18.03 Class 25, April 7, 2010

Convolution, or, the superposition of impulse response

1. General equivalent initial conditions for delta response
2. Two implications of LTI
3. Superposition of unit impulse responses
4. Convolution product

1. Equivalent for general delta-response (t > 0) :

p(D) x = a_n x^(n) + ... + a_0 x   (a_n not zero)

p(D) w = delta(t) with rest initial conditions specifies the
unit impulse response.

For t > 0 , this equation is equivalent (for t > 0) to

a_n x^(n) + ... + a_0 x = 0
with initial conditions   x(0) = ... = x^(n-2)(0) = 0 ,   x^(n-1)(0) = 1/a_n .

Ex: n = 2 : mx" + bx' + kx = delta(t) , rest initial conditions,
is equivalent to   mx" + bx' + kx = 0 ,   x(0) = 0 ,   x'(0) = 1/m .

[2] We work with a system modeled by an LTI operator p(D) = a_n D^n + ...
The time invariance has two important consequences:

(a) If p(D) x = q(t) , then p(D) x' = q'(t) .

This is because D p(D) = p(D) D (because the coefficients are constant)
so p(D) x' = p(D) Dx = D p(D) x = D q = q' .

In particular, since u'(t) = delta(t), v'(t) = w(t) :
the derivative of the unit step response is the unit impulse response.

e.g. D v = D (u(t) (1/k) (1 - e^{-kt})) = e^{-kt} = w .
(Notice that v is continuous: there are no delta terms in its
generalized derivative.)

(b) It doesn't really matter when you start the clock.

So, for example, we found that
the unit impulse response for D+kI is w(t) = u(t) e^{-kt}
and the unit step response is v(t) = u(t) (1/k) (1 - e^{-kt}) .

By time invariance the solution to x' + kx = delta(t-a) with x(t) = 0
for t < a is x = u(t-a) e^{-k(t-a)}) .

[3] We learn about a system by studying how it responds to various input
signals.

I claim that the unit impulse response   w(t)   contains complete
information about the LTI operator   p(D)   (and so about the system it
represents).

In fact there is a formula which gives the system response (with rest
initial conditions) to any input signal   q(t)   (with   q(t) = 0   for   t < 0)
as a kind of "product" of   w(t)   with   q(t) , namely   w(t)*q(t) .

Definition. Let   f(t) and g(t)   be such that   f(t) = 0 = g(t)   for   t < 0 .
The *convolution product* of   f(t)   and   g(t)   is

       f(t) * g(t) = integral_0^t f(t-tau) g(tau) d tau

I hope this crazy looking formula becomes sensible to you by the end of the
lecture!

Story:

Suppose phosphates from a farm run off fields into a lake, at a rate
q(t)   which varies with the seasons. For definiteness let's say

         q(t)   =   u(t) (1 + cos(bt))

Once in the lake, the phosphate load decays -- it's carried out of the stream
at a rate proportional to the amount in the lake:

     x' + ax =   q(t)   ,   x(0) = 0

We will find the solution to this differential equation in a clever way.

The unit impulse response for this system is

         w(t)   =   u(t) e^{-at}

This tells you how much of each pound is left after t units of time have
elapsed. If   c   pounds go in at time   0 , then   w(t) c   pounds remain
at time   t > 0 . More generally (by time invariance)
if   c   pounds go in at time   tau , then   w(t-tau) c   pounds remain
at time   t > tau.

More realistically, let   h   be a small time interval. If   q(tau) h   pounds
go in between time   tau   and time   tau + h ,   then   w(t-tau) q(tau) h
pounds remain at time   t > tau.

Now fix a time   t . We'll think about how   x(t)   gets built up from the
contributions made during the time between   0   and   t .

Break time up into small intervals each of width   h , the "step size."

In the first interval of time, between   t = 0   and   t = h , approximately

     q(0) h

pounds of phosphate gets dumped into the lake.

By time t this has decayed to the amount

    w(t) q(0) h

In the next time interval, between t = h and t = 2h , approximately

    q(h) h

pounds of phosphate gets added, and by time t it has decayed to

    w(t-h) q(h) h

In the next interval we have

    q(2h) h

which at time t has been reduced to

    w(t-2h) q(2h) h

This continues (assuming t is a multiple of h) up to time   t - h,
which yields at time t :

    w(h) q(t-h) h

Add these up:

   w(t) q(0) h + w(t-h) q(h) h + w(t-2h) q(2h) h + ... + w(h) q(t-h) h

Take the limit as h --> 0 : this is a Riemann sum approximating the
integral

     x(t) = integral_0^t w(t-tau) q(tau) d tau   (#)

That's the formula. For us,

     x(t) = integral_0^t e^{-a(t-tau)} (1 + cos(b tau)) d tau

The unit impulse response serves as a weight here - it tells you what
the long term impact of an action is.
"Weight function" = "Unit impulse response."

I used the "Convolution: Accumulation" tool with signal f(t) = 1 + cos(bt)
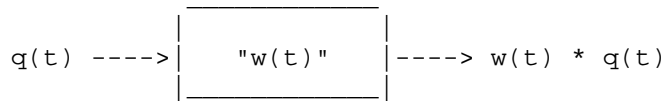and weight function (written g(t) in the tool) u(t)e^{-at} .

Restatement: Suppose w(t) is the weight function of p(D) . Then the
solution of p(D)x = q(t) satisfying rest initial conditions is
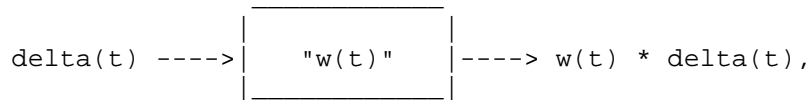
     x(t) = integral_0^t w(t-tau) q(tau) d tau

This works for any order. In the second order case, you think of the
input signal as made up of many little blows, each producing a decaying
ringing into the future. They get added up by superposition, and this is
the convolution integral. It is sometimes called the superposition
integral. It's the superposition of unit impulse responses.

You learn about a system by studying how it responds to input signals.
The interesting thing is that just one single system response suffices
to determine the system response to any signal at all. This is based on
two major assumptions: Linear, and Time Invariant.

[4] This means we can represent the system by its unit impulse response.
In terms of an input/output diagram (where rest initial conditions are assumed)

```
                    _____
                   |             |
    q(t) ---->|      "w(t)"   |----> w(t) * q(t)
                   |_____|
```

In particular,

```
                    _____
                   |             |
   delta(t) ---->|      "w(t)"   |----> w(t) * delta(t),
                   |_____|
```

but also by definition the output is w(t) : w(t) * delta(t) = w(t).

This * is NOT just the product w(t)f(t) . Nevertheless it deserves
to be called a "product." For one thing, it is associative and commutative:

        (f(t) * g(t)) * h(t) =   f(t) * (g(t) * h(t))

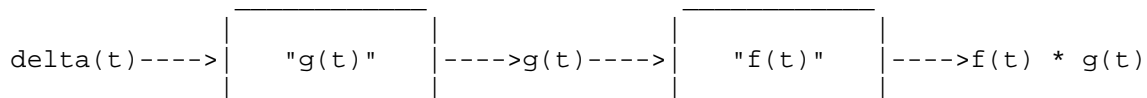   f(t) * g(t) =   g(t) * f(t)

an distributive:

    f(t) (c_1 g_1(t) + c_2 g_2(t)) =   c_1 f(t) * g_1(t) + c_2 f(t) * g_2(t)

Also   f(t) * delta(t) = f(t) shows that the delta function serves as a
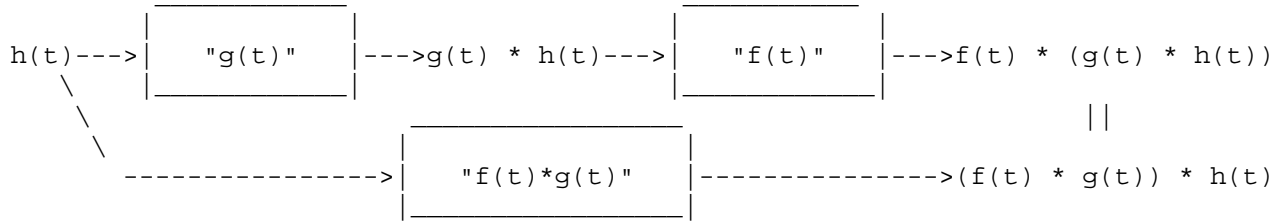"unit" for this "multiplication."

The book carries out the integration manipulation you need to do to see
these facts. Here's a proof of associativity using systems and signals:
Think about feeding the output of the "g(t)" system into the "f(t)" system:

```
               _____                    _____
              |             |                   |             |
    ---->|    "g(t)"    |------------->|    "f(t)"    |---->
              |_____|                   |_____|
```

This is a new system, a "composite" system. What is its unit impulse response?

```
               _____                    _____
              |             |                   |             |
 delta(t)---->|    "g(t)"    |---->g(t)---->|    "f(t)"    |---->f(t) * g(t)
              |_____|                   |_____|
```

so an input of h(t) gives an output of (f(t) * g(t)) * h(t) .
On the other hand,

```
                  _____                      _____
              |              |                    |             |
h(t)--->|     "g(t)"   |--->g(t) * h(t)--->|    "f(t)"   |--->f(t) * (g(t) * h(t))
     \      |_____|                    |_____|
      \                                                                ||
       \                   _____
        ------------------>|                |                        ||
                           |   "f(t)*g(t)"  |--------------->(f(t) * g(t)) * h(t)
                           |_____|
```

18.03 Differential Equations
Spring 2010