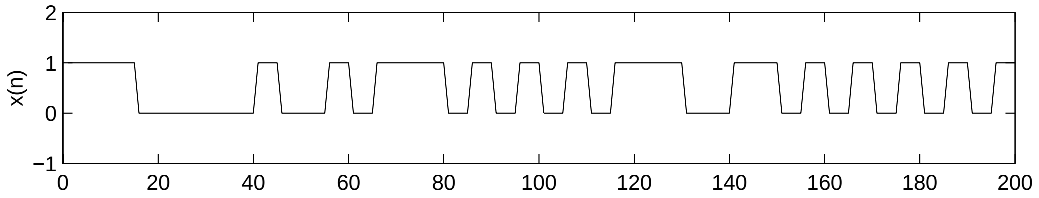
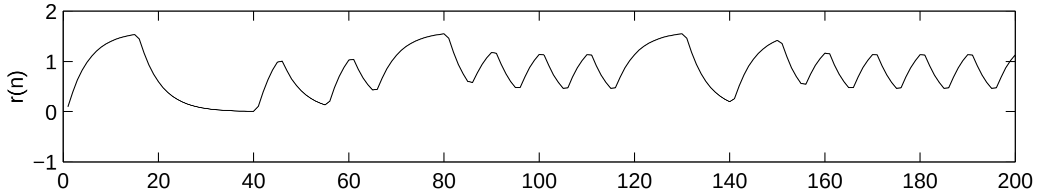
Notes on Stable Inverses of Discrete-Time LTI Systems

Suppose the signal $x(n)$ represents binary data that is to be transmitted.



However, the channel over which $x(n)$ is transmitted introduces some distortion. The signal $r(n)$ that is received is changed.



In this example, the channel can be modeled as an LTI system described by the difference equation

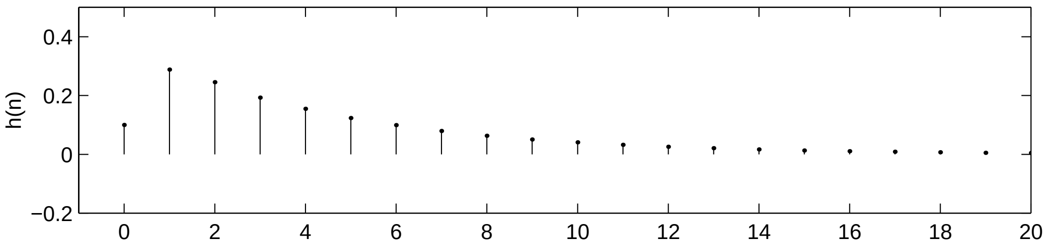
$$r(n) - 11\, r(n-1)/20 - r(n-2)/5 = x(n)/10 + 7\, x(n-1)/30 + 2\, x(n-2)/30$$

where $r(n)$ represents the received signal. Equivalently, the channel can be modeled by the transfer function

$$H(z) = \frac{1}{10} \cdot \frac{1 + \frac{7}{3} z^{-1} + \frac{2}{3} z^{-2}}{1 - \frac{11}{20} z^{-1} - \frac{1}{5} z^{-2}}, \tag{24}$$

$$H(z) = \frac{1}{10} \cdot \frac{z^2 + \frac{7}{3} z + \frac{2}{3}}{z^2 - \frac{11}{20} z - \frac{1}{5}}, \tag{25}$$

or equivalently by its impulse response:



Define

$$G(z) := \frac{1}{H(z)}. \tag{26}$$

Then

$$G(z) = 10 \cdot \frac{z^2 - \frac{11}{20}z - \frac{1}{5}}{z^2 + \frac{7}{3}z + \frac{2}{3}}. \tag{27}$$

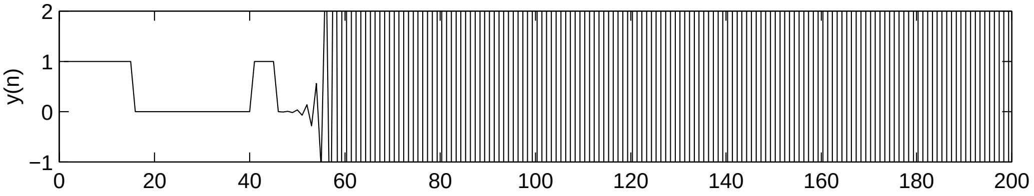
A causal system with the transfer function $G(z)$ can be implemented by the difference equation

$$y(n)/10 + 7\,y(n-1)/30 + 2y(n-2)/30 = r(n) - 11\,r(n-1)/20 - r(n-2)/5$$

where $r(n)$ represents the **input** to the system and $y(n)$ represents the **output** of the system. This system can be implemented in a causal fashion by writing the difference equation as

$$y(n) = 10\,r(n) - 11\,r(n-1)/2 - 2\,r(n-2) - 7\,y(n-1)/3 - 2\,y(n-2)/3.$$

However, when this is done, the signal $y(n)$ that you obtain is:



Because the causal implementation of this difference is **unstable**, the signal blows up after a short while. In order to obtain a **stable** inverse for the system $H(z)$, we need to be more careful. Let us find a partial fraction expansion of $G(z)/z$

$$\frac{G(z)}{z} = \frac{10z^2 - \frac{11}{2}z - 2}{z^3 + \frac{7}{3}z^2 + \frac{2}{3}z} \tag{28}$$

We can use Matlab to help with the partial fraction expansion.

```
>> [r,p,k] = residue([10 -11/2 -2],[1 7/3 2/3 0])
r =
   14.7000
   -1.7000
   -3.0000
p =
   -2.0000
   -0.3333
         0
k =
     []
```

Therefore, we have

$$\frac{G(z)}{z} = \frac{14.7}{z+2} + \frac{-1.7}{z+1/3} + \frac{-3}{z} \tag{29}$$

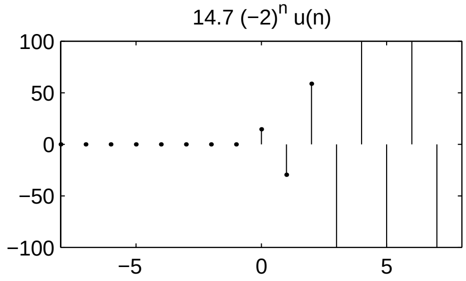
or

$$G(z) = \frac{14.7\,z}{z+2} + \frac{-1.7\,z}{z+1/3} - 3 \tag{30}$$

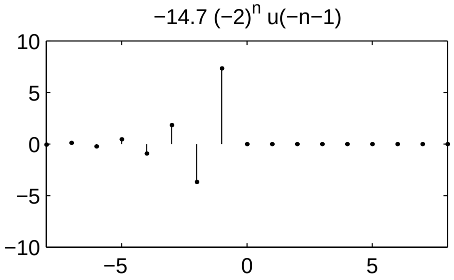
and

$$g(n) = \left\{ \begin{array}{c} 14.7\,(-2)^n\,u(n) \\ \text{or} \\ -14.7\,(-2)^n\,u(-n-1) \end{array} \right\} + \left\{ \begin{array}{c} -1.7\,(-1/3)^n\,u(n) \\ \text{or} \\ 1.7\,(-1/3)^n\,u(-n-1) \end{array} \right\} - 3\,\delta(n).$$

Consider the first component — the causal signal $14.7\,(-2)^n\,u(n)$ grows without bound:

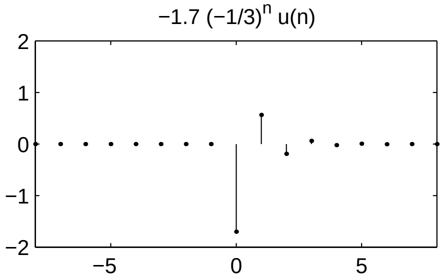


On the other hand, the noncausal signal $-14.7\,(-2)^n\,u(-n-1)$ decays:

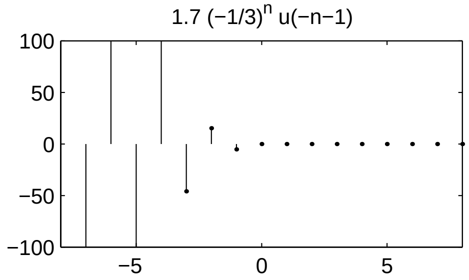


To obtain a stable system, we must choose the noncausal form for this component.

Consider the second component — the causal signal $-1.7\,(-1/3)^n\,u(n)$ decays:



On the other hand, the noncausal signal $1.7\,(-1/3)^n\,u(-n-1)$ grows without bound:

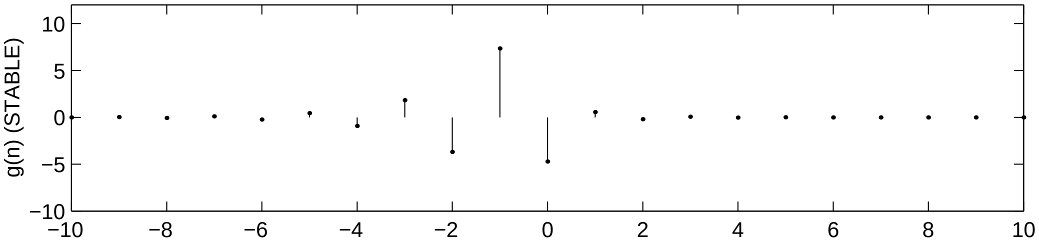


To obtain a stable system, we must choose the causal form for this component.

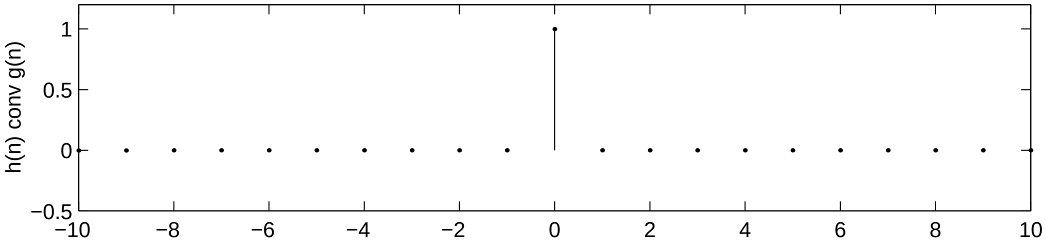
Therefore, to obtain a stable system, the impulse response $g(n)$ must be given by

$$g(n) = -14.7\,(-2)^n\,u(-n-1) - 1.7\,(-1/3)^n\,u(n) - 3\,\delta(n). \tag{31}$$

This impulse response $g(n)$ is shown in the figure.



For an inverse system, it is required that $h(n) * g(n) = \delta(n)$. We can verify this by numerically convolving the two functions.



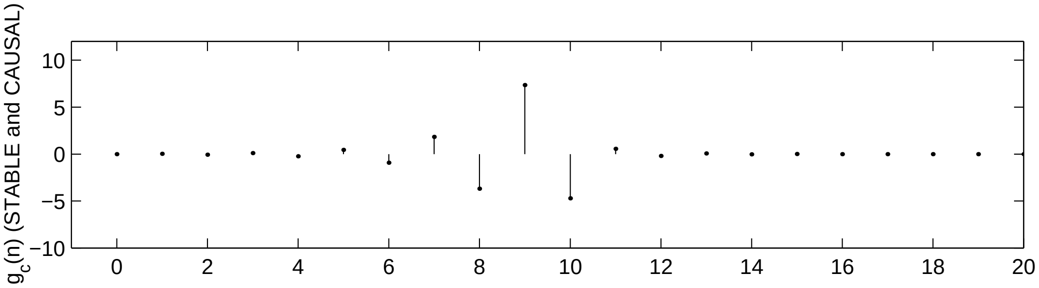
We can now recover $x(n)$ from $r(n)$ by using the LTI system with impulse response $g(n)$. Note that even though $g(n)$ is noncausal, for $n < -10$, the values of $g(n)$ are very small and for practical purposes can be set to zero. If they are set to zero, then the shifted signal $g(n-10)$ will be causal. So you can define a causal system with impulse response $g_c(n)$:

$$g_c(n) := \begin{cases} g(n-10) & n \geq 0 \\ 0 & n < 0 \end{cases} \tag{32}$$

or more compactly as

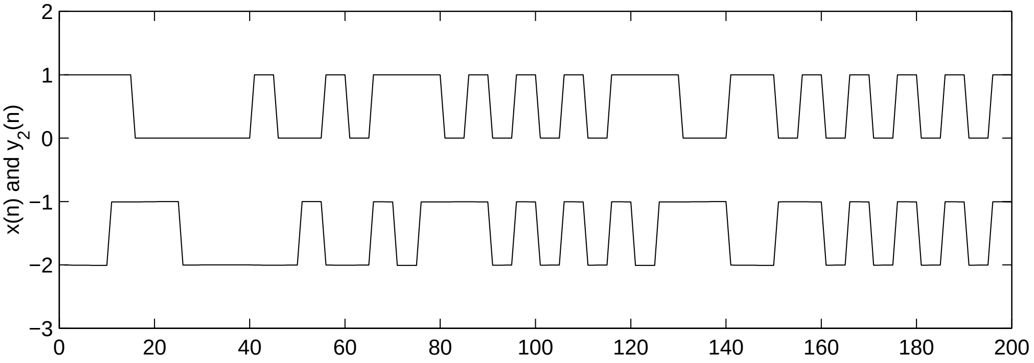
$$g_c(n) := g(n-10)\,u(n). \tag{33}$$

This is shown in the figure:



The signal $g_c(n)$ is a good approximation to $g(n)$ because the values of $g(n)$ that are lost due to truncation are close to zero.

Using the causal stable system $g_c(n)$ to filter the received signal $r(n)$ gives:



The top signal is $x(n)$. The lower signal is $y_2(n)$, obtained by convolving $r(n)$ with $g_c(n)$. Notice that $y_2(n)$ is almost the same as $x(n)$ except for a delay of 10 samples.

Therefore, we can get a stable near-inverse of the channel if we allow for some delay.

The following Matlab code, used for this problem, can be found on the course webpage.

```
% generate a data signal (don’t worry about understanding this command)

x = kron(round(rand(1,40)),ones(1,5));

% ------ Plot data signal ------
figure(1)
subplot(4,1,1), plot(x), ylabel(’x(n)’)
axis([0 200 -1 2])

% ------ Define System I ------
b = [1 +7/3 2/3]*(1/10);
a = [1 -11/20 -4/20];

r = filter(b,a,x);
```

```
% r(n) is the distorted version of x(n)

subplot(4,1,2), plot(r), ylabel('r(n)')

y = filter(a,b,r);

% y(n) is what you get when you try the "simple" thing.
% The "simple" thing is to reverse the difference equation.

subplot(4,1,3), plot(y), ylabel('y(n)')

% --> the "simple" thing is unstable!


im = [1 zeros(1,100)];   % define impulse signal
h  = filter(b,a,im);     % impulse response of System I

% ------ Let G(z) = 1/H(z) ------

% ------ Compute partial fraction expansion of G(z)/z ------

[R,P,K] = residue([10 -11/2 -2],[1 7/3 2/3 0]);
%
% R =
%    14.7000
%    -1.7000
%    -3.0000
%
% P =
%    -2.0000
%    -0.3333
%          0
%
% K =
%      []

% so g(n) = -14.7 (-2)^n u(-n-1) - 1.7 (-1/3)^n u(n) - 3 delta(n)

% ------ create stable signal g(n) corresponding to G(z) ------

n = -10:10;
g1 = -14.7 .* (-2).^n .* (-n-1>=0);
g2 = -1.7 .* (-1/3).^n .* (n>=0);
g3 = -3*(n==0);
g = g1 + g2 + g3;

% ------ plot h(n) and stable g(n)  ------
% ------ also plot h(n) convolve g(n) ----

figure(2)
subplot(4,1,1), stem(h,'.'),  ylabel('h(n)')
subplot(4,1,2), stem(g,'.'),  ylabel('stable g(n)')
chck = conv(h,g);
subplot(4,1,3), stem(chck,'.'), ylabel('h(n) conv g(n)')

% ------ send r(n) through the system G(z) ------
```

```
y2 = conv(r,g);
figure(1)
subplot(4,1,4), plot(y2); ylabel('y_2(n)')
axis([0 200 -1 2])

% ------ plot x(n) and recovered signal y2(n) ----
% ------ on same graph ------

figure(3)
k = 1:200;
plot(k,x(k),k,y2(k)-2)
axis([0 200 -3 2])
ylabel('x(n) and y_2(n)')

% notice that y2(n) is a delayed version of x(n)
% ---> with a delay we can get x(n) back from y(n) using
%      a STABLE system.
```