Lecture 0: Preliminaries

Erfan Nozari

November 25, 2020

Welcome to ME 120! This course provides you with the basics of linear systems analysis and control and is aimed to catalyze, at an introductory level, between pure math and engineering applications.

In this note I brush up some of the basic mathematical concepts that we need to begin with: differential equations, complex numbers, and the Laplace transform. Remember that I am assuming you have learned these in related courses before, and here only aim to briefly review them, so feel free to make a pass at your old notes and textbooks if you feel the need to!

Contents

0.1 Ordinary Differential Equations (ODEs)

Remember that an ODE is a mathematical algebraic relationship between a function, say $x(t)$, and its derivatives. The independent variable t does not need to be time, but in this course we are only interested in derivatives with respect to time. Some toy examples are

$$\dot{x}(t) = x(t) \tag{0.1a}$$
$$a\ddot{x}(t) + b\dot{x}(t) + cx(t) = u(t) \tag{0.1b}$$
$$\cos\left(x^3(t) - 4\dot{x}^2(t) - e^{\ddot{x}(t)}\right) - \sin(8t) = 0 \tag{0.1c}$$

but there are also extensively used real-world examples all over science and engineering, such as

Newton's 2nd law:	$u(t) = m\ddot{x}(t)$	(0.2a)
Harmonic oscillator:	$\ddot{x}(t) = -\alpha x(t)$	(0.2b)
Van der Pol's oscillator:	$\ddot{x}(t) + \mu(x^2(t) - 1)\dot{x}(t) + x(t) = 0$	(0.2c)
Hodgkin-Huxley neuron model:	$u(t) = C\dot{x}(t) + g_{\text{K}}(x(t) - V_{\text{K}}) + g_{\text{Na}}(x(t) - V_{\text{Na}}) + g_{\text{L}}(x(t) - V_{\text{L}})$	(0.2d)
Chemical reactions:	$\dot{x}(t) = \gamma_1 k_1 x(t)^a u(t)^b + \gamma_2 k_2 x(t)^c u(t)^d$	(0.2e)

Notation 0.1.1 (Time derivative) As you have noticed, we use dots to show derivatives, so

$$\dot{x}(t) = \frac{d}{dt}x(t)$$
$$\ddot{x}(t) = \frac{d^2}{dt^2}x(t)$$

and so on. $\hspace{1cm}\square$

In all the ODEs in Eq. (0.1) and Eq. (0.2), the constant variables $a, b, c, m, \alpha, \mu, C, ...$ are parameters. More important is the function $u(t)$ of time, which represents "external inputs" to the ODE. In general $u(t)$ may be any function of time, such as

$$u(t) = 0$$
$$u(t) = \alpha e^{\beta t}$$
$$u(t) = A\cos(\omega t + \theta_0)$$
$$\vdots$$

Solving an ODE means finding a function (or functions) $x(t)$ that satisfy the ODE. The function $x(t) = e^t$, for example, clearly satisfies Eq. (0.1a). But so does $x(t) = ke^t$ for any number k. This is why solving an ODE for a *unique solution* needs more information, typically provided as "initial conditions"

$$x(t_0) = x_0$$
$$\dot{x}(t_0) = \dot{x}_0$$
$$\vdots$$

where t_0 is some "initial time" (often 0) and $x_0, \dot{x}_0, \ldots$ are given numbers. The number of initial conditions needed to solve an ODE for a unique solution is typically as many as the largest order of derivative in the ODE.

As you might remember from your course in differential equations, not all ODEs can be analytically solved, and solving them is quite some art! Fortunately, we here only care about linear ODEs, which have various systematic method to solve. We will review one such method using the Laplace transform in Section 0.4.

MATLAB 0.1.2 (Solving ODEs) Recall that you can use `dsolve()` to solve ODEs analytically (if possible) and `ode45()` (or many other variants) to solve them numerically. To get you started,

```
1  syms x(t)
2  sol = dsolve(diff(x) == x, x(0) == 2);
```

solves Eq. (0.1a) with $x(0) = 2$ analytically, while

```
1  odefun = @(t, x)x;
2  tinterval = [0 100];
3  x0 = 2;
4  [tspan, x] = ode45(odefun, tinterval, x0);
```

solves it numerically over $t \in [0, 100]$. What would you do in each case if you have $\ddot{x}(t)$ in the equation? □

0.2 Complex Numbers

Complex numbers are essential in the linear systems theory because of their role in finding the roots of polynomial functions (to be discussed shortly). Recall (or learn!) that a complex number is nothing but an ordered pair (x, y) of real numbers $x, y \in \mathbb{R}$, with a specific rule for multiplication such that

$$(x_1, y_1) \cdot (x_2, y_2) = (x_1 x_2 - y_1 y_2, x_1 y_2 + x_2 y_1) \tag{0.3}$$

For simplicity, we then define

$$j = (0, 1)$$

as the imaginary unit number, and take complex number $(x, 0)$ equivalent to the real number x. Then, it is easy to show that any complex number (x, y) can also be represented as $x + jy$ (check yourself!), which is what we often know as complex numbers. It also follows from Eq. (0.3) that

$$j^2 = j \cdot j = (0, 1) \cdot (0, 1) = (-1, 0) = -1$$

or, as commonly known, $j = \sqrt{-1}$.

One of the most important elements in complex analysis is the complex exponential. Recall that the real exponential e^x is defined though the infinite sum

$$e^x = 1 + x + \frac{x^2}{2} + \frac{x^3}{3!} + \cdots = \sum_{k=0}^{\infty} \frac{x^k}{k!}$$

and has the property that $e^{x+y} = e^x e^y$.

Notation 0.2.1 (Real and complex numbers) You have noticed that we use $\mathbb{R}$ to show the set of real numbers. We will similarly use $\mathbb{C}$ for the set of complex numbers. □

Definition 0.2.2 (Complex exponential) For a complex number $z = x + jy \in \mathbb{C}$, the complex exponential is defined using the same sum as the real exponential,

$$e^z = \sum_{k=0}^{\infty} \frac{z^k}{k!}$$

3

and has the property that, for $x \in \mathbb{R}$,

$$e^{jx} = \cos x + j \sin x$$

which is called the Euler's formula. □

Similar to the real exponential, the complex exponential satisfies $e^{z_1 + z_2} = e^{z_1} e^{z_2}$. Therefore,

$$e^{x+jy} = e^x e^{jy} = e^x \cos y + j e^x \sin y$$

This motivates a very useful transformation in complex numbers, that to "polar coordinates".

Definition 0.2.3 (Cartesian and polar forms) The complex number $z = x + jy$ is said to be in Cartesian coordinates, since it can be drawn as the point (x, y) on the Cartesian plane (shown below). Let (r, θ) be the representation of the same point in the plane in polar coordinates, as shown. Then

$$z = x + jy = r \cos \theta + jr \sin \theta = r\left(\cos \theta + j \sin \theta\right) = re^{j\theta}$$

The representation $z = re^{j\theta}$ is called the polar form of z.

□
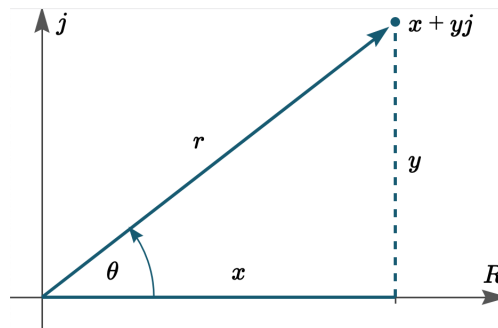
We are now ready for a few more fundamental definitions about complex numbers.

Definition 0.2.4 (Complex number attributes) For a complex number $z = x + jy = re^{j\theta}$,

- x is called the real part of z, shown as $\mathrm{Re}\{z\}$;

- y is called the imaginary part of z, shown as $\mathrm{Im}\{z\}$;

- r is called the modulus of z, shown as $|z|$, and equal to $\sqrt{x^2 + y^2}$; note that

$$|e^{j\theta}| = \sqrt{\cos^2 \theta + \sin^2 \theta} = 1$$

- θ is called the argument of z, shown as $\angle z$

- $\bar{z} = x - jy = re^{-j\theta}$ is called the complex conjugate of z.

□

As I mentioned earlier, our main reason to dealing with complex numbers in linear systems and control is because of their role in the roots of polynomial functions. The following theorem shows how.

4

Theorem 0.2.5 (Fundamental theorem of algebra) Consider the polynomial function

$$p(\lambda) = a_n \lambda^n + a_{n-1} \lambda^{n-1} + \cdots + a_1 \lambda + a_0.$$

where the coefficients $a_0, \ldots, a_n$ are complex numbers and not all equal to 0. Then, the equation

$$p(\lambda) = 0 \tag{0.4}$$

has exactly n (potentially repeated) complex solutions called the roots of the polynomial p. ☐

In this course, we only care about the cases where the coefficients $a_0, \ldots, a_n$ are real. In this case, it is not hard to show that all the solutions to Eq. (0.4) are either real, or if they are complex, they come in complex conjugate form. In other words, if $p(\lambda) = 0$, then necessarily $p(\bar{\lambda}) = 0$ as well.

Exercise 0.2.6 (Roots of cubic polynomial) How many of the roots of $p(\lambda) = a_3 \lambda^3 + a_2 \lambda^2 + a_1 \lambda + a_0$ can be complex? ☐

In order to find the roots, one can break the polynomial into its factors. If the roots of Eq. (0.4) are called $\lambda_1, \ldots, \lambda_n$,

$$p(\lambda) = a_n(\lambda - \lambda_1)(\lambda - \lambda_2) \cdots (\lambda - \lambda_n) \tag{0.5}$$

This factorization can be obtained, in theory, using polynomial long division (which you hopefully remember, don't you?). However, one major difficulty in working with a factorization like Eq. (0.5) is that the numbers $\lambda_1, \ldots, \lambda_n$ can be complex. But if so, they have to come in complex conjugate pairs, so for any root $\lambda_i = x + jy$ we also have $\bar{\lambda}_i = x - jy$ among the roots. Then, notice that

$$(\lambda - \lambda_i)(\lambda - \bar{\lambda}_i) = (\lambda - x - jy)(\lambda - x + jy) = \lambda^2 - 2x\lambda + x^2 + y^2$$

which is all real. Therefore, the complex factorization in Eq. (0.5) has the alternative real factorization of

$$p(\lambda) = a_n(\lambda - \sigma_1) \cdots (\lambda - \sigma_m)(\lambda^2 - 2\operatorname{Re}\{\kappa_1\}\lambda + |\kappa_1|^2) \cdots (\lambda^2 - 2\operatorname{Re}\{\kappa_k\}\lambda + |\kappa_k|^2) \tag{0.6}$$

where $\sigma_1, \ldots, \sigma_m$ are the real roots of p and $\kappa_1, \bar{\kappa}_1, \ldots, \kappa_k, \bar{\kappa}_k$ are its complex roots. Going from Eq. (0.6) to Eq. (0.5) is then easy, right?

MATLAB 0.2.7 (Roots of polynomials) Numerically, use the function `roots()` to find the roots of a polynomial. Provide only the polynomial coefficients to the function, as for $p(\lambda) = 3\lambda^5 - 5\lambda^4 + 3\lambda^2$ below

```
1  p = [3 -5 0 3 0 0];
2  lambda = roots(p)
```

Alternatively, define $p(\lambda)$ as a symbolic expression and use `factor()` to obtain its real or complex factorization:

```
1  syms lambda
2  p(lambda) = 3 * lambda^5 - 5 * lambda^4 + 3 * lambda^2;
3  p_cplx_fact = factor(p, 'FactorMode', 'complex')
4  p_real_fact = factor(p, 'FactorMode', 'real')
```

☐

5

0.3   The Laplace Transform

Broadly speaking, when you are working in systems and controls, you are either workin "in the time domain" or "in the frequency domain". Recall the ODEs of Section 0.1 and the fact that the dependent variable t is always time for us. So working in the time domain means working with the ODE directly. For example, you should have seen plenty of "classical methods" for solving ODEs in your differential equations courses (the separation of variables, etc.), which are "in the time domain" because they only work with $x(t)$, its derivatives, and so on.

To get to the frequency domain, we use the Laplace transform (or its special case, the Fourier transform). Recall that for a function $x(t)$, its Laplace transform is defined as

$$X(s) = \mathcal{L}\{x(t)\} = \int_{-\infty}^{\infty} e^{-st} x(t) dt \tag{0.7}$$

where the variable s is complex (unlike t which was real), assuming that the integral exists. If $x(t)$ is defined only over $[0, \infty)$, which is usually the case for us, then the domain of the integral automatically shrinks to $\int_0^{\infty}$ (assuming $x(t) = 0$ for $t < 0$). If t is time (in seconds), then s will be frequency (in Hertz), so that their product st is dimensionless (why?). Therefore, $X(s)$ is said to be in the frequency domain. Recall that the Fourier transform, which is often the default when one talks about the "frequency domain", is a special case of the Laplace transform:

Definition 0.3.1 (Fourier transform) The Fourier transform of the signal $x(t)$ is its Laplace transform evaluated at $s = j\omega$,

$$X(j\omega) = \int_{-\infty}^{\infty} e^{-j\omega t} x(t) dt$$

assuming that the integral exists. □

Computing the integral in Eq. (0.7) is not always possible, but is doable for most elementary functions. Before getting there, let's define (recap?) a few of these elementary functions.

Definition 0.3.2 (Step, impulse, and ramp functions) A few functions are of particular importance, including

   (i) A "step" function (also called a Heaviside function) is one that has the form

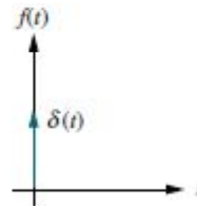$$H(t) = \begin{cases} 0 & \text{if } t < 0 \\ 1 & \text{if } t \geq 0 \end{cases}$$
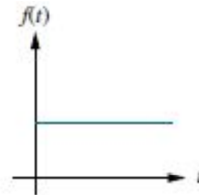
It basically models the output of a switch turned from off to on at time 0.

   (ii) An "impulse" function (also called a Dirac delta function) is the derivative of the step signal:

$$\delta(t) = \frac{d}{dt} H(t) = \begin{cases} 0 & \text{if } t \neq 0 \\ \infty & \text{if } t = 0 \end{cases}$$
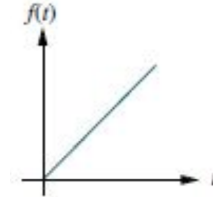
$$\int_{-\epsilon}^{+\epsilon} \delta(t) dt = 1 \qquad \text{for all } \epsilon > 0$$

6

It models, e.g., the instantaneous impact of an object hitting another. If the definition (and the plot) is a bit weird, don't worry, you'll get used to it! (but there are perfectly rigorous ways to mathematically define it as well.)

(iii) A "ramp" function is the integral of the step function:

$$r(t) = \int_0^t H(\tau)d\tau = \begin{cases} 0 & \text{if } t < 0 \\ t & \text{if } t \geq 0 \end{cases} = tH(t)$$

It is used, e.g., to model an error that accumulates in time.

□

The following theorem reviews the Laplace transforms that we will need in this course.

Theorem 0.3.3 (Basic Laplace transforms) The following is a list of Laplace transform pairs. In it, all functions of t are defined for $t \geq 0$ and are 0 for $t < 0$ (which is why I have written $H(t) = 1$ and $r(t) = t$ in the table).

$x(t)$	$X(s)$
$\delta(t)$	1
$H(t) = 1$	$\frac{1}{s}$
$r(t) = t$	$\frac{1}{s^2}$
$\frac{1}{n!}t^n$	$\frac{1}{s^{n+1}}$
$e^{-\lambda t}$	$\frac{1}{s+\lambda}$
$\frac{1}{n!}t^n e^{-\lambda t}$	$\frac{1}{(s+\lambda)^{n+1}}$
$\sin \omega t$	$\frac{\omega}{s^2+\omega^2}$
$\cos \omega t$	$\frac{s}{s^2+\omega^2}$
$e^{-\lambda t}\sin \omega t$	$\frac{\omega}{(s+\lambda)^2+\omega^2}$
$e^{-\lambda t}\cos \omega t$	$\frac{s+\lambda}{(s+\lambda)^2+\omega^2}$

□

If memorizing all of the above is difficult, remember that some can be obtained from others using the properties of the Laplace transform, such as the frequency shift property:

Theorem 0.3.4 (Laplace transform properties) Let $X(s)$ and $Y(s)$ be the Laplace transforms of $x(t)$ and $y(t)$, respectively. The following properties hold (prime shows differentiation with respect to s):

Time domain	Frequency domain
$\alpha x(t) + \beta y(t)$	$\alpha X(s) + \beta Y(s)$
$tx(t)$	$-X'(s)$

7

$t^n x(t)$	$(-1)^n X^{(n)}(s)$
$\dot{x}(t)$	$sX(s) - x(0)$
$\ddot{x}(t)$	$s^2 X(s) - sx(0) - \dot{x}(0)$
$e^{-\lambda t} x(t)$	$X(s + \lambda)$
$x(t) * y(t) = \int_0^t x(\tau)y(t - \tau)d\tau$	$X(s)Y(s)$
$x(t - t_0)$	$e^{-st_0} X(s)$

$\square$

Using these basic transforms and properties, one can take the Laplace transform of a wide range of functions. A couple of examples are shown next.

Example 0.3.5 (Laplace transforms) A few simple examples to remind yourself of taking Laplace transforms:

(i) $x(t) = 7\cos 3t - \sin 5t$

$$X(s) = 7\mathcal{L}\{\cos 3t\} - \mathcal{L}\{\sin 5t\} = \frac{7s}{s^2 + 9} - \frac{5}{s^2 + 25} = \frac{7s^3 - 5s^2 + 175s - 45}{s^4 + 34s^2 + 225}$$

(ii) $x(t) = 2H(t - 3) - r(t - 4)$

$$X(s) = 2e^{-3s}\frac{1}{s} - e^{-4s}\frac{1}{s^2} = \frac{2se^{-3s} - e^{-4s}}{s^2}$$

(iii) $x(t) = t^2 e^t \cos(\pi t - 2)$

$$\mathcal{L}\{\cos(\pi t - 2)\} = \mathcal{L}\{\cos 2 \cos \pi t + \sin 2 \sin \pi t\} = \frac{s\cos 2 + \pi \sin 2}{s^2 + \pi^2}$$

$$\mathcal{L}\{t\cos(\pi t - 2)\} = -\frac{d}{ds}\frac{s\cos 2 + \pi \sin 2}{s^2 + \pi^2} = -\frac{s^2 \cos 2 + 2\pi s \sin 2 - \pi^2 \cos 2}{(s^2 + \pi^2)}$$

$$\mathcal{L}\{t^2 \cos(\pi t - 2)\} = \frac{d}{ds}\frac{s^2 \cos 2 + 2\pi s \sin 2 - \pi^2 \cos 2}{(s^2 + \pi^2)^2}$$

$$= \frac{2s^3 \cos 2 + 6\pi s^2 \sin 2 - 6\pi^2 s \cos 2 - 2\pi^3 \sin 2}{(s^2 + \pi^2)^3}$$

$$X(s) = \frac{2(s - 1)^3 \cos 2 + 6\pi(s - 1)^2 \sin 2 - 6\pi^2(s - 1)\cos 2 - 2\pi^3 \sin 2}{((s - 1)^2 + \pi^2)^3}$$

$\square$

Using the Laplace transform, we are able to go from the time domain to the frequency domain. Sometimes, that is all we need, and the rest of our analysis is all performed in the frequency domain. Other times, however, we still need to get back to the time domain. This is achieved using the inverse Laplace transform. It has a formal definition using complex integrals, but the inverse Laplace transform is almost always taken by inspection, i.e., by decomposing $X(s)$ into smaller terms whose Laplace transforms we know from tables. Here are some examples.

Example 0.3.6 (Inverse Laplace transform) Pretty much all throughout the course, we will be working with rational Laplace transforms (the ratio between two polynomials). For example,

$$\begin{aligned}
X(s) &= \frac{s^3 + 3s^2 + s + 3}{s^4 + 3s^3 + 2s^2} \\
&= \frac{s^3 + 3s^2 + s + 3}{s^2(s+1)(s+2)} \\
&= -\frac{7}{4s} + \frac{3}{2s^2} + \frac{4}{s+1} - \frac{5}{4(s+2)} \\
x(t) &= -\frac{7}{4}H(t) + \frac{3}{2}tH(t) + 4e^{-t}H(t) - \frac{5}{4}e^{-2t}H(t), \qquad \text{for all } t \\
&= -\frac{7}{4} + \frac{3}{2}t + 4e^{-t} - \frac{5}{4}e^{-2t}, \qquad \text{for all } t \geq 0
\end{aligned}$$

□

The question that you must have at this point, unless you know the answer, is how did we get from

$$X(s) = \frac{s^3 + 3s^2 + s + 3}{s^2(s+1)(s+2)}$$

to

$$X(s) = -\frac{7}{4s} + \frac{3}{2s^2} + \frac{4}{s+1} - \frac{5}{4(s+2)}$$

which is the key step. This is called "Partial Fraction Expansion", and there are different ways for doing it. The simplest method, also called the coverup method, is well explained here. For the purpose of this class, however, you can use MATLAB:

MATLAB 0.3.7 (Laplace and Laplace inverse) To compute the Laplace transform of a function in MATLAB, simply define it symbolically and use the `laplace()` function. The `ilaplace()` takes the inverse transform. Also, if you want (or are asked!) to take the Laplace inverse using partial fraction expansion, you can "check your answers" either using the `residue()` or, even easier, the `partfrac()` functions. The `residue()` function is a numerical one, while `partfrac()` is symbolic. Also, `residue()` will break your denominator all the way into order 1 factors, which may not remain real-valued. With `partfrac()` you have the option of going all the way to order 1 factors (using `'FactorMode', 'complex'`), allowing for quadratic factors in the denominator to ensure real-valued answers (using `'FactorMode', 'real'`), or even allowing for higher order factors in the denominator to ensure rational answers (using `'FactorMode', 'rational'`). Here is some code for the above example:

```matlab
% Using partfrac
syms s
X(s) = (s^3 + 3*s^2 + s + 3)/(s^4 + 3*s^3 + 2*s^2);
X_rat(s) = partfrac(X, 'FactorMode', 'rational')
X_real(s) = partfrac(X, 'FactorMode', 'real')
X_cplx(s) = partfrac(X, 'FactorMode', 'complex')
% Using residue
b = [1 3 1 3];
a = [1 3 2 0 0];
[r, p] = residue(b, a)
```

□

9

0.4 Solving ODEs Using the Laplace Transform

Equipped with the Laplace transform machinery, let's get back to the ODEs which are the bread and butter of linear systems theory. As mentioned earlier, there is a host of methods, some of which you should have seen in your differential equations course, to solve ODEs directly in the time domain without any use of Laplace transform. In fact, *for nonlinear ODEs, that's the only option* (we will formally review linearity in later lectures). But if the your ODE is linear, you also have the option of solving it using the Laplace transform, and it is usually simpler. Here is how:

Consider, for example, the following second-order linear ODE,

$$a\ddot{x}(t) + b\dot{x}(t) + cx(t) = u(t), \qquad x(0) = x_0, \quad \dot{x}(0) = \dot{x}_0$$

Recall that $u(t)$ is a *given* function of time which act as external input to the ODE. Use the properties of Theorem 0.3.4 and take the Laplace transform of both sides:

$$a(s^2 X(s) - sx(0) - \dot{x}(0)) + b(sX(s) - x(0)) + cX(s) = U(s)$$

Now, just solve for $X(s)$, which is a simple *algebraic* equation (instead of an ODE):

$$(as^2 + bs + c)X(s) = (as + b)x_0 + a\dot{x}_0 + U(s)$$

$$X(s) = \frac{as + b}{as^2 + bs + c}x_0 + \frac{a}{as^2 + bs + c}\dot{x}_0 + \frac{1}{as^2 + bs + c}U(s) \qquad (0.8)$$

Then, use the expertise that you just acquired in taking inverse Laplace transforms to get $x(t)$ back!

$$x(t) = \mathcal{L}^{-1}\{X(s)\} = \mathcal{L}^{-1}\left\{ \frac{as + b}{as^2 + bs + c}x_0 + \frac{a}{as^2 + bs + c}\dot{x}_0 + \frac{1}{as^2 + bs + c}U(s) \right\}$$

Remember, $U(s)$ is given (say $u(t) = \sin t$ or whatever, which gives you $U(s)$). Also, doesn't Eq. (0.8) look so nicely additive? We'll get back to it.