Theorem 2.15, it follows that, as $\epsilon \to 0$,

$$\lim_{\epsilon \to 0}(H_\epsilon * u)(t) = \int_{\mathbb{R}} \mathbb{1}_{t \geq \tau}(\tau) C \exp(A(t-\tau)) B u(\tau) d\tau + \lim_{\epsilon \to 0} D \int_{\mathbb{R}} \delta_\epsilon(t-\tau) u(\tau) d\tau \tag{2.331}$$

$$= \int_{\mathbb{R}} \mathbb{1}_{t \geq \tau}(\tau) C \exp(A(t-\tau)) B u(\tau) d\tau + Du(t) \tag{2.332}$$

$$= C \int_0^t \exp(A(t-\tau)) B u(\tau) d\tau + Du(t) \tag{2.333}$$

$$= \rho(t, 0, 0, u(\cdot)), \tag{2.334}$$

where in the final step, we recognize the formula for the zero-state response. □

Thus, we can gain a similar result to Corollary 2.3 *without* the use of distributions—all we needed to state the proposition above was a family of well-behaved, continuous-time signals.

2.4.4 The Laplace Transform

Let's summarize what we've done so far in this section. First, we determined that the response of a linear I/O system to any (sufficiently regular) input signal can be written in terms of the impulse response map. Following this, we showed that, for linear, time-invariant systems, the output is equal to the *convolution* of the impulse response and input signal. What's next in our study of I/O systems?

While the convolution sums and integrals we defined certainly provide clean expressions for the output of a system, they *are* still composed of sums and integrals that might be challenging to analyze directly. Can we do better than the convolution formulas? Let's set the stage by laying out a "wish list" of what we'd like in our analysis.

1. State equations become algebraic: state equations are generally challenging to work with. If we can reduce the state equations (which are ordinarily differential equations or recurrence relations) to purely *algebraic* equations, perhaps we'll be able to gain more insight into our systems.
2. Simplify convolutions: convolutions are also challenging operations to work with. We'd like a way to simplify convolution to an easy to compute operation.
3. Analyze long-term behavior: we'd like a way to study the long-term behavior of our system without performing hard analysis.
4. Simple interpretation: we should be able to determine important features of our systems by inspection.

Although this list seems a little ambitious, it turns out that in both the continuous and discrete-time LTI cases, we'll be able to meet every single point through the use of *transforms*. An amazing and quite surprising insight into this problem is that by transforming our objects of study into the complex plane, we can simplify the study of I/O systems to the study of complex, rational functions. What's more, under these transformations, convolution simplifies to *multiplication*.

We'll begin our study of these "magic transforms" in continuous-time, where we'll focus on the *Laplace transform*, and will then move to discrete-time, where we'll focus on the $\mathcal{Z}$-transform. Here, we'll just touch upon the most basic aspects of these transforms—we'll develop some more nuanced aspects of transform theory as the course proceeds.

2.4.4.1 A Little Complex Analysis

Above, we mentioned that our approach to achieving the points of our wish list will involve *transforming* system objects into objects in the complex plane. As such, it behooves us to learn the basic language of complex analysis, the study of functions in the complex plane.

First, we'll state some basic notation. Typically, when referring to a complex number, we'll use either s or $z \in \mathbb{C}$. Every complex number $s \in \mathbb{C}$ admits a unique decomposition,

$$s = \mathrm{Re}(s) + j\,\mathrm{Im}(s), \tag{2.335}$$

where $\mathrm{Re}(s) \in \mathbb{R}$ is the *real part* of s, $\mathrm{Im}(s) \in \mathbb{R}$ is the *complex part* of s, and $j = \sqrt{-1}$.[10] We stress—both the real *and* the imaginary part of a complex number are *real numbers*. The complex magnitude is a map $|\cdot| : \mathbb{C} \to \mathbb{R}$, taking,

$$s \mapsto |s| = \sqrt{(\mathrm{Re}(s))^2 + (\mathrm{Im}(s))^2}. \tag{2.336}$$

Since both $\mathrm{Re}(s)$ and $\mathrm{Im}(s)$ are real for all $s \in \mathbb{C}$, the complex magnitude is a *real-valued* function. One may show that the complex magnitude defines a *norm* on the complex numbers, which makes $(\mathbb{C}, |\cdot|)$ a normed vector space over the field $\mathbb{C}$.

Since $(\mathbb{C}, |\cdot|)$ has the structure of a normed vector space, all of the analysis tools we've developed for normed vector spaces immediately apply. Thus, using the complex magnitude, one can develop a notion of open and closed sets in $\mathbb{C}$, just like one might in $\mathbb{R}$. In particular, a set $\Omega \subseteq \mathbb{C}$ is declared *open* if, for all $s_0 \in \Omega$, there exists an $\epsilon > 0$ such that,

$$B_\epsilon(s_0) = \{s \in \mathbb{C} : |s - s_0| < \epsilon\} \subseteq \Omega. \tag{2.337}$$

Likewise, a set $\Omega \subseteq \mathbb{C}$ is declared *closed* if its complement Ω^c is open, and bounded if there exists an $M \geq 0$ such that $|s| \leq M$ for all $s \in \Omega$. A set Ω is *compact* if and only if it is closed and bounded. Using these definitions, one may state a standard ϵ-δ definition for continuity of complex functions, identical to that in a normed vector space.

An important function on $\mathbb{C}$ is the *complex exponential*, $\exp : \mathbb{C} \to \mathbb{C}$, which is defined,

$$e^s = e^{\mathrm{Re}(s)} e^{j\,\mathrm{Im}(s)} := e^{\mathrm{Re}(s)}[\cos(\mathrm{Im}(s)) + j\sin(\mathrm{Im}(s))],\ \forall s \in \mathbb{C}, \tag{2.338}$$

where $e^{(\cdot)}$, $\sin(\cdot)$, and $\cos(\cdot)$ are the real exponential, sine, and cosine functions. It's important to note: this is the *definition* of the exponential, not a theorem! The exponential gives us another way to write a complex number. Any $s \in \mathbb{C}$ can be written in *polar form* as,

$$s = re^{j\theta},\ r \geq 0, \theta \in [0, 2\pi). \tag{2.339}$$

[10] To quote [19] in reference to the use of j for $\sqrt{-1}$, "Only electrical engineers, and those under their influence, use this crazy notation." If you feel more comfortable using $i = \sqrt{-1}$, you're welcome to do so—we won't consider any applications in circuit theory in this text, so there is no danger of conflating $i = \sqrt{-1}$ with electrical current i.

Note that, for $|s| > 0$, the polar coordinates (r, θ) for a given complex number are unique (up to θ modulo 2π). For $|s| = 0$, this is not the case, as θ becomes nonunique. The number r is to be interpreted as the magnitude of s, $r = |s|$, and θ as the angle between the vector $(\mathrm{Re}(s), \mathrm{Im}(s))$ and the axis $(1, 0)$. One may show that, for s with $\mathrm{Re}(s) \neq 0$, $\theta = \arctan(\mathrm{Im}(s)/\,\mathrm{Re}(s))$.

Now that we've discussed the very basics of complex numbers, we can talk about *calculus* in the complex plane. Let $\Omega \subseteq \mathbb{C}$ be an open set and $f : \Omega \to \mathbb{C}$ be a complex function. The function f is said to be *differentiable* at $s_0 \in \mathbb{C}$ if,

$$\lim_{s \to s_0} \frac{f(s) - f(s_0)}{s - s_0}, \tag{2.340}$$

exists. In this case, the limit is declared to be the *derivative* of f at s_0, denoted $f'(s_0)$. It's important to note—although this definition looks identical to the definition in $\mathbb{R}$, here, the limit is being taken in the complex plane. We give differentiable complex functions the following special name.

Definition 2.39 (Analytic Function) Let $\Omega \subseteq \mathbb{C}$ be an open set. A function $f : \Omega \to \mathbb{C}$ is *analytic* at $s_0 \in \Omega$ if there exists an $\epsilon > 0$ such that f is differentiable at every point in $B_\epsilon(s_0)$. If f is analytic at every $s_0 \in \Omega$, then f is said to be *analytic on* Ω.

Remark 2.50 In the theory of single-variable complex functions the terms "analytic" and "holomorphic" are often used interchangeably. One might see an *analytic* function defined as a function with a convergent power series in a neighborhood of a point, and a *holomorphic* function being a (complex) differentiable function. For single-variable complex functions, these two definitions are *equivalent*, and can be used interchangeably. Since differentiability is an easier condition to check, we define analyticity via differentiability.

Amazingly, one may show that a function $f : \Omega \to \mathbb{C}$ is analytic at $s_0 \in \mathbb{C}$ if and only if it has a convergent power series expansion in a neighborhood of s_0. This is *not* true even for infinitely differentiable functions[11] in $\mathbb{R}$. These basic properties hint that complex analysis can often be more revealing than real analysis—transforming a real function into the complex plane immediately gives us access to these new results!

Before we get too carried away, it's important to perform a few sanity checks regarding the complex derivative. Since the complex derivative has essentially the same definition as the derivative in $\mathbb{R}$ (with the limit being taken in $\mathbb{C}$ instead of $\mathbb{R}$), it enjoys all of the same basic properties (linearity, chain rule, product rule, etc.) as the real derivative.

Theorem 2.16 (Properties of the Complex Derivative) *Let $f, g : \Omega \to \mathbb{C}$ be analytic functions on an open set $\Omega \subseteq \mathbb{C}$.*

1. *Linearity: $f + g$ is analytic in Ω and $(f + g)' = f' + g'$.*
2. *Product Rule: fg is analytic in Ω and $(fg)' = f'g + fg'$.*
3. *Chain Rule: If $U \subseteq \mathbb{C}$ is open, $h : U \to \mathbb{C}$ is analytic on U, and $f(\Omega) \subseteq U$, then $h \circ f$ is analytic on Ω with $(h \circ f)'(s) = h'(f(s))f'(s)$.*

Exercise 2.23 Write a version of Theorem 2.16 in which each function f, g, h is analytic at a point. Use Theorem 2.16 to prove your theorem.

We refer the reader to [35] for proofs of the results stated in this section, as well as for a more thorough treatment of complex analysis.

[11] A counterexample can be constructed by taking a nonzero, smooth function which is very *flat* in a region around the origin. The function $f(x) = e^{-1/x},\ x < 0, f(x) = 0,\ x \leq 0$ is an example of this.

2.4.4.2 Definition & Basic Properties of the Laplace Transform

Now that we've reviewed some of the basic language of complex analysis, we're ready to describe transformations which take *real* functions and transform them into *complex* functions. In this section, we'll study the Laplace transform, which performs this transformation for certain classes of continuous-time signals. We'll first present the definition of the transform and then discuss its properties.

Definition 2.40 (Laplace Transform) Consider a piecewise continuous, matrix-valued signal, $f(\cdot) \in PC(\mathbb{R}, \mathbb{R}^{n\times m})$. The one-sided Laplace transform of f is the complex-valued function, $\hat{F} : \Omega \to \mathbb{C}^{n\times m}$, defined

$$\hat{F}(s) = \mathcal{L}(f)(s) := \lim_{\epsilon\to 0^-} \int_{\epsilon}^{\infty} e^{-st} f(t) = \int_{0^-}^{\infty} e^{-st} f(t) dt, \tag{2.341}$$

where $\Omega := \{s \in \mathbb{C} : \mathcal{L}(f)(s) \text{ converges absolutely}\}$ is called the *region of absolute convergence* of $\hat{F}$.

Remark 2.51 Here, we define the Laplace transform using an integral of a matrix-valued function. Recall that integrals of matrix-valued functions are defined *element-wise*—that is, we integrate each element of the matrix-valued function individually. In order to take the Laplace transform of a matrix-valued function, we therefore take the transform of each element function of the matrix.

Remark 2.52 The notation 0^- means that the lower bound of the integral should be taken in the limit as 0 is approached from the *left*. This lets the Laplace transform interact well with approximations to the identity, which approximate the Dirac delta. In particular, if we did not include this left limit, the Laplace transform would *not* correctly interact with approximations that are defined on all of $\mathbb{R}$. If we're given a signal $f(\cdot)$ that is only defined on $\mathbb{R}_{\geq 0}$, we compute the limit as $t \to 0^-$ under the assumption that $f(t) = 0$ for $t < 0$—this aligns with our "standard" way of extending the domains of signals to all of $\mathbb{R}$.

Remark 2.53 Typically, we will use a capital letter with a hat to denote the Laplace transform of a signal. For instance, if f is a signal, we will denote by $\hat{F}$ its Laplace transform. In some circumstances, one will see the transform written without the hat—we'll use the hat here for the sake of clarity. This is useful when writing the transform of a signal which is already specified by a capital letter.

Remark 2.54 By inspection, we can tell that, for an arbitrary signal, the Laplace transform is *not* guaranteed to converge for every $s \in \mathbb{C}$. For instance, if f grows faster than $\text{Re}(e^{st})$, the integral will diverge at s. This leads us to define $\hat{F}$ on the region of absolute convergence, Ω. We'll often refer to the "region of absolute convergence" as the "region of convergence," and will use the letters R.O.C. for shorthand.

From this definition, we observe that a Laplace transform integrates f (which is a function of t) against e^{-st} (which is function of t and s), to get a function of s. Since the time variable is integrated out, *all that remains* when the transform is computed is a function of s. Just like we refer to t as a "time" variable, we will refer to s as *frequency* variable. For now, we will justify the use of this language via the complex exponential,

$$e^{st} = e^{\text{Re}(st)}(\cos(\text{Im}(st)) + j\sin(\text{Im}(st))). \tag{2.342}$$

This function, which appears in the Laplace transform, is defined in terms of sines and cosines whose frequencies are determined by s. We'll explore a deeper connection between s and frequency later in the course, when we discuss *frequency response*. In line with this language, we refer to the real numbers $\mathbb{R}$ as the time domain and the complex numbers $\mathbb{C}$ as the *frequency domain*. Thus, one says that the Laplace transform takes a signal from the *time domain* and transforms it into the *frequency domain*.

It's *extremely* important to note that the Laplace transform we've defined above is only *one-sided*. This means that the integral in the transform *does not* extend to $t = -\infty$. This one-sided definition reflects the fact that the LTI systems we're interested in analyzing are *causal*. Above, we showed that the impulse response map of a causal LTI system is zero for $t < 0$—starting the transform at $t = 0^-$ is therefore justifiable.

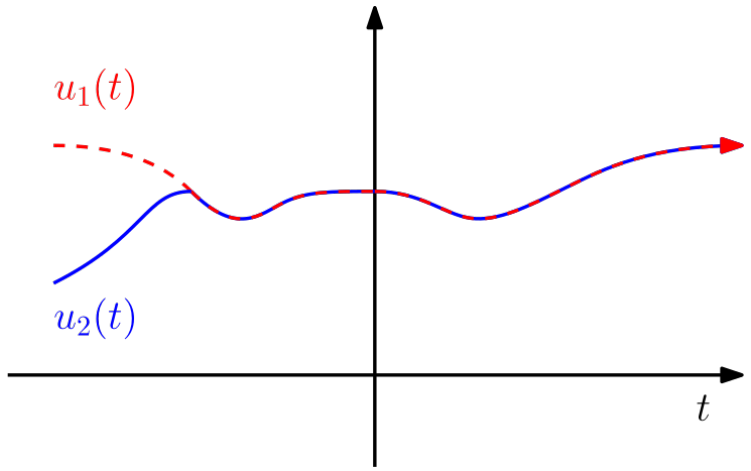


Fig. 2.7 The signals u_1 and u_2 share the same one-sided Laplace transform.

Above, we defined the *region of absolute convergence*, Ω, to be the region in which the integral defining the transform converges absolutely. Which signals have a nonempty region of absolute convergence? Do there exist signals with no region of convergence? To determine the class of signals with a well-defined region of convergence, let's take another look at the definition of the Laplace transform. We have,

$$\mathcal{L}(f)(s) = \int_{0^-}^{\infty} e^{-st} f(t) dt. \tag{2.343}$$

In order for this integral to converge absolutely for some values of s, it seems appropriate to request that f be bounded by some exponential growth. This leads us to the following definition.

Definition 2.41 (Function of Exponential Order) A function $f \in PC(\mathbb{R}_{\geq 0}, \mathbb{R}^{n \times m})$ is of exponential order α if there exists an $M \in \mathbb{R}_{\geq 0}$ for which $\|f(t)\| \leq Me^{\alpha t}$, for all $t \in \mathbb{R}_{\geq 0}$.

Remark 2.55 Notice that the condition $\|f(t)\| \leq Me^{\alpha t}$ is specified in terms of the values of the signal f, which are members of a finite-dimensional vector space. Due to norm equivalence in finite dimensional vector spaces, the choice of norm $\|\cdot\|$ on $\mathbb{R}^{n \times m}$ doesn't matter. A different choice of norm will simply result in a scaling of M by a positive constant.

Now, we show that all functions of exponential order have well-defined Laplace transforms on a nonempty region of convergence.

Proposition 2.26 (Laplace Transform of a Function of Exponential Order) *Let $f \in PC(\mathbb{R}_{\geq 0}, \mathbb{R}^{n\times m})$ be a function of exponential order α. Then, the region of absolute convergence of the Laplace transform of f contains the set*

$$\{s \in \mathbb{C} : \mathrm{Re}(s) > \alpha\} \subseteq \mathbb{C}. \tag{2.344}$$

Proof Suppose f satisfies $\|f(t)\| \leq Me^{\alpha t}$ for all $t \geq 0$. Then, one has

$$\int_{0^-}^{\infty} |e^{-st}| \, \|f(t)\| \, dt \leq M \int_{0^-}^{\infty} e^{(\alpha - \mathrm{Re}(s))t} dt. \tag{2.345}$$

This integral will converge when $\alpha - \mathrm{Re}(s) < 0$, or when $\alpha < \mathrm{Re}(s)$. Thus, for all s satisfying $\alpha < \mathrm{Re}(s)$, the Laplace transform will converge absolutely. This implies the given set is contained in the region of absolute convergence of the Laplace transform of f. □

As an illustration of what can go wrong if we don't enforce the exponential order assumption, we consider a signal whose Laplace transform has an empty region of convergence.

Example 2.4 To choose a signal with no Laplace transform, we choose a signal that grows faster than any exponential. For instance, consider the signal, $f(t) = e^{e^t}$, which grows faster than an exponential since e^t grows faster than t. Such a signal has no Laplace transform.

Now that we've defined the Laplace transform and a basic class of signals with a well-defined transform, we'll get some practice computing Laplace transforms.

Theorem 2.17 (Common One-Sided Laplace Transforms) *Consider the following collection of signals, transforms, and regions of absolute convergence of their transforms.*

Signal Name	*Signal*	*Laplace Transform*	*Region of Convergence*
Dirac Delta	$\delta(t)$	1	$\mathbb{C}$
Unit Step	$\mathbb{1}(t) = \begin{cases} 1 & t \geq 0 \\ 0 & t < 0 \end{cases}$	$\frac{1}{s}$	$\mathrm{Re}(s) > 0$
Unit Ramp	t	$\frac{1}{s^2}$	$\mathrm{Re}(s) > 0$
Polynomial	$t^n,\ n \in \mathbb{N}$	$\frac{n!}{s^{n+1}}$	$\mathrm{Re}(s) > 0$
Exponential	e^{at}	$\frac{1}{s-a}$	$\mathrm{Re}(s) > \mathrm{Re}(a)$
Sine	$\sin(\omega t)$	$\frac{\omega}{s^2+\omega^2}$	$\mathrm{Re}(s) > 0$
Cosine	$\cos(\omega t)$	$\frac{s}{s^2+\omega^2}$	$\mathrm{Re}(s) > 0$
Damped Sine	$e^{at}\sin(\omega t)$	$\frac{\omega}{(s-a)^2+\omega^2}$	$\mathrm{Re}(s) > \mathrm{Re}(a)$
Damped Cosine	$e^{at}\cos(\omega t)$	$\frac{s-a}{(s-a)^2+\omega^2}$	$\mathrm{Re}(s) > \mathrm{Re}(a)$

Shortly, we'll see that the Laplace transform behaves well under algebraic combinations of signals—this makes it possible to derive the transforms of more complex signals from the table above. In order to illustrate the general technique of computing a Laplace transform, we'll derive the transform of the Dirac delta and the unit step functions, arguably the two most important transforms signals. The rest of the table entries are found by grinding out integrals.

Example 2.5 (Laplace Transform of the Dirac Delta) Let's work out the Laplace transform of the Dirac delta function. Using our informal integral "rules," we have

$$\mathcal{L}(\delta)(s) = \int_{0^-}^{\infty} e^{-st}\delta(t)dt = \int_{-\infty}^{\infty} e^{s\cdot 0 - st}\delta(t)dt = e^{st}|_{t=0} = 1, \tag{2.346}$$

where we conclude that the lower bound can be extended to $-\infty$, since the Dirac delta "function of time " vanishes for $t < 0$. This derivation is made rigorous using approximations to the identity (see the previous, starred subsection for the details).

Example 2.6 (Laplace Transform of the Unit Step) Next, we calculate the transform of the unit step signal, $\mathbb{1}(\cdot)$. We have,

$$\mathcal{L}(\mathbb{1})(s) = \int_0^{\infty} e^{-st}dt = \Big[-\frac{1}{s}e^{-st}\Big]_0^{\infty} = \frac{1}{s} \text{ (for } \mathrm{Re}(s) > 0). \tag{2.347}$$

Exercise 2.24 Complete the rest of the Laplace transform table in Theorem 2.17. See [18] (or any other standard text on signals and systems) for a solution.

Let's examine the table of Laplace transforms and their regions of convergence in a more critical light. Above, we showed that the Laplace transform of the unit step function is,

$$\frac{1}{s} \text{ with R.O.C. } \mathrm{Re}(s) > 0. \tag{2.348}$$

The function $1/s$ is defined *everywhere* in the complex plane but $s = 0$—yet, the region of convergence of $\mathcal{L}(\mathbb{1})(s)$ is only $\mathrm{Re}(s) > 0$. This seems a little bit contradictory! Let's look at another example. The transform of e^{at} is $1/(s-a)$—defined everywhere but $s = a$—yet the transform converges absolutely for $\mathrm{Re}(s) > a$.

It appears as if we *should* be able to define our transforms on much larger regions than the region of convergence of the integral. Can we extend the domain of definition of our transforms in some canonical way? An important concept from complex analysis, *analytic continuation*, lets us do exactly this.

Theorem 2.18 (Analytic Continuation) *Consider two complex functions, $f : \Omega \to \mathbb{C}$ and $g : \Omega' \to \mathbb{C}$ on nonempty open domains $\Omega \subseteq \Omega' \subseteq \mathbb{C}$. If f is analytic on Ω, g is analytic on Ω', and $f(s) = g(s)$ for all $s \in \Omega$, then g is the unique analytic function on Ω' for which $f = g|_{\Omega}$. In this case, g is called the analytic continuation of f to Ω'.*

Proof See Chapter 2, Corollary 4.9 of [35]. □

Let's think about the implications of analytic continuation for the transforms we discussed above. Consider, for example, the transform of $\mathbb{1}(t)$, $\mathbb{1}(s) = 1/s$, which has region of convergence $\Omega = \mathrm{Re}(s) > 0$ and is analytic (complex-differentiable) on Ω. We know that $1/s$ can be defined on the open set $\Omega' = \mathbb{C} \setminus \{0\}$, and is also analytic on Ω'. Thus, by Theorem

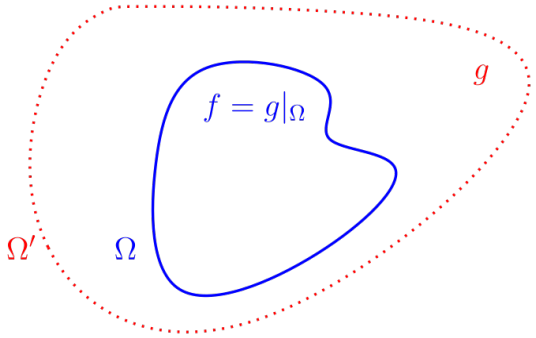


Fig. 2.8 If two analytic functions agree on a small domain, then there is only *one* way to extend the analytic function on the smaller domain to the larger domain. Here, this means that g is the *unique* analytic continuation of f to Ω'—the only possible analytic function on Ω' that matches f on Ω.

2.18, the *unique analytic continuation* of $1/s$ (on $\text{Re}(s) > 0$) to $\mathbb{C} \setminus \{0\}$ is given by $1/s$ on $\Omega' = \mathbb{C} \setminus \{0\}$. Thus, provided there exists an analytic continuation of the transform on a domain larger than the region of convergence, we can *unambiguously* extend the domain of the transform. This is because the analytic continuation is the *only possible* analytic extension of the original transform, so there is no ambiguity in "choosing between" different possible analytic extensions.

Due to the uniqueness of analytic continuation, when the extension of a transform is easy to compute, we can easily identify a transform with its analytic continuation to a larger domain. This lets us work with transforms such as $1/s$ on a much larger domain than just the region of convergence of the integral. To illustrate this point further, suppose we have a signal with transform,

$$F(s) = \frac{1}{(s-a)(s+b)}, \tag{2.349}$$

on some nonempty region of convergence Ω. We observe by inspection that we can analytically continue the transform from its region of convergence Ω to $\mathbb{C} \setminus \{a, -b\}$. Thus, we can unambiguously work with the transform on this larger domain. It's important to note—for general signals that have a Laplace transform, extensions are *not* as obvious as in the cases above. However, as a large class of signals of interest have simple transforms, this technique is something we can often apply.

Now that we've discussed some basic examples of Laplace transforms, we can study some of their important properties. First, we'll state the three *most fundamental* properties of the Laplace transform. Following this, we'll examine some more general-purpose properties of the transform.

Theorem 2.19 (Key Properties of the Laplace Transform) *Let $f, g \in PC(\mathbb{R}_{\geq 0}, \mathbb{R})$ be signals of exponential orders α and β, respectively.*

1. *Analyticity: For any $\epsilon > 0$, at all s for which $\text{Re}(s) > \alpha + \epsilon$, $\mathcal{L}(f)(s)$ is analytic.*
2. *Linearity: $\mathcal{L}(k_1 f + k_2 g)(s) = k_1 \mathcal{L}(f)(s) + k_2 \mathcal{L}(g)(s)$ $\forall k_1, k_2 \in \mathbb{R}$, $s : \text{Re}(s) > \max\{\alpha, \beta\}$.*
3. *Convolution: $\mathcal{L}(f * g)(s) = \mathcal{L}(f)(s) \cdot \mathcal{L}(g)(s)$ for all s satisfying $\text{Re}(s) > \max\{\alpha, \beta\}$.*

Remark 2.56 Remember: if $\mathcal{L}(f)$ and $\mathcal{L}(g)$ have analytic continuations to larger domains, we can extend the domains on which these properties hold!

Remark 2.57 Each of these properties is easily extended to the case where f and g are vector or matrix-valued signals! Here, one simply needs to take care to ensure the dimensions of f and g match.

Proof (Sketch) We'll give a proof sketch of these three properties—for a formal proof, one should be more precise in dealing with the convergence of indefinite integrals. We start with property (1). Fix $\epsilon > 0$, and consider $\text{Re}(s) > \alpha + \epsilon$. Since exponential growth dominates polynomial growth, if follows that, if $|f(t)| \leq me^{\alpha t}\ \forall t \geq 0$, there exists an $m \geq 0$ for which $|tf(t)| \leq m'e^{(\alpha+\epsilon)t}\ \forall t \geq 0$ (we'll prove this formally in the next chapter). Then,

$$\frac{d}{ds}\int_{0^-}^{\infty} e^{-st}f(t)dt = \int_{0^-}^{\infty} \frac{d}{ds}e^{-st}f(t)dt \tag{2.350}$$

$$= \int_{0^-}^{\infty} -te^{-st}f(t)dt \tag{2.351}$$

$$= -\mathcal{L}(tf(t)), \tag{2.352}$$

where the integral converges for all s satisfying $\text{Re}(s) > \alpha + \epsilon$. Thus, the derivative of the Laplace transform exists for $\text{Re}(s) > \alpha + \epsilon$. We conclude the analyticity of the transform. Next, we prove the linearity property. We have, for $k_1, k_2 \in \mathbb{R}$, and $\text{Re}(s) > \max\{\alpha, \beta\}$

$$\mathcal{L}(k_1 f + k_2 g) = \int_{0^-}^{\infty} e^{-st}(k_1 f(t) + k_2 g(t))dt \tag{2.353}$$

$$= k_1 \int_{0^-}^{\infty} e^{-st}f(t)dt + k_2 \int_{0^-}^{\infty} e^{-st}g(t)dt \tag{2.354}$$

$$= k_1\mathcal{L}(f)(s) + k_2\mathcal{L}(g)(s), \tag{2.355}$$

where we use that each integral converges absolutely for $\text{Re}(s) > \max\{\alpha, \beta\}$. This shows the linearity property. Finally, we show the convolution property. We have,

$$\mathcal{L}(f * g)(s) = \int_{0^-}^{\infty} e^{-st} \int_{0^-}^{\infty} f(t-\tau)g(\tau)d\tau dt \tag{2.356}$$

$$= \int_{0^-}^{\infty} \int_{0^-}^{\infty} e^{-st}f(t-\tau)g(\tau)d\tau dt \tag{2.357}$$

$$= \int_{0^-}^{\infty} \int_{0^-}^{\infty} e^{-s(t-\tau)}e^{-s\tau}f(t-\tau)g(\tau)d\tau dt \tag{2.358}$$

$$= \int_{0^-}^{\infty} \int_{0^-}^{\infty} e^{-s(t-\tau)}f(t-\tau)e^{-s\tau}g(\tau)d\tau dt \tag{2.359}$$

$$= \int_{0^-}^{\infty} e^{-s\tau}g(\tau) \int_{0^-}^{\infty} e^{-s(t-\tau)}f(t-\tau)dtd\tau \tag{2.360}$$

$$= \int_{0^-}^{\infty} e^{-s\tau}g(\tau) \int_{\tau}^{\infty} e^{-s(t-\tau)}f(t-\tau)dtd\tau, \tag{2.361}$$

where in the last step, we used that the signal f is zero for $t < \tau$. We recognize the inner integral as the Laplace transform of f, which converges for $\text{Re}(s) > \max\{\alpha, \beta\}$. Thus,

$$= \int_{0^-}^{\infty} e^{-s\tau} g(\tau)\mathcal{L}(f)(s)d\tau \tag{2.362}$$

$$= \mathcal{L}(f)(s) \int_{0^-}^{\infty} e^{-s\tau} g(\tau)d\tau \tag{2.363}$$

$$= \mathcal{L}(f)(s) \cdot \mathcal{L}(g)(s), \tag{2.364}$$

where the final transform also converges for $\text{Re}(s) > \max\{\alpha, \beta\}$. We conclude that one-sided convolution becomes multiplication under the one-sided Laplace transform! □

Exercise 2.25 Restate Theorem 2.19 in the case where f and g are matrix-valued signals.

This result highlights some of the principal reasons why the Laplace transform will be so useful for us. First, the Laplace transform is *linear* in signals—this means that, for any of the simple signals we stated transforms for, we *immediately* know the transforms of their linear combinations. Secondly, we found that the Laplace transform of the convolution of two signals becomes the product of the Laplace transforms of the signals. Thus, the Laplace transform fulfills our wish of simplifying the convolution operation. This is key for studying the response of linear, time-invariant systems, which is governed by convolution.

Now that we've stated these three fundamental properties, we summarize a number of additional properties concerning the interaction of the Laplace transform with integrals, derivatives, and delays.

Theorem 2.20 (Further Properties of the Laplace Transform) *Let $f : \mathbb{R} \to \mathbb{R}$ be a signal of exponential order α on $\mathbb{R}_{\geq 0}$ with one-sided Laplace transform $F(s)$ on Ω. Then, the following properties hold on a nonempty subset of Ω.*

Operation	*Signal*	*Laplace Transform*
Derivative	$f'(t)$	$sF(s) - f(0^-)$
n'th Derivative	$f^{(n)}(t)$	$s^n F(s) - s^{n-1} f(0^-) - ... - f^{(n-1)}(0^-)$
Integral	$\int_{0^-}^{t^+} f(\tau)d\tau$	$\frac{F(s)}{s}$
Product with t	$tf(t)$	$-F'(s)$
Product with t^n, $n \in \mathbb{N}$	t^n	$(-1)^n F^{(n)}(s)$
Delay by τ	$f(t-\tau)$	$e^{-\tau s} F(s)$
Time scaling by $a \neq 0$	$f(at)$	$\frac{1}{\lvert a \rvert} F(s/a)$

Exercise 2.26 Prove Theorem 2.20 via direct calculation. See [18] (or any other standard text on signals and systems) for a solution.

As a nice application of Theorem 2.20, one can confirm some of the basic Laplace transforms stated earlier using the various operations above. For instance, the ramp function is the integral of the step function, which is consistent with the ramp function having transform $1/s^2$ and the step function having transform $1/s$.

2.4.4.3 Transfer Functions

Now that we've studied the basic definition and properties of the Laplace transform, we're ready to apply it to the input/output analysis of continuous-time, LTI systems. Earlier, we showed that for an input signal $u(\cdot) \in PC(\mathbb{R}_{\geq 0}, \mathbb{R}^m)$, the zero-state response of a continuous-time LTI system, $y(t)$, is computed

$$y(t) = (H * u)(t), \tag{2.365}$$

at every $t \in \mathbb{R}$ at which u is continuous. Above, we showed that one-sided convolution becomes multiplication under the Laplace transform. Therefore, if we wish to understand the zero-state response of a system using the Laplace transform, we can apply the rule,

$$\mathcal{L}(y)(s) = \mathcal{L}(H * u)(s) = \mathcal{L}(H)(s)\mathcal{L}(u)(s). \tag{2.366}$$

Since convolution becomes multiplication in the frequency domain, *all we need* to compute the transform of the zero-state response is the product of the transform of the impulse response and the transform of the input signal. Thus, in the frequency domain, multiplication by the transform of the impulse response directly transfers us from input to output. This yields the following definition.

Definition 2.42 (Transfer Function) Consider a continuous-time LTI system with LTI impulse response map $H(\cdot) : \mathbb{R}_{\geq 0} \to \mathbb{R}^{p \times m}$. The transfer function of the system is the map $\hat{H} : \Omega \subseteq \mathbb{C} \to \mathbb{C}^{p \times m}$, defined

$$\hat{H}(s) = \mathcal{L}(H)(s), \ \forall s \in \Omega, \tag{2.367}$$

where Ω is the region of convergence of the transform.

Remark 2.58 Typically, one uses either $\hat{H}$ or $\hat{G}$ to refer to the transfer function of a system. We'll interchange between the two.

Remark 2.59 Remember, $\hat{H}$ can be extended to a larger domain than Ω if it has an analytic continuation to a larger domain! We'll see a few examples of this shortly.

Remark 2.60 Since $\hat{H}(s) \in \mathbb{C}^{p \times m}$ for all $s \in \Omega$, it follows that a MIMO system will have a matrix-valued transfer function, while a SISO system will have a scalar-valued transfer function. Since one has $Y(s) = \hat{H}(s)U(s)$, we can calculate the transfer function via $Y(s)/U(s) = \hat{H}(s)$ in the SISO case.

A few questions immediately appear upon making this definition. Does the Laplace transform of an impulse response always have a nonempty region of convergence? How does one most easily compute the Laplace transform of $H(\cdot)$? Let's start with the first question. Above, we showed that every map of exponential order has a nonempty region of convergence. Is the impulse response map of an LTI system a map of exponential order? Recall that the LTI impulse response map is defined,

$$H(t) = \begin{cases} C\exp(At)B + D\delta & t \geq 0 \\ 0 & t < 0. \end{cases} \tag{2.368}$$

By linearity of the Laplace transform it's sufficient for $C\exp(At)B$ and $D\delta(t)$ to have well-defined transforms in order for $H(\cdot)$ to have a well-defined transform. We know that $D\delta$ has a transform $\mathcal{L}(D\delta)(s) = D$, since $\mathcal{L}(\delta)(s) = 1$ for all $s \in \mathbb{C}$. What about $C\exp(At)B$? It certainly *seems* that the matrix exponential should be of exponential order! Fortunately, this turns out to be true—we'll accept this as a fact for now and will return to the proof next chapter, in our study of stability.

Fact For any $A \in \mathbb{R}^{n\times n}$, the matrix exponential $\exp(At)$ is of exponential order. □

Since $\exp(At)$ is of exponential order, we conclude that $C\exp(At)B$ is also of exponential order. The LTI impulse response map $H(\cdot)$ will therefore have a well-defined Laplace transform on a nonempty region of convergence.

Now, we focus on to the second question: how do we actually *compute* the transfer function of a given LTI system? One option is to compute the transform directly from the definition of $H(t)$, posed in terms of the matrix exponential. However, since the whole point of the Laplace transform is to avoid dealing with things like the matrix exponential, it would hardly be a great success if we needed to compute a Jordan form, change coordinates, and compute the transform of each entry. We'll show that there is a *much simpler* way to compute the transfer function. What's more, this simpler method will give us a way to compute the matrix exponential *without* appealing to the Jordan form. First, consider the following lemma.

Lemma 2.14 (Characterizing the Transfer Function) *Consider a continuous-time, LTI system representation (A, B, C, D) with LTI impulse response map H. A function $\hat{H} : \Omega \to \mathbb{C}^{p\times m}$ is the transfer function of the system if and only if $\hat{Y}(s) = \hat{H}(s)\hat{U}(s)$ for all transforms $\hat{U}(s)$ and $\hat{Y}(s)$ of admissible input signals $u(\cdot)$ and their corresponding zero-state responses $y(\cdot)$.*

Proof First, suppose $\hat{H}$ is the transfer function of the system. Then, by the convolution property, $\hat{Y}(s) = \hat{H}(s)\hat{U}(s)$. Now, we show the other direction. Suppose a function $\hat{H}$ satisfies $\hat{Y}(s) = \hat{H}(s)\hat{U}(s)$ for all admissible inputs and their corresponding zero-state outputs. Consider the input signal $u = \delta e_j$, where $e_j \in \mathbb{R}^m$ is the j'th standard basis vector of $\mathbb{R}^m$. Then, $\hat{Y}(s) = \hat{H}(s)e_j$. By definition of the impulse response map, however, the j'th column of $\mathcal{L}(H)(s)$ must also be $\hat{H}(s)e_j$. We conclude that $\hat{H}$ must be the transfer function of the system. □

With this lemma in mind, we show how the transfer function of a continuous-time, LTI system is easily computed.

Proposition 2.27 (Transfer Function of an LTI System Representation) *Consider a continuous-time, LTI system representation (A, B, C, D). The transfer function $\hat{H}$ of the system is computed,*

$$\hat{H}(s) = C(sI - A)^{-1}B + D, \ \forall s \in \Omega, \tag{2.369}$$

where $\Omega = \mathbb{C} \setminus \operatorname{spec}(A)$, the complex plane minus the eigenvalues of A.

Proof Recall that the system representation (A, B, C, D) satisfies,

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{2.370}$$
$$y(t) = Cx(t) + Du(t). \tag{2.371}$$

Let's take the Laplace transform of both of these equations for zero initial condition, $x(0) = 0$, and admissible input u. Computing the transforms, one has,

$$s\hat{X}(s) = A\hat{X}(s) + B\hat{U}(s) \tag{2.372}$$

$$\hat{Y}(s) = C\hat{X}(s) + D\hat{U}(s). \tag{2.373}$$

Since $(sI - A)$ is nonsingular for $s \notin \text{spec}(A)$, for $s \in \mathbb{C} \setminus \text{spec}(A)$, $\hat{X}$ must satisfy,

$$\hat{X}(s) = (sI - A)^{-1}B\hat{U}(s). \tag{2.374}$$

Substituting into the formula for Y, it follows that,

$$\hat{Y}(s) = (C(sI - A)^{-1}B + D)\hat{U}(s). \tag{2.375}$$

We conclude that,

$$\hat{H}(s) = C(sI - A)^{-1}B + D. \tag{2.376}$$

The function $(sI - A)^{-1}$ is defined (and analytic) on $\mathbb{C} \setminus \text{spec}(A)$. Identifying the transfer function of the system with its analytic continuation, the result follows. □

This gives us an easy way to compute the transfer function of any continuous-time, LTI system representation—all we need to do is invert a single matrix function of s (computers are quite good at this for small matrices) and perform some matrix multiplication. Now that we have a nice way to compute transfer functions, we can use them to solve for the zero-state response of a system in the time domain. In order to convert from the frequency domain to the time domain, we define the inverse Laplace transform.

Definition 2.43 (Inverse Laplace Transform) The inverse Laplace transform is a map $\mathcal{L}^{-1}$ taking a complex-valued function $\hat{H} : \Omega \to \mathbb{C}^{p\times m}$ to a real-valued signal $H(\cdot) : \mathbb{R}_{\geq 0} \to \mathbb{R}^{p\times m}$ satisfying $\mathcal{L}(H)(s) = \hat{H}(s)$ for all $s \in \Omega$.

Remark 2.61 There are a few subtleties regarding the definition of the inverse Laplace transform. First, the inverse Laplace transform is *not* guaranteed to exist for any complex-valued function! Second, if the inverse Laplace transform *does* exist, it is not guaranteed to be unique. For a general complex-valued function, the best one can hope for is a unique inverse Laplace transform up to equality outside of sets of measure zero—sets which are ignored by integrals. This is a consequence of the Laplace transform being an integral transform. If one takes signals $f, g \in PC(\mathbb{R}_{\geq 0}, \mathbb{R}^{p\times m})$ which are equal outside of a set of measure zero, the signals will have the same Laplace transform. Therefore, the inverse Laplace transform of $\mathcal{L}(f) = \mathcal{L}(g)$ can at best be uniquely defined up to equality outside of sets of measure zero.

The existence of an inverse Laplace transform is generally challenging to establish for an arbitrary, complex-valued function. However, for a large class of functions, a well-defined, *unique* inverse does exist.

Lemma 2.15 (Inverse Laplace Transform of a Strictly Proper Rational Function) *Consider a complex function which is the ratio of two polynomials with real coefficients,*

$$\hat{G}(s) = \frac{a_m s^m + ... + a_1 s + a_0}{b_n s^n + ... + b_1 s + b_0},\ a_i, b_i \in \mathbb{R},\ a_m, b_n \neq 0. \tag{2.377}$$

If $m < n$, then $\hat{G}$ is called a strictly proper rational function. If $m < n$, then $\hat{G}$ has a unique inverse Laplace transform, up to equality outside of sets of measure zero.

Remark 2.62 Loosely speaking, time-domain signals comprised of algebraic combinations of polynomials, trigonometric functions, and exponential functions of time will have transforms of the form above. For such functions, inverse Laplace transforms are uniquely defined. We refer the interested reader to [19], Chapter 15.3, for the details.

Remark 2.63 Recalling that the Laplace transform of a matrix-valued signal is defined *element-wise*, we can extend this result to the case where each element of a matrix-valued function is a strictly proper rational function.

Proof (Sketch) We'll provide a sketch of the existence proof in the case where the roots of the numerator and denominator polynomials are real and non-repeated. For the general case and the details of uniqueness, we refer the reader to [19]. If the numerator and denominator polynomials of $\hat{G}$ have real, non-repeated roots, $\hat{G}$ can be factored,

$$\hat{G}(s) = \frac{(s - s_{a,1}) \cdot ... \cdot (s - s_{a,m})}{(s - s_{b,1}) \cdot ... \cdot (s - s_{b,n})}. \tag{2.378}$$

By partial fraction expansion, there exist constants $c_1, ..., c_n \in \mathbb{R}$ for which,

$$\hat{G}(s) = \frac{c_1}{s - s_{b,1}} + ... + \frac{c_n}{s - s_{b,n}}. \tag{2.379}$$

We recognize each term as the Laplace transform of an exponential signal, $e^{s_{b,i}t}$. Applying linearity of the Laplace transform, we conclude $\hat{G}$ has an inverse Laplace transform. □

Thus, for the class of strictly proper, rational functions, we can always find an inverse Laplace transform. We'll provide an example of this shortly. First, we show how to apply the inverse Laplace transform to compute the zero-state response of an LTI system.

Proposition 2.28 (Laplace Function Solution of I/O Systems) *Consider a continuous-time, LTI system with transfer function $\hat{H} : \Omega \to \mathbb{C}^{p \times m}$. The zero-state response of the system to an input signal $u(\cdot) \in PC(\mathbb{R}_{\geq 0}, \mathbb{R}^m)$ of exponential order is computed,*

$$y(t) = \mathcal{L}^{-1}(\hat{H}(s)U(s))(t), \tag{2.380}$$

for all $t \in \mathbb{R}_{\geq 0}$ at which u is continuous, provided the inverse Laplace transform exists.

Proof We know that $y(t) = (H * u)(t)$ for all $t \geq 0$ at which u is continuous. Taking the Laplace transform, one then has $Y(s) = \hat{H}(s)U(s)$, which implies $y(t) = \mathcal{L}^{-1}(\hat{H}(s)U(s))$.□

Let's get some practice computing both the transfer function and the zero-state response of a continuous-time LTI system using the Laplace transform method.

Example 2.7 (Step Response of a SISO System) Consider the SISO, LTI system,

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ -1 & -2 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t) \tag{2.381}$$

$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} x(t) \tag{2.382}$$

Let's compute the *step response* the system—the zero-state response of the system to a step function input. First, let's identify the transfer function of the system. There are a couple ways of doing this. First, we can use the formula $\hat{H}(s) = C(sI - A)^{-1}B + D$. This leaves us with,

$$\hat{H}(s) = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} s & -1 \\ 1 & s+2 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \tag{2.383}$$

which, since the system is small, we can compute using a symbolic calculator. Second, we can directly solve for the function $\hat{H}$ which satisfies $\hat{Y}(s) = \hat{H}(s)\hat{U}(s)$. In order to do this, we examine the components of the state equation. We have,

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ -x_1 - 2x_2 + u \end{bmatrix}. \tag{2.384}$$

Let's take the transform of each of component with $x_i(0^-) = 0$. We get,

$$s\hat{X}_1(s) = \hat{X}_2(s) \tag{2.385}$$

$$s\hat{X}_2(s) = -\hat{X}_1(s) - 2\hat{X}_2(s) + \hat{U}(s). \tag{2.386}$$

Computing the output, we find $\hat{Y}(s) = \hat{X}_1(s)$. Therefore, to get $\hat{Y}(s) = \hat{H}(s)\hat{U}(s)$, we solve,

$$s^2\hat{X}_1(s) = -\hat{X}_1(s) - 2s\hat{X}_1(s) + \hat{U}(s) \tag{2.387}$$

$$(s^2 + 2s + 1)\hat{X}_1(s) = \hat{U}(s) \tag{2.388}$$

$$\hat{X}_1(s) = \frac{1}{s^2 + 2s + 1}\hat{U}(s). \tag{2.389}$$

Plugging into the output, it follows that

$$\hat{Y}(s) = \frac{1}{s^2 + 2s + 1}\hat{U}(s) := \hat{H}(s)\hat{U}(s). \tag{2.390}$$

This formula for $\hat{H}$ is identical to that which we would've found from the matrix inversion method. Note that the form of the state equation & output equations given here are particularly amenable to the direct solution method—later, we'll outline a general class of system representations for which this method is readily applicable.

Now that we've identified the transfer function of the system, we can compute its step response. We want to find the inverse Laplace transform of,

$$\hat{Y}(s) = \hat{H}(s)\hat{\mathbb{1}}(s) = \frac{1}{s^2 + 2s + 1}\frac{1}{s}. \tag{2.391}$$

A nice method to do this "by inspection" is to break up this transform into pieces composed of the common Laplace transforms we identified earlier. We'll accomplish this using partial fraction expansion[12], which splits up the product of functions into a series of simpler fractions. With a little bit of algebra, we decompose $\hat{H}(s)\hat{\mathbb{1}}(s)$ into,

[12] If you haven't seen this technique before, consult an introductory feedback control textbook such as [3] for the details on how to perform a partial fractions expansion by hand. Symbolic calculators such as Matlab symbolic have a function `partfrac` which will do this for you.

$$\frac{1}{s^2+2s+1}\frac{1}{s} = \frac{1}{s} - \frac{1}{(s+1)^2} - \frac{1}{s+1}. \tag{2.392}$$

Now, we apply linearity of the Laplace transform and find the inverse transform of the entire expression by finding the inverse transform of each piece. Using the table of transforms and transform rules we derived earlier, we find,

$$\mathcal{L}^{-1}\Big(\frac{1}{s}\Big) = \mathbb{1}(t),\ \mathcal{L}^{-1}\Big(\frac{1}{(s+1)^2}\Big) = te^{-t},\ \mathcal{L}^{-1}\Big(\frac{1}{s+1}\Big) = e^{-t}. \tag{2.393}$$

We conclude that the step response of the system is,

$$y(t) = \mathbb{1}(t) - te^{-t} - e^{-t},\ t \geq 0. \tag{2.394}$$

From start to finish, *all we needed* was a little bit of algebra—no fancy matrix exponentials or Jordan forms required!

In the example above, we found that the form of the (A, B, C, D) matrices allowed for a particularly simple solution of the transfer function. We now outline the general case of this simple structure.

Proposition 2.29 (Representation with a Simple Transfer Function) *Consider a SISO system representation, (A, B, C, D), in which*

$$A = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & 1 \\ -a_0 & -a_1 & \dots & -a_{n-2} & -a_{n-1} \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}$$
$$C = \begin{bmatrix} c_0 & c_1 & \dots & c_{n-2} & c_{n-1} \end{bmatrix} \qquad D = 0. \tag{2.395}$$

For such a system representation, the transfer function is computed,

$$\hat{H}(s) = \frac{c_{n-1}s^{n-1} + c_{n-2}s^{n-2} + \ldots + c_1 s + c_0}{s^n + a_{n-1}s^{n-1} + \ldots + a_1 s + a_0}. \tag{2.396}$$

Proof See Problem 2.24. □

Later in the course, we'll discuss a more general version of Proposition 2.29 which lets us compute the transfer function of any SISO system representation using something called the *Markov parameters*. For now, however, this simple case will suffice.

Earlier, we mentioned that Proposition 2.28 enables one to compute the matrix exponential using an inverse Laplace transform. In the following result, we see exactly how to do this. What's more, we'll find that this method is *entirely independent* from the Jordan canonical form!

Proposition 2.30 (Matrix Exponential via Inverse Laplace Transform) *Consider a matrix $A \in \mathbb{R}^{n\times n}$. The exponential of At is computed via the inverse Laplace transform as*

$$\exp(At) = \Big[\mathcal{L}^{-1}[(sI - A)^{-1}]\Big](t),\ \forall t \geq 0. \tag{2.397}$$

Proof Consider the initial value problem,

$$\dot{X}(t) = AX(t),\ X(0) = I, \tag{2.398}$$

the solution of which is the matrix exponential $\exp(At)$. Taking the Laplace transform,

$$s\hat{X}(s) - X(0) = A\hat{X}(s) \tag{2.399}$$

$$(sI - A)\hat{X}(s) = I \tag{2.400}$$

$$\hat{X}(s) = (sI - A)^{-1}. \tag{2.401}$$

Thus, the matrix exponential is the inverse Laplace transform of $(sI - A)^{-1}$. □

Exercise 2.27 Compute the matrix exponential $\exp(At)$ for the matrix

$$A = \begin{bmatrix} 0 & 1 \\ -1 & -2 \end{bmatrix} \tag{2.402}$$

using the inverse Laplace transform method.

Let's quickly summarize what we've found about Laplace transforms and compare it with our "Laplace transform wish list" that we came up with at the start of this section. So far, we've checked off the following two items:

1. Differential Equations Become Algebraic: we've shown that we can analyze the I/O response of state space equations using algebraic expressions that result from the Laplace transform. Additionally, we showed how to compute the matrix exponential using an entirely algebraic method.
2. Simplify Convolutions: we showed that, under the Laplace transform, convolution simply becomes *multiplication* of transforms. This is a significant simplification of the convolution operation.

Currently, we have two items remaining on our wish list: *analyze long-term behavior* and *simple interpretations*. To fully answer these questions, we'll need to study stability theory (which we'll do in the next chapter) and realization theory (which we'll do in the chapter following that). In particular, we'll find that the transfer function encodes valuable information about the I/O stability of an LTI system, and that relationship between the numerator and denominator polynomials of a transfer function is wrapped up in the concepts of controllability and observability.

2.4.5 The $\mathcal{Z}$-Transform

Now, we develop an analogous transform for discrete-time systems. The $\mathcal{Z}$-transform is the discrete-time analogue of the Laplace transform, and is defined to shares many of its key properties with the Laplace transform. As such, much of the theory of the $\mathcal{Z}$-transform will immediately feel familiar to us.

The main insight into defining the $\mathcal{Z}$-transform is the following: the discrete-time analogue of an exponential signal, e^{at}, is the *geometric* signal, a^k. In fact, the geometric signal a^k simply comes from sampling an exponential signal e^{ct} at integer time steps, $t = k\Delta$. When

we combine the translation from exponential to geometric with the standard translation from integral to sum, we find a natural, candidate definition for the $\mathcal{Z}$-transform that mirrors that of the Laplace transform.

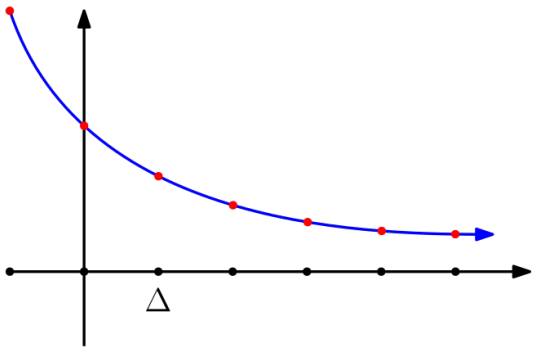


Fig. 2.9 Sampling an exponential signal at a fixed sampling time step Δ produces a geometric signal.

Definition 2.44 (One-Sided $\mathcal{Z}$-Transform) Consider a discrete-time, matrix-valued signal, $f[\cdot] : \mathbb{Z}_{\geq 0} \to \mathbb{R}^{n\times m}$. The one-sided $\mathcal{Z}$-transform of $f[\cdot]$ is the complex-valued function, $\hat{F} : \Omega \to \mathbb{C}^{n\times m}$, defined

$$\hat{F}(z) = \mathcal{Z}[f](z) := \sum_{k=0}^{\infty} z^{-k} f[k], \tag{2.403}$$

where $\Omega := \{z \in \mathbb{C} : \mathcal{Z}(f)(z) \text{ converges absolutely}\}$ is called the *region of absolute convergence* of the transform $\hat{F}$.

Remark 2.64 Just like the one-sided Laplace transform, the one-sided $\mathcal{Z}$-transform starts at 0 and goes to ∞ to reflect the causality of our systems (the discrete-time impulse response of a causal, LTI system is zero for $k < 0$). One can, of course, extend the definition of the $\mathcal{Z}$-transform to $-\infty$. Here, we'll stick with the one-sided definition—whenever we say "$\mathcal{Z}$-transform," we mean the one-sided $\mathcal{Z}$-transform.

Remark 2.65 As with the case of the Laplace transform, we use a capital letter with a hat to distinguish between the $\mathcal{Z}$-transform of a signal and the original signal. Above, for instance, we write the $\mathcal{Z}$-transform of $f[\cdot]$ as $\hat{F}$.

Take a moment to examine the formula for the $\mathcal{Z}$-transform, and convince yourself that *all we've done* is swap out the exponential signal e^{-st} for a geometric signal z^{-k} and an integral from 0^- to ∞ for a sum from 0 to ∞. In discrete time, of course, 0^- is no different from 0 (since we're working in $\mathbb{Z}$ and not $\mathbb{R}$), so our sum starts from 0.

Now that we've established a definition for the $\mathcal{Z}$-transform, let's try tracing a few more steps we took when defining the Laplace transform. Recall that, after defining the Laplace transform, we sought out a class of signals with a well-defined Laplace transform, which turned out to be the signals of *exponential order*—this followed from the presence of an exponential in the definition of the transform. In the definition of the $\mathcal{Z}$-transform, instead of an exponential, we have a geometric growth term, z^{-k}. The following definition is therefore a natural candidate for a class of signals with well-defined $\mathcal{Z}$-transforms.

Definition 2.45 (Sequence of Geometric Order) A sequence $f : \mathbb{Z}_{\geq 0} \to \mathbb{R}^{p\times m}$ is of geometric order $\alpha \geq 0$ if there exists an $M \geq 0$ such that, for all $k \in \mathbb{Z}_{\geq 0}$, $\|f[k]\| \leq M\alpha^k$.

Remark 2.66 Unlike in the continuous-time case, we request that the order α be nonnegative. If we don't make this restriction, α^k could be negative for odd k.

Exercise 2.28 Confirm that the choice of norm $\|\cdot\|$ does not affect the order α of a given sequence. Conclude that Definition 2.45 does not depend on the choice of norm.

With this definition in hand, we now state and prove a result concerning the convergence of the $\mathcal{Z}$-transform of sequences of geometric order.

Proposition 2.31 ($\mathcal{Z}$-Transform of a Sequence of Geometric Order) *Consider a sequence $f[\cdot] : \mathbb{Z}_{\geq 0} \to \mathbb{R}^{p\times m}$ of geometric order $\alpha \geq 0$. The region of absolute convergence of the $\mathcal{Z}$-transform of $f[\cdot]$ contains the set*

$$\{z \in \mathbb{C} : |z| > \alpha\} \subseteq \mathbb{C}. \tag{2.404}$$

Proof Suppose f is a sequence of geometric order $\alpha \geq 0$. Then, one has

$$\sum_{k=0}^{\infty} |z^{-k}| \, \|f[k]\| \leq M \sum_{k=0}^{\infty} |z^{-k}|\alpha^k = M \sum_{k=0}^{\infty} |z^{-1}\alpha|^k. \tag{2.405}$$

If $|z^{-1}\alpha| < 1$, this series will converge. This occurs for $|z^{-1}| \cdot |\alpha| < 1 \iff |z| > \alpha$. □

As opposed to the Laplace transform, where regions of convergence of functions of exponential order were of the form $\mathrm{Re}(s) > \alpha$, for $\mathcal{Z}$-transforms and sequences of geometric order, one has $|z| > \alpha$—that the $\mathcal{Z}$-transform converges outside a disk centered at the origin. Now that we've outlined this "nice" class of signals, we consider some common $\mathcal{Z}$-transforms.

Theorem 2.21 (Common One-Sided $\mathcal{Z}$-Transforms) *Consider the following collection of signals, transforms, and regions of absolute convergence of their transforms.*

Signal Name	*Signal*	*$\mathcal{Z}$-Transform*	*R.O.C.*
Unit Impulse	$\delta[k]$	1	$\mathbb{C}$
Unit Step	$\mathbb{1}[k] = \begin{cases} 1 & k \geq 0 \\ 0 & k < 0 \end{cases}$	$\frac{1}{1-z^{-1}}$	$\lvert z\rvert > 1$
Geometric	a^k	$\frac{1}{1-az^{-1}}$	$\lvert z\rvert > \lvert a\rvert$
	$\sin(\omega k)$	$\frac{\sin(\omega)z^{-1}}{1-2\cos(\omega)z^{-1}+z^{-2}}$	$\lvert z\rvert > 1$
	$\cos(\omega k)$	$\frac{1-\cos(\omega)z^{-1}}{1-2\cos(\omega)z^{-1}+z^{-2}}$	$\lvert z\rvert > 1$
	$a^k \sin(\omega k)$	$\frac{1-a\sin(\omega)z^{-1}}{1-2a\cos(\omega)z^{-1}+a^2z^{-2}}$	$\lvert z\rvert > \lvert a\rvert$
	$a^k \cos(\omega k)$	$\frac{1-a\cos(\omega)z^{-1}}{1-2a\cos(\omega)z^{-1}+a^2z^{-2}}$	$\lvert z\rvert > \lvert a\rvert$

Remark 2.67 Note that here, we assume all signals are zero for $k < 0$. One can multiply each signal by the unit step $\mathbb{1}[k]$ to make this more explicit.

Remark 2.68 Even though we've switched to discrete-time, the same remarks about analytic continuation still apply, as the transforms are still nothing more than complex functions. We can use analytic continuation to unambiguously extend the domain of each transform beyond the region of convergence of its corresponding infinite sum.

Thus, we observe that the set of common, one-sided $\mathcal{Z}$-transforms have regions of absolute convergence that are *punctured planes*—complex planes with a disk removed. As with the Laplace transform, we work out a couple of important examples of $\mathcal{Z}$-transforms.

Example 2.8 ($\mathcal{Z}$-Transform of a Unit Impulse) Since $\delta[k]$ is nonzero only for $k = 0$,

$$\mathcal{Z}(\delta)(z) = \sum_{k=0}^{\infty} z^{-k}\delta[k] = z^0 = 1. \tag{2.406}$$

It follows that $(\mathcal{Z}(\delta))(z) = 1$ for all $z \in \mathbb{C}$, and that the transform converges for all $z \in \mathbb{C}$.

Example 2.9 ($\mathcal{Z}$-Transform of a Unit Step) Now, we consider the unit step function, which is identically 1 for $k \geq 0$. We have,

$$\sum_{k=0}^{\infty} z^{-k}\mathbb{1}[k] = \sum_{k=0}^{\infty} z^{-k}, \tag{2.407}$$

which converges absolutely for $|z| > 1$. Recalling the formula for an infinite geometric series, we conclude that for $|z| > 1$,

$$(\mathcal{Z}(\mathbb{1}))(z) = \sum_{k=0}^{\infty}(z^{-1})^k = \frac{1}{1 - z^{-1}}. \tag{2.408}$$

Exercise 2.29 Complete the $\mathcal{Z}$-transform table in Theorem 2.21. See [18] (or any other standard book on signals and systems) for a solution.

Now, we state a theorem regarding the most important properties of the $\mathcal{Z}$-transform. Since the $\mathcal{Z}$-transform was constructed in a similar spirit to the Laplace transform, all of the same key properties hold! Due to similarities to the analogous Laplace transform results, we leave the proof of the following theorem as an exercise.

Theorem 2.22 (Key Properties of the $\mathcal{Z}$-Transform) *Let $f, g : \mathbb{Z}_{\geq 0} \to \mathbb{R}$ be sequences of geometric orders α and β, respectively.*

1. *Analyticity: at any $|z| > \alpha + \epsilon$, $\epsilon > 0$, $\mathcal{Z}(f)(z)$ is analytic.*
2. *Linearity: $\mathcal{Z}(k_1 f + k_2 g)(z) = k_1\mathcal{Z}(f)(z) + k_2\mathcal{Z}(g)(z)$ $\forall k_1, k_2 \in \mathbb{R}$, $z : |z| > \max\{\alpha, \beta\}$.*
3. *Convolution: $\mathcal{Z}(f * g)(z) = \mathcal{Z}(f)(z) \cdot \mathcal{Z}(g)(z)$ for all z satisfying $|z| > \max\{\alpha, \beta\}$.*

Exercise 2.30 Prove Theorem 2.22.

Thus, we observe that—as with the Laplace transform—the $\mathcal{Z}$-transform satisfies a number of desirable properties. First, the $\mathcal{Z}$-transform is linear—this means that we can easily take $\mathcal{Z}$-transforms of linear combinations of a few simple signals. Secondly, we observe that

convolution becomes *multiplication* under the $\mathcal{Z}$-transform. Recall that—in the case of the Laplace transform—this is the key property that enabled the use of transfer functions to study LTI systems. Since we have the same convolution property in the case of the $\mathcal{Z}$-transform, we'll find that the same definitions and properties of transfer functions hold for the $\mathcal{Z}$-transform. Before we turn our attention to this, however, we state some further important properties of the $\mathcal{Z}$-transform.

Theorem 2.23 (Further Properties of the $\mathcal{Z}$-Transform) *Let $f : \mathbb{Z}_{\geq 0} \to \mathbb{R}$ be a signal of geometric order α with $\mathcal{Z}$-transform $F(z)$ on Ω. Then, the following properties hold on a nonempty subset of Ω.*

Operation	*Signal*	$\mathcal{Z}$*-Transform*
Step Forward by 1	$f[k+1]$	$zF(z) - zf[0]$
Step Forward by $n > 0$	$f[k+n]$	$z^n F[z] - z^n f[0] - ... - zf[k-1]$
Delay by $n > 0$	$f[k-n]$	$z^{-n}F(z)$
Time Reversal	$f[-k]$	$F(z^{-1})$
Product with k	$kf[k]$	$-kF'(z)$
Scaling by a^k	$a^k f[k]$	$F(a^{-1}z)$

(2.409)

Exercise 2.31 Prove Theorem 2.23. See [18] (or any other standard book on signals and systems) for a solution.

2.4.5.1 Transfer Functions

We'll finish up our first pass at $\mathcal{Z}$-transforms by discussing the transfer functions of discrete-time, LTI systems. Due to the convolution property of the $\mathcal{Z}$-transform, we once again find that knowing the $\mathcal{Z}$-transform of the impulse response of an LTI system is sufficient to characterize its zero-state response to any input. As such, we define the *transfer function*, which entirely characterizes the system's response, as follows.

Definition 2.46 (Discrete-Time Transfer Function) Consider a discrete-time, LTI system with LTI impulse response map $H[\cdot] : \mathbb{Z} \to \mathbb{R}^{p\times m}$. The transfer function of the system is the map $\hat{H} : \Omega \subseteq \mathbb{C} \to \mathbb{C}^{p\times m}$,

$$\hat{H}(z) = \mathcal{Z}(H)(z), \ \forall z \in \Omega, \tag{2.410}$$

where Ω is the region of convergence of the transform.

As in the continuous-time case, the discrete-time transfer function is equivalently characterized by its interaction with input signals and the zero-state response.

Lemma 2.16 (Characterizing the Transfer Function) *Consider a discrete-time, LTI system representation (A, B, C, D) with LTI impulse response map $H[\cdot]$. A function $\hat{H} : \Omega \to \mathbb{C}^{p\times m}$ is the transfer function of the system if and only if $\hat{Y}(z) = \hat{H}(z)\hat{U}(z)$ for all transforms $\hat{U}(z)$ and $\hat{Y}(z)$ of admissible input signals $u[\cdot]$ and their corresponding zero-state responses $y[\cdot]$.*

Proof The proof of this result follows identically to the continuous-time case. If $\hat{H}$ is the transfer function of the system, then the convolution property of the $\mathcal{Z}$-transform necessitates $\hat{Y}(z) = \hat{H}(z)\hat{U}(z)$ for all pairs $u[\cdot], y[\cdot]$ of input and zero-state response. Now, suppose $\hat{H}$ satisfies the given property. Consider an input signal $u[\cdot] = \delta[\cdot]e_j$. Then, $\hat{Y}(z) = \hat{H}(z)e_j$. Since $\hat{Y}(z)$ must also equal the transform of column j of the impulse response map, it follows that $\hat{H}(z)$ must be the transform of the impulse response map. □

As in the continuous-time case, this result makes the computation of the transfer function of an LTI system quite simple.

Proposition 2.32 (Transfer Function of a DT-LTI System Representation) *Consider a discrete-time LTI system representation (A, B, C, D). The transfer function of the system is computed,*

$$\hat{H}(z) = C(zI - A)^{-1}B + D, \ \forall z \in \Omega, \tag{2.411}$$

where $\Omega = \mathbb{C} \setminus \text{spec}(A)$, the complex plane minus the eigenvalues of A.

Remark 2.69 When proving statements about the $\mathcal{Z}$-transform, it's useful to draw comparisons to analogous theorems about the Laplace transform. In the case above, for instance, the "step ahead by 1" operation in discrete-time is the analogue of the derivative in continuous-time. This is reflected in their transforms—step ahead by 1 corresponds to multiplication by z, whereas differentiation corresponds to multiplication by s. Observations like this often give away the entire proof structure of a property of the $\mathcal{Z}$-transform once the analogous Laplace transform proof has been completed.

Exercise 2.32 Prove Proposition 2.32, using the continuous-time case as a guide.

Once we've defined the transfer function of the system, we can determine the system response in the *time-domain* to an input signal using the inverse $\mathcal{Z}$-transform.

Definition 2.47 (Inverse $\mathcal{Z}$-Transform) The inverse $\mathcal{Z}$-transform is a map $\mathcal{Z}^{-1}$ taking a complex-valued function $\hat{H} : \Omega \to \mathbb{C}^{p\times m}$ to a real-valued sequence $H[\cdot] : \mathbb{Z}_{\geq 0} \to \mathbb{C}^{p\times m}$ satisfying $\mathcal{Z}(H)(z) = \hat{H}(z)$ for all $z \in \Omega$.

Lemma 2.17 (Inverse $\mathcal{Z}$-Transform of a Strictly Proper Rational Function) *Consider a complex function which is the ratio of two polynomials with real coefficients,*

$$\hat{G}(z) = \frac{a_m z^m + \ldots + a_1 z + a_0}{b_n z^n + \ldots + b_1 z + b_0}, \ a_i, b_i \in \mathbb{R}, \ a_m, b_n \neq 0. \tag{2.412}$$

If $m < n$, then $\hat{G}$ has a unique inverse $\mathcal{Z}$-transform.

Exercise 2.33 Provide a proof of Lemma 2.17 in the special case where the numerator and denominator polynomials each have distinct, real roots.

We can compute the response of a system to any input using knowledge of the system's transfer function and the input's $\mathcal{Z}$-transform.

Proposition 2.33 ($\mathcal{Z}$-Transform Solution of I/O Systems) *Consider a discrete-time, LTI system with transfer function $\hat{H} : \Omega \to \mathbb{C}^{p\times m}$. The zero-state response of the system to an input signal $u[\cdot] : \mathbb{Z}_{\geq 0} \to \mathbb{R}^m$ of geometric order is computed,*

$$y[k] = \mathcal{Z}^{-1}(\hat{H}(z)\hat{U}(z))[k],\ \forall k \in \mathbb{Z}_{\geq 0}, \tag{2.413}$$

provided the inverse $\mathcal{Z}$-transform exists.

Remark 2.70 Since continuity of the input signal is not an issue in discrete-time, we no longer need to append the condition "at every t at which u is continuous."

Exercise 2.34 Prove Proposition 2.33.

Just as we can compute a matrix exponential using the inverse Laplace transform, we can compute a matrix *power* using the inverse $\mathcal{Z}$-transform.

Proposition 2.34 (Matrix Power via Inverse $\mathcal{Z}$-Transform) *Consider a matrix $A \in \mathbb{R}^{n\times n}$. The exponent A^k, $k \in \mathbb{Z}_{\geq 0}$ is computed via the inverse $\mathcal{Z}$-transform as*

$$A^k = \left[\mathcal{Z}^{-1}[z(zI - A)^{-1}]\right][k],\ \forall k \in \mathbb{Z}_{\geq 0}. \tag{2.414}$$

Exercise 2.35 Prove Proposition 2.34. Remember to account for the initial condition!

Let's summarize what we've found about the $\mathcal{Z}$-transform. Across the board, we've found that the $\mathcal{Z}$-transform mirrors the Laplace transform in discrete-time. We found that the $\mathcal{Z}$-transform is linear, turns convolution into multiplication, and interacts well with standard operations such as time shifts and delays. Like the Laplace transform, we'll return to a deeper study of the $\mathcal{Z}$-transform later in the course.

2.4.6 Further Reading

The treatment of impulse response in discrete and continuous-time is based on [2] and [6]. The informal approach to the Dirac delta "function" as a function of time is based on the treatment given by [2]. For a more formal treatment of the Dirac delta distribution, we refer the reader to [19] for a nice, self-contained exposition. For more information on approximations to the identity, we recommend [34] and [36]. A nice book on complex analysis is [35]. For further information on the analytical properties of the Laplace and $\mathcal{Z}$-transforms, as well as on a rigorous construction of distributions within the context of control, the reader is encouraged to consult [19, 20, 21]. A more engineering-oriented introduction to the Laplace and $\mathcal{Z}$-transforms is provided in [18], [25], and [38].

2.4.7 Problems

Problem 2.21 (Algebraic Properties of Convolutions) In this problem, we'll examine the basic algebraic properties of convolutions. For both the continuous and discrete-time convolutions, confirm that the following properties hold:

1. Linearity: $(\alpha f_1 + \beta f_2) * g = \alpha f_1 * g + \beta f_2 * g$.
2. Commutativity: $f * g = g * f$.
3. Associativity: $f * (g * h) = (f * g) * h$.

You may assume that each convolution (e.g. $f * g$, $g * h$, etc.) is defined for all time.

Problem 2.22 (Transforms & Transition Matrices) The Laplace transform offers yet another way of computing the state transition matrix. For a continuous-time, LTI representation (A, B, C, D), the matrix exponential is computed $\exp(At) = \mathcal{L}^{-1}[(sI - A)^{-1}](t)$, for all $t \geq 0$. In this problem, we'll consider an analogue in discrete-time, and use both the continuous and discrete formulas to compute some transition matrices.

1. Show that the state transition matrix of a discrete-time, LTI representation (A, B, C, D) is computed,

$$A^k = \mathcal{Z}^{-1}[z(zI - A)^{-1}][k], \ \forall k \geq 0. \tag{2.415}$$

2. Using the transform formulas, compute the continuous and discrete-time transition matrices associated to the matrix,

$$A = \begin{bmatrix} 0 & 1 \\ -1 & -2 \end{bmatrix}. \tag{2.416}$$

   Comment on the benefits and drawbacks of this method of computing the transition matrix. You may use a symbolic calculator to compute the inverse of $(sI - A)$.

Problem 2.23 (Transfer Functions & Change of Basis) Consider a linear, time-invariant system representation (A, B, C, D). Recall that under a change of state coordinates, $z = Tx$, the representation *transforms* to $(TAT^{-1}, TB, CT^{-1}, D)$. Does the transfer function associated to the system representation change under a change of state coordinates? Provide a proof or counterexample to back up your answer.

Problem 2.24 (A Simple SISO Transfer Function) Consider a continuous-time SISO, LTI system representation (A, B, C, D),

$$A = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & 1 \\ -a_0 & -a_1 & \dots & -a_{n-2} & -a_{n-1} \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \tag{2.417}$$

$$C = \begin{bmatrix} c_0 & c_1 & \dots & c_{n-2} & c_{n-1} \end{bmatrix} \qquad D = 0,$$

Show that the transfer function of such a system is computed,

$$\hat{H}(s) = \frac{c_{n-1}s^{n-1} + c_{n-2}s^{n-2} + ... + c_1 s + c_0}{s^n + a_{n-1}s^{n-1} + ... + a_1 s + a_0}. \tag{2.418}$$

Problem 2.25 (Some Laplace & $\mathcal{Z}$-Transforms) In this problem, we'll establish a couple of basic Laplace and $\mathcal{Z}$-transforms.

1. Let f be a signal and $\tau > 0$. Define the signal g,

$$g(t) = \begin{cases} 0, & 0 \leq t < \tau. \\ f(t-\tau), & t \geq \tau. \end{cases} \tag{2.419}$$

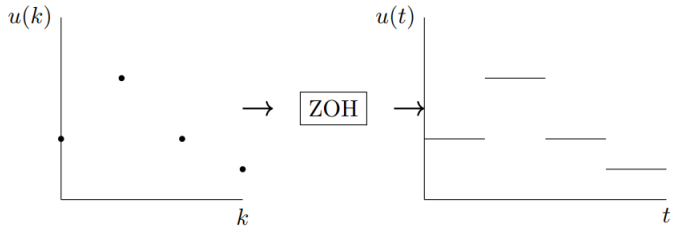
   Show that $\hat{G}(s) = e^{-s\tau}\hat{F}(s)$.
2. Show that the $\mathcal{Z}$-transform of the unit step function,

$$\mathbb{1}[k] = \begin{cases} 1, & k \geq 0 \\ 0, & k < 0, \end{cases} \tag{2.420}$$

   is $\hat{\mathbb{1}}(z) = z/(z-1)$.

Problem 2.26 (Sampled-Data Systems [37]) In this problem, we'll take a frequency-domain approach to zero-order hold discretization. Consider a continuous-time system where inputs and outputs can only be accessed at discrete times $t = k\Delta,\ k \in \mathbb{Z}$, with a sampling period $\Delta \in \mathbb{R}_{>0}$. The discrete-time input $u[k] = u(k\Delta)$ is passed through a zero-order hold (ZOH) digital-to-analog (D/A) converter that accepts the input $u(k\Delta)$ at $t = k\Delta$ and holds it constant until the next input is applied at $t = (k+1)\Delta$. The continuous-time system processes the ZOH output, and its resulting continuous-time output is sampled by the A/D converter to produce the discrete-time output $y[k]$. The goal is to compute the discrete-time transfer function of the overall system considering the effects of D/A and A/D converters.

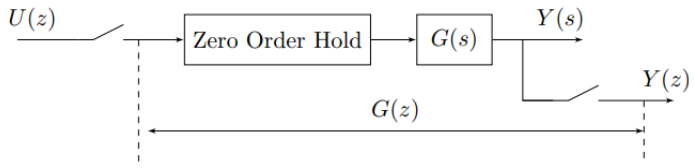
1. First, we will compute the transfer function of the zero order hold. The zero-order hold takes in a continuous-time input signal (which has been sampled with interval Δ) and returns a continuous-time *held* version of the signal.



   Show that the transfer function of the zero-order hold is

$$H(s) = \frac{1 - e^{-s\Delta}}{s}. \tag{2.421}$$

   Assume that the sampler translates a discrete-time unit impulse to a continuous-time Dirac delta.
2. Now, we apply the zero-order hold block to a system. Consider the block diagram,



where $G(s)$ is the transfer function of a continuous-time system and the latches represent sampling. Show that the transfer function from $U(z)$ to $Y(z)$ is,

$$G(z) = (1 - z^{-1})\mathcal{Z}\left[\mathcal{L}^{-1}\left(\frac{G(s)}{s}\right)\right]. \tag{2.422}$$

Hint: Split $H(s)G(s)$ into two components. How do the two components relate?

3. If the continuous-time transfer function is,

$$G(s) = \frac{a}{s^2}, \tag{2.423}$$

what is the corresponding discrete-time transfer function $G(z)$?

Problem 2.27 (The $\mathcal{Z}$-Transform & the Fibonacci Sequence) The Fibonacci sequence is defined recursively as,

$$x[0] = 0, x[1] = 1, x[k+1] = x[k] + x[k-1], \ \forall k \geq 1. \tag{2.424}$$

Compute the $\mathcal{Z}$-Transform of the Fibonacci sequence. Then, calculate the roots of the denominator of the $\mathcal{Z}$-transform. Do you recognize any of the roots? *Look up the "golden ratio" if you do not.*

Problem 2.28 (Analytic Functions) Recall that a given function $f : \Omega \to \mathbb{C}$, where $\Omega \subseteq \mathbb{C}$ is open in $\mathbb{C}$, is an *analytic function* if it is (complex) differentiable in a neighborhood of every point of $\mathbb{C}$. For each of the *scalar* functions,

$$f_1(s) = \frac{1}{s}, \ f_2(s) = e^s, \ f_3(s) = \frac{(s-1)}{(s+1)(s-1)(s+2)}, \ G(s) = C(sI - A)^{-1}B, \tag{2.425}$$

determine the largest subset of $\mathbb{C}$ on which the function is analytic.

References

1. Stephen Abbott et al. *Understanding analysis*, volume 2. Springer, 2001.
2. Panos J Antsaklis and Anthony N Michel. *Linear systems*, volume 8. Springer, 1997.
3. Karl Johan Åström and Richard Murray. *Feedback systems: an introduction for scientists and engineers*. Princeton university press, 2021.
4. Sheldon Axler. *Measure, integration & real analysis*. Springer Nature, 2020.
5. Sheldon Axler. *Linear algebra done right*. Springer Nature, 2024.
6. Frank M Callier and Charles A Desoer. *Linear system theory*. Springer Science & Business Media, 2012.
7. Chih-Yuan Chiu, Claire Tomlin, and Yi Ma. *Linear Systems*. Available online at https://ucb-ee106.github.io/106b-sp23site/assets/Linear_Systems___Professor_Ma.pdf, 2019.
8. Mohammed Dahleh, Munther A Dahleh, and George Verghese. *Lectures on dynamic systems and control*. Massachusets Institute of Technology, 2004.
9. John C. Doyle. Analysis of feedback systems with structured uncertainties. In *IEE Proceedings D (Control Theory and Applications)*, volume 129, pages 242–250. IET Digital Library, 1982.
10. John C. Doyle, Bruce A Francis, and Allen R Tannenbaum. *Feedback control theory*. Courier Corporation, 2013.
11. Geir E Dullerud and Fernando Paganini. *A course in robust control theory: a convex approach*, volume 36. Springer Science & Business Media, 2013.
12. *Linear Dynamical Systems*, 2008.
13. Stephen H Friedberg, Arnold J Insel, and Lawrence E Spence. *Linear Algebra*. Pearson, 2014.
14. Michael Green and David J.N. Limebeer. *Linear Robust Control*. Dover, 1995.
15. Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Dover, 1985.
16. Haimin Hu. *Computing Jordan Form Using Jordan Chain*. Online Distributiuon, UC Berkeley EE221A, 2017.
17. Hassan K. Khalil. *Nonlinear Systems*. Prentice Hall, 2002.
18. Edward A. Lee and Pravin Varaiya. *Structure and Interpretation of Signals and Systems, Second Edition*. Available online at https://ptolemy.berkeley.edu/books/leevaraiya/, 2011.
19. Andrew D. Lewis. *A Mathematical Approach to Classical Control*. Online Distribution, 2016.
20. Andrew D. Lewis. *Introduction to Differential Equations*. Online Distribution, 2017.
21. Andrew D. Lewis. *A Mathematical Introduction to Signals and Systems, Volume 4*. Available at https://mast.queensu.ca/ andrew/teaching/SigSys/pdf/volume4.pdf, 2022.
22. John Lygeros and Federico Ramponi. Lecture notes on linear system theory. *Automatic Control Laboratory, ETH Zurich*, 2010.
23. Alexandre Megretski. Multivariable control systems. 2004.
24. Richard M Murray. Feedback systems: Notes on linear systems theory. 2020.
25. Alan V. Oppenheim and Alan S. Willsky. *Signals and Systems*. Prentice Hall, 1997.
26. Andrew Packard and John Doyle. The complex structured singular value. *Automatica*, 29(1):71–109, 1993.
27. Andrew Packard, Roberto Horowitz, Kameshwar Poola, and Francesco Borrelli. *Dynamic Systems and Feedback, Class Notes*. Avaliable online, 2018.
28. Lawrence Perko. *Differential Equations and Dynamical Systems*. Springer, 2000.

29. Ian Postlethwaite and Sigurd Skogestad. *Multivariable Feedback Control, Analysis & Design*. Wiley, 2005.
30. Walter Rudin. *Principles of mathematical analysis*, volume 3. McGraw-hill New York, 1964.
31. Walter Rudin. *Real and Complex Analysis*. McGraw-Hill, 1987.
32. Wilson J. Rugh. *Linear System Theory*. Prentice Hall, 1996.
33. Eduardo D Sontag. *Mathematical control theory: deterministic finite dimensional systems*, volume 6. Springer Science & Business Media, 2013.
34. Elias M Stein and Rami Shakarchi. *Real analysis: measure theory, integration, and Hilbert spaces*, volume 3. Princeton University Press, 2009.
35. Elias M Stein and Rami Shakarchi. *Complex analysis*, volume 2. Princeton University Press, 2010.
36. Elias M Stein and Rami Shakarchi. *Fourier analysis: an introduction*, volume 1. Princeton University Press, 2011.
37. Masayoshi Tomizuka. *Advanced Control Systems I*. 2022.
38. Fawwaz T. Ulaby and Andrew E. Yagle. *Signals and Systems: Theory and Applications*. Available online at https://ss2-2e.eecs.umich.edu/, 2018.
39. Kemin Zhou and John C. Doyle. *Essentials of robust control*. Prentice hall, 1998.
40. Kemin Zhou, John C. Doyle, and Keith Glover. *Robust and Optimal Control*. Prentice hall, 1996.