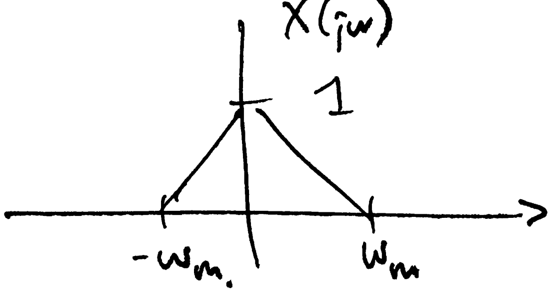
6.003 Recitation 20

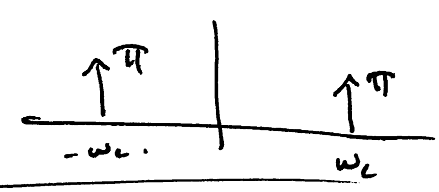
Russ Tedrake

Today: Modulation

Basic Idea is simple:



$c(t) = \cos(\omega_c t)$.



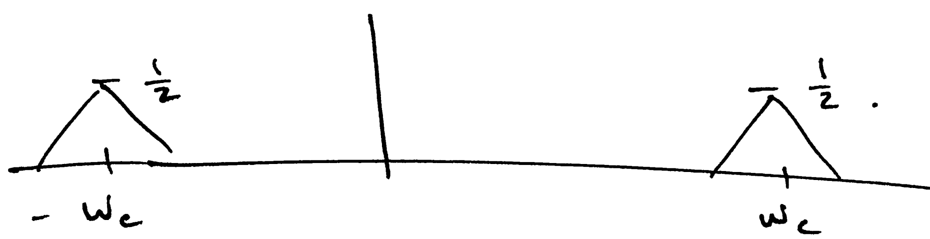
$a(t) \cdot b(t)$.

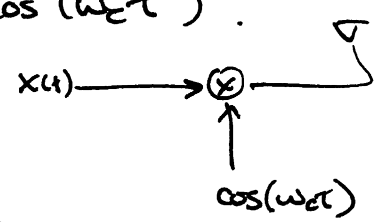
$\Updownarrow$

$\frac{1}{2\pi}\int A(j\omega) B(j(\omega-\omega_1))d\omega$

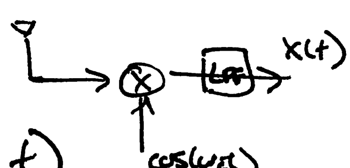
$x(t) \cdot e^{j\omega_c t}$.

or $x(t)\cos(\omega_c t)$.

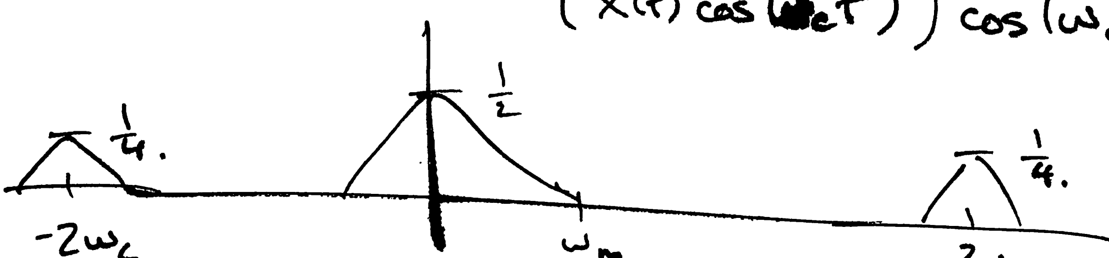




then to demodulate.



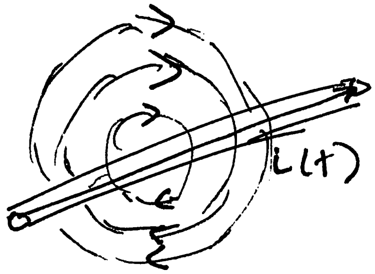
$[x(t)\cos(\omega_c t)]\cos(\omega_c t)$.



low-pass filter and ×2.

Example using demodulation + sampling (DT processing of CT signal) on real data.

Recall my powerline estimation problem.



$i(t) = A\cos(\omega_c t)$ alternating current on a wire.

generates a magnetic field.

$$b(t) = \frac{\mu_0 \, i(t)}{2\pi r}$$

μ_0 ← constant of permeability.

r ← radius.

Estimation problem:

magnetic field sensor on moving airplane.

Measures $b(t)$. (and knows A).

Can we reconstruct r? moving $\Rightarrow r(t)$.

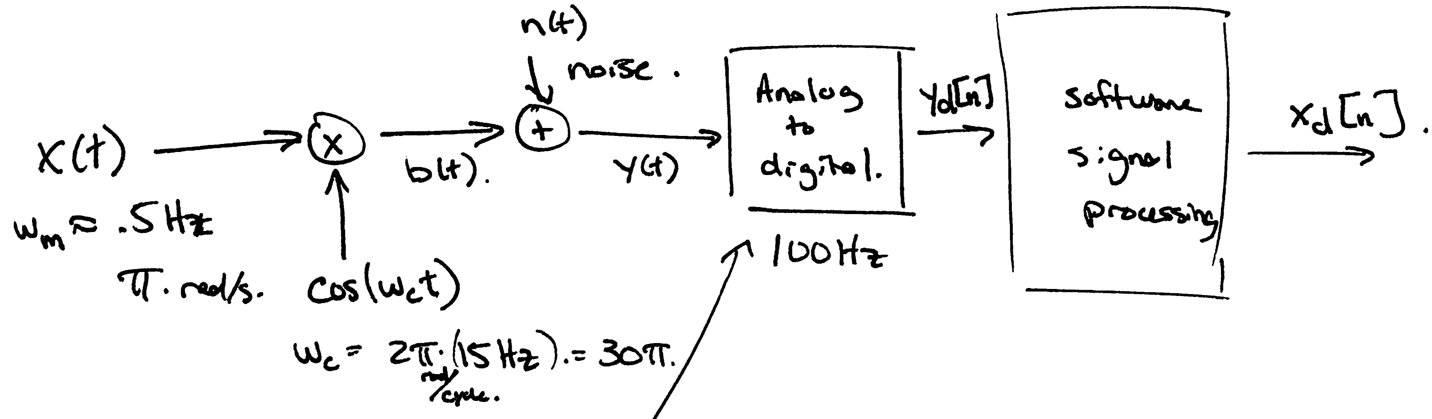
Change variables $x(t) = \frac{1}{r(t)}$.

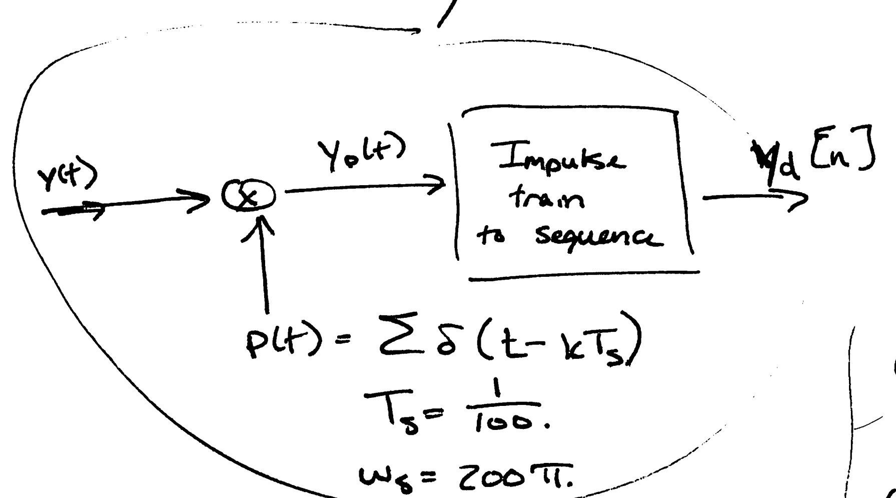
then $b(t) = \alpha \, x(t) \cos(\omega_c t)$.

α ← const.

So estimation requires demodulation!

Powerline estimation.





Q: is w_c large enough?

Q: is w_s large enough?

Q: what if $w_c = 160\pi$?

Our task is to design the software to estimate $x(t)$.

On the projector:

- Show $y_d[n]$.

Q: What should we do first? Filter yd? or demodulate?

A: filter first. otherwise demod can alias your noise onto your signal.

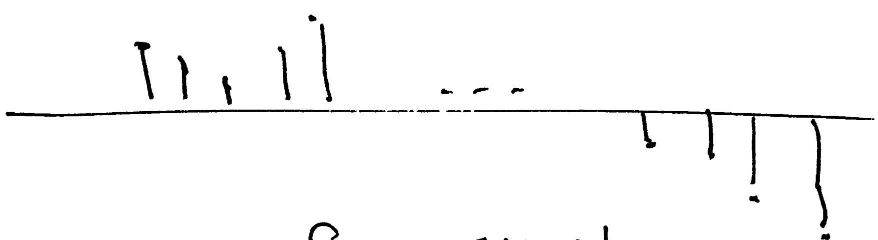
Let's look a frequency content of $y_d[n]$.

Q: Do I want DTFS or DTFT?

[Q: which does the FFT algorithm give me?
A: DTFS.

A: Not totally clear. FFT assumes a periodic extension of your finite data.

Can be ugly:



If low freq. signal happens to start & end in different places, DTFS can add high frequency errors.

...

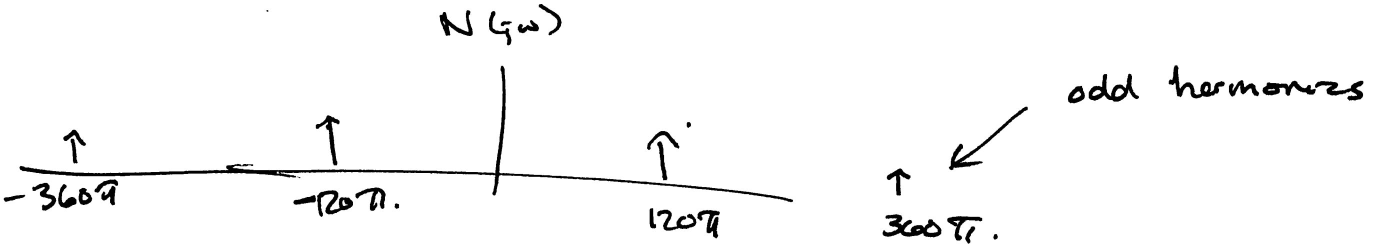
On projector: show a_k from DTFS of $y_d[n]$.

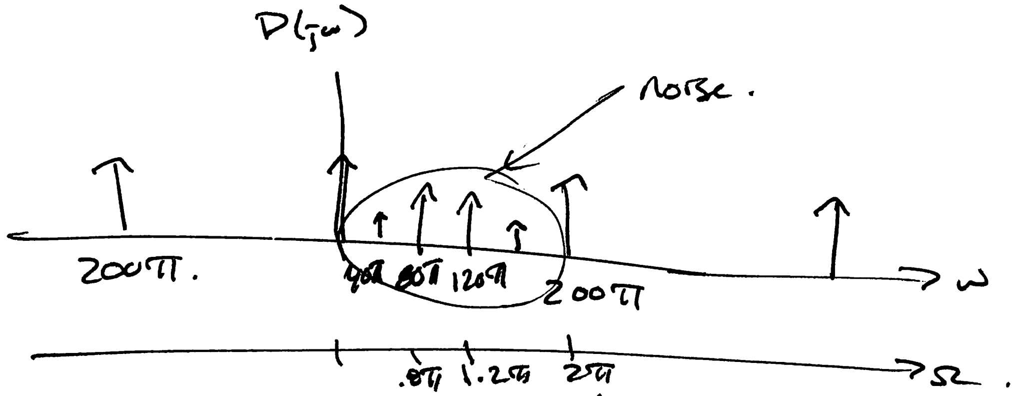
approx. DTFT (scale samples by 2π, x-axis by $\frac{2\pi}{N}$, $\Omega = \omega T$.).

~~Q: when~~

Q: Where is the 60Hz noise?

$n(t) = n_0 \cos(120\pi t)$.





On projector: see 60Hz noise.

Design. Band-pass butterworth filter.

use 4 poles.

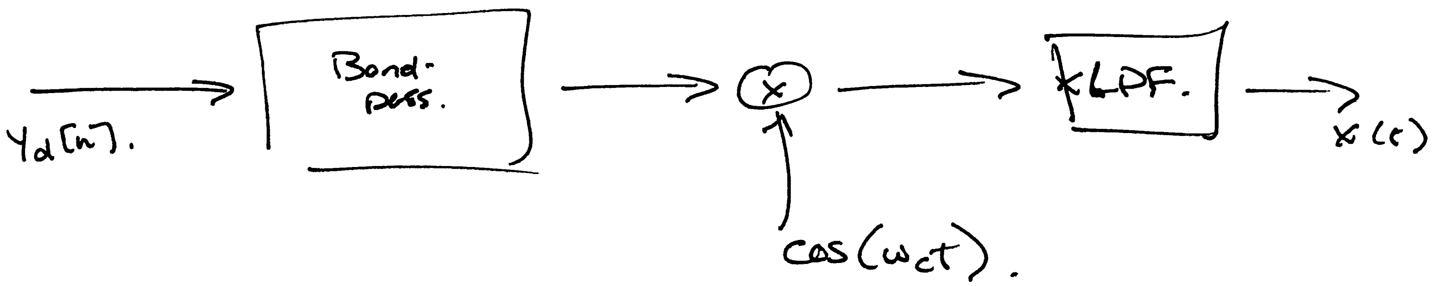
Q: Why not more?

A: Poles add phase. want to use this estimate in a feedback loop.

On: projector:

Show butterworth, poles + zeros, filtered data.

Now demodulate.

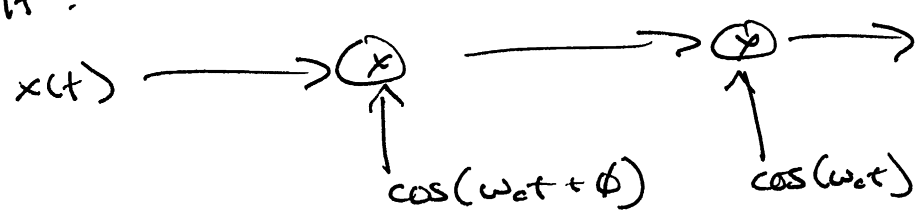


On projector: Looks close, but off by a scale factor.

Q: What went wrong?

A: I assumed synchronous demodulation, but don't actually know the phase of $i(t)$.

result.



$$x(t) \cos(\omega_c t + \phi) \cos(\omega_c t).$$

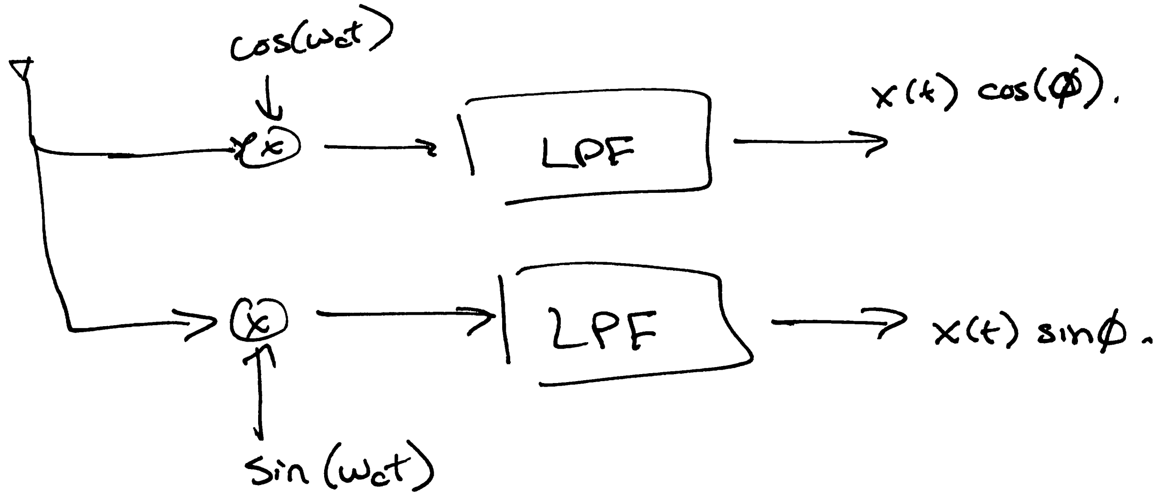
$$= x(t) \left[\tfrac{1}{2} \cos \phi + \tfrac{1}{2} \cos(2\omega_c t + \phi) \right].$$

↑ off by $\cos \phi$.

The fix?

adding the carrier + envelope detection doesn't work here. [I don't get to change the modulation].

Quadrature demodulation.



combine w/

$$\sqrt{x^2(t)\cos^2\phi + x^2(t)\sin^2\phi} = |x(t)|.$$

Can also estimate ϕ this way.

Since ϕ is constant, not $\phi(t)$,

use least-squares estimator.

which you'll learn about in 6.011.

And it works!

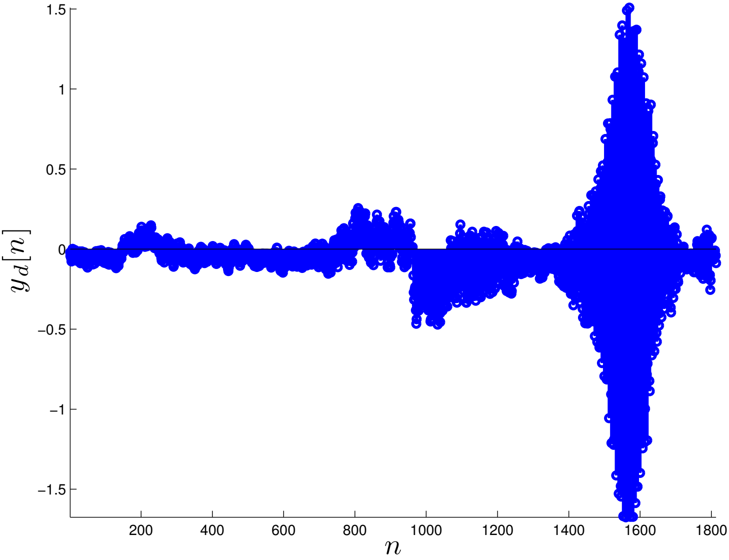
Airplane can land on a power line.

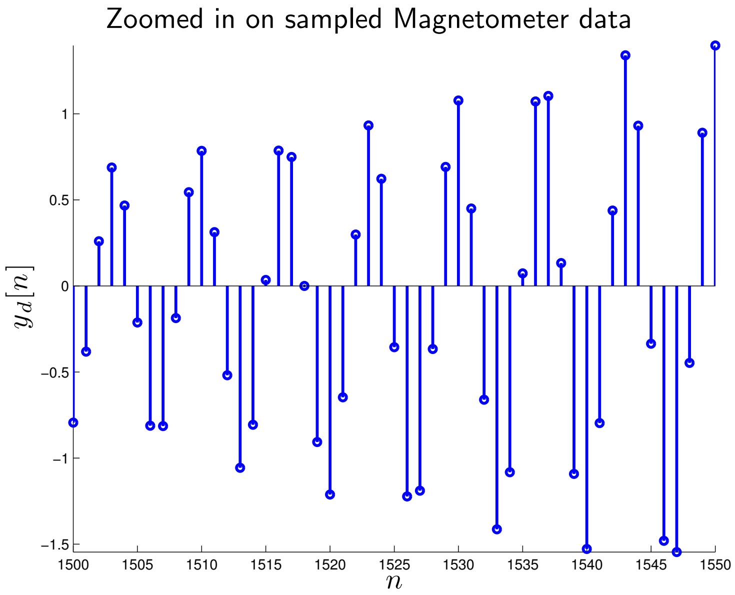
Example: Demodulation for Powerline Perching

6.003 Fall 2011, Recitation 20

December 2, 2011

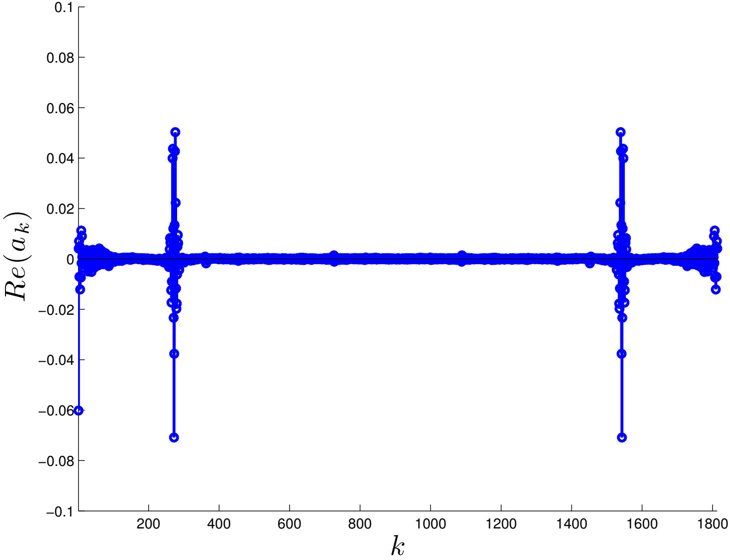
18 seconds worth of sampled Magnetometer data



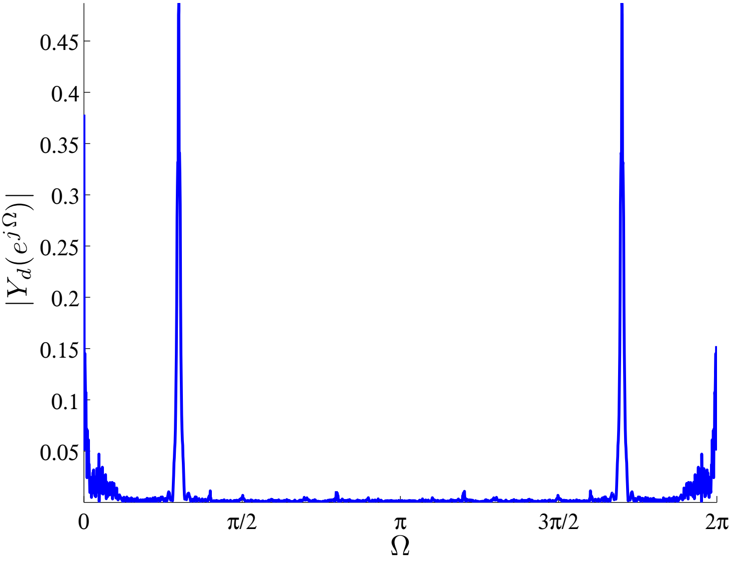


Zoomed in on sampled Magnetometer data
1
0.5
0
−0.5
−1
−1.5
$y_d[n]$
1500
1505
1510
1515
1520
1525
1530
1535
1540
1545
1550
n

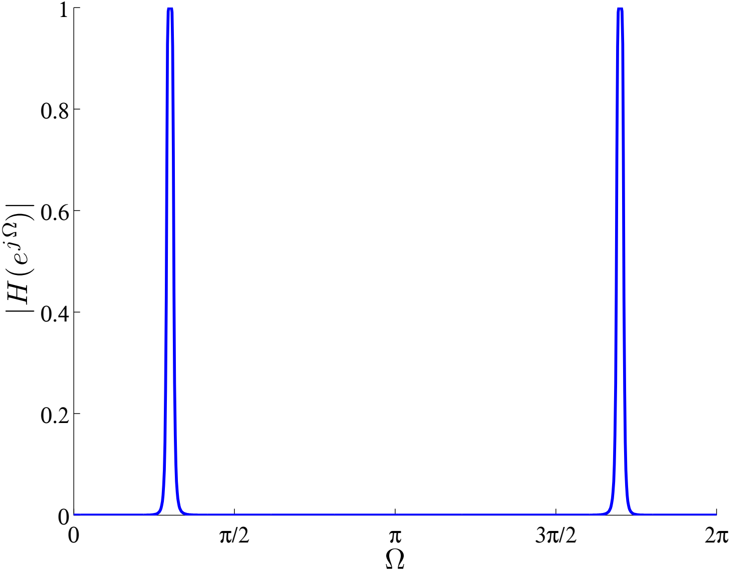
DTFS (via FFT algorithm) of sampled Magnetometer data



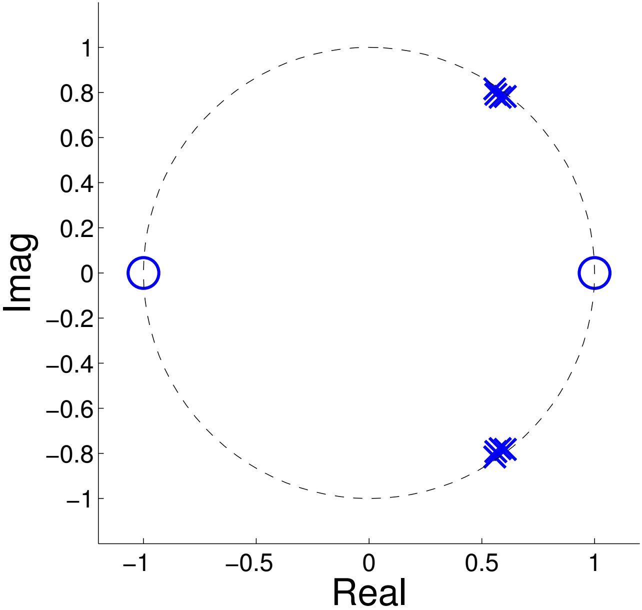
DTFT reconstruction from DTFS



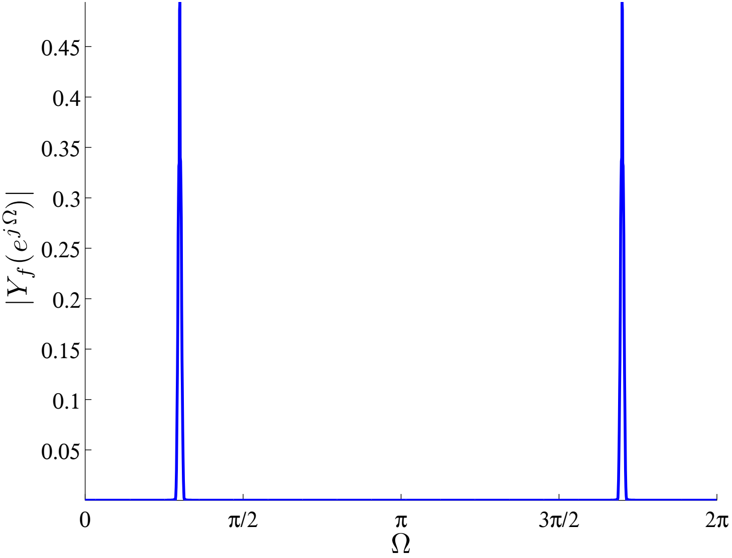
Band-pass Butterworth Filter (4 poles, Pass-band 15 Hz ± .5 Hz)

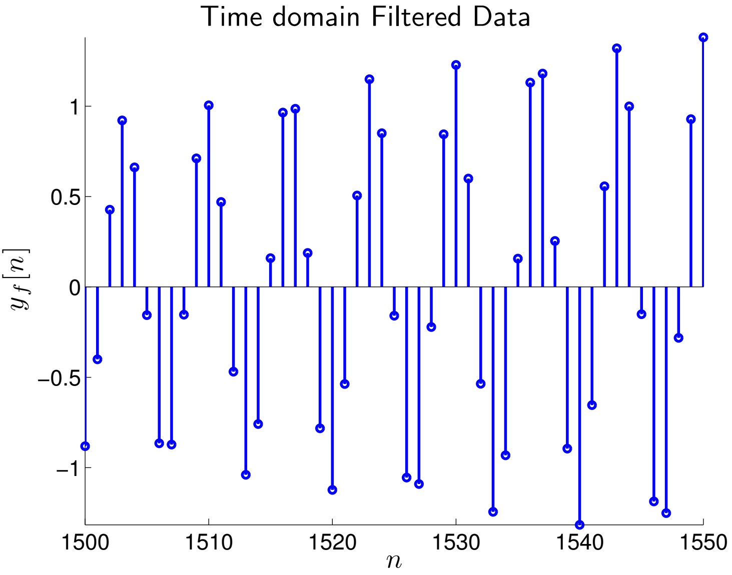


Pole/Zero Diagram of DT Butterworth Filter

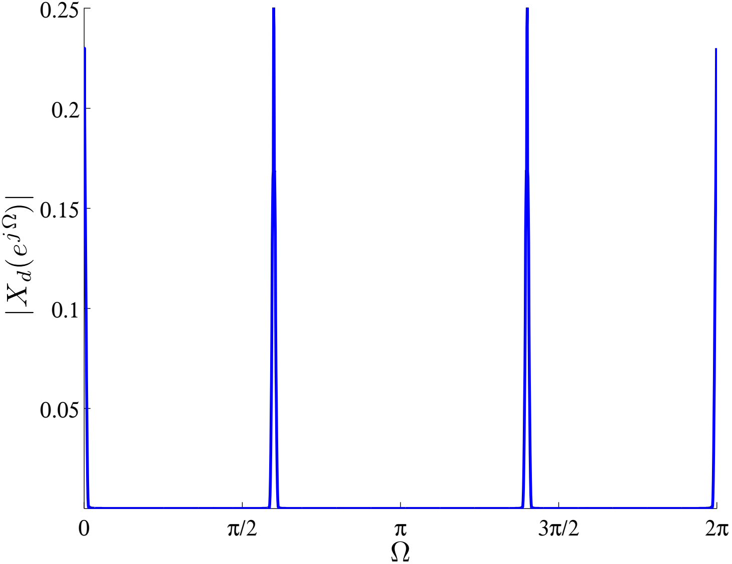


DTFT of Filtered Data

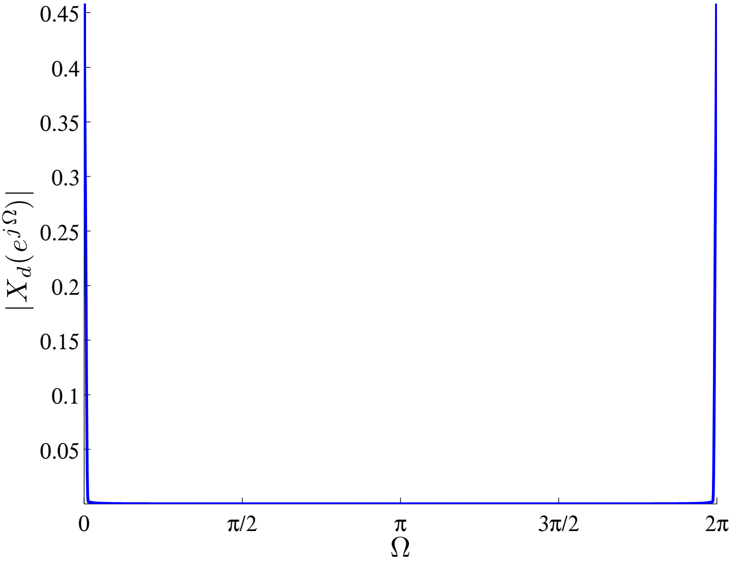



Time domain Filtered Data
1
0.5
0
−0.5
−1
$y_f[n]$
1500
1510
1520
1530
1540
1550
n

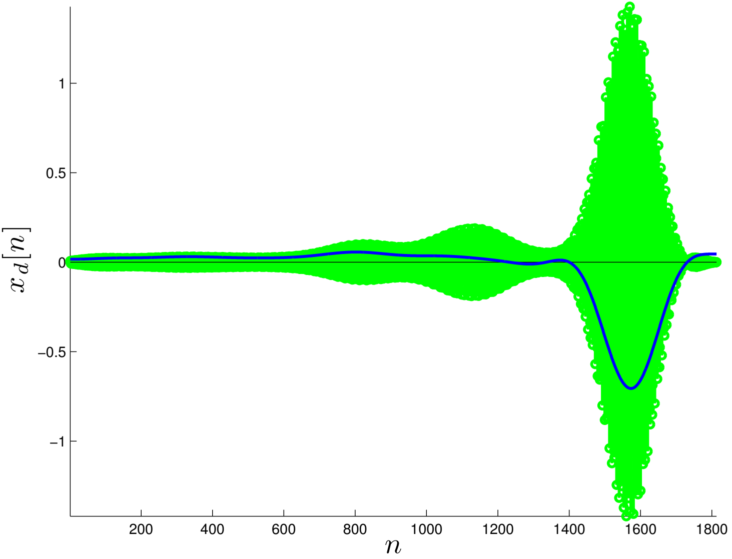
$x_d(t) = y_f(t)cos(\omega_c t)$



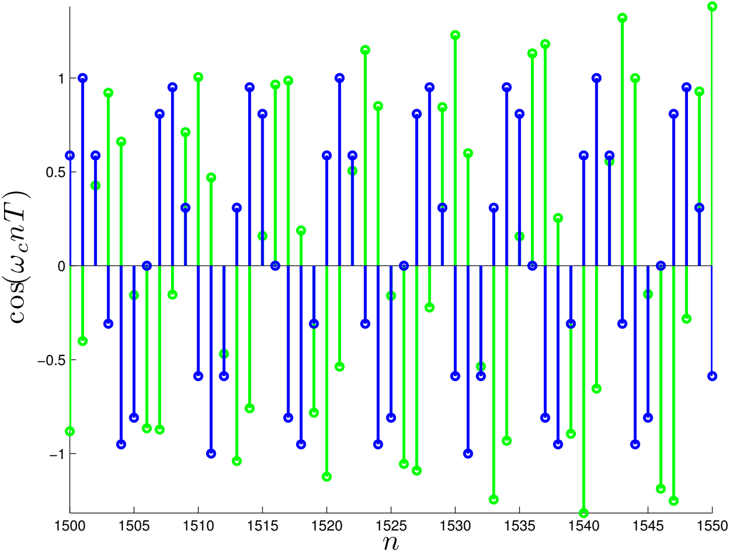
Finish demodulation with low-pass filter



Time domain estimate of $x_d[n]$



$\cos(\omega_c t)$ and $y_d[n]$ have a phase difference



After Quadrature Asynchronous Demodulation