6.01 Introduction to EECS via Robotics

Lecture 3: Analyzing System Behavior

Lecturer: Adam Hartz (hz@mit.edu)

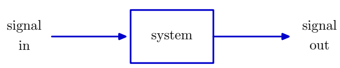
As you come in...

- Grab one handout (on the table by the entrance)

Notes

The Signals and Systems Abstraction

Describe a **system** (physical, mathematical, or computational) by the way it transforms an input signal into an output signal.



Focus on **Linear, Time-Invariant** (LTI) Systems.

Notes

Signals and Systems: Representations

Last week, 3 main representations:

- **Difference Equation**
- **Block Diagram**
- **Operator Equation**

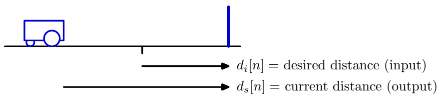
Today, 2 new representations:

- **System Functional**
- **Poles**

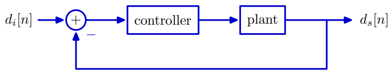
Notes

Example: `wallFinder`

Consider the `wallFinder` from design lab 1 and 2:

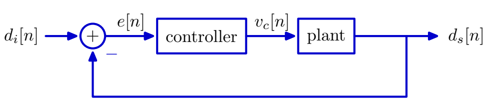


Think about this system as having 2 parts:



Notes

Example: `wallFinder`



Controller (brain): sets commanded velocity $\propto$ error:

$$v_c[n] = ke[n] = k(d_i[n] - d_s[n])$$

Plant (robot locomotion): given $v_c[n]$, derives new position:

$$v_a[n] = v_c[n-1]$$

$$p[n] = p[n-1] + Tv[n-1]$$

$$d_s[n] = -p[n]$$

Notes

Check Yourself!

Solving difference equations:

$$v_c[n] = ke[n] = k(d_i[n] - d_s[n])$$
$$v_a[n] = v_c[n-1]$$
$$p[n] = p[n-1] + Tv_a[n-1]$$
$$d_s[n] = -p[n]$$

How many equations? How many unknowns?

1. 4 equations; 4 unknowns
2. 4 equations; 5 unknowns
3. 5 equations; 5 unknowns
4. 4 equations; 8 unknowns
5. none of the above

Hint: T and k are fixed (constant) parameters and the input is known.

Notes

Check Yourself!

Solving operator equations:

$$V_c = k(D_i - D_s)$$
$$V_a = \mathcal{R} V_c$$
$$P = \mathcal{R} P + T \mathcal{R} V_a$$
$$D_s = -P$$

How many equations? How many unknowns?

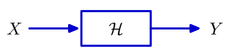
1. 4 equations; 4 unknowns
2. 4 equations; 5 unknowns
3. 5 equations; 5 unknowns
4. 4 equations; 8 unknowns
5. none of the above

Hint: T and k are fixed (constant) parameters and the input is known.

Notes

System Functional

We can express the relation between the (known) input and the (unknown) output using the system functional $\mathcal{H}$.



The system functional $\mathcal{H}$ is an operator.

Applying $\mathcal{H}$ to X yields Y.

$$Y = \mathcal{H} X$$

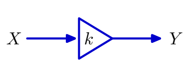
It is also convenient to think of $\mathcal{H}$ as a ratio:

$$\mathcal{H} = \frac{Y}{X}$$

Notes

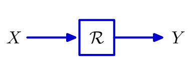
System Functional: Primitives

Gain:



$$\frac{Y}{X} = k$$

Delay:

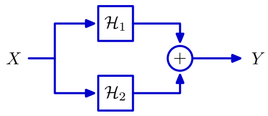


$$\frac{Y}{X} = \mathcal{R}$$

Notes

System Functional: Feedforward Add

Consider two systems (with system functionals $\mathcal{H}_1$ and $\mathcal{H}_2$) connected in *feedforward add* configuration:

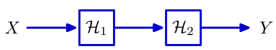


What is the system functional $\frac{Y}{X}$ of this composite system?

Notes

System Functional: Cascade

Consider two systems (with system functionals $\mathcal{H}_1$ and $\mathcal{H}_2$) connected in *cascade* configuration:

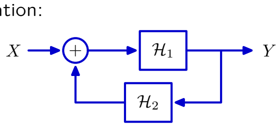


What is the system functional $\frac{Y}{X}$ of this composite system?

Notes

System Functional: Feedback

Consider two systems (with system functionals $\mathcal{H}_1$ and $\mathcal{H}_2$) connected in *feedback add* configuration:



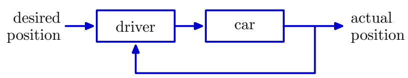
What is the system functional $\frac{Y}{X}$ of this composite system?

Notes

Feedback

Feedback (as we saw in lab last week) is pervasive in natural and artificial systems.

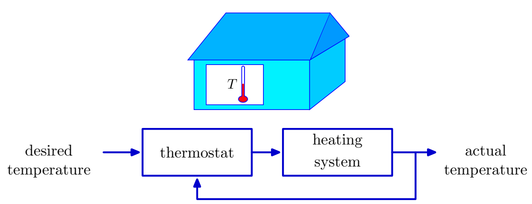
Driving, trying to keep the car in the center of the road:



Notes

Feedback

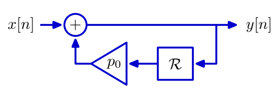
Control Systems: Feedback is useful for regulating a system's behavior



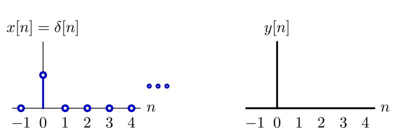
Notes

Feedback

Consider a small feedback system:



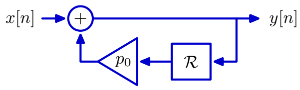
Find $y[n]$ given $x[n] = \delta[n]$: $y[n] = x[n] + p_0 y[n-1]$



Notes

Feedback

Alternatively, we can think about *signals* instead of *samples*.



$$Y = X + p_0\mathcal{R}Y$$

$$(1 - p_0\mathcal{R})Y = X$$

$$\frac{Y}{X} = \frac{1}{1 - p_0\mathcal{R}}$$

Notes

Feedback

$$\frac{Y}{X} = \frac{1}{1 - p_0\mathcal{R}}$$

We can show that this is right algebraically:

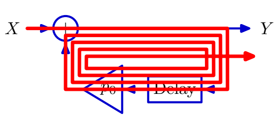
$$\begin{array}{r|l} & 1 \;\; +p_0\mathcal{R} \;\; +p_0^2\mathcal{R}^2 \;\; +p_0^3\mathcal{R}^3 \;\; +\cdots \\ \hline 1 - p_0\mathcal{R} & 1 \\ & \underline{1 \;\; -p_0\mathcal{R}} \\ & p_0\mathcal{R} \\ & \underline{p_0\mathcal{R} \;\; -p_0^2\mathcal{R}^2} \\ & p_0^2\mathcal{R}^2 \\ & \underline{p_0^2\mathcal{R}^2 \;\; -p_0^3\mathcal{R}^3} \\ & p_0^3\mathcal{R}^3 \\ & \underline{p_0^3\mathcal{R}^3 \;\; -p_0^4\mathcal{R}^4} \\ & \cdots \end{array}$$

Therefore, $\frac{1}{1-p_0\mathcal{R}} = 1 + p_0\mathcal{R} + p_0^2\mathcal{R}^2 + p_0^3\mathcal{R}^3 + p_0^4\mathcal{R}^4 + \cdots$

Notes

Feedback: Cyclic Signal Flow Paths

We can also see this graphically:



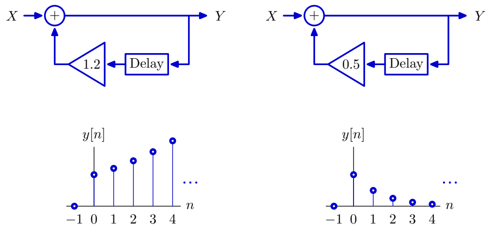
$$\frac{Y}{X} = \frac{1}{1 - p_o\mathcal{R}} = 1 \;+p_0\mathcal{R} \;+p_0^2\mathcal{R}^2 \;+p_0^3\mathcal{R}^3 \;+p_0^4\mathcal{R}^4 \;+\ldots$$

Cyclic flow paths: **persistent** response to **transient** input.

Notes

Geometric Growth

If traversing the cycle decreases or increases the magnitude of the signal, then the output will decay or grow.

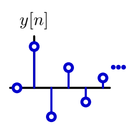


Geometric Sequences: $y[n] = (1.2)^n$ and $(0.5)^n$ for $n \geq 0$.
These responses can be characterized by a single number (the **pole**), which is the base of the geometric sequence.

Notes

Check Yourself!

What value of p_0 is associated with the signal below?

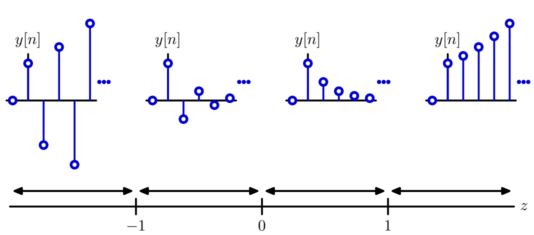


0. $p_0 = 0.7$
1. $p_0 = -0.7$
2. $p_0 = 0.7$ interspersed with $p_0 = -0.7$
3. $p_0 = -0.5$
4. $p_0 = 0.5$ interspersed with $p_0 = -0.5$
5. None of the above

Notes

Geometric Growth

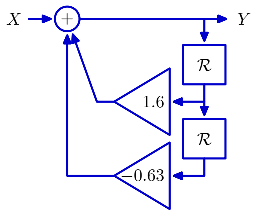
The value of p_0 determines the rate of growth:



Notes

Second-order Systems

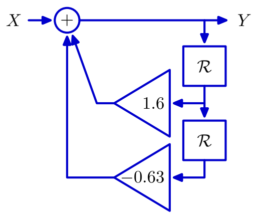
The unit-sample response of more complicated feedback systems is more complicated.



Notes

Check Yourself!

Let $x[n] = \delta[n]$. Find $y[2]$.

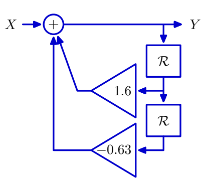


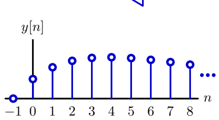
1. 1.6
2. $1.6 - 0.63$
3. $(1.6)^2 - 0.63$
4. $1.6(1.6 - 0.63)$
5. None of the above

Notes

Second-order Systems

The unit-sample response of more complicated feedback systems is more complicated.



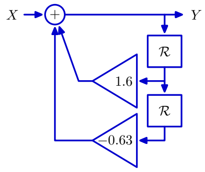


Not geometric! Grows, and then decays.

Notes

Equivalent Forms

Factor the operator expression to break the system into two simpler systems:



$$Y = X + 1.6\mathcal{R}Y - 0.63\mathcal{R}^2Y$$

$$(1 - 1.6\mathcal{R} + 0.63\mathcal{R}^2)Y = X$$

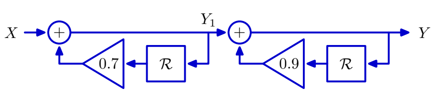
$$(1 - 0.7\mathcal{R})(1 - 0.9\mathcal{R})Y = X$$

Notes

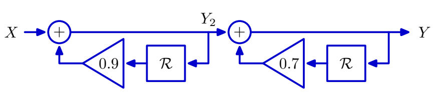
Equivalent Forms

Factored form corresponds to a cascade of simpler systems:

$$(1 - 0.7\mathcal{R})(1 - 0.9\mathcal{R})Y = X$$



$(1 - 0.7\mathcal{R})\,Y_1 = X$ $(1 - 0.9\mathcal{R})\,Y = Y_1$



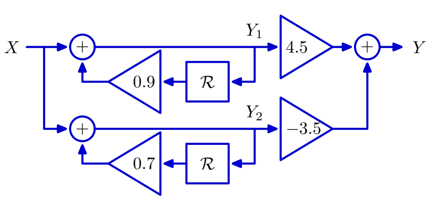
$(1 - 0.9\mathcal{R})\,Y_2 = X$ $(1 - 0.7\mathcal{R})\,Y = Y_2$

Notes

Equivalent Forms

Even better, the system functional can also be written as a **sum** of simpler parts:

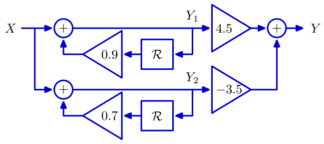
$$\frac{Y}{X} = \frac{1}{(1 - 0.9\mathcal{R})(1 - 0.7\mathcal{R})} = \frac{4.5}{1 - 0.9\mathcal{R}} - \frac{3.5}{1 - 0.7\mathcal{R}}$$



Notes

Equivalent Forms

USR is the **sum** of scaled geometric sequences.



$$\frac{Y}{X} = \frac{4.5}{1 - 0.9\mathcal{R}} - \frac{3.5}{1 - 0.7\mathcal{R}}$$

Let $x[n] = \delta[n]$

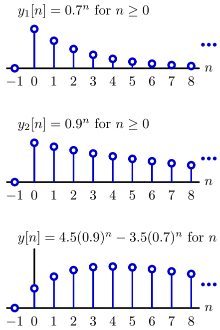
Then $y_1[n] = (0.9)^n$ and $y_2[n] = (0.7)^n$, so

$$y[n] = 4.5(0.9)^n - 3.5(0.7)^n$$

Notes

Equivalent Forms

USR is the **sum** of scaled geometric sequences.



Notes

Finding Poles

Poles can be identified by factoring the denominator of the system functional:

$$\frac{Y}{X} = \frac{b_0 + b_1\mathcal{R} + b_2\mathcal{R}^2 + \ldots}{1 + a_1\mathcal{R} + a_2\mathcal{R}^2 + \ldots}$$

$$\frac{Y}{X} = \frac{b_0 + b_1\mathcal{R} + b_2\mathcal{R}^2 + \ldots}{(1 - p_0\mathcal{R})(1 - p_1\mathcal{R})(1 - p_2\mathcal{R})\ldots}$$

The poles are the p_i values. One geometric mode p_i^n arises from each pole.

Notes

Finding Poles

$$\frac{Y}{X} = \frac{b_0 + b_1\mathcal{R} + b_2\mathcal{R}^2 + \ldots}{(1-p_0\mathcal{R})(1-p_1\mathcal{R})(1-p_2\mathcal{R})\ldots}$$

Partial fraction expansion:

$$\frac{Y}{X} = \frac{c_0}{1-p_0\mathcal{R}} + \frac{c_1}{1-p_1\mathcal{R}} + \frac{c_2}{1-p_2\mathcal{R}} + \ldots + f_0 + f_1\mathcal{R} + f_2\mathcal{R}^2 + \ldots$$

If the system functional is a *proper* rational polynomial, then the unit sample response is:

$$y[n] = c_0 p_0^n + c_1 p_1^n + c_2 p_2^n + \ldots$$

Notes

Finding Poles

The poles can also be found by finding the roots of the denominator polynomial after expressing the system functional as a ratio of polynomials in $z = \mathcal{R}^{-1}$.

$$\frac{Y}{X} = \frac{1}{1 - 1.6\mathcal{R} + 0.63\mathcal{R}^2} = \frac{1}{1 - \frac{1.6}{z} + \frac{0.63}{z^2}} = \frac{z^2}{z^2 - 1.6z + 0.63}$$

Poles at $z = 0.7$, $z = 0.9$

Notes

Long-term Behavior: Dominant Pole

When analyzing systems' poles, we are interested in **long-term** behavior (not specific samples).

As $n \to \infty$, how does $y[n]$ behave?

We have seen that a system's unit sample response can be written in the form:

$$y[n] \sim \sum_k c_k p_k^n$$

In the "large-n" case, all poles but the one with the largest magnitude die away, and so looking at the dominant pole alone tells us about the behavior of the system in that case.

Notes

Check Yourself!

Consider the system described by:

$$y[n] = -\frac{1}{4}y[n-1] + \frac{1}{8}y[n-2] + x[n-1] - \frac{1}{2}x[n-2]$$

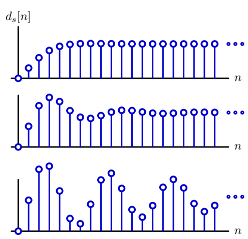
How many of the following are true?

1. The unit sample response converges to 0.
2. There are poles at $z = 0.5$ and $z = 0.25$.
3. There is a pole at $z = 0.5$.
4. There are two poles.
5. None of the above.

Notes

Wall Finder Revisited

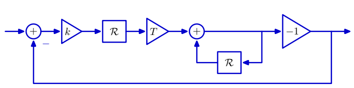
The "bunny" system always has the same behavior ($y[n] \to \infty$ as $n \to \infty$) no matter what. By contrast, our "wall-finder" robot exhibited drastically different behaviors depending on the choice of gain k.



Today: Examine that dependence, develop a means for determining "best" k analytically.

Notes

Wall Finder: Poles



Notes

Dependence of Poles on Gain

Notes

Complex Poles

What if a pole has a non-zero imaginary part?

Example:

$$\frac{Y}{X} = \frac{1}{1 - \mathcal{R} + \mathcal{R}^2}$$

Poles at $z = \frac{1}{2} \pm \frac{\sqrt{3}}{2}j$.

Unit sample response still goes like poles raised to the power n!

Need to understand what happens when complex numbers are raised to integer powers.

Notes

Complex Poles

Easiest to understand when poles are represented in *polar form*:

A number $p_0 = a_0 + b_0 j$ can be represented by a magnitude and an angle in the complex plane:

$$a_0 + b_0 j = r(\cos(\theta) + j\sin(\theta))$$

where $r = \sqrt{a_0^2 + b_0^2}$ and $\theta = \tan^{-1}(b_0, a_0)$

By Euler's formula:

$$a_0 + b_0 j = re^{j\theta}$$

Furthermore, we can express $(re^{j\theta})^n$ as $r^n e^{jn\theta}$. This is a complex number with magnitude r^n and angle $n\theta$.

Notes

Complex Poles

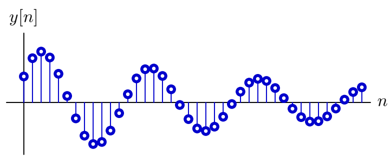
Complex poles, but real-valued response. This happens because poles come in complex conjugate pairs (summing $p_0^n + p_1^n$ yields a real number if p_0 and p_1 are complex conjugates).

The period of oscillation of the resulting real-valued signal is the same as the periods of the complex-valued signals!

Notes

Check Yourself!

Output of a system with poles at $z = re^{\pm j\omega}$



Which statement is true?

1. $r < 0.5$ and $\omega \approx 0.5$
2. $0.5 < r < 1$ and $\omega \approx 0.5$
3. $r < 0.5$ and $\omega \approx 0.08$
4. $0.5 < r < 1$ and $\omega \approx 0.08$
5. None of the above

Notes

Poles for Design

The **poles** of the system tell us something about how we expect it to behave in the long term.

By adjusting k, we change the poles of the system.

Our design problem can be thought of as choosing k to move the poles to a "desirable" location in the complex plane.

Notes

Summary

Feedback → cyclic signal flow paths

Cyclic paths → persistant responses to transient inputs

We can characterize persistent responses with poles

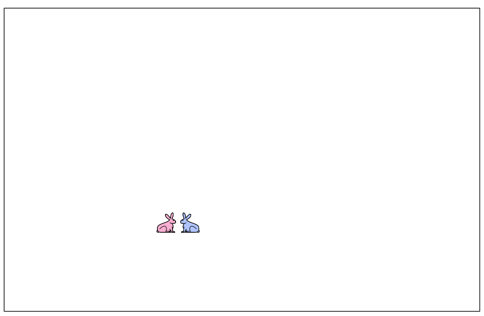
Poles provide a way to characterize the behavior of a system in terms of a mathematical description as a system functional

Poles provide a way to reason about the long-term behavior of a system

Powerful Representations (here polynomials) lead to **powerful abstractions** (e.g., poles)

Notes

Bunnies



Notes

Fibonacci!

```
>>> from functools import reduce
>>> fib=lambda n:reduce(lambda x,n:[x[1],x[0]+x[1]],range(n),[0,1])[1]
>>> fib(0)
1
>>> fib(1)
1
>>> fib(2)
2
>>> fib(3)
3
>>> fib(4)
5
>>> fs = [fib(i) for i in range(30)]
>>> fs[:12]
[1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144]
>>> fr = [j/i for i,j in zip(fs,fs[1:])]
>>> fr
[1.0, 2.0, 1.5, 1.6666666666666667, 1.6, 1.625,
1.6153846153846154, 1.619047619047619, 1.6176470588235294,
1.6181818181818182, 1.6179775280898876, 1.6180555555555556,
```


Notes

Bunnies Revisited

$$Y = \mathcal{R}Y + \mathcal{R}^2Y + X$$

$$\frac{Y}{X} = \frac{1}{1 - \mathcal{R} - \mathcal{R}^2}$$

$$\frac{Y}{X} = \frac{1}{1 - \frac{1}{z} - \frac{1}{z^2}}$$

$$\frac{Y}{X} = \frac{z^2}{z^2 - z - 1}$$

$$p0, p1 = \frac{1 \pm \sqrt{5}}{2}$$

Notes

Bunnies Revisited

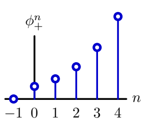
Recall that the USR of the composite system can be represented as:

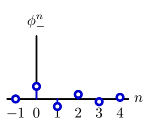
$$y[n] = \sum_i c_i p_i^n$$

Poles at:

$$\phi_+ = \tfrac{1+\sqrt{5}}{2} \approx 1.618 \qquad \phi_- = \tfrac{1-\sqrt{5}}{2} \approx -0.618$$

Two modes:





Notes

Bunnies Revisited

What if we want to find the response exactly?

$$y[n] = c_0(\phi_+^n) + c_1(\phi_-^n)$$

Two unknowns, and so need two equations.

$$y[0] = 1 = c_0(\phi_+^0) + c_1(\phi_-^0) = c_0 + c_1$$
$$y[1] = 1 = c_0(\phi_+^1) + c_1(\phi_-^1) = c_0\phi_+ + c_1\phi_-$$

Solving:

$$c_0 = \tfrac{1+\sqrt{5}}{2\sqrt{5}} \qquad c_1 = \tfrac{\sqrt{5}-1}{2\sqrt{5}}$$

$$fib(n) = \left(\frac{1+\sqrt{5}}{2\sqrt{5}}\right)\left(\frac{1+\sqrt{5}}{2}\right)^n + \left(\frac{\sqrt{5}-1}{2\sqrt{5}}\right)\left(\frac{1-\sqrt{5}}{2}\right)^n$$

$$\sqrt{5} \approx 2.2360679774997896964091736687312762354406183596115257242 7$$

Notes